IP COMMUNICATIONS

Practical Cisco Unified Communications Security

Brett Hall, CCIE® No. 20774
Nik Smith

ciscopress.com
https://t.me/learningnets

Practical Cisco Unified Communications Security

Brett Hall, CCIE® R&S, Collaboration #20774
Nik Smith

Cisco Press
Hoboken, New Jersey

https://t.me/learningnets

@cisco_in_persian_channel

Practical Cisco Unified Communications Security

Brett Hall, Nik Smith

Copyright© 2021 Cisco Systems, Inc.

Cisco Press logo is a trademark of Cisco Systems, Inc.

Published by:
Cisco Press

All rights reserved. This publication is protected by copyright, and permission must be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. For information regarding permissions, request forms, and the appropriate contacts within the Pearson Education Global Rights & Permissions Department, please visit www.pearson.com/permissions.

No patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this book, the publisher and authors assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein.

ScoutAutomatedPrintCode

Library of Congress Cataloging-in-Publication Number: 2020916934

ISBN-13: 978-0-13-665445-2
ISBN-10: 0-13-665445-2

Warning and Disclaimer

This book is designed to provide information about the practice and methodologies used to secure a modern Cisco Unified Communications environment. Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied.

The information is provided on an "as is" basis. The authors, Cisco Press, and Cisco Systems, Inc., shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the discs or programs that may accompany it.

The opinions expressed in this book belong to the authors and are not necessarily those of Cisco Systems, Inc.

Trademark Acknowledgments

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Cisco Press or Cisco Systems, Inc., cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Special Sales

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at corpsales@pearsoned.com or (800) 382-3419.

For government sales inquiries, please contact governmentsales@pearsoned.com.

For questions about sales outside the U.S., please contact intlcs@pearson.com.

https://t.me/learningnets

@cisco_in_persian_channel

Feedback Information

At Cisco Press, our goal is to create in-depth technical books of the highest quality and value. Each book is crafted with care and precision, undergoing rigorous development that involves the unique expertise of members from the professional technical community.

Readers' feedback is a natural continuation of this process. If you have any comments regarding how we could improve the quality of this book, or otherwise alter it to better suit your needs, you can contact us through email at feedback@ciscopress.com. Please make sure to include the book title and ISBN in your message.

We greatly appreciate your assistance.

Editor-in-Chief: Mark Taub

Director, ITP Product Management: Brett Bartow

Executive Editor: Nancy Davis

Sponsoring Editor: Mudita Sonar

Managing Editor: Sandra Schroeder

Development Editor: Ellie Bru

Project Editor: Mandie Frank

Copy Editor: Chuck Hutchinson

Technical Editors: Paul Giralt, Christopher Hsu

Editorial Assistant: Cindy Teeters

Designer: Chuti Prasertsith

Composition: codeMantra

Indexer: Ken Johnson

Proofreader: Donna E. Mulder

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **www.cisco.com/go/offices.**

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

https://t.me/learningnets

@cisco_in_persian_channel

Credits

Figure 1-1 oOhyperblaster/Shutterstock

Figure 2-2 Kyryl Gorlov/123RF

Figure 2-4 The Wireshark team

Figure 2-22 © CYBERDATA CORPORATION 2020

Figure 2-23 Hywit Dimyadi/Shutterstock

Figure 10-6 Screenshot © 2020 Apple Inc

Figure 10-7 Screenshot © FileZilla

https://t.me/learningnets

@cisco_in_persian_channel

About the Authors

Brett Hall, CCIE R&S, Collaboration #20774, is a Customer Solutions Architect supporting Cisco's product and service offerings for enterprise customers across the U.S. Army and defense agencies. He has more than 20 years' IT experience and has worked with Cisco to support federal government initiatives for more than 13 years. He also works with Cisco business units to define product and service strategies and helps lead a global team of Cisco architects. To support customer requirements, Brett works with presales teams to engineer solutions that lead to customer adoption. Brett also drives solution development to help support future needs of his customers.

Nik Smith, Technical Leader for Collaboration at Cisco, supports Cisco Unified Collaboration (UC) products and service offerings for the public sector, enterprise, and defense agencies. His 24 years of networking experience cover technologies ranging from RF communications and telecommunications to UC. For 14 years, he has supported some of the world's largest Cisco UC deployments. During this time, he has led several large implementation teams involved in migrating from TDM to VoIP, along with network support and modernization. He now leads a team of UC engineers supporting the public sector, Department of Defense and providing guidance and mentoring to ensure that Cisco delivers best-in-class, highly secure UC capabilities.

https://t.me/learningnets

@cisco_in_persian_channel

About the Technical Reviewers

Paul Giralt, CCIE R&S, CCIE Voice, CCIE Collaboration #4793, is a Distinguished Engineer in the Customer Experience organization where he focuses on Cisco collaboration technologies and has been with Cisco since 1998. He spends much of his time helping customers accelerate the adoption of Cisco technologies and solutions and building services capabilities to provide more proactive and predictive services as well as improve product serviceability. Paul has spent his career at Cisco in the TAC, the Collaboration Technology Group, Solution Validation Services, Cisco Advanced Services, and most recently as part of the Customer Experience organization. He has spent years troubleshooting and diagnosing issues on some of the largest and most complex Cisco Collaboration deployments. Paul is passionate about the intersection of programmability and Cisco Collaboration products as well as making it easier for customers to diagnose issues on their own. He is also the author of the Cisco Press book *Troubleshooting Cisco IP Telephony* and a member of the Cisco Live Distinguished Speaker Hall of Fame Elite. He holds a degree in computer engineering from the University of Miami.

Christopher Hsu is a Solutions Architect in the Cisco Unified Collaboration domain. He has over 20 years of extensive hands-on experience in designing and implementing Cisco Unified Collaboration technologies. He has been with Cisco for 14 years and specializes in large-scale Unified Collaboration solutions. He has certifications in CCIE and CISSP.

https://t.me/learningnets

@cisco_in_persian_channel

Dedications

Writing a book for Cisco Press has always been a bucket list item for me. Despite all of the craziness in my day job, my wife understood this and allowed me to pursue my dreams of writing this book. I cannot thank my wife, Wendy, enough for the patience, understanding, and support that she has provided me during the writing process, but more importantly the past 20 years that we have been married. I also would like to thank my kids, Gabriel and Hannah. You understood the stress that was accumulating during those late nights and didn't hold it against me as I raced toward the finish line. I pray that this example will leave a lasting impression on you to always chase your dreams and to work hard in the process of doing so.

—Brett

I would like to thank my wife, Elizabeth, and kids, for putting up with me while I worked on this book. The immense patience they showed makes me even more grateful for having them in my life. I especially want to thank my wife for checking on the progress of each chapter I wrote. You're an amazing PM, and yes, now I can get started on the honey-do list you've been compiling over the last couple of months.

—Nik

https://t.me/learningnets

@cisco_in_persian_channel

Acknowledgments

Sometimes it is not what you know, but who you know and the support system that you have around you. I want to thank God for blessing me by putting me in a place in my career where I had such a tremendous team surrounding me with mentorship and engineering talent. I also would like to acknowledge the people who have been willing to help when I needed it most. Depending on whether I lacked perspective or knowledge in certain areas, the team around me has never hesitated to give me a hand.

Cliff Potts and Paul Giralt, you have both been great mentors to me and have helped me understand many things. This book would not be possible without either of you. Nick Russo, I remember having a conversation with you about this crazy idea that I had about UC Security. You helped connect me with Nik, who had a similar idea so that we could make this book a reality. Also, for those I managed to cajole into either helping me with my lab environment or providing a second set of eyes for my work—Dave Fusik, Baron Rawlins, Dan Keller, Hussain Ali, Kevin Roarty, Laurent Pham, Tony Mulchrone, Joe Tansey, Vernon Depee, Robbie Horgan, Jason Newman—I owe you guys one!

—Brett

Over the years I've been blessed to have the chance to work with numerous exceptional engineers who have helped me become the engineer that I am today.

Dan Gallagher, you are a mentor to me and one of the most productive, knowledgeable, and prepared people I have ever had the pleasure of knowing.

Dave Duncan, you are and always have been that calm, reassuring voice at 2 a.m. in the middle of a change going south. You've kept us on track and even-keeled many a time with your demeanor and in-depth knowledge of UC and data networking expertise.

Bill Donaldson, among the engineers I've worked with, I'm pretty sure I've worked with you the longest. Thank you for being able to provide an alternative point of view to a design or solution to a problem.

For this book specifically I was very lucky to be able to work with Paul Giralt and Chris Hsu as my technical editors. You guys were brutal, and I hope I did your comments justice. I also want to thank Laurent Pham and James Arias for giving me a second set of eyes on a couple of the chapters. It would have been a lot more difficult without your inputs.

Lastly, I want to give a special thanks to Franklin Hall, who got me interested in VoIP so many years ago.

—Nik

https://t.me/learningnets

@cisco_in_persian_channel

Contents at a Glance

Introduction xix

Chapter 1 The Importance of Practical UC Security 1

Chapter 2 Physical Security and Life Safety 15

Chapter 3 Security Through Network Fundamentals 55

Chapter 4 Maintaining Security Across UC Deployment Types 89

Chapter 5 Hardening the Core Cisco UC Appliance Operating Systems 125

Chapter 6 Core Cisco UC Application Lockdown 161

Chapter 7 Encrypting Media and Signaling 217

Chapter 8 Securing Cisco Unified Communications Manager (Cisco) 273

Chapter 9 Securing Cisco Unity Connection 305

Chapter 10 Securing Cisco Meeting Server 339

Chapter 11 Securing the Edge 383

Chapter 12 Securing Cloud and Hybrid Cloud Services 427

Afterword 471

Index 475

https://t.me/learningnets

@cisco_in_persian_channel

Contents

Introduction xix

Chapter 1 The Importance of Practical UC Security 1

Identifying the Threat Landscape 2
The Danger of Shadow IT 4
Balancing Operations and Security 5
Minimizing Complexity 7
Visibility and Management 10
Introduction to ACME: Case Study 11
Summary 13
Additional Resources 14

Chapter 2 Physical Security and Life Safety 15

Introduction to Physical Security and Life Safety 15
A Physical Security Methodology 17
Preparation 17
Prevention 19
Detection 20
Response 21
Practical Physical Security for Your UC Environment 22
Physical Security for the Data Center 22
Power Plant Considerations 24
Electrostatic Discharge 25
Cable Plant Considerations 26
Life and Safety Considerations 28
Introduction to Enhanced 911 28
Terms and Acronyms 29
Regulatory Considerations 30
Native E911 Call Routing with Cisco Unified CM 31
E911 Call Routing with Cisco Emergency Responder 34
E911 Call Flow with Cisco Emergency Responder 35
E911 Design 36
ERL Creation and Network Discovery 38

https://t.me/learningnets

@cisco_in_persian_channel

Call Routing Considerations 43
PSAP Callback 47
Management, Verification, and Compliance 49
Additional Life and Safety Solutions 51
Computer-Aided Dispatch 52
Summary 53
Additional Resources 54

Chapter 3 Security Through Network Fundamentals 55
Introduction to Network Security 57
Segmentation 58
Micro Segmentation 59
Secure Network Access 64
Port Security 65
802.1x Authentication 67
MAC Authentication Bypass (MAB) and Network Access Control (NAC) 72
Security Features 75
VLAN Hopping 77
DHCP Snooping 78
ARP Inspection 80
NTP 80
DNS 83
Firewalls and Access Controls 84
Continuous Monitoring 86
Summary 86
Additional Resources 87

Chapter 4 Maintaining Security Across UC Deployment Types 89
Common Enterprise Collaboration Deployment Models and Security Considerations 90
An Overview of How to Secure Cluster Communications 96
NTP Authentication Enablement and Verification 100
Securing Intra-Cluster Signaling and Traffic 103
Securing the Signaling Traffic to IOS Voice and Analog Gateways 110
Securing the Integration with Cisco Emergency Responder 118
Enable Cisco Emergency Responder to Use Secure JTAPI 119
Summary 123
Additional Resources 124

https://t.me/learningnets
@cisco_in_persian_channel

Chapter 5 Hardening the Core Cisco UC Appliance Operating Systems 125

Defining the Core UC Applications 126

The UC Appliance Is Not a Standard Linux Server 127

Restricted and Unrestricted Versions of UC Software 134

Standard Practices for Patch/Version Management 136

How Root Access Is Granted 137

Hardening the Voice Operating System 138

Enabling Federal Information Processing Standard (FIPS) 140-2 139

Enabling Enhanced Security Mode 143

Enabling Common Criteria ISO/IEC 15408 Compliance 144

Summarizing FIPS 140-2 / Enhanced Security Mode / Common Criteria 147

Performing OS Lockdown via CLI 147

Process to Change Passwords for OS/GUI/Database 148

Configuring Password Aging 151

Enabling Password Complexity 152

Activating Account Locking and Inactive Account Disablement 155

Account Recovery Procedures 157

Summary 159

Additional Resources 159

Chapter 6 Core Cisco UC Application Lockdown 161

Types of Users in Cisco Unified Communications Manager and Cisco Unity Connection 162

Strengthening Local User Account Controls 163

Creating User Account Control Policies on Unified CM and Unity Connection 164

Using and Working with Cisco Unified CM Access Control Groups 170

Assigning User Roles and Credential Policies to Users 175

Importing End Users from a LDAP Directory 175

Enabling the Required Services for Importing End Users Using LDAP 176

LDAP Directory Configuration and Overview 177

Configuring LDAP Authentication for Imported End Users 184

Using Single Sign-On to Simplify the Login Experience 186

Intro to Security Assertion Markup Language (SAML) 186

Configuring Cisco Unified CM for SAML SSO 191

https://t.me/learningnets

@cisco_in_persian_channel

Synching End Users Between Unity Connection and Unified CM Using Universal PIN 197
Credential Change Service 200
Locking Down the GUI 201
Screen Timeout 201
Login Banner 202
Enabling System Monitoring Using SNMP and Syslog 204
Configurating and Using SNMP for System Monitoring 204
Defining the Alerting Types and Configuring Logging 209
Alarms 209
Audit Logs 211
Disaster Recovery Planning and Best Practices 213
Summary 214
Additional Resources 214

Chapter 7 Encrypting Media and Signaling 217

Licensing (Encryption License) and Allowing Export Restrictions Requirements 218
FIPS Considerations When Enabling Secure Signaling and Media Encryption 222
Public Key Infrastructure Overview 222
Utilizing Public Key Infrastructure with Cisco Unified Communications 229
Next-Generation Encryption Support Using Elliptical Curve Cryptography 233
IP Phone Certificates Types 235
TFTP File Encryption 237
Overview of the Endpoint Registration Process 239
Security by Default 239
What It Means to Place a Unified CM Cluster into Mixed Mode 245
CTL Files 247
Using SIP OAuth to Secure Signaling and Encryption 250
Why Is OAuth Used to Secure Signaling and Media? 251
Using SAML for Authentication with OAuth 252
Utilizing OAuth for Authorization 255

https://t.me/learningnets
@cisco_in_persian_channel

Enabling OAuth on Unified CM and Unity Connection 258
Configuring Secure Phone Profiles to Enable Secure Signaling and Media Encryption 261
Applying the Secure Phone Profiles and LSC to the Phones 264
Summary 271
Additional Resources 271

Chapter 8 Securing Cisco Unified Communications Manager (Cisco) 273
Endpoint Hardening 274
Where to Configure the Settings 274
Features and Services to Consider 275
Secure Conferencing 276
Ad Hoc Conferencing 278
Secure Conferencing Using Hardware-Based DSPs 278
Secure Conferencing Using Cisco Meeting Server 290
Meet-Me Secure Conferencing 297
Conference Now 298
Smart Licensing 298
Summary 302
Additional Resources 303

Chapter 9 Securing Cisco Unity Connection 305
Baseline Security Considerations Overview 306
Securing User Access to the Unity Connection 311
Securely Integrating Unity Connection with Unified CM 316
Integrating with Cisco Unified CM Securely 317
Applying Certificates Against the SIP Trunk Integration 318
Securing Messages 323
Preventing Toll Fraud in Unity Connection 325
Do Not Skip PIN Logins 325
Restriction Tables 325
Hardening Access to the TUI/GUI 330
TUI Voicemail Restricting Alternate Contact Numbers 330
System Transfers from Call Handlers and Nonsystem Numbers 333
Summary 336
Additional Resources 337

https://t.me/learningnets
@cisco_in_persian_channel

Chapter 10 Securing Cisco Meeting Server 339
CMS Overview and Deployment Modes 340
Operating System Hardening 342
Infrastructure Considerations 347
Securing CMS Connections 351
Database Security 354
Certificate Verification 356
Transport Layer Security 359
Certificate Assignment 360
Application Programming Interfaces (APIs) 367
Inbound and Outbound Calling 370
Unified CM Configuration 374
Securing CMS Spaces 377
Management and Visibility 377
Summary 381
Additional Resources 382

Chapter 11 Securing the Edge 383
Business Requirements for the Collaboration Edge 383
Security Considerations 383
Cisco's Collaboration Edge Architecture 384
IP-Based PSTN Access 386
Deploying CUBE 388
Toll Fraud Protection 390
CUBE-Based TDoS Protection 391
Session Control and Protection 392
Enabling CUBE for TLS Connectivity 395
VPN-Based Solutions 402
VPN-less Solutions 402
Business-to-Business Communication 403
Security Features Within Expressway 403
Deploying Mobile and Remote Access 406
DNS 407
Certificate Requirements for Mobile and Remote Access 410
Firewall Traversal 412
Authentication and Authorization for MRA 413

https://t.me/learningnets
@cisco_in_persian_channel

Phone Security Profiles 416
Token Scopes and Revocation 418
MRA Troubleshooting 418
Interactive Connectivity Establishment (ICE) 419
Defending Against Attacks at the Edge 420
B2B Connectivity 422
Monitoring and Compliance 423
Summary 424
Additional Resources 425

Chapter 12 Securing Cloud and Hybrid Cloud Services 427
Business Drivers for Cloud and Hybrid UC Services 428
Coordinating for Governance and Compliance 430
Transport Security and Compliance 432
Considerations for Secure Calling 433
Who's Who and What Privileges? 437
User Onboarding and Role-Based Access 437
Directory Connector 438
SAML 2.0 440
OAuth 2.0 441
SCIM 443
Device Onboarding 443
Securing Messaging Services 446
End-to-End Message Encryption 447
External Communications and Content Management 448
Data Loss Prevention 451
Mobility Management 455
Meeting Management and Security Controls 456
Un-Scheduled and Scheduled Meetings 457
Meeting Authentication 460
End-to-End Encryption for Meetings 461
In Meeting Privacy Controls 462
Protection of Data at Rest 463
Security Across Emerging Features 463
Facial Recognition 464
People Insights 465

https://t.me/learningnets
@cisco_in_persian_channel

Webex Assistant 466
Meeting Transcription 467
IoT Security 467
Summary 470
Additional Resources 470

Afterword 471

Index 475

https://t.me/learningnets

@cisco_in_persian_channel

Icons Used in This Book

Command Syntax Conventions

The conventions used to present command syntax in this book are the same conventions used in Cisco's Command Reference. The Command Reference describes these conventions as follows:

- **Boldface** indicates commands and keywords that are entered literally as shown. In actual configuration examples and output (not general command syntax), boldface indicates commands that are manually input by the user (such as a **show** command).
- *Italics* indicate arguments for which you supply actual values.
- Vertical bars (|) separate alternative, mutually exclusive elements.
- Square brackets [ ] indicate optional elements.
- Braces { } indicate a required choice.
- Braces within brackets [{ }] indicate a required choice within an optional element.

Note This book covers multiple operating systems, and a differentiation of icons and router names indicates the appropriate OS that is being referenced. IOS and IOS XE use router names like **R1** and **R2** and are referenced by the IOS router icon. IOS XR routers use router names like **XR1** and **XR2** and are referenced by the IOS XR router icon.

https://t.me/learningnets

@cisco_in_persian_channel

Introduction

Security in a Unified Communications network has become an essential aspect that organizations must account for in any modern UC environment. The purpose of this book is to provide a solid foundation to those individuals interested in learning the methodologies and technologies involved in securing a UC environment. In simple terms the purpose of this book is to start the conversation about how to secure a UC environment and where to look for more information.

Our goal in building the foundational knowledge for UC security is not to simply restate the various Solution Reference Network Design (SRND) guides or other configuration guides. The goal is to provide practical examples of when and how to secure aspects of the UC environment while providing direction on where to look for more detailed explanations of the technologies.

The methodology used in this book to convey the information is *explain, demonstrate, and verify.* Using this method, we first explain the concept (why is this being done). We follow up with a demonstration (how this is implemented). Lastly, we provide the process to verify the security aspect implemented (how you know it worked).

Goals and Methods

The primary focus of the book is that of helping Collaboration engineers and IT managers understand the need for security in their Collaboration environment and how to implement it. The content is intended to provide a foundation for the concepts and technologies used to secure modern Collaboration environments. The structure of the book is to first explain the concept and then walk through the configuration process. Lastly, this methodology will aid engineers and IT leaders in understanding how to verify their Collaboration environment is operating in a secure manner.

The core sections of the text show how to secure a modern Cisco UC environment that supports voice, video, IM, and presence to include the integrations with other real-time Collaboration technologies that facilitate Mobile and Remote Access (MRA) and bring your own device (BYOD). We also help you understand the reference network design to support UC services; portions of text are dedicated to helping you understand the attack surface and how to secure sections of the network in a logical progression through the different Cisco UC application domains.

We provide the relevant reference links for a more in-depth explanation as required. Chapter summaries provide a quick checklist of the learning objectives covered.

Because we provide the consolidated leading practices to protect a Cisco UC environment using the methodology of *explain, demonstrate, and verify*, you are afforded an understanding of the threat that is being protected against, followed by the steps to implement the security features, and lastly how to verify that the implemented security features are working.

https://t.me/learningnets

@cisco_in_persian_channel

Who Should Read This Book?

The book's primary audience is administrators and leaders who are planning to implement or are interested in improving their Collaboration security. One of the challenges for those who are interested or required to secure UC environments is the need to utilize numerous different resources to find the proper steps to secure systems, and unfortunately, the reasons are often not fully explained. This book provides the "why" that is often missing from the documentation and provides the foundational knowledge of current threats to a UC environment.

This book is intended for those with an intermediate level of understanding of the applications being discussed. You then will be able to take the information from this book and ensure that your Enterprise Unified Communications environment is deployed to the highest security standards and aligned with industry leading practices, giving you the best chance at achieving all levels of compliance to various types of security audits and inspections.

How This Book Is Organized

For those who are linear thinkers, our writing style is for you. This book takes you on a natural progression of securing a Unified Communications environment. Starting with the basic principles of security (physical security), we also include how to provide life and safety solutions for the most important asset of an organization: its workforce. From there, we progress to helping you understand the importance of securing the network before the UC applications, because the network is often the second point of insertion for an attacker (after physical). In a practical way, we spend time explaining how to enable security features on core components of a UC environment, which are the UC applications hosted within the network. Lastly, we communicate the importance of securing the edge of the network and UC services that are located outside of a traditional on-premise environment to help you understand what implications there are when consuming services from a cloud environment. Using a fictional case study to provide a basis for thought, we employ a storytelling style to help simplify the message.

Although there is a natural progression for securing UC environments, we have written chapters so that you can focus on specific areas of interest, ensuring that you do not miss out on introductory material if you decide not to read through the book from start to finish. Given this information, however, we recommend reading through each of the chapters. Because security is often about exploiting the weakest link, we recommend thinking about security from an architectural perspective. This is also how you will be able to get the full benefit of the book.

https://t.me/learningnets

@cisco_in_persian_channel

Chapter 1

The Importance of Practical UC Security

The demand for expertise that can comprehend the cybersecurity threat landscape and the technical controls that can be implemented in both the data and Unified Communication (UC) realms has never been higher. Across the world, security analysts have been seeing an increase in attacks in service provider networks and enterprise organizations supporting UC environments. According to Chuck Robbins, CEO of Cisco, Cisco blocked 7 trillion threats in 2019 on behalf of its customers. Meanwhile, hackers are finding innovative and inexpensive ways to increase the volume of attacks that are often difficult to detect and defend against. An example of one of these techniques is robocalling. According to *USA Today*, Americans received approximately 58.5 billion robocalls in 2019, which is a 22 percent increase from 2018. As the number of cyber threats increases, the more likely it is that your organization will come under attack. In other words, it's not a matter of *if* the organization will be attacked; it's a matter of *when*.

As attacks against UC environments are increasing, standards bodies and federal organizations such as the Federal Communications Commission (FCC) and Federal Trade Commission (FTC) are attempting to provide assistance. For example, the Internet Engineering Task Force (IETF) has developed a working group called the Secure Telephony Identity Revisited (STIR), while the Alliance for Telecommunications Industry Solutions (ATIS) and the SIP forum have worked to produce the Signature-based Handling of Asserted Information Using toKENs (SHAKEN). These new technical standards form the STIR/SHAKEN framework, which the FCC has mandated that service providers deploy in the IP portions of their networks by June 30, 2021. The outcome of this mandate is that phone companies will be responsible for implementing caller-ID authentication, which would protect consumers against malicious caller-ID spoofing techniques that are often used during robocall campaigns to trick consumers into answering their calls. The FCC estimates that the benefits of eliminating the wasted time and nuisance caused by illegal scam robocalls will exceed $3 billion annually. The FCC has also proposed $225 million in fines to discourage the continued increase of robocalls.

https://t.me/learningnets

@cisco_in_persian_channel

In a recent study conducted by ZK Research, 93 percent of CXOs responded with data suggesting that business models are changing. Survey data showed that CXOs can see a permanent 30 percent uplift in the number of remote employees across an organization. Survey responders also stated that maintaining productivity is the top challenge when shifting a workforce to business models where remote employees work from home. This means that security controls must extend beyond the traditional organizational boundaries. A combination of threat intelligence, visibility into network threats, and security controls is needed to rapidly detect and mitigate security threats at the most optimal place in the network.

The questions now become, What is practical UC security, and how can it help? Practical UC security is the notion of balancing security versus operations so that organizations can provide UC services to their workforce that are highly available and secure at the same time. Network administrators have been faced with this struggle for many years, so it is fair to say that there is more expertise in this area. Additionally, this means that the practice of securing the network is more mature, with better-defined processes and procedures than those used for securing a UC environment. Although necessary and more mature, network security often has a detrimental effect on the UC environment. With this in mind, one of the goals that we aim to help you understand over the next several chapters is the role that networks have with providing stable and reliable UC operations. Our secondary goal is to help UC administrators understand how to secure the UC environment against different types of threats and attacks without impacting the availability of UC services.

For the remainder of this chapter we continue to reinforce the importance of UC security so that you can gain an increased understanding of the threat landscape. Additionally, we provide examples illustrating how security can possibly be detrimental if implemented without a proper understanding of the operational considerations. At this point, you are well on your way to adopting the principles of practical UC security.

Identifying the Threat Landscape

Before going much further, we need to identify the most common threats that UC environments are faced with and further examine what impact that surrounding environments have to providing secure and stable UC services. As an example, consider the impact that social media can have against UC security. In 2011, a popular rapper called The Game offered an opportunity to become one of his interns. This message went out to more than 500,000 followers over Twitter. All that was required from followers was to call a certain phone number to show interest.

Unbeknownst to the rapper's followers, the phone number actually belonged to the Compton branch of the Los Angeles County Sheriff's Office. In a short amount of time, his followers started placing calls to the telephone number and overwhelmed the telephone system owned by the Los Angeles County Sheriff's Office, effectively creating a distributed telephony denial of service (TDoS) on the sherriff's office. The attack lasted for the next three days and prevented the sheriff's office from receiving and responding

https://t.me/learningnets

@cisco_in_persian_channel

to any real emergency calls. If this could happen nine years ago, how much more vulnerable are organizations to this type of attack after all these years of social media growth?

Some of the other common threats and their impact on either the UC environment or its users include but are not limited to the following:

- **Eavesdropping:** Listening to someone else's conversation. This threat creates privacy concerns.
- **Identity theft:** Stealing a user's credentials without permission to gain access to information. This attack can impact both instant messaging clients and IP endpoints.
- **Call spoofing:** Placing a call while changing the caller ID to impersonate another user to gain access to information. This specific attack is related to identity theft.
- **Robocalling:** Using an outbound dialing system to generate calls that are meant to deliver prerecorded messages (robot). This attack is often used for illegal scams.
- **Voice phishing (vishing):** Making calls that are intended to obtain personal or financial information about an individual. This attack is often used for identity theft or fraud.
- **Session replay:** Recording a voice or video call with the intent to use it for malicious intent (for example, a blackmail attempt).
- **Denial of service (DOS)/telephony denial of service (TDoS):** Denying valid users the ability to use UC services (e.g., creating a telephony denial of service, or TDoS, so that users cannot place inbound or outbound calls).
- **Media tampering:** Hijacking a voice or video session to challenge the integrity of a session.
- **Toll fraud:** Utilizing voice or video systems without authorization, thus increasing the cost of the system (e.g., placing unauthorized long-distance calls, which results in high telephone bills).
- **Malicious discovery of private information:** Using caller ID, password/accounts, calling patterns, or presence information for identity theft, call spoofing, robocalling, and so on.

Now that you have a better understanding of the threats against the UC environment, it is important to analyze which of these are the most important ones to protect against and which techniques can be applied in a practical manner to least disrupt operations of the UC environment.

If an organization doesn't have the security controls in place to address any of the threats that we have highlighted, a paradigm shift is in order to move away from the status quo and to prioritize UC security. By now, it should be evident that the challenges with security in a UC environment are very different from those used in the past, especially prior to a time in which VoIP became so prevalent. This section is not intended to critique customers who have little amounts of security implemented; instead, it is intended to address

https://t.me/learningnets

@cisco_in_persian_channel

the growing number of risks that organizations are faced with if changes are not made to address the growing threat landscape. On the other end of the spectrum are those organizations that fully understand the risk landscape and have very strict security controls in place. In organizations that do have various security controls in place, they should ask whether any of the controls are impacting the productivity of the workforce, whether the controls are still relevant, what attacks are protected against, and whether the organization is best positioned to address new threats on a rapid basis. In an environment that has a proper balance between security and operations, the environment will be as secure as it needs to be to address the necessary threats without introducing any of the unnecessary security complexities that may impact the stability or availability of UC services that are offered to the workforce. It will also be agile so that new security threats can be easily identified and mitigated. Otherwise, there is risk of revolt from the workforce if the organization introduces applications with capabilities that may not so easily be secured. This growing trend is often referred to as shadow IT.

The Danger of Shadow IT

The term *shadow IT* can be defined as any technology that is deployed within an organization without formal approval from the IT department. The reasons that organizations have shadow IT issues are almost always due to good intentions. Typically, shadow IT arises with the desire to be more effective or to improve productivity in daily activities. For example, a project manager believes that by sharing information more effectively, project members will be able to increase productivity. Project members agree that the ability to make quick decisions and freely share information between each other will make their job easier. Therefore, the project manager and team members decide to download and install a free instant messaging and file sharing client. This decision to install and use an instant messaging client is done without any guidance or formal approval from the IT department with the intent to better collaborate with coworkers.

The risk of shadow IT encompasses each and every area of the network, including UC and personal computing. Examples of where users may be currently deploying unsanctioned hardware and software to increase their capabilities include but are not limited to

- **Personal devices (BYOD):** Computers, tablets, phones, storage devices, and so on
- **Productivity applications:** Slack, Trello, Hive, Asana, Workplace by Facebook
- **Communication applications:** Skype, Chanty, Google Hangouts, Viber
- **Cloud storage:** Google Drive, Box, Dropbox, and so on

Recent studies regarding shadow IT provide additional insight into how prevalent this type of threat is to an organization:

- Eighty percent of end users are using software that isn't approved by IT.
- Eighty-three percent of IT staff admit to using unsanctioned software or services.
- Only 8 percent of all enterprises actually know the impact that shadow IT has within their organization.

https://t.me/learningnets

@cisco_in_persian_channel

The important takeaway from these statistics is that end users are savvy enough to take matters into their own hands if they feel as though they do not have the capabilities that they need to be productive. Unfortunately, many users are unaware of the security risks of installing unsanctioned software or hardware that is likely not deployed in a secure manner (see Figure 1-1). Shadow IT also creates additional challenges or difficulties with maintaining compliance with initiatives such as Sarbanes-Oxley, Gramm-Leach-Biley, HIPAA, FISMA, COBIT, and ITIL.

Figure 1-1 *A Major Security Threat for Organizations Is Shadow IT*

To prevent the revolt of the user community and to minimize the amount of shadow IT that an organization has to deal with, organizations must consider enabling the capabilities that users want so that the user community isn't tempted to take matters into their own hands while turning to insecure solutions. In short, we propose that it is counterproductive to the business to create an environment that is either not flexible or agile enough to provide solutions that users need. It is especially these types of environments in which shadow IT becomes a risk. One of the biggest concerns that organizations have when enabling new solutions or specific new features is sustainment. Therefore, you must make sure that the new solutions or features that are deployed across a system are easily manageable, can be easily secured, and do not place an unnecessary burden on administrators. The next few sections address this concern.

Balancing Operations and Security

The most secure UC environment is one that is isolated from the outside world. Unfortunately, when you are isolated from the outside world, you fail to leverage most

https://t.me/learningnets

@cisco_in_persian_channel

of the capabilities that you have in your UC environment. Therefore, you need a balance between operations and UC security. When implementing security controls to mitigate against the threats to the organization, you should know that varying levels of complexity are associated with each control. As security features are enabled to mitigate threats, there is risk of the system becoming more complex to troubleshoot and operate.

To further help you understand the varying levels of complexity, we have created a matrix of security features and provided an analysis of what can be considered low, medium, and high complexity. Features that are *low* complexity are those that are already built into the operating system and application. In some cases, especially when security features are enabled inside the operating system or application by default, it might not be evident that they are there. This is a good thing because it minimizes the effort needed to provide security to an operational system.

Features that provide *medium* security are those that protect against the threats without adding too much operational complexity. It can be said that this type of security is effective and reasonable to enable.

Lastly, a category of features and/or architectural components is associated with a *higher* level of operational complexity. This higher level of complexity is not meant to depict a scenario that creates an environment that is not sustainable. Features that are included in categories with higher levels of complexity come with more dependencies. This is not to say that these features or components are not needed, but just that an additional cost (e.g., capital and/or labor) may be associated with these features and components. Based on requirements, compliance, laws, and IT security policies, each organization needs to analyze which of the security features are relevant and why. This is the balance of continuous operations (availability of services) and security that is necessary to mitigate threats. Table 1-1 displays the progression of features. We discuss these items in more detail in subsequent chapters.

As an example, one method of securing access to the network is to use switchport security, which is a feature that can be enabled in Cisco Catalyst switches. This feature is very simple to deploy, which makes this a common method of securing the network for many organizations. This option does have its pitfalls though. If care is not taken, administrators can cause unintentional consequences that may limit devices from roaming around the network without administrator intervention. Therefore, many organizations are adopting solutions that provide MAC Authentication Bypass (MAB) functionality, which allows UC devices to be moved around a network with minimal intervention from an administrator. An administrator can centrally manage the devices that are admitted onto the network by MAC address and other attributes. The dependency with this option is that a RADIUS server is required for MAB; therefore, an additional system must be managed. Lastly, with 802.1x and NAC, very specific configurations are required on switch ports, and very specific security policies on the RADIUS server are required to enforce security policies.

https://t.me/learningnets

@cisco_in_persian_channel

Table 1-1 *Varying Levels of Complexity Associated with UC Security Features*

Category	Low (Easy or Default)	Medium (Moderate)	High (Advanced)
X.509 certificates	Self-signed certificates	Public key infrastructure (PKI)	FIPS 140-2 compliant PKI
Platform security	Hardened platform (OS)	Enforced security policies: password complexity, logging, etc.	LDAP-enforced security
Infrastructure security	Trust Verification Service (TVS)	Encrypted signaling/ media for endpoint	Encrypted signaling/media for infrastructure
Network access	Switchport security	MAC Authentication Bypass (MAB)	802.1x and NAC
Segmentation	Separate voice and data VLAN	Separate VRF for voice and data	Micro segmentation
Authentication	LDAP authentication	LDAP single sign-on (SSO)	Dual-factor authentication
Edge security	Firewall, ACLs, IPS	Session Border Controller (SBC)	SBC integration with UC-aware IDS/IPS

Minimizing Complexity

Practical UC security is built on the premise that it is possible to increase security to mitigate risks while minimizing complexity. When following practical UC security principles and a well-defined architectural framework, you will be able to deploy the UC solutions and features that users desire while maintaining various other aspects of design guidance such as

- Interoperability
- Scalability
- Quality user experience
- Simplicity
- Security

Conventional thinking has caused a bad precedent in this area of minimizing complexity. The conventional thought is that to reduce complexity, security features should be disabled or minimized. Although it is not often considered a high-tech approach to security, one of the simplest ways of keeping a UC environment secure while minimizing complexity is to stay current with up-to-date versions of application software, firmware, and operating systems by patching systems regularly. For UC environments, Cisco regularly

https://t.me/learningnets

@cisco_in_persian_channel

closes vulnerabilities in software so that customers are not vulnerable to the latest exploits and vulnerabilities. We discuss this topic further in Chapter 5, "Hardening the Core Cisco UC Appliance Operating Systems."

Another reason that organizations hesitate with enabling security features is the sustainment with business operations. There is concern from a performance perspective: that systems may perform more slowly and also become more difficult to use. Additional concerns are around the business. If heightened security controls across one or more systems can create issues with interoperability and/or backward compatibility, especially with critical business systems, employees, or business partners, the benefits do not outweigh the risk. In other words, there is a perception that this is an all-or-nothing approach. This is why there is such a tension between enabling security in a UC environment.

While interoperability, performance (scalability), and complexity/ease of use are all issues that an organization should be concerned about, we propose that security is not an all-or-nothing approach. Various security features can be implemented without compromising any or all of these items. If following a good architectural framework, organizations can deploy UC security while minimizing complexity. As an example, Cisco Unified Communications Manager (CM) provides the capability for mixed-mode operations. By enabling mixed mode, organizations can freely deploy a mixture of UC endpoints that require transport security (e.g., encrypted signaling and media) and those that do not. We cover this topic in more detail in Chapter 7, "Encrypting Media and Signaling."

Examples of different architectural frameworks that we follow over the next several chapters include the National Institute of Standards and Technology (NIST) cybersecurity framework, the Solution Reference Network Design (SRND) guide for collaboration, and preferred architectures. The NIST framework, version 1.1, which was published in April 2018, defines five core functions that organizations can use to identify threats and mitigate risk: identify, protect, detect, respond, and recover. This framework is primarily intended for federal organizations but is still applicable for other types of organizations ranging from small and medium businesses to state government, academia, and so on. The NIST framework can be found here: www.nist.gov/cyberframework/framework. In addition to industry standard frameworks such as NIST, Cisco provides a design zone for Collaboration. The design zone for Collaboration includes resources that can help direct users to the SRND. Otherwise, you may navigate directly to the SRND by browsing to www.cisco.com/go/srnd. The design zone can be found here: www.cisco.com/c/en/us/solutions/enterprise/design-zone-collaboration/index.html. This resource also includes access to preferred architectures for certain-sized organizations. These design guides are excellent references for anything that is not specifically covered in this book.

An approach for implementing UC security while minimizing complexity is to adopt a tiered model of security controls and privileges. To do this, you can create user profiles based on roles or access to information to provide a right-sized approach to security across an organization's UC environment. As an example, an organization has analyzed that 70 percent of its workforce (and associated UC endpoints) are responsible for sharing information that may be considered trivial or commercially available. A user profile that is based on characteristics of a commercial user can be created that provides security

https://t.me/learningnets

@cisco_in_persian_channel

features that are considered easy to implement and manage. The same organization has determined that 20 percent of its workforce (and associated UC endpoints) are using or sharing information that is business related but not considered sensitive. A user profile that includes incremental security characteristics such as LDAP authentication, encrypted signaling, and encrypted media can be created and assigned to these users and devices. This profile would help prevent specific attacks such as identity theft, eavesdropping, media tampering, and session replay. For the remaining 10 percent of the organization, such as administrators who are responsible for managing the system, executives, or users who are working with sensitive information, trade secrets, acquisitions, mergers, and so on, a more stringent security profile can be created and assigned to users and devices. This profile would include all of the characteristics of the enterprise profile but with added security controls such as multifactor authentication, or enhanced encryption algorithms can be used to prevent attacks for a longer time period.

By minimizing security complexity for the majority of the workforce (70 percent of the organization), administrators are able to focus on creating and maintaining a secure environment for the 30 percent of the remaining users. When organizations minimize complexity, they gain increases in productivity. Systems become simpler to manage, users have the ability to consume services in an easy fashion, and businesses are able to easily communicate with their business partners so that they can easily conduct business meetings and transactions together in a simplified way. Simplicity also leads to agility so that organizations can adapt business models to trends and campaigns to help drive growth. A representation of this type of tiered model is shown in Figure 1-2.

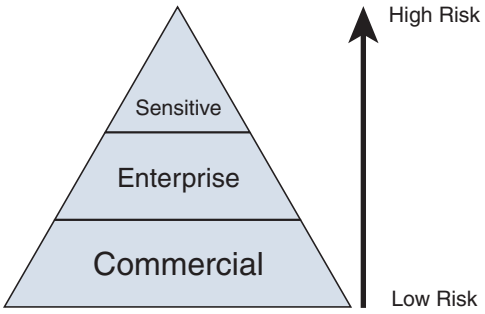


Figure 1-2 *Organizational Structure Depicting a Ratio of Users and Devices to Various Security Profiles to Simplify Complexity of Managed Features*

Security, interoperability, and simplicity are all key aspects of a properly architected UC environment. The US Department of Defense (DoD), which has some of the most stringent security policies, controls, and procedures in place to protect its information, has a unique perspective on this topic. In a memo effective October 7, 2019, the DoD Chief Information Officer (CIO) states: "DoD will implement a multi-tiered cybersecurity risk management process to protect U.S. interests." The tiers in the DoD CIO's approach differ from profiles based on the type of user, but the concept is similar. In the same memo, the DoD CIO states that "interoperability will be achieved through adherence

https://t.me/learningnets
@cisco_in_persian_channel

to DoD architecture principles, adopting a standards-based approach, and by all DoD Components sharing the level of risk necessary to achieve mission success."

Such interoperability was not always the case. In 1991 during Operation Desert Storm, a coalition-based military was deployed into the Persian Gulf with the goal of expelling the occupying Iraqi forces from Kuwait. Initially, the call failure rate was approximately 70 to 80 percent per day. There were numerous reasons for these call failures, some of which were due to communications strategy, incompatible call signaling, insufficient amounts of training, and high call volume (over 700,000 calls and 150,000 messages daily). According to Lieutenant General James Cassity, "The services put more electronic communications connectivity into the Gulf in 90 days than we put into Europe in 40 years." The point of this case study is to place a priority on the architecture and design to reduce complexity. Unfortunately, it took members of the military and GTE approximately three months to identify incompatibility issues with call signaling. Hindsight is always 20/20, but if there had been a better communication strategy in place, performance issues could have also been properly addressed while ensuring interoperability between departments. This is also where specific types of management tools can be beneficial. In today's world there will always be challenges, complexities, and unknowns. To properly address these scenarios and to mitigate security in our UC environments, organizations need to consider acquiring a toolset that can help them increase their visibility to the issues and properly manage the risks as new threats emerge across a growing threat landscape.

Visibility and Management

Management tools play a key role in helping monitor the performance of a UC environment and developing an understanding for baseline activities. In this regard, management tools help address emerging issues and complexity, some of which are security related. As a rule of thumb, you cannot protect or fix what you don't know about! Therefore, it is critical that UC teams leverage management tools that provide visibility to the environment. Types of activities that UC teams should monitor include the number of calls, the number of authentications, failed authentications, and the number of connections being made.

Similar to cybersecurity attacks, the sooner that UC teams are able to identify potential attacks, the faster that they can be mitigated. This is especially true for day-0 types of attacks, which have not yet been classified. These types of attacks are best identified by monitoring for anomalies and any deviations from normal activity, such as a spike in the number of attempted calls or connection attempts to detect a TDoS attack. Cisco's free Real-Time Monitoring Tool (RTMT) can help UC administrators begin to monitor performance and view alarms on Cisco Unified CM and Unity Connection. RTMT was originally developed with the intention of monitoring the performance and stability of Cisco Unified CM and Unity Connection, but if reviewing from a security point of view, it can provide security insights such as the number of registered phone devices and the number of calls attempted/in progress/completed. You can download RTMT from **Cisco Unified CM Administration Application>Plugin**.

https://t.me/learningnets

@cisco_in_persian_channel

Organizations and service providers that need more sophisticated management tools to help monitor UC threats should investigate use of a Unified Communications threat management (UCTM) system. Unlike RTMT, UCTMs were built for security, so UCTMs are able to help organizations establish a baseline for normal activity while using patented algorithms to block unwanted traffic and/or well-known attacks, such as DoS, call flooding, robocalls, and hijacking. UCTMs are also able to monitor for vulnerabilities at the protocol level (e.g., SIP) and prevent traffic from entering into the network to authorized users.

Introduction to ACME: Case Study

To the extent that is possible, over the subsequent chapters we reference a fictional company named ACME and develop scenarios around ACME's security endeavors. The case study is the foundation for an instructional methodology that provides the following information:

- Explains the theory behind various UC security features
- Demonstrates how to configure various UC security features
- Verifies the expected results

To do this, we introduce you to a ACME Commercial Machine Equipment, which is a midsize company that has recently chosen to deploy Cisco Unified Communications. ACME has recently gone through some restructuring within its IT department, and as part of this restructuring, the company has hired a new UC administrator named Anthony Starke. Over the next few months, Anthony's main function will be to maintain, operate, and enhance the company's UC environment.

Part of the restructuring was due to drive growth in ACME's business sector with the release of its new product, the ACME Mechanized Solutions Widget. To drive business growth, the company has plans to establish additional relationships and connections to its closest business partners. ACME has asked Anthony to focus on reviewing the security posture of the currently deployed UC equipment so that ACME and its business partners can have a better understanding about the security threats that each organization will need to remediate so that ACME can maintain its compliance with mechanized standards. Prior to establishing business connectivity, ACME has ensured its partners that it will provide them with the consulting services needed for understanding how to align to industry best practices and regulations so that the business partners do not risk becoming out of compliance with their existing systems.

The ACME case study is intended to chronicle Anthony's considerations, experiences, trials, and tribulations as he tries to assess and improve the security posture of the ACME UC environment. The case study also chronicles Anthony's experiences as he works with other personalities and departments to ensure that the organization provides users with the ability to maximize the features and functions afforded by the Cisco Unified Communications architecture while causing minimal disruption to end users.

https://t.me/learningnets

@cisco_in_persian_channel

Currently, the ACME corporate UC environment is centralized within the corporate HQ. As shown in Figure 1-3 and Tables 1-2 and 1-3, ACME's UC infrastructure consists of the basic Cisco Unified Communications applications and network infrastructure. ACME also has several remote offices with field agents who receive voice services from the HQ. As ACME continues to refine and optimize the security that is deployed in its UC environment, it will find that there are options available to better support its workforce and business partners. Similar to way in which the chapters are organized, the ACME case study starts by first looking inward at physical and life safety concerns, which are typically overlooked when implementing UC security. In this chapter we help ACME integrate solutions for E911, such as Cisco Emergency Responder.

We then progress outward to network services, in which we go into more depth of providing network-based security and authentication to services, such as through the addition of Cisco Identity Services Engine (ISE). At this point we help ACME secure the core Unified Communications applications. Ultimately, the case study will help you better understand the considerations of providing collaboration services at the edge of the network, leveraging Cisco's Collaboration edge architecture and supported devices. Lastly, we review considerations for migrating to a hybrid/cloud architecture such as Webex calling to provide support for emerging requirements and capabilities while reviewing the security features that Cisco provides in these types of environments.

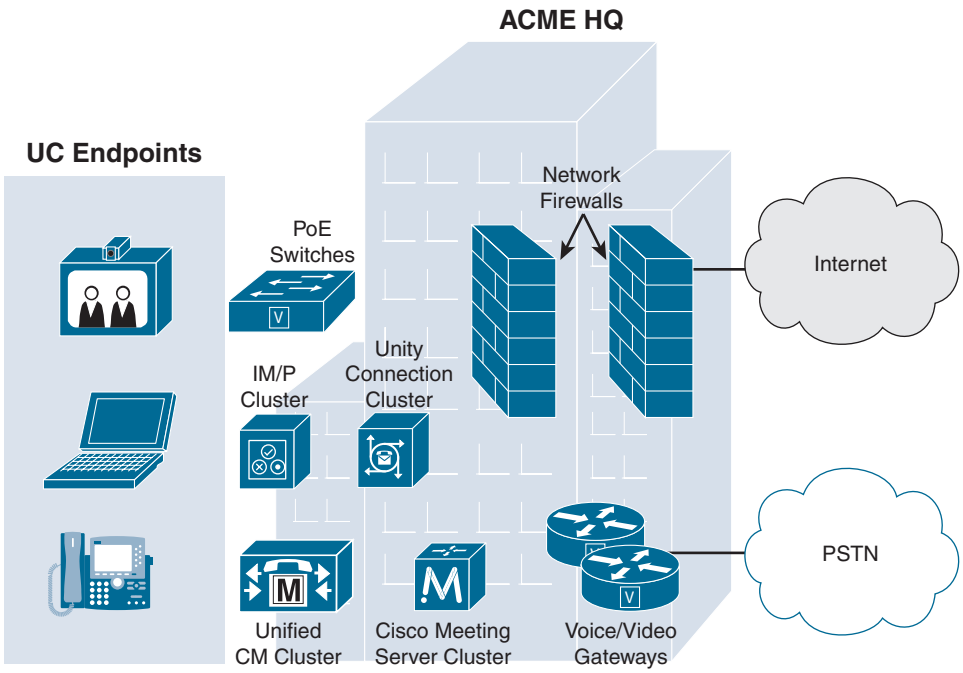


Figure 1-3 *ACME's Current UC Architecture*

https://t.me/learningnets

@cisco_in_persian_channel

Table 1-2 *ACME's Current UC Applications and Versions*

Application Name	Application Version	Application Role
Cisco Unified Communications Manager	12.5	Call control: VoIP and video devices, user provisioning
Cisco Unity Connection	12.5	Voicemail, basic ACD
Cisco Instant Messaging and Presence	12.5	Instant messages, presence, calling
Cisco Meeting Server	2.9	Audio/video conferencing

Table 1-3 *ACME's Current Network Devices and Versions*

Platform Type	Software Version	Platform Role
ISR 4400 router	IOS XE 16.9	PSTN gateway
Catalyst 3850 and 9000 series switches	IOS XE 16.9	Campus switching, PoE

Summary

There is a growing demand for providing security for Unified Communication environments. As the world and technology inside of it change, new and more sophisticated threats continue to arise. When these threats are exploited by individuals or groups, whether they are internal or external to an organization, those threats can be damaging. At best, if an organization is attacked and exploited, the outcome can be embarrassing. At worst, it can be damaging to the business or mission of an organization. In the case of the Los Angeles County Sheriff's Office, staff were unable to answer and respond to emergency calls for three days. The attack originated from a simple post on social media.

As technology changes, it is up to organizations to provide users with the capabilities that they need in a secure manner. Otherwise, there is the risk that users will revolt and deploy capabilities in an unsecure manner, which may increase an organization's attack surface. For security to be effective, it must be in place. Security features can be rolled out in phases, or all at once, but it must be there to be effective. In many cases, the security that is provided by default is sufficient in mitigating security threats. This leaves us with the final thought of not being able to protect or fix what you don't know about. If you need to manage and secure a UC environment, tools that provide visibility are essential. These tools allow organizations to address emerging issues related to performance and security while also providing the confidence that they can operate their UC environment in a safe and predictable manner.

https://t.me/learningnets

@cisco_in_persian_channel

Additional Resources

www.nist.gov

www.fcc.gov

www.cisco.com/go/srnd

https://t.me/learningnets

@cisco_in_persian_channel

Chapter 2

Physical Security and Life Safety

Introduction to Physical Security and Life Safety

Physical security is best defined as the protection of personnel and equipment from physical actions that could cause harm to individuals or extended loss of service or data. Organizations often consider the use of physical security controls a "no brainer," in that they provide safety and security to individuals within the organization, and rightly so because an organization's most valuable asset is its individuals.

Because many organizations have lost assets due to improper physical security, experience teaches us about its importance. Yet, despite our experience regarding the importance of physical security, it is difficult to place an estimated cost on what is needed to provide security for assets along with safety for individuals. Recent data from IBM Security Services shows that 71 percent of all attacks on the financial industry and 58 percent of all attacks in the health-care industry were carried out by malicious insiders or inadvertent actors (accidental events). For these reasons, it is important to discuss how physical security can provide benefits in increased reliability, availability, and productivity for the entire network, including the Unified Communications (UC) environment.

In this chapter, we highlight the threats and risks associated with having suboptimal physical security as it pertains to the availability of services from the Unified Communications environment. We also discuss the importance of life safety as it pertains to using an organization's Unified Communications environment to provide safety for individuals inside an organization and a rapid response in the event of a life-threatening situation. Our goal is to help you understand why it is critical to increase physical security and life safety solutions. Additionally, our goal is to help you implement these solutions while minimizing costs and risks to life-threatening situations.

https://t.me/learningnets

@cisco_in_persian_channel

Assessing ACME

As a new employee of ACME, Anthony has decided that to find out what type of shape the organization is in, he should talk to some of his coworkers to understand what physical security-related incidents have happened over the past few years. Anthony is shocked to find out that there have been quite a few issues related to physical security.

In one of Anthony's interviews, he learns that an engineer named Pete was involved in a large outage due to loss of power. Pete needed to power up a new server, but there weren't any spare ports close enough to the available port that was free on the power distribution unit (PDU) inside the data center rack. Rather than slowing down the installation of the new server, Pete decided to just rearrange some of the ports so that everyone would be happy, and he could move on with his project. Pete noticed that the server had dual-power supplies, so he didn't think that it would be a big deal and that probably nobody would even notice that he momentarily unplugged a connection. Unbeknownst to Pete, this server hosted several virtual machines that provide Unified Communication services to the organization. Pete was also unaware that a week earlier a power surge on the PDU had caused the circuit breaker to flip. In less than five minutes, several virtual machines providing Unified Communications services to the organization came to a hard crash. Luckily, the virtual machines were not corrupted, and the services came back up after restoring power.

During these interviews, Anthony learns from a few managers that ACME has employees working in locations that have recently passed Enhanced 911 (E911) legislation. Managers want to ensure they will be compliant to legal and/or regulatory obligations to reduce their risk of liability related to emergency calls, especially since the company is poised for large growth over the next one to three years. Because ACME has a large HQ facility with onsite security personnel, Anthony and his management team have decided to use Cisco Emergency Responder (CER) for ACME's E911 solution, which would help them take advantage of onsite alerting and notification's functionality. ACME also believes that CER has the necessary compliance features built into the product to comply with local and federal laws. ACME does have plans for opening small satellite offices at some point in the future, so Anthony will need to take them into consideration when building out the E911 architecture.

After finishing the interview process, Anthony has a better appreciation for the tasks that he has been hired to accomplish. He also has a greater appreciation for some of the things he learned in school and the reality that he is faced with. Anthony has never been involved in deploying an E911 solution before, so he is struggling with how to start creating a design for ACME's HQ and its remote sites. Anthony next decides to meet with his manager to find out what additional information he can learn about the issues that ACME has been faced with and to get support for improving ACME's physical security prior to any enhancements on its existing environment. Although there have been some service-impacting incidents, the general consensus is that the organization is excited about the future of a Unified Communication environment.

https://t.me/learningnets

@cisco_in_persian_channel

Questions that you should ask:

1. What physical security controls do you have in your organization?
2. What exposure does your organization have to someone making on-the-fly decisions about power?
3. How can an organization proactively detect an issue with a circuit or power supply?
4. What E911 legislation is applicable in your area?

A Physical Security Methodology

A methodology for providing physical security should involve ways to *prepare, prevent, detect, and respond* to physical security attacks. As shown in Figure 2-1, this methodology never ends. Organizations should always be preparing to prevent, detect, and respond to physical security attacks. Simply stated, the goal of physical security is to provide safeguards and countermeasures to protect against any physical threat or vulnerability to an organization or its data. As an example, it is certainly possibly to have physical security safeguards in place such as a lock on a door to the data center without having a procedure to detect or respond to unauthorized users in the data center. If this is the case for your business, it is worth identifying these types of gaps and applying countermeasures to mitigate against them as quickly as possible.

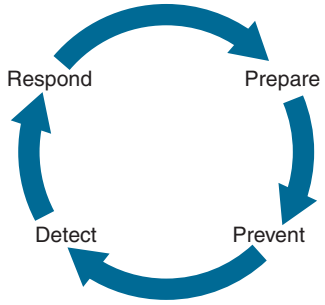


Figure 2-1 *Prepare, Prevent, Detect, and Respond Methodology*

Preparation

The first step to ensuring the highest level of physical security is preparation. Preparation for all physical security initiatives should be based on measures needed to protect people, the organization, its data, and access to services. Business objectives help drive the vision for high availability (HA), business continuity, continuity of operations, and disaster recovery goals. The most prepared organizations are the ones that understand the various types of attacks and the most likely scenarios that they are vulnerable to.

https://t.me/learningnets
@cisco_in_persian_channel

An appropriate example can be taken from a National Guard unit from the State of California. Because the National Guard unit understands that an earthquake could be a catastrophic event for their state, it is something that they have (and continue) to prepare for. Based on the preparation phase in this methodology, the California National Guard would have a plan in place for how to respond to a natural disaster, such as an earthquake. Preparations may include any or all of the following processes and procedures for communities that the National Guard is deployed to:

- How to notify members from National Guard units
- How/when to work with local communities to provide medical support
- How/when to work with law enforcement agencies to provide safety and security for local citizens
- How to use communications infrastructure to communicate with federal, state, and local agencies

When preparing to notify members from National Guard units, leadership may prepare something such as an alert roster, which helps individuals understand how to contact each other in an emergency situation. The alert roster may have information such as cell phone number, home phone number, and phone number of family members. Generally speaking, the alert roster is hierarchical, so the first individuals to find out they are being deployed are those in leadership. They, in turn, are responsible for communicating the situation and the plan of action to the individuals listed under their care in the alert roster.

While they will not be able to prevent the earthquake, there is a sense of urgency and prioritization in place because if the National Guard is properly prepared for the various scenarios, they may be able to help prevent anarchy in the surrounding communities and therefore help save lives by responding quickly and effectively. If physical security describes the overarching goal for protecting against threats and vulnerabilities, individual controls are the options that an organization has for deploying countermeasures to assist with improving physical security.

There are three main categories for physical security controls: management (administrative), physical, and technical. Management controls are best suited for the preparation phase. This is the most important category for implementing physical security because this category is necessary for providing the vision and resources for physical and technical controls. Physical and technical controls are listed in the "Prevention" section.

> **Management (administrative) controls:** These controls are focused on the establishment of policies, procedures, processes, guidelines, and so on. A previous example of using an alert roster is a type of management control that can provide an organized procedure and process to share information with others. Another example of a management control is an evacuation plan in the event of a fire, with well-defined evacuation routes and meet-up areas.

To be best prepared for physical security threats, an organization should conduct a risk assessment so that the threats and vulnerabilities can be identified and then prioritized

https://t.me/learningnets

@cisco_in_persian_channel

for mitigation. Lastly, policies should be provided around monitoring of physical and technical controls so that the organization can reevaluate levels of risk to verify whether the appropriate controls and countermeasures are in place. This level of control will help organizations adapt to new risks based on organizational goals.

Prevention

Prevention is the area of physical security that most organizations focus on because it is the area that is focused on mitigating physical security risk. Many different individual types of security controls fall into the categories of physical and technical (logical) controls. Prevention techniques range from using the necessary controls to provide the first line of defense until the last to prevent and deter physical security attacks from happening. As shown in Figure 2-2, building bollards can be used to protect the structure of a building from vehicles getting too close to the building or possibly driving into the building and impacting its structure and employees inside. Other preventive controls include protecting critical areas of buildings, such as the main entrance to a building or data center with access control systems. Physical and technical controls are needed for the following reasons:

Physical controls: These physical controls exist inside and outside of buildings to prevent unwanted individuals or natural phenomena from accessing a location or impacting a piece of equipment. Examples of these controls include various types of locks, doors, walls, and even security guards. Outside the building, they include bollards, fences, security gates, and security guards.

Technical (logical) controls: Technical controls exist inside and outside of buildings to prevent unwanted individuals or natural phenomena from impacting the services and data that are provided by an organization. Examples include badge readers that provide authentication and authorization to areas of the building, closed-circuit TV (CCTV) and cameras, motion detection, fire and smoke monitoring, fire suppression, uninterruptible power supplies (UPSs), and external power generators.

Figure 2-2 *Building Bollards, Typically Used to Protect the Entry Point of a Building*

https://t.me/learningnets

@cisco_in_persian_channel

Inside a building, physical controls can be implemented to ensure visitors don't have unfettered access to the building. Locks are perhaps the most common way of preventing access to parts of the building. One well-known approach for bypassing physical controls that locks do not necessarily prevent is *piggybacking,* which is entering through an area that someone else was authorized to enter. A turnstile-based system helps protect against these types of "social" attacks and also possibly aids in detection if paired with a badge or scanner-based system. These systems can generate an alarm if unauthorized entry is detected. Other types of preventive systems include full-body-scanning machines, such as what the Transportation Security Administration (TSA) uses to detect objects in an individual's clothing or within the body to ensure that weapons are not carried into the building.

In short, the best way to control and prevent physical security attacks is to adopt multiple solutions and controls. Although this section focuses on physical security, you can borrow from some of the best practices used in the network, such as taking a defense-in-depth approach, which is defined by not allowing a single point of failure to impact the physical security. This way, an organization can remain confident of its security if one particular control or solution fails because others can still be used to prevent an attack. An example of a defense-in-depth approach is a building that has a turnstile in place with an alerting system and a security guard monitoring incoming/outgoing access to the building and through the turnstile. If at any time an unauthorized guest tries to gain access to the building, multiple controls are in place (turnstile, alerting system, security guard) to mitigate this from happening.

Prevention techniques should consider the necessary perspectives to support physical and technical controls. This leads us to the next phase in our methodology: detection.

Detection

Detection is about reacting to physical security violations in an optimized manner. Because organizations consist of many different types and sizes, there is flexibility for how controls are monitored and detected. Available options are security operations centers, local security teams, incident response teams, and so on. The best way to detect incidents is to be prepared for them. Therefore, determining which people, practices, processes, and tools that should be used on a regular basis to detect incidents will help organizations react and respond to violations in an optimized manner. Remember, physical security detection is focused not only on detecting unauthorized entry into an area. Detection also encompasses any type of threat or vulnerability, such as a fire or electrical issue, to protect an organization's data.

Leveraging tools such as CCTV or IP-based video surveillance, cameras provide the ability for detection and real-time situational awareness. Some of these technologies can be integrated into facial recognition software systems. Additionally, sensors should be leveraged to trigger alarms or to detect fire, water, seismic activity, or radiation and should be regularly monitored by a security or operations team. To help with detection, security teams and decision makers should analyze whether certain locations (outside or inside) of a building need more granular detective controls. For example, buildings

https://t.me/learningnets

@cisco_in_persian_channel

in a high-risk urban crime zone may need more sophisticated technology to prevent and detect security issues in a more automated fashion. Additionally, an area in which an organization operates and maintains high-value services or resources to its employees, such as its Unified Communication systems, should include improved detection controls.

Whether services are provided from a low-risk suburban area, a high-risk urban area, or from an isolated location, such as in the deep woods, detection should be optimized to rapidly respond to security violations in an optimized manner. As we discuss later, many policy violations occur within an organization, with natural phenomena or bad actors. In many of these scenarios, it behooves an organization to have corrective and recovery controls. An example of a corrective control includes the ability to remediate an issue such as fire using fire-suppression methods such as water, foam, or dry powder after a fire is detected. An additional safeguard would be to simultaneously call in the fire department for support and assistance before the fire escalates to the point beyond which normal corrective systems can handle the problem. Recovery control involves the return back to business. More details about these controls are included in the next section.

Response

To properly articulate the importance of the response phase, we reference a quote from the Greek philosopher Epictetus: "It's not what happens to you, but how you react to it that matters." Our point is that an efficient response can mean the difference between life and death. In a response to incidents, different types of controls can be used. As previously discussed, the first step in a response is to mitigate the incident if it is still possible, such as to put out a fire as quickly as possible. Responses to an event can also be manual, such as engaging an organization's safety and security team. The controls that can be implemented in the response phase are as follows:

- **Corrective controls:** Controls that involve physical, administrative, and technical methods of responding to the detection of an undesirable event. These controls are designed to eliminate or reduce the damage incurred by the event. This includes something such as integrating sensors into the phone system so that calls can be placed automatically by the system to an entity such as the fire department.
- **Recovery controls:** Controls that involve the restoration of a system back to a normal operating state after data has been corrupted or has experienced a loss of availability. This includes the process of configuring system backups, either manually or automatically on a scheduled basis, and the process to restore data from the system backup.
- **Forensic controls:** Controls that involve the investigation of the incident, including reviewing the type and time of the breach, investigating logs, reviewing user access, and reviewing response time.

In this section we introduce the notion of forensic controls. These controls are typically used after recovery has taken place. The primary goal of using forensics is to collect

https://t.me/learningnets

@cisco_in_persian_channel

as much information as possible to investigate who, how, why, when, and where the incident occurred so that decision makers and technical owners can work to prevent future incidents from occurring. In the typical sense, items that are collected include blood, fingerprints, hair, ballistics (bullet, casings), and digital evidence (hard drives, mobile phones, and other data storage devices). A chain of custody is built while collecting these items to ensure that they are labeled correctly and to avoid any possibility of evidence tampering. A chain of custody should establish, for example, who had contact with evidence, when it was handled, and what changes were made.

Practical Physical Security for Your UC Environment

Before going too deeply into physical security for your Unified Communications environment, first remember what we discussed in the first chapter about confidentiality, integrity, and availability. By now, you likely understand why physical security (or lack thereof) can have a severe impact on the confidentiality, integrity, and availability of an organization's production UC environment. For many organizations, such as the federal government, some of the highest priorities involve defending the network and protecting the confidentiality of information. This topic is a recurring theme as we discuss securing different areas of the network and different applications in upcoming sections and chapters. Many other organizations, especially those organizations that are public facing, such as Google, are focused on maintaining availability. With any loss of availability, an organization such as Google could lose credibility, which could cause an adverse impact to the value of their stock, which impacts shareholders.

As we examine the threats to availability, we must examine the threats of virtualized servers. Most if not all of Cisco's current UC applications are now deployed on virtual machines (VMs). If engineered incorrectly, or without consideration of how the UC application can be designed for fault tolerance, such as installing all of the UC applications on a single physical server, an organization can be at risk for a severe loss of availability. A scenario that organizations often fail to plan for is an authorized internal user with malicious intent against the organization. There is a lot to be said about trust of certain individuals, but there is also a lot to be said about implementing certain types of controls that do not give users the proverbial "keys to the kingdom." If an organization implements only certain types of physical security controls, a malicious insider could easily bring down an organization's UC services and impact the reputation of the organization along with the management team responsible for delivery of UC services. Additionally, organizations need to understand the technical controls to detect the user responsible for causing the service interruptions. The next few sections discuss what steps an organization can take to protect against loss of service due to a single mistake or a preplanned attack.

Physical Security for the Data Center

IT personnel will always need to have access to equipment inside the data center. In our fictional case study with ACME, we pick on Pete a little bit. After all, he single-handedly took down services for thousands of users by not verifying that it was safe to move

https://t.me/learningnets

@cisco_in_persian_channel

power cabling around. Pete didn't act alone in his misguided attempt to finish his project, however. Notice that no type of change control process was involved in his decision making. Also notice that very little verification was done to verify the potential impact of the power changes. Given the issues and risks we have just highlighted, the question is whether Pete's organization had any of the right controls in place to prevent and detect the outage. In the ACME case study, Pete actually had good intentions and was authorized to be in the data center. What if Pete had had bad intentions? What could be the impact to an organization if an unauthorized user had access to the server room or data center? The next few sections focus on mitigating the damage for these two different circumstances.

To get started investigating how to prevent certain types of physical security attacks, you can take a risk assessment, starting with access-level security. What could happen if a key or combination to a door to the data center were to get into the hands (or memory) of someone that it shouldn't? Who would know about it? How would an organization mitigate the amount of damage that someone could cause? Borrowing from some of the key principles of role-based access, an administrative user should have the least amount of privilege necessary to complete necessary job duties. Additionally, users and access to resources should be continuously monitored to make sure that the authorized user is following access policies. The question is, What practical ways are there to improve physical security?

Understandably, some of you may love the technical solutions for physical security. Many of these are available, but the technical solutions are not always the most practical. Besides, it is often blocking and tackling that wins football games, not the Hail Mary pass, which often drives such excitement. Keeping things simple and executing on what is cheap and easy is often the most practical and first step toward practical UC security. With that said, one of the easiest things that an organization can do is provide user training. Personnel should be trained not to share keys and key codes. Properly trained users will also ensure that unauthorized users are not piggybacking into a room, especially into a data center. Management controls and principles can be applied by ensuring that personnel are never alone but always accompanied by a trusted individual. Monitoring can be done the old-fashioned way by using a log book in which individuals sign in and then sign out when they enter and exit the data center.

Regarding technical solutions from inside the data center, door rack security enables an organization to extend the last line of defense to the most critical pieces of equipment, providing the most critical services. As shown in Figure 2-3, the doors to the server rack demonstrate built-in rack security controls. Door-level security comes by default from many different vendors. We should point out that if door racks all use the same keys, the racks should be rekeyed to provide the appropriate amount of role-based access to control how a piece of equipment that belongs to a different team is accessed. Because this is the last line of defense for many organizations, additional access controls may be implemented, such as through badged or electronic systems. Electronic systems such as badge, badge proximity, or biometric security are good ways to prevent unauthorized access to pieces of equipment.

https://t.me/learningnets

@cisco_in_persian_channel

Whether malicious or not, an insider to an organization is often the biggest threat to loss of services from inside the data center. Unfortunately for organizations, malicious insiders have devised sophisticated schemes for attacking internal systems. Multiple reports have been shared about malicious insiders planting video cameras and keyloggers inside a data center so that they can gain access to a data center rack. To mitigate against these types of vulnerabilities, many physical-security-focused customers have leveraged access systems that have the capability to set up time restrictions. This way, access to a server rack can be limited to certain time periods that team members are expected to arrive and/or for a specific duration that a certain task is supposed to take. Another approach to limit the damage that an insider can cause is to monitor badge-based access control systems. This approach of monitoring access systems applies to anyone who gains access to the data center to proactively detect and prevent incidents. Examples of monitoring systems that can help monitor badge-based access control systems include CCTV, video surveillance, alarm generation, and log monitoring.

Figure 2-3 *Cisco R-Series R42612 Rack with Locking Front and Rear Doors and Side Panels to Offer Added Security*

Power Plant Considerations

A previous example described how an engineer (Pete) mistakenly unplugged power from the server supporting multiple virtual machines. Additional risks can contribute to a loss of service (blackout), such as a complete loss of power from the power company/provider. Some countermeasures to mitigate the risks of service interruption due to blackouts and power faults are an uninterruptible power supply (UPS), battery, battery bank, generator, and power distribution unit (PDU) that are redundant. Other considerations are an intentional reduction of power (brownout) by the power provider or a surge in voltage in the power supply, such as through an electrical storm. When dealing with brownouts, customers should deploy surge protectors, circuit breakers, transformers, and UPSs.

https://t.me/learningnets

@cisco_in_persian_channel

From a compliance perspective, the National Fire Protection Association (NFPA) 110 specifies the Standard for Emergency and Standby Power Systems, in conjunction with NFPA 70: National Electrical Code (NEC). Some mission-critical environments such as hospitals and data centers may be required to comply with NFPA 110. Because emergency power is so vital to safeguarding individuals within a building, these laws help ensure safety while minimizing loss of power due to poor installation, planning, and maintenance practices, which are all common causes of emergency power supply systems (EPSS) failure.

A final thought on this topic brings us back to the physical security methodology and life cycle. Is there a process to verify that the backup power systems are healthy? Is there a process by which the organization maintains backup power systems, such as a backup generator? How should an organization respond to users or customers if there is a degraded power situation? Listening to anecdotes and stories is sometimes fun, unless your generator does not start during an emergency because it has not been properly maintained. These are all things that an organization's leadership should be prepared for; they should provide guidance for administrators to ensure that preventive maintenance and associated operational checks are conducted on a regular basis.

Electrostatic Discharge

Organizations cannot be too careful about electrostatic discharge (ESD). In an article published by InCompliance, independent consultants found that ESD damage costs the average electronic manufacturer 4 to 8 percent of their total annual corporate revenue. From a cost perspective, this translates to over $80 billion per year across the international electronics industry. The ESD Association estimates that spending in ESD control measures can provide a return on investment (RoI) from hundreds to thousands of percentage points of the original investment. If you have never heard of ESD, it occurs when two different types of materials rub together, causing electrons to build up and transfer from one material to another. This action creates an electrical surge, which often is transferred to electrical components and ultimately results in equipment failure.

Many times, ESD is caused by the improper handling of electronic devices, such as memory cards, boards, and other materials. A basic set of ESD controls typically consists of

- A static-controlled work area
- An operator wearing ESD wrist straps, connected to a common ground point
- A static-controlled protective bag

A key component of ESD control is ESD awareness training. Through this training, administrators and users can become more aware of the urgency of ESD and become more familiar with the techniques and tools that they have available to them. This training also provides education for proper grounding techniques, recommended work areas, and transport of sensitive items, such as avoiding carpeted areas when a static control such as an ESD wrist strap or protective bag is unavailable.

https://t.me/learningnets

@cisco_in_persian_channel

Cable Plant Considerations

It should be no surprise that legacy telephone systems and many of their features weren't necessarily designed with security in mind. Caller-ID spoofing is a well-known attack that scammers have used for many years to wreak havoc by using fraudulent methods for financial gain. If you are unfamiliar with the term *caller-ID spoofing*, it is the process in which a caller deliberately falsifies the calling number to disguise the actual caller ID. An area of the network that can be a physical security liability is the cable plant. The cable plant, which supports both IP and analog telephony, can be an easy entry point for an attacker to compromise the integrity of the UC environment or individual voice conversations. An example of how security can be compromised is through use of a lineman's set, which is sometimes affectionately known as a *butt set*. A lineman's set is traditionally used to test and monitor an analog phone line for voice quality. However, for less than $100, a lineman's set can be acquired and used for malicious intent, such as to eavesdrop on a voice conversation or spoof a phone call from anywhere along the physical path that the analog line is wired in, such as inside the central office, at the main frame, or at the distribution frame. The threat is the loss of confidentiality when someone can eavesdrop on a voice conversation. The loss of integrity occurs as soon as someone is unable to recognize the voice at the distant end of the voice conversation when a lineman's set is used to place outbound calls (spoofing the caller ID) or to answer incoming calls.

Organizations continue to use analog phones for multiple good reasons, such as areas of the network that cannot support Power over Ethernet (PoE) due to legacy cabling or cable distance. Examples include in an elevator or at a security gate or to support legacy alarm systems. The point is that analog wiring is vulnerable at multiple points in the network; therefore, physical security should be implemented at these locations.

Eavesdropping on VoIP calls is also possible if an attacker is able to gain access to the switched network and acquire a packet capture or a copy of the voice conversation between IP phones. Once access to the switched network is obtained, a perpetrator can configure features such as SPAN, RSPAN, or ERSPAN to obtain a packet capture, within a packet capture tool such as Wireshark. Subsequently, Wireshark, by default, has the ability to filter by VoIP calls and SIP flows to detect calling number and called number. Most importantly, as shown in Figure 2-4, Wireshark can open unencrypted RTP flows so attackers can play back an entire telephone conversation.

If necessary, organizations may deploy protective distribution systems (PDS), such as conduit to protect the physical wire from being tapped as well as secure network ports or adapters.

Many physical security deficiencies, such as those by analog endpoints, can be overcome by converting to IP telephony. Technology such as Phybridge's Power Over Long-Reach Ethernet (PoLRE) is able to support IP telephony over a single twisted pair of telephony-grade wire (CAT3, CW1308), which would allow you to follow the recommendations in Chapter 7, "Encrypting Media and Signaling," to increase confidentiality and integrity of the UC environment.

https://t.me/learningnets

@cisco_in_persian_channel

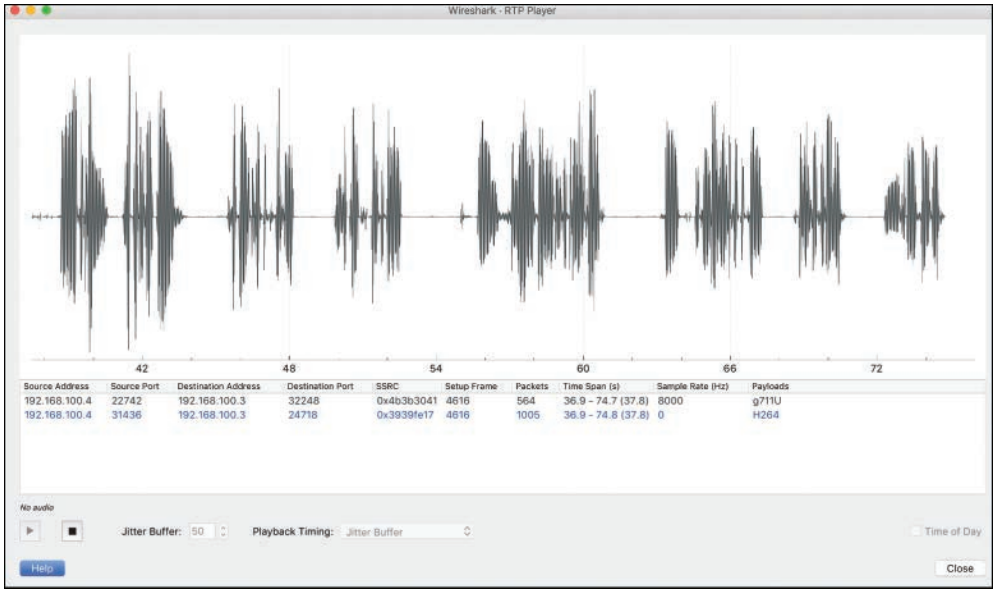


Figure 2-4 *A VoIP Conversation That Has Been Captured by Wireshark and Replayed*

PoLRE also supports access points that could be used for wireless local-area networks (WLANs). If a WLAN can be deployed, organizations can deploy low-cost Voice over WLAN (VoWLAN) telephones, with encrypted media and signaling for increased security. For more information about Voice over a WLAN, review Cisco's Enterprise Mobility design guide.

At this point, we have only touched the surface of technical controls that can be used for physical security and life safety. By now, you should be more familiar with the holistic approach and the importance of the various phases that we have provided regarding the prepare, prevent, detect, and respond (PPDR) methodology. It has been said that it is "best to prepare for the worst but hope for the best." The first half of this section should help you have a better focus on physical security. In upcoming chapters, we continue this methodology by describing how to prepare for a regional incident by leveraging Geo-Survivability features that are built into various UC applications. This design option helps reduce the risk of a service interruption due to simple mistakes or a physical issue in the data center. Additionally, in Chapter 6, "Core Cisco UC Application Lockdown," we describe disaster recovery mechanisms and demonstrate how to implement them.

We talk about detection and verification quite a bit over the next few chapters. Based on our experience, many service interruptions occur because IT administrators neglect the various stages in the PPDR life cycle. Whether the reason is that they don't have the right tools to properly detect issues or simply do not have a methodology to follow, it is a bit of a mystery. One thing for sure is that the "set it and forget it" approach is not a recipe for success.

https://t.me/learningnets

@cisco_in_persian_channel

Life and Safety Considerations

As we transition into life and safety considerations, the PPDR methodology still applies. To place the appropriate amount of prioritization toward the life and safety for individuals within an organization, an organizational management team needs to have policies and processes in place that prioritize safety. Management should also provide the technical controls that allow for rapidly detecting and responding to life and safety issues. From a technical perspective, IT managers should do their best to prepare for initiatives involving life and safety. Doing so includes proactively documenting and maintaining physical and logical connections that a network has across the organization.

The National Emergency Number Association (NENA), a nonprofit organization, estimates that 240 million calls are made to 911 in the US each year. When an agent answers a 911 call at a public safety answering point (PSAP), the agent's job is to obtain a physical address, callback telephone number, and information about the incident. If a call is disconnected for any reason, there is risk that the agent will not be able to obtain the necessary information. Because VoIP phones can be used virtually anywhere there is IP connectivity, there is increased risk of not being able to identify the exact physical location of the 911 caller. The same challenge with location exists with mobile devices.

Enhanced 911 (E911) was created to solve these challenges. By default, E911 systems automatically provide 911 agents with a callback telephone number and physical address. Floor numbers and suites can be provided also. As organizations look for ways to minimize the risk of not properly responding to health and life safety incidents, Cisco solutions supporting E911 requirements are becoming popular. The rest of this section is intended to help you understand more about E911 and various other life and safety solutions as well as how to implement them.

Introduction to Enhanced 911

At a high level, the primary goal of E911 is to extend the features of a basic 911 emergency call to make it more reliable and to provide additional functionality. Cisco provides a variety of solutions to enable E911 functionality. Depending on the current deployment or organizational requirements, there are varying benefits of Cisco's various E911 solutions. Cisco's E911 solutions are currently listed as

- Cisco Unified CME-based E911
- Cisco Unified CM-based E911
- Cisco Emergency Responder

As we discuss E911, it is important to remember that this is primarily a solution that is focused on the United States because 911 is the standard way of reporting an emergency in the US. Within the US, many organizations are working to improve E911 standards, including but not limited to the National Emergency Number Association (NENA) and the Federal Communications Commission (FCC). Unfortunately, there is not a global standard for reporting emergencies. Other countries have different numbers for reporting emergencies or requesting emergency services. As an example, an individual living in

https://t.me/learningnets

@cisco_in_persian_channel

Germany would call 110 for police emergencies or 112 for fire and/or medical emergencies. This section around E911 also focuses on campus environments that are owned and managed by organizations such as ACME. It is currently estimated that over 80 percent of 911 calls are placed from cell phones. For the sake of this discussion, cell phones are out of scope since the responsibility for providing E911 services lies on the service provider and not individual organizations.

In the future, next-generation 911 (NG911) solutions will help address some of these issues. Currently, E911 standards specify use of Emergency Response Locations (ERLs) and Emergency Location Identification Numbers (ELINs) to identify the physical location of a caller. In the future Presence Identification Data Format–Location Object (PIDF-LO), defined in RFC 5139, may be used to identify the location of a caller, especially in other countries outside the US. Since the NG911 solution is IP based, emergency callers are able to send pictures and videos of an emergency as well as SMS text messages in cases where the caller cannot speak. The biggest challenge in deploying NG911 is that the public safety answering points will need to upgrade their infrastructure. Unfortunately, PSAPs have been slow to upgrade their existing system to support NG911 solutions. Given this information, the rest of the chapter focuses on E911.

In keeping with ACME's architecture shown in Chapter 1, "The Importance of Practical UC Security," the next few sections address how to implement E911 based off the Cisco Unified CM and Emergency Responder applications. For additional details about implementing E911 with Unified CME, review the section on Enhanced 911 services.

Terms and Acronyms

E911 has many unique terms and acronyms. As we focus on helping articulate the considerations for deploying E911 solutions, we want to consolidate these terms and acronyms in a single location to help with familiarization:

- **ANI:** Automatic Number Identification. This is the calling party number (aka caller ID).
- **PSAP:** Public safety answering point. This is the publicly funded facility where 911 calls are routed so that emergency responders can be accurately dispatched.
- **ERL:** Emergency Response Location. This is a physical location that the customer manages. An ERL could be a cubicle, a floor of a building, or the entire building itself.
- **ELIN:** Emergency Location Identification Number. A valid North American number that is assigned to an ERL.
- **ALI DB:** Automated Location Identification Database. This is the database that PSAP uses to map the calling party number to a dispatch address/destination.
- **PS-ALI:** Private Switch Automatic Location Identification. This E911 enhancement allows for more detailed information to be sent to the PSAP than a physical address. Examples include floor number, suite, and room. This information is helpful in reducing the time to respond to an emergency.

https://t.me/learningnets

@cisco_in_persian_channel

Regulatory Considerations

Multiple federal statutes regarding E911 have been recently signed into law by the president. These statutes include Kari's Law and RAY BAUM's Act. These new statutes are closely related. Kari's Law originates from a tragedy in which the caller was not able to call 911 due to the requirement of dialing 9 before being able to get an outside line. Ray Baum was a lawyer who served in multiple public roles. RAY BAUM's Act was created in his honor, but it is also an acronym that stands for Repack Airwaves Yielding Better Access for Users of Modern Services. The following is a summary of these two new laws:

- **Kari's Law:** This law mandates *direct dial* and notification of 911 calls without having to dial a prefix to reach an outside line and to provide for notification (e.g., to a front desk or security office) to facilitate entry of emergency responders into a building when a 911 call is made. This law applies to multiline telephone systems (MLTS), which are commonly used in office campuses, large buildings, hotels, and so on and that may require a prefix such as the number 9 before being able to dial an off-premise telephone number.
- **RAY BAUM's Act:** This act requires that a "dispatchable location" is conveyed with 911 calls, regardless of the technological platform used. A dispatchable location is a verifiable address, including the suite or apartment number to help an emergency responder find the caller. The intent of this act is for 911 call centers to receive the caller's location automatically and enable the PSAP dispatcher to dispatch emergency responders more quickly and more accurately.

These two new statues are forward looking and apply only for MLTS that are manufactured, imported, or offered for sale after February 16, 2020. Kari's Law was enacted in 2018 and took effect as of February 2020. RAY BAUM's Act has different compliance deadlines depending on whether a 911 call originates from a fixed device (e.g., wired IP/analog phones) on-premises, at a customer site, or whether a device is located off-premises from IP phones that can be moved between locations or used while in motion (nonfixed). The FCC's primary focus is devices that reside on-premises because of the challenges of obtaining accurate location information while connected off-site by technology such as a virtual private network (VPN). In a fact sheet provided by the FCC, the FCC concludes that "MLTS providers should not be subject to the same location requirements for off-premises MLTS calls to the extent compliance is not technically feasible." For additional information on the FCC's rules regarding E911, Kari's Law, or RAY BAUM's Act, visit www.fcc.gov.

Regarding providers of VoIP telephone systems that are interconnected to the PSTN, the FCC currently requires that providers meet E911 obligations such as but not limited to

- Obtaining a customer's address prior to activation and providing ways for customers to update the location
- Routing all 911 calls, with the callback telephone number and registered address to the correct PSAP

https://t.me/learningnets

@cisco_in_persian_channel

- Taking appropriate actions to ensure customers have a clear understanding of the limitations of their 911 service

At the time of this writing, 23 US states have adopted legislation and/or regulations covering the 911 functionality required for organizations that are supporting users and devices with a multiline telephone system. Figure 2-5 shows which states have passed E911 legislation. Three states—Nebraska, New Jersey, and New York—have legislation pending. Organizations in process of implementing E911 solutions should review state legislation in case the state where they are located requires compliance to more specific items than federal legislation. The national 911 program provides the most up-to-date information on this topic. To review whether any legislation exists in your state, visit www.911.gov or visit the National Conference of State Legislators (NCSL) to view a searchable database on 911 bills.

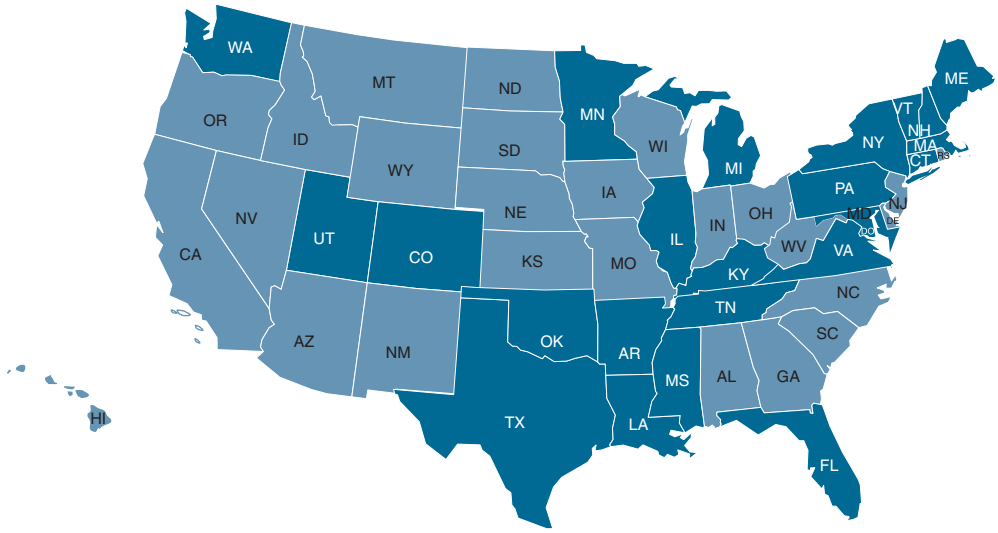


Figure 2-5 *States That Have Adopted E911 Legislation in Black*

Native E911 Call Routing with Cisco Unified CM

As of Cisco Unified Communications Manager 11.x and higher, customers who require E911 services can use the Cisco Unified CM's native E911 features to support E911 calling. This option is nice for customers who either have a single site or small number of locations for a PSAP to dispatch emergency services teams to. This option may also be beneficial for customers who have new E911 requirements but do not have the necessary licensing to deploy Cisco Emergency Responder yet. If organizations are considering this option, they should be aware of a current limitation within Unified CM: At this

https://t.me/learningnets
@cisco_in_persian_channel

time Unified CM's emergency call handler supports a maximum of only 100 ELINs per cluster.

With the native features, starting in Unified CM 11.x, an administrator can define ELINs, as shown in Figure 2-6. Unified CM allows these ELINs to be assigned to individual devices or at the device pool level, as shown in Figure 2-7. By following this two-step process for configuring E911 within Cisco Unified CM, an organization can gain the following capabilities:

- Automatic replacement of the calling party number with the appropriate ELIN
- Routing of emergency calls to the appropriate gateway for emergency call completion
- Dynamic association of the ELIN to the calling phone for callback purposes

Note Native emergency call routing within Cisco Unified CM should not be enabled if an organization is using an external emergency calling solution such as Cisco Emergency Responder.

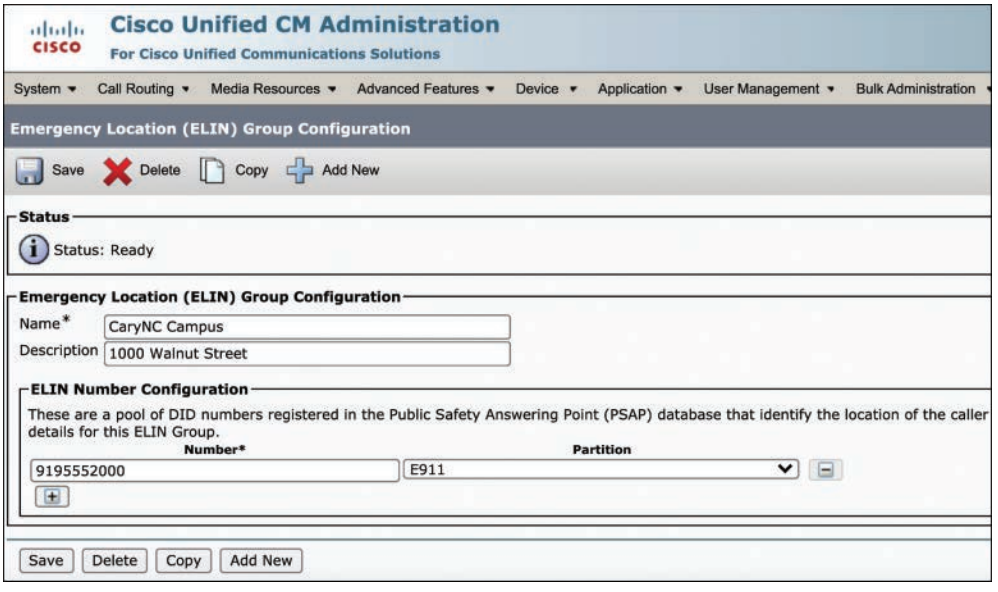


Figure 2-6 *Configuration of ELINs Within Unified CM for Use with the Native Emergency Call Handler*

https://t.me/learningnets

@cisco_in_persian_channel

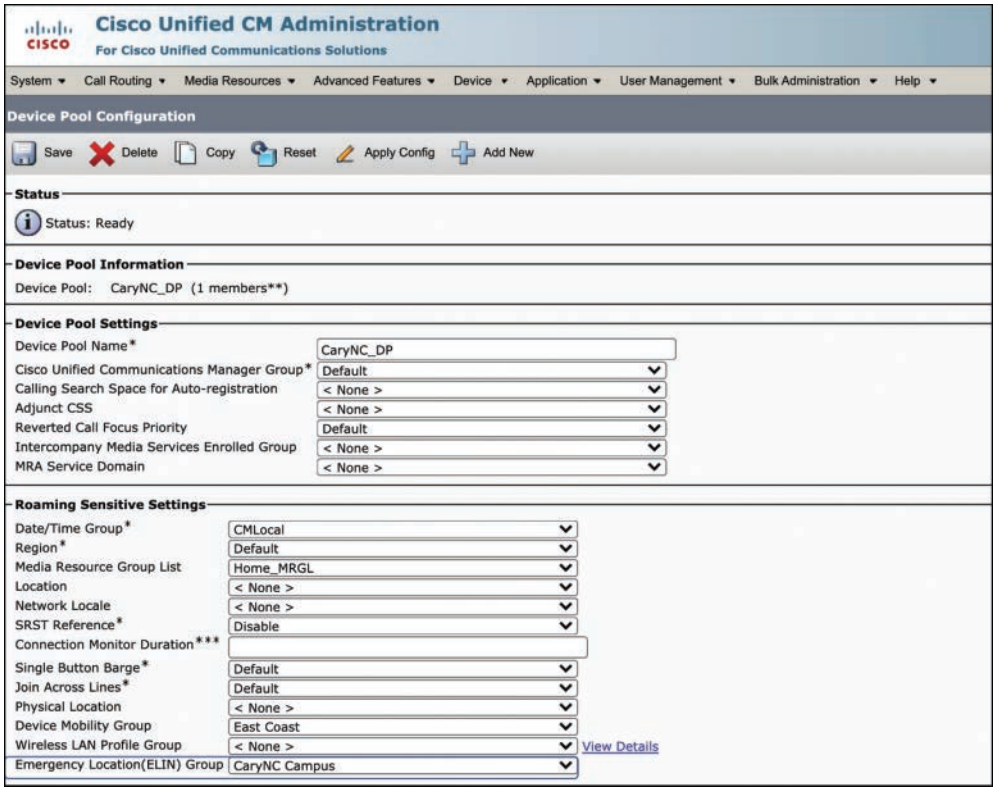


Figure 2-7 *Assignment of an ELIN Group to the Device Pool*

The final configuration step is to make sure that the 911 call is routed to the appropriate voice gateway and ultimately the correct public safety answering point. To do this, you need to configure a route pattern under **Call Routing > Route/Hunt > Route Pattern** to specify how to route a call to a PSTN gateway or route list. As shown in Figure 2-8, the checkbox specifying that a call matching this route pattern is an emergency services number should be checked. In some cases, one or more translation patterns may also be required to be able to use this route pattern. As an example, a translation pattern can be used to convert a call in which a user dials 9911 into 911 to comply with Kari's Law and to ensure that the emergency call is routed to the PSAP.

To recap the steps performed, here is a short summary:

1. Created an ELIN group using the location name CaryNC campus. Assigned an ELIN of 9195552000.
2. Assigned the ELIN group to the device pool CaryNC_DP.

For the PSAP to dispatch emergency services to the correct location, an administrator needs to make sure that PSAPs have the correct location for the ELIN of 9195552000 inside their Automatic Location Information Database (ALI-DB). Notice that Unified

https://t.me/learningnets
@cisco_in_persian_channel

CM does not require any address information for the CaryNC campus location—just a descriptive name that is locally significant to the UC administrator. Cisco recommends contacting the PSAP prior to any 911 calls being made to ensure accurate dispatch to the right location.

If additional scale or granularity is required, Cisco Emergency Responder (CER) can be used. The next few pages focus on implementing E911 within CER.

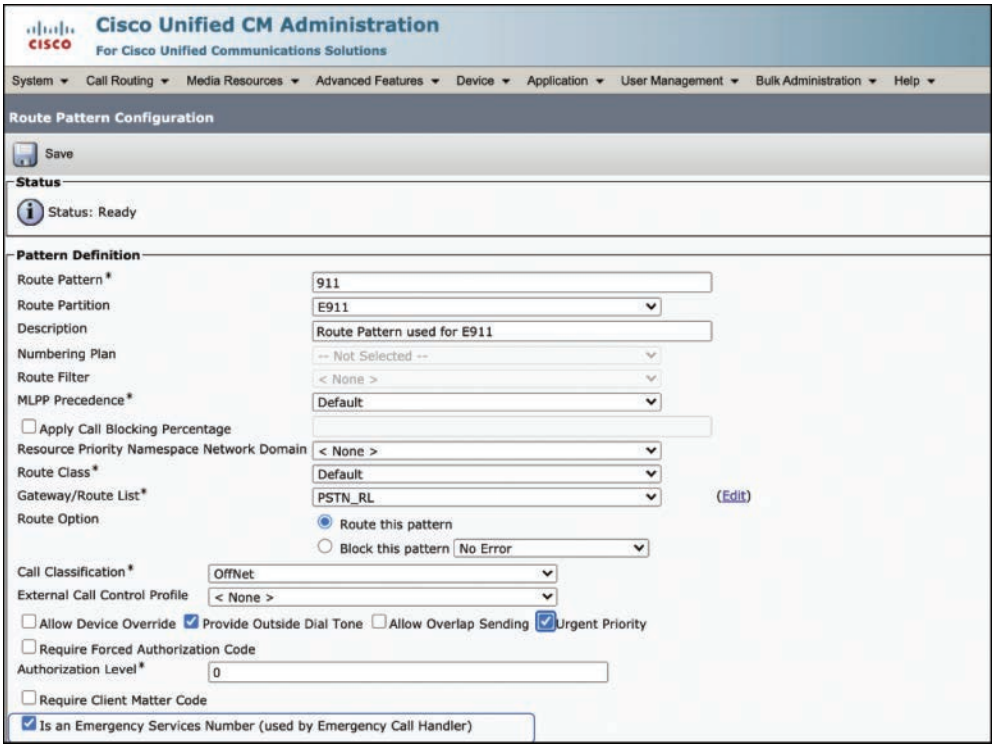


Figure 2-8 *Configuration of Route Pattern and Use of the Emergency Call Handler*

E911 Call Routing with Cisco Emergency Responder

In a campus environment or in a building with multiple floors, being able to direct emergency services responders can be challenging. This is especially true if callers are unaware of the building number they are in, an area of the building, a conference room, or a specific area of cubicles. So, in scenarios in which users are moving their devices, this movement inadvertently adds risk if they need to call 911 to report an emergency. The most powerful capability of Cisco Emergency Responder is automatically tracking the addition or movement of IP phones across a network and dynamically routing emergency calls to the appropriate local PSAP.

https://t.me/learningnets
@cisco_in_persian_channel

With Cisco Emergency Responder, if someone needs to call 911, the network infrastructure is used to dynamically assign the caller to an Emergency Response Location, without any user intervention. The 911 call is enhanced by the ability of CER to dynamically assign an ELIN to the 911 call, which the PSAP is able to use to automatically determine the location of the caller via the ALI-DB. Cisco Emergency Responder also provides the ability to generate notifications so that local personnel can help prepare for emergency services teams to arrive on-site and triage the emergency prior to the first responders' arrival. This proactive alert serves as a way to provide first aid to the caller while trained emergency services are on the way. Local notifications may also enable a company's incident response personnel to respond prior to emergency personnel and to provide life-saving CPR, contact family members to get a better understanding of the caller's health, or at the very least, keep the caller calm.

E911 Call Flow with Cisco Emergency Responder

To better help explain how Cisco Emergency Responder fits into the UC architecture, we first compare and contrast a basic 911 call to an E911 call. In a Cisco environment, when a basic 911 call is made, call routing logic within Cisco Unified CM should take the call to a Cisco PSTN gateway. If everything is configured correctly, the PSTN service provider routes the call to the public safety answering point, which is staffed with emergency response agents. In the 911 call flow, the call is delivered to the PSAP, but no information about the caller is delivered. This means the PSAP agent is responsible for obtaining information about the nature of the incident and obtaining the location information of the incident. Once everything is understood, the PSAP agent then arranges the dispatch of the appropriate emergency response, such as sending police, fire, or ambulance teams.

When we compare a standard 911 call with an E911 call, the biggest difference is that an E911 call delivers a calling party number to the PSAP, which can be used to automatically obtain the address of the caller based on the calling party lookup in the ALI-DB. As mentioned earlier, all ELIN and ERL information must be provided to the PSAP before placing a call to 911 to ensure the correct address will be used when an emergency call is placed.

In a Cisco environment, with Cisco Emergency Responder configured to provide E911, the call routing logic is a little more complex. Since Cisco Emergency Responder maintains the list of ERLs and their location information, Cisco Unified CM must redirect the 911 call to Emergency Responder so that the correct ERL can be associated with the caller and the correct ELIN can be delivered to the PSAP. The call flow diagram in Figure 2-9 depicts the E911 call flow, along with the different components: IP phone, Cisco Unified CM, CER, and the PSTN (VoIP) gateway. For an E911 call to be routed to the PSAP, these six steps take place:

1. Cisco Unified CM detects a 911 dial string and redirects the call to Cisco Emergency Responder (CER).
2. CER tracks the physical location of the calling device to an ERL and then assigns an ELIN (which is associated with the ERL) to modify the calling party ID for callback purposes.

https://t.me/learningnets

@cisco_in_persian_channel

3. CER routes the 911 call back to Cisco Unified CM.
4. Cisco Unified CM routes the emergency call out of the proper PSTN gateway and to the proper PSAP.
5. The caller location information is displayed at the emergency agent's terminal. This information is obtained by using the ELIN to query the Automatic Location Information database.
6. CER (and Unified CM) provides an on-site alert(s) to designated personnel within the ERL (optional).

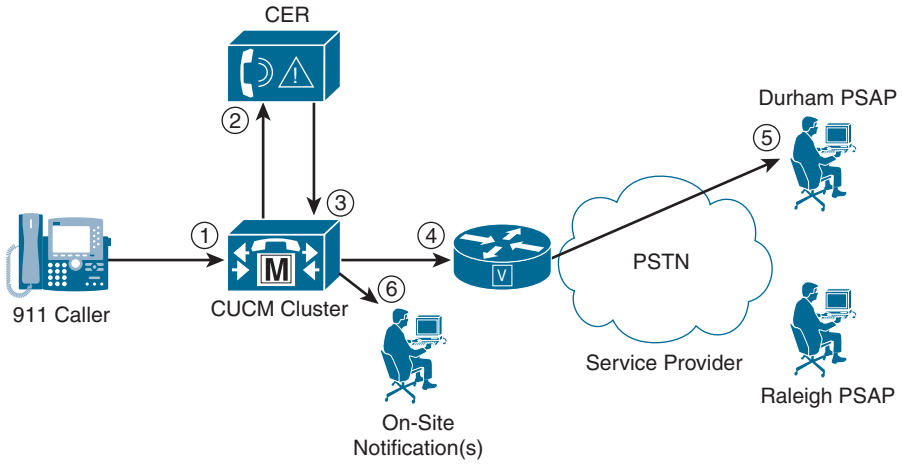


Figure 2-9 *Typical Call Flow for E911 with Cisco Emergency Responder*

Note IT administrators should consider developing an ERL naming strategy that helps onsite emergency personnel respond quickly to emergencies. An example is to incorporate building names, floor numbers, and other readily understood location information in the name.

As described in step 6, this is an example of how CER can help align to an organization's emergency response plan or procedures to support policies and procedures that help increase life safety while responding to emergency calls.

E911 Design

To make sure that an E911 solution meets all organizational, state, and federal requirements, we need to review some key terms and work through sample design scenarios. As previously discussed, legislation relating to size of an E911 Emergency Response Location

https://t.me/learningnets
@cisco_in_persian_channel

can vary for different cities, counties, states, and countries. It is up to you to learn your local statutes and design an E911 solution that will satisfy those statutes. A good place to start is with the local service provider because providers can help you understand the laws and how they would like you to submit the location information to them.

Perhaps the most critical aspect of the E911 design is the designation of Emergency Response Locations. An ERL can be a building, an area within a building, or an outside area (if you extend phone service outdoors) that is to be considered as a single location for emergency response purposes. From the PSAP agent's perspective, the ELIN is crucial in providing the caller's location. The reason is that the PSAP's Automatic Location Information database is based on the caller ID, which is the ELIN that is assigned to a call when that call is redirected by CER. For this reason, it is important to consider how ERLs are defined and whether there are enough ELINs available to support multiple calls from the same ERLs.

Guidelines and legislation at the state level are intended to help organizations create E911 boundaries. These include but are not limited to the following examples:

- Emergency Response Location size between 7,000 and 40,000 sq/ft
- One Emergency Response Location per floor of a multistory building
- Use of Emergency Response Locations and Emergency Location Identification Numbers

Given this information, here are some examples of how you may want to design your ERLs:

1. You have a five-story building, and each floor has 8,000 square feet of office space. Based on the guideline of one ERL per floor, the organization needs five ERLs.

   If floor size is a concern, because each floor is greater than the recommended square feet, the organization can divide each floor until the ERL becomes manageable in size. If a floor is 15,000 square feet, the organization can divide the area in half and assign two ERLs per floor (10 total ERLs). This allows emergency responders to locate an emergency caller's location with better accuracy.

2. You have 20 buildings in a campus environment. Each building is approximately 4,000 square feet. Based on square foot guidelines, the organization simply needs to create 20 ERLs, which would be one per building, even if some of the buildings have multiple floors.

Based on these examples, an organization has different options for tracking the location of an IP phone across the network. When an organization has a full understanding of where each network cable terminates, it can be more specific when assigning ERLs to a physical location. As an example, it is preferable to assign ERLs to physical switch ports because doing so provides the most accurate method of determining the location of a 911 caller. If an organization does not know where a network cable terminates, it has to assign an ERL to a more abstract location. For example, if an organization has an Ethernet switch that services a specific floor of its building, it might assign all ports of

https://t.me/learningnets

@cisco_in_persian_channel

the switch to a single ERL for that floor. In order of preference, the typical way that CER tracks the location of devices is through the use of

1. **Switch port tracking:** When an IP phone is tracked behind one or many switch ports that are assigned to an ERL.
2. **Access point association:** When a wireless device is associated with a specific access point that is assigned to an ERL.
3. **IP subnet tracking:** When an IP phone is tracked behind a specific IP subnet across the network.
4. **Manual tracking:** When a line is manually configured for an ERL to support devices that cannot be automatically tracked such as analog phones or fax machines connected to a voice gateway or analog telephone adapter (ATA).
5. **Unlocated phone assignment:** When an administrator reassigns IP phones by MAC address to an ERL.
6. **Default ERL:** When no other tracking methods are available to determine the location, a default location is used. Although this option is not very desirable, it is important to point out because it ensures that a 911 call will never fail.

ERL Creation and Network Discovery

After one or more ERLs have been designated, and an organization understands the methods that devices can be tracked, we can move on to the configuration aspect of the network and CER. When an ERL is created, a unique name is required along with a route/translation pattern and ELIN. From within CER, an administrator has the option of using route/translation patterns to steer 911 calls to the correct PSTN gateway. The route/translation patterns need to match what is already configured within Unified CM so that calls are routed correctly. If an organization has a centralized cluster, with multiple PSTN gateways, a good approach is to use the area code as a prefix, which adds another layer of significance and simplicity when trying to configure call routing. We cover more details on this subject in the section on call routing.

If desired, on-site notifications can be assigned to ERLs to align with an enterprise's life and safety policies. The mechanisms of how to alert on-site personnel include a real-time web interface, a telephone call, or an email. If notification to a pager or an SMS device is needed, an email gateway service may be needed. An example of how to configure an on-site alert is shown in Figure 2-10. You can configure on-site alerting prior to creating any ERLs or afterward. If you configure it afterward, you have to update the ERLs that require notifications. An example of ERL creation and how different parameters are assigned is displayed in Figure 2-11. Notice that the details about the Automatic Location Information (ALI) can be edited within the ERL itself. If multiple PSTN gateways are in place, call steering can be used by applying a specific route/translation pattern.

https://t.me/learningnets

@cisco_in_persian_channel

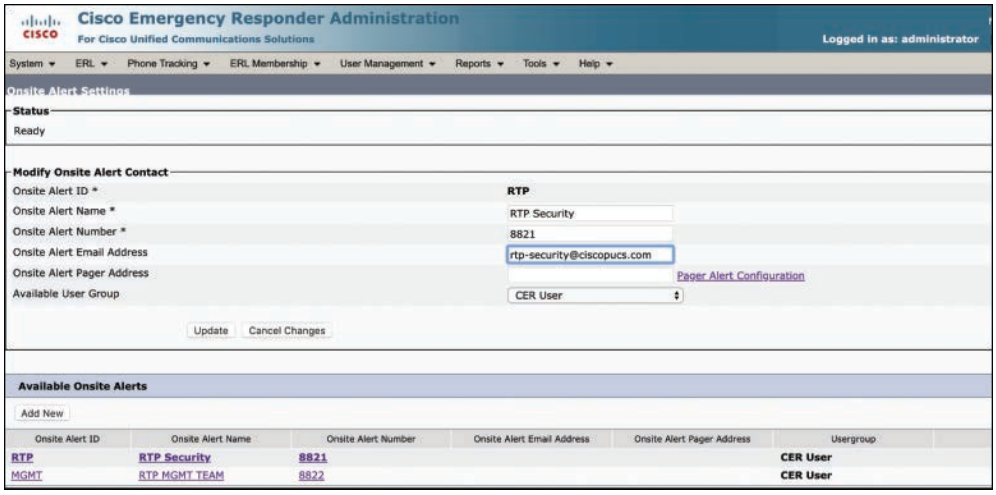


Figure 2-10 *Configuration of On-Site Alerting to Help Prepare for Emergency Response Teams*

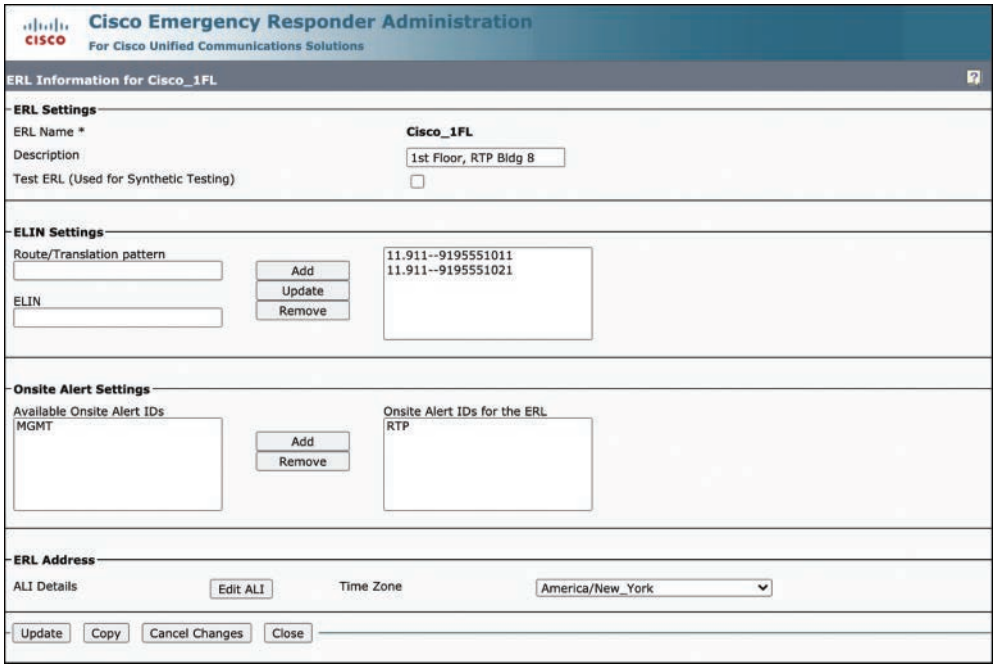


Figure 2-11 *ERL Creation, ELIN Assignments, and On-Site Alerting*

https://t.me/learningnets

@cisco_in_persian_channel

For CER to track the location of an IP phone, it must query Cisco Unified CM for a list of phones registered with the cluster and also the switches on the network. This way CER can determine which switch ports the IP phones are connected to. To allow CER to poll switches in the network, an administrator needs to navigate to **Phone Tracking > LAN Switch Details** and provide a **Switch Host Name / IP Address (Required)** and specify how to track the location of IP phones, as shown in Figure 2-12. CER currently uses Cisco Discovery Protocol (CDP) to locate phones connected to Cisco switches. CER currently does not support Link Layer Discovery Protocol (LLDP). If CDP is not enabled on the switch, or third-party switches are in use, IP Subnet tracking is the best and only option currently supported. Although Cisco mentions obtaining information about devices from the Content Addressable Memory (CAM) table on the switch, this option is not widely adopted. Use of the CAM to obtain information from Cisco devices is less efficient than CDP and is generally recommended only to support legacy IP phones that do not support CDP. Optionally, administrators can choose to display port descriptions from the switch so that ERLs can easily be assigned to the right ports.

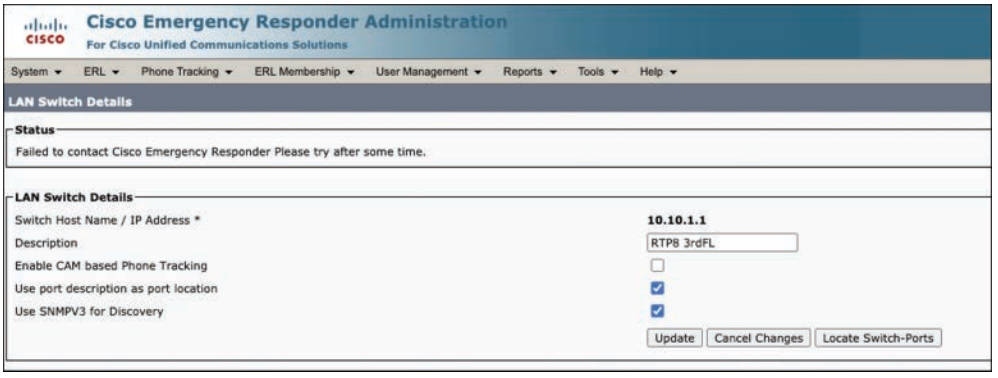


Figure 2-12 *Adding a LAN Switch into CER*

To poll Cisco Unified CM and supported Cisco switches, an organization can use Simple Network Management Protocol (SNMPv2 or SNMPv3). Many security-focused customers use SNMPv3 because it addresses the security deficiencies in SNMPv2. Specifically, SNMPv3 is a user-based security model that provides message-level security by using authentication and privacy. To configure SNMP settings in CER, an administrator should browse to **Phone Tracking > SNMPv2 Settings for SNMPv2 or SNMPv3**. To allow CER to poll Unified CM via SNMP, the administrator must configure SNMP within Unified CM. An administrator should navigate to **Cisco Unified Serviceability > SNMP** and choose either **V1/V2 > community string** or **V3 > User**. Currently, Unified CM supports SHA authentication and AES128 encryption. An example of an SNMPv3 configuration inside Unified CM is shown in Figure 2-13. Because CER only performs READ operations, the Access Privileges should be set to ReadOnly.

https://t.me/learningnets

@cisco_in_persian_channel

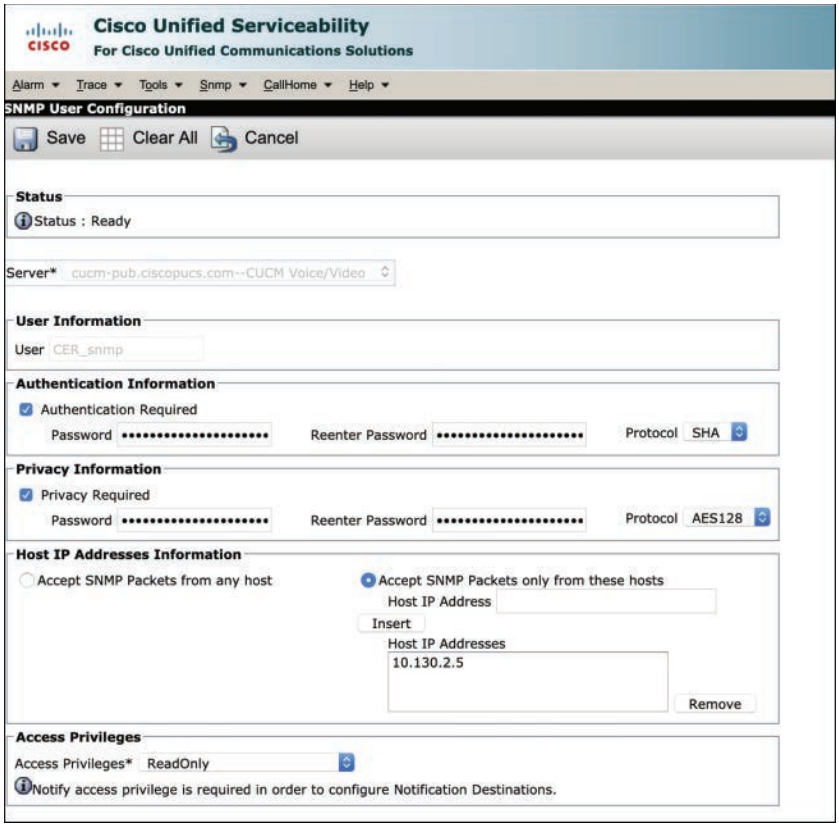


Figure 2-13 *Configuring SNMPv3 Settings Within Cisco Unified CM's Serviceability Screen to Allow for Secure Polling*

The last portion of configuration for SNMP occurs inside the switch. For a full list of supported network devices, review the CER release notes. For details about configuring SNMPv2 and/or SNMPv3 in the network devices, refer to the relevant configuration guide(s).

By default, CER queries Cisco Unified CM and network devices every 30 minutes, but this interval is customizable. This way, if a user moves a phone and is stationary for more than 30 minutes, CER is able to dynamically locate the IP phone and record the correct location if a 911 call is placed from it.

To assign ERLs within CER, an administrator should browse to the ERL Membership tab and select one of the supported options: Switch Ports, Access Points, IP Subnets, Unlocated Phones, or Manually Configured Phones. Figure 2-14 shows how an ERL named Cisco_3FL has been assigned to multiple different switch ports and how SNMP is used to append the description of the switch port to assist with ERL assignment.

https://t.me/learningnets

@cisco_in_persian_channel

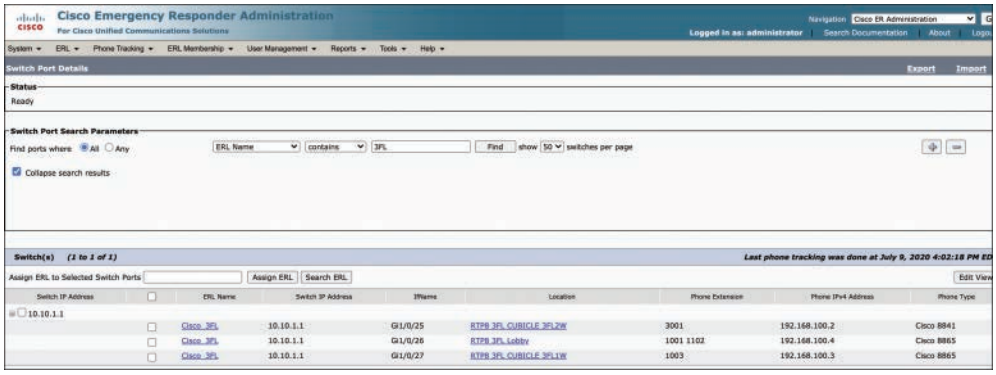


Figure 2-14 *Assigning an ERL to Multiple Switch Ports Within CER*

Note CER does not show an IP phone connected to a network switch unless there is a phone with the same MAC address inside Unified CM.

Mobile devices and wireless clients that are connected to the network via wireless access points must be tracked differently than wired devices. The biggest challenge in identifying the location of wireless devices is that these devices are highly mobile and may frequently change locations as an employee moves wireless devices around a building. Prior to certain releases of software in Cisco Unified CM and CER, wireless devices were tracked based on their IP subnet. Starting with Cisco Unified CM 11.5(1), the enhanced location tracking feature allows Unified CM and CER to accurately locate wireless devices. Starting in Unified CM 12.5(SU1), CER is able to track Jabber clients to a specific access point while they are on the corporate network.

The following wireless devices support location tracking via location awareness:

- Cisco Unified Wireless IP Phone 7925G: supported in Cisco Unified CM and CER 11.5(1) or later
- Cisco Unified Wireless IP Phone 7925G-EX: supported in Cisco Unified CM and CER 11.5(1) or later
- Cisco Unified Wireless IP Phone 7926G: supported in Cisco Unified CM and CER 11.5(1) or later
- Cisco Jabber clients: supported in Cisco Unified CM and CER 12.5(1)SU1 or later

Note Cisco Unified Wireless IP Phone 8821 does not currently support location awareness; however, this capability is likely to be supported in a future release of software code.

https://t.me/learningnets

@cisco_in_persian_channel

The location awareness feature allows an administrator to import the access points within an enterprise into Unified CM for wireless client association. Access points can be imported into Cisco Unified CM automatically or manually. Access points can be imported automatically if the customer is using Cisco's Wireless LAN Controller (WLAN). All other WLAN deployments need to manually import their access points into Unified CM.

Note Meraki APs are not currently supported.

After the access points are loaded into Cisco Unified CM, wireless clients report the access point information to Unified CM, which then adds the information to the database. Because mobile and wireless clients change their location more frequently than wired devices, CER updates any wireless clients that have moved since the last check (the default is every two minutes). Access points are associated with an ERL the same way that switches or IP subnets are. If a mobile device or client roams to an access point that has not been added into Unified CM, the device is tracked by either IP subnet or uses the default ERL.

Although a company can provide a solution for its users to manually update their physical location when outside the corporate network, that type of solution is outside the scope of this book. Given the fact that RAY BAUM's Act does not require a company to provide this service to its employees, this topic is outside the scope of this chapter.

To verify which ERLs an organization's IP phones use, you can use the ERL Debug tool. This tool can be found under **Tools > ERL Debug Tool**. Sample output from the ERL Debug tool is shown in Figure 2-15.

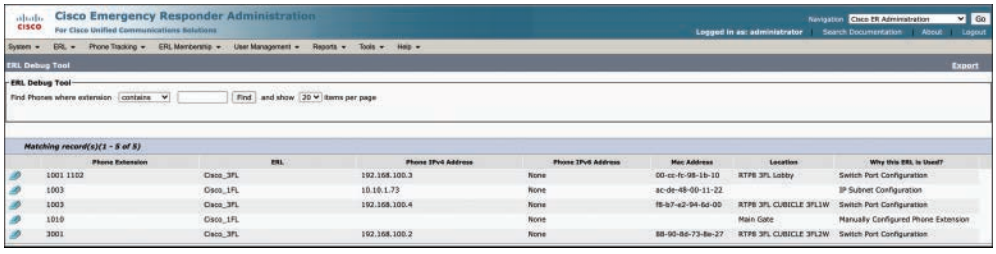


Figure 2-15 *Using the ERL Debug Tool to Analyze Which IP Phones Are Assigned to Specific ERLs*

Call Routing Considerations

The primary workload performed within CER is to associate a device to an ERL and ELIN. But the delivery of the location information to external emergency services is dependent on how an organization routes a call to the PSTN. Specifically, the types of PSTN interfaces that an organization uses determine the level of granularity that can

https://t.me/learningnets
@cisco_in_persian_channel

be achieved for obtaining the location of the emergency caller. For example, if the organization's interface to the PSTN is a POTS line in a voice gateway, those calls use a static ANI (caller ID) because POTS lines do not allow for digit manipulation. The reason is that analog POTS lines have their ANI set by the carrier and do not allow the enterprise to manipulate the calling party number. If an organization needs to deliver an emergency caller's location using dynamic ANI, it needs to use one of the following voice gateway options:

- ISDN Primary Rate Interface (PRI)
- Session Initiation Protocol (SIP) trunk
- Centralized Automatic Message Accounting (CAMA)

All three of these gateway options allow the customer to send a calling party number that allows the PSAP agent to have the correct address of the calling party displayed on their desktop. Although CAMA is a technology used to deliver called party information into the emergency calling system, customers rarely use these interfaces. PRI and SIP interfaces are the preferred connection type to deliver calling party number information for emergency call routing. In a typical scenario, the PSAP's Automatic Location Information database is populated with a physical address to dispatch responders to. For customers who want to share more specific information about the location of a 911 caller, CER supports use of Private Switch Automatic Location Information (PS-ALI). PS-ALI is an enhancement to E911 response systems because it allows organizations to provide more specific address and location information, which can be easily imported into the service provider's ALI database. If an organization is interested in PS-ALI service, it must request the service from the local exchange carrier or service provider so that the provider can provision it. To provision the PS-ALI service, the organization typically must pay a one-time fee to the local exchange carrier or service provider.

Jurisdictional Issues

Continuing with Anthony's task to design ACME's emergency call routing, he now understands that he must take into account the location of the PSTN gateways used by ACME and review jurisdictional boundaries of the PSTN gateway interfaces. As Anthony evaluates emergency call routing over PRI and SIP trunks, he learns that both services accept a 911 call, and both route the call to a PSAP. However, there is one significant difference between a PRI trunk and an SIP trunk, which is that PRIs are limited to the local regional PSAP, whereas SIP trunks can deliver an emergency call to any PSAP across the country. Anthony finds out that ACME's UC services are based in San Francisco, California. Additionally, the San Francisco location is where all UC and the PSTN PRI gateway services are located for all of the West Coast facilities for this company. As Anthony deploys the emergency call routing services, his testing goes smoothly, and when any caller from the San Francisco site calls 911, the call is routed out the PRI gateway in San Francisco, and the PSAP can correctly identify their location. This success is short-lived, however, when he tests an outbound 911 call with a user based in Seattle. Although

https://t.me/learningnets

@cisco_in_persian_channel

the Seattle user's location can be identified by CER, and the call goes out of the San Francisco PRI, the PSAP in San Francisco is not be able to identify the location of the caller. Anthony discovers that because the calling party number and address of the Seattle caller is not in the same jurisdiction as the PRI gateway, emergency services cannot be provided to callers outside the San Francisco area.

Anthony is aware that ACME is looking to move to SIP trunk services to replace its PRI trunks as part of a cost-reduction project. When investigating whether an emergency call can be delivered to a service provider over a SIP trunk, Anthony is informed that PSTN SIP providers must deliver the correct PSAP based on the calling party number. So, Anthony could leverage the same call flows as before: place an emergency call from a San Francisco caller (area code 415) that routes to the San Francisco PSAP while a caller from Seattle (area code 206) would be able to route a call to the Seattle PSAP. Additionally, Anthony has also found out that he can install a PRI into the Cisco router that is already installed in the Seattle office and use a route/translation pattern within CER to steer the call out the new PRI in Seattle. If a new PRI is installed into the Seattle router, ACME can also use Cisco Unified CM's native emergency services call handler and standard local route groups (SLRGs).

One concern that Anthony has about SIP trunks is that although the law requires SIP providers to deliver an emergency call to the local PSAP of the caller, each carrier may comply with the law differently. Therefore, if ACME changes SIP trunk providers at a later date, someone would have to have the sense to talk with the new SIP provider to make sure that emergency calls are routed correctly after they are delivered to the SIP provider. Anthony decides to take these three main options to his manager to help determine which is the best long-term option for ACME.

Cisco design guidance recommends that all devices have access to call 911 directly but recommends the use of a special partition designated for E911 to help with call routing. Care should be taken such that the calling search space applied to a phone/line has an ordered list of partitions, which include the partition containing the 911 pattern or translation. In environments where users are required to dial an external access code such as a 9 to reach the PSTN (for example, 9911), there is a potential challenge with a secondary dial tone when sending a call off-premises to an external destination. As previously discussed, Kari's Law was created so that 911 calls can be made without use of an external access code such as 9. Use of Cisco Unified CM translation patterns can be a simple yet powerful way to eliminate challenges of secondary dial tone issues and also to comply with legislation such as Kari's Law. In this case, you could use the following steps to create a predictable way of routing calls over to CER and ultimately out to the PSTN:

1. Configure translation patterns for both 911 and, if necessary, 9911. An example of how 911 calls can be normalized using a translation pattern's ability to perform "PreDot" digit stripping is shown in Figure 2-16. These translation patterns should have a CSS to reach the CTI route point responsible for call routing to CER.

https://t.me/learningnets

@cisco_in_persian_channel

2. Configure one or more CTI route points within Unified CM. They should be assigned with the numbers 911, 912 (backup), and a directory number prefix that is used for return calls from the PSAP, such as 913XXXXXXXXXX. These points should match the attributes within **CER > System > Telephony Settings**. When successfully configured, CER will register to these CTI route points.

3. Configure one or more CTI ports to help provide for on-site notifications. The on-site notification numbers on the CTI ports do not have to be reachable from the PSTN.

4. Configure Cisco Unified CM route patterns in the special partition so that the CER calls to 911 can be routed to the right PSTN gateway and, ultimately, the right PSAP. An example is shown in Figure 2-17. To reach the right gateway, customers use steering codes out of CER. Unified CM might have a route pattern of 415.911 for calls that are to use a San Francisco PRI gateway and a route pattern of 206.911 for calls that have to use the Seattle PRI gateway.

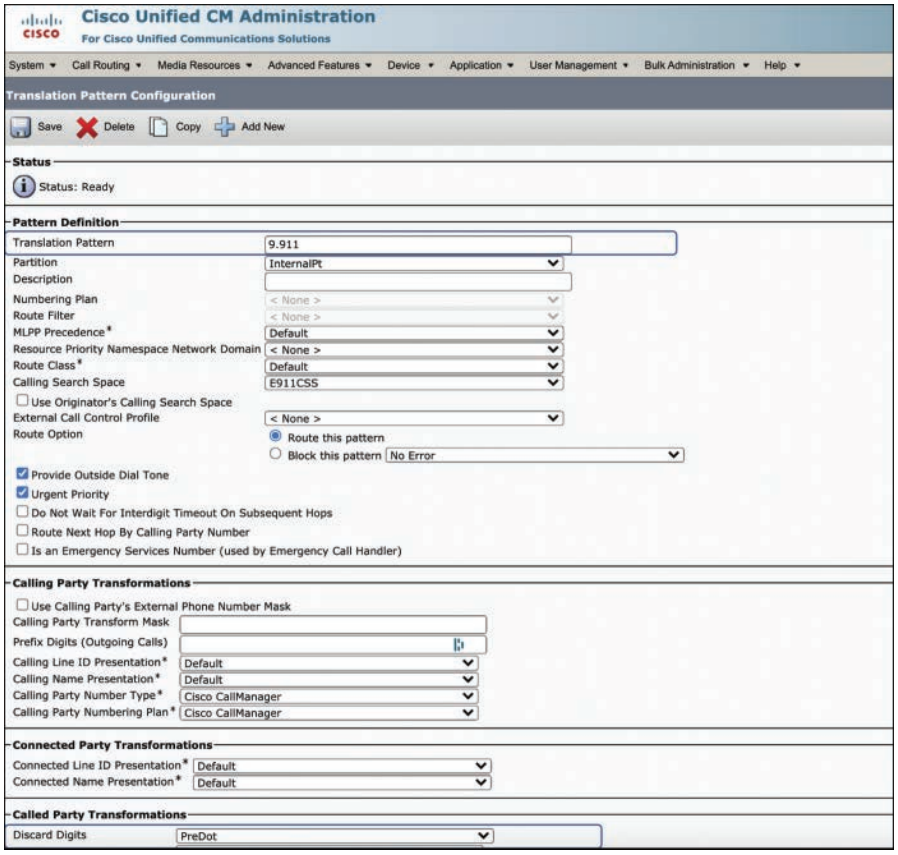


Figure 2-16 *Use of a Translation Pattern with PreDot Digit Stripping to Normalize a 911 Call*

https://t.me/learningnets
@cisco_in_persian_channel

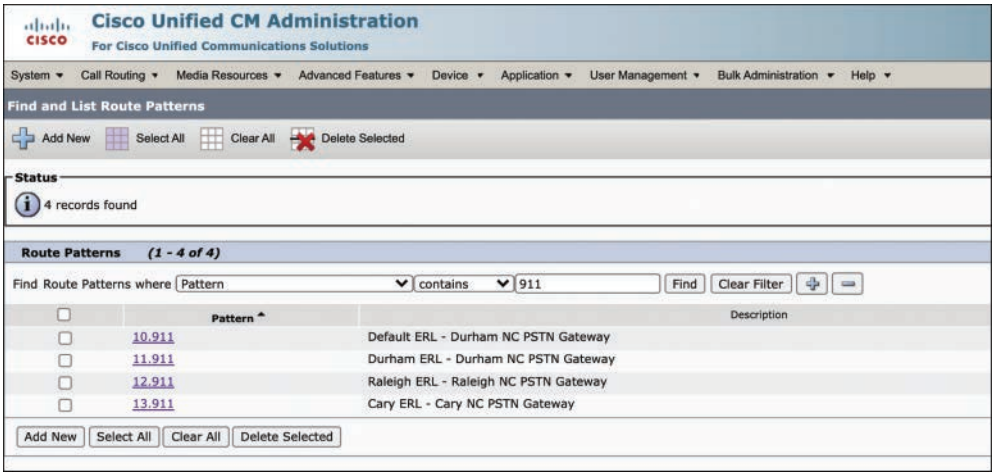
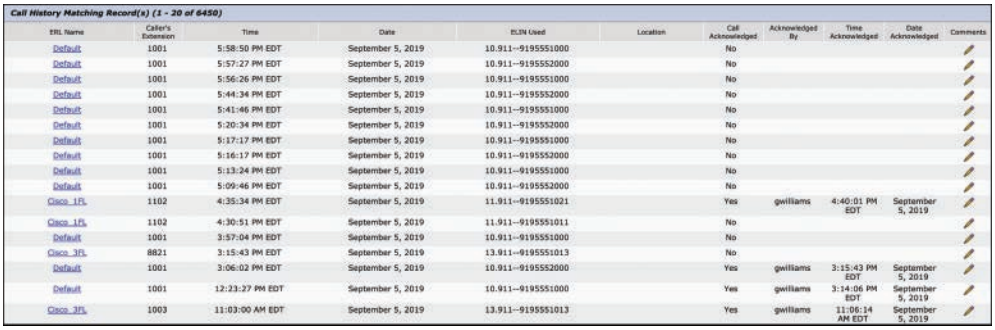


Figure 2-17 *Route Patterns Used for Outbound Calling to Local PSTN Gateways*

PSAP Callback

The last feature of E911 that we have not yet discussed is the ability for the PSAP to call back a 911 caller if a call is disconnected. As previously discussed, a CTI route point is used so that calls can be redirected to CER. In step 2 in the preceding section, we included the definition of a CTI route point with a directory number of 913XXXXXXXXX. For the PSAP to call back a user, Unified CM must identify the return call from the PSAP and deliver it to CER to reach the original caller. To do this, the administrator must define a translation pattern in Unified CM for all ELINs so that a 913 prefix can be added to the original 10 digits of the calling party; this way Unified CM will route that call to CER and ultimately the original caller. The translation pattern can use a mask for contiguous blocks of ELINs (91955510XX) or be specific (i.e., 9195551011) for non-contiguous number blocks or different locations such as Seattle. After a call is redirected back to CER, CER strips the 913 portion of this pattern and then does a search for the ELIN/Callback number in CER's call history. At this point, CER transfers the call back to Cisco Unified CM with the directory number of the endpoint (phone) that made the 911 call. An example of CER's call history is shown in Figure 2-18 to help depict sample E911 call flows. A sample diagram for an outbound call flow is shown in Figure 2-19. A sample diagram for an inbound call flow is shown in Figure 2-20.

https://t.me/learningnets
@cisco_in_persian_channel



Call History Matching Record(s) (1 - 20 of 6450)

ERL Name	Caller's Extension	Time	Date	ELIN Used	Location	Call Acknowledged	Acknowledged By	Time Acknowledged	Date Acknowledged	Comments
Default	1001	5:58:50 PM EDT	September 5, 2019	10.911--9195551000		No				
Default	1001	5:57:27 PM EDT	September 5, 2019	10.911--9195552000		No				
Default	1001	5:56:26 PM EDT	September 5, 2019	10.911--9195551000		No				
Default	1001	5:44:34 PM EDT	September 5, 2019	10.911--9195552000		No				
Default	1001	5:41:46 PM EDT	September 5, 2019	10.911--9195551000		No				
Default	1001	5:20:34 PM EDT	September 5, 2019	10.911--9195552000		No				
Default	1001	5:17:17 PM EDT	September 5, 2019	10.911--9195551000		No				
Default	1001	5:16:17 PM EDT	September 5, 2019	10.911--9195552000		No				
Default	1001	5:13:24 PM EDT	September 5, 2019	10.911--9195551000		No				
Default	1001	5:09:46 PM EDT	September 5, 2019	10.911--9195552000		No				
Cisco_1FL	1102	4:35:34 PM EDT	September 5, 2019	11.911--9195551021		Yes	gwilliams	4:40:01 PM EDT	September 5, 2019	
Cisco_1FL	1102	4:30:51 PM EDT	September 5, 2019	11.911--9195551011		No				
Default	1001	3:57:04 PM EDT	September 5, 2019	10.911--9195551000		No				
Cisco_3FL	8821	3:15:43 PM EDT	September 5, 2019	13.911--9195551013		No				
Default	1001	3:06:02 PM EDT	September 5, 2019	10.911--9195552000		Yes	gwilliams	3:15:43 PM EDT	September 5, 2019	
Default	1001	12:23:27 PM EDT	September 5, 2019	10.911--9195551000		Yes	gwilliams	3:14:06 PM EDT	September 5, 2019	
Cisco_3FL	1003	11:03:00 AM EDT	September 5, 2019	13.911--9195551013		Yes	gwilliams	11:06:14 AM EDT	September 5, 2019	

Figure 2-18 *Web Interface Inside of CER Used to Show 911 Call History and Provide Compliance Monitoring*

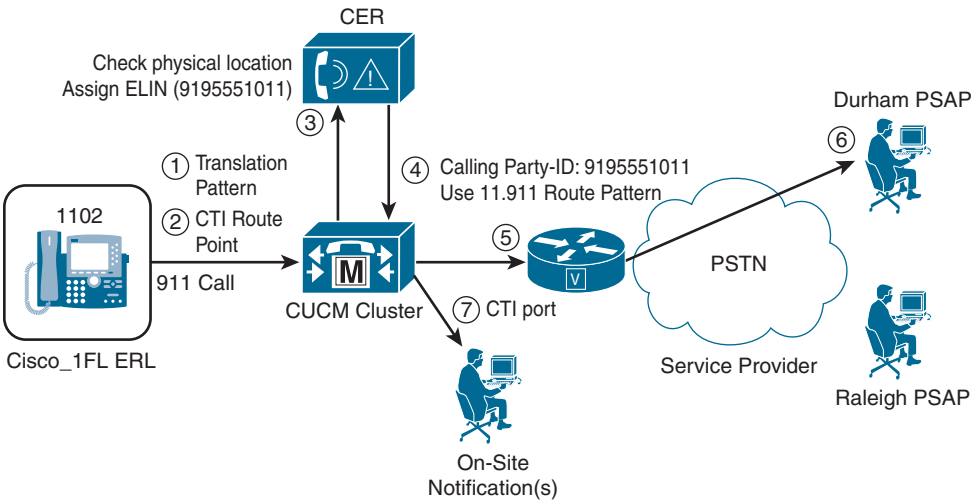


Figure 2-19 *Outbound E911 Calling Logic*

When you're designing a company's ERL and ELIN usage, you need to understand two concepts about ERLs and ELINS. First, an ELIN is used to represent only an ERL, not a user. The separation of ELINs from being a regular user's direct inward dialing (DID) number ensures that calls to an ELIN number will be handled as an emergency callback number versus a regular call to a user. The second concept is that an ELIN can be associated only with a single ERL, whereas an ERL can be associated with multiple ELINs. Having multiple ELINs assigned to a single ERL is beneficial in case multiple employees report an emergency at the same time. As previously shown during ERL creation, in Figure 2-10, CER supports the ability to assign multiple ELINs to a single ERL. Because there is a cost for DID numbers, which are required for ELINs, an organization may want

https://t.me/learningnets
@cisco_in_persian_channel

to limit the number of ELINs that are assigned to an ERL. Cisco recommends using at least two ELINs per ERL.

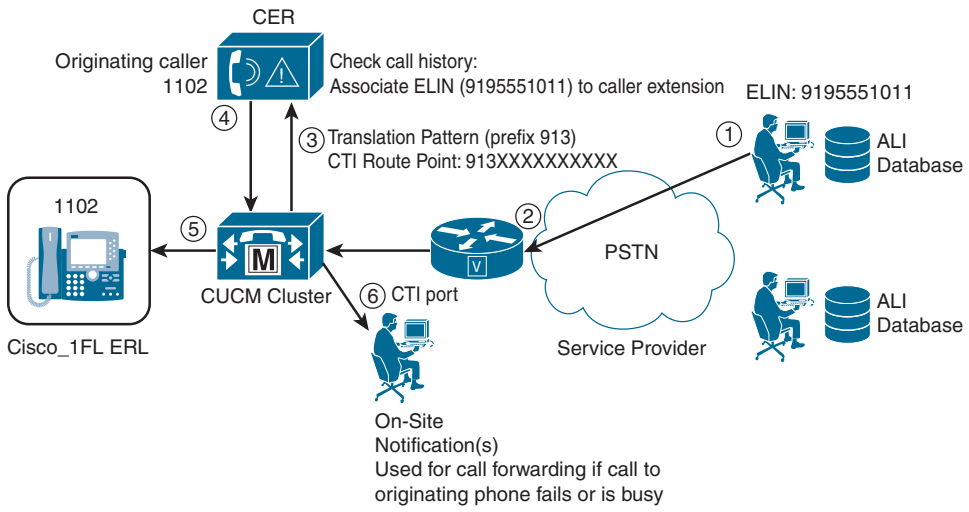


Figure 2-20 *Calling Logic for Inbound PSAP Callback*

CER can reuse ELINs and have them reassigned to another ERL to help minimize ELIN/DID usage. Any time an ELIN is added, deleted, or changed, the company must update its PSAP with the new dispatch information. In a scenario in which two ELINs are assigned to an ERL, the first two callers are uniquely assigned to each ELIN. If a third person in the same location calls 911 to report an emergency, the first caller's extension is replaced by the third caller's extension. When CER reassigns an ELIN, it can keep track of the emergency calls that are placed for up to three hours. For additional considerations regarding the design for emergency services, Cisco has provided a section called "Emergency Services" under "Call Control and Routing" in the Solution Reference Network Design (SRND) guide. You can visit this page along with administration guides for CER and Cisco Unified CM for more detail. To access the SRND, visit www.cisco.com/go/srnd.

Management, Verification, and Compliance

After integration and configurations are completed on Cisco's call control platforms (CME, Unified CM) and CER, it is highly recommended that you verify all functionality by conducting tests of 911 calls from various emergency response locations. When placing test 911 calls, it is recommended that you communicate with the PSAP that you are testing a new E911 solution. Explain that you need help determining whether the PSAP sees the correct ELIN information and ask how it would like you to share information

https://t.me/learningnets
@cisco_in_persian_channel

that can help it populate its ALI database so that future 911 calls can be populated with the correct physical locations. As previously discussed, CER supports use of Private Switch ALI (PS-ALI) records, which can be exported to service providers who support PS-ALI in NENA 1.0, 2.0, or 3.0 formats. Cisco also provides an ALI formatting tool so that ALI records can be imported by service providers that do not leverage NENA formatting. Cisco recommends that these tests occur on a regular basis to verify that changes do not negatively impact the accuracy of the E911 system.

To make changes within CER, Cisco provides customers with role-based access controls, so users who have accounts created locally within CER or remotely (in Cisco Unified CM) can have access to the necessary resources. By default, the system provides the following seven roles. The following list details the type of access to the system that these roles provide:

- **CER System Admin:** Has access to all system administration pages
- **CER Serviceability:** Has access to all serviceability pages
- **CER Admin Utility:** Has access to all administrative utility pages
- **CER Network Admin:** Has access to a LAN switch, SNMP settings, and Cisco Unified CM pages
- **CER ERL Admin:** Has access to all ERL-related pages
- **CER Audit Admin:** Has access to the audit page in serviceability
- **CER User:** Has access to the user pages

For a full list of the features that are included in these roles, see the CER administration guide.

Separate from the CER administrative page, Cisco provides a user dashboard to receive web-based alerts for monitoring and verifying 911 calls. This page is updated immediately after the call has been dispatched back to Cisco Unified CM for routing the call to the PSTN Emergency Services. UC administrators can access the CER user page by navigating to the Cisco ER User screen or through https://[IPADDRESS]/ceruser/. As shown in Figure 2-21, the page enables you to acknowledge the awareness of 911 calls for compliancy purposes. After a 911 call is acknowledged, a CER user can add comments to the call record, including what happened and any action performed by the company. This information can be used to audit the company response to the emergency when evaluating what actions were taken by the company and also to ensure compliance to an organization's PPDR policy.

https://t.me/learningnets

@cisco_in_persian_channel

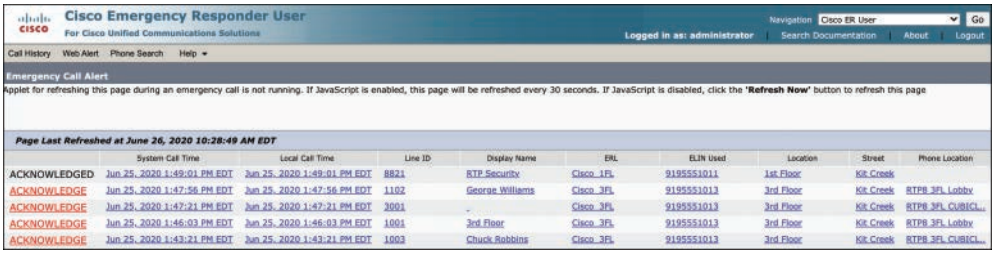


Figure 2-21 *Web Alerts Provided by Cisco Emergency Responder for Compliancy*

Additional Life and Safety Solutions

In addition to Cisco's E911 solutions, Cisco has partnered with companies such as Intrado (formerly West) and RedSky to help organizations enhance E911 services. As an example, Intrado and RedSky provide the capability to host ALI records in their own Dynamic ALI databases. Organizations that are considering PS-ALI services may be able to use this capability to save costs or avoid fees/contracts for PS-ALI services. Intrado and RedSky solutions may also be useful for organizations that want to provide E911 services for their workforce while located off-premises. In this scenario, users can update their locations by using a display screen so that they can be routed to the correct PSAP.

In addition to E911 solutions, additional life and safety solutions integrate tightly with Cisco's UC infrastructure. These solutions are purposely built to react and respond to certain types of events that are documented within any organization's safety and security policy. Singlewire InformaCast is an example of a software-based solution that can help an organization provide emergency alerting and responses. An example is an SIP-based panic button, such as one manufactured by Cyber Data as shown in Figure 2-22. Because the panic button supports SIP, it can register to Cisco Unified CM and be integrated with InformaCast Software to generate alerts. These types of panic buttons can be placed within reception areas or underneath a user's desk to signal that assistance is requested without alerting an offender or intruder. When Cisco Unified CM is integrated with Singlewire's InformaCast solution, speed dials can be configured on existing Cisco phones to generate XML messages that can be pushed to the screen of other wired IP phones, wireless IP phones, desktops, and mobile devices as well.

Whenever deploying systems such as InformaCast, you may need to implement some prerequisites to realize the full benefits. An example of what needs to be checked is whether the network supports IP Multicast. Multicast is required because it allows you to push out an audio stream to thousands of network devices simultaneously. If the network doesn't support Multicast, there are ways to convert multicast streams to unicast streams, but they come at an additional cost that an organization may not want to incur with short notice.

https://t.me/learningnets
@cisco_in_persian_channel

Figure 2-22 *A Panic Button That Can Be Used to Alert Emergency On-Site Personnel*

Computer-Aided Dispatch

Communications systems such as InformaCast or Instant Connect (the enterprise software platform from Dillon Kane Group) can also enable computer-aided dispatch to integrate with devices that emergency responders already have on them, such as land mobile radios (LMRs) or mobile devices. Instant Connect offers a versatile interoperable communications environment in which first responders can use any device they want (e.g., smartphones, tablets, radios, laptops, IP phones, or POTS) to connect in flexible talk groups, which are voice channels. Instant Connect, which was originally engineered by Cisco after the events of 9/11, is meant to help keep team members connected and communicating as they proactively respond to events (either scheduled or unplanned). This connectivity helps team members maintain situational awareness and drive more efficient and productive operations. This solution can provide benefits to life safety when deployed in any venue including campus, energy, mining, manufacturing, transportation, or retail environments.

Consider a scenario in which a large college campus has local police or security guards who are responsible for providing security for the campus environment. Rather than placing separate calls to the local police, security teams, or off-site to the PSAP, an Instant Connect Dispatch Console, such as the one shown in Figure 2-23, can be used to link different types of groups together, using any devices they choose, in an inclusive team communications environment. This solution helps simplify the real-time distribution of information to help respond to an incident faster and more proficiently. Given

https://t.me/learningnets
@cisco_in_persian_channel

the types of incidents that have occurred at schools, such as those with active shooters, these group communication solutions have the potential to save hundreds of lives by reducing the response times by just a few minutes.

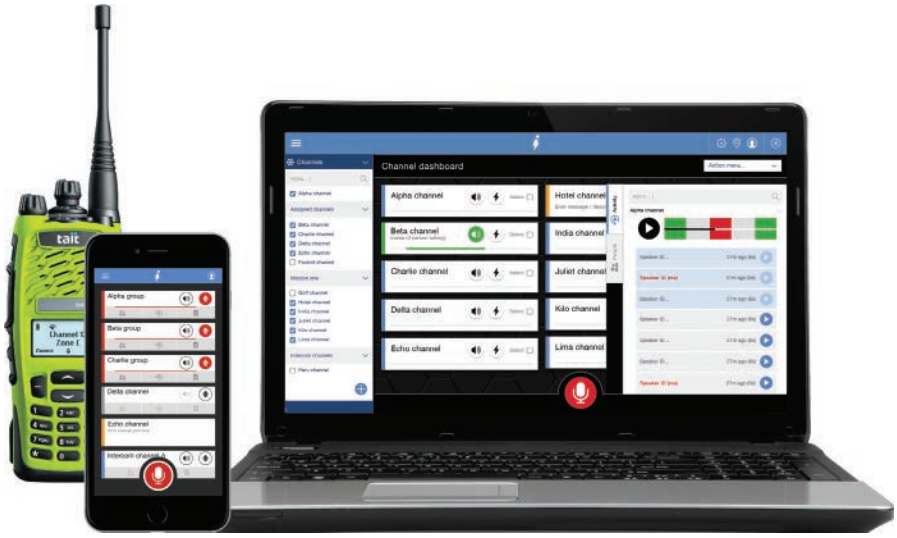


Figure 2-23 *A Computer-Aided Dispatch Console by InstantConnectNow*

Summary

The goal of physical security within an organization is to protect the safety of the employees, data, and services of an organization. Addressing physical security is not something that is done once and then forgotten about. Organizations that continuously follow the simple methodology of prepare, prevent, detect, and respond have the best chance of providing physical security. Physical security controls have a large impact on the confidentiality, integrity, and availability of data and services that an organization provides its users. This is especially true in the realm of Unified Communications, in which users need to have access to services (availability) to use them and trust that they are secure (confidentiality); otherwise, users will not use services provided by the UC infrastructure.

To ensure that there are not any single points of failure in an organization, a physical security approach that includes defense in depth continues to be the most effective way to ensure that the organization is less susceptible to physical threats. In this approach, if a single control is bypassed, the organization can still protect itself against one or more attacks. This includes providing physical security controls within a building and all of its infrastructure, such as a cable plant, power plant, and its server infrastructure, which in turn helps protect the UC infrastructure.

https://t.me/learningnets
@cisco_in_persian_channel

Life and safety considerations should be an integral part of an organization's physical security plan. Solutions that provide enhanced 911 capabilities, such as Cisco Unified Communications Manager and Cisco Emergency Responder, are key to enhancing life and safety through timely emergency response services. Additionally, these solutions help organizations comply with state and regulatory obligations that have been passed to increase life safety within states and municipalities.

Cisco offers a number of E911 solutions that can help organizations meet regulatory obligations. Cisco's UC Architecture and Ecosystem further allow customers to integrate and incorporate solutions from other vendors such as Intrado, RedSky, Singlewire InformaCast, and InstantConnectNow. When combining these solutions, organizations have the capabilities to communicate and respond to incidents in a timely fashion.

Additional Resources

www.cisco.com/c/en/us/solutions/collateral/data-center-virtualization/unified-computing/white_paper_c11-680202.pdf

www.fcc.gov

www.911.gov/

www.nena.org/

www.singlewire.com

www.instantconnectnow.com/

https://t.me/learningnets

@cisco_in_persian_channel

Chapter 3

Security Through Network Fundamentals

In this chapter, we discuss the dependencies that Unified Communications (UC) systems have on the network while also highlighting the security features that can be implemented to provide additional security to the UC environment. In its most simple form, a network's purpose is to help connect things. In the case of most organizations, these things include personal computers, printers, phones, and mobile devices. To enhance a network's capabilities, various features can be enabled to help simplify the user experience. In this case, there must be a balance to make sure that the process of how the different types of devices connect to the network is not oversimplified. Otherwise, it is security that is often left out.

A sophisticated attacker understands how an organization's critical services are built and also understands how to leverage weaknesses in the architecture to launch attacks. While implementing various protocols and features, Cisco provides many enhancements that can be beneficial to network security as well as Unified Communications security. By the end of this chapter, you should have an increased understanding of how the network can be used to safeguard against the most common threats used against the various protocols that exist within a Unified Communications environment. This chapter also provides a fundamental approach for increasing the amount of security on the network to protect against common threats and different types of attacks.

Working with ACME's Network Team

A recent network audit has determined that ACME has failed to maintain compliance with the Mechanized Equipment Specifications and Standards. ACME leadership considers this a serious issue. If the company is unable to resolve network issues in 90 days, it could be penalized financially, and it could possibly lose some of its existing customers to competitors.

https://t.me/learningnets

@cisco_in_persian_channel

ACME's Chief Operating Officer (COO), Dr. Nicholas Fury, has invited all of the IT leaders to a series of meetings to determine the options that they can take to gain compliance to industry standards in a short and operationally effective way. Dr. Fury has also invited leaders from business units to a series of meetings to develop a list of requirements and priorities so that the IT leaders can develop a strategy for modernization efforts that will help business units increase sales and productivity to meet ACME's growth targets. Business unit leadership wants to take advantage of this opportunity by deploying additional security solutions that would allow the company to collaborate with business partners and integrators in a secure fashion.

Network leadership seems to be interested in implementing both network virtualization and network access control solutions to improve the ability to segment the network and strengthen the security at the edge of the network. Fortunately, several members from the UC team have a network background and have established good working relationships with the network and data center teams, so they are willing to work together if doing so means getting back into compliance in a more streamlined fashion.

ACME's newly hired UC administrator, Anthony Starke, is responsible for articulating the network requirements for the UC environment and making recommendations to get back into compliance. Anthony is a bit overwhelmed with this challenge because he does not have a lot of real-world experience with network security, so he has scheduled a series of planning meetings with the network team to collect details about the proposed security enhancements. Anthony believes that this will help him understand which security enhancements are being proposed, allow him to recommend specific features for ACME's UC environment, and develop a better relationship with the network team members.

After an interview with the manager of the network team, Anthony has discovered that not only has ACME failed a recent network audit but that ACME's network was hacked. As part of the network audit, ACME decided to hire a team of penetration testers to discover how much security the network had in place. One of the penetration testers was able to gain access to the network through MAC address spoofing. According to the reports, a MAC address was obtained from the IP phone located in the lobby of the building. After changing the MAC address of his laptop, unplugging the IP phone, and plugging his laptop into the same network port, the penetration tester had access to the entire voice VLAN.

According to network team members, they had only recently implemented Cisco Identity Service Engine (ISE), so they were in the initial stages of deploying 802.1x to a portion of ACME's network. To make sure that the UC environment was functional, they temporarily decided to use MAC Authentication Bypass (MAB) until they were ready to roll out 802.1x for the UC environment. Anthony has been asked to prepare a briefing that helps the network team understand how to further prevent MAC spoofing attacks and to provide a status update for which IP phones support 802.1x, what methods can be used to support 802.1x, and where IP phones are located across ACME's network.

Informally, Anthony is wondering whether MAC spoofing attacks will continue to be a risk that he needs to worry about until he can get funding to replace the older IP phones. In the

https://t.me/learningnets

@cisco_in_persian_channel

meantime, he decides to go back to his UC team to figure out what challenges he should expect to encounter when ACME's IP phones will use 802.1x authentication to join the network.

Questions that you should ask:

1. What network security is currently in place to protect the UC environment?
2. What network features help increase the availability of the UC environment?
3. What mechanisms exist to provide continuous monitoring of the network?
4. What authentication and authorization policies are in place for UC endpoints?

Introduction to Network Security

After physical security, the second layer of defense for the UC infrastructure is the network. Following the Open Systems Interconnection (OSI) model, the various layers include Data Link, Network, and Transport. To best secure connectivity across these layers, experts have traditionally recommended use of defense-in-depth principles. The recommendation is no different when securing UC applications inside an organization. Cisco recommends that security be implemented at the edge of the network, starting at the access layer. From the access layer, organizations can continue to extend security into the distribution layer or layer on additional security features across the rest of the network and at the network perimeter. When the system is secured properly, organizations can seamlessly and securely connect to services in a cloud environment or allow remote teleworkers to connect to local resources. We address these topics in more detail in Chapters 11, "Securing the Edge," and 12, "Securing Cloud and Hybrid Cloud Services."

Using the access layer as a starting point for implementing security allows organizations to minimize the attack surface without encroaching on the availability of the UC applications, which reside in the data center. A methodology and practical approach for implementing security in the network for UC involves a three-step approach that involves

1. Segmentation (logical)
2. Secure network access
3. Security features

This security approach enables an organization to secure the environment while maintaining optimal performance. The idea is to develop a modular approach, which allows for the addition of security anywhere in the network while minimizing complexity and not interfering with operations. Extending security into the rest of the network is based on the existing network architecture and which types of network devices are used to support the different layers in the network, such as in the distribution layer, and also the devices used to support server infrastructure, which typically resides in the data center.

https://t.me/learningnets

@cisco_in_persian_channel

Segmentation

A best practice in today's networks is to provide logical segmentation through the use of virtual LANs (VLANs). Within a UC environment, this approach is implemented on access switches at layer 2 in the form of voice VLANs and also at layer 3 to prevent attacks on an IP subnet by applying access control lists (ACLs) on VLAN interfaces. By segmenting UC and data traffic, organizations can reduce the attack surface to which attackers have access. Ideally, attackers would have access only to the segment they are connected to. Segmentation also helps with simplifying the network by allowing an administrator to view and distinguish the different types of traffic that traverse specific interfaces in and out of the logical network. By segmenting the network (virtually) into smaller networks (in this case, VLANs), administrators can easily control the types of traffic that are permitted across different layer 2 and layer 3 interfaces to prevent unwanted traffic.

Dedicated UC and data segments also help minimize the size of a broadcast domain, which reduces the amount of traffic flooding across a network. Within a single VLAN or broadcast domain, there is potential for devices to generate (and flood) the network with traffic, which can be problematic from a security perspective. We discuss this issue later in the section on security features. Large broadcast domains also have the potential to impact the performance of the network. When segmenting a network, an organization is better insulated from a broadcast storm or denial-of-service (DoS) attack. This way, only a portion of the network can be disrupted if an attack were to occur on a particular VLAN. Cisco best practices currently recommend limiting the size of a logical segment to 256 devices, if possible, and not to exceed 512 devices.

When an organization uses the latest IOS-XE platforms (e.g., 16.9), the data and UC networks can be logically segmented by applying configurations for data and voice VLANs on a single interface. The following example shows the syntax:

```
switch# conf t
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# int gig1/0/10
switch(config-if)# switchport mode access
switch(config-if)# switchport access vlan 10
switch(config-if)# switchport voice vlan 100
switch(config-if)#
```

At this point, we have discussed segmentation at layer 2 of the OSI model. The next section provides additional detail about layer 3 segmentation using Virtual Route Forwarding (VRF).

VRF is a way to segment the network segmentation at layer 3 of the OSI model. VRF was originally designed for service providers to essentially provide existing and new customers with their own virtual private network (VPN) across the service provider's physical infrastructure. This solution allowed service providers to take on new customers with minimal costs, without having to worry about a customer's existing IP scheme or

https://t.me/learningnets

@cisco_in_persian_channel

infrastructure. Within a VPN built with VRFs, each VRF instance has a separate layer 3 routing table. IP packets from one VRF are intentionally isolated from other VRFs.

If the flow of UC traffic isn't taken into consideration, use of VRFs may unintentionally create user experience issues. These issues could be related ACLs inside a fusion router/firewall that break functions of a UC environment, such as screen sharing. Additionally, their use could cause issues with performance by introducing latency, jitter, and/or packet loss as traffic is routed between interfaces on a fusion router or firewall during a time in which the network has high utilization. As an example, an organization decided to segment the network with VRF. To minimize the attack surface, it decided to create VRFs for each line of business (sales and marketing, research and development, support) and also for each device type (e.g., printer, IP phone, Internet of Things [IoT] sensor). The belief was that creating specific groups would simplify how ACLs are applied on the network, with a fusion firewall that supports group-based security policy. During testing, engineers were able to print, make phone calls between IP phones, and access shared resources. Unfortunately, network administrators didn't consider all of the network requirements for a UC client such as Jabber. Therefore, they unintentionally created firewall policies that prevent Jabber calls between IP phones.

Certainly, changes can be made to network policy, especially to help resolve problems as they arise. Some questions that a UC administrator can ask to proactively prevent issues with VRF include but are not limited to

- Will a VRF disrupt the user workflow or calling patterns?
- Where will the inter-VRF routing take place to ensure a quality user experience between UC endpoints (e.g., Jabber, IP phone, video device)?

It is important to realize that care should be taken when designing segmentation in the network to ensure that the network supports the desired features within a UC environment. It is also important to make sure that the network does not force UC traffic to flow across the network in ways that are less than optimal to help avoid the addition of latency, delay, and jitter.

At this point, we have discussed basic segmentation as one approach for providing security at different logical layers of the network so that organizations can apply security controls at whichever level of granularity is necessary to comply with security policies. Basic segmentation by itself, which logically separates different types of devices or business entities, can be referred to as segmentation at a "macro" level. As part of a defense-in-depth approach, another layer of segmentation is possible to further reduce the attack surface. This approach is known as micro segmentation, is discussed next.

Micro Segmentation

Micro segmentation has many security benefits. As previously discussed, it can be very effective in preventing specific unwanted traffic flows within a VLAN segment. One specific use case of micro segmentation may be to restrict the spreading of malware laterally

https://t.me/learningnets

@cisco_in_persian_channel

within a VLAN (that is, spreading of malware to neighboring devices) by only permitting specific protocols that are expected across a network. As an example, a micro segmentation policy would allow a network administrator to instruct the network to allow only PCs with a UC client, such as Cisco Jabber, to establish VoIP calls with other PCs running the Jabber client over specific TCP/UDP ports that have been designated for UC. This type of granularity allows UC clients such as Jabber to function exactly as a user expects by supporting all of the UC features, such as instant messaging, VoIP/video calling, and content sharing. It also helps meet security requirements to minimize the attack surface on a network by denying specific protocols that may traditionally be used by worms, malware, or an attacker.

Customers who previously had requirements to restrict access within a network segment such as a VLAN have traditionally been limited to either VLANs, with VLAN ACLs (VACLs), or private VLANs. Two different modern architectures that provide micro segmentation capabilities are Cisco Software-Defined Access (SDA) and Cisco TrustSec, which offer security policy based on scalable groups. The next section provides an introduction to the SDA solution while also highlighting how TrustSec can be incorporated into SDA to obtain a deeper level of granularity for restricting traffic flows within a network.

Cisco SDA is a solution within the Cisco digital network architecture (DNA) that provides software-defined networking for the campus environment. SDA provides network security by facilitating end-to-end segmentation of network traffic between users, devices, and applications. A software-defined network, providing centralized management, allows organizations to enable security features on a more aggressive basis because there is less of a burden for enabling security across network devices on a device-by-device basis, which is often a reason that networks are not as secure as they could be. SDA also provides organizations with a means and methodology for increasing visibility of network traffic and applying network policy to wired and wireless network devices (such as switches and access points) in an automated manner as a user or devices move around a network. Using the earlier example with Jabber, SDA would ensure that the security policy that has been assigned to a user with a Jabber client will stay with that user while roaming between different places in the network (e.g., desk, conference room, cafeteria).

This type of security approach requires a paradigm shift when compared to a traditional approach of managing ACLs that are applied to network devices to protect specific IP subnets. Using an architecture, such as SDA, security controls are no longer based on IP subnets. Security controls are associated with the identity of a user or device. This is why a policy can still be assigned to users as they roam around a network. The following components are required to support SDA:

- **Cisco DNA Center (DNA-C):** This component provides centralized management of network infrastructure; it is used to create policies (e.g., security, QoS) and automate provisioning of software features and images across network devices. It also provides visibility into network incidents, network telemetry, and analytics.
- **Cisco Identity Services Engine (ISE):** This component integrates with DNA-C to provide access controls (e.g., ACLs, dynamic VLAN assignments), group-based

https://t.me/learningnets

@cisco_in_persian_channel

policy, and policy enforcement. It integrates with external repositories such as Active Directory for authentication and authorization.

- **Network infrastructure (wired and wireless):** This infrastructure includes network devices, such as Catalyst switches and access points.

The last element that is included inside an SDA solution is the network fabric. The network fabric is essentially a virtual network that is overlaid on top of the existing physical network with a separate control and forwarding plane. The separation of the control plane from the forwarding plane helps improve the overall performance and scalability of the solution while also simplifying policy, provisioning, and management. As an example, it allows network administrators to provide a consistent policy to a user/device as it moves around the network. To do this, DNA-C and ISE work in unison with network devices to enforce policy that either permits or denies the different types of traffic allowed across a segment of the network. Figure 3-1 depicts a sample SDA solution.

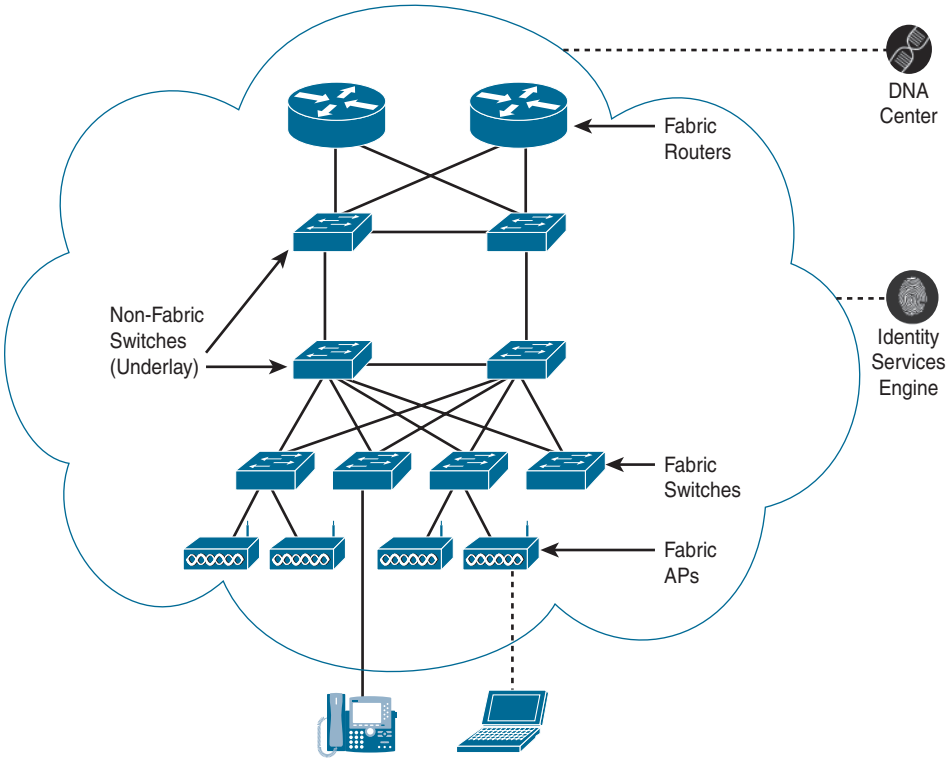


Figure 3-1 *SDA Solution Featuring DNA Center, ISE, and Network Fabric*

https://t.me/learningnets

@cisco_in_persian_channel

Now that we have introduced SDA, we must also introduce a few more key terms regarding how the SDA solution provides security:

- **Virtual network:** This network provides logical separation (macro segmentation) between devices and other virtual networks. This is analogous to a Virtual Route Forwarding technology.
- **Scalable group:** This mechanism for grouping functions or devices is based on business roles (e.g., employee, contractor, finance, printers, IP phones). Once a group is created, a unique scalable group tag (SGT) is assigned to the scalable group to provide micro segmentation. This security concept is based on Cisco TrustSec.
- **Contracts:** These are used for enforcing policies that have been created within DNA-C to specifically permit or deny certain types of traffic. They are also known as security group ACLs (SGACLs).

At the time of configuration, an administrator needs to align to the virtual networks that have been previously defined by an organization. Because we have been discussing how to create a micro segmentation policy to support UC security within DNA-C, we can use DNA-C as the central place of management and policy configuration. To follow this example, you should take the following steps:

1. Define virtual networks (e.g., ACME_HQ).
2. Define scalable groups (e.g., ACME_UC, ACME_WIDGETS, Employees, Contractors, Printers).
3. Assign scalable groups to virtual networks. This process allows access to resources within a virtual network (e.g., it allows employees and contractors to communicate with IP phones and printers).
4. Create contracts (e.g., Permit_VoiceUDP) that specify the traffic flows that are allowed within scalable groups using SGACLs.
5. Deploy the micro segmentation policy to network devices.

An example of step 3 is provided in Figure 3-2, and an access contract is provided in Figure 3-3, showing how an administrator can permit VoIP traffic on an SDA environment.

https://t.me/learningnets

@cisco_in_persian_channel

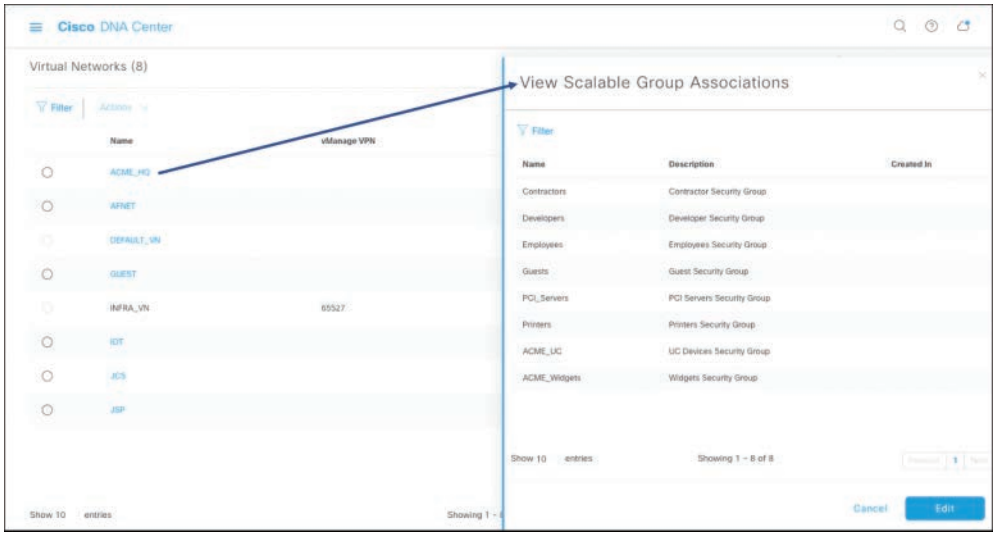


Figure 3-2 *Assigning Scalable Groups of Resources to a Virtual Network Within DNA Center*

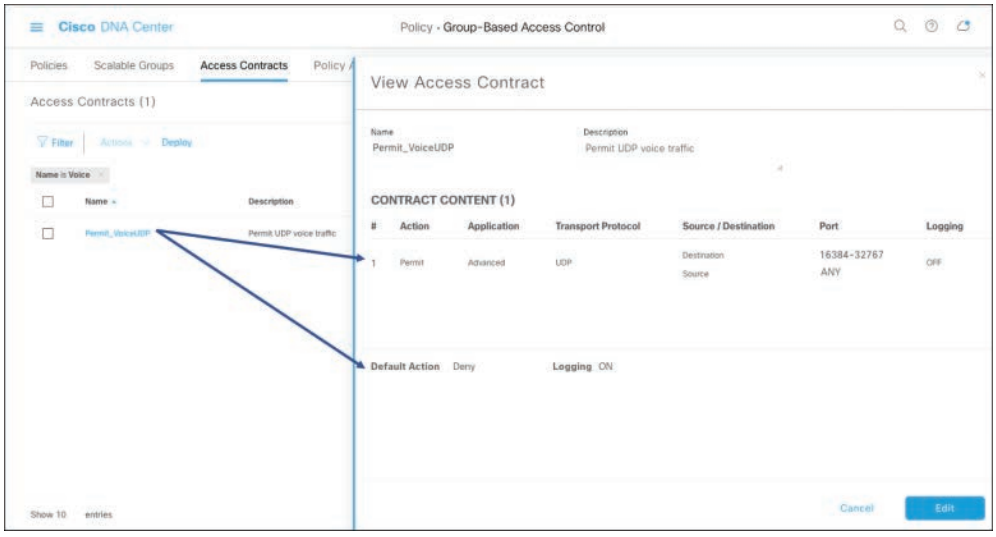


Figure 3-3 *An Access Contract Created Within DNA-C to Permit UDP-Based VoIP Traffic, While Denying All Other Types of Traffic*

https://t.me/learningnets

@cisco_in_persian_channel

Once the scalable groups and the associated contracts are defined in DNA-C, they are then also shared with ISE over a RESTful API. This setup allows ISE to be the authoritative point of security enforcement across the network. An example illustrating how ISE imports the policy from DNA-C and creates security group ACLs to enforce policy is included in Figure 3-4.

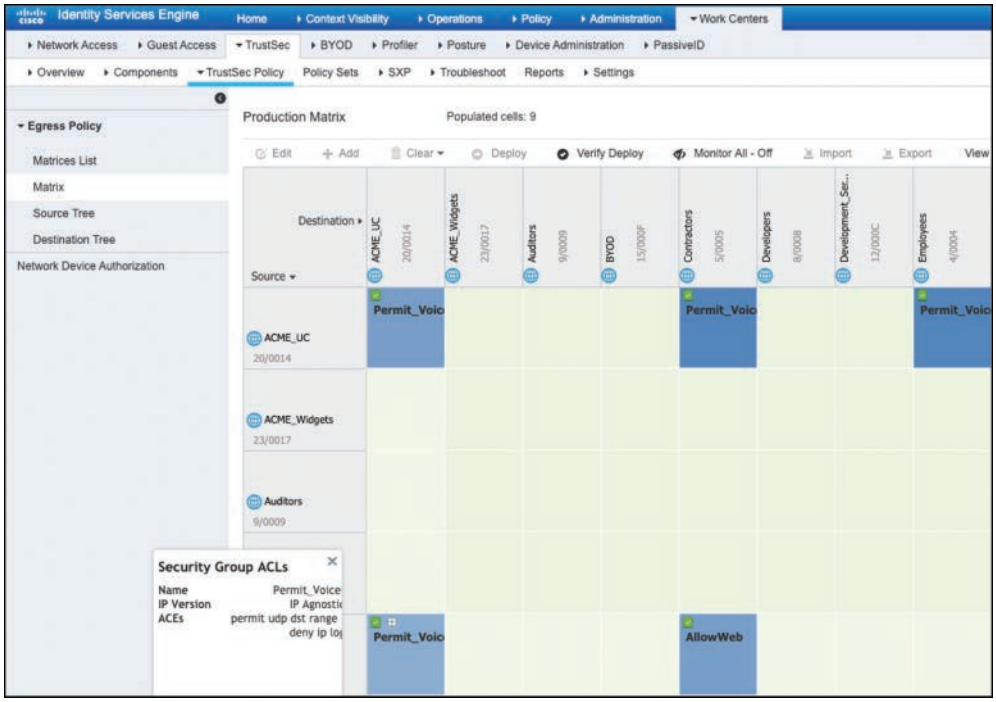


Figure 3-4 *A Micro Segmentation Policy Within Cisco ISE to Permit VoIP Traffic Between Scalable Groups*

Due to the simplicity involved in this type of workflow, organizations are able to respond to new security threats while reducing operational costs by using a centralized policy engine such as DNA-C. As the threat landscape continues to grow, organizations may need to adopt micro segmentation capabilities to gain more granular security controls instead of the traditional macro-based segmentation approach. It is beyond the scope of this chapter to cover all of the necessary steps needed to design, configure, and provision VRF, DNA-C, ISE, and the network to support SDA. For additional information about SDA, visit www.cisco.com/go/sda.

Secure Network Access

Providing security at the edge of the network is perhaps the most overlooked approach for UC security despite the power that it can bring to an organization and the simplicity involved. Put simply, if an organization can strengthen how users and UC devices join the

https://t.me/learningnets
@cisco_in_persian_channel

network, it can help reduce the attack surface that attackers have access to while making it easier to protect against UC attacks. Security at the edge of network access comes in many forms. The oldest, and perhaps most common, is through use of switchport security. The strongest approach is through implementation of 802.1x authentication. The next few sections explain in more detail how port security can help an organization and show how it is configured. After that, we transition over to 802.1x authentication and then wrap up by covering what can be considered the middle ground—MAC Authentication Bypass (MAB) and how MAB can be further secured when using Network Access Control (NAC).

Port Security

The port security feature enables organizations to specify what identities can join the network by specifying which MAC addresses are allowed. Port security has traditionally been popular because it provides an easy way for an organization to limit the number of devices that are allowed to connect to an access port and prevent shadow IT.

As an example, port security can help prevent a user from plugging in an access point that has not been previously sanctioned by the IT department to add wireless capability. Port security is also useful for protecting against attacks against the network, such as flooding the Content Addressable Memory (CAM) table on a switch with false MAC addresses to create a man-in-the-middle attack.

If an organization chooses to use port security at the edge of the network, administrators should know that Cisco switches do not support explicit configuration of MAC addresses for the voice VLAN. Therefore, administrators should consider allowing a maximum of two MAC addresses to connect to each switch port to account for the IP phone and the PC plugged into the back of the IP phone. If any type of port security is enabled on the access VLAN, dynamic port security is automatically enabled on the voice VLAN.

To enable switchport security, you need to take the following steps:

1. Specify the maximum number of MAC addresses allowed on the network:

```
switch(config)# int gig1/0/1
switch(config-if)# switchport port-security maximum 2
```

2. Specify the MAC address that is permitted for the network port(s):

```
switch(config-if)# switchport port-security mac-address
00cc.fc98.1b10
```

3. Specify what action to take if the switch doesn't recognize a MAC address that is trying to join the network:

```
switch(config-if)# switchport port-security violation restrict
```

4. Enable the switch port for port security:

```
switch(config-if)# switchport port-security
```

https://t.me/learningnets

@cisco_in_persian_channel

When the **restrict** key word is used, and a violation occurs, an SNMP trap is sent, a syslog message is logged, and the violation counter increments. The other options are to protect and shut down. When configured to protect, the switch just drops packets with an unknown source address. When configured for shutdown, the interface becomes error-disabled, an SNMP trap is sent, a syslog message is logged, and the violation counter increments.

Now that we are discussing port security, it is important to understand that a configuration that limits the specific MAC addresses that are allowed to connect to an access port is a potential issue. This type of environment assumes that the environment is static and that a limited number of phones require Moves/Adds/Changes (MAC) around the network. If the desire is to have a dynamic but yet secure environment, the organization can convert from static MAC addresses to dynamic "sticky" MAC addresses, in which the switch converts all of the dynamic MAC addresses to the running configuration on the switch. To enable port security that is dynamic, leverage the **mac-address sticky** command, as in the following example:

```
Switch(config)# int gig1/0/11
Switch(config-if)# switchport port-security maximum 2
Switch(config-if)# switchport port-security violation restrict
Switch(config-if)# switchport port-security mac-address sticky
Switch(config-if)# switchport port-security
```

This option is more flexible because sticky MAC addresses do not automatically become part of the start-up configuration file, which is the configuration used each time the switch restarts. Learned sticky MAC addresses are just added to the running configuration as shown:

```
switch# sh run int gig1/0/11
Building configuration...

Current configuration : 364 bytes
!
interface GigabitEthernet1/0/11
 switchport access vlan 20
 switchport mode access
 switchport voice vlan 100
 switchport port-security maximum 2
 switchport port-security violation restrict
 switchport port-security mac-address sticky
 switchport port-security mac-address sticky f8b7.e294.6d00 vlan voice
 switchport port-security
 spanning-tree portfast
end
```

Although effective in securing network access for the common user, switchport security has some downfalls. MAC addresses can easily be spoofed or falsified to allow

https://t.me/learningnets

@cisco_in_persian_channel

unauthorized devices onto a network. If attackers do this, they can attack the network segment(s) that they are connected to laterally, unless layer 2 VLAN ACLs (VACLs) are in place. This is one of the reasons why we have previously discussed use of segmentation in the network (macro and micro). In any regard, because organizations understand the limitations of port security, many of them are moving away from this approach and toward 802.1x authentication. The next section discusses this technology in further detail.

802.1x Authentication

The IEEE 802.1x standard method of authentication is widely considered the strongest method of authenticating users and devices to the network. In typical implementations with 802.1x authentication enabled, the access port allows only Extensible Authentication Protocol over LAN (EAPoL), Cisco Discovery Protocol (CDP), and Spanning Tree Protocol (STP) traffic to the port until an endpoint is authenticated. In addition to providing port-based authentication, an architecture that provides 802.1x offers increased visibility that may be useful for security audits, forensics, and troubleshooting. The downside of this method of authentication is that there are several dependencies on the infrastructure components. These components are not typically managed by UC administrators, so there is an element of cooperation and teamwork needed for a fully functional 802.1x solution. The basic components of an 802.1x solution include

- Authenticator (Access Switch)
- Authentication server, such as Cisco Identity Services Engine (ISE)
- Authentication database
- Client supplicant
- Public key infrastructure (PKI)

The 802.1x authenticator (access switch) helps relay authentication information over Extensible Authentication Protocol (EAP). Common EAP methods in 802.1x environments are EAP-MD5, EAP-FAST, EAP-TLS, and Protected EAP – Microsoft Challenge Handshake Authentication Protocol version 2 (PEAP-MSCHAPv2). EAP-TLS uses certificates for client/server authentication, whereas EAP-MD5, EAP-FAST, and PEAP-MSCHAPv2 use passwords for authentication. Cisco Catalyst switches fully support authentication of UC devices and PCs through the use of multidomain authentication (MDA) configuration parameters.

The authentication server, such as Cisco Identity Services Engine, provides authentication, authorization, and accounting (AAA) for devices trying to access the network by leveraging standards-based protocols, such as EAP over LAN (EAPoL) and Remote Authentication Dial-In User Service (RADIUS). The authentication server enables organizations to create flexible and granular security controls as they request access to the network and once they have authenticated to the network by incorporating authorization policy. In practical terms, this means that administrators can dynamically place

https://t.me/learningnets

@cisco_in_persian_channel

authenticated users and devices into separate logical network segments and apply certain security policies to those groups and devices. Within ISE, this is done with downloadable ACLs (dACLs).

An authentication database, such as Microsoft Active Directory or ISE, is a critical component of an 802.1x authentication implementation. The authentication database holds the credentials of the users to be authenticated in a centralized location. This type of solution provides the ability to align with organizational security policies. If users do not update their credentials, they are cut off (or restricted) from acquiring network access based on the authentication policy that is configured.

A client supplicant is software running on a device that attempts to gain access to the network. Operating systems such as Windows and OS X provide native supplicants to aid with 802.1x authentication. Software such as Cisco AnyConnect can run on top of an OS to also provide a supplicant. Last but not least, the firmware on Cisco IP phones also has a native supplicant for performing 802.1x authentication to a network. The next section discusses this topic in further detail.

The current release of the native supplicant inside a Cisco IP phone leverages either EAP-FAST or EAP-TLS for 802.1x authentication. EAP-TLS is the only option that supports X.509 certificates to simplify the 802.1x authentication process. The two different certificate types that are currently supported are the manufacturing installed certificate (MIC) and a locally significant certificate (LSC). A MIC is preinstalled at the factory during manufacturing of the IP phone and signed by one of the Cisco Manufacturing CA certificates:

- Cisco Manufacturing CA
- Cisco Manufacturing CA SHA2
- CAP-RTP-001
- CAP-RTP-002

These certificates are important because for ISE to successfully authenticate a phone onto the network via 802.1x, it needs to be imported into the ISE's trusted certificate store. From a certificate perspective, one thing that we must discuss is a certificate's chain of trust. This chain is important because it validates the authenticity of an X.509 server (or phone) certificate. Three components are needed to establish a chain of trust between certificates:

- **Root certificate:** An X.509 certificate that belongs to a certificate authority. It is used to issue other certificates.
- **Intermediate certificate:** An X.509 certificate that is subordinate to the root and issues server certificates.
- **Server certificate:** An X.509 certificate for a specific server or device.

A phone certificate and its certificate trust chain are shown in Figure 3-5.

https://t.me/learningnets

@cisco_in_persian_channel

Figure 3-5 *A MIC and Certificates in Its Certificate Trust Chain*

For ISE to authenticate a phone by its MIC, the manufacturing certificates need to be imported into ISE. You can find the MIC and then export it out of Cisco Unified CM by navigating to **Cisco Unified OS Administration > Security > Certificate Management.** The certificate should be exported in a .pem format. When importing into ISE, you should navigate to **Administration > System > Certificates > Trusted Certificates** and then choose **Import**. Figure 3-6 shows an example of ISE displaying the manufacturing certificates that have been imported.

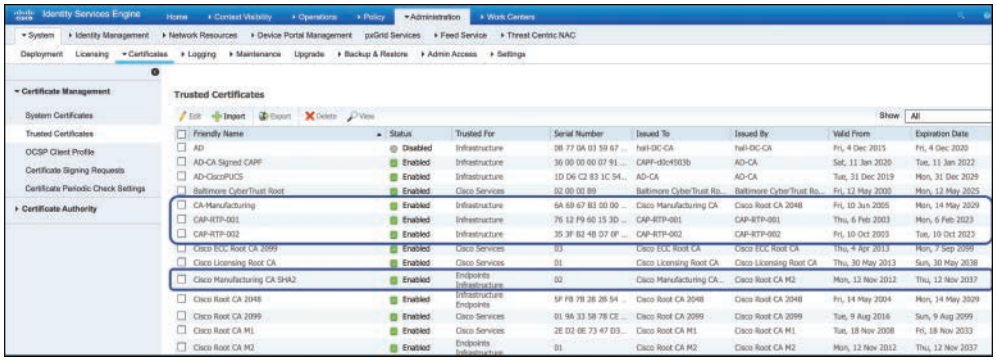


Figure 3-6 *Cisco CA Signed Certificates Inside the Trusted Certificate Store Within ISE*

https://t.me/learningnets

@cisco_in_persian_channel

Now that certificates have been imported into ISE, we can further explain the authentication process. Once ISE receives an authentication request from a client supplicant, it must examine the authentication policy and, ultimately, the sequence of identity stores that can be used in sequential order to authenticate a device. Within ISE, this is known as the identity source sequence. Because IP phones authenticate locally to ISE, a simple identity source sequence can be defined, such as to use only the internal ISE database. After this, you simply need to reference a certificate authentication profile that specifies what attribute to use inside the x.509 certificate to authenticate an IP phone. As shown in Figure 3-7, ISE uses the Subject – Common Name attribute. An example of the identity source sequence is shown in Figure 3-8.

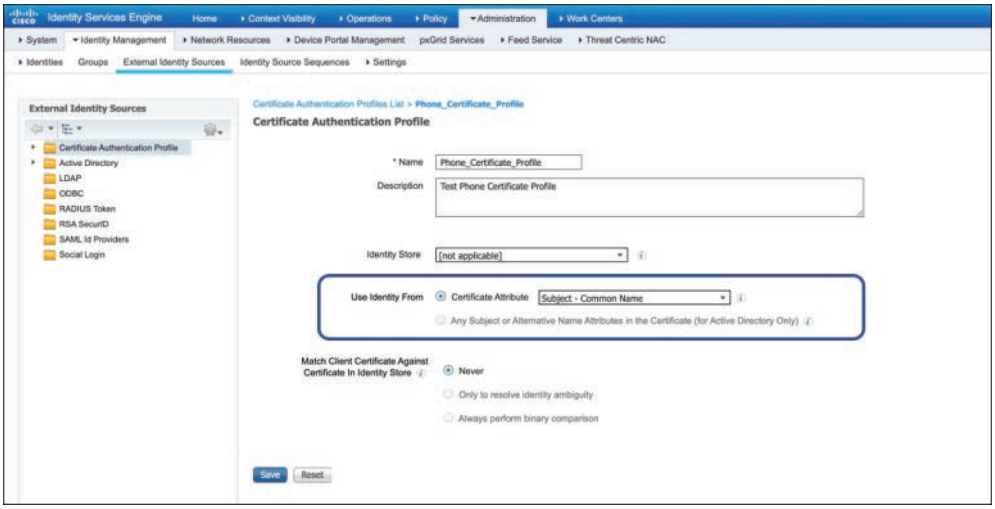


Figure 3-7 *The Certificate Authentication Profile Within ISE Using the Common Name Attribute*

Now that you understand how certificates are used and have a basic understanding of how Cisco ISE uses certificates to authenticate devices, we can discuss the differences in the certificates (MIC versus LSC) and why an organization may choose to use one or the other.

A downfall to using MICs is that it is difficult to prove that the phone belongs to a customer's Unified CM cluster(s) or that it even belongs on the network. This means that an attacker could place a rogue phone with a MIC on the network and potentially register it if auto-registration is enabled on Unified CM. Locally significant certificates (LSCs) provide an additional layer of security and verification that a phone belongs to the network because LSCs are signed by the Unified CM Publisher's Certificate Authority Proxy Function (CAPF) certificate based on RFC 5280, which allows Unified CM to take on the role of a root certificate authority. When CAPF facilitates the signing of LSCs, and an administrator intentionally deploys LSCs to IP phones used by UC administrators, an LSC provides a higher level of security and therefore is preferred (and prioritized) by the IP phone over the MIC.

https://t.me/learningnets
@cisco_in_persian_channel

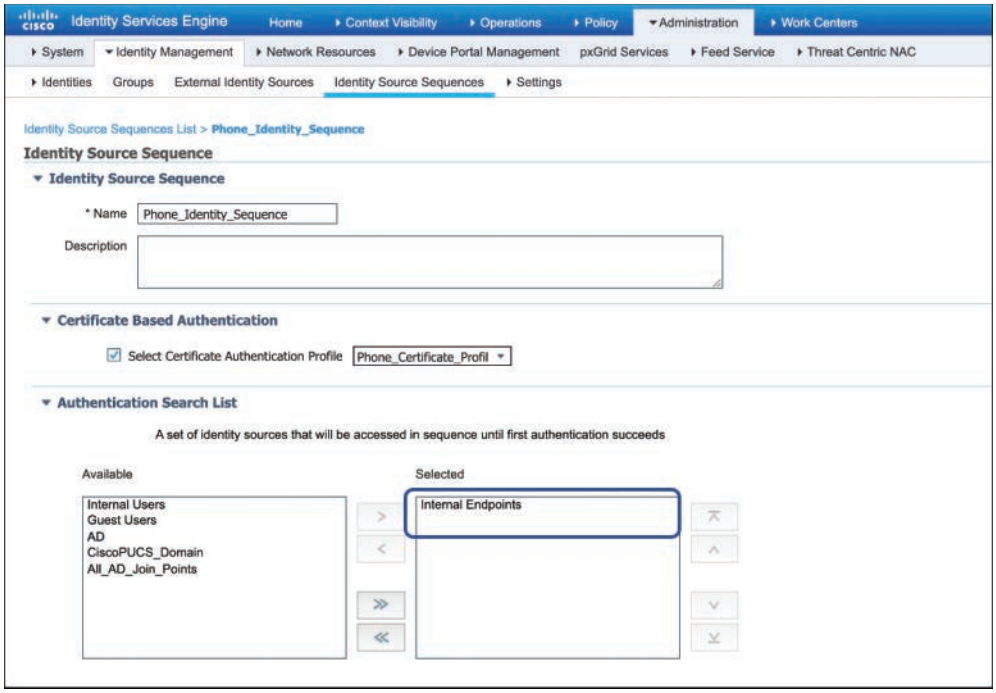


Figure 3-8 *The Identity Source Sequence Within ISE, Which Is Used to Reference the Location (and Ordered List) of Authentication Databases for Endpoints*

When you're deciding which certificate to use for implementing 802.1x security on the network, one approach is to begin by using the MIC because it is the quickest and simplest option. When phones are able to join the network and are able to register to Cisco Unified CM, LSCs can be deployed at any time with minimal changes to the network (e.g., modification of authentication policy). It is a good idea to eventually use LSCs for 802.1x authentication because for many organizations, they are useful for encrypting voice and video traffic. We discuss this topic in more detail in Chapter 7, "Encrypting Media and Signaling."

While 802.1x authentication for Cisco phones has been supported since Unified CM 7.1.2, it is possible that an organization is currently using an IP phone that does not support 802.1x—for example, Cisco IP phones such as 7935, 7936, 7940, and 7960. In some cases, older phone models may have previously supported 802.1x, but with legacy protocols, such EAP-MD5, which have been deprecated. In some cases, the MICs may be expired and therefore no longer valid, so 802.1x authentication with certificates is possible only through use of LSCs. Further, some of the newer TLS 1.2 algorithms used with LSCs may not be supported on legacy phones. Cisco currently recommends checking the release notes for your current version of Unified CM and IP phone firmware to determine support for 802.1x.

https://t.me/learningnets

@cisco_in_persian_channel

Within Cisco's authentication framework, different modes of deployment are available so that organizations can implement a phased approach for strengthening authentication onto the network. Currently, organizations can use three different deployment modes to ensure that PCs and UC endpoints are not prevented from joining the network:

- **Monitor mode:** Provides a nondisruptive environment to monitor the impact that 802.1x can have to the organization without preventing access to the network
- **Low-impact mode:** Uses a pre-authentication ACL (PACL) to allow a subset of traffic prior to authentication, such as DHCP requests
- **Closed mode:** Prevents access to the network prior to authentication/authorization

By using one of these modes, an organization can deploy security in phases. As an example, an organization can deploy 802.1x in monitor mode in conjunction with MAC Authentication Bypass so that it can monitor network authentication failures and adjust policies on an as-needed basis. This way, leadership can evaluate risks while ensuring operations are not impacted by the addition of more stringent security policy being applied to the network infrastructure, which includes Cisco Catalyst switches, wireless LAN controllers, wireless access points, DNA Center, and ISE. For additional detail about configuring the network infrastructure to support 802.1x and MAB, see the relevant security configuration guide for the version of IOS/IOS-XE software that your organization is using. The next section provides additional detail around benefits and use cases for MAB.

MAC Authentication Bypass (MAB) and Network Access Control (NAC)

MAC Authentication Bypass is helpful for scenarios in which you need to authenticate devices that do not have a client supplicant that will support 802.1x authentication. Devices that do not typically support a client supplicant include fax machines, printers, and IoT devices. When MAB is enabled, the Cisco switch uses an endpoint's MAC address as the client identity. For a device to join the network with MAB enabled, MAC addresses of endpoints such as IP phones must be whitelisted in a database that is present in Cisco ISE.

One benefit of Cisco's authentication framework is that it supports flexible authentication methods. As an example, MAB can be enabled as a backup authentication method to 802.1x authentication. An authentication policy that can be created to prioritize 802.1x authentication and to use MAB only as a backup method is shown in Figure 3-9.

When MAB is configured as a backup authentication method, and an administrator designates the voice VLAN as critical, if ISE does not respond to an authentication request, a switch port goes into critical authentication mode. When traffic coming from an endpoint is tagged with the voice VLAN, the endpoint (e.g., IP phone) is put into the voice VLAN that was previously configured on the switch port and allowed onto the network. As previously discussed, the voice VLAN can be learned dynamically through Cisco Discovery Protocol (CDP) or through LLDP. Critical voice VLAN support prevents a scenario in which IP phones become usable because they cannot access the network or the UC infrastructure.

https://t.me/learningnets

@cisco_in_persian_channel

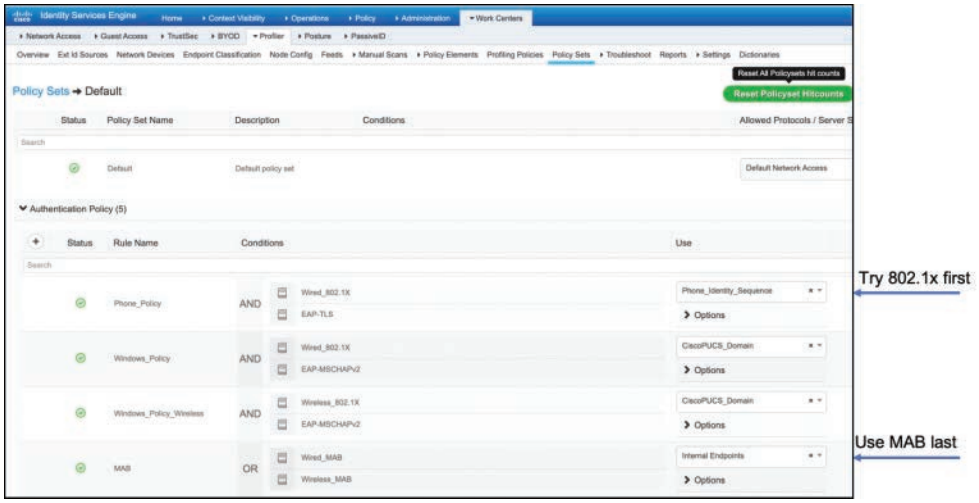


Figure 3-9 *An Authentication Policy Within ISE That Prioritizes 802.1x and Uses MAB as a Failover Mechanism*

It is worth mentioning that MAB is not truly an authentication method. Because MAC addresses can be easily spoofed, MAB is considered more of a backup authentication method for situations when an endpoint is unable to perform 802.1X authentication. In scenarios that 802.1x cannot be deployed, here are some questions that you should ask:

- What are the risks to unauthorized users gaining access to the network infrastructure through a spoofed MAC address?
- What is the likelihood of someone gaining access to the network environment?

As an additional security measure, to minimize MAC spoofing, Cisco ISE can be used to provide network access control (NAC) for the network. This is possible because the Cisco ISE profiler can be used to dynamically detect and classify the types of endpoints that are connected to the network. While still using a device's MAC addresses as the unique identifier, ISE is able to collect various attributes from each endpoint and then use a profiler policy to determine the Total Certainty Factor (TCF) of the credibility of the device. In other words, ISE is able to determine whether a device is really who it says it is. As shown in Figure 3-10, the process is cumulative based on how a device matches the collected attributes to prebuilt or user-defined conditions, which are then correlated with an extensive library of profiles. These profiles include but are not limited to a wide range of device types, such as IP phones, mobile devices, cameras, printers, gaming consoles, and IoT sensors. A TCF that has been assigned to an endpoint, such as a Cisco IP phone, is shown in Figure 3-11.

https://t.me/learningnets

@cisco_in_persian_channel

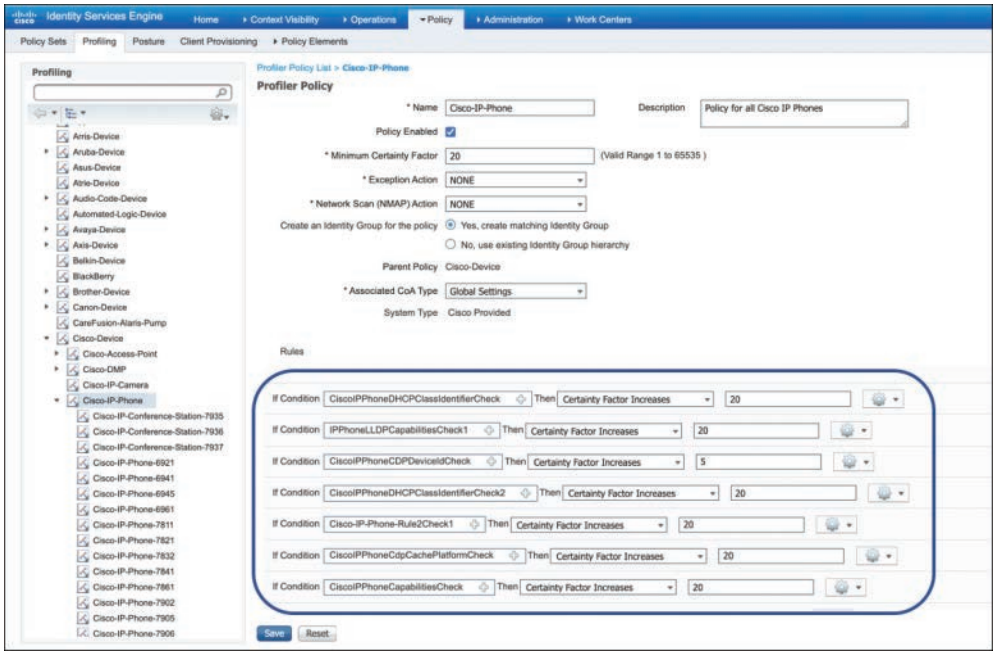


Figure 3-10 *The Way the Total Certainty Factor Is Dynamically Created Within ISE*

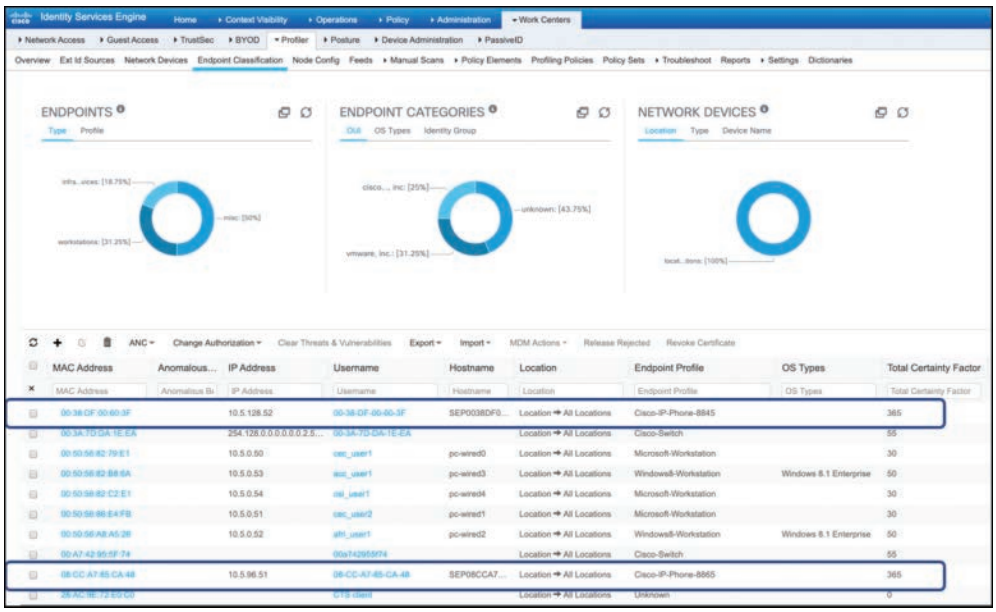


Figure 3-11 *The Total Certainty Factor That Is Assigned by ISE*

https://t.me/learningnets

@cisco_in_persian_channel

Once endpoints are classified and granted access to the network, an authorization profile can be created to specify the type of network access that should be granted to a device based on the profile. The theory behind this approach is that because certain devices are more trusted than others, the level of network privileges should reflect it. An example of the controls that administrators have is that different devices can be put into different VLANs and, if necessary, downloadable ACLs (dACLs) can be assigned to limit access to specific resources. This solution helps prevent or limit the exposure of access to a network if an attacker were to spoof MAC addresses to connect to the network. Practically, the authorization policy may have more trust for UC endpoints that are authenticated via 802.1x with an LSC than devices that use MAC Authentication Bypass as an authentication method. An authorization policy is shown in Figure 3-12.

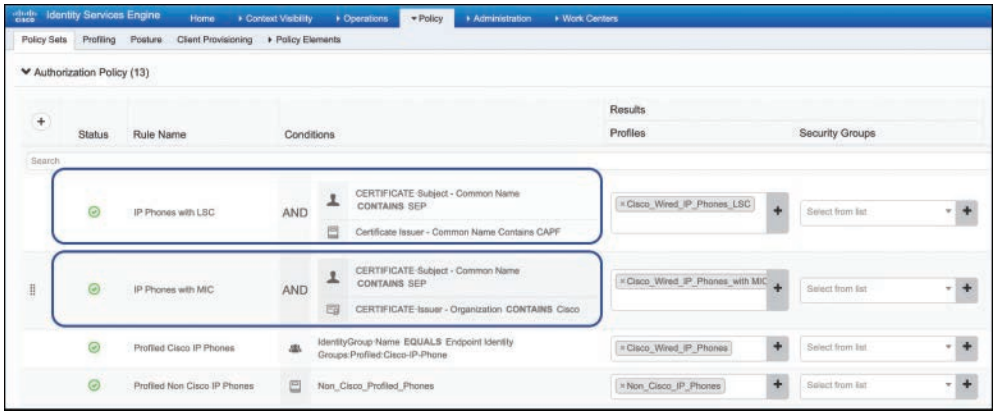


Figure 3-12 *An Authorization Created Within ISE to Apply Different Levels of Trust to Endpoints*

Security Features

A secure UC environment requires the coordinated design of network services such as Dynamic Host Configuration Protocol (DHCP), Address Resolution Protocol (ARP), Trivial File Transfer Protocol (TFTP), Network Time Protocol (NTP), and Domain Name System (DNS). To provide a resilient UC environment, security features should work in unison with these network services.

Figure 3-13 shows which network services are needed to support a UC environment across the network and for which purpose.

https://t.me/learningnets

@cisco_in_persian_channel

Protocol	Purpose
CDP/LLDP	To obtain VLAN/Voice VLAN, negotiate PoE
EAPOL	To authenticate endpoints to the access port
DHCP	To obtain IP address and initial configuration information
ARP	To discover the network's default gateway
TFTP/HTTPS	To obtain device configuration files, firmware, Certificate Trust List, etc.
NTP	Synchronization to the network clock, ensure accurate CDRs
DNS	To resolve hostnames to IP addresses

Figure 3-13 *Protocols and Their Purposes for a UC Environment*

To help stress the importance of leveraging security features in the network, let's discuss the ease of access to information that an attacker can get from the settings button on an IP phone. The network settings page lists many of the network elements and detailed information that is needed for the phone to operate, such as

- IP address of the router (default gateway of IP phone)
- IP address of the DNS server
- IP address of the TFTP server(s) within the Unified CM cluster

By obtaining these pieces of information, an attacker could initiate a network reconnaissance attack, which is the first step in learning more information about a network. The goal of a reconnaissance attack is typically how to gain access to or attack the network or attack the UC infrastructure. Common examples of network reconnaissance attacks include port scanning, ping sweeping, packet sniffing/captures, and more. A network setting screen from an IP phone is shown in Figure 3-14. Unified CM provides the ability to disable access to network settings. To do this, you should navigate to **Unified CM Administration > Device > Phone (Phone Configuration) > Product Specific Configuration Layout > Settings Access** and then choose the disabled parameter.

Figure 3-14 *Information That Can Be Gathered from the Settings Button on an IP Phone*

https://t.me/learningnets

@cisco_in_persian_channel

VLAN Hopping

Before the phone has its IP address, an endpoint discovers which voice VLAN it should be located in by means of the Cisco Discovery Protocol (CDP) or Link Layer Discovery Protocol (LLDP). The auto-assignment of a voice VLAN is useful for providing dynamic segmentation at layer 2 from other types of endpoints on the network. This segmentation typically allows administrators to prevent unwanted traffic across a network with a security control such as an access control list. An example to depict the typical traffic flow is shown in Figure 3-15.

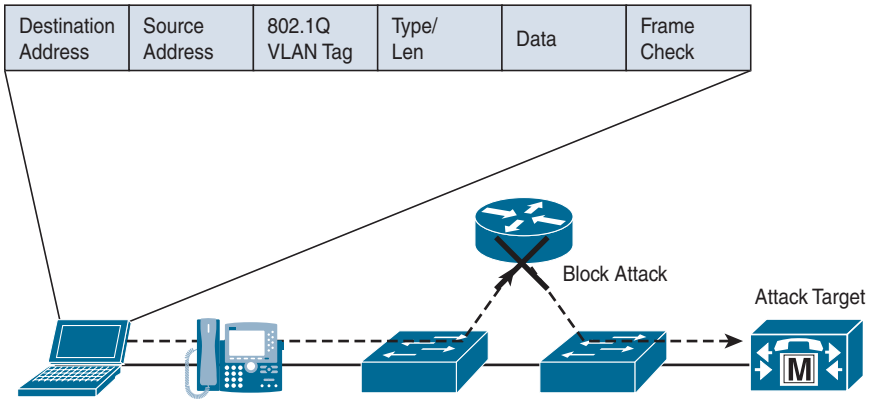


Figure 3-15 *The Way an ACL on the Network Can Be Used to Provide Security*

A threat, known as *VLAN hopping*, is able to bypass security controls in a layer 3 device such as a router or firewall using two different approaches:

- **Double tagging:** When a hacker crafts an IP packet with dual 802.1q tags to send IP traffic to a target device. An inner tag is the VLAN that an attacker wants to reach, and the outer tag is the native VLAN that a device is supposed to be on.
- **Switch spoofing:** When a hacker PC masquerades as a switch and negotiates a trunk connection using Dynamic Trunk Protocol (DTP). If this happens, an attacker can discover information about the native VLAN and possibly elect itself as the root switch for the network. This is possible when a port is configured for "dynamic auto" or "dynamic desirable."

When a VLAN hopping attack is executed properly, an attacker may can launch an attack against infrastructure without alerting security personnel. As shown in Figure 3-16, an attacker can bypass infrastructure that would ordinarily provide packet filtering.

https://t.me/learningnets
@cisco_in_persian_channel

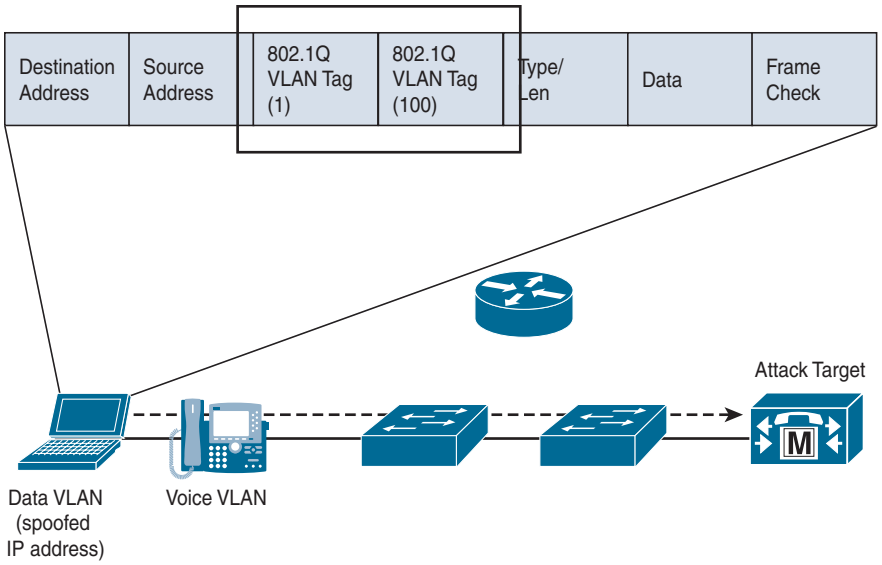


Figure 3-16 *An IP Packet That Has Been Crafted to Bypass Security Controls*

To mitigate a double tagging attack, you can disable PC Voice VLAN Access within Cisco Unified CM. When disabled, this feature does not allow the devices plugged into the PC port on the phone to "jump" VLANs and get onto the voice VLAN by sending 802.1q tagged information destined for the voice VLAN to the PC port on the back of the phone. For most customers, this setting should be disabled. An exception is when a PC is running monitoring and recording applications for training or quality control for someone such as a customer service agent. If this is the case, it may make sense to leave this setting enabled. To disable PC Voice VLAN access, you should navigate to **Unified CM Administration > Device > Phone (Phone Configuration) > Product Specific Configuration Layout > PC Voice VLAN Access** and choose the disabled parameter.

To prevent switch spoofing, dynamic switchport trunking should be disabled on the switch. By default, Cisco switches are enabled to negotiate trunks using the Dynamic Trunking Protocol. Within a switch running 16.9 IOS-XE code, the **switchport nonnegotiate** command can be used to stop a Cisco switch from trying to negotiate a trunk. Network administrators should also consider changing the native VLAN that is used on trunk ports to something different than VLAN 1, which is used by default, and assigning access ports to VLANs that are not be used by other devices. Lastly, administrators should consider mechanisms to prevent a rogue switch from claiming the spanning tree root role. This can be done by configuring **spanning-tree rootguard** and **spanning-tree bpdu guard** on Cisco switches that have been previously designated as the root switch.

DHCP Snooping

In a campus environment, it is common for both PCs and UC endpoints to leverage Dynamic Host Configuration Protocol (DHCP) to minimize the administrative burden

https://t.me/learningnets
@cisco_in_persian_channel

of manually configuring each device with an IP address and other configuration information. By leveraging DHCP, administrators do not have to worry when a user wants to move an endpoint between different locations (and between IP subnets). The configuration information is provided by a DHCP server located in the network, which responds to DHCP requests from DHCP-capable clients.

A well-known attack on the network's DHCP server is called a *DHCP starvation* attack. In practice, a hacker attempting to launch a DHCP starvation attack against an organization leverages tools to create bogus DHCP requests from one or more random source MAC addresses and/or with different DHCP payloads to consume all of the valid IP addresses in the existing DHCP scope(s) of the organization's DHCP server. When the valid DHCP scope becomes exhausted, an attacker can deploy one or more rogue DHCP servers and take control of how devices obtain their network settings, along with other settings that are relevant to UC endpoints, such as TFPT, which is typically included in a DHCP request using Option 150 to obtain configuration information from Cisco Unified CM.

A feature within Cisco IOS/IOS-XE called *DHCP snooping* prevents a nonapproved/rogue DHCP server from handing out IP addresses on a network by blocking all replies to a DHCP request unless that port is allowed to reply. Because most phone deployments use DHCP to provide IP addresses to the phones, you should use the DHCP snooping feature in the switches to secure DHCP messaging. Rogue DHCP servers can attempt to respond to the broadcast messages from a client to give out incorrect IP addresses, or they can attempt to confuse the client that is requesting an address.

When DHCP snooping is enabled, switches across the network provide security by acting like a firewall and filtering out DHCP messages between untrusted hosts and DHCP servers.

To do this, the switch must build and maintain a DHCP snooping binding database. Care should be taken to ensure there is a valid backup of the binding database; otherwise, valid users and endpoints may not have access to DHCP services. The database can be backed up locally on the flash file system or remotely with FTP, TFTP, HTTP, and RCP.

By default, access layer switch ports are considered untrusted for use of DHCP services. Therefore, DHCP snooping is configured only on network ports that connect to a DHCP server.

The following example shows how to enable DHCP snooping:

```
Switch(config)# ip dhcp snooping (enables DHCP snooping)
Switch(config)# ip dhcp snooping database flash:dhcp_snooping_db
(location of DHCP snooping database)
Switch(config)# ip dhcp snooping database write delay 15 (delay before
writing changes to the database)
Switch(config)# ip dhcp snooping vlan 1 100 (VLAN ranges for DHCP
snooping)
Switch(config)# interface gig1/0/20
Switch(config)# ip dhcp snooping trust (interface that DHCP server is
connected to)
Switch(config)# ip dhcp snooping limit rate 1000 (maximum DHCP packets
per second rate)
```

https://t.me/learningnets

@cisco_in_persian_channel

ARP Inspection

The Address Resolution Protocol (ARP) is used to map an IP address to a MAC address. Operationally, if an endpoint tries to communicate with another endpoint, it sends out an ARP request as an IP broadcast message. The endpoint that owns the IP address provides an ARP response (with its IP and MAC address) to the requesting endpoint. The response is stored in its ARP cache for a limited time. For Microsoft Windows, the default lifetime is 2 minutes; for Linux, it is 30 seconds; and for Cisco IP phones, the default lifetime is 40 minutes.

Because ARP allows for gratuitous replies, even if an ARP request was not received, an ARP spoofing attack and/or the poisoning of ARP caches can easily occur. In this type of attack, all traffic from the device under attack can be intercepted by an attacker, as a man-in-the-middle attack, before it is forwarded to a local host, a switch, or an upstream router.

Dynamic ARP Inspection (DAI) is a feature that is configured along with DHCP snooping. Operationally, DAI helps inspect ARP requests and replies whether they are gratuitous or nongratuitous and whether they come from untrusted ports to ensure that the request matches a valid IP-to-MAC address binding in the DHCP snooping database. If DAI is enabled without DHCP snooping, the configuration results in a self-imposed denial of service to any device in that VLAN because none of the devices are to use ARP.

Dynamic ARP inspection is also enabled on a per-VLAN basis by using this global configuration command:

```
Switch(config)# ip arp inspection vlan 1-100 (range of VLANs to inspect)
```

NTP

Network Time Protocol (NTP) allows network devices to synchronize their clocks to a network time server or network-capable clock over UDP port 123. Synchronizing time is critical for troubleshooting network devices or the timestamps that are placed on logs, traces, call detail records (CDRs), and system reports. In fact, many UC applications such as Cisco Unified CM cannot be installed until synchronizing with an NTP server first.

The requirement for NTP also extends to additional servers in the organization, such as a domain controller that contains information about users on the network (such as ACME.com) and ISE servers, which are used for 802.1x authentication. If time is not synchronized on all your devices, certificates cannot be properly validated. In most cases the opposite outcome actually occurs, and certificates are considered untrusted. For this reason, an administrator should make sure that the UC infrastructure, security infrastructure (e.g., ISE), and server infrastructure (e.g., Active Directory) use a common NTP source and are synchronized.

As a point of reference, using Windows Time Services as an NTP source is not recommended or supported for UC infrastructure. The reason is that Windows Time Services often uses Simple Network Time Protocol (SNTP), and Cisco Unified CM cannot

https://t.me/learningnets

@cisco_in_persian_channel

successfully synchronize with SNTP. To avoid potential compatibility, accuracy, and network jitter problems, an NTP server supporting NTPv4 is recommended. An IOS-XE router or Linux server may be utilized to support NTPv4. The Cisco router uses the following version of NTP:

```
cube# show ntp information
Ntp Software Name              :  Cisco-ntpv4
Ntp Software Version           :  Cisco-ntpv4-1.0
Ntp Software Vendor            :  CISCO
Ntp System Type                :  Cisco IOS
cube#
```

It is generally recommended to configure all network infrastructure to connect to NTP time sources that are connected to an accurate and authoritative time source, such as GPS, a radio, or an atomic clock. The internal clock of an IOS/IOS-XE device is not very accurate, so Cisco doesn't recommend its use. A *stratum* is used to describe how many NTP hops away the device is from an authoritative time source. When a network device has access to one or more NTP sources, it uses an algorithm to detect which time source it should synchronize with. In most cases, the algorithm chooses the NTP source with the lowest stratum time, but it is also able to detect when a clock is inaccurate and synchronize with the most accurate time source. Cisco currently recommends having more than one NTP server for high availability/accuracy.

If an organization has an authoritative time source that it would like to use, you can issue the following global configuration commands on an IOS/IOS-XE router:

```
Router(config)# ntp master 2
Router(config)# ntp source [source interface]
```

If an organization doesn't have an authoritative time source to use, network devices can connect to many publicly available NTP sources. An example of a public time source is NIST, which is accessible by directing network infrastructure to *time.nist.gov*, which load-balances NTP requests across its NTP servers. Once a device, such as a router at the edge of a network, is synchronized with a time source such as NIST, it is able to provide NTP to NTP clients across the network. To sync up with one or more authoritative NTP servers, from a network device running IOS/IOS-XE software, you can use the following global configuration commands:

```
Switch(config)# ntp server [ip address of authoritative NTP source]
Switch(config)# ntp source [source interface]
```

Previously, we gave NIST as an example of a public resource for NTP. To ensure authenticity of a public time source, NIST also operates NTP servers that support authentication for registered users. NTP servers that provide authentication sessions are beneficial to an organization because they minimize the risk of the organization encountering an NTP poisoning attack, which happens when a time source is advertised on a public network by an attacker with malicious intent to attack the organization's network

https://t.me/learningnets

@cisco_in_persian_channel

infrastructure. NTP authentication can be configured globally on Cisco IOS/IOS-XE devices. To do this, you can use the following configuration commands.

On an IOS-XE device acting as an NTP server:

```
Router(config)# ntp authenticate
Router(config)# ntp authentication-key 5 md5 [authentication key]
Router(config)# ntp trusted-key 5
Router(config)# ntp server [ip address of authoritative NTP source]key 5
```

On an IOS/IOS-XE acting as an NTP client:

```
Switch(config)# ntp authenticate
Switch(config)# ntp authentication-key 5 md5 [authentication key]
Switch(config)# ntp trusted-key 5
```

As of Cisco Unified CM 11.5(1)SU3, NTP authentication is supported. This feature is based on symmetric key-based authentication with SHA1-based encryption. Unified CM authentication leverages NTP version 4.2.6 and higher. While, in theory, NTP version 4 is backward compatible with version 3, many issues were observed with attempts to use different NTP versions. These issues are documented as of Unified CM version 9.x and later, which specify requirements for NTPv4 servers to be used for NTP. Cisco also currently recommends that UC administrators connect UC applications, such as the Unified CM Publisher, to NTP servers that are not higher than stratum 4 (e.g., stratum-1, stratum-2, or stratum-3). This way, they ensure that the UC cluster time is synchronized with an accurate external time source.

As shown in Figure 3-17, you, as the UC administrator, can check the status of an NTP on a UC cluster by issuing the **utils ntp status** command from the command-line interface. To add one or more NTP servers, you can issue the **utils ntp server add [ntp server]** command.

```
admin:utils ntp status
ntpd (pid 5936) is running...

     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
*10.130.2.1      132.163.97.1     2 u  283 1024  377    0.818  -21.140   9.480

synchronised to NTP server (10.130.2.1) at stratum 3
   time correct to within 77 ms
   polling server every 1024 s

Current time in UTC is : Thu Oct 15 21:33:59 UTC 2020
Current time in America/New_York is : Thu Oct 15 17:33:59 EDT 2020
admin:
```

Figure 3-17 *Checking NTP Status Within Cisco Unified CM*

To check the NTP status from a network device running IOS/IOS-XE software, you can issue the **show ntp associations** command.

https://t.me/learningnets

@cisco_in_persian_channel

DNS

The Domain Name System (DNS) enables the mapping of host names and network services to IP addresses within a network or networks. Based on the configuration of a UC environment, use of DNS is not always required to obtain services from UC applications. However, Cisco highly recommends use of DNS to support full UC functionality. As an example, DNS is currently required to support features and use cases such as

- Use of X.509 certificates with fully qualified domain names (FQDN)
- Discovery of UC services for Jabber clients (internal and external)
- Single sign-on for Jabber clients
- Resolution of FQDN for SIP trunk destinations and patterns
- Simplified system management: using host names instead of IP addresses

With X.509 certificates, the use of fully qualified domain names with certificates is mandatory. As you find out in Chapter 7, X.509 certificates are required to support encrypted signaling and media. We also discuss how Jabber clients use DNS SRV records to find UC services in Chapter 11. In short, a secure and highly functional collaboration solution heavily relies on DNS to function correctly for a number of services. For this reason, DNS servers should be deployed in a redundant fashion and be able to resolve host names inside the organization and also external to the organization.

While many organizations leverage DNS internally, many organizations leave it to their service provider to provide external DNS requests. As more organizations leverage and provide Collaboration services at the edge of the network and use Internet connectivity as a transport, the need for visibility of external DNS requests has increased. The reason is that DNS requests precede the IP connection, which enables DNS resolvers to log requested domains regardless of the connection's protocol or port. Monitoring DNS requests, as well as subsequent IP connections, is an easy way to provide better accuracy and detection of compromised systems, improving security visibility and network protection.

When DNS servers are used to resolve host names externally, cloud-based platforms such as OpenDNS (operated by Cisco) and Cisco Umbrella can be used to provide name resolution along with increased visibility of the DNS requests of various users and devices. This increased visibility allows organizations to identify patterns of users and devices, to investigate anomalous activity, and to prevent DNS-based attacks. To obtain information about current threats, Talos (Talos Security Intelligence and Research Group) integrates with the security community and analyzes millions of malware samples per day. Talos directly integrates into DNS platforms such as OpenDNS and Umbrella to dynamically block traffic that is destined to a wide variety of malicious domains, IP addresses, and URLs. Talos is also able to provide security

https://t.me/learningnets

@cisco_in_persian_channel

intelligence feeds into other network infrastructure such as firewalls, IPSs, email security appliances, and web security platforms. To find out more information about why security is needed for external DNS requests, you can visit OpenDNS by navigating to www.opendns.com/ or https://umbrella.cisco.com/. Organizations that do not currently have DNS security in place for external DNS requests should consider a trial version of OpenDNS. Currently, OpenDNS allows customers to register for free DNS accounts. To find out more information about Talos, visit https://talosintelligence.com/.

Firewalls and Access Controls

Traditional data firewalls can be used in conjunction with access control lists to protect Unified Communications infrastructure along with voice gateways from entities that should not be communicating with UC endpoints. In some cases, firewalls may introduce complexities into a design for Unified Communications solutions that include real-time services such as VoIP and video. As an example, if a VoIP call is encrypted, a firewall does not have much ability to inspect the VoIP traffic, and the firewall serves little purpose because it cannot dynamically inspect SIP-TLS traffic. This is why organizations are deploying Session Border Controllers, which act as a VoIP firewall to provide security controls for VoIP and video traffic. We cover Session Border Controllers in further detail in Chapter 11, "Securing the Edge."

Additionally, some limitations need to be considered with security infrastructure, such as IPv6 addressing. As an example, many organizations are starting to leverage IPv6 addressing for their IP phones because they have exhausted their IPv4 addresses. The ASA firewall currently supports IPv6 for collaboration traffic, but not all other Cisco security devices support IPv6. Until all of Cisco's security products support IPv6 for collaboration traffic, Cisco recommends keeping all IPv6 voice traffic contained within an enterprise network or to use a Session Border Controller, such as CUBE.

It is worth mentioning that UC environments have unique data flows that are both client to server and client to client. Using firewalls and/or ACLs to protect real-time traffic flows will likely frustrate firewall administrators based on the additional complexity of managing all of the required ACLs on a firewall to support the various scenarios for UC. As shown in Figure 3-18, a firewall policy that is based on denying all traffic and allowing only what is explicitly permitted by an organization can become quite extensive, even for a small environment, because an administrator would have to account for all of the client-to-client flows. In the following example, to permit a dynamic range of UDP ports, a firewall administrator must open a range of ports from 16384 to 32767 across six different subnets in each direction. It is at this point that the firewall administrator may be concerned about the security risks that are associated with punching so many holes in the firewall for UC traffic to flow correctly. To further compound the challenges with ACLs, several different ACLs would need to be entered to permit ports required for client/server traffic.

https://t.me/learningnets

@cisco_in_persian_channel

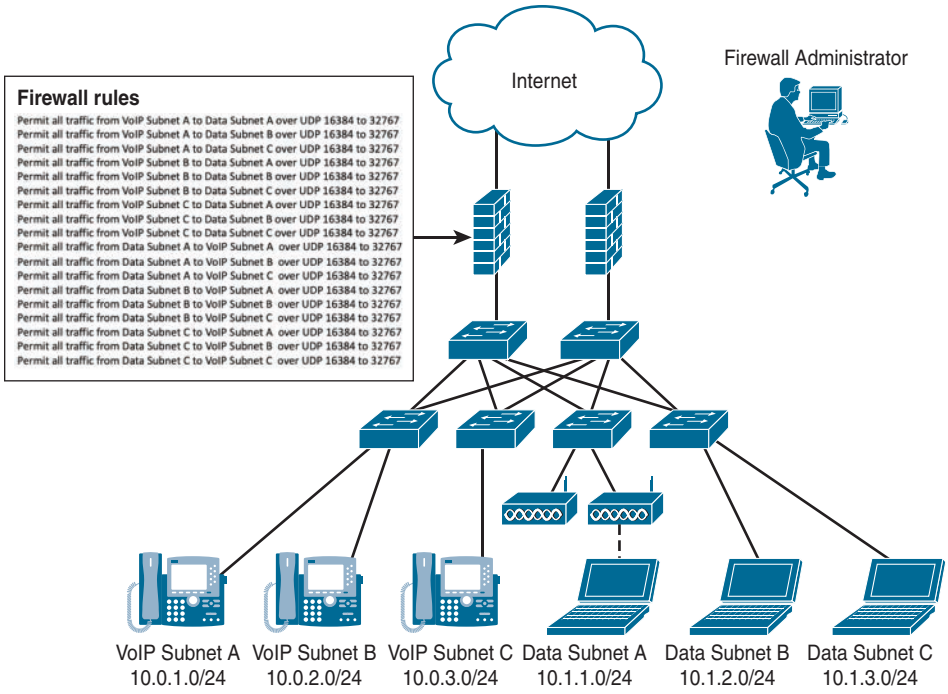


Figure 3-18 *Management Complexity That Is Added on a Firewall for Client-to-Client Traffic*

As when using VRFs, care should be taken when implementing ACLs; otherwise, UC traffic may be less than optimal or lack functionality. For this reason, you must take care in understanding the traffic flows for UC and to position firewalls in a manner so that the quality of the real-time traffic is not impacted by additional latency, delay, or jitter imposed by firewalls. Large amounts of real-time traffic can also cause an undue amount of stress on a firewall. For example, if real-time traffic is encrypted, the firewall cannot perform inspection on the traffic, so the firewall is providing limited functionality.

If organizations are required to place firewalls between UC signaling or real-time traffic for security purposes, the general rule is to monitor the CPU usage of the firewalls and to make sure that it does not exceed more than 60 percent for normal usage. If the CPU consistently runs over 60 percent, it increases the risk of impacting IP phone registration, call setup, and quality of a voice conversation. If firewalls are required to protect VoIP gateways, they can be placed either in front of the gateway or behind the gateway. If you are able to place the firewall in front of the VoIP gateway, the firewall provides filtering of unwanted connections and streams and protects the gateway from denial-of-service attacks. In Chapter 11, we discuss voice-specific firewalls, also known as Session Border Controllers, and the additional protection that they can provide at the edge of the network.

https://t.me/learningnets
@cisco_in_persian_channel

Continuous Monitoring

Continuous monitoring of the network and its security controls for effectiveness is key to the overall health and security of the network. NIST has released publication 800-137 on this topic of continuous monitoring and establishing the practice of monitoring. MITRE provides Common Vulnerabilities and Exposures (CVEs), which are the industry standard for identifying common vulnerability and exposure identifiers. Lastly, there is a Common Vulnerability Scoring System (CVSS) provided by the Forum of Incident Response and Security Teams (FIRST). CVSS is a published standard that is used by organizations worldwide. In principle, the CVSS captures the severity of a vulnerability by associating a numerical score to it.

For new vulnerabilities, the Cisco Product Security Incident Response Team (PSIRT) creates and maintains publications, commonly referred to as PSIRT Advisories, for security-related issues in Cisco products. The method used for communication of less severe issues is the Cisco Security Response.

To get access to Cisco PSIRT information, you have these different options:

- Visit the Cisco PSIRT website.
- Subscribe to RSS feeds.
- Integrate with the Cisco PSIRT's openVuln API, which can be used for programmability and automation of security functionality.

To learn more about accessing and using the openVuln API, visit the Cisco PSIRT page on the Cisco DevNet website: https://developer.cisco.com/site/PSIRT.

Summary

The purpose of the network is to help connect things. These things include IP phones, video teleconferencing devices, laptops, mobile devices, and many other things. A sophisticated attacker understands how an organization's critical services are built (e.g., UC services) and also understands how to leverage weaknesses in the architecture to launch attacks. Therefore, the network can and should be used to secure an organization's UC environment against unexpected attacks and behavior. An approach that organizations can take is based on defense-in-depth principles. Techniques such as segmentation and secure network access (e.g., 802.1x authentication) can reduce the attack surface. Lastly, security features can be enabled to protect against protocol-level attacks.

To gain the most functionality out of the UC environment and implement these various layers of security, an organization needs cross-functional alignment that includes the UC, network, and security teams. When these teams work together for the common good of an organization, this cross-functional alignment positions the organization for the most success.

https://t.me/learningnets

@cisco_in_persian_channel

Last but not least, it is not possible to protect or fix what you cannot detect. This statement has never been truer when considering the use of a network to provide security. When monitoring the network environment and also sharing details about possible issues in the most efficient manner, organizations can stop or minimize the damage of an attack before it can escalate out of control.

Additional Resources

www.cisco.com/go/sda

www.cisco.com/go/ise

www.nist.gov/

http://cve.mitre.org/

www.first.org/cvss/

www.cisco.com/go/psirt

https://t.me/learningnets

@cisco_in_persian_channel

This page intentionally left blank

https://t.me/learningnets

@cisco_in_persian_channel

Chapter 4

Maintaining Security Across UC Deployment Types

This chapter covers a variety of topics on how to maintain security across the different Unified Communications deployment models. To start, we cover the different UC deployment models with some associated security considerations surrounding availability and providing redundancy and resiliency for the UC environment. Then we follow up with common methods for securing the communication within the core UC applications. We focus on securing Network Time Protocol (NTP) to ensure the authenticity of the NTP servers the UC applications are synchronizing with. Next, we follow up with securing intracluster signaling between clustered UC application nodes such as Unified CM using IPsec. IPsec is also used to secure protocols such as H323 and Media Gateway Control Protocol (MGCP) to ensure that Secure Real-time Transport Protocol (SRTP) keys are not sent in the clear. The last section covers how to secure the CTI signaling between Cisco Unified CM and Cisco Emergency Responder (CER).

We left out some key protocols and integrations or only covered them lightly because they are covered in greater depth in subsequent chapters. For example, public key infrastructure (PKI) is covered in depth in Chapter 7, "Encrypting Media and Signaling." Chapter 6, "Core Cisco UC Application Lockdown," covers securing LDAP integrations and enabling single sign-on (SSO). Chapter 11, "Securing the Edge," covers securing SIP trunks meant for the Edge. The goal of breaking up these specific topics in separate chapters is to allow for a focused discussion with specific use cases. Consequently, some aspects of these topics are covered at a higher level than others because they are discussed in more depth in the following chapters.

Practical Security Examples

Building on what Anthony Starke has learned about increasing security through the fundamentals of networking, he realized that he needs to look at how the ACME UC environment has been deployed. He is curious to know whether the company has deployed its environment in a secure and resilient manner. Based on what he has learned about

https://t.me/learningnets

@cisco_in_persian_channel

network fundamentals, he is interested to know what information is being shared between the different servers in the environment. Because the UC deployment follows the campus deployment model, how will that model change as the company begins to bring in remote offices and business partners over the Internet?

As Anthony begins to understand the various deployment models for Cisco UC, as a UC engineer, he needs to be able to speak to the security currently implemented within the environment. As ACME's business grows and the UC infrastructure grows with it, it is imperative that he can proactively plan for required updates to its security stance.

Anthony has reviewed the existing UC environment at ACME and consulted with his manager, Jonah, to ensure his understanding of the existing network matches that of the leadership team. He has verified his understanding of the expected growth the company was foreseeing so that he can adequately plan for growth in the UC infrastructure. After verifying the process, he has begun to explain his next steps to address some security concerns he found in the base deployment.

Questions that you should ask:

1. What network security is currently in place to protect the UC environment based on the deployment model used?
2. What protocols are used to communicate between the core UC application nodes in an environment, and can they communicate securely?
3. How can you secure the intracluster communication channels and protocols within a UC environment?

Common Enterprise Collaboration Deployment Models and Security Considerations

It is important to understand the different deployment models within a UC environment because the security requirements can change based on the deployment in use. Within a Cisco UC environment, you may encounter three main Enterprise Collaboration deployment types; they are discussed at a high level in this chapter. For an in-depth explanation of the different UC deployment models available, review the latest Cisco Collaboration Solution Reference Design Guide located at www.cisco.com/go/srnd.

- **Campus deployment model:** The UC and other Collaboration services, endpoints, and gateways are co-located on a single high-speed LAN or MAN.
- **Centralized deployment model:** The UC and other Collaboration services are located in a central campus or data centers. The remote endpoints, gateways, media resources, and other components are deployed in remote sites interconnected by a QoS-enabled WAN.

https://t.me/learningnets

@cisco_in_persian_channel

- **Distributed deployment model:** This model is similar to the centralized deployment model except that multiple campus and/or centralized deployments are interconnected by SIP trunks or a dial plan aggregation platform, such as Cisco Unified CM Session Manager Edition (SME), over a QoS-enabled WAN.

Another deployment model is cloud-based and is discussed in Chapter 12, "Securing Cloud and Hybrid Cloud Services." Chapter 12 discusses the specific security requirements when working with Cisco cloud-based platforms like Webex and Webex Teams. With the different deployment models available, is there a hard-and-fast rule that says you must use only a specific deployment model? No, there isn't. In fact, most deployments tend to have a mix of different deployment models in use. Currently, the most common one is a hybridization of the on-premises Enterprise Collaboration deployments that integrate various cloud services like Webex Calling or Mobile and Remote Access (MRA), both of which are discussed in Chapters 11 and 12.

What types of redundancy are built into each of these models, and what types of communication do each of these deployment types exchange between nodes in the various clusters and other components?

Using the ACME UC environment as an example (Figure 4-1), you can first see that it uses the campus deployment model, and the following sections discuss the redundancy available that is built into the campus deployment model.

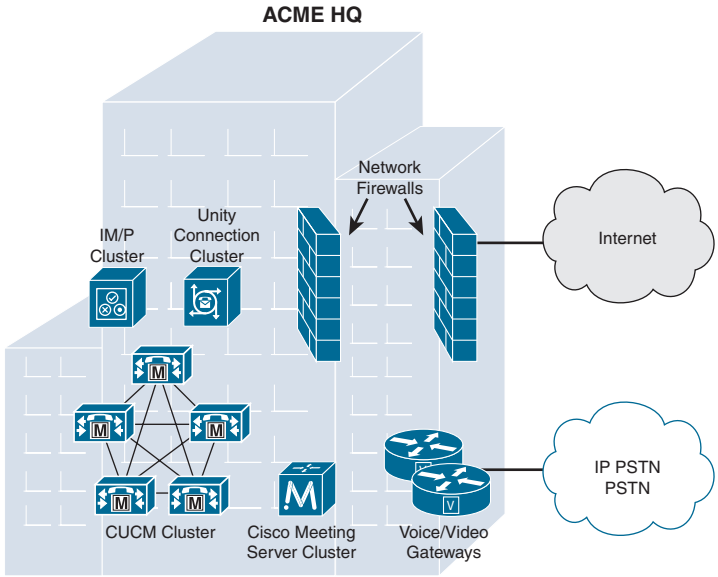


Figure 4-1 *ACME UC Environment (Campus Deployment)*

Because the UC environment is located within a single campus, it is considered all on-premises. All the traffic traverses a high-speed LAN or MAN, and none of the traffic traverses a WAN. In the event of a failure, you expect the UC environment located on the

https://t.me/learningnets
@cisco_in_persian_channel

local LAN to be resilient enough to survive a hardware failure. This means that there are redundant links between devices with multiple paths available. Additionally, redundant power supplies that are homed to different power sources, ideally using an uninterruptible power supply (UPS) of some sort, are deployed to increase resilience in the event of a power failure.

Virtual machine (VM) placement also plays a part in the context of survivability. The adage of not putting all your eggs in one basket is an important consideration. For most deployments, the virtual machines running the Cisco UC applications should be spread across multiple hypervisors. Within a UC environment, the Publisher or Primary nodes contain the master database within a UC cluster. Typically, the terms Publisher and Primary nodes can be used interchangeably when speaking about the application node with the primary database. These nodes should likewise be spread across the hypervisors where possible. Because these nodes typically contain the primary database, the impact on restoration is lessened if you lose a hypervisor that contains only one Publisher or Primary node rather than all of them for the environment. Building on that, the placement of the various nodes within a UC environment should allow for the seamless failover between hypervisors. For the applications that support failover between nodes, like Cisco Unified Communications Manager (Unified CM), the subscribers contained within a single Unified CM Group should not all reside on the same hypervisor. If they are in the same group and there is an outage where the hypervisor fails, all the devices using that CallManager group fail as well.

With the existing UC environment, public switched telephone network (PSTN) connections are made using T1 PRIs to the local telephony provider. There are plans to move to centralized SIP trunks using Cisco Unified Border Elements (CUBE) as part of a planned move to utilize cloud services; however, they are still in the planning stages. The current resiliency and redundancy to the PSTN are based on both hardware and software. For example, at the software level, redundancy is provided based on how the site route lists and route groups are configured to route calls out the local T1 connections. For voice gateways, it makes sense to have redundant voice gateways and then load-balance the circuits across them. Additionally, if separate T1s handle different types of traffic (local, long-distance, CAMA) and if there is more than one of each type, those circuits should be spread across the different voice gateways.

If you look at the UC environment, once it begins to add remote sites, their environment evolves from strictly using the Campus deployment model to overlaying a Centralized deployment model on top of the existing environment. This is possible because the HQ environment still uses the local resources while the remote sites leverage the resources over the WAN. With this model, you must account for the accessibility and redundancy at not only the HQ location but also at the remote sites.

What happens at the remote site when there is a WAN outage? What happens at the remote site when the WAN bandwidth is exceeded for voice?

With the Centralized deployment model, shown in Figure 4-2, the local UC environment at the HQ location doesn't change. The Campus model provides for local survivability within the local LAN/MAN. However, with the addition of remote sites, you now have to account for their availability of services in the event of a failure.

https://t.me/learningnets

@cisco_in_persian_channel

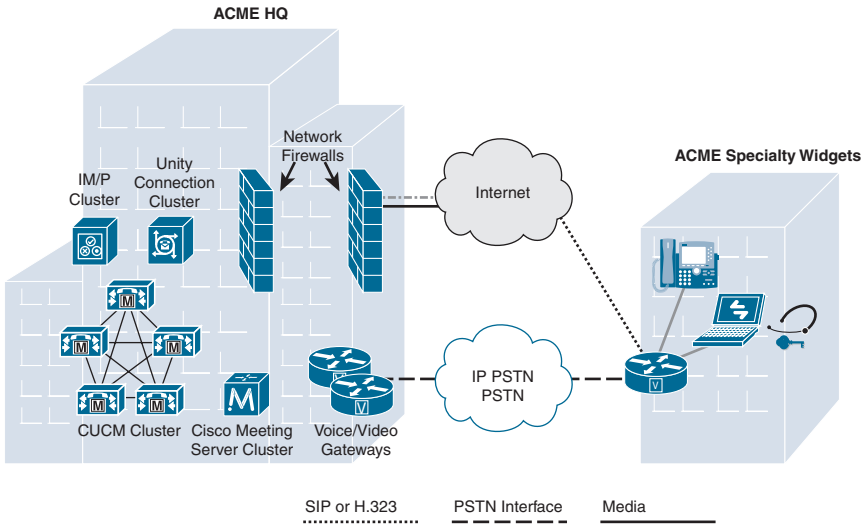


Figure 4-2 *ACME UC Environment (Centralized Deployment)*

Looking back at the definition for the Centralized deployment, you can see that the UC and other Collaboration services are located in a central campus or data centers. The remote endpoints, gateways, media resources, and other components are deployed in remote sites interconnected by a QoS-enabled WAN. For survivability at the remote sites, the main considerations to account for outside the standard network resiliency planning are how to provide services in the event of a WAN outage, centralized services failure, or even WAN congestion. Luckily, you can account for these issues through the software configuration either in Cisco Unified CM or the local voice gateways at the site.

In the event of a WAN or centralized services outage, some of the services impacted would be device registrations, inbound/outbound calling, and conferencing. You can facilitate survivability of these features by enabling and configuring either Survivable Remote Site Telephony (SRST) or Enhanced Survivable Remote Site Telephony (E-SRST). When standard SRST is enabled, basic phone functionality is preserved when the remote site is unable to connect to the centralized services. SRST should not be considered anything other than redundant because it does not provide a full feature set. The following features are available while SRST is active:

- Intra-site dialing between IP phones
- SIP/SCCP phone to PSTN/router voice-port
- SCCP phone to WAN VoIP using SIP or H.323
- SIP phone to WAN VoIP using SIP

https://t.me/learningnets

@cisco_in_persian_channel

With this basic feature set available, most users within a remote site should be able to perform their normal activities. However, with E-SRST enabled, additional features are made available, such as

- Voice Hunt Group
  - Shared lines
  - Mixed shared lines (SIP and SCCP phones)
  - Hunt statistics collection
  - Mixed deployment (SIP and SCCP phones)
- Shared lines
- BLF
- Video
- B-ACD

There are additional caveats to keep in mind about the availability of services. For example, does the site need voicemail access while in SRST or E-SRST? If voicemail is centralized as well, you need to consider additional configuration tasks to implement a solution like Survivable Remote Site Voicemail (SRSV). The use of SRSV allows a site to access its voicemails and message waiting indicators (MWIs) while in survivability mode. Depending on the outage type, other Unified Communications features like instant messaging or centralized conferencing likely are not available.

What happens at the site when the WAN bandwidth is exceeded? In this scenario, the WAN is up, and the centralized services are available to the remote site. When encountering this type of situation, you should take a couple of factors into account. The first is that the quality of service (QoS) for each remote site should have its traffic classified and marked appropriately to allow priority for voice and video traffic (signaling and media). Although defining and implementing QoS within an enterprise network is outside the scope of this book, you must account for the amount of bandwidth available to a site and the number of calls (and codec used) that should be allowed. The second is that this information should be configured within Cisco Unified CM to limit the number of on-net calls routed to the site and specify the codecs used for intra-site and inter-site calls. This way, you avoid poor speech and video quality. The third is that Cisco Unified CM uses the concept of automated alternate routing (AAR) to route calls out the PSTN and on to the remote site when the call comes from another site within the cluster when on-net calling is enabled.

As an organization's business needs expand into other regions or it acquires other businesses to enhance its portfolio, it will reach a point where it will have multiple deployments that it will need to join in a secure manner. From a design perspective, there is not much difference from the centralized deployment model with remote sites and a distributed deployment model when you look at availability. A distributed model can have

https://t.me/learningnets

@cisco_in_persian_channel

multiple independent sites, each with its own call processing agent cluster connected to an IP WAN that carries voice traffic between the distributed sites.

In Figure 4-3 you can see that redundancy and resiliency are accomplished using the same methods as the centralized deployment model where remote sites are deployed. Failover between endpoints within Cisco Unified CM is accomplished with Unified CM groups. Voicemail can be local to a campus, site, or even centralized within a region. In the event of a failure, depending on where the failure is, SRSV can be implemented at a local level to ensure availability and access to voicemail. One caveat with centralized PSTN access is that depending on how the circuits are configured, a location that goes into survivability mode may not be able to place or receive calls or may place only outbound calls. You should consult with your lines of business when planning to deploy centralized PSTN access so that you can plan how to handle situations where sites are in survivability mode.

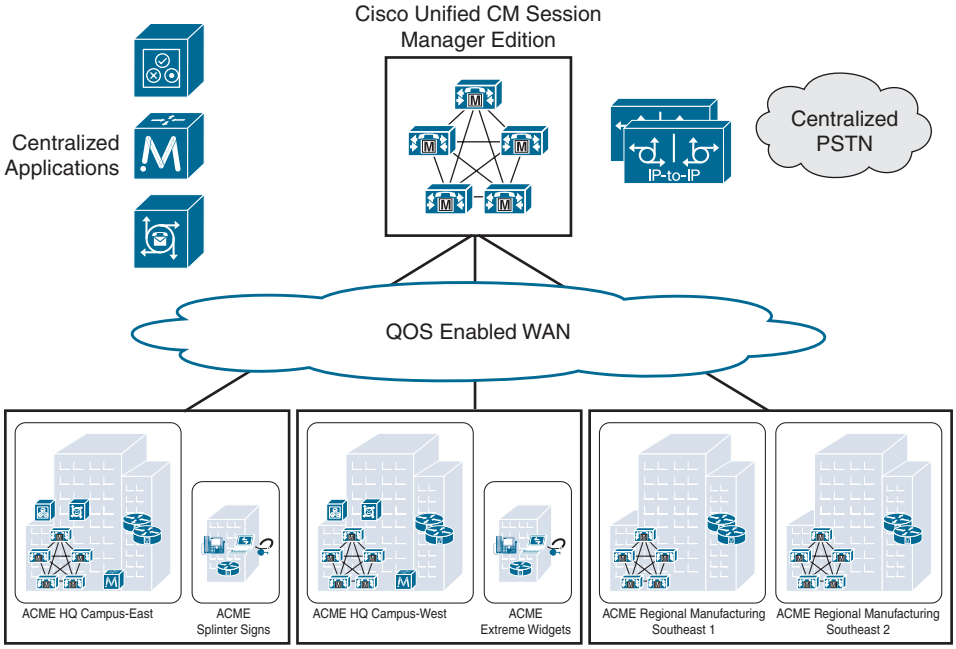


Figure 4-3 *Distributed Deployment Model*

Each of the deployment models we have discussed can be combined with cloud-based services like Cisco Webex. When this is done, it is considered a hybrid deployment, and the same considerations should be taken into account for availability. If a service or feature is considered mission critical, you should take care to understand the risks involved if it is moved to the cloud. Additionally, an entire UC deployment (outside of endpoints) can be moved to the cloud using Cisco WebEx and even Cisco Hosted Collaboration Services (HCS). We cover these technologies in more detail in Chapter 12, as we build on the initial UC environment.

https://t.me/learningnets

@cisco_in_persian_channel

An Overview of How to Secure Cluster Communications

Now that we've discussed the basic deployment types, we address what types of communications take place within a Cisco UC environment and whether they are secure. We look at a few different types of communication that take place. The first of these is the communication between the different UC application nodes, such as Cisco Unified CM Publisher and Subscribers. This type of communication is called Intra-Cluster Communication Signaling (ICCS) and is shown in Figure 4-4.

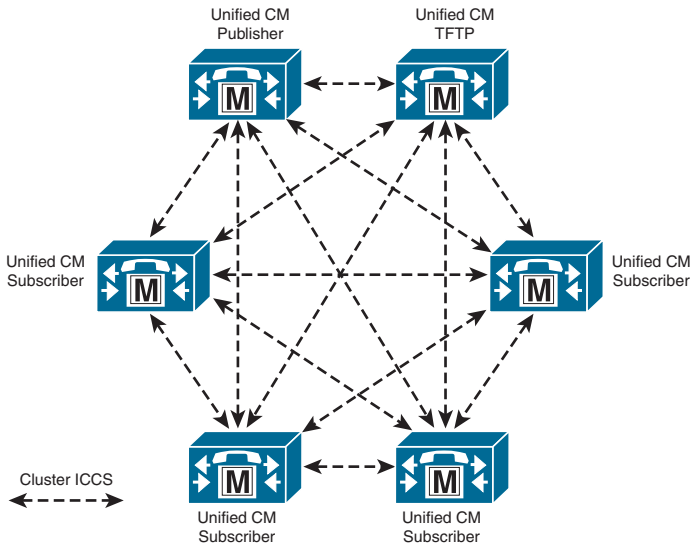


Figure 4-4 *ICCS Traffic*

The ICCS traffic types are classified as either priority or best effort. Priority ICCS traffic is marked with IP Precedence 3 (DSCP 24 or PHB CS3). Best-effort ICCS traffic is marked with IP Precedence 0 (DSCP 0 or PHB BE).

There are different types of intracluster traffic for Unified CM:

- ICCS real-time traffic consists of signaling, Call Admission Control, and other information regarding calls as they are initiated and completed. ICCS uses a Transmission Control Protocol (TCP) connection between all servers that have the Cisco CallManager Service enabled. The connections are a full mesh between these servers. This priority ICCS traffic is marked dependent on release and service parameter configuration.
- Traffic from the IBM Informix Dynamic Server (IDS) database provides the main configuration information. This data is pushed from the Unified CM Publisher to the subscribers in the cluster, as shown in Figure 4-5. The IDS traffic may be reprioritized in line with Cisco QoS recommendations to a higher-priority data service

https://t.me/learningnets
@cisco_in_persian_channel

if required by the business needs. An example of this is extensive use of Extension Mobility, which relies on IDS database configuration.

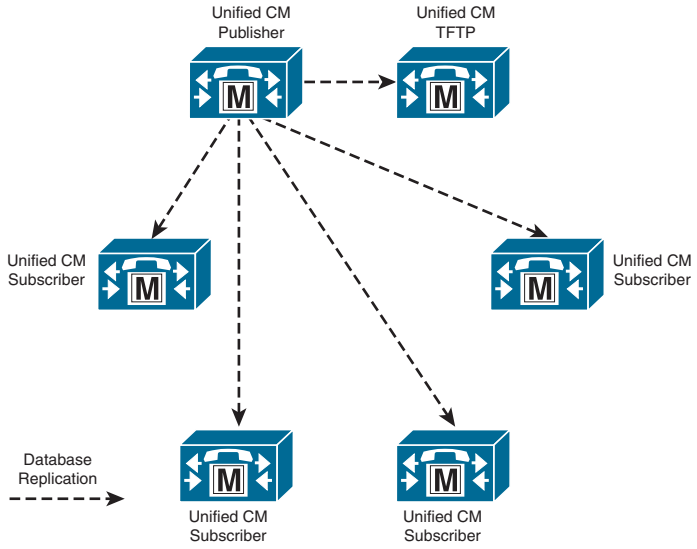


Figure 4-5 *Database Replication*

- Firewall management traffic (as shown in Figure 4-6) is used to authenticate the subscribers to the publisher to access the publisher's database. The management traffic flows between all servers in a cluster. The management traffic may be prioritized in line with Cisco QoS recommendations to higher-priority data if required by the business needs.

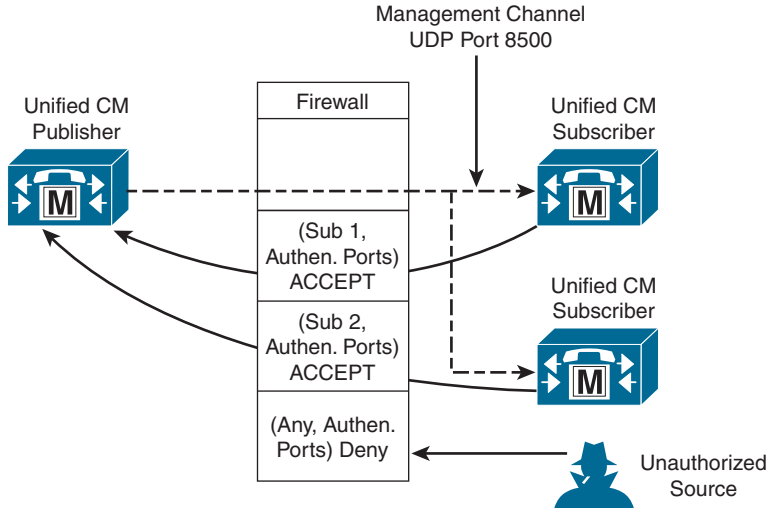


Figure 4-6 *Firewall Management Traffic*

https://t.me/learningnets

@cisco_in_persian_channel

- CTI Manager real-time traffic is used for CTI devices involved in calls or for controlling or monitoring other third-party devices on the Unified CM servers. This traffic is marked as priority ICCS traffic and exists between the Unified CM server with the CTI Manager and the Unified CM server with the CTI device.
- CDR and CAR records are logged by the call processor handling the call and are periodically replicated to the Publisher node of the cluster.

Usually, ICCS traffic is not encrypted, so additional steps need to be taken to secure this traffic. Within the core Cisco UC applications running Cisco's voice operating system (VOS), encryption is usually configured using IPsec, which is discussed later in this chapter. Some UC applications like Cisco Meeting Server or Cisco Expressway can use X.509 digital certificates to secure this traffic using TLS. In later chapters specific to those Cisco UC applications, we discuss the steps that can be taken to secure them.

Note Configuring and enabling IPsec within a UC environment adds complexity and overhead to the network, affects scalability, and impacts day-to-day operations of the environment. In most cases, IPsec is not needed because the UC applications are deployed within protected and secured data centers. In this book, however, we cover the steps required to configure IPsec for use as a reference.

Another type of communication occurs between the different applications and endpoints. Typically, this communication between applications within a Cisco UC environment happens using several protocols. The most common protocols used between UC applications and endpoints are Session Initiation Protocol (SIP), Skinny Client Control Protocol (SCCP), Media Gateway Control Protocol (MGCP), and the H.323 suite. These protocols are used for device registrations, call signaling, and presence notification. Like the different types of ICCS, this traffic is also sent in the clear, and the process to secure these types of communication usually involves digital certificates and/or IPsec. Device registrations can be secured through several methods either using the Manufacturer Installed Certificate (MIC) or Local Session Certificates (LSCs), and in newer versions of Unified CM, OAuth can be used to secure the signaling and media between endpoints and other UC applications. The process of securing the registration of devices is discussed in Chapter 7, where the steps required to enable its deployment and the different methods are discussed in detail. Call signaling using SIP or SCCP uses a digital certificate to transport the shared keys that are used to secure the signaling and media. While SCCP is relevant mostly to Cisco Unified CM and Cisco Unity Connection, SIP is more prevalent across the UC environment and should be used in most instances for integrating the different UC applications together. Lastly, the signaling for MGCP and H.323 is secured with the use of IPsec between the voice gateway or endpoint and Cisco Unified CM.

https://t.me/learningnets

@cisco_in_persian_channel

IPsec and digital certificates are used to secure protocols that do not have built-in security mechanisms for their communications within a UC environment. When IPsec is used, you should understand the network requirements and the impact on the overall network before implementing it. Ancillary network services like Network Time Protocol (NTP) serve an important role when using digital certificates and IPsec because it not only provides consistent timing across the network devices but also is used to validate whether the digital certificates are valid. Because of the additional network services involved, additional complexity and interworking between devices can arise and cause an outage within the environment. An example of this complexity is the misconfiguration of an NTP server in a network where the year is set five years or even one year in the future; this configuration could cause certificates to be considered invalid, and connections using those certificates could be refused.

The last type of communication to discuss is the communication between UC applications that use ancillary services (like NTP) and Computer Telephony Integration (CTI). These are just a few of the services to consider when looking to secure the signaling within a Cisco UC environment. As discussed previously, using NTP is important for keeping a consistent time across the environment for several reasons, such as consistent log timestamps (to ease troubleshooting, forensics, and so on), and for ensuring the validity of certificates in a PKI infrastructure. When NTP is used, using authentication is recommended but not required. This authentication is not encryption but merely a means to verify the NTP traffic coming from the host is legitimate.

Two different types of authentication are supported by Cisco UC applications: symmetric-key and auto-key. Symmetric-key authentication in NTP version 3 uses the same preshared key to encrypt plaintext data and decrypt ciphertext data. Cisco currently supports the use of SHA1 hashing for symmetric-key NTP authentication. One downside to using symmetric-key authentication is that it does not scale well.

Auto-key authentication uses public key infrastructure (PKI) algorithms from the OpenSSL library that contains various message digest, digital signature, and encryption methods. Auto-key, defined in NTP version 4 (https://tools.ietf.org/html/rfc5906), uses digital certificates to verify the source using timestamped digital signatures. Currently, this method of authentication is supported only when the Federal Information Processing Standard (FIPS) and Common Criteria are enabled.

There is some give-and-take with these authentication mechanisms; depending on the network, they may dictate which is used. Typically, when NTP authentication is enabled, a symmetric-key is used to authenticate the messages between server/peer/client, but as stated previously, this can become cumbersome when updates are required and many devices must have their configurations updated. Using NTP authentication with auto-key is one way to work around the issue of scalability. Auto-key was found to have some flaws that led to auto-key version 2, or Network Time Security, which is currently unsupported by Cisco UC applications.

https://t.me/learningnets

@cisco_in_persian_channel

Why is it important to implement NTP authentication? The simple answer is to limit the scope and types of attacks that can disrupt time synchronization within the network. Specific examples of these attacks are

- **Replay and spoofing attacks:** These attacks happen when an attacker can intercept and replay messages between a client and server or server and client. These messages would be properly verified and result in replies potentially being delayed. These types of attacks could result in time synchronization errors. Because the packets have been either spoofed or replayed, they can be difficult to identify because this type of error could be linked to a normal network issue.
- **Man-in-the-middle attacks:** These attacks allow an attacker to intercept, modify, and replay NTP messages. They can lead to time errors on the host or the redistribution of the wrong time to other hosts.
- **Denial-of-service attacks:** These attacks cause the disruption of time synchronization within the environment.

The last aspect that we covered for the information passed within a UC environment is not so much for the information being passed but how to restrict where it can be sent and accessed from. This is nothing more than implementing an intelligent set of access control lists (ACLs) or firewalls between the UC infrastructure (UC applications and endpoints) and the rest of the network. Using ACLs as an example, network administrators can create either a generic ACL for the UC applications and then an ACL for endpoints, or they can be more granular, such as a separate ACL for each UC application. This choice is completely up to network administrators and what fits their network needs. The recommendation is to review the comprehensive list of ports and protocols used within a Cisco UC environment by consulting the TCP and UDP Port Usage Guide for the relevant Cisco UC application that is being secured.

NTP Authentication Enablement and Verification

The first place to start securing is NTP. It is typically an easy target that is already either configured or easily configured on existing NTP servers. If it is not already configured, the first step is to enable authentication on the NTP servers within the network. Because the Cisco UC application supports SHA1, you need to verify that the NTP servers are NTPv4 compliant and support the use of SHA1 keys.

The next step is to log in to the CLI of the Cisco Unified CM Publisher and verify the current NTP server status with the command **utils ntp status**, as shown in Example 4-1. Checking the current status serves two purposes: first, it allows you, as administrator, to verify that the configured NTP servers are reachable, and it also allows you to verify that the correct NTP servers are configured in the system.

https://t.me/learningnets

@cisco_in_persian_channel

Example 4-1 *Verify Current NTP Status*

```
admin:utils ntp status
ntpd (pid 16694) is running...
     remote      refid          st t   when    poll reach    delay    offset   jitter
==============================================================================
*10.1.10.108    10.1.10.97     2  u   18      64   1        0.213    0.086    0.52

synchronised to NTP server (10.1.10.97) at stratum 2
   time correct to within 37 ms
   polling server every 1024 s

Current time in UTC is : Sat Aug 29 12:00:07 UTC 2020
Current time in America/New_York is : Sat Aug 29 08:00:07 EDT 2020
```

Note Example 4-1 reflects a single configured NTP server. It is the recommended best practice to configure multiple NTP servers to allow for redundancy.

After verifying the server is configured, you can check to see if NTP authentication is already enabled by using the **utils ntp auth symmetric-key status** command (see Example 4-2). The reason behind verifying the existing NTP authentication state is to verify whether any NTP servers are already configured for authentication using a symmetric-key, and if so, what the current status is. If you are trying to configure symmetric-key authentication and auto-key is already configured, an error is displayed alerting you that auto-key should be disabled first.

Example 4-2 *Verify NTP Authentication Disabled*

```
admin:utils ntp auth symmetric-key status
10.1.10.108  :  NTP Authentication is disabled.
ind assid status  conf reach auth condition  last_event cnt
============================================================
  1 54322  963a   yes   yes  none  sys.peer    sys_peer   3
```

Now that you've verified NTP authentication as being disabled, you can enable authentication. This process can be repeated for the Primary node within a cluster; the secondary servers or Subscriber nodes synchronize their time with the Primary node or Publisher node. You use the **utils ntp auth symmetric-key enable** command shown in Example 4-3 to start the process to enable symmetric-key authentication for NTP. This process prompts you to select the NTP server that will be enabled for authentication, which key ID to use, and the symmetric-key to use for authentication. The key ID used for authentication does not need to match the key ID configured within the NTP server.

https://t.me/learningnets

@cisco_in_persian_channel

Example 4-3 *Enable NTP Symmetric-Key Authentication*

```
admin: utils ntp auth symmetric-key enable
The List of NTP servers Configured:
1. 10.1.10.108
q. press q to exit
Enter the selection for which to configure NTP Authentication:
1
Please enter the Key ID [1-65534]:
1
Please enter the Symmetric Key of the NTP Server (SHA1):
Restarting NTP
Please run the 'utils ntp auth symmetric-key status' to check the status of NTP
  Authentication.
```

Give the NTP service a couple of minutes to come back into service and then verify that it is up by using the **utils ntp status** command shown in Example 4-4. The actual synchronization can take a few more minutes to complete.

Example 4-4 *NTP Service Verification*

```
 admin: utils ntp status
ntpd (pid 9254) is running...
  remote           refid        st t when poll reach   delay   offset  jitter
==============================================================================
*10.1.10.108     10.1.10.97       2 u   54  64    1      0.159  0.023   0.018
synchronised to NTP server (10.1.10.108) at stratum 3
   time correct to within 965 ms
   polling server every 64 s
Current time in UTC is : Sat Aug 29 13:06:44 UTC 2020
Current time in America/New_York is : Sat Aug 29 09:06:44 EDT 2020
```

The last steps in the process shown in Example 4-5 are to verify that NTP authentication is enabled and okay. This process can take a few additional minutes to complete after the NTP service starts.

Example 4-5 *NTP Authentication Enabled and Synchronized*

```
admin: utils ntp auth symmetric-key status
10.1.10.108  :  NTP Authentication is enabled.
ind assid status  conf reach auth condition  last_event cnt
===========================================================
  1 22480 f63a    yes  yes    ok  sys.peer   sys.peer    3
```

https://t.me/learningnets

@cisco_in_persian_channel

Securing Intra-Cluster Signaling and Traffic

With NTP authentication configured, we can examine the process of securing ICCS traffic within the environment. Here we look only at the traffic between nodes of the same UC application. In later chapters we cover securing the different integration types between the core UC applications, specifically securing SIP trunks using TLS.

The first step in securing ICCS communication is to evaluate where in the network the communication is taking place. The current UC environment can be considered in a secure location because the applications are deployed within the campus data centers. With that in mind, simple ACLs or firewall rules based on the relevant TCP/UDP Port Usage Guides for the UC application should suffice in securing their communication. However, as the environment grows, to account for a UC environment that is distributed across the WAN, you will need to account for ICCS traversing the WAN. For this deployment model, it is necessary to validate whether the ICCS between the application nodes needs to be secured. The recommended solution is to deploy IPsec between these nodes, and ideally, IPsec will be configured within the infrastructure rather than the actual UC applications due to the added overhead encountered when IPsec is configured. In this instance, however, we explain the process to configure IPsec within UC applications.

This configuration process is common across the core UC applications that are built on the voice operating system and is discussed in depth in Chapter 7. This section focuses on enabling IPsec to secure ICCS. Additionally, for IPsec to communicate successfully, it should be deployed in a full mesh between the nodes and can be secured using a digital certificate or preshared key.

The process to configure IPsec to use certificate-based authentication can be broken down into three steps:

1. Generate and sign the certificates used to secure the IPsec policies.
2. Enable IPsec to configure the policies.
3. Verify connectivity after IPsec enablement.

The first step is the generation and signing of certificates used to secure the IPsec policies. These certificates need to be shared between the nodes in the UC application prior to the IPsec configuration. These certificates should be signed by the enterprise certificate authority (CA) to reduce complexity as additional systems are integrated. This step requires the UC administrator to not only get the IPsec certificates signed by the enterprise CA but also gather the certificates of the root and intermediate CAs in the certificate path that signed the IPsec certificate. Table 4-1 shows the IPsec certificate requirements.

https://t.me/learningnets

@cisco_in_persian_channel

Table 4-1 *IPsec CSR Key Usage Extensions*

		Extended Key Usage			Key Usage				
	Multi server	Server Authentication (1.3.6.1.5.5.7.3.1)	Client Authentication (1.3.6.1.5.5.7.3.2)	IP Security End System (1.3.6.1.5.5.7.3.5)	Digital Signature	Key Encipherment	Data Encipherment	Key Cert Sign	Key Agreement
ipsec	N	Y	Y	Y	Y	Y	Y	N	N

The first step in generating the IPsec certificate is to create the certificate signing request (CSR). To do so, you log in to the Primary node Cisco Unified Operating System Administration GUI and navigate to **Security > Certificate Management**. Next, you select **Generate CSR**, and when the pop-up opens, you select **ipsec** for Certificate Purpose. Depending on your requirements, you also set the Key Length and Hash Algorithm. After these settings are updated, select **Generate**. These steps are shown in Figure 4-7.

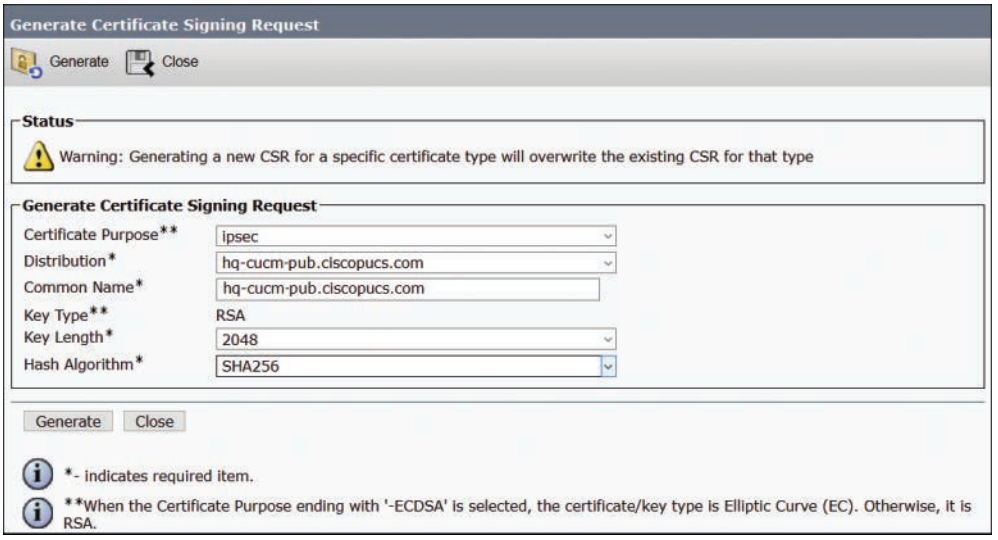


Figure 4-7 *IPsec CSR Generation*

You can download the IPsec CSR by selecting **Download CSR** and following the prompts. If you have multiple CSRs pending, you may need to select the drop-down for Certificate Purpose and select **ipsec** from the list of CSRs. Submit the downloaded CSR to your enterprise certificate authority to be signed. Once the CSR is signed, the root and intermediate CA certificates need to be uploaded to the IPsec-Trust certificate store, and the signed IPsec identity certificate should be uploaded to the IPsec certificate store.

https://t.me/learningnets
@cisco_in_persian_channel

You accomplish this task in a similar manner to creating a CSR. This time, after you are logged in to Cisco Unified OS Administration, select **Upload Certificate/Certificate chain** to load the new certificates. Select **ipsec** from the Certificate Purpose drop-down, click **Browse**, and then select the new IPsec identity certificate. When these steps are complete, select **Upload**.

The full chain of trust for the certificate chain needs to be loaded in one of two ways. The easier way is to upload the certificate bundle that comes from the CA and typically contains the identity certificate (ipsec), root CA, and intermediate CA (issuing CA). When this method is used, the certificates can be uploaded in a single attempt. The second method is to upload the certificates individually. When this method is used, you need to upload the certificates starting with the root CA certificate and intermediate CA certificates using the Certificate Purpose ipsec-trust; then you upload the identity certificate (ipsec) using the Certificate Purpose ipsec. Chapter 7 covers working with certificates for the different Core Cisco UC applications. One aspect that is overlooked from time to time is restarting services when certificates are loaded. When the IPsec certificate is loaded, you need to restart the following services:

- Cisco DRF Master (Publisher only)
- Cisco DRF Local (All nodes)

Each application node also needs to have the IPsec certificates of the remote nodes uploaded to their IPsec-Trust certificate stores; this topic is discussed further in Chapter 7. Figure 4-8 reflects the successful upload of the IPsec identity cert, root, and intermediate certificates.

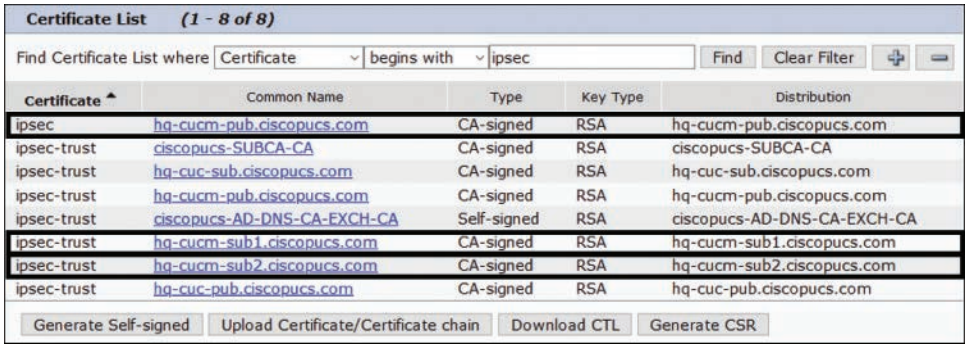


Figure 4-8 *IPsec Certificate List*

The second step in enabling IPsec is to configure the policies. Note that the UC applications use IPsec policies that use transport mode rather than tunnel mode. When you use transport mode, just the payload of the identified traffic is encrypted. Additionally, depending on whether FIPS 140-2 is enabled within the application, not all encryption and hashing algorithms may be available. The specific requirements for FIPS and the

https://t.me/learningnets
@cisco_in_persian_channel

restrictions imposed on the cryptographic algorithms allowed are discussed in Chapter 5, "Hardening the Core Cisco UC Appliance Operating Systems."

At this point, log in to the Cisco Unified OS Administration GUI, navigate to **Security > IPsec Configuration**, and select **Add New**. You then need to provide the information found in Table 4-2 to configure the new policy.

Table 4-2 *IPsec Policy Configuration*

IPsec Policy Details	
Policy Group Name*	Specifies the name of the IPsec policy group. The name can contain only letters, digits, and hyphens. **Note** Do not use more than one hyphen when creating the Policy Group Name.
Policy Name*	Specifies the name of the IPsec policy group. The name can contain only letters, digits, and hyphens. **Note** Do not use more than one hyphen when creating the Policy Name.
Authentication Method**	Has the following two options: ■ If Preshared Key is selected, the Preshared Key field is editable. (It is disabled when FIPS is enabled.) ■ If Certificate is selected, the Preshared Key field is not present and the Certificate Name field is editable.
Peer Type*	Specifies that the peer type is different. This is the only option and specifies to the system that the IPsec identity certificate for the remote peer will be specified.
Certificate Name*	Specifies the filename of the remote peer IPsec certificate. For example: hq-cucm-pub.ciscopucs.com.pem
Destination Address*	Specifies the IP address of the destination (FQDN is not supported).
Destination Port*	Specifies the port number at the destination.
Source Address*	Specifies the IP address of the source (FQDN is not supported).
Source Port*	Specifies the port number at the source.
Mode*	Specifies Transport mode.
Remote Port*	Specifies the port number to use at the destination.
Protocol*	Specifies the specific protocol, or Any: ■ TCP ■ UDP ■ Any

https://t.me/learningnets

@cisco_in_persian_channel

IPsec Policy Details	
Encryption Algorithm*	Enables you to choose the encryption algorithm from the drop-down list. Choices are ■ DES (disabled when FIPS is enabled) ■ 3DES ■ AES 128 ■ AES 256
Hash Algorithm*	Specifies the hash algorithm: ■ SHA1: Hash algorithm that is used in Phase One IKE negotiation ■ MD5: Hash algorithm that is used in Phase One IKE negotiation (disabled when FIPS is enabled)
ESP Algorithm*	Enables you to choose the ESP algorithm from the drop-down list. Choices are ■ AES 128 ■ AES 256 ■ DES (deprecated and disabled when FIPS is enabled) ■ 3DES
Phase 1 DH Group	
Phase One Life Time*	Specifies the lifetime for Phase One IKE negotiation in seconds. The default of 3600 is sufficient.
Phase One DH*	Enables you to choose the Phase One DH value from the drop-down list. The choices are 2, 5, 14–18 depending on application version. Currently, the default DH Group 14 is sufficient for most deployments. ■ DH Group 2: 1024 bit modulus ■ DH Group 5: 1536 bit modulus ■ DH Group 14: 2048 bit modulus ■ DH Group 15: 3072 bit modulus ■ DH Group 16: 4096 bit modulus ■ DH Group 17: 6144 bit modulus ■ DH Group 18: 8192 bit modulus
Phase 2 DH Group	
Phase Two Life Time*	Specifies the lifetime for Phase Two IKE negotiation in seconds. The default of 3600 is sufficient.

https://t.me/learningnets

@cisco_in_persian_channel

Phase 2 DH Group	
Phase Two DH*	Enables you to choose the Phase Two DH value from the drop-down list. The choices are 2, 5, 14–18 depending on application version. Currently, the default DH Group 14 is sufficient for most deployments. ■ DH Group 2: 1024 bit modulus ■ DH Group 5: 1536 bit modulus ■ DH Group 14: 2048 bit modulus ■ DH Group 15: 3072 bit modulus ■ DH Group 16: 4096 bit modulus ■ DH Group 17: 6144 bit modulus ■ DH Group 18: 8192 bit modulus
IPsec Policy Configuration	
Enable Policy	Enables the policy when the checkbox is selected.

Figure 4-9 shows a sample IPsec tunnel configured between the Unified CM Publisher and Subscriber. When the tunnels are configured, you should not enable the IPsec policy when initially created. Because the policies are bidirectional, it is best to wait until the policy is configured and on the remote node before enabling it on the local policy. This method limits the impact to communication.

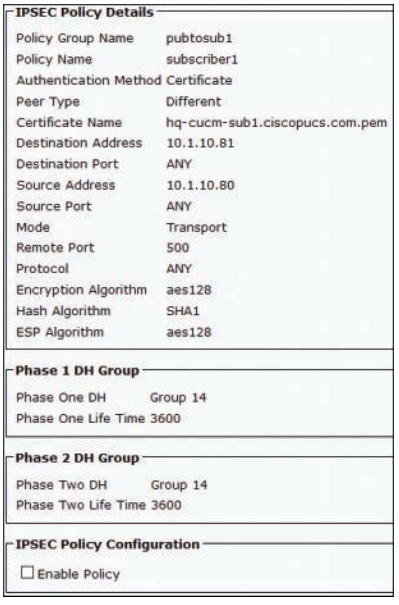


Figure 4-9 *IPsec Policy Creation*

https://t.me/learningnets

@cisco_in_persian_channel

Once the policy is configured, you can go to the remote node that was configured in the policy and create an IPsec policy pointing back to the first node, this time enabling the policy on both ends. Figure 4-10 shows the order of creating the IPsec policies within a three-node Unified CM cluster starting with the Unified CM Publisher node.

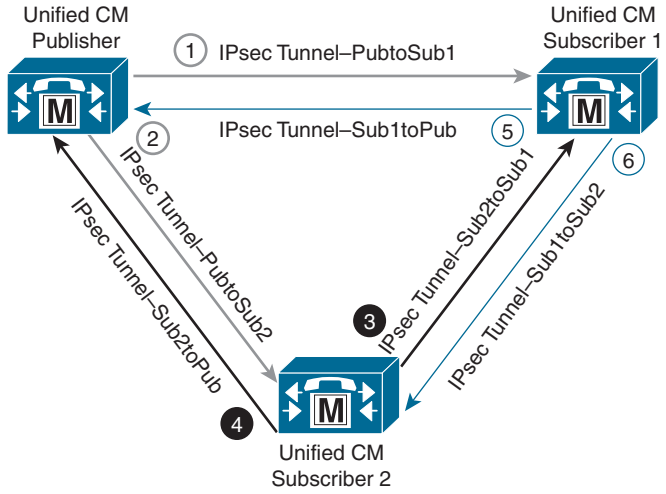


Figure 4-10 *IPsec Full Mesh*

After each IPsec policy is created and enabled, you should verify it is functional. Failure to verify the policy is configured and working correctly can result in database replication errors and other types of outages.

To verify the IPsec policy that was just created, within the Cisco Unified OS Administration GUI, navigate to **Services > Ping**. From here, perform a ping to the remote node with the option for validate IPsec selected, as shown in Figure 4-11.

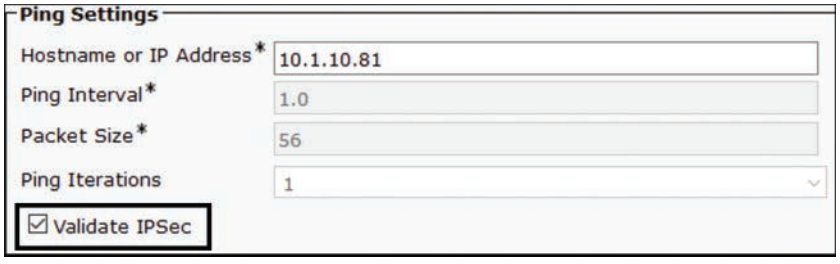


Figure 4-11 *IPsec Policy Verification*

The expected result is "Successfully validated IPSec connection," as shown in Figure 4-12.

https://t.me/learningnets
@cisco_in_persian_channel

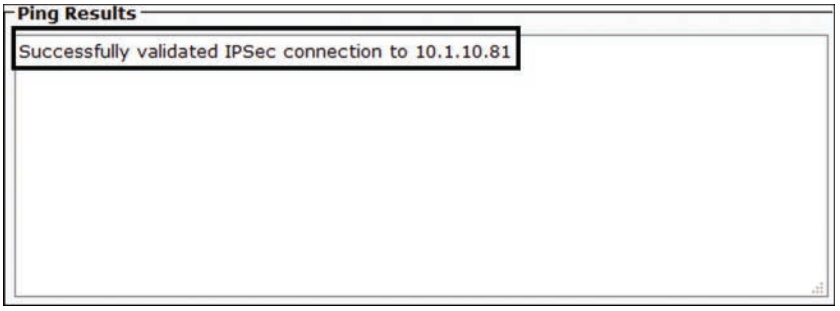


Figure 4-12 *IPsec Policy Verification Successful*

Any result other than successful should be considered a failure and needs to be resolved before you create additional policies.

There are some caveats to be aware of once the IPsec policies have been created and enabled. First, NTP is important, and when NTP is not synchronized, communications can fail, especially when using certificates for the IPsec policies. Second, if certificates are used (this is required if the cluster is FIPS enabled) in the IPsec policy, it is advisable to set up the Certificate Monitor within Cisco Unified OS Administration to notify the UC administrator when the IPsec certificates are about to expire. If the certificates in use expire, the communication fails, and an outage occurs. The Certificate Monitor configuration, by default, sends a notification when a certificate will expire within 30 days and will resend the notification every 7 days. A third aspect for IPsec policies is that you cannot edit them after they have been saved other than to enable or disable them. The inability to edit matters if or when you need to change the host name or IP address in use for a given UC application node within the cluster. If the certificates are being renewed due to their validity periods expiring, the policies should not need to be rebuilt because the certificate names should be consistent.

There are two caveats about using IPsec policies. First, disabling them prior to performing an upgrade is recommended. Upgrades can be time-consuming to accomplish successfully, so by disabling the IPsec policies during this process and then enabling them after the upgrade is successful, you can remove some of the complexity from the process. A second caveat to IPsec policies is that they must be disabled if you are changing IP addresses and host names or changing the certificate used in the policy.

Securing the Signaling Traffic to IOS Voice and Analog Gateways

Just like securing the intra-cluster communication between UC application nodes, the protocols used between the UC applications and endpoints, gateways, and trunks are typically sent in the clear. Additional steps to secure these protocols must be taken if

https://t.me/learningnets
@cisco_in_persian_channel

an organization requires that they be secured. This section focuses on securing voice gateways configured for MGCP and H.323 to demonstrate the process of creating the IPsec policies between the gateway and Cisco Unified CM, as shown in Figure 4-13.

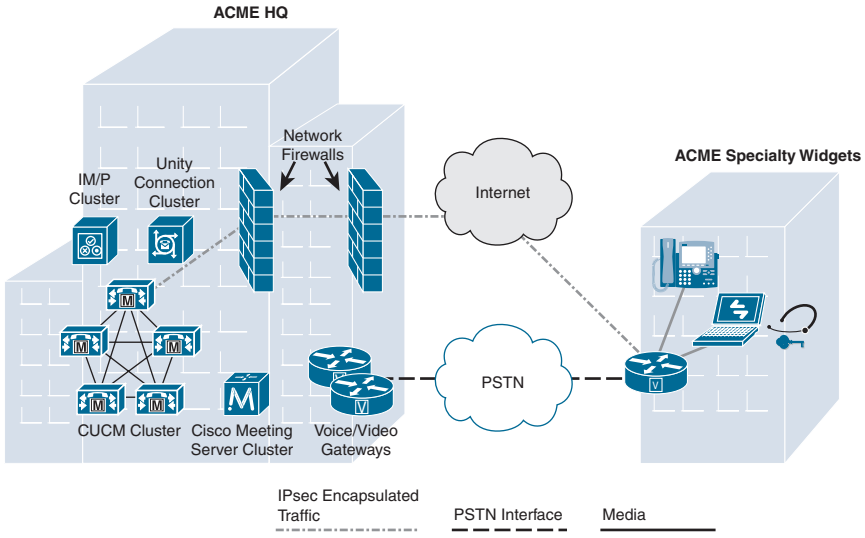


Figure 4-13 *Securing Signaling Protocols*

When MGCP and H.323 are used for signaling, they are not secured by default. The standard practice is to create IPsec policies that can then be used to secure the signaling. The media can be sent securely using SRTP without IPsec configured. However, without a method to encrypt the signaling like IPsec, the keys used for SRTP are sent in the clear.

Using digital certificates rather than a preshared key is the preferred method when creating the IPsec policies due to the reduced complexity and scalability that this approach provides. Digital certificates, their uses, and how they apply to Cisco Unified Communications are discussed in detail in Chapter 7, with later chapters also discussing the individual UC applications.

The process to implement IPsec policies between Cisco Unified CM and a voice gateway is not much different from configuring IPsec policies to secure ICCS. There is a five-step process to properly configure IPsec between Cisco Unified CM and a voice gateway:

1. Generate a CA-signed certificate for the voice gateway.
2. Generate CA-signed certificates for Cisco Unified CM.
3. Import the CA-signed certificates to the respective devices.
4. Configure the IPsec policy in the voice gateway.
5. Configure the IPsec policy in Cisco Unified CM.

https://t.me/learningnets

@cisco_in_persian_channel

First, you need to verify some basic settings, namely, that the voice gateway has a meaningful host name and that a domain is configured. You can do this by using the **show running-config** command shown in Example 4-6.

Example 4-6 *show running-config Filtered Output*

```
sited-rtr# show running-config | in hostname | domain-name
hostname sited-rtr
 domain-name ciscopucs.com
```

Starting with the voice gateway, you can generate the CSR so that it can be signed by an enterprise CA. This scenario utilizes a standard Windows CA to generate the digital certificates used for IPsec.

Note Self-signed certificates are not supported for IPsec between the voice gateway and Cisco Unified CM.

A couple of steps take place in generating a CSR for the gateway. The first of these is to create the RSA key pair shown in Example 4-7 with the **crypto key generate rsa modulus 2048 label ipsec** command. The optional keyword **exportable** can be added if the key pair needs to be exported for backup purposes.

Example 4-7 *Create RSA Keys for IPsec Tunnel*

```
sited-rtr(config)# crypto key generate rsa modulus 2048 label ipsec
The name for the keys will be: ipsec

% The key modulus size is 2048 bits
% Generating 2048 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 9 seconds)
```

Next, you create the trust points for the CA certificates that will sign the IPsec certificate for the gateway. In this scenario a root and intermediate CA need to have their certificates imported.

Example 4-8 shows how to create the trust point with the **crypto pki trustpoint Root-CA** command. The **enrollment terminal** option allows the base64-encoded certificate to be copied into the gateway configuration, and **revocation-check** is not enabled. Lastly, the **crypto pki authenticate Root-CA** command prompts for the import of the root CA certificate.

https://t.me/learningnets

@cisco_in_persian_channel

Example 4-8 *Configure IPsec Tunnel Trustpoint*

```
sited-rtr(config)# crypto pki trustpoint Root-CA
sited-rtr(ca-trustpoint)# enrollment terminal
sited-rtr(ca-trustpoint)# revocation-check none
sited-rtr(ca-trustpoint)# exit
sited-rtr(config)# crypto pki authenticate Root-CA

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
MIIDhzCCAm+gAwIBAgIQMAe3B21axIZGiqHmifSuNjANBgkqhkiG9w0BAQsFADBW
MRMwEQYKCZImiZPyLGQBGRYDY29tMRkwFwYKCZImiZPyLGQBGRYJY2lzY29wdWNz
***omitted***
QylU5dwQbAIoeQo6H+thSjN+VU3kjG8ve35z2hO8woucSz99vfkCHDw0FB3Xsg0T
o0byehjZYKO+ZIrNTYZflfiMB3c0iCSi5udvniZm+lnVkmZD448/Vma6gA==
-----END CERTIFICATE-----

Certificate has the following attributes:
       Fingerprint MD5: 767891F9 5EFE4505 B379E299 52DBCD31
      Fingerprint SHA1: 6C179AB0 2D9AEA52 19BB9FE3 6025A757 BD6B1216

% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported
```

The IPsec certificate trustpoint is configured with the **crypto pki trustpoint IPsec_Traffic** command shown in Example 4-9.

Example 4-9 *Configure IPsec Tunnel Certificate*

```
sited-rtr(config)# crypto pki trustpoint IPsec_Traffic
sited-rtr(ca-trustpoint)# enrollment terminal
sited-rtr(ca-trustpoint)# serial-number none
sited-rtr(ca-trustpoint)# fqdn sited-rtr.ciscopucs.com
sited-rtr(ca-trustpoint)# ip-address none
sited-rtr(ca-trustpoint)# subject-name CN=sited-rtr.ciscopucs.com,OU=Cisco
  Press,O=Practical UC Security,L=RTP,ST=NC,C=US
sited-rtr(ca-trustpoint)# revocation-check none
sited-rtr(ca-trustpoint)# ocsp disable-nonce
sited-rtr(ca-trustpoint)# rsakeypair IPsec
sited-rtr(ca-trustpoint)# hash sha256
```

https://t.me/learningnets

@cisco_in_persian_channel

The trustpoint in Example 4-10 imports the intermediate CA certificate used to authenticate the IPsec identity certificate with the **crypto pki authenticate IPsec_Traffic** command.

Example 4-10 *Import Signing Certificate Authority Digital Certificate*

```
sited-rtr(config)# crypto pki authenticate IPsec_Traffic

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
MIIFezCCBGOgAwIBAgITHAAAAANnNT673RXs3wAAAAAAAzANBgkqhkiG9w0BAQsF
ADBWMRMwEQYKCZImiZPyLGQBGRYDY29tMRkwFwYKCZImiZPyLGQBGRYJY2lzY29w
***omitted***
5guHuCYWzxB7ZejiMF/V37j0EalvPkHeFvDBHI8mfKph4BGjgoUd2pbCumd7aRJK
ldeGtC4Hu3g2H4oVv8aG
-----END CERTIFICATE-----

Certificate has the following attributes:
       Fingerprint MD5: BAC09690 17AB5389 723B540D 77CE2FF5
      Fingerprint SHA1: 037EE4BB 4547C089 CD36F95C B9A805DC F139CC6C
Certificate validated - Signed by existing trustpoint CA certificate.

Trustpoint CA certificate accepted.
% Certificate successfully imported
```

You generate the CSR for the IPsec certificate that will be signed by the intermediate CA by using the **crypto pki enroll IPsec_Traffic** command. Example 4-11 shows the output that will be displayed to the terminal so that it can be copied out and submitted to the CA.

Example 4-11 *Create and Export the IPsec Tunnel Certificate Signing Request*

```
sited-rtr(config)# crypto pki enroll IPsec_Traffic
% Start certificate enrollment..

% The subject name in the certificate will include: CN=sited-rtr.ciscopucs.com,
  OU=Cisco Press,O=Practical UC Security,L=RTP,ST=NC,C=US
% The subject name in the certificate will include: sited-rtr.ciscopucs.com
Display Certificate Request to terminal? [yes/no]: yes
Certificate Request follows:

MIIDDzCCAfcCAQAwgagxCzAJBgNVBAYTAlVTMQswCQYDVQQIEwJOQzEMMAoGA1UE
BxMDUlRQMR4wHAYDVQQKExVQcmFjdGljYWwgVUMgU2VjdXJpdHkxFDASBgNVBAsT
```

https://t.me/learningnets

@cisco_in_persian_channel

```
***omitted***
lrsi2PjhRkFgkdgh73p9Y1j9Wh3H3R7CjhTiazO/aO63YeP0GyBRJOY/wpeRhKO7
z45Nwmlrg/DxKlXW4p64ag2wtg==

---End - This line not part of the certificate request---

Redisplay enrollment request? [yes/no]: no
sited-rtr(config)#
```

The CSR that is displayed should be copied to a text file and saved with a .CSR extension. The header **-----BEGIN CERTIFICATE-----** and the footer **-----END CERTIFICATE-----** can be added to the output of the CSR to make it more readable. This CSR can then be submitted to the same CA that was used to authenticate the trustpoint so that it can be signed.

The signed certificate needs the attributes outlined in Table 4-3 applied to it.

Table 4-3 *IPsec Key Usage*

	Extended Key Usage				Key Usage				
	Multi server	Server Authentication (1.3.6.1.5.5.7.3.1)	Client Authentication (1.3.6.1.5.5.7.3.2)	IP Security End System (1.3.6.1.5.5.7.3.5)	Digital Signature	Key Encipherment	Data Encipherment	Key Cert Sign	Key Agreement
ipsec	N	Y	Y	Y	Y	Y	Y	N	N

With the signed certificate returned from the CA, it is imported into the IPsec policy on the voice gateway with the **crypto pki import IPsec_Traffic certificate** command.

Next, you can create an access-list using Example 4-12 as a reference to identify traffic that should be secured with the IPsec policy. This traffic should be restricted to the bound interface of the protocol and the IP addresses assigned to the Cisco Unified CM nodes.

Example 4-12 *Define Interesting Traffic for IPsec Tunnel*

```
sited-rtr(config)# ip access-list extended IPsec_Traffic
sited-rtr(config-ext-nacl)# remark Identify interesting traffic for IPsec Policy
sited-rtr(config-ext-nacl)# permit ip host 10.1.10.6 host 10.1.10.80
sited-rtr(config-ext-nacl)# permit ip host 10.1.10.6 host 10.1.10.81
sited-rtr(config-ext-nacl)# permit ip host 10.1.10.6 host 10.1.10.82
```

Using Example 4-13 as a reference, you can create the Internet Security Association Key Management Protocol (ISAKMP) policy that will be used between the voice gateway and

https://t.me/learningnets

@cisco_in_persian_channel

Cisco Unified CM nodes. This information must match the IPsec policy configuration in Cisco Unified CM.

Example 4-13 *Configure ISAKMP Policy*

```
sited-rtr(config-ext-nacl)# crypto isakmp policy 10
sited-rtr(config-isakmp)# encryption aes
sited-rtr(config-isakmp)# group 14
sited-rtr(config-isakmp)# lifetime 3600
sited-rtr(config-isakmp)# exit
sited-rtr(config)# crypto isakmp identity dn
sited-rtr(config)# crypto isakmp keepalive 10
sited-rtr(config)# crypto isakmp aggressive-mode disable
```

The next step shown in Example 4-14 is to define the IPsec transform set, which defines the authentication and encryption used across the IPsec tunnel. These settings can vary depending on the deployment and whether FIPS is enabled on the Cisco Unified CM cluster.

Example 4-14 *Configure Transform Set*

```
sited-rtr(config)# crypto ipsec transform-set CUCM12 esp-aes esp-sha-hmac
sited-rtr(cfg-crypto-trans)# mode transport
sited-rtr(cfg-crypto-trans)# crypto ipsec df-bit clear
sited-rtr(config)# no crypto ipsec nat-transparency udp-encapsulation
```

Next, you need to define the crypto map and bind it to the interface, identify where the protocol (e.g., MGCP, H.323) is bound, set the remote destinations, and then specify which transform set to use and which traffic to apply the IPsec tunnel to, as shown in Example 4-15.

Example 4-15 *Configure Crypto Map*

```
sited-rtr(config)# crypto map CUCM12_Map local-address GigabitEthernet0/0
sited-rtr(config)# crypto map CUCM12_Map 10 ipsec-isakmp
sited-rtr(config-crypto-map)# set peer 10.1.10.80
sited-rtr(config-crypto-map)# set peer 10.1.10.81
sited-rtr(config-crypto-map)# set peer 10.1.10.82
sited-rtr(config-crypto-map)# set transform-set CUCM12
sited-rtr(config-crypto-map)# match address IPsec_Traffic
```

The last step in the IPsec configuration in the voice gateway is to apply the crypto map to a physical interface, as shown in Example 4-16.

https://t.me/learningnets

@cisco_in_persian_channel

Example 4-16 *Apply Crypto Map*

```
sited-rtr(config-crypto-map)# interface GigabitEthernet0/0
sited-rtr(config-if)# description uplink to Cisco PUCS Network
sited-rtr(config-if)# ip address 10.1.10.6 255.255.255.0
sited-rtr(config-if)# crypto map CUCM12_Map
```

The remaining steps in implementing IPsec policies to secure the signaling protocols are completed on the Cisco Unified CM cluster and are the same as the process used to secure the ICCS. First, you must upload the voice gateway IPsec certificate to the IPsec-Trust certificate store on each node within the cluster. When that step is complete, you can complete the IPsec policy configuration, as shown in Figure 4-14. The configuration must use the same encryption and hashing algorithms along with the same Diffie-Hellman groups and lifetime that were specified in the voice gateway.

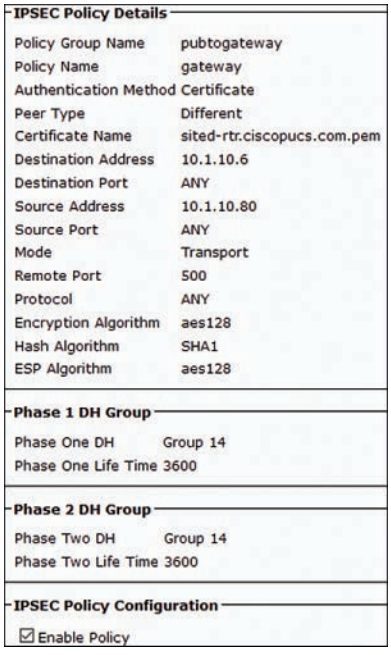


Figure 4-14 *IPsec Configuration for Voice Gateway*

The quickest way to verify that the configuration is successful is to perform a ping from the Cisco Unified CM with the validate IPsec is configured. Because a tunnel needs to be configured on each node within the cluster, a ping with validate IPsec should be completed from each node as well.

https://t.me/learningnets

@cisco_in_persian_channel

Securing the Integration with Cisco Emergency Responder

This section looks at securing Computer Telephony Integration (CTI)/Java Telephony Application Programming Interface (JTAPI). These protocols are used to provide real-time information on endpoints registered to Cisco Unified CM and perform call control functions. The specific integration that we discuss is the JTAPI integration with Cisco Emergency Responder to secure the integration previously discussed in Chapter 2, "Physical Security and Life Safety."

Cisco Unified CM enables you to secure the signaling and media streams between the Cisco CTIManager service and CTI/JTAPI/TAPI applications. This is accomplished through a mutually authenticated TLS handshake between the application (CER) and the CTIManager service. To authenticate with CER, the CTIManager service presents a certificate for the Cisco CallManager service. This certificate is added automatically to the CTL file when a Cisco Unified CM cluster is placed in mixed mode. Using mixed mode and enabling SRTP along with TLS are discussed further in Chapters 5 and 7. When CER attempts to connect to the CTIManager service, it downloads the CTL file from the Cisco Unified CM TFTP server that is configured. Because the CallManager certificate is included in the CTL file, CER trusts the connection. From the perspective of CER, it authenticates to the CTIManager service using a certificate issued by the Cisco Unified CM Certificate Authority Proxy Function (CAPF).

The CAPF service can provide the following functions for CTI/TAPI/TAPI applications, depending on the application and deployment:

- Authenticates the JTAPI/TSP client via an authentication string
- Issues locally significant certificates (LSC) to CTI/JTAPI/TAPI application users or end users
- Upgrades existing locally significant certificates
- Retrieves certificates for viewing and troubleshooting

When a JTAPI/TSP client interacts with CAPF to request a certificate or upgrade an existing certificate, the client authenticates to CAPF by using an authentication string that is configured for the user's CAPF profile. The client then generates its public and private key pair, forwarding its public key to the CAPF server in a signed message. The private key remains in the client and never gets exposed externally. The CAPF service signs the certificate (LSC) and then sends it back to the client in a signed message.

Certificates are issued to application users or end users by configuring the appropriate settings in either the Application User CAPF Profile Configuration window or End User CAPF Profile Configuration window.

The CAPF profile provides a method to issue locally significant certificates to secure users so that a TLS connection can be opened between Cisco Unified CM and CER.

https://t.me/learningnets

@cisco_in_persian_channel

For each instance of a service, you need to create a unique CAPF profile. Using CER as an example, a CER cluster consisting of two nodes, a Publisher and Subscriber, would require two profiles be created and assigned to the user configured for the integration.

Once the CAPF profiles are created and assigned, you need to add the user to additional access control groups. By default, the user that is used to integrate with CER needs two access control groups to connect without security. These groups are Standard CTI Enabled and Standard CTI Allow Calling Number Modification. To connect securely, this user needs an additional two access control groups assigned:

- **Standard CTI Secure Connection:** Allows a TLS-authenticated connection to be set up between the CTIManager service and CER
- **Standard CTI Allow Reception of SRTP Key Material:** Allows the use of SRTP to secure the media streams between CER and Cisco Unified CM

Note Authentication serves as the minimum requirement for encryption; that is, you cannot use encryption if you have not configured authentication.

Note Unified Communications Manager Assistant, Cisco QRT, and Cisco Web Dialer do not support encryption. CTI clients that connect to the CTIManager service may support encryption if the client sends voice packets.

Lastly, although Cisco Unified CM can support secure calls to and from CTI ports and route points, CER must be configured to support secure calls.

Enable Cisco Emergency Responder to Use Secure JTAPI

Before attempting to enable CER to communicate securely with Cisco Unified CM, you need to account for some prerequisites. These topics are discussed in more detail in later chapters and are provided here so that you are aware of them.

- **Cluster has an encryption license.** Starting with Cisco Unified CM 11.5(SU3), an encryption license became mandatory to enable SRTP/TLS within a Cisco Unified CM cluster. This requirement was backported to 10.5(2) and continues in the current 12.5 versions of the software where the Smart Account used needs to have Export Restrictions enabled.
- **Cluster is in mixed mode.** The Cisco Unified CM cluster must be in mixed mode for SRTP and TLS to be supported. You can verify the cluster status by logging in to Cisco Unified CM Administration, navigating to **System Enterprise Parameters**, and then verifying that **Cluster Security Mode = 1**.
- **CAPF service is running and has a valid certificate.** The CAPF service must be activated and started on the Publisher node.

https://t.me/learningnets

@cisco_in_persian_channel

- **Cluster is not running an unrestricted version of Cisco Unified CM.** The version of Cisco Unified CM cannot be the unrestricted version because this version has SRTP and TLS disabled.

To begin the process of securing the communication between Cisco Unified CM and CER, copy the existing CER application user and rename the copy **SecureCERUser** (see Figure 4-15). Then add the access control groups Standard CTI Secure Connection and Standard CTI Allow Reception of SRTP Key Material, as shown in Figure 4-16.

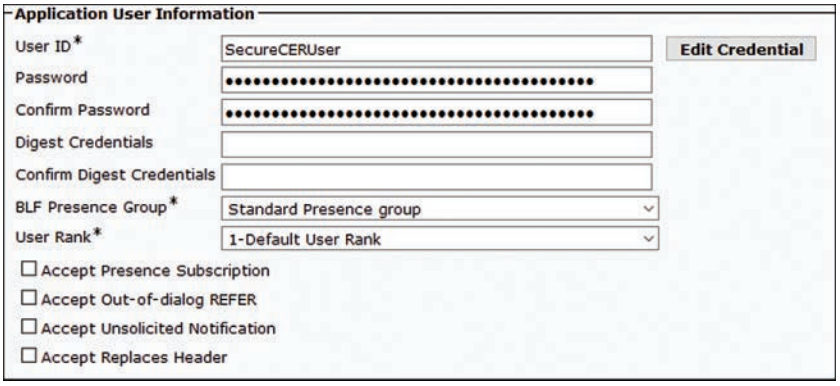


Figure 4-15 *CER Application User*

Figure 4-16 *CER Application User New Access Control Groups*

Next, create a CAPF profile for the user by navigating to **User Management > User Settings > Application User CAPF Profile** and selecting **Add New**. The help page for the CAPF Profile settings provides a good reference to what can be configured. In this instance, only a handful of information is needed, and Figure 4-17 provides a reference.

- **Application User:** From the drop-down list box, choose the application user (SecureCERUser) for the CAPF operation.
- **Instance ID:** Enter 1–128 alphanumeric characters (a–z, A–Z, 0–9). The Instance ID identifies the user for the certificate operation. This Instance ID is configured as SecureCER; make a note of the Instance ID for use later.

https://t.me/learningnets
@cisco_in_persian_channel

- **Certificate Operation:** Install/Upgrade installs a new or upgrades an existing LSC.
- **Authentication Mode:** The default By Authentication String is used.
- **Authentication String:** The Generate String button is used. Make a note of the authentication string for use later.
- **Key Order:** RSA Only is used.
- **RSA Key Size (Bits):** Select the default 2048.
- **Operation Completes by:** This needs to be a date in the future to specify the date at which the operation will stop.

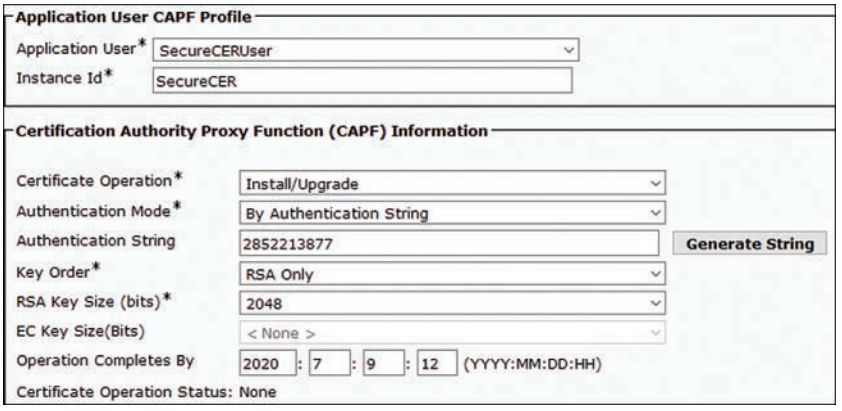


Figure 4-17 *Application User CAPF Profile*

Now you select **Save**. When the operation is complete, verify that it was successful. Navigate back to **User Management > Application User** and select **SecureCERUser**. The user should now have an associated CAPF Profile, as shown in Figure 4-18. The certificate operation status should still reflect "Operation Pending."

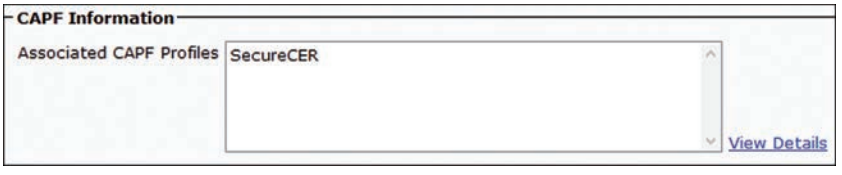


Figure 4-18 *Verify CAPF Profile Assigned to User*

Now you must configure the CER to support secure communication with Cisco Unified CM. Log in to Cisco Emergency Responder and update the Cisco Unified Communications Manager configuration (**Phone Tracking > Cisco Unified**

https://t.me/learningnets
@cisco_in_persian_channel

Communications Manager) to use a secure connection. You do this by selecting the **Enable Secure Communication** checkbox and then adding the following information:

- TFTP Service IP Address**
- TFTP Server Port**
- Backup TFTP Server IP Address
- CAPF Server IP Address**
- CAPF Server Port**
- Instance ID for Publisher**
- Secure Authentication String for Publisher**
- Instance ID for Subscriber** (if present)
- Secure Authentication String for Subscriber**

The updates to secure the communication between Unified CM and Emergency Responder are shown in Figure 4-19.

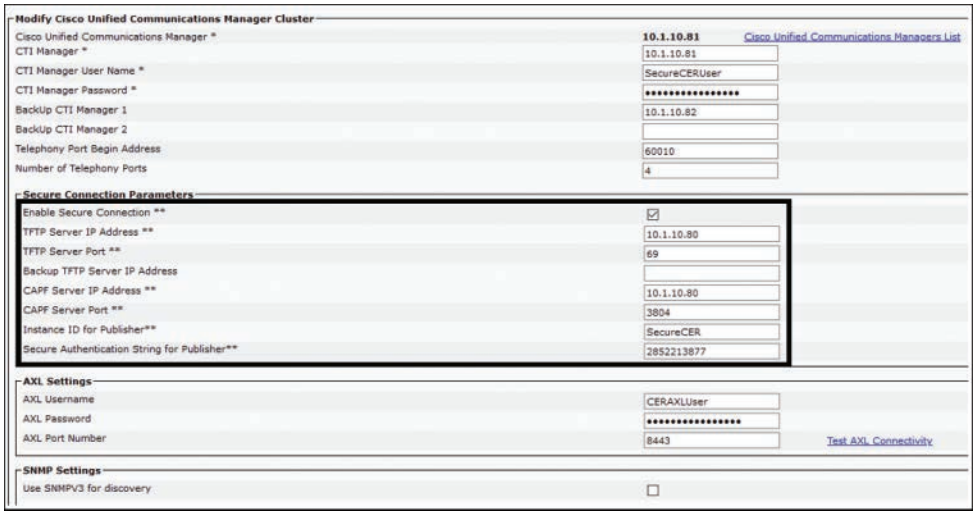


Figure 4-19 *CER Secure Connection Parameters*

Within CER, navigate to **Cisco ER Serviceability**, then **Tools > Control Center**, and restart the Cisco Emergency Responder service.

Note The CER service can take a few minutes to restart. If it is deployed with HA, restart the Cisco Emergency Responder service on the CER subscriber as well.

https://t.me/learningnets

@cisco_in_persian_channel

When this operation is complete, navigate back to Cisco Unified CM Administration and verify that the application user now has a valid CAPF certificate, as shown in Figure 4-20.

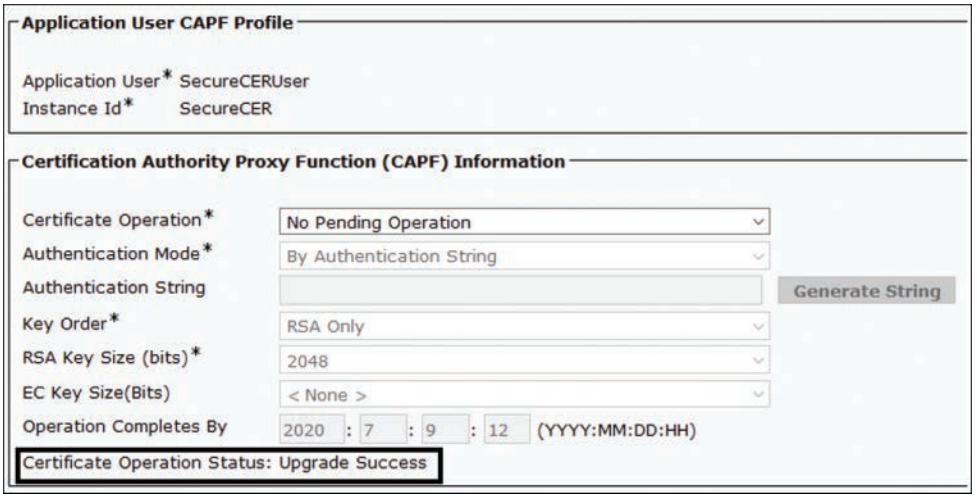


Figure 4-20 *Application User LSC Deployed*

The last step is to verify that the CTI route points and CTI ports have registered back.

Summary

This chapter covered several topics ranging from design considerations for the different deployment options to the process used to secure communication between Cisco Unified CM and CER. For the different deployment options, we described a few types of designs to simplify the process of identifying how resiliency and redundancy can be accounted for in the different models. You should refer to the relevant Solution Network Design Guide for a more detailed and nuanced understanding of the best practices associated with each deployment model. Next, we described the ancillary services and the impact they can have on the UC environment. We called out NTP and how to secure it because of its importance to the health of a Unified CM cluster and on authentication and encryption. We highlighted some of the potential attacks that could be leveraged against an unauthenticated time source and showed how to enable symmetric-keys for authentication. With an authenticated time source available, we covered how and why to use digital certificates and IPsec to secure ICCS between application nodes as well as endpoints and other UC applications. Lastly, we started to describe how Cisco Unified CM uses CAPF to deploy LSCs for authentication and encryption of signaling and media. This last part is continued in more depth in Chapter 7, where we discuss security by default along with the process to enable SRTP between the endpoints in a cluster.

https://t.me/learningnets
@cisco_in_persian_channel

Additional Resources

Manage IPsec Policies - https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/admin/12_5_1SU3/adminGd/cucm_b_administration-guide-1251su3/cucm_b_test-admin-guide_chapter_010001.html?bookSearch=true

Trunk and Gateway SIP Security - https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/security/12_5_1SU3/cucm_b_security_guide_1251SU3/cucm_m_trunk-and-gateway-sip-security_reog.html#topic_71459456E4AD33EFFB550A3B30B8120D

MGCP Configuration Guide, Cisco IOS Release 15M&T – Media and Signaling Authentication and Encryption - https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/voice/mgcp/configuration/15-mt/vm-15-mt-book/vm-gw-med-sig.html

Authentication and Encryption Setup for CTI, JTAPI, and TAPI – https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/security/12_5_1SU3/cucm_b_security_guide_1251SU3/cucm_b_security-guide-1251SU2_chapter_010101.html

Configure Cisco Emergency Responder - https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cer/12_5_1/english/administration/guide/cer0_b_cisco-emergency-responder-administration-guide-1251/cer0_b_cer-administration-online-help-1251_chapter_0100.html?bookSearch=true

Cisco Emergency Responder Port Usage - https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cer/12_5_1/english/administration/guide/cer0_b_cisco-emergency-responder-administration-guide-1251/cer0_b_cer-administration-online-help-1251_appendix_010101.html?bookSearch=true

https://t.me/learningnets

@cisco_in_persian_channel

Chapter 5

Hardening the Core Cisco UC Appliance Operating Systems

In this chapter we discuss the process to secure core Cisco UC Appliance operating systems and center it around three core areas:

- The differences between the UC Appliance OS and a standard Linux server
- Enabling Operating System Advanced Security Hardening Mechanisms
- Securing the OS command line and enabling command-line user access controls

We start the conversation with differentiating between the operating system used by the Cisco UC applications and a standard Linux server. This discussion covers how the UC application operating system is hardened by default, limiting numerous aspects like shell access, user access, and ability to install software when compared to a standard Linux server. Following that we explain the differences between restricted and unrestricted, and how the version deployed can affect the security available to a deployment. Next, we move to discussing several built-in OS advanced hardening mechanisms, like FIPS 140-2 and Common Criteria. In this discussion, we cover the requirements imposed on a system when these mechanisms are enabled. We end the chapter discussing how to harden command-line access through the use of user account controls to align with typical corporate user access control policies.

Over the weekend the ACME IT Security department performed a network scan and identified the core Cisco Unified Communications (UC) applications as Linux servers. Once the scan was complete, the team sent the senior IT manager some pointed questions surrounding the use of the servers and whether they were in compliance with ACME corporate IT security policies.

Shortly after the IT manager's second cup of morning coffee, he called in his trusted UC engineer, Anthony, to help answer some of the questions that the Security group asked.

https://t.me/learningnets

@cisco_in_persian_channel

With the findings from the IT Security department in hand, Anthony accepted the tasks to get the information requested by his manager as well as ensure that the applications also comply where applicable with the other corporate security policies.

Questions that you should ask:

1. Why should the UC servers be considered an appliance rather than a standard Linux server?
2. What is the process to verify the version of the UC application that is running, and is there a process to apply patches or upgrades to the system?
3. What level of access is available for system access? Is root or super user access available?
4. Do the UC applications conform to industry standard security certifications?
5. Are the corporate user access and password policies implemented on the UC applications?

Defining the Core UC Applications

When we're considering the multitude of Cisco UC applications, it is helpful to define what is a *core UC application* for the purposes of this book. Cisco has a lot of different UC applications on the market that cover audio and video call control, session and meeting management, along with instant messaging and presence status. This chapter defines the core UC applications as those applications that run on top of the voice operating system (VOS). The core UC applications discussed in this book run on top of the VOS. This platform was originally developed for Cisco Unified Communications Manager and has since been repurposed for many of the products within the Cisco UC portfolio; it is now more of an application-independent operating system than when it was originally developed. You may also see the underlying operating system referred to as Cisco Collaboration Communication OS (C3OS), but in this book we use VOS.

This chapter focuses on the Cisco Unified Communications Manager (UCM), Cisco Unity Connection (CUC), Cisco Instant Messaging and Presence (IM&P), and finally Cisco Emergency Responder (CER) as the core on-premises UC applications. Some of the other UC applications that run on VOS are Unified Contact Center Express (UCCX), Virtualized Voice Browser (VVB), and Cisco Finesse. While these latter applications are outside the scope of this book, the hardening steps we cover are typically similar.

These applications make up the foundation of a robust and feature-rich on-premises UC environment utilizing the Cisco UC portfolio.

- Cisco Unified CM provides audio and video device registration and tracking, along with call setup and teardown, including supplementary services like hold, resume, and park.
- Cisco Unity Connection provides access to voice and video mail and basic interactive voice response (IVR)/automatic call distribution (ACD) functionality.

https://t.me/learningnets

@cisco_in_persian_channel

- Cisco IM&P is the cornerstone of the environment for IM and presence tracking and calendaring integration.
- Cisco Emergency Responder provides real-time tracking of device locations based on switch port, subnet, and manual configuration that allows for voice and video endpoint location reporting for life safety calls.

The UC Appliance Is Not a Standard Linux Server

When ACME's UC engineer is asked why the UC appliances should not be considered Linux servers, what should the reply be? The short answer is Cisco Unified Communications appliances run the customized VOS and, starting with the Cisco Systems Release (CSR) 12, run a specialized and hardened version of CentOS. Versions of the applications running prior to CSR 12 were based on purpose-built custom versions of RedHat Enterprise Linux. In this section we explain why these UC applications should be considered appliances. However, with the applications running VOS built on different versions of Linux, it is not atypical for them to be identified incorrectly as a standard Linux server during a network audit.

With these points in mind, you need to understand some of the key differences between the Cisco voice operating system and the standard Linux operating system. To start this comparison, we must define the characteristics of a server that runs a standard version of Linux. While being a robust and stable operating system, Linux is not necessarily designed to be deployed out of the box as a secure OS. We're not saying Linux is not secure; we're just pointing out that additional work is required after the installation to secure the Linux server. For example, you have to ensure the latest updates have been applied, required services are started, and unneeded services are stopped or removed. Depending on the role of the server, numerous other tasks often need to be completed to have a "secure" installation.

With VOS, the underlying OS installation and hardening are already taken care of. Cisco's intention with the use of an appliance methodology for VOS is simple: provide a secure and hardened platform. The key takeaways that make VOS stand out from the standard Linux server are as follows:

- Reduced number of default installed packages and applications
- No root access
- No standard shell access
- No third-party applications or packages can be installed independently without approval from Cisco
- IP Tables enabled by default for firewall usage
- Security Enhanced Linux (SELinux) enabled by default for host protection

https://t.me/learningnets

@cisco_in_persian_channel

As would be expected, VOS is a customized version of Linux that is purpose built to support Cisco UC applications. Using this requirement, Cisco removed many of the nonessential packages and applications that come installed by default on a normal Linux server. Because the number of default installed packages and applications is reduced, the attack surface of the server is greatly reduced. Cisco has restricted shell access to VOS and uses a customized shell that allows for a restricted CLI environment. Additionally, two key aspects are removed from the system or restricted by the system. They are the ability, by default, to log in as root or another user considered to be a super user and the ability to install packages or other nonapproved applications directly on an appliance running VOS. We discuss the process of gaining root access later in this chapter.

Note Although you could manipulate VOS to allow root logins and then install nonstandard packages and applications, the system is specifically built to deter these practices. If you're working with the Cisco Technical Assistance Center (TAC) and the TAC team can identify additional root accounts or nonapproved packages and applications, you can expect the support provided to be minimal until the system has been restored to an approved baseline.

The ability to use various Linux shells is not available to the normal end user. When logging in with the standard OS administration account to gain command-line access, the user is presented with a predefined set of commands that can be run. These specific basic system commands apply directly to the application that is installed on top of VOS. In a nutshell, when logged in to the command-line interface (CLI) of an appliance running VOS, you do not have access to a shell-like Bash. The access made available is more akin to the CLI structure of a router.

By default, the installation of an appliance running VOS enables IP Tables as the firewall and SELinux for host-based intrusion protection. IP Tables are configured for network rate limiting and to allow only the necessary ports that are required for the application and/or features that run on top of VOS. You are not allowed to change or update the firewall settings without assistance from TAC in the case of a defect where a firewall setting was not configured properly, which is a rare scenario.

Note Typically, any updates made manually with or without the assistance of TAC do not survive an upgrade, so changing the default firewall settings of the system is not recommended.

If you run into a network audit, for example, where ports are reported as open on the appliance, you can use either the CLI or the Cisco Unified Communications OS Administration web page to view the status of the ports and whether or not they are allowed or translated.

Note Based on the application that is running on the appliance, the rules for traffic permitted or denied can vary.

https://t.me/learningnets

@cisco_in_persian_channel

Although you cannot change the firewall rules, if your security team needs to know what firewall rules are installed on the system, you can issue the **show network ipprefs [all|enabled|public]** command in the CLI to verify the status of the firewall rules, as shown in Figure 5-1.

```
admin:show network ipprefs all
Application     IPProtocol      PortValue       HashLimit (max:rate)            H-Status        ConnLimit       C-Status        Type            XlatedPort      Status          Description
sshd            tcp             22              1500:25/second                  enabled         -               disabled        public          -               enabled         sftp and ssh access
CiscoDRFMaster  tcp             4040            2000:35/second                  enabled         -               disabled        private         -               enabled         DRS Master Agent port
syslogd         udp             514             10000:35/second:all             enabled         -               disabled        public          -               disabled        syslog port
ntpd            udp             123             35:5/second                     enabled         -               disabled        public          -               enabled         network time sync
dhcp6c          udp             546             10000:35/second:all             enabled         -               disabled        public          -               disabled        DHCPv6 client
netdump         udp             6666            10000:35/second:all             enabled         -               disabled        public          -               disabled        this port needs to be open on systems
 running the netdump server
racoon          esp             -               -                               -               -               disabled        public          -               disabled        ipsec traffic
racoon          udp             500             10000:35/second:all             enabled         -               disabled        public          -               disabled        ipsec setup port
clm             udp             8500            25:5/second:all                 enabled         -               disabled        public          -               enabled         cluster manager
clm             tcp             8500            100:5/second                    enabled         -               disabled        public          -               enabled         cluster manager
cmoninit        tcp             1500            7500:50/second                  enabled         -               disabled        private         -               enabled         IDS port for DB clients
dbinpc          tcp             1515            2000:35/second                  enabled         -               disabled        private         -               enabled         DB replication port
dbmon           tcp             8001            2000:35/second                  enabled         -               disabled        private         -               enabled         DB change notification port
dbmon           tcp             2750            2000:35/second                  enabled         -               disabled        private         -               enabled         TLS change notification port
```

Figure 5-1 *Output for show network ipprefs*

Three output options are available for the **show** command:

- **All:** All defined ports
- **Enabled:** All enabled ports
- **Public:** All ports with remote connections allowed

You can find more information on the **show network ipprefs all** command in the latest Command-Line Interface Reference Guide for the application you're working with.

A second method to view the current IP Tables settings is through the Cisco Unified Communications OS Administration web page. Once logged in to the web page, navigate to **Show > IP Preferences**. From this page you can view the current port statuses, as shown in Figure 5-2.

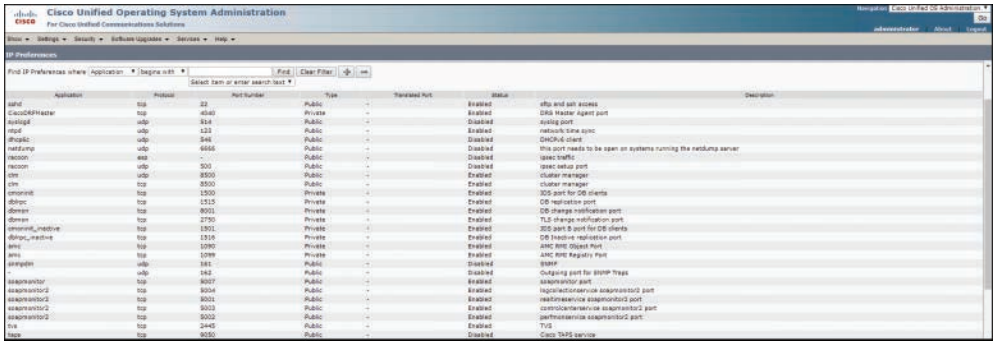


Figure 5-2 *GUI Output for show IP Preferences*

Because VOS has implemented SELinux, it is important to understand what SELinux does for the system. SELinux provides VOS with a mechanism in which to implement a mandatory access control (MAC) security protocol. Without the use of this security protocol, the system would have only discretionary access control (DAC) methods, which can be equated with file permissions and access control lists (ACLs). With DAC, an administrator

https://t.me/learningnets
@cisco_in_persian_channel

has no ability to enforce least privilege, and an end user can enable full control to sensitive files like SSH keys. When SELinux is implemented, VOS is able to implement a methodology of least privilege. This methodology is implemented at the kernel level, and the behavior of applications and users is inspected with policies enforced.

SELinux has three modes of operation on a typical Linux server: enforcing, permissive, and disabled. In the framework and shell provided to administrators with which Cisco has integrated SELinux, only the options for setting SELinx to enforcing (default) and permissive are available. Cisco advises administrators to set SELinux to permissive mode only when troubleshooting issues. The only way to implement the third setting for SELinux, disabled, is to enable the remote support account while working with TAC.

Recent enhancements to the implementation of SELinux allow you to apply dynamic rulesets when necessary. However, it is not advisable to modify the SELinux ruleset without consulting Cisco TAC. Let's look at a hypothetical example to examine the process. This example is based on a recent real-world scenario with Cisco UCM where the SELinux ruleset denied the installation of custom .wav files for Music on Hold (MoH) or Annunciator (CSCvo69002). This denial caused the upload to fail with an error stating, "The .wav file could not be translated by the Audio Translator Application."

Note If you decide to modify the ruleset on your own, you should be aware that you are potentially allowing harmful applications to run within the system.

After you identify a need to enable dynamic policies to update SELinux, adding the new policies to the system is a seven-step process. These seven steps are detailed in Figure 5-3.

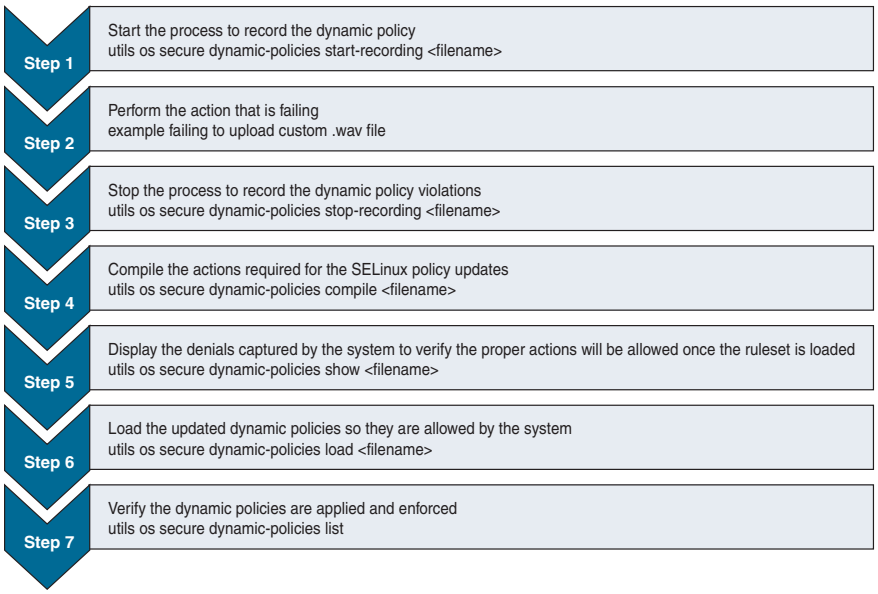


Figure 5-3 *Process Overview to Update SELinux Ruleset*

https://t.me/learningnets
@cisco_in_persian_channel

First, try to upload a customer MoH file to Cisco Unified CM. When you do, you receive the error shown in Figure 5-4.

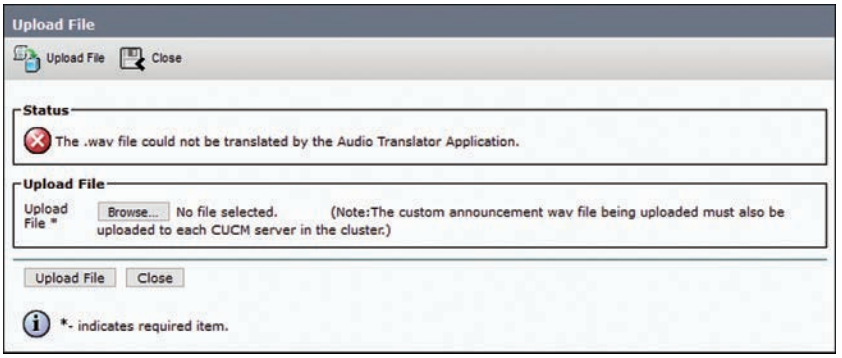


Figure 5-4 *MoH File Upload Failure*

The error reported confirms that you are hitting the defect called out.

Now do the following:

Step 1. Log in to the CLI of the Cisco Unified CM Primary node to begin recording the updated ruleset using the **utils os secure dynamic-policies start-recording <filename>** command, as shown in Example 5-1.

Example 5-1 *utils os secure dynamic-policies start-recording*

```
admin:utils os secure dynamic-policies start-recording audiotranslate
Starting to record SELinux denials under "audiotranslate"
Installing VMTools default version
This may take some time. Please don't press any button
1.Stop the recording once you see VMTools status as Running(Current).
2.If you see the vmtool status as Running(Out-of-date) try upgrading vmtools
  manually and stop the recording after you see status as Running (Current)
Recording has started for "audiotranslate"
################################################
# WARNING:                                     #
# Recording for too long can be hazardous to the #
# system's health and security. Your recording   #
# will be terminated after 1 hour. You are       #
# encouraged to terminate the recording sooner.  #
################################################
You can stop recording with:
   utils os secure dynamic-policies stop-recording audiotranslate
admin:
```

https://t.me/learningnets

@cisco_in_persian_channel

Step 2. With the recording running, repeat the action that failed previously. For this example, attempt to upload the MoH file again. Figure 5-5 shows the upload is now successful.

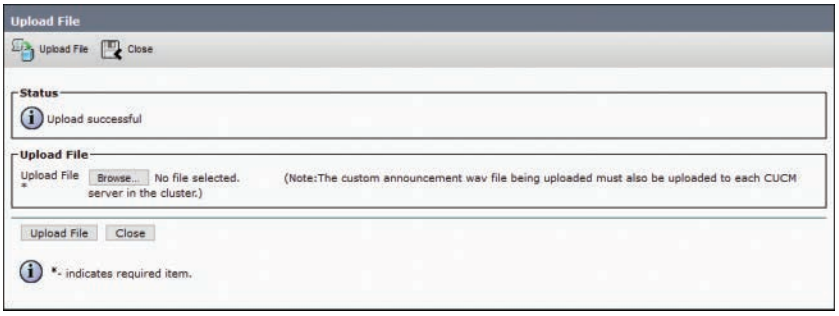


Figure 5-5 *MoH File Successfully Uploaded*

Step 3. Stop the recording by entering the **utils os secure dynamic-policies stop-recording <filename>** command.

```
admin:utils os secure dynamic-policies stop-recording
audiotranslate
Halting the SELinux denial recording for "audiotranslate"
Please wait. This may take several minutes.
New SELinux denials were caught under "audiotranslate"
Run the following to compile into a dynamic-policy:
   utils os secure dynamic-policies compile audiotranslate
admin:
```

Step 4. Compile the denials so that they can be added to the ruleset. The command to accomplish this is **utils os secure dynamic-policies compile <filename>**.

```
admin:utils os secure dynamic-policies compile audiotranslate
Compiling "audiotranslate" into a dynamic-policy
New dynamic-policy was compiled under "audiotranslate"
Review new dynamic-policy rules with:
   utils os secure dynamic-policies show audiotranslate
Reject new dynamic-policy rules with:
   utils os secure dynamic-policies remove audiotranslate
Accept new dynamic-policy rules with:
   utils os secure dynamic-policies load audiotranslate
admin:
```

Step 5. Verify the new rules that have been compiled by using the **utils os secure dynamic-policies show <filename>** command.

```
admin:utils os secure dynamic-policies show audiotranslate
Showing new rules for "audiotranslate"
```

https://t.me/learningnets
@cisco_in_persian_channel

```
Rules for audiotranslate:
allow tomcatd_t ld_so_cache_t:file execute;
allow tomcatd_t ld_so_cache_t:file execute;
Reject these rules with:
   utils os secure dynamic-policies remove audiotranslate
Accept these rules with:
   utils os secure dynamic-policies load audiotranslate
admin:
```

Step 6. Load the new rules into the system by using the **utils os secure dynamic-policies load <filename>** command, as shown in Example 5-2.

Example 5-2 *utils os secure dynamic-policies load <filename>*

```
admin:utils os secure dynamic-policies load audiotranslate
Loading dynamic-policy "audiotranslate" into SELinux
Dynamic-policy "audiotranslate" is being enforced.
################################################
# WARNING:                                     #
# Loading dynamic-policies into the system can #
# cause unwanted and insecure behavior. Should #
# problems begin to arise, consider removing   #
# this dynamic-policy.                         #
################################################
Remove this dynamic-policy from SELinux with:
   utils os secure dynamic-policies remove audiotranslate
admin:
```

Step 7. View the dynamic-policies loaded into the system by using the **utils os secure dynamic-policies list** command.

```
admin:utils os secure dynamic-policies list
Viewable dynamic-policies
  [POLICY_NAME] ->   [STATUS]
----------------------------------------
  audiotranslate ->   [Loaded]
admin:
```

To remove the dynamic-policies, use the **utils os secure dynamic-policies remove <filename>** command.

```
admin:utils os secure dynamic-policies remove audiotranslate
Purging audiotranslate
libsemanage.semanage_direct_remove_key: Removing last audio-
translate_vos module (no other audiotranslate_vos module
exists at another priority).
libsemanage.semanage_direct_remove_key: Removing last
```

https://t.me/learningnets

@cisco_in_persian_channel

```
audiotranslate_sys module (no other audiotranslate_sys module
exists at another priority).
Successfully removed dynamic-policy audiotranslate
admin:
```

A few key lingering components that are shared to varying degrees between a standard Linux server and Cisco VOS are password complexity requirements (which we discuss later in the chapter) and the use of SSH, HTTPS, and SFTP over their less secure counterparts as outlined in Table 5-1.

Table 5-1 *Secure versus Non-Secure Protocols*

Secure Protocol	Protocol	Use Case
Secure Shell (SSH)	Telnet	Command-line access
Secure File Transfer Protocol (SFTP)	File Transfer Protocol (FTP)	File upload and download
Hypertext Transfer Protocol Secure (HTTPS)	Hypertext Transfer Protocol (HTTP)	System Access via GUI

Restricted and Unrestricted Versions of UC Software

Cisco VOS runs on two different versions of software: Restricted and Unrestricted. The first version, Restricted, allows for strong encryption and functionality such as Secure Real-time Transport Protocol (SRTP), whereas the second version, Unrestricted, does not allow strong encryption or SRTP. While the naming can seem less than intuitive at times, the key concept is that the version that contains strong encryption is export restricted, so it follows that this version would be called Restricted.

The two versions of VOS implementations align with US export restrictions on encryption technology and also align with the import restrictions of other countries.

When you're considering an upgrade or install of one of the core UC appliances (Cisco UCM, Unity Connection, or Cisco IM&P), it is important to receive the proper version. While it is true that you can convert a restricted version to unrestricted, the opposite is not true. If you install an unrestricted version or upgrade from a restricted version to an unrestricted version, you cannot revert back. To convert from an unrestricted to a restricted version, typically requires you to perform a complete rebuild.

Disk images downloaded from the Cisco website are clearly identifiable; they have the UNRST identifier called out in the filename to indicate the difference between the unrestricted and restricted versions, as shown in Figure 5-6. The restricted version does not have an identifier called out in the filename.

https://t.me/learningnets

@cisco_in_persian_channel

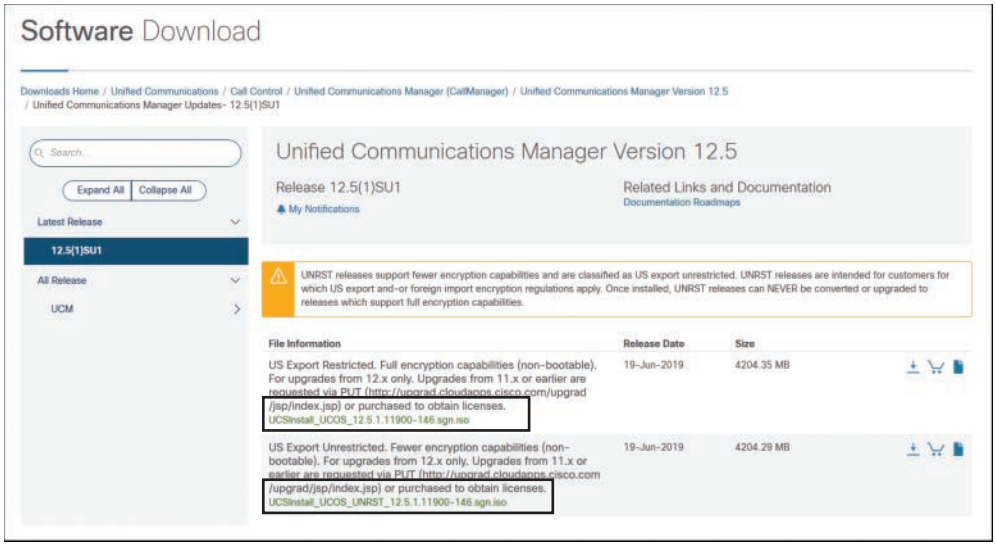


Figure 5-6 *Unrestricted Image Naming*

Post install/upgrade, you can use the initial web administration page for the UC application to identify whether or not the system is running the restricted or unrestricted versions, as shown in Figure 5-7.

Figure 5-7 *Unrestricted Naming Convention in GUI*

Note Just as with the disk images available on the Cisco software downloads website, only the Unrestricted version of the software is specifically called out, as you can see in Figure 5-7 and the following output. The restricted software versions just indicate the software version number.

From the CLI, you can identify the UC application as either unrestricted or restricted by using the **show version active** command.

```
admin:show version active
Active Master Version: 12.5.1.10000-22 Unrestricted
Active Version Installed Software Options:
No Installed Software Options Found.
admin:
```

https://t.me/learningnets

@cisco_in_persian_channel

Standard Practices for Patch/Version Management

All installation packages—whether they are the full core UC application installation files, phone firmware, or device packages—are signed by the Cisco manufacturing certificate authority (CA) to ensure the integrity and authenticity of the software. Updates, installs, and patches to the various systems cannot be applied unless they are signed by these CAs.

Using Cisco Unified CM version 12.5(1) SU1 / 12.5.1.11900-146 as an example, the following explains the versioning syntax.

Version Syntax: X.Y(Z.A-B)

- X incrementing represents a major release:
  - **Feature Release (major) e.g. 12.5(1) / 12.5.1.11900-146:** Complete install. New features, possibly new architecture, and bug fixes.
- Y incrementing represents a minor release:
  - **Feature Release (minor) e.g. 12.5(1) / 12.5.1.11900:** Complete install. New features and bug fixes.
- Z incrementing represents a maintenance release:
  - **Maintenance Release (MR) e.g. 12.5(1) / 12.5.1.11900:** Complete install. In addition to bug fixes, it may contain minor featurettes.
- A incrementing represents a service update, respin, or engineering special:
  - **Service Update (SU) e.g. 12.5(1)SU1 / 12.5.1.11900:** Update (patch) only. This contains bug fixes and possibly new minor features. Service update releases undergo more extensive regression testing than an engineering special because they are a cumulative release containing numerous engineering specials typically.
  - **Respin e.g. 11.5(1)SU3b:** Contains specific high-severity bug fixes that require release to manufacturing for subsequent new shipments.
  - **Engineering Special (ES) e.g. 12.5.1.12014-1:** Update (patch) only by default. Available only from TAC and contains only bug fixes. Engineering specials receive minimal testing.
- B incrementing represents a build number:
  - **Build e.g. 12.5.1.12900-115:** Indicates an error during compiling that resulted in another attempt to create the image.

There are several key takeaways when it comes to the version and version tracking of the different core UC applications.

First, versions matter:

- Stay current with the versions of applications you run. It is easy to get behind a few versions, which can make a simple minor upgrade turn into a painful major upgrade.

https://t.me/learningnets

@cisco_in_persian_channel

- You should always read the upgrade/install guides for the version you're upgrading to. These guides cover the necessary steps and the required processes to perform a basic upgrade or install. Read the fine print.
- Verify the versions of software when performing an install or upgrade to ensure they are compatible with the versions of the UC applications.
- It is recommended to wait to get bug fixes in a fully tested SU, MR, or feature release when possible.
- Because ESs are cumulative, they are only built on the top of each active Unified CM release train.
- When performing a restore, you must match the version of the system on which you're performing a restore with the version that is contained within the actual restore.

Second, RTM:

- Although it may not seem important, reading the relevant guides for the upgrade, install, or patch that is being applied is always necessary. The readme and other guides provide information on resolved and unresolved defects as well as any new features or behaviors that are introduced as part of the install.

How Root Access Is Granted

By design, Cisco VOS does not allow root access via the CLI or access to any of the typically available Linux shells like Bash. However, VOS does allow for the creation of a Remote Support Account (RSA) that can be used while working with TAC as a method to acquire a root shell. The RSA provides access to the underlying Linux system that VOS is built on. Enabling the RSA is a two-step process in which the local administrator enables the RSA and generates a passphrase. Cisco TAC then uses the passphrase to generate a password that can be used to log in to the VOS CLI with root access.

There are some caveats to understand when generating a Remote Support Account. First, when the account is enabled, specifying an expiry is mandatory. This expiry is the number of days before the account becomes disabled. Valid values for the life of the account are between 1 and 30 days.

Throughout the rest of this section, we walk you through the process to create a Remote Support Account.

Step 1. Use the **utils remote_account status** command to verify whether an RSA is created or not. This step is important because only one account can be enabled at a time.

```
admin:utils remote_account status
Remote Support
Status         : disabled
admin:
```

https://t.me/learningnets

@cisco_in_persian_channel

Step 2. If the RSA is not active, enable the account by using the **utils remote_account enable** command.

```
admin:utils remote_account enable
Successful in enabling RemoteSupport
admin:
```

Step 3. Create the account with the **utils remote_account create <account_name> <expiry>** command, as shown in Example 5-3.

Example 5-3 *utils remote_account create <account_name> <expiry>*

```
admin:utils remote_account create ciscotac 10
Account Successfully created
Account        : ciscotac
Passphrase     :
-----BEGIN PASSPHRASE-----
1vs+wWWTX2zlZ/+blLx1ppEQn6toSA7+37LDvnvtZzmjm7TbS4tWlxubTT0u5Ub0NLQEshOS6UCbZXb2yvWB
  PUV0uE5oOl0QhjvHnXP0uou9GwyLYfGI0V1Icap4h4jj7yqhNZzJrFQ8I2B7r2v0MKPNACoQSXK8fXerG/
  gAwirnNSnuYn3amJWpdIHVwPKRanH4errdrgeEe+el3LNiPnJII/poqWitG7jSqsdM9IrP53/kxy3CKtJX
  86RWBlG0cxAwBp9aNtWN6hbAgiMiKHC2NYrbMvgH9eH2RU/SkAFPgU/AS4zJgWT4R9jFBc6kKyAwAxwR
  bjcC/y4mJfOQSg==
-----END PASSPHRASE-----
Version        : 2
Expiry         : 12-3-2019:19:00:00 (MM-DD-YYYY:Hr:Min:Sec)
admin:
```

Lastly, the passphrase that is included in the output of the command—if it completes successfully—is provided to Cisco TAC. TAC uses this information to generate the decrypted password. Customers cannot decode this passphrase themselves, and TAC cannot log in to a system without the customer generating the passphrase.

With the account enabled and created, TAC is then able to continue troubleshooting to resolve the issue at hand.

Hardening the Voice Operating System

To various degrees, the core UC applications support the implementation of FIPS 140-2, Enhanced Security Mode, and Common Criteria. These different enhancements to the underlying OS that the UC appliance uses enforce different levels of security within the system. In this section we delve into the different aspects of each to help identify whether it makes sense within your organization to implement any of them. Keep in mind that, depending on the line of business or customers you work with, you may be required to implement some or all of these enhancements. As a UC engineer, you should work with your security group to identify what is and what is not required.

https://t.me/learningnets

@cisco_in_persian_channel

One reminder before beginning: when you're working with the CLI in VOS, most of the commands used in this chapter do not replicate to the other nodes in the cluster. This means if you are enabling FIPS on a three-node Cisco UCM cluster, you must run the **utils fips enable** command on all three nodes. If you execute the command on only one node, the other two nodes are not FIPS compliant.

Enabling Federal Information Processing Standard (FIPS) 140-2

When you're considering whether to implement the built-in, enhanced security postures, your effort starts with enabling FIPS 140-2. The Federal Information Processing Standard is a US and Canadian governmental certification that specifies the requirements that cryptographic modules must follow. FIPS 140-2, Security Requirements for Cryptographic Modules, was implemented on May 25, 2001, and immediately superseded FIPS 140-1, which established the Cryptographic Module Validation Program (CMVP), which validated cryptographic modules for FIPS. The CMVP is a collaborative effort between NIST and the Canadian Center for Cyber Security (CCCS). Not all versions of the UC applications are FIPS 140-2 compliant, so it is important to verify that compliance before attempting to implement it. The Cisco FIPS website (see the link at the end of the chapter) can be used to verify this compliance. If the application is not listed, the commands to enable it may be present and may appear to complete successfully when implemented; however, you should not take this status as being compliant. The reason is that third-party certification is completed on each application in order to gain FIPS 140-2 compliance. Without this certification, even if the system is running the proper modules, it is not compliant. For FIPS compliance, the software version must exactly match the associated Security Policy on the NIST website.

FIPS 140-2 is disabled by default and is not dependent on whether the unrestricted or restricted version of the software is enabled.

When FIPS 140-2 is enabled, it uses the following FIPS 140-2 level 1 cryptographic modules:

- CiscoSSL 1.0.2n.6.2.194 with FIPS Module CiscoSSL FOM 6_2_0
- CiscoJ 5.2.1
- RSA CryptoJ 6_2_3
- openssh 7.5.9
- libreswan
- NSS

The process to enable FIPS 140-2 is simple and straightforward, but with FIPS enabled, you must consider some caveats. When you enable FIPS, it regenerates various digital certificates. Table 5-2 provides a reference for the certificates that are regenerated on some of the core UC applications.

https://t.me/learningnets

@cisco_in_persian_channel

Table 5-2 *FIPS Enablement Certificate Regeneration*

Cisco Unified Communications Manager	Cisco Instant Messaging and Presence	Cisco Unity Connection
CallManager	Tomcat	Tomcat
Tomcat	IPsec	IPsec
IPsec	CUP	
TVS	Cup-xmpp	
CAPF	Cup-xmpp-s2s	
SSH		

Note The actual certificates regenerated vary based on the application where FIPS is enabled or disabled.

It is important to understand the impact that this regeneration has on the system. First, there is an impact to the system uptime. If you're working with a single-node deployment, you likely have to complete prerequisite tasks. From a financial perspective, if you're using third-party CA signed certificates from a public entity and not an internal CA, you need to obtain new certificates; those new certificates can cost money, depending on where you get them and what they are used for.

How is the system uptime impacted by the enablement of FIPS? When you enable FIPS, it forces a reboot of the node to load the FIPS-validated cryptographic modules. If devices are registered to this node, or it is processing calls, voicemail, or other activities, they are impacted during the reboot. Additionally, the implementation of FIPS forces a known good baseline for the cryptographic modules. If the modules do not pass self-tests, the system does not boot. In the event that these FIPS modules are corrupted and the system does not boot, you need to use the Recovery CD to fix the system so that it can boot properly.

Given the example of a single-node Cisco UCM deployment when FIPS is enabled, you need to do one or two things because the CallManager and TVS certificates are regenerated at the same time:

1. Configure the Enterprise Parameter – Prepare Cluster for Rollback to pre-8.0 setting to True. With the introduction of Security by Default (SBD), when an IP phone registers to Cisco UCM, it downloads an ITL file. This ITL file is essentially a trust list for the phone and is used to restrict which servers the IP phone can communicate with. If you set Enterprise Parameter to True, Cisco UCM serves an empty ITL file to the IP phones, which allows them to register any Cisco UCM. Security by Default is covered in more detail in Chapter 7, "Encrypting Media and Signaling," and covers the IP phone registration process.
2. Run the Certificate Trust List (CTL) client (or use the tokenless method) to deploy a CTL to client devices. The CTL is also discussed in Chapter 7.

https://t.me/learningnets

@cisco_in_persian_channel

If you fail to take prerequisites like this into account when implementing security features like FIPS, you risk having to touch every endpoint to restore service to them. Feature enhancements starting in Cisco Unified CM 12.5(1)SU1 work to mitigate some of these factors. Even taking in to account the new feature updates in newer versions of the software, you should still be prepared. With Cisco Unified CM 12.5(1) SU1, one of the feature enhancements is the way that the ITL and CTL files are signed. Prior to Cisco Unified CM version 12.5(1) SU1, the Cisco CallManager certificate signed the ITL file. Starting with version 12.5(1) SU1, the ITLRecoveryfile signs the ITL and CTL files. A more detailed explanation is provided in Chapter 7. At this time, the key takeaway is that this capability makes working with certificates easier.

One final impact on the system post-FIPS enablement is on what is allowed cryptographically. Two prime examples are IPsec and SNMPv3. After FIPS is enabled, IPsec requires CA signed certificates. You cannot use self-signed certificates. Also, certain DH group key values are not supported, such as DH group key values 1, 2, and 5. When configured on a FIPS-enabled system, SNMPv3 must be configured to use SHA as the authentication protocol and AES128 as the privacy protocol. Additionally, with FIPS enabled, some protocols are disabled by default, including DES and MD5, in favor of more secure protocols such as SHA and AES. You should refer to the relevant Cisco Security Guide during the planning phase for implementing FIPS to ensure ancillary applications and features are supported after enabling FIPS.

You can complete three options when working with FIPS and Cisco UC applications. They are **utils fips [enable|disable|status]**. These options are reasonably self-explanatory in that **enable** enables FIPS on the node where the application runs. The **disable** option disables FIPS on the node, and **status** returns the current system status of FIPS and self-tests run at startup for the node.

Using Cisco UCM as a reference, you enable FIPS in the following manner. From the CLI, execute the **utils fips enable** command, as shown in Examples 5-4 and 5-5.

Example 5-4 *utils fips enable*

```
admin:utils fips enable
Security Warning: The operation will regenerate certificates for1)
CallManager

2)Tomcat
3)IPsec
4)TVS
5)CAPF
6)SSH
Any third party CA signed certificates that have been uploaded for the above
components will need to be re-uploaded. If the system is operating in mixed
mode, then the CTL client needs to be run again to update the CTL file.
******************************************************************************
This will change the system to FIPS mode and will reboot.
******************************************************************************
Do you want to continue (yes/no)?
```

https://t.me/learningnets

@cisco_in_persian_channel

Select **Yes**.

Example 5-5 *utils fips enable Continued*

```
Generating certificates...Setting FIPS mode in operating system.
FIPS mode enabled successfully.
******************************************************
It is highly recommended that after your system restarts
that a system backup is performed.
******************************************************
The system will reboot in a few minutes.
```

After the command finishes running, the node reboots. When the process is complete, you can use the **status** attribute for the **utils fips** command to verify the current status of the node, as shown in Example 5-6.

Example 5-6 *utils fips status*

```
admin:utils fips status
The system is operating in FIPS mode. Self test status:

- S T A R T ---------------------
Executing FIPS self tests
runlevel is N 3
Start time: Thu Apr 28 15:59:24 PDT 2011
NSS self tests passed.
Kernel Crypto tests passed.
Operating System OpenSSL self tests passed.
Libreswan self tests passed.
OpenSSL self tests passed.
CryptoJ self tests passed...
```

As with most things completed at the command line for UC applications running VOS, you must enable FIPS on each node independently of the others. Because of this method of implementation, it is important to enable FIPS on only one server at a time and to wait for the application to come fully into service before proceeding to the next node.

One last note in regard to FIPS: if you're upgrading to a version of a UC application that is not FIPS compliant, you need to disable FIPS prior to attempting the upgrade. Failure to disable FIPS can cause the upgrade to fail due to the requirements that specific software modules be used. When you're upgrading to a non-FIPS-certified release, if FIPS is enabled, the necessary modules may not be present in a non-FIPS-certified release, which could cause the upgrade to fail.

https://t.me/learningnets

@cisco_in_persian_channel

Enabling Enhanced Security Mode

Enhanced Security Mode is enabled only on a system that runs with FIPS enabled. This mode of operation implements strict security and risk mitigation controls that help secure the system. Just like enabling FIPS was configured on a node-by-node basis, so is Enhanced Security Mode. In general, Enhanced Security Mode across the different UC applications provides the following:

- Stricter credential policy is implemented for user passwords and password changes.
- If the protocol for remote audit logging is set to TCP or UDP, the default protocol is changed to TCP. If the protocol for remote audit logging is set to Transport Layer Security (TLS), the default protocol remains TLS. If Common Criteria Mode is also enabled, strict host name verification is implemented. Hence, you are required to configure the server with a fully qualified domain name (FQDN) that matches the certificate.

Before enabling this mode of operation, you should account for the impact on day-to-day operations. With Enhanced Security Mode enabled, strict user account policies are enabled, and they could cause the need for repeated password recovery.

Table 5-3 provides some of the common settings surrounding user and password policies for the different core UC applications when Enhanced Security Mode is enabled.

Table 5-3 *Cisco Unified CM, Cisco IM&P, and Cisco Unity Connection Enhanced Security Password Settings*

Password Policy	Default Enhanced Security Mode Setting
Password Length (min – max)	14 to 127 characters
Password Complexity	1 uppercase, 1 lowercase, 1 special character
Stored Number of Previous Credentials	None of the 12 previous passwords can be used
Minimum Password Age (min – max)	Minimum of 1 day, maximum of 60 days
Password Character Difference	Minimum of at least 4 characters must differ between new passwords
Credential Expires (days)	180
Failed Login Attempts	3
Reset Failed Logon Attempts Every (minutes)	30

Across the core UC applications, you enable or disable Enhanced Security Mode and verify the status using the same subset of commands.

https://t.me/learningnets

@cisco_in_persian_channel

Note When Enhanced Security Mode is enabled, the node reboots. Do not proceed to the next node until the system is online and in full service.

Note You may be prompted to change the password for your OS administration user account after the node reboots.

You enable Enhanced Security Mode by using the **utils EnhancedSecurityMode enable** command, as shown in Example 5-7.

Example 5-7 *utils EnhancedSecurityMode enable*

```
admin:utils EnhancedSecurityMode enable
The system is operating in FIPS and NOT operating in Enhanced Security Mode mode
Warning : This operation will modify the password policies
1)Password Length should be between 14 to 127 characters.
2)Password should have at least 1 lowercase, 1 uppercase, 1 digit and 1 special
  character.
3)Any of the previous 24 passwords cannot be reused.
4)Minimum age of the password is 1 day and Maximum age of the password is 60 days.
5)Any newly generated password's character sequence will need to differ by at least
  4 characters from the old password's character sequence.
**********************************************************************
This will change the system to Enhanced Security Mode and will reboot.
**********************************************************************
Do you want to continue (yes/no) ? yes
Setting the remote syslog communication protocol to TCP.
The protocol for communication with remote syslog server is set to tcp.
Do you want to continue (yes/no) ? yes
contact search authentication mode enabled.
Setting password restrictions as part of Enhanced Security Mode enable
The system will reboot in a few minutes.
```

When verifying whether Enhanced Security Mode is enabled or disabled, you use the **utils EnhancedSecurityMode status** command.

```
admin:utils EnhancedSecurityMode status
The system is operating in Enhanced Security Mode.
admin:
```

Enabling Common Criteria ISO/IEC 15408 Compliance

Common Criteria (CC) is an international security certification standard (ISO/IEC 15408) for IT products and their security profiles. The certification ensures that the profiles perform to a high and consistent standard. Twenty-six countries around the world recognize

https://t.me/learningnets

@cisco_in_persian_channel

the Common Criteria certification provided by the Common Criteria Recognition Arrangement (CCRA). The purpose of the CCRA is to put forward those standards by encouraging a situation in which IT products and security profiles that earn a Common Criteria certificate can be procured or implemented without the need for further evaluation. Further, it aims to provide a baseline for confidence in the reliability of the judgments on which the original certificate was based by requiring that Common Criteria certificates should meet high and consistent standards. Within the US, the NSA-operated National Information Assurance Partnership (NIAP) Common Criteria Evaluation Validation Scheme (CCEVS) is the certifying authority. So, what does that mean? Simply put, the requirements for the Common Criteria build on FIPS 140-2, which is a joint US/Canada certification, and then enhances it to meet the information security requirements of the international community.

Cisco UCM, Cisco IM&P and Unity Connection are currently a few of the UC applications that support Common Criteria. The decision to enable CC should not be made lightly due to the impact on the security posture of the application it enforces. Specifically, when CC is enabled, two key factors are enforced. First, tls1.0 is disabled and cannot be used, so you must plan on using tls1.1 or tls1.2. Second, X.509 certificates are required for some interfaces.

You might wonder about the impact of disabling legacy tls1.0. The truth is that not everything supports the newer tls1.1 or tls1.2. Older Cisco IP phones such as the 79XX, 69XX, and 99XX models support only tls1.0. It is important to verify device and application compatibility with tls1.2 using the Cisco tls1.2 Compatibility Matrix for Cisco Collaboration Products. You can find the TLS 1.2 Compatibility Matrix for Cisco Collaboration Products at www.cisco.com/c/en/us/td/docs/voice_ip_comm/uc_system/unified/communications/system/Compatibility/TLS/TLS1-2-Compatibility-Matrix.html.

With the requirement to use X.509 certificates, you must observe specific caveats:

- The ExtKeyUsage extension within the peer certificate must contain one of the following OID values:
  - 1.3.6.1.5.5.7.3.1 = server authentication
  - 1.3.6.1.5.5.7.3.2 = client authentication
- Certificate key length cannot be less than 2048 bits (as per FIPS requirement).
- Strict host name verification is incorporated in CC mode, and configuration of IP address and SRV is not supported in CC mode for syslog and SIP configurations (in most cases).
- In addition to these, all the restrictions that are applicable to minimum tls features are applicable here also.

https://t.me/learningnets

@cisco_in_persian_channel

Because CC relies on FIPS 140-2 being enabled, the following scenarios are true when you look to enable or disable CC:

- When enabled on a non-FIPS-enabled server, FIPS mode is enabled along with the CC mode, which causes the certificates on the server to regenerate.
- When enabled on a FIPS-enabled server, certificate regeneration does not happen while enabling CC mode because FIPS is already enabled.
- If CC mode is disabled but FIPS mode is not disabled, it doesn't cause any certificate regeneration, but if FIPS mode is also disabled along with CC mode, the certificate is regenerated.

You can use the **utils fip_common_criteria [enable|disable|status]** command to enable, disable, and view the operational status of CC. With the proper argument, this command should be run on each node in the Cisco Unified CM and IM&P cluster if you decide that CC should be enabled.

To verify the current system status, connect to the CLI of the node where you are enabling CC and issue the **utils fips_common_criteria status** command:

```
admin:utils fips_common_criteria status
Server is not in Common Criteria mode.
```

After the **utils fips_common_criteria status** command has run successfully, you can then enable CC. This system is already FIPS enabled; otherwise, if the cluster does not have FIPS enabled, it is enabled as part of this process and the relevant x.509 certificates are regenerated.

You enable CC with the **utils fips_common_criteria enable** command:

```
admin:utils fips_common_criteria enable
************************************************************
  Warning : TLS 1.0 will be disabled in Common Criteria mode.
************************************************************
This will enable Common Criteria mode and system will reboot. Do you
want to continue? (yes/no) :
Common Criteria mode enabled successfully
Broadcast message from administrator@cucm-pub
        (unknown) at 17:58 ...
The system is going down for reboot NOW!
admin:
```

After the command completes, the system reboots. Do not proceed to the next server in the cluster until this node is back online and in full service.

The **utils fips_common_criteria disable** command can be used to disable CC.

https://t.me/learningnets

@cisco_in_persian_channel

Summarizing FIPS 140-2 / Enhanced Security Mode / Common Criteria

It is possible to build on the security baseline provided by VOS after the initial deployment of the UC appliance. Depending on several factors such as line of business, corporate IT security policies, and governmental regulations, you may need to enable one or all of the baseline security settings described in the preceding sections. It is important to remember that with each option you enable, there is an impact to the system in compatibility with the existing infrastructure as well as from an administrative point of view. Ensure that you read the documentation for each baseline security option before enabling so that you are able to account for the nuances that are added to the UC environment.

Performing OS Lockdown via CLI

In the following sections, we walk through the process to secure the OS to ensure user and password policy compliance and show how to perform account recovery if you become locked out of the CLI or forget the password. This section outlines the process to enable logging and configuring and enforcing user account control policies and how those policies can be applied at the CLI.

The following requirements are used to illustrate these configuration steps:

- System account passwords should expire after 60 days.
- System accounts should use complex passwords consisting of at least one each of the following: one uppercase, one lowercase, one digit, and one special character.
- A password history for system accounts should store the previous 10 passwords and not allow them to be reused.
- Trivial passwords are not allowed.
- The minimum length of a password for a system account is 10 characters.
- Accounts should automatically be locked out after five incorrect password attempts and unlock after 30 minutes.
- Inactive accounts should be disabled after 10 days once the password expires.

The objective for this section is not to dictate the exact settings that should be configured within any given UC environment but to provide insight into the settings that can be configured. Take a moment before beginning and ask yourself this question, "How many times in a day or week do I log in to the CLI of my UC applications?" It's likely that you don't log in very often, which is why there is a tendency to not enable basic user account settings. The reason is most likely that they are separate from the traditional Enterprise

https://t.me/learningnets

@cisco_in_persian_channel

LDAP and NAC systems. With VOS-based UC applications, the actual UC application (administration web page access) itself can be integrated with the enterprise environment. The CLI, however, is slightly limited. An important factor to remember is that most of the logging and user account settings configured at the CLI for VOS-based systems do not replicate to the rest of the nodes in the cluster. You typically have to update these settings on a node-by-node basis.

It's true that you could enable Enhanced Security Mode across the different UC applications to enable most of the user account settings in one fell swoop. Those settings are pretty aggressive, though, so you end up updating some of the settings to make them less aggressive in their posture. The upcoming sections walk through the manual process to perform what the Enhanced Security Mode mostly accomplishes in one command.

Before you enable any of these settings, however, you should work with your security group to find out what the different policies are so that you can make sure you are in line with the larger corporate policies.

Process to Change Passwords for OS/GUI/Database

You need to understand a couple of important objectives about the different account and passwords that can be configured and updated from the CLI. Essentially, with few exceptions, if something is configured at the command line within a VOS-based UC application, it typically does not replicate to the other nodes within the cluster. This point is important when you consider user account settings because having passwords that are different across a cluster is painful and can cause a failure.

These three main passwords can be reset from the CLI:

- **CLI administrator:** This account is created during the initial installation of the application and provides access to the CLI, OS Administration, and Disaster Recovery Service web pages.
- **Security passphrase:** The cluster database authentication passphrase is used to secure intra-cluster communication between nodes in a cluster at the platform level.
- **UI application password:** This account is used to sign in to the UC application web administration pages. This command varies by application, so you can find the appropriate command in the relevant Command-Line Reference Guide for the product you're working with.

The CLI administrator account is defined when the system is installed and cannot be renamed. If you lose the name for this account, the most expedient way to identify it is to perform a password recovery, and the name is provided as part of the recovery process.

If you are going to enable a stricter user account policy with password aging and account inactivity, creating a secondary CLI administrator account is recommended. The **set account enable <username>** command creates the new account for you. When creating this account, consider that the password aging for the account should be staggered

https://t.me/learningnets

@cisco_in_persian_channel

from that of the default CLI administrator account. If you do not stagger the aging, you run the chance of being locked out of both CLI accounts.

Example 5-8 provides a reference for creating a second CLI administrator account. Key concepts for the creation of CLI administrators are

- Privilege Levels:
  - Administrator with level 0 privilege: This administrator has read-only access privilege on the interface.
  - Administrator with level 1 privilege: This administrator has both read and write access privilege on the interface.
  - Administrator with level 4 privilege: This administrator has both read and write access privilege on the interface. This access level is assigned only to the OS administrator account created during install.

  Each command that is available at the command line is assigned a privilege level. If the CLI account privilege level is too low for the command, it is not available to the account. You can refer to the CLI commands and their privilege levels in the Command Line Interface Reference Guide for Cisco Unified Communications Solutions, Release 12.5(1).
- SSO Enablement: The CLI administrator accounts can be enabled for SSO using either sAMAccountName or UPN, thus leveraging the ability to authenticate logins against an IDP.

Example 5-8 *Creating a Second CLI Administrator Account*

```
admin:set account name bkupadmin
Privilege Levels are:
   Ordinary - Level 0
   Advanced - Level 1
Please enter the privilege level :1
Allow this User to login to SAML SSO-enabled system through Recovery URL ?
  (Yes / No) :yes
To authenticate a platform login for SSO, a Unique Identifier (UID) must be provided
  that identifies this user to LDAP (such as sAMAccountName or UPN).
        Please enter the appropriate LDAP Unique Identifier (UID) for this
  user:[bkupadmin]bkupadmin
      Please enter the password :*********
            re-enter to confirm :*********
Account successfully created. This user must login to the CLI and update the
  password before they can login to OS Administration.
admin:
```

https://t.me/learningnets

@cisco_in_persian_channel

When changing the CLI administrator account password, you need to log in with the account you are going to change and then run the **set password user admin** command. You are prompted for the current password followed by the new password. When changing the password using this method, you need to follow the password policies on the system. If you are changing the password proactively and enabling stricter password policies on the system, those new stricter policies should be followed.

```
admin:set password user admin
Please enter the old password: *********
   Please enter the new password: *********
Reenter new password to confirm: *********
Please wait...
Password updated successfully.
admin:
```

The second password to consider for the CLI is the cluster security passphrase. This is the password used by the communication between the nodes in the cluster to authenticate the communication between them. You are not able to change the settings on the password policy for this password like you can for the CLI account. While it is not a hard-and-fast rule, it is recommended that you change this password when you change the CLI administrator password or at least once every twelve months. Changing this password does require some extra steps. First, it must be changed on the primary node in the cluster, followed by a reboot of that node. Then the remaining nodes in the cluster can have the password updated, followed by a reboot of the node to complete the password change process. The additional step that you should take before and after changing the cluster security passphrase is to check database replication both before and after the change to ensure that replication is set up correctly. You use the **set password user security** command shown in Example 5-9 to change the cluster security passphrase.

Note The Disaster Recovery System is dependent on this security password. If you need to restore using a backup that uses an older security password, you are prompted to provide it.

Example 5-9 *Changing the Cluster Security Passphrase*

```
admin:set password user security
Please enter the old password: *********
   Please enter the new password: *********
Reenter new password to confirm: *********
WARNING:
The Disaster Recovery System is dependent on this security password you are
  attempting to change.
If you need to use any of the older backup archive to restore this system, you need
  to remember the
older security password. To avoid this scenario, we recommend you to conduct a DRS
  Backup of your
```

https://t.me/learningnets

@cisco_in_persian_channel

```
system/cluster immediately after this password change.
Please make sure that the security password on the publisher is changed first.
The security password needs to be the same on all cluster nodes,
or the publisher and subscriber(s) will not communicate.
After changing the security password on a cluster node, please restart that node.
Continue (y/n)?y
Please wait...
Password updated successfully.
admin:
```

The last password that we discuss before describing the process to enforce a stricter password policy is the UI administrator account. This account is created during the initial installation and used to manage the web GUI of the UC application, such as Cisco Unified Communications Manager Administration or Cisco Unity Connection Administration. This account can be renamed either through the GUI or the CLI for VOS-based UC applications. To keep relevant topics together, we describe the methods to update this password or change the actual user account name in Chapter 8, "Securing Cisco Unified Communications Manager (Cisco)," on securing the web administration interfaces for the UC applications.

Configuring Password Aging

If there is one setting that should be set, it is password aging. With this setting enabled, you are forced at some point to change the password for the CLI accounts. That's a good thing, right? Well, it is unless you're enabling this setting on a system that has been in place for a while and the password has not been changed since the system was set up. At that point, depending on the policies you have implemented, the account is usually disabled due to inactivity. A best practice to keep in mind when enabling user account settings is to update the password or account so that it adheres to the new settings prior to enabling the new user account settings.

To enable and set the time frame for the maximum password age at a system level, you use the **set password age [minimum|maximum] <days>** command. Using the security policy defined previously, set the maximum password age to 60 days. With this setting enabled, after 60 days, you will be prompted to change the password when you log in.

```
admin:set password age maximum 60
Successfully set the maximum password age to 60
NOTE! All CLI Admin account password max-age values have been changed
to the new setting.
If you need some CLI Admin accounts to have a different password
max-age value, you will need
to specifically change them with the "set password expiry user
maximum-age configure" command.
admin:
```

https://t.me/learningnets

@cisco_in_persian_channel

You can use the **show password age** command to view the minimum and maximum password age settings.

```
admin:show password age
Maximum Password Age is : 60 days
Minimum Password Age is : 1 days
admin:
```

If you need to change individual user accounts, you can use the **set password expiry user maximum-age configuration** command. Best practice, however, is to change the system administrator accounts and security passphrase every quarter or six months. The implemented time frame depends on your company's security policies.

Enabling Password Complexity

With VOS, you are able to enforce password requirements that force exceptionally complex passwords. However, you need to keep human nature first and foremost in your considerations when implementing password complexity requirements. Human nature is to take the path of least resistance with passwords: that means writing them down. Passwords written down, no matter how complex, are not good passwords. If you implement extremely complex password requirements, consider the use of a password vault to store the passwords. There are lots of password vaults on the market, both on-premises and in the cloud, but we do not describe them in depth in this book.

As we stated earlier, you could simply enable Enhanced Security Mode, but some of the settings would be stricter than what the policies define. So, instead in this section, we demonstrate a more manual process of enabling these policies through several individual commands. Following are the relevant settings we called out earlier:

- System accounts should use complex passwords consisting of at least one each of the following: one uppercase, one lowercase, one digit, and one special character.
- A password history for system accounts should store the previous 10 passwords and not allow them to be reused.
- Trivial passwords are not allowed.
- The minimum length of a password for a system account is 14 characters.

The first two policies can be enabled with one command: **set password complexity character enable <num-char>**. The **num-char** argument relates to the number of uppercase, lowercase, digit, and special characters that are required. The default for **num-char** is 1, and using this default is usually safe.

Once this command is enabled, it effectively runs a few other commands: **set password character difference**, **set password character max-repeat**, and **set password history**.

https://t.me/learningnets

@cisco_in_persian_channel

With password complexity character enabled, the following password policies are enabled:

- It must have at least the current **num-chars** setting of lowercase characters.
- It must have at least the current **num-chars** setting of uppercase characters.
- It must have at least the current **num-chars** setting of digit characters.
- It must have at least the current **num-chars** setting of special characters.
- You cannot use adjacent characters on the keyboard; for example, you cannot use *qwerty*.
- You cannot reuse any of the previous passwords that match the passwords retained by password history.
- By default, the admin user password can be changed only once in 24 hours.

Example 5-10 shows how to enable password complexity character with the default value of 1.

Example 5-10 *password complexity character enable*

```
admin:set password complexity character enable
Successfully configured password Character complexity
You will need to change the passwords of the existing CUCM OS accounts for the
  password
complexity rules to be effective. Passwords generated subsequently will need to
  have at least
1 lowercase, 1 uppercase, 1 digit and 1 special character(s). All adjacent
  characters on
the keyboard will not be accepted. Passwords need to be at least 14 characters
  long. Additionally any of the previous 24 passwords cannot be reused.
The linux OS user account password can be changed only once in a single day. If you
  attempt
to change a password for a particular account more than once in a single day, it
  will fail.
admin:
```

With a password of *C@pit4lization*, you can see that it meets the criteria we've looked at so far. It has at least one uppercase, lowercase, digit, and special character. In this case, it is 14 characters long. If you try to use it as a password, does it work?

```
admin:set password user admin
Please enter the old password: **************
   Please enter the new password: **************
Reenter new password to confirm: **************
```

https://t.me/learningnets

@cisco_in_persian_channel

```
Please wait...
Password updated successfully.
admin:
```

The answer is yes, it does! What happens if you use the **set password complexity character enable 4** command? Does a similar password with at least one each of the characters work? Let's enable it and try to change the password for the CLI administrator account. This time, use the password *N3utr@lization*.

```
admin:set password user admin
Please enter the old password: **************
   Please enter the new password: **************
Reenter new password to confirm: **************
Please wait...
BAD PASSWORD: is too simple
admin:
```

Nope! Why does it not allow you to change the password? In this case, you cannot change the password because you need to have a minimum of four each of the uppercase, lowercase, digit, and special characters. What does that mean for the password? Here, it means that at a minimum the password must be at least 16 characters long to meet the password complexity requirements specified. If the original password, *C@pit4lization*, is updated to meet the criteria specified, does it work? Let's try to use *C@PIt4liz@7i10N*!* as the password.

```
admin:set password user admin
Please enter the old password: **************
   Please enter the new password: ******************
Reenter new password to confirm: ******************
Please wait...
Password updated successfully.
admin:
```

Success, it works!

The next password policy to tackle is the minimum password length. By default, with password complexity enabled without any num-chars specified, the minimum password length is 14. The specified requirements are that the length should be 10 at a minimum. The **set password complexity minimum-length 10** command allows you to update the setting.

First, try to update the password to *P1zz@Hat* and the current minimum and maximum length settings because the new password doesn't meet the minimum length requirements:

```
admin:set password user admin
Please enter the old password: **************
   Please enter the new password: *********
```

https://t.me/learningnets

@cisco_in_persian_channel

```
Password length must be between 14 to 127 characters
   Please enter the new password:
Control-C pressed
admin:
```

Now update the setting to a minimum length of 10:

```
admin:set password complexity minimum-length 10
The system wide minimum password length has been changed to 10.
This change will not affect the password of any existing OS accounts,
unless password of such accounts is changed.
```

Verify the setting has been updated:

```
admin:show password complexity length
Minimum Password Length is : 10
admin:
```

If you try to change the password to something 10 characters long, does it work? What happens when you use *C!sc0pizz@* as the password?

```
admin:set password user admin
Please enter the old password: **************
   Please enter the new password: **********
Reenter new password to confirm: **********
Please wait...
Password updated successfully.
admin:
```

Success!

The security policy settings for the CLI administrator account have now been configured.

Activating Account Locking and Inactive Account Disablement

Now that secure passwords are enabled, you need to meet the rest of the criteria for the user account security policies. Specifically, you need to enable the accounts to lock out and/or expire as well as become disabled if they are not used routinely within a specific period of time.

In this case, the following requirements should be met:

- Accounts should automatically be locked out after five incorrect password attempts and should unlock after 30 minutes.
- Inactive accounts should be disabled 10 days after the password expires.

https://t.me/learningnets

@cisco_in_persian_channel

By default, account inactivity and account locking are not enabled. This means that even if your password expires for the CLI account, the account is always active. With account locking disabled, you can enter the password wrong as many times as you want without worry that the account will become disabled. Neither of those is advisable in a production environment. Accounts that are not used should become disabled; otherwise, they can become a security risk. The same is true with accounts that do not get locked out due to attempts to log in with an invalid password too many times. The latter just opens up the opportunity for brute-force attacks to gain CLI access.

What is the best way to configure the system to be compliant with security policies that the servers are not in compliance with currently? You can use a couple of commands to first enable the accounts to expire eventually and then to enable the accounts to lock after a certain number of invalid login attempts.

First, you should use the **set password inactivity enable 10** command to set the password to become disabled after the password has been expired for 10 days. The command set has two arguments you can use to either enable or disable password inactivity.

```
admin:set password inactivity enable 10
 WARNING:  Please make sure that the OS Admin account does not become
inactive prior to other Admin accounts.
Enabled password inactivity successfully
admin:
```

It is important to remember how often you do use or do not use the CLI in an application like Cisco UCM. In all likelihood, you don't use it all that often, so you should set a recurring reminder to change the password when it is about to expire. If you don't and the account becomes disabled, you have to perform password recovery to get back in.

If you need to verify whether password inactivity is enabled and what the threshold is set to, you can execute the **show password inactivity** command.

```
admin:show password inactivity
Password Inactivity : Enabled and is currently set to 10 days
admin:
```

With the requirement to disable inactive accounts met, the next policy to look at is account locking. By default, the CLI of UC applications does not have account locking enabled, and you can verify the current status by using the **show accountlocking** command.

```
admin:show accountlocking
Account Lockout is disabled
admin:
```

Now to enable account locking, execute the **set accountlocking enable** command.

```
admin:set accountlocking enable
Account lockout successfully enabled
```

https://t.me/learningnets

@cisco_in_persian_channel

```
WARNING! Login Account will get locked if three consecutive
unsuccessful login attempts are made.
You will need to wait for at least 5 mins before the account gets
unlocked.
admin:
```

To meet the security policy requirements, the account should be locked after five unsuccessful attempts to log in and remain locked for 30 minutes. The default counts of three unsuccessful attempts and 5 minutes for the timeout do not meet these requirements. You need to modify the configuration a little more.

To set the account locking, you simply use the **set accountlocking count 5** command.

```
admin:set accountlocking count 5
Login retry count for lockout successfully configured
admin:
```

Now to set the unlock timer, you use the **set accountlocking unlocktime <seconds>** command.

```
admin:set accountlocking unlocktime 1800
Account lockout successfully configured
admin:
```

After completing these steps, you need to ensure that the specified security requirements have been met. Next, we cover the steps to perform a password recovery in the event the accounts expire.

Account Recovery Procedures

If the CLI administrator account becomes locked or disabled, there are two options to consider for regaining access. If you can log in with a second CLI administrator account, you can use that account to unlock/reset the other CLI administrator account. Security policies vary by company, and if you are able to use a second CLI administrator account, great! Just keep in mind that the expiration dates for the administrator accounts should be staggered; otherwise, you will likely end up being locked out of both CLI accounts when you realize you cannot log in.

Note With versions of UC applications prior to 12.x that used Prime License Manager (PLM), when the CLI account expired or was changed, PLM displayed a helpful message across the integrated product instances to inform you that they were not in compliance. This message was helpful to remind you to change the password of the CLI administrator and to let you know that after you updated the password, you did not update the PLM with the new password. Now that Smart Licensing is in use, PLM will fall by the wayside. The many benefits of Smart Licensing are covered in Chapter 8.

https://t.me/learningnets

@cisco_in_persian_channel

If you have a second CLI administrator account, you can enable a disabled account due to inactivity by using the **set account enable** command. After you enable the account, and you attempt to log in, you are prompted to change the password for the user.

The second method to recover an account that is disabled is to perform a password recovery for the account, which is shown in Example 5-11. This method is slightly more involved in that you must log in via the console of the virtual machine within VMware ESXi and mount and then unmount an ISO file. The purpose of this approach is to simulate having physical access to the server. For this process, you must log in to vSphere and open the console of the virtual machine. When prompted to log in, use the username *pwrecovery* and the password *pwreset*. Using this method, you are able to reset the passwords for both the CLI administrator account that was created when the system was installed and the cluster security passphrase.

Note If you change the cluster security passphrase, you must change it on all the nodes within the cluster. Otherwise, replication will fail. Also, changing the cluster security passphrase requires a reboot, and it is a best practice to perform a fresh backup via the Disaster Recovery System after the cluster security passphrase is changed on a cluster.

When prompted, mount an ISO. It can be any ISO at all; it does not have to be specifically a Cisco image.

Example 5-11 *Password Recovery Initial Login*

```
      ************************************************************
      ************************************************************
      **                                                        **
      **   Welcome to the Platform password reset               **
      **   Admin and Security password reset is possible        **
      **                                                        **
      ************************************************************
      ************************************************************


**  VMWare Installation: 1 vCpu@2670 Hz, 1 Disk(s), Disk 1:80 GB , 6144MB RAM **

You will be required to remove, then insert any valid CD/DVD media through
  VMware-vSphere client
To begin you will need to remove any media from the VMWare client CD/DVD drive.
You may press Control-C at any time to abort.

NOTE: At any point when it requests insert a CD/DVD media it should be mounted
  through vSphere client for VMWare server

Please connect the CD/DVD Drive from VMWare client and press any key when ready.
```

https://t.me/learningnets

@cisco_in_persian_channel

When prompted, unmount the ISO file within the virtual machine and press any key to continue. Then with the ISO file not connected, the system verifies the status and prompts you to mount the ISO file again. Once it is verified, you are presented with a menu of options for the passwords you can reset, as shown in Example 5-12. You can reset the default CLI administrator (defined at initial install) or the cluster security passphrase, or you can simply quit.

Example 5-12 *Password Recovery Options Menu*

```
Thank you, you may now proceed with Platform password reset.

Enter a for admin password reset.
Enter s for security password reset.
Enter q to Quit.
```

From here, you can select the option to reset the admin password and follow the prompts that are provided. This method does perform password checks against the configured system policies to ensure they are compliant.

Remember that after you reset the password, you need to disconnect the ISO image. If you fail to disconnect the ISO image, you could cause a system outage if the boot order first tries to boot off the ISO file that you did *not* disconnect before your next reboot.

Summary

In this chapter we covered a lot of foundation information on the underlying framework that the core UC applications are built on. The chapter started with why the UC application should be considered an appliance rather than a typical Linux server. Then it covered the naming conventions used for the different UC application versioning, including how the different upgrade files and patches should be applied. In talking about how the systems are hardened, we covered how Cisco aligns to and complies with the different US and Canadian information security requirements (FIPS 140-2) up to and including international standards like Common Criteria. For the majority of the chapter, we discussed the common steps taken to secure the CLI of the core UC applications to enable secure system account policies that can comply with corporate IT security requirements.

Additional Resources

Security Guide for Cisco Unified Communications Manager, Release 12.5(1)SU3 - https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/security/12_5_1SU3/cucm_b_security_guide_1251SU3.html

CUCM 12.5 - SELinux prevents custom .wav file uploads - CSCvo69002 - https://bst.cloudapps.cisco.com/bugsearch/bug/CSCvo69002/?rfs=iqvred

https://t.me/learningnets

@cisco_in_persian_channel

Cisco FIPS 140 Compliance - www.cisco.com/c/en/us/solutions/industries/government/global-government-certifications/fips-140.html

Cryptographic Module Validation Program (CMVP) - https://csrc.nist.gov/projects/cryptographic-module-validation-program

Security Guide for Cisco Unified Communications Manager 12.5(1) SU1 - www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/security/12_5_1_SU1/cucm_b_security-guide-125SU1/cucm_b_security-guide-for-cisco-unified125SU1_chapter_011011.html

Security Guide for Cisco Unity Connection Release 11.x - www.cisco.com/c/en/us/td/docs/voice_ip_comm/connection/11x/security/b_11xcucsecx.html

TLS 1.2 Compatibility Matrix for Cisco Collaboration Products - www.cisco.com/c/en/us/td/docs/voice_ip_comm/uc_system/unified/communications/system/Compatibility/TLS/TLS1-2-Compatibility-Matrix.html

TLS 1.2 Configuration Overview Guide - www.cisco.com/c/en/us/td/docs/voice_ip_comm/uc_system/TLS/TLS-1-2-Configuration-Overview-Guide.html

https://t.me/learningnets

@cisco_in_persian_channel

Chapter 6

Core Cisco UC Application Lockdown

In Chapter 5, "Hardening the Core Cisco UC Appliance Operating Systems," we discussed the methods and leading practices to secure Cisco's voice operating system along with the core differences between VOS and the typical Linux application server. In this chapter we discuss the methods and procedures to secure and maintain the core Cisco UC application (GUIs). To accomplish this, the chapter covers how to enable User Access Control (UAC) for the application GUIs, enable logging, and then configure and monitor the Disaster Recovery System. The scenario we used in Chapter 5 is expanded to cover these topics.

As ACME's UC engineer, Anthony has secured various aspects of Cisco UC application command lines. Now, he has realized that he needs to secure the UC applications themselves, and after securing them, enable monitoring for them. After a quick meeting with his manager to get a better understanding of ACME's security requirements for UAC and monitoring, he was able to identify the following items that need to be addressed:

- When end users are manually configured, they should adhere to ACME's UAC security policies.
- Logging is enabled for the Network Management System to collect logs via Syslog and SNMPv3.
- Backups should be configured with failures generating alerts.

To start the process, the system administrator needs to identify how ACME is currently enabling administrators and other users' access to the UC applications.

Questions that you should ask:

1. What logging is currently implemented?
2. Are logs being collected via Syslog or SNMP?
3. What is the company's disaster recovery strategy? Does it even have one?

https://t.me/learningnets

@cisco_in_persian_channel

Types of Users in Cisco Unified Communications Manager and Cisco Unity Connection

For the purpose of this book, we cover how the different Unified Communications applications create user accounts at a high level. The goal is to show how to secure these accounts when created locally within Cisco Unified Communications Manager and Unity Connection in addition to when they are imported from an external directory. Before you secure the users, however, it is important to understand the different types of users available and the ways that users can be added to the systems. With Cisco Unified CM as an example, two types of accounts are available: application users and end users.

Application users are associated with Cisco IP Communications features or services. These users usually do not require an interactive login. Several application users are created as part of the Cisco Unified CM installation process that are used either for administration or by the different applications within Cisco Unified CM. The application UI administrator account is included in these accounts. This application user account is used to access the Cisco Unified CM Administration, Cisco Unified Serviceability, and Cisco Unified Reporting GUIs without having to sign on multiple times. It's worth noting that the application administrator account does not have rights to log in to Cisco OS Administration (GUI/CLI interfaces) or Cisco Disaster Recovery Service (DRS) web pages.

Table 6-1 lists the default application user accounts created during installation. These accounts cannot be renamed or deleted.

Table 6-1 *Default Application Users*

Application User Account Name	Purpose
CCMAdministrator	Unified CM Administration (default "super user"). This account is created during installation, where the account name and password can be specified.
CCMSysUser	Cisco Extension Mobility
CCMQRTSecureSysUser	Cisco Quality Reporting Tool
CCMQRTSysUser	
IPMASysUser	Cisco Unified Communications Manager Assistant
WDSecureSysUser	Cisco WebDialer
WDSysUser	
TabSyncSysUser	Used to synchronize the IP Phone Personal Address Book
CUCService	SOAP—CDR on Demand

Application users should be used for system integrations with third-party products or other IP communications accounts. Examples of third-party integrations include using UnifiedFX phone view to provide remote phone control for a help desk and Cisco Unity Connection phone view to provide a visual representation of a voicemail box's TUI menu

https://t.me/learningnets

@cisco_in_persian_channel

on a Cisco IP phone. Application users for administrator accounts should be limited and tightly controlled because application user accounts are always local to the system.

End users are the second type of account that can be created within Cisco Unified CM. End-user accounts are assigned to a specific person with an interactive login. End users differ from application users in that no default end users are created when the system is installed. These users can either be local to the system or imported from an LDAP directory. End-user accounts can perform the same types of roles and functions as application users. The same cannot be said for application users; several aspects of end-user accounts are not available to application users. One such aspect is the ability to apply a UC profile to an account. Because you have the ability to import and then sync with an LDAP directory, importing end users allows you to offload the UAC functionality from Cisco Unified CM to LDAP. We cover integrating the core Cisco UC applications with LDAP in the following sections.

For Cisco Unity Connection and Cisco Emergency Responder user accounts, these systems do not use the same concept of application users and end users for integrating third-party systems. For these systems, it is easiest to consider that they use an account like the end-user account type within Cisco Unified CM.

Now that we've covered the fundamentals for the different types of user accounts, let's look at the process of securing them within Cisco UC applications to strengthen local account controls.

Strengthening Local User Account Controls

Out of the box, only minimal UAC elements are enforced on the core Cisco UC applications. You need to configure and then apply the policies to the appropriate users for them to be enabled and enforced. The standard process to apply a UAC to user accounts within Cisco UC applications involves three steps, as shown in Figure 6-1.

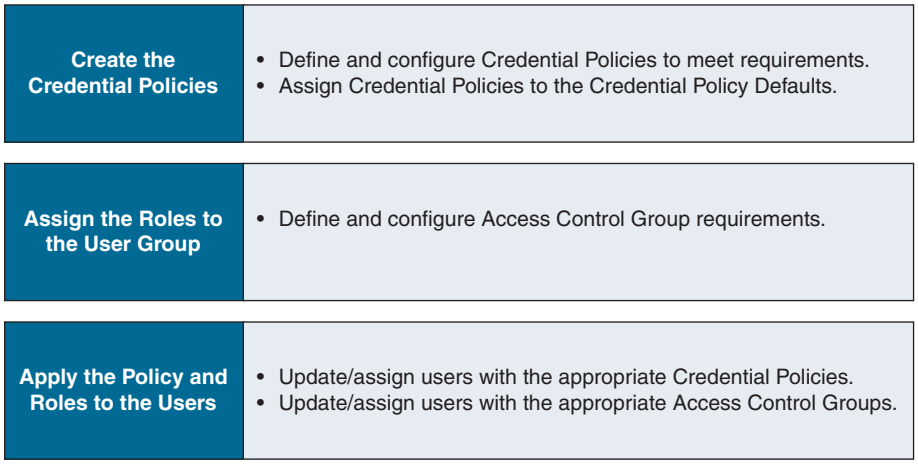


Figure 6-1 *UAC Order of Operations*

https://t.me/learningnets

@cisco_in_persian_channel

Referring to the requirements that ACME's system administrator was given, while gathering the requirements for the User Account Control policies, the security group told the UC engineer that the user account security policies should mirror those of the VOS system accounts.

- User account passwords should expire after 60 days.
- User accounts should use complex passwords consisting of at least one each of the following: one uppercase, one lowercase, one digit, and one special character.
- A password history for system accounts should store the previous 10 passwords and not allow them to be reused.
- Trivial passwords are not allowed.
- The minimum length of a password for an end-user account is 10 characters.
- The minimum length of a PIN for an end-user account is 10 characters.
- The minimum length of a password for an application user account is 14 characters.
- Accounts should automatically be locked out after five incorrect password attempts and unlocked after 30 minutes.
- Inactive accounts should be disabled 10 days after the password expires.
- Default service accounts are granted an exception to the standard UAC policies, but additional service accounts used for third-party integrations should adhere to the corporate security policies.

ACME has made an exception for the default application user accounts (service accounts) that were created during installation. These accounts do not need to have their passwords changed and should be set to never expire. This exception was given to these service accounts because they are not to be used for day-to-day operations and maintenance.

Creating User Account Control Policies on Unified CM and Unity Connection

The process to secure user accounts in Cisco Unified CM starts with defining the credential policies that will be used. Three credential policies are created. They are based on the Default Credential Policy and apply, one each, to the password settings for application and end users with a last credential policy created for the PIN settings for end users.

One item to keep in mind when creating credential policies is that these policies apply only to local accounts. Users who are synched with LDAP and use LDAP authentication do not utilize the credential policy applied to their accounts for the password. An important caveat to understand is that users who are enabled for LDAP authentication do not authenticate their PINs to LDAP. PINs, regardless of the user type and authentication

https://t.me/learningnets

@cisco_in_persian_channel

method, are always authenticated against the UC application. End users use the password policies applied to their accounts from LDAP. If LDAP authentication is not used, however, and the account is simply synched or imported from LDAP, the password credential policy is applied. A later section covers the intricacies of securing LDAP integrations for Cisco applications in more depth.

A base installation of Cisco Unified CM has two credential policies configured: the Default Credential Policy (see Table 6-2) and Enhanced Security Credential Policy (see Table 6-3).

Versions of Cisco Unified CM prior to 12.0 came with the Default Credential Policy configured only with a base installation. Although the Default Credential Policy provides fundamental UAC capabilities. The addition of the Enhanced Security Credential Policy in Cisco Unified CM 12.0 allows a UC administrator to provide more robust UAC from the initial deployment.

Additionally, the login credentials for local users starting with version 12.5(1) are hashed using SHA2. Prior versions of Cisco Unified CM used SHA1 to hash these credentials. Those end-user credentials that are hashed with SHA1 automatically have their hashes updated when they perform the following functions:

- Logging in to Extension Mobility or Directory Access on the phone updates the PIN.
- Logging in to Cisco Jabber, Cisco Unified Communications Self Care Portal, or Cisco Unified CM Administration updates the password.

In Cisco Unified CM 12.5(1), the Cisco Unified Reporting web page has been updated with an additional report titled "Unified CM Users with the Out-of-Date Credential Algorithm." This report helps identify those users who have PINs and passwords that are hashed with SHA1.

To create the credential policies from Cisco Unified CM Administration, you navigate to **User Management > User Settings > Credential Policy**. Then you select the option to add a new credential policy. From there you have the options to enable various UAC settings. The first policy to be created is named **Standard End User - Password**. This policy is applied to the local end users within the Cisco Unified CM application. The settings for the password UAC are set according to Figure 6-2.

Table 6-2 *Default Credential Policy*

Credential Policy Information	
Display Name*	Default Credential Policy
Failed Logon*	5
Reset Failed Login Attempts Every (minutes)*	30
No Limit for Failed Logons	Not Checked
Lockout Duration (minutes)*	30
Administrator Must Unlock	Not Checked

https://t.me/learningnets

@cisco_in_persian_channel

Credential Policy Information	
Minimum Duration Between Credential Changes (minutes)	0
Credential Expires After (days)*	0
Never Expires	Checked
Minimum Credential Length*	1
Stored Number of Previous Credentials*	0
Minimum Number of Character Changes Between Successive Credentials*	1
Inactive Days Allowed	0
Expiry Warning Days*	0
Check for Trivial Passwords	Checked

Table 6-3 *Enhanced Security Credential Policy*

Credential Policy Information	
Display Name*	Enhanced Security Credential Policy
Failed Logon*	3
Reset Failed Login Attempts Every (minutes)*	15
No Limit for Failed Logons	Not Checked
Lockout Duration (minutes)*	0
Administrator Must Unlock	Checked
Minimum Duration Between Credential Changes (minutes)	1440
Credential Expires After (days)*	60
Never Expires	Not Checked
Minimum Credential Length*	12
Stored Number of Previous Credentials*	12
Minimum Number of Character Changes Between Successive Credentials*	1
Inactive Days Allowed	0
Expiry Warning Days*	0
Check for Trivial Passwords	Checked

https://t.me/learningnets

@cisco_in_persian_channel

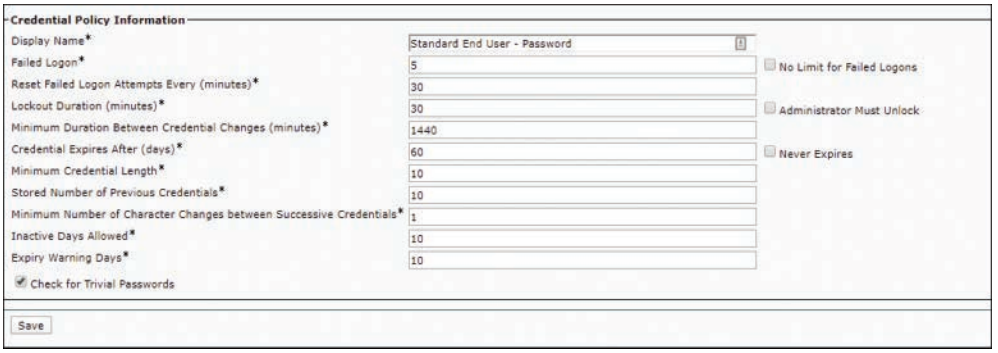


Figure 6-2 *Standard End User - Password Credential Policy*

As you can see in Figure 6-2, when creating a credential policy, you, as UC administrator, can fine-tune the access controls for the users. To meet the corporate security policies provided, you need to update several settings. These settings are explained in greater detail in Table 6-4.

Table 6-4 *Credential Policy Settings*

Setting	Use Case
Failed Login*	Specify the number of allowed failed login attempts. When this threshold is reached, the system locks the account. Enter a number in the range 1–100. To allow unlimited failed logins, enter 0 or check the No Limit for Failed Logons checkbox. Uncheck the checkbox to enter a value greater than 0. The default setting specifies 5.
Reset Failed Login Attempts Every (minutes)*	Specify the number of minutes before the counter is reset for failed login attempts. After the counter resets, the user can try logging in again. Enter a number in the range 1–120. The default setting specifies 30.
Lockout Duration (minutes)*	Specify the number of minutes an account remains locked when the number of failed login attempts exceeds the specified threshold. Enter a number in the range 1–1440. Enter 0 or check the Administrator Must Unlock checkbox so that accounts will remain locked until an administrator manually unlocks them. Uncheck the checkbox to enter a value greater than 0. The default setting specifies 30.
Minimum Duration Between Credential Changes (minutes)*	Specify the number of minutes that are required before a user can change credentials again. Enter 0 to allow a user to change credentials at any time. Uncheck the checkbox to enter a value greater than 0. The default setting specifies 0.

https://t.me/learningnets

@cisco_in_persian_channel

Setting	Use Case
Credential Expires After (days)*	Specify the number of days before a credential will expire. Enter a number in the range 1–365. To allow credentials to never expire, enter 0 or check the Never Expires checkbox. Uncheck the checkbox to enter a value greater than 0. Use the 0 option for low-security accounts or multiple user accounts, for example. The default setting specifies 180.
Minimum Credential Length*	Specify the minimum length for user credentials (password or PIN). Do not enter 0 because blank passwords are not allowed. The default setting specifies 8. The minimum setting must equal at least 1.
Stored Number of Previous Credentials*	Specify the number of previous user credentials to store. This setting prevents a user from configuring a recently used credential that is saved in the user list. Enter a number in the range 0–25. If no previous credentials should be stored, enter 0. The default setting specifies 12.
Inactive Days Allowed*	Specify the number of days that a password can remain inactive before the account gets locked. Enter a number in the range 0–5000. The default setting specifies 0.
Check for Trivial Passwords	Check this checkbox to require the system to disallow credentials that are easily hacked, such as common words and repeated character patterns. Nontrivial passwords meet the following requirements: ■ Must contain three of the following four characteristics: uppercase character, lowercase character, number, or symbol. ■ Must not use a character or number more than three times consecutively. ■ Must not repeat or include the alias, username, or extension. ■ Cannot consist of consecutive characters or numbers. For example, passwords such as 654321 or ABCDEFG are not allowed. PINs can contain digits (0–9) only. A nontrivial PIN meets the following criteria: ■ Must not use the same number more than two times consecutively. ■ Must not repeat or include the user extension, mailbox, or the reverse of the user extension or mailbox. ■ Must contain three different numbers. For example, a PIN such as 121212 is trivial.

https://t.me/learningnets

@cisco_in_persian_channel

Setting	Use Case
Check for Trivial Passwords	■ Must not match the numeric representation (i.e., dial by name) for the first or last name of the user. ■ Must not contain groups of repeated digits, such as 408408, or patterns that are dialed in a straight line on a keypad, such as 2580, 159, or 753. Cisco Unity Connection PIN has the following attributes: ■ The PIN cannot match the numeric representation of the first or last name of the user. ■ The PIN cannot contain the primary extension or alternate extensions of the user. ■ The PIN cannot contain the reverse of the primary extension or alternate extensions of the user. ■ The PIN cannot contain groups of repeated digits, such as 408408 or 123123. ■ The PIN cannot contain only two different digits, such as 121212. ■ A digit cannot be used more than two times consecutively (e.g., 28883). ■ The PIN cannot be an ascending or descending group of digits (e.g., 012345 or 987654). ■ The PIN cannot contain a group of numbers that are dialed in a diagonal, vertical, or horizontal straight line on the phone keypad (e.g., the user cannot use 159, 159730, 147, 147365, 123, or 123597 as a PIN.

The process that was used to create the Standard End User - Password credential policy is used to create the policies for Standard End User - Pin and Standard Application Users - Password. A key aspect to remember is that end users are considered to use an interactive login. Because of this interactive login, you should create policies to account for both passwords and PIN entries.

After the credential policies are created, they need to be associated with specific types of credentialed users. While you are logged in to Cisco Unified CM Administration GUI, navigate to **User Management > User Settings > Credential Policy Defaults**. Once there, you see the three Credential Policy Default types—two for end users and one for application users—as listed in Table 6-5.

https://t.me/learningnets

@cisco_in_persian_channel

Table 6-5 *Credential Policy Defaults*

Credential Policy Default	Credential User	Credential Type
Default Credential Policy	End User	Password
Default Credential Policy	Application User	Password
Default Credential Policy	End User	Pin

Cisco Unified CM applies a default credential policy to both end users and application user credentials. This setting is called Credential Policy Defaults. You need to update these settings when you do not use the credential policy named Default Credential Policy. The initial configuration for all the Credential Policy Defaults is set to use the Default Credential Policy. To apply the new Credential Policies that were created, you simply select the first Credential Policy Default and update it with the appropriate credential policy. Figure 6-3 uses the End User - Password credential policy default as an example for updating the credential policy.

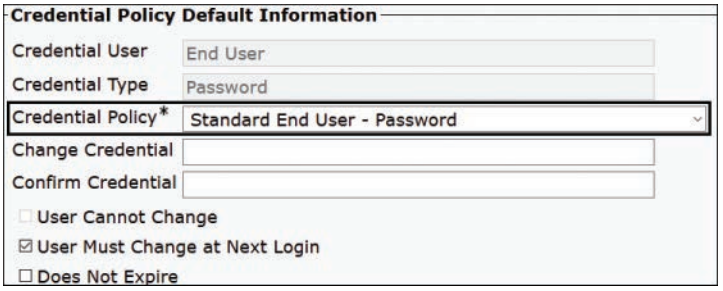


Figure 6-3 *Applying Credential Policy*

The remaining two Credential Policy Default types should be updated according to Table 6-6 to apply the appropriate UAC settings.

Table 6-6 *Credential Policy Defaults Updates*

Credential Policy Default	Applied Credential Policy
End User – Pin	Standard End User - PIN
Application User – Password	Standard Application User - Password

Using and Working with Cisco Unified CM Access Control Groups

After the credential policies are created and applied to the credential policy defaults, the next task is to identify the roles the users need. Typically, these roles fall into two categories: administrators and standard user accounts. Administrator accounts are system administrators whose role is to manage the system. The roles that these accounts need within the application are decidedly different from those of the standard user. It

https://t.me/learningnets
@cisco_in_persian_channel

is a best practice to not use administrator accounts for end-user roles because of the elevated permissions these accounts have within the system. Standard user accounts are for the day-to-day user (not administrator) interaction with the UC application and physical endpoints (IP phones, video endpoints) and soft clients (Cisco Jabber, Webex Teams) among others. Cisco Unified CM provides several different roles that allow for the least privileges to be assigned while still being able to perform the expected duties of the role. In fact, additional custom access control groups can be created and tailored to specific roles within the environment—for example, call recording and CTI control of devices. Figure 6-4 illustrates the differences between administrative and end-user accounts.

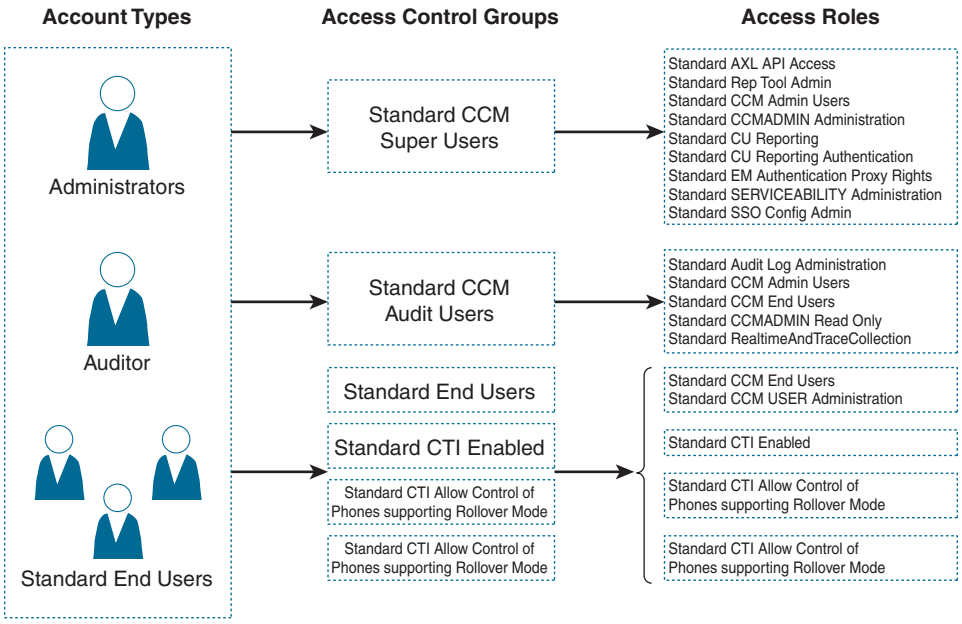


Figure 6-4 *Role-Based Access Control Grouping*

Cisco Unified CM uses access control groups to apply specific roles to users. Although you cannot modify the default groups that are present, you can edit any new access control groups that are added. A good practice is to add two additional access control groups for the end users. These groups are then applied to the users based on their role:

- **Standard End User:** Assigned to all end users with a configured endpoint. This group allows access to the Cisco Unified CM Self Care Portal.
- **Standard Administrator:** Assigned to UC administrators. This group allows read/write access to Cisco Unified CM and restricts access to audit logs.

Table 6-7 covers the roles that should be assigned for these access control groups.

https://t.me/learningnets
@cisco_in_persian_channel

Table 6-7 *Standard Access Control Groups*

Access Control Group Name	Assigned Roles	Role Description
Standard End User	Standard CCM End User	Allows access to Self Care Portal
	Standard CTI Enabled	Allows control of IP phone remotely Used for soft clients like Cisco Jabber
	Standard CTI Allow Control of Phones supporting Connected Xfer and conf	Allows control of IP phone remotely Used for soft clients like Cisco Jabber
	Standard CTI Allow Control of Phones supporting Rollover Mode	Allows control of IP phone remotely Used for soft clients like Cisco Jabber
Standard Administrator	Standard AXL API Access	Allows access to the AXL APIs
	Standard Admin Rep Tool Admin	Allows you to view and configure Cisco Unified Communications Manager CDR Analysis and Reporting (CAR)
	Standard CCM Admin Users	Allows access to the Cisco Unified CM Administration web pages
	Standard CCMADMIN Administration	Allows administration of all aspects of CCMAdmin system
	Standard CUReporting	Allows application users to generate reports from various sources
	Standard CUReporting Authentication	Allows users to authenticate to Cisco Unified Reporting page
	Standard EM Authentication Proxy Rights	Manages EM authentication rights
	Standard SERVICEABILITY Administration	Administers all aspects of Serviceability system
	Standard SSO Config Admin	Administers SAML SSO configuration

Note The administrator group created in the Table 6-7 intentionally does not have the Standard Audit User role assigned to it. It is a leading practice to have a separation between administrator accounts and auditor accounts.

https://t.me/learningnets

@cisco_in_persian_channel

An additional step that can be taken regarding access control groups available to a user and the roles available is user ranking. When you rank users, you can refine which users have access to specific access control groups based on the rank assigned to the user. User ranks can be applied to both end users and application users manually, using the Bulk Administration Tool (BAT), or when imported from an LDAP directory. Administrators can create their own user rank structure that can be applied when users are provisioned and access control groups are configured. The default user rank of 1 is applied to all users and access control groups unless additional user ranks are configured; they are not, by default. Up to 10 user ranks can be configured ranging from the default of 1 (highest) through 10 (lowest). The specified user rank applied to an access control group is the minimum user rank a user must meet to have the access control group applied. Using the Standard Administrator access control group as an example, if you change the user rank from 1 to 4, only the users with a user rank of at least 4 can have that access control group applied to it. If a user has a user rank between 5 and 10, it would not be possible to apply that access control group to them.

User accounts can be further restricted with the configuration of Advanced Role Settings, which enable you to restrict access for tasks such as

- Adding users
- Updating passwords
- Editing user ranks
- Editing the roles in access control groups

To access Advanced Role Settings, you must create a custom role. This task can be as simple as copying the existing Standard CCMADMIN Administration role located in Cisco Unified CM Administration GUI under **User Management > User Settings > Role**. After the role is copied, you select **Advanced Role Configuration** in the Related Links drop-down. Then you select the appropriate Resource Web Page, as shown in Figure 6-5.

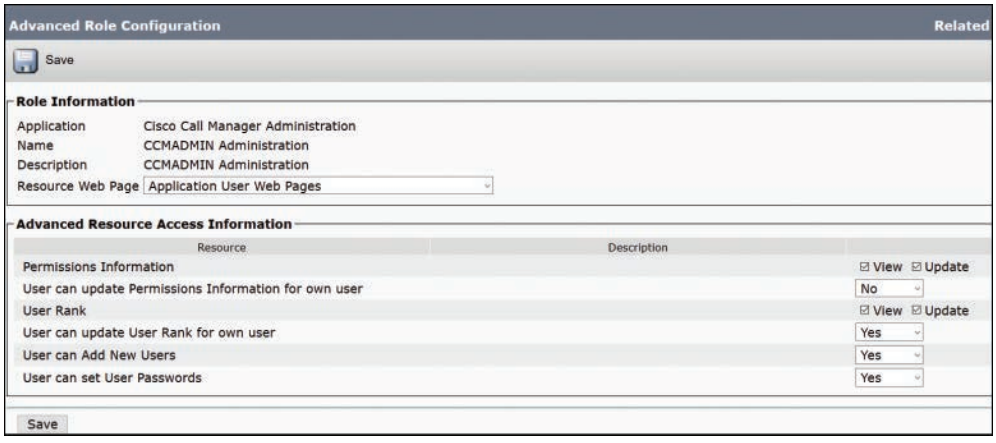


Figure 6-5 *Advanced Role Configuration*

https://t.me/learningnets
@cisco_in_persian_channel

Table 6-8 outlines the configuration options available with the Advanced Role Configuration settings.

Table 6-8 *Advanced Resource Access Information*

Advanced Resource	Access Control
Permission Information	Controls the ability to update or change access control groups. View: A user has read-only rights to view access control groups. Update: A user can create, update, and remove access control groups.
User can update permission information for own user	Controls a user's ability to change their own permissions. Yes: A user can change their own permissions. No: A user cannot change their own permissions, but they do have the ability to change the permissions for other users who have the same or lower user rank.
User Rank	Controls the ability to change the user rank. View: A user can view the user rank but cannot change it. Update: A user can change the user rank.
User can update User Rank for own user	Controls a user's ability to change their own user rank. Yes: A user can change their own user rank. No: A user cannot change their own user rank, but they do have the ability to change the user rank for other users who have the same or lower user rank.
Add New Users	Controls the ability to add a new user. Yes: The user can add new users. No: The Add New button is not present.
Passwords	Controls the ability to change passwords. Yes: The user can change the user passwords under Application User Information. No: The Password and Confirm Password under Application User Information section is not present.

A more detailed list of the standard roles is available in the "Manage User Access" section of the Administration Guide for Cisco Unified Communications Manager.

https://t.me/learningnets

@cisco_in_persian_channel

Assigning User Roles and Credential Policies to Users

With the proper credential policies and access control groups created, they should be assigned to end users. There are several methods available to assign the roles and policies to accounts, such as manually, with the Bulk Administration Tool (BAT), and automatically with the integration with LDAP.

- **Manually assigned user roles and credential policies:** This method is best for creating or updating a small number of users. It affects only one user at a time. With this method, all editable aspects of the user can be assigned or updated at once.
- **Bulk Administration Tool:** This tool has been around since the early days of Cisco Unified CM and can be used to create or update end users in bulk either using a query or a CSV file. The query is limited in the scope of user attributes that can be updated while the CSV file allows for updating most of the editable end-user attributes. Administrators should be careful when using this method due to the number of accounts that can be created or updated at once.
- **LDAP integration:** This is the standard method used by most organizations due to the ability to automatically sync end-user accounts to the existing corporate directory. With this method, only the information that is applicable to the end-user within the UC application can be modified. First name (givenname), last name (sn), and user password are among some of the attributes stored within the corporate LDAP directory. User attributes such as the roles, controlled devices, home cluster, and PIN are configurable via the LDAP directory integration during import.
- **API:** The UC applications have APIs that provide a programmatic interface that can be used to create, update, assign, and delete user roles and credential policies. The specifics for using the APIs are beyond the scope of this book.

Importing End Users from a LDAP Directory

Users imported via an LDAP integration are another type of end user. Cisco UC applications support integration with any LDAPv3-compliant directory such as Microsoft Active Directory and OpenLDAP. Although the Cisco UC applications can integrate with different types of LDAP directories, we only cover how to secure the integration with Microsoft Active Directory. We chose Microsoft AD because it is widely prevalent and the steps to secure LDAP are consistent across the different LDAP directories. One of the benefits you get with the UC application integrated with an LDAP directory is a single logon experience. End users who are part of the LDAP integration can use their corporate username and password to log in. This single login reduces the number of usernames and passwords that end users must keep track of. This type of integration also allows a UC administrator to build out a corporate directory that can be accessed via the UC endpoints. For a more detailed explanation of the pros and cons of integrating UC applications with an LDAP directory, refer to the "Directory Integration and Identity Management" chapter of the Cisco Collaboration System 12.x Solution Reference Network Design.

https://t.me/learningnets

@cisco_in_persian_channel

When you are integrating with an LDAP directory, it is important to understand the steps involved with enabling the integration. The first is to simply enable synchronizing from the LDAP directory. This step is followed by configuring the LDAP directory that the application will be synchronized with. Lastly, you need to configure LDAP authentication. This last step can be to the LDAP directory, or it can be to an identity provider (IdP) if you enable single sign-on (SSO). The next few sections walk you through the steps for enabling integration with an LDAP directory.

Enabling the Required Services for Importing End Users Using LDAP

To enable Cisco UC applications to synchronize end users with an LDAP directory, you first need to start the Cisco DirSync service. For Cisco Unified CM, you enable the service by selecting the service listed in **Cisco Unified Serviceability > Tools > Service Activation**, as shown in Figure 6-4. When it is enabled, if needed, you can verify the service either by using the web page shown in Figure 6-6 or using the **utils service list page** command at the CLI.

Figure 6-6 *Cisco DirSync Enabled (GUI)*

With the Cisco DirSync service enabled, the next step is to enable synchronizing from LDAP and select the attribute on the user account that will be used as the user ID. For Microsoft Active Directory, this is typically the sAMAccountName or the UserPrincipalName (UPN) for large or multiforest deployments. A key difference between the sAMAccountName and UPN is that the UPN is guaranteed to be unique within Microsoft Active Directory. For Cisco Unified CM, you enable LDAP synchronization by logging in to the Cisco Unified CM Administration GUI and selecting **System > LDAP > LDAP System Configuration**, as shown in Figure 6-7.

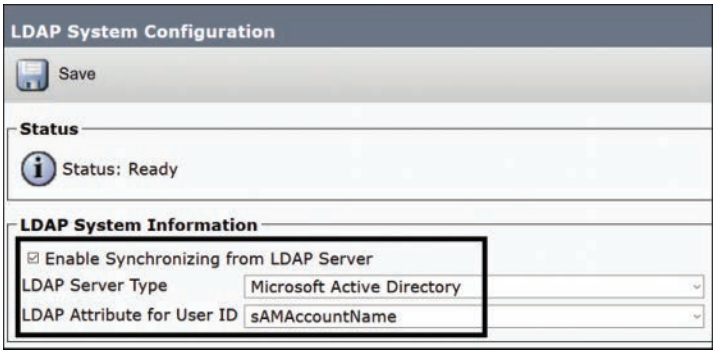


Figure 6-7 *LDAP System Configuration*

https://t.me/learningnets
@cisco_in_persian_channel

LDAP Directory Configuration and Overview

After you enable LDAP directory synchronization, the next step is to configure a directory to synchronize with. Still using Microsoft AD, you need to understand a few concepts as they relate to the LDAP directory:

- **Location matters:** It is important to understand the difference between the roles for a domain controller (DC) and a global catalog (GC) server. The ports used for the two roles are different. A domain controller uses TCP/UDP port 389 when not using SSL and TCP port 636 when SSL is used. A global catalog server uses TCP ports 3268 and 3269 for nonsecure and secure communication, respectively.
- **Directory hierarchy:** A user account is required to query the LDAP directory to import the users into the UC application. The account used needs only read access to the organizational units (OUs) and containers (CNs) where the imported accounts are located. It is a good practice to limit this account to have only read access to the locations where the imported accounts reside.
- **Scale:** A Cisco Unified CM cluster can have up to 20 different synchronizations to an LDAP directory. If the number of users is more than 80,000, the number of synchronizations drops to 10. Although, technically, UC applications like Cisco Unified CM do not enforce a hard limit on the number of users and groups that can be imported, a good rule of thumb is to import only the users who need access to the application or up to twice the number of devices on the cluster. From a design perspective, Cisco recommends not importing more than 80,000 users.
- **Secure connections:** On August 13, 2019, Microsoft issued Security Advisory ADV190023, "Guidance for Enabling LDAP Channel Binding and LDAP Signing." With this security advisory, Microsoft mandates that connections to Active Directory should use Secure LDAP (LDAPS). The impact for LDAP integrations with Cisco UC products is that now you need to either install the LDAP server certificate or the root and/or the applicable intermediate certificates of the certificate authority (CA) that signs the LDAP server certificate to the Tomcat-trust store of the application. An additional implication for this type of connection is that the secure LDAP ports should be used.

The next step is to collect the LDAP certificate chain. First, you need to collect the LDAP root CA and subordinate CA certificates. These can be collected in numerous ways, like downloading them directly from the certificate authority, which is usually the easiest method, or using the OpenSSL tool.

For this example, use OpenSSL because it also allows you to download the LDAP server SSL certificate along with the CA chain. Using OpenSSL from the command line, run **openssl s_client -connect <LDAP_Server>:636** replacing **<LDAP_Server>** with the IP address or FQDN of the LDAP server you will be configuring in Cisco Unified CM. This command outputs the LDAP SSL certificate, which can then be used to extract the CA certificates used to sign the LDAP SSL certificate.

https://t.me/learningnets

@cisco_in_persian_channel

Potentially, two different ports can be used for connecting securely with the LDAP server. Depending on whether the server is a domain controller or global catalog, you should try to connect to either port 636 or 3269, respectively. Example 6-1 demonstrates the output of the **openssl** command when it is used to collect the LDAPS certificate.

Example 6-1 *LDAP SSL Certificate*

```
 PS C:\Program Files\OpenSSL-Win64\bin> .\openssl s_client -connect 10.1.10.76:636
CONNECTED(0000017C)
Can't use SSL_get_servername
depth=1 DC = com, DC = ciscopucs, CN = ciscopucs-SUBCA-CA
verify error:num=20:unable to get local issuer certificate
verify return:1
depth=0
verify return:1
---
Certificate chain
 0 s:
   i:DC = com, DC = ciscopucs, CN = ciscopucs-SUBCA-CA
 1 s:DC = com, DC = ciscopucs, CN = ciscopucs-SUBCA-CA
   i:DC = com, DC = ciscopucs, CN = ciscopucs-AD-DNS-CA-EXCH-CA
---
Server certificate
-----BEGIN CERTIFICATE-----
MIIFwjCCBKqgAwIBAgITWAAAAAOK91F8PuwP6wAAAAAAAzANBgkqhkiG9w0BAQsF
ADBNMRMwEQYKCZImiZPyLGQBGRYDY29tMRkwFwYKCZImiZPyLGQBGRYJY2lzY29w
dWNzMRswGQYDVQQDExJjaXNjb3B1Y3MtU1VCQ0EtQ0EwHhcNMjAwMzI2MjIwNTM5
WhcNMjEwMzI2MjIwNTM5WjAAMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKC
AQEAofuLhw4renTrEOd98i8lRQLE5qd9rn5Q302DxvQ2QyC7kk/WruTcAtEEnLTi
obQhx/KkQmMf06rq0fjbFlMpGBSMoguxzTs6sV5gJKvIAfQ7yZcChswWq52yZ/LL
**Output Ommited**
Q2xhc3M9Y2VydGlmaWNhdGlvbkF1dGhvcml0eTBEBgNVHREBAf8EOjA4ghxBRC1E
TlMtQ0EtRXhjaC5jaXNjb3B1Y3MuY29tgg1jaXNjb3B1Y3MuY29tgglDSVNDT1BV
Q1MwDQYJKoZIhvcNAQELBQADggEBAINK5dbQ9IJPaegf+43mJv4QPi5sSpH3b+cz
Fm8Lr7bZenApHE6AUbEnOF5jBnmaFC5Vv1M7oYorNBhVJyUht/xg8KZkx7oJ2KxO
MQuDK3ZiIu1xuagnv78LcLeajHDx+1251+qAbDPyy05LEwY+V/vkiWvSy27BE+2n
GD2EFodMe2uYud3c8OCPXE21HQJ3qjIxpLUrUEiiVwHcqDuyp4n96AFdrmSj5LBv
8RY36GM34Gt+UldwykpFj5c65br0dF+du2C44zBztadfamxAcJorQrzncMOJe9z0
B8Pznuc2Fy6DtmkEIKXZUDKkP1rQzYx4avgAccRpD0ggvSuQeFY=
-----END CERTIFICATE-----
```

Next, you copy the certificate information starting with **-----BEGIN CERTIFICATE-----** through **-----END CERTIFICATE-----** and paste it into a text file named LDAP_Server.cer. You can then open this certificate, which provides the certification path, as shown in Figure 6-8.

https://t.me/learningnets

@cisco_in_persian_channel

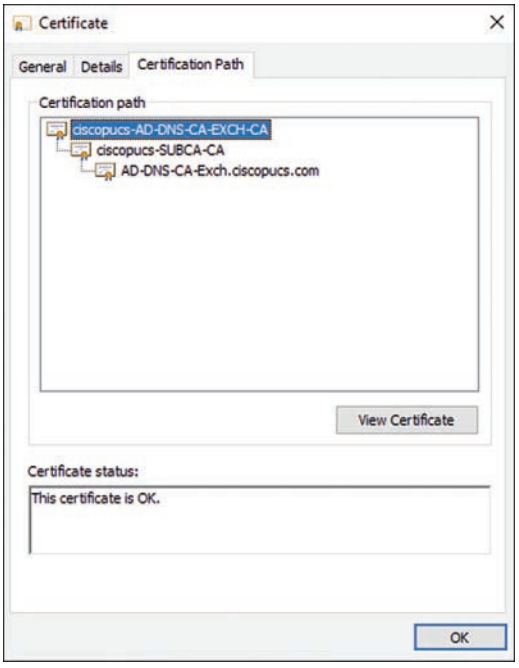


Figure 6-8 *LDAP Sever Certificate Chain*

Selecting each certificate in the chain followed by **View Certificate** allows you to export the certificate for the root and intermediate CA, respectively.

Note It is important to save the root and intermediate CA certificates as Base64-encoded X.509 (.CER).

Note The LDAP certificate by itself can be used in lieu of the root and subordinate CA certificates. However, if multiple LDAP servers are configured, a UC administrator needs to upload all LDAP certificates to the Tomcat-Trust store that will be configured.

Once exported, the root and intermediate certificates should be uploaded in Cisco Unified CM to the Tomcat-Trust within Cisco OS Administration. You do this by logging in to Cisco Unified OS Administration and navigating to **Security > Certificate Management**. The process to upload a certificate to the Unified CM Tomcat-Trust is illustrated in Figure 6-9.

https://t.me/learningnets

@cisco_in_persian_channel

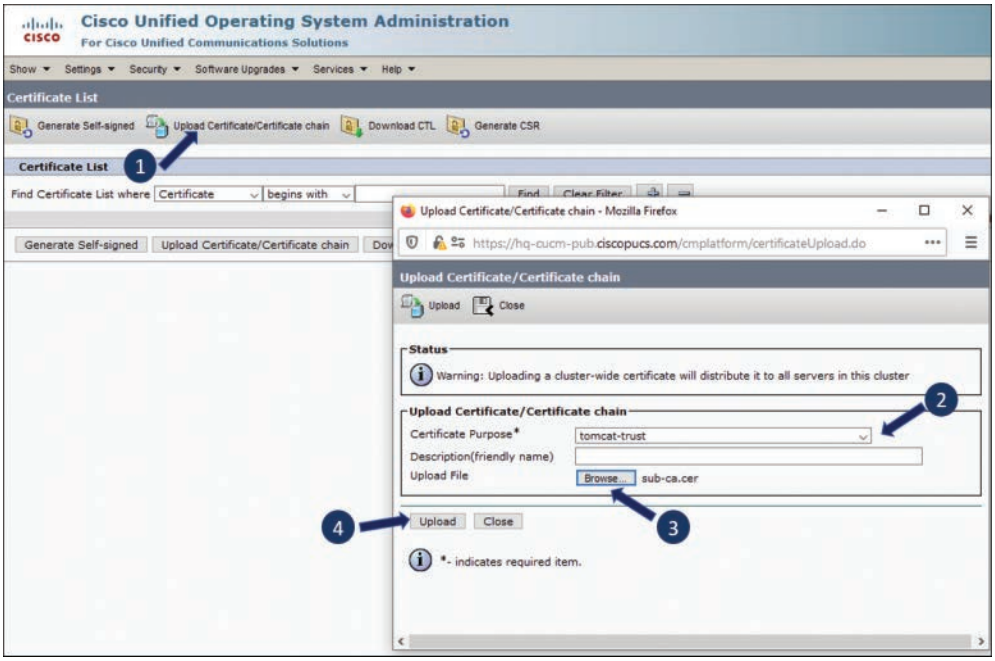


Figure 6-9 *Uploading the LDAP SSL Certificate to Tomcat-Trust*

The steps are as follows:

1. Select **Upload Certificate/Certificate Chain.**
2. In the new window, open the **Certificate Purpose** drop-down and select **tomcat-trust.**
3. Upload the issuing CA certificate or the LDAP SSL certificate chain.
4. Select **Upload.**

When the proper certificate chain is installed in Cisco Unified CM, you can begin the LDAP directory configuration process. Cisco offers several ways to assign default permissions when they are added to the local database. A key component of those are the access control groups, which we discussed previously.

Now that the proper certificates are loaded into Cisco Unified CM, the process to enable LDAP directory continues with logging in to Cisco Unified CM Administration GUI and navigating to **System > LDAP > LDAP Directory**. For this step, you need a couple of configuration items (see Table 6-9).

https://t.me/learningnets

@cisco_in_persian_channel

Table 6-9 *LDAP Directory Configuration*

LDAP Directory Information	
LDAP Configuration Name	Descriptive name for the LDAP directory (e.g., UC Users).
LDAP Manager Distinguished Name	Distinguished name for the user performing the simple bind to the directory (e.g., CN=LDAP Sync,OU=Unified Communications,OU=Service Accounts,DC=ciscopucs,DC=com).
LDAP Password	The password for the account used for the directory bind and the password should be set to never expire. This account should have only read access to the OU/CN where the users/groups are located.
LDAP User Search Base	Search base within the LDAP directory where the users that will be imported are located (e.g., OU=Congress,DC=ciscopucs,DC=com).
LDAP Filter	Any LDAP filter that you configure must comply with the LDAP search filter standards that are specified in RFC4515. In most cases, the default LDAP filter can be used for user imports. Custom LDAP filters can be created so that custom attributes within LDAP (such as Region Codes/Site Codes) can be used to further refine the users and groups that are imported into the directory. The default LDAP filter that is applied is **(&(objectclass=user)(!(objectclass=Computer))(!(UserAccountControl: 1.2.840.113556.1.4.803:=2)))** The default filter includes all users who are not also computers and are not disabled. The LDAP filter can be customized to selectively import users based on attributes set on their accounts. The following example builds on the default LDAP filter and further filters users to include only users with specific attributes set: **(&(objectclass=user)(!(objectclass=Computer))(!(UserAccountControl: 1.2.840.113556.1.4.803:=2))(|(ciscoPUCSAttribute1=Congress) (ciscoPUCSAttribute1=Judiciary)))**
LDAP Server Information	
Host Name or IP Address for Server*	Up to three LDAP servers can be configured. These servers can be either DCs or GCs and should be configured to use either port 636 or 3269 based on the role of the LDAP server configured. TLS should be enabled to secure the communications regardless of which LDAP implementation is used.

https://t.me/learningnets

@cisco_in_persian_channel

The directory should be set to periodically synchronize with the corporate LDAP servers so that moves, adds, and changes can be replicated into the end-user database. Synchronizations can be configured to perform a single synchronization or based on time frames of hours, days, weeks, and months, with the minimum time between scheduled repeating synchronizations being 6 hours. A good practice is to perform a synchronization every 12–24 hours unless there is an identified impact to the LDAP directory. Customers should consider the time between synchronizations to account for the average number of user adds, deletes, or the time frame that accounts are disabled. New accounts are not imported until the UC application syncs with the directory. You should also consider this delay before accounts are populated in the end-user directory. Accounts that are deleted from the LDAP directory are not considered inactive until a synchronization takes place. After a synchronization is complete and the account is marked inactive, the account is still not deleted from the end-user directory for another 24 hours. This delay before the account is marked inactive and ultimately removed from the end-user database must be accounted for because the end user has access to the system until the account is ultimately removed from the end-user database. This process is illustrated in Figure 6-10 where Bob's account is deleted at 1:00 p.m. on January 1.

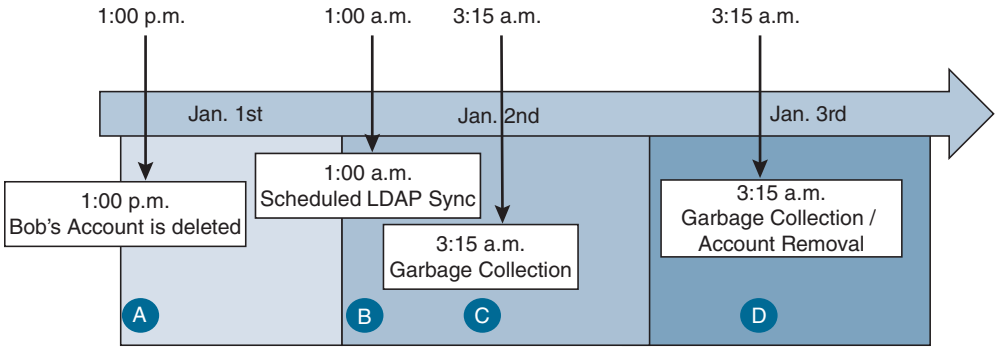


Figure 6-10 *LDAP Synchronization Timeline*

An active LDAP-synchronized end user in the UC application uses the timeline shown in Figure 6-10 for deletion or disablement.

A. Bob's LDAP account is deleted in Active Directory at 1:00 p.m., and the configured LDAP directory synchronizations are configured to begin at 1:00 a.m. daily. In this instance Bob's user account is not marked inactive until 1:00 a.m. the following day. During this time, if LDAP authentication is enabled, Bob cannot log in via his password because it resides in Active Directory. The PIN, however, still successfully authenticates until the account is marked inactive because it is local to Cisco Unified CM.

B. The scheduled synchronization completes, and users who have been deleted are marked inactive. At this point, Bob cannot log in using his password or PIN.

https://t.me/learningnets

@cisco_in_persian_channel

C. The nonconfigurable Garbage collection service runs at 3:15 a.m. every day. Because Bob's account has not been marked inactive for at least 24 hours, the account is not removed from the end-user database.

D. The Garbage collection service runs at 3:15 a.m., and because Bob's account has been marked inactive for over 24 hours now, the account is permanently removed from the end-user database.

For more details on the process to synchronize (add) and remove end users in the UC application synchronized from an LDAP directory, refer to the "Directory Integration and Identity Management" chapter in the Cisco Collaboration System 12.x Solution Reference Network Designs.

Additional options to set via the LDAP directory integration include the fields for mapping users. You can use these attributes to customize the users during import. The choice of whether to use the mail attribute or msRTCSIP-primaryuseraddress to configure the end-user Directory URI setting when the user is imported to the end-user database is shown in Figure 6-11.

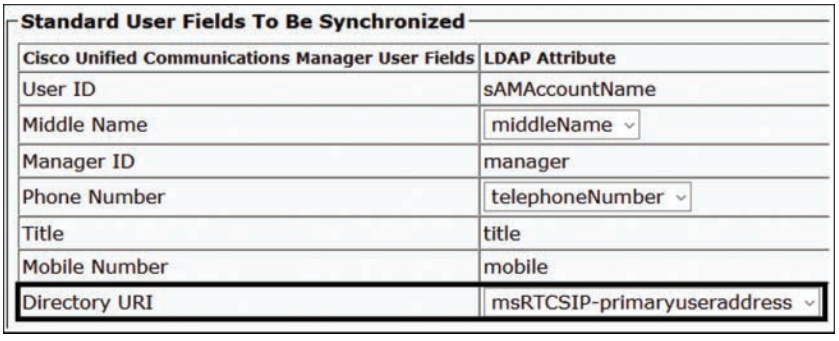


Figure 6-11 *LDAP Directory User Attributes*

When implementing an LDAP integration, you should carefully consider assigning default roles to end users when they are imported. Based on the directory structure, multiple synchronizations can be used to assign roles to users as either a standard user, administrator, or no access (see Figure 6-12). If you do not assign users to an access control group when they are imported or created, they do not have access to the Cisco UC application but are available in the local directory.

Note Using separate user accounts for administration of the UC applications and day-to-day end-user functions is recommended.

https://t.me/learningnets

@cisco_in_persian_channel

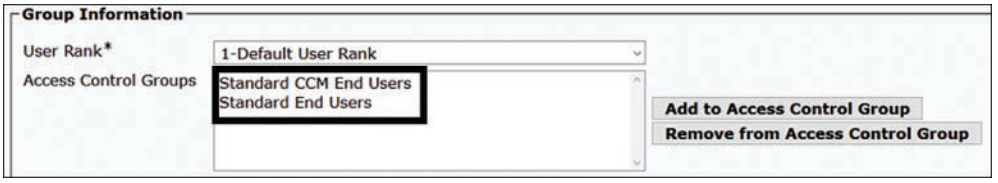


Figure 6-12 *LDAP Directory Group Information*

After you configure the directory, save the configuration and perform a manual synchronization to import the users into the directory. Then continue with configuring authentication against the corporate LDAP directory directly or enabling single sign-on using SAML 2.0. Both methods of authentication are described in the upcoming sections.

Note End users synched to an LDAP directory are not required to authenticate with the LDAP directory. If LDAP authentication is not enabled, the imported end users can authenticate against the local end-user database or via SSO using SAML.

Configuring LDAP Authentication for Imported End Users

When LDAP authentication is enabled, Cisco Unified CM can authenticate LDAP-synchronized end users to the corporate LDAP directory. This authentication uses an LDAPv3 connection established between the Identity Management System (IMS) module within Cisco Unified CM and the corporate LDAP directory. Locally configured users and application users never have their passwords authenticated against the corporate LDAP directory. Additionally, all users' PINs are authenticated only against the local database.

There are two methods to configure authentication. The first is to enable authentication with the LDAP servers directly, and the second is to use SSO. When you're enabling authentication directly against LDAP servers, the configuration process is similar to the LDAP directory synchronization configuration. In this case, up to three LDAP servers can be configured for redundancy with or without LDAP over SSL (LDAPS) enabled. The process of enabling LDAPS uses the configuration steps you used when configuring the LDAP directory.

Enabling LDAP authentication is a logical process; the first step is to enable LDAP authentication directly with the configured LDAP servers. Authenticating directly to the corporate LDAP directory provides end users with a consistent login experience in which they can use the same credentials used to log in to their workstations and other network resources. Following the configuration of the initial authentication method, you, as

https://t.me/learningnets
@cisco_in_persian_channel

administrator, can configure SSO if it is deployed elsewhere in the network. When SSO is enabled, users need to log in to their browsers only once to access the Cisco UC application GUI that is enabled and that they have rights for.

Note It is not a requirement to have LDAP authentication enabled when you enable LDAP directory within Cisco UC applications. However, the reverse is not true; if you are enabling LDAP authentication, LDAP directory must be properly configured.

Now let's walk through the process of enabling authentication with the corporate LDAP servers. To do so, you log in to Cisco Unified CM Administration GUI and navigate to **System > LDAP > LDAP Authentication**. Then you populate the LDAP authentication configuration appropriately as shown in Table 6-10.

Table 6-10 *LDAP Authentication*

LDAP Authentication	
Use LDAP Authentication for End Users	Selected.
LDAP Manager Distinguished Name	The distinguished name for the user performing the simple bind to the directory (e.g., CN=LDAP Sync,OU=Unified Communications,OU=Service Accounts,DC=ciscopucs,DC=com). This account should be able to read all the OUs where users and groups imported into the directory are located.
LDAP Password	The password for the account used for the directory bind and the password should be set to never expire.
LDAP User Search Base	The search base within the LDAP directory where the users who will be imported are located (e.g., OU=Congress,DC=ciscopucs, DC=com). If you have multiple synchronizations, the specified path should have visibility to all the accounts that have been imported.

You can have only a single LDAP authentication configuration. You should configure the corporate LDAP servers used for authentication to connect on TCP port 636 for a domain controller and port 3269 for a global catalog with TLS enabled. Failure to enable LDAP over TLS results in the end-user passwords being sent in cleartext across the network. For example, user Adam Smith is trying to log in to the Cisco Unified CM Self Care Portal; in Figure 6-13, you can see from a packet capture that the password is sent in the clear.

https://t.me/learningnets

@cisco_in_persian_channel

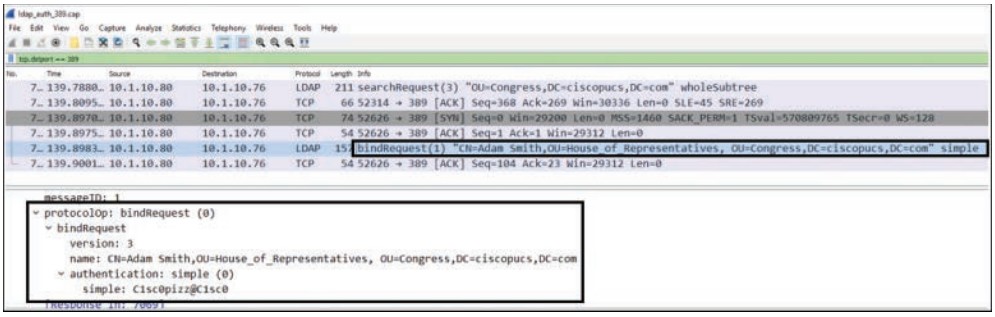


Figure 6-13 *LDAP Authentication Without TLS*

At this point, LDAP-synchronized end users can log in to both Cisco Unified CM Administration GUI along with the Cisco Unified CM Self Care Portal. As a reminder, application users or local end users to the UC application still authenticate their passwords against the local user databases. Even with LDAP authentication enabled, all user PINs still authenticate against the local end-user database. The process to enable corporate LDAP directory integration is consistent across both Cisco Unified CM and Cisco Unity Connection. The other Cisco UC applications like Cisco Emergency Responder and Cisco IM&P do not integrate directly with LDAP but rather through their connections to Cisco Unified CM or by being part of the Cisco Unified CM cluster.

Using Single Sign-On to Simplify the Login Experience

Cisco uses SAML to facilitate the ability to use single sign-on (SSO) to the Unified CM and other UC applications. When SSO is enabled, it allows users to log in one time with a set of credentials to access the Cisco UC applications they have access to. This section introduces SAML, which facilitates the SSO login. Then we describe the process to configure Unified CM to support SSO for both end users and administrators. When SSO is enabled, it provides greater security compliance because the authentication is removed from the application—in this case, Unified CM. User satisfaction also tends to increase due to the need to log in once per session when accessing the UC applications. This last benefit reduces overhead for trouble tickets and management of the system.

Intro to Security Assertion Markup Language (SAML)

This section provides a foundational level of understanding about SAML as it relates to enabling single sign-on for the different UC applications. SAML was developed by the Security Service Technical Committee at the Organization for the Advancement of Structured Information Standards (OASIS) and is intended to provide a framework for exchanging security and identity information between different systems. SAML does not perform any authentication of services. What it does, however, is provide a method for

https://t.me/learningnets
@cisco_in_persian_channel

interacting with authentication servers. The current version of SAML used within Cisco Unified Communications is version 2, and it is not backward compatible with version 1.0 or version 1.1. The components that make up SAML ultimately are reliant on a web browser and are outlined in Figure 6-14.

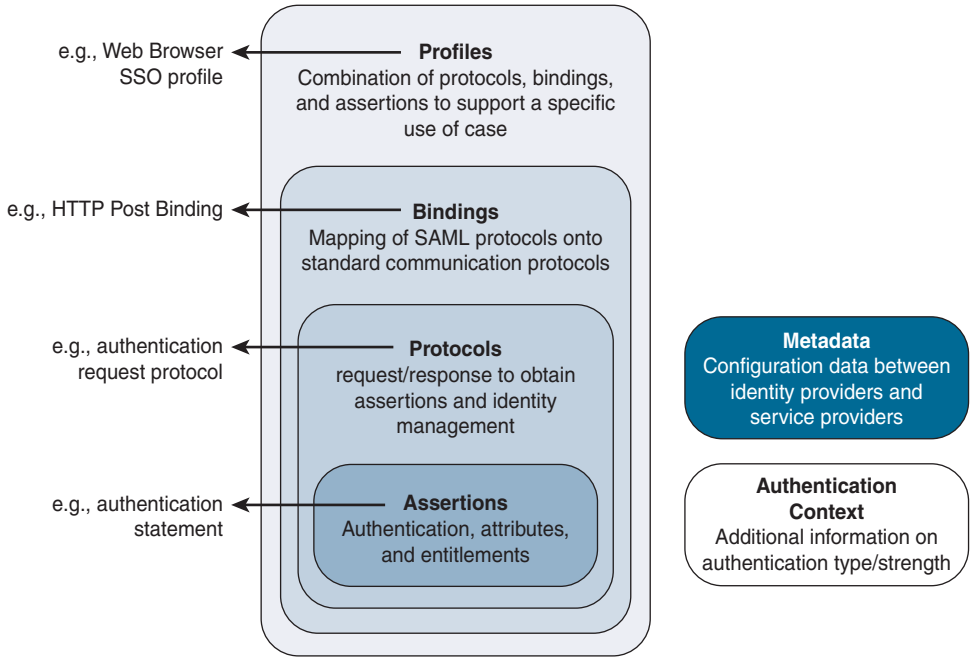


Figure 6-14 *SAML Components*

SAML consists of these three main concepts (see Figure 6-15):

- **Client (the user's client):** This is a browser-based client or a client that can leverage a browser instance for authentication. It is the entity making the request for access and providing the credentials for authentication (username/password).
- **Service provider:** This is the application or service that the client is trying to access—for example, Unified Communications Manager. It can also be called the relying party because the SP relies on the IdP to trust whether access should be granted.
- **Identity provider (IdP) server:** This can otherwise be known as an asserting party; it verifies you are who you say you are. In other words, it is the authentication (AuthN) server and uses an Identity Management System like LDAP to perform the actual authentication. The IdP can also verify whether the access request to a resource was previously approved or not.

https://t.me/learningnets
@cisco_in_persian_channel

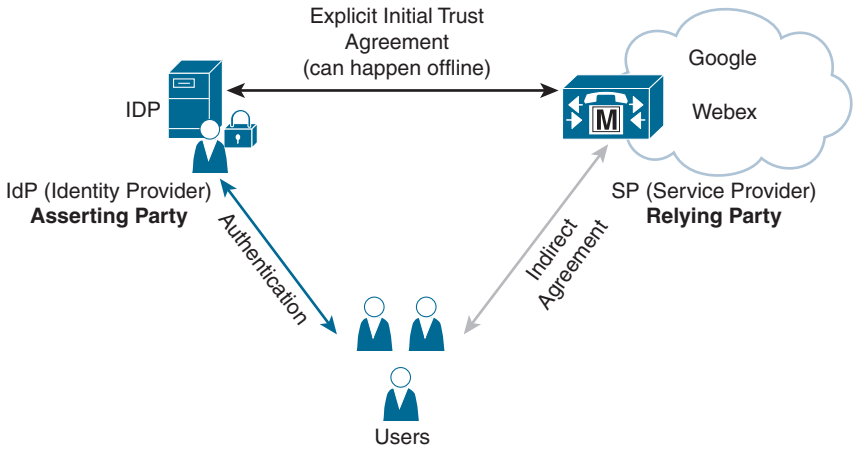


Figure 6-15 *SAML Authentication Framework*

Additional components that take part in SAML are

- **Lightweight Directory Access Protocol (LDAP) users:** These users are integrated with an LDAP directory—for example, Microsoft Active Directory or OpenLDAP. Non-LDAP users reside locally on the Unified Communications server.
- **SAML assertion:** It consists of pieces of security information that are transferred from IdPs to the service provider for user authentication.
- **SAML request:** This authentication request is generated by a Unified Communications application. To authenticate the LDAP user, Unified Communications application delegates an authentication request to the IdP.
- **Circle of Trust (CoT):** It consists of the various service providers that share and authenticate against one IdP in common.
- **Metadata:** This XML file is generated by an SSO-enabled Unified Communications application (e.g., Unified Communications Manager, Cisco Unity Connection) as well as an IdP. The exchange of SAML metadata builds a trust relationship between the IdP and the service provider. The metadata contains the public certificates of the SP along with other information like the entity ID, entity attributes, and role descriptor, among others.
- **Assertion Consumer Service (ACS) URL:** This URL instructs the IdPs where to post assertions. The ACS URL tells the IdP to post the final SAML response to a specific URL.

Note In the explicit trust relationship between the IdP and SP, the agreements are completed offline with a metadata exchange.

https://t.me/learningnets

@cisco_in_persian_channel

Let's examine a simplified scenario where a user logs in to the Cisco Unified CM Administration GUI. In this scenario, the following actions take place. First, a UC administrator attempts to log in to Cisco Unified CM Administration. The Unified CM responds with a redirect with an authentication request to the user's web browser, which redirects to the IdP. The location of the IdP is contained in the metadata that is initially shared when Unified CM is configured for single sign-on. (We cover the configuration process shortly.) After the request is redirected, the user is prompted to authenticate. The IdP uses the configured Relying Party Claim Issuance Policy that is specified for the service provider to perform that actual authentication. With a successful authentication, the IdP sends an SAML response with assertion that is signed by the IdP with a cookie to the user's web browser. Based on the URL response, the browser sends an HTTP Post with an SAML assertion back to the SP. Because the metadata that was shared previously between the IdP and Unified CM contains the public certificates, the SP can validate the signature of the SAML assertion. At this point, the SP can provide access based on the verified SAML assertion and sends it back to the browser.

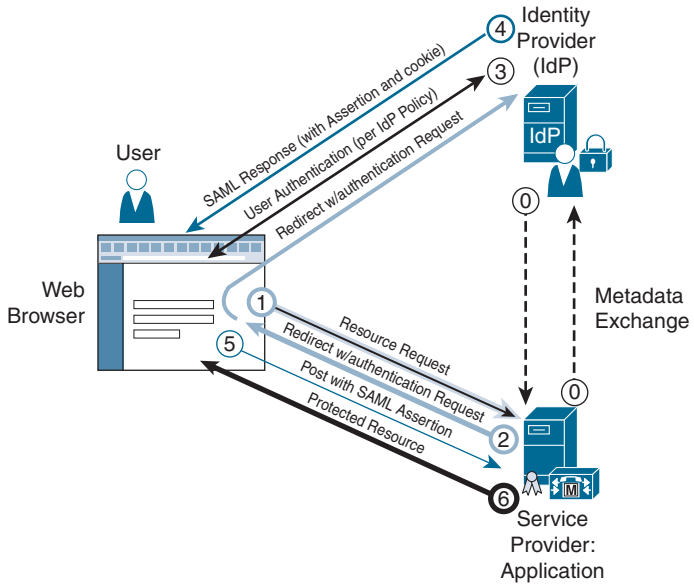


Figure 6-16 *SP-Initiated Web Browser SSO Flow*

Figure 6-16 illustrates a service provider–initiated web browser flow with SAML SSO. The steps occur as follows:

0. Metadata exchange happens during initial configuration between IdP and SP.

1. The user wants access to a restricted resource (SP) such as Unified CM.

2. The SP resource issues a redirect with authentication request to the user's web browser to the IdP. Metadata is used to identify where to send the request.

https://t.me/learningnets

@cisco_in_persian_channel

3. Authentication happens between the IdP and user, but neither the IdP nor the SP controls authentication. The authentication can be username/password, MFA, and so on, and is specified via the configured IdP policy.
4. Successful authentication sends an SAML response with assertion (signed by IdP) with a session cookie to the user.
5. Based on the URL in the response, the web browser sends a POST with an SAML assertion to the SP. The SP has the public certificates that are contained in the metadata to verify the SAML assertion.
6. The SP can provide access based on the verified SAML assertion and send it back to the browser.

Cisco's single sign-on implementation uses SAML. As stated earlier, SAML allows system administrators and end users access to a set of Cisco Collaboration applications after signing in. SAML is an authentication protocol used by service providers like Cisco Unified CM to authenticate users and facilitate the exchange of security information between trusted systems. Through SAML, the exchange of security authentication information between an identity provider and a service provider is accomplished. This feature provides a secure mechanism to use common credentials and relevant information across various applications.

SAML SSO establishes a circle of trust (CoT) by exchanging metadata and certificates as part of the provisioning process between the IdP and the service provider. The service provider trusts the IdP's user information to provide access to the various services or applications.

A strong reason to implement SSO with Cisco UC applications like Cisco Unified CM and the soft client Cisco Jabber is the streamlined authentication process. Once logged in, a user needs to authenticate only once to one of the Cisco UC services and then can access any of the other services provided by the service provider GUIs without having to log in again.

Enabling SSO does the following:

- Moves the authentication from the UC application that hosts the service to the IdP. The IdP and service provider form a circle of trust through the exchange of metadata with the IdP to authenticate users.
- Allows for advanced authentication methodologies such as two-factor authentication, tokens, and smart cards. This is accomplished by using SSO and removing the authentication from the application and using the IdP to control the authentication based on a predefined configuration.
- Provides encryption functions to protect authentication information passed between the IdP, service provider, and user. SAML SSO can also hide authentication messages passed between the IdP and the service provider from any external user.
- With SAML SSO enabled the number of individual logins is reduced. A benefit of the reduced logins is potentially reduced costs due to decreased calls to the help desk for password resets.

https://t.me/learningnets

@cisco_in_persian_channel

- SAML SSO authorization uses role-based access control (RBAC) configured within the Cisco UC application.

The following Cisco UC applications support SAML SSO:

- Cisco Unified CM – End User SAML SSO supported from version 10.x
- Cisco IM&P – End User SAML SSO supported from version 10.x
- Cisco Unity Connection – End User SAML SSO supported from version 10.x
- Cisco Emergency Responder – End User SAML SSO supported from version 11.5
- Cisco Prime Collaboration – End User SAML SSO supported from version 10.x
- Windows version of the Real-Time Monitoring Tool (RTMT)
- Cisco Expressway

Note VOS-based Cisco UC applications starting with version 12.x support the integration of SAML SSO with the Platform applications (CLI, DRS, OS Administration). Versions of applications 11.5 and earlier did not support this.

Configuring Cisco Unified CM for SAML SSO

Next, we describe the three-step process to configure SAML SSO for Cisco UC applications (shown in Figure 6-17) using Cisco Unified CM as an example. Because all networks differ, it is best to reference the latest configuration guides for SAML SSO on the Cisco website prior to beginning the configuration process.

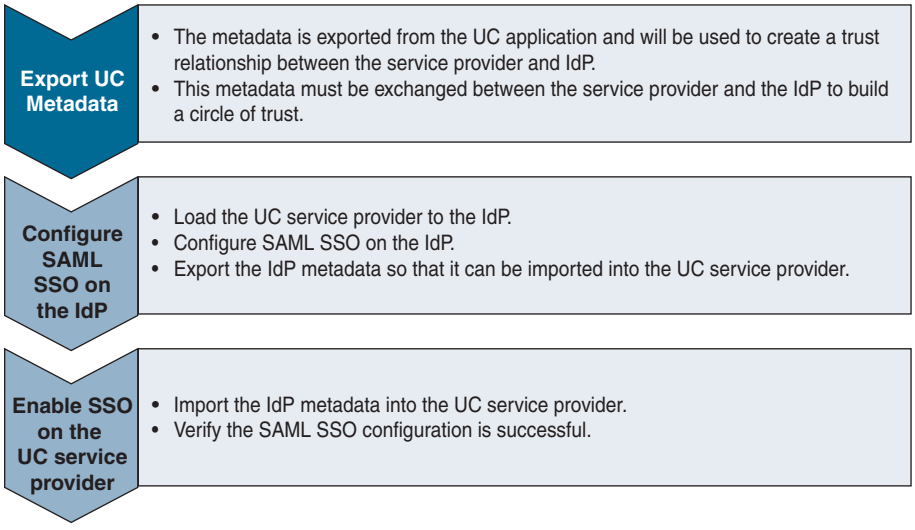


Figure 6-17 *SAML SSO Configuration Steps*

https://t.me/learningnets
@cisco_in_persian_channel

When you're configuring SAML SSO, Cisco recommends that you use the cluster-wide option to consolidate the metadata that is sent to the IdP. A prerequisite for using the clusterwide option for most of the UC applications is that it requires the use of Multiserver (SAN) Tomcat certificates. Using the clusterwide option serves two purposes. The first is to reduce the number of signed X.509 certificates that need to be created, which reduces costs and maintenance requirements. The second is to reduce the number of relying party trusts that must be configured on the IdP to let it know which service providers it will be communicating with and what authentication will take place. Each UC service provider needs a relying party trust configured for the services it provides configured within the IdP. This means that Cisco Unified CM cannot use the relying party trust for Cisco Unity Connection.

A caveat to enabling SAML SSO is that the NTP servers for the UC service provider and the IdP must be synched. The maximum allowed time difference between the IdP and the UC service provider is three seconds.

To test SSO while configuring it, you need an LDAP-synchronized end user with the standard CCM super user role assigned.

With these caveats acknowledged, the next step in the process is to collect the metadata from the IdP. For this example using Microsoft Active Directory Federation Services (ADFS), the metadata is located at http://*hostname*/federationmetadata/2007-06/federationmetadata.xml, where *hostname* is the FQDN of the IdP server. The simple XML file that is downloaded can be viewed in a browser if needed. Figure 6-18 shows some sample Microsoft ADFS metadata.

Figure 6-18 *IdP Federation Metadata*

The next step is to log in to **Cisco Unified CM Administration GUI > System > SAML Single Sign-On** (see Figure 6-19). When prompted, select **Cluster Wide for SSO Mode** and **Use Tomcat Certificate** in the Certificate section followed by **Enable SAML SSO**. Two things to consider here are the SSO mode to use and the certificates used. You can select the option for SSO mode to use "Per Node," but if it is used, each node needs to

https://t.me/learningnets

@cisco_in_persian_channel

have a corresponding configuration in the IdP. Using the Cluster Wide option allows the cluster as a whole to be configured in the IdP. The second consideration is the certificate used for the agreement. The option to use the ITLRecovery certificate is possible because the certificate is used only between the IdP and the SP. We chose to use the Tomcat certificate for ease of management, and the impact on endpoints is minimal in the event it needs to be regenerated.

Figure 6-19 *Enabling SAML SSO*

When prompted that existing web server connections will be restarted, you select **Continue**. A prompt to import the IdP Metadata trust file into the UC application is then displayed. At this point, you upload the IdP metadata file that was previously downloaded from the IdP. If the import fails, you need to verify the correct file was downloaded from the IdP. A successful import is shown in Figure 6-20.

Figure 6-20 *IdP Metadata File Import*

The next step is to download the UC service provider metadata file and prepare to upload it to the IdP, as shown in Figure 6-21.

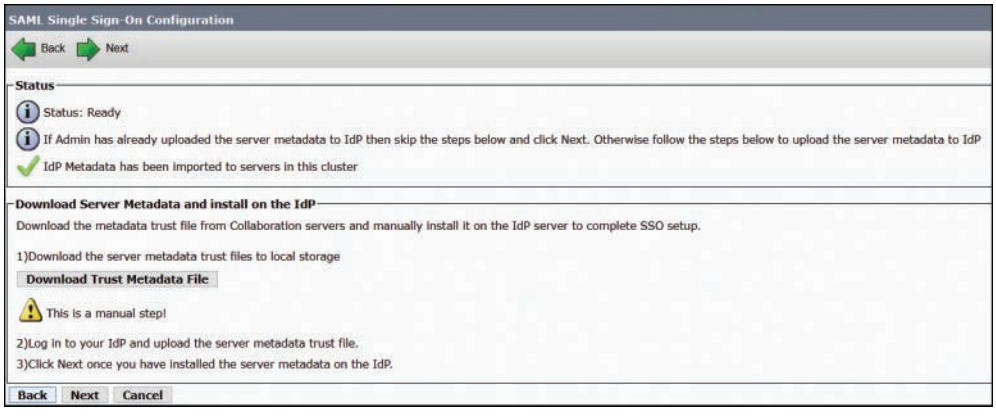


Figure 6-21 *UC Service Metadata File Export*

https://t.me/learningnets
@cisco_in_persian_channel

The next steps in the configuration process take place on the ADFS server so that the relying party trusts and claim rules can be completed. The relying party trusts specify the service providers that will communicate with the IdP. Part of configuring the relying party trusts is to import the SP metadata. The claim rules are used to specify how the authentication will take place.

At this point, you open the ADFS Management window and select **Add a Trusted Relying Party**. The following steps walk you through specifying the UC Service metadata file (see Figure 6-22).

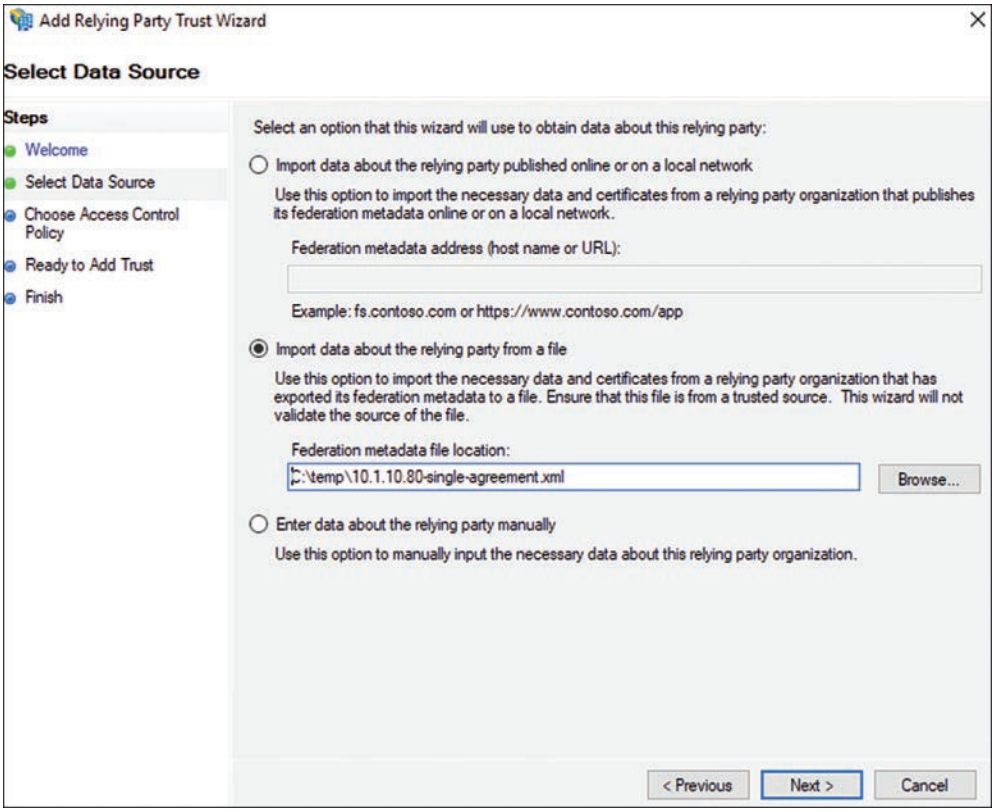


Figure 6-22 *UC Service Provider Metadata File Import*

For the Access Control Policy, you select **Permit Everyone** to grant access to everyone, as shown in Figure 6-23.

https://t.me/learningnets

@cisco_in_persian_channel

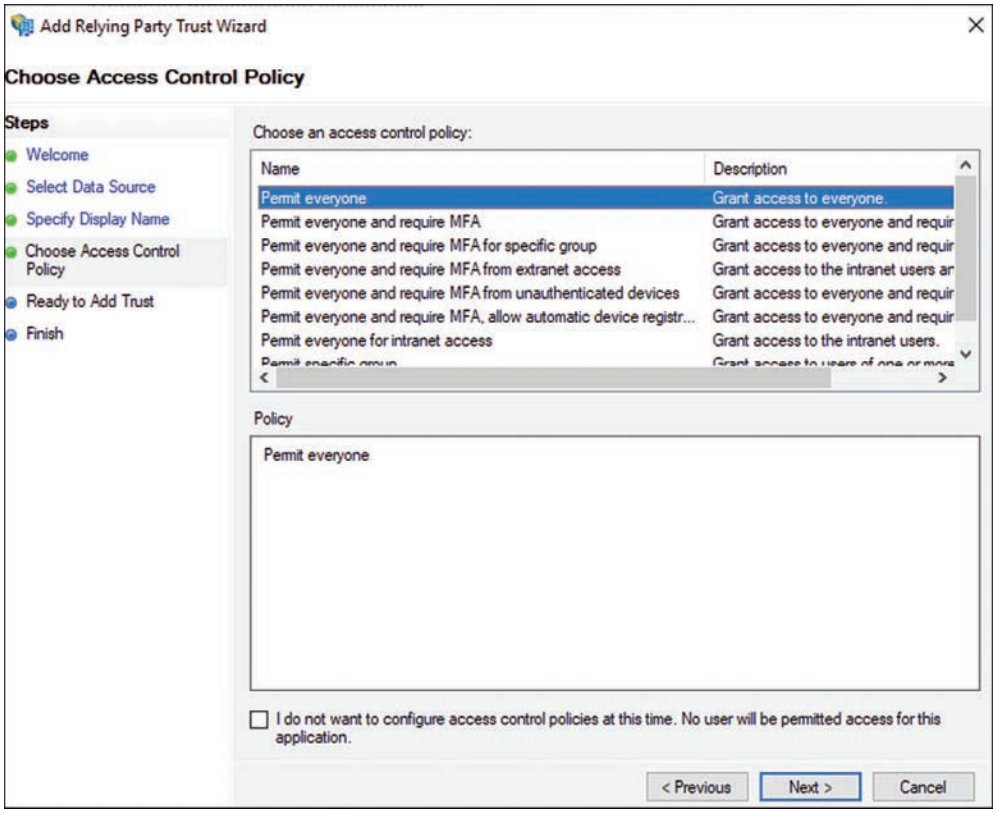


Figure 6-23 *Access Control Policy*

Next, you create two claim rules (see Figure 6-24). One is for the sAMAccountName that is being used as the user ID in the LDAP Synchronization, and the second is a custom claim rule. These claim rules are used to specify how the authentication will take place when the user attempts to authenticate. In this example, the users will use their sAMAccountName when they attempt to log in. The custom claim rule specifies how the connection between the IdP and SP should take place.

The second rule that needs to be configured is a custom claim rule. You can download this rule from the Cisco website. See AD FS Version 2.0 Setup for SAML SSO Configuration Example at www.cisco.com/c/en/us/support/docs/unified-communications/unified-communications-manager-callmanager/118771-configure-samlsso-00.html.

https://t.me/learningnets

@cisco_in_persian_channel

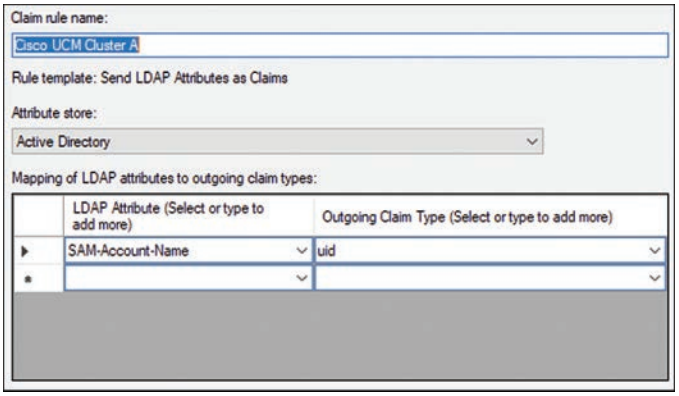


Figure 6-24 *AD FS Claim Rule - LDAP*

You need to modify this rule to include the existing ADFS Service Name and the CUCM Entity ID. You can find the first in the exported ADFS Federation metadata file and the second in the exported Cisco Unified CM metadata file:

```
c:[Type == "http://schemas.microsoft.com/ws/2008/06/identity/claims/
windowsaccountname"]
 => issue(Type = "http://schemas.xmlsoap.org/ws/2005/05/identity/
claims/nameidentifier", Issuer = c.Issuer, OriginalIssuer =
c.OriginalIssuer, Value = c.Value, ValueType = c.ValueType, Properties
["http://schemas.xmlsoap.org/ws/2005/05/identity/claimproperties/
format"] = "urn:oasis:names:tc:SAML:2.0:nameid-format:transient",
Properties["http://schemas.xmlsoap.org/ws/2005/05/identity/
claimproperties/namequalifier"] = "http://<AD_FS_SERVICE_NAME>/adfs/
com/adfs/service/trust", Properties["http://schemas.xmlsoap.org/
ws/2005/05/identity/claimproperties/spnamequalifier"] =
"<CUCM_ENTITY_ID>");
```

After creating both rules, navigate back to the Cisco Unified CM SAML SSO GUI. You need at least one LDAP synchronized user with the standard CCM super user role configured before you can continue.

Note If the UC application is currently running with FIPS enabled, you need to run an additional command against the ADFS server where the relying party trust is configured. From Powershell, you run the following command (replacing "Cisco Unified CM Cluster A" with the name of the relying party trust you configured):

```
set-ADFSRelyingPartyTrust -TargetName "Cisco Unified CM Cluster A"
-EncryptClaims $False
```

This command sends the requests unencrypted but does not remove the signing of the requests.

Next, you need to select a configured user that is a Unified CM administrator and then select **Run SSO Test** (see Figure 6-25). A new window opens asking you to log in.

https://t.me/learningnets

@cisco_in_persian_channel

Figure 6-25 *Launching SSO Test Page*

After the test completes successfully, you can enable RTMT and the Platform services so that they are integrated with SSO. You do this in Unified CM by setting the enterprise parameter **Use SSO for RTMT** to **True**.

Synching End Users Between Unity Connection and Unified CM Using Universal PIN

Keeping in line with simplifying the logon process for users by using consistent credentials across the different UC appliances is the concept of a universal PIN. Starting with Cisco Unified CM 11.5.1 and Cisco Unity Connection 11.5.1, system administrators can enable a common PIN for end users that can be synchronized between the two systems. When this PIN is configured, end users who use UC features like Meet-Me conferences, Extension Mobility, and voicemail can use a common PIN. From a security perspective, a fundamental reason to do this is to reduce the number of credentials that users have to remember. The more complex a credential is, the more likely users are to write it down, especially if it is not used very often and password fatigue becomes a concern.

When you're enabling a common PIN between the systems, enabling two specific features is recommended. The first is to enable the common PIN feature; the second is to follow up by enabling the PIN Credential phone service. This service allows end users to change their existing PINs from the IP endpoint without having to call in to a help desk or log in to the Cisco Unified Self Care Portal.

What are the requirements to enable a common PIN between Cisco Unified CM and Cisco Unity Connection? Chapter 7, "Encrypting Media and Signaling," provides a more in-depth explanation on how a chain of trust and trust stores work. The intent here is to explain the relationship between the trusted certificates needed to enable the universal PIN feature. For a universal PIN, the Tomcat certificates should be trusted between the two systems. This can be accomplished by uploading the Tomcat certificates from Cisco Unity Connection to the Cisco Unified CM Tomcat-Trust or vice versa with the Tomcat certificates for Cisco Unified CM loaded to the Cisco Unity Connection Tomcat-Trust. The easiest and recommended method, however, is to use Tomcat certificates signed by a certificate authority that is trusted by both UC systems. This requires the root and

https://t.me/learningnets
@cisco_in_persian_channel

intermediate CA certificates be uploaded to the Tomcat-Trust stores for the systems being configured.

The AXL service must be activated and started on the Cisco Unified CM node that will be defined as the primary AXL server within the Cisco Unity Connection Telephony Integration. TCP port 8443 should be open between the two UC systems.

Two users need to be created: one in Cisco Unified CM for Cisco Unity Connection to update PINs using AXL and a second account in Cisco Unity Connection for Cisco Unified CM to update the voicemail subscribers.

Starting with Cisco Unified CM, you create an application user with the Standard AXL API Access role (see Figure 6-26).

Figure 6-26 *Common PIN Application User Role*

To configure this user, on the Cisco Unity Connection, you navigate to **Telephony Integrations > Phone System**. Under the AXL Servers Settings, you select **Enable End User PIN Synchronization for Primary AXL Server** (see Figure 6-27). Cisco recommends not using the default administrator account within Cisco Unified CM as the AXL user. Although it has the roles necessary to enable the feature, the roles on the account should be limited to only those that it needs. The user specified here is the same user that was previously created within Unified CM with the proper permissions to allow the feature to work while still abiding by the principle of least privilege.

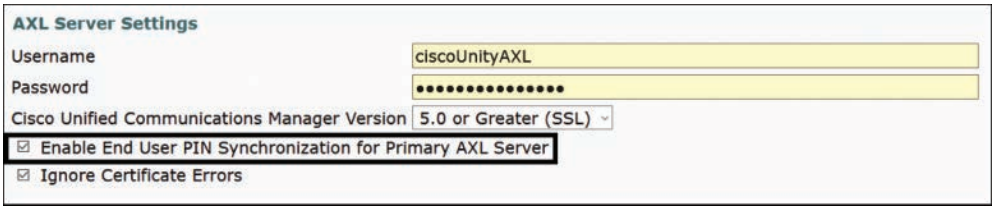


Figure 6-27 *Cisco Unity Connection AXL Server Settings*

Note Multiple AXL servers can be specified for each Telephony Integration, with the lowest priority being preferred.

https://t.me/learningnets
@cisco_in_persian_channel

Now in Cisco Unity Connection, you configure a user without a mailbox based on the administrator template or ideally a custom template that has the Help Desk Administrator role assigned (see Figure 6-28). This user should have the same name and password as the account just created in Cisco Unified CM.

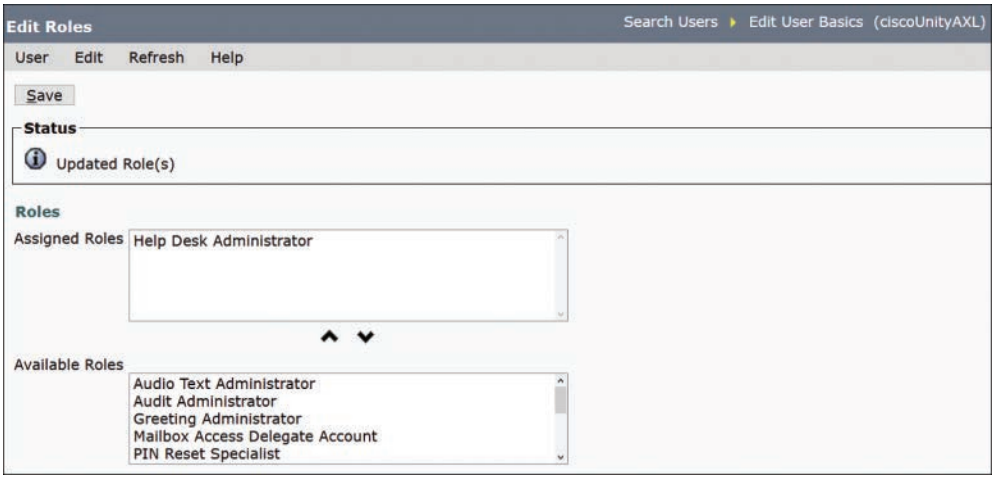


Figure 6-28 *Cisco Unity Connection AXL User Role*

The same user will be configured in Cisco Unified CM under **System > Application Server** as the AXL user for the Cisco Unity Connection AXL user. This user will have the **Enable End User Pin Synchronization** option enabled, as shown in Figure 6-29.

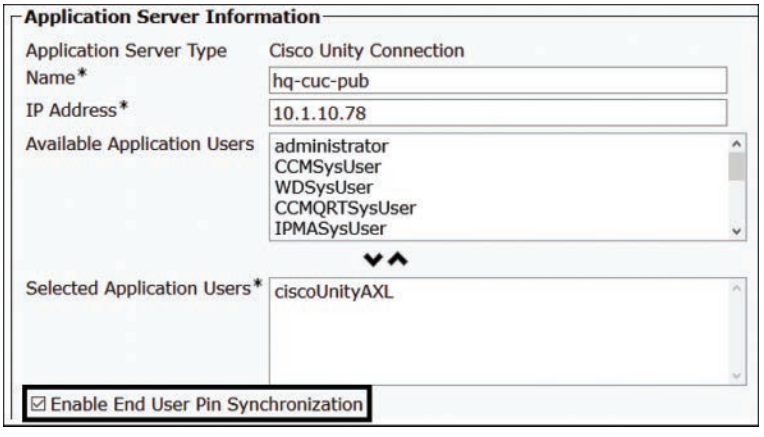


Figure 6-29 *Cisco Unified CM Application Server Settings*

Now when a user's PIN is changed in either Cisco Unified CM or Cisco Unity Connection, that user is able to use that new PIN on either system.

https://t.me/learningnets
@cisco_in_persian_channel

Credential Change Service

Deploying the Change Credential phone service enables end users to update their PINs using a registered IP phone. This service is useful in that it can enable end users to change their PINs when they do not have access to the Cisco Unified CM Self Care Portal and need to update the PINs for features like Meet-Me conferencing and Extension Mobility. If PIN synchronization is enabled, this service also allows end users to change their voicemail PINs using the display on the phone rather than Cisco Personal Communications Assistant (Cisco PCA) or the telephone user interface (TUI).

The Change Credential phone service is configured via **Cisco Unified CM Administration > Device > Device Settings > Phone Services**. To configure the service, you need to specify the information shown in Table 6-11.

Table 6-11 *Change Credential Phone Service*

Service Information	
Service Name*	Meaningful service name
Service Description	Short description of what the service does
Service URL*	http://server:8080/changecredential/ChangeCredentialServlet?device=#DEVICENAME#
Secured Service URL	https://server:8443/changecredential/ChangeCredentialServlet?device=#DEVICENAME#
Service Category*	XML Service
Service Type*	Standard IP Phone Service
Enable	Selected

Note Using the IP address of the server instead of the FQDN is recommended unless the endpoints are configured to support DNS.

Phone services can be made available using two methods. One option is to make the services available at the enterprise level. When phone services are made available at the enterprise level, all IP phones can access the services. For most services such as voicemail and directories, advertising the service at the enterprise level makes sense. However, for services like the Change Credential service and Extension Mobility, it is advisable to deploy them only to endpoints where they are required. This brings up the second method of advertising the phone service—manual subscription. This method simply means that endpoints are not automatically subscribed to the phone service. This keeps endpoints in areas like lobbies and elevators from being able to use services they typically do not need access to.

When the Change Credential service is selected, the end user is prompted to enter an existing user ID, current PIN, and then a new PIN that they want to use, as shown in Figure 6-30.

https://t.me/learningnets

@cisco_in_persian_channel

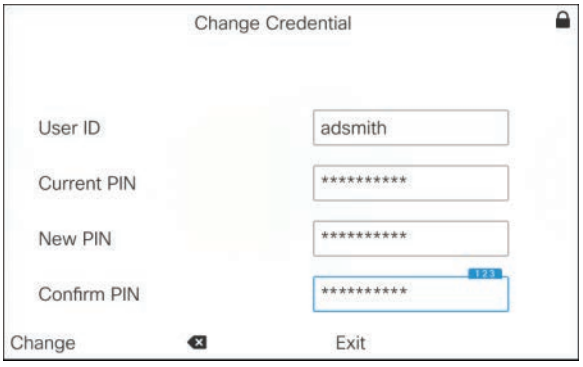


Figure 6-30 *Change Credential Service*

When the change is successful, the user is notified (see Figure 6-31).

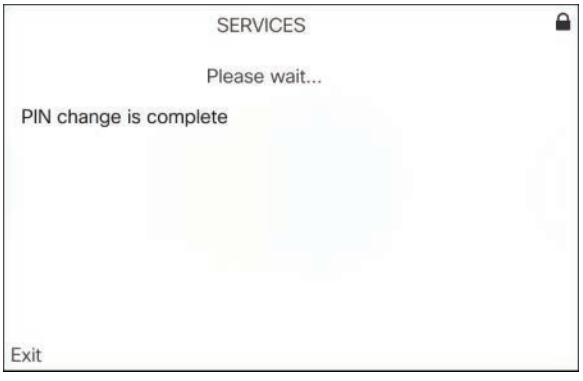


Figure 6-31 *Change Credential Service Successful*

Locking Down the GUI

To this point, we have covered the process to secure user access to the UC systems. Now it's time to speak to some basic options available to secure the UC web pages. The easiest way to secure the UC web pages outside of restricting network traffic is to configure a session timeout for inactive users logged in and a simple login banner.

Screen Timeout

By default, the Cisco UC applications time-out inactive logged-in users every 30 minutes. You can verify this time by using the **show webapp session timeout** command. Sample output from the command is provided in Figure 6-32.

https://t.me/learningnets
@cisco_in_persian_channel

```
admin:show webapp session timeout

The current session-timeout used for web sessions and applications is 30 minutes.

admin:
```

Figure 6-32 *Using show webapp session timeout*

If you need to change the timeout from the default of 30 minutes, you can use the **set webapp session timeout <value>** command, where **<value>** can be used to change the number of minutes allowed to elapse before the user is logged out, as shown in Figure 6-33. When the command is initiated, you are prompted to restart the Cisco Tomcat service. If you do not restart the Cisco Tomcat service, the new timeout does not take effect and the **show webapp session timeout** command does not reflect the new timeout.

```
admin:set webapp session timeout 15

Continuing with this operation will set the session-timeout for web sessions to 15 minutes
after the 'Cisco Tomcat' service has been restarted or after the server has been rebooted.
Continue (y/n)?y
Tomcat session-timeout updated to 15 minutes.

The 'Cisco Tomcat' service needs to be restarted for the changes to take effect immediately.
This will disconnect active web sessions.
Continue (y/n)?y
Do not press Ctrl+C while the service is restarting. If the service has not restarted
properly, execute the same command again.
Service Manager is running
Cisco Tomcat[STOPPING]
Cisco Tomcat[STOPPING]
Cisco Tomcat[STOPPING]
Cisco Tomcat[STOPPING]
Cisco Tomcat[STARTING]
Cisco Tomcat[STARTING]
Cisco Tomcat[STARTED]

The 'Cisco Tomcat' service was restarted successfully.  New web sessions will timeout after 15
minutes.
```

Figure 6-33 *Using set webapp session timeout*

Login Banner

Implementing a login banner is a quick task that should be completed when initially implementing any computer system. A login banner announces to those wishing to access a system these three main areas of concern:

- Acceptable use of the system (acceptable use policy, or AUP) defines who is authorized to access the system and what functions or actions they can perform while connected.
- The banner informs users that the system is being monitored or privacy is not ensured. Users connecting to the system must understand that their actions are

https://t.me/learningnets

@cisco_in_persian_channel

monitored. This monitoring can consist of the commands run while connected, time frames that they are connected, and other actions taken while connected to the system. Although it might not seem like much, it is an important legal aspect to inform the user connecting that actions while connected to the system have no inherent privacy.

- Accepting the login banner provides implicit consent to the AUP and waves privacy rights while accessing the system. The last line of the login banner should inform the user that continuing with the login process is implied consent to the AUP and privacy expectations.

Because legal verbiage can vary by location, it is important to consult your legal department on what is allowed and what is not allowed.

Here are some simple dos and don'ts for login banners:

Dos:

- Inform the user connecting that only authorized parties should proceed.
- Outline the acceptable use requirements while connected to the system.
- Explain that the systems and communications are monitored.
- Advise that by continuing the login process, the party connecting accepts the terms.

Don'ts:

- Do not welcome the user.
- Do not provide information about the system they are connecting to. This information can be used in fingerprinting for an attack.
- Do not provide any information that is not relevant to the AUP or logon process.

To enable the login banner on the core Cisco UC applications, you log in to the **OS Administration webpage > Software Upgrades > Customized Logon Message** and upload a text file. It is important to select the **Require User Acknowledgment** option. When enabled, this last option requires that the login banner be acknowledged (implying consent) before the user is able to log in to the system. Uploading the login banner allows the banner to be displayed on the GUI when logging in as well as the CLI (see Figure 6-34).

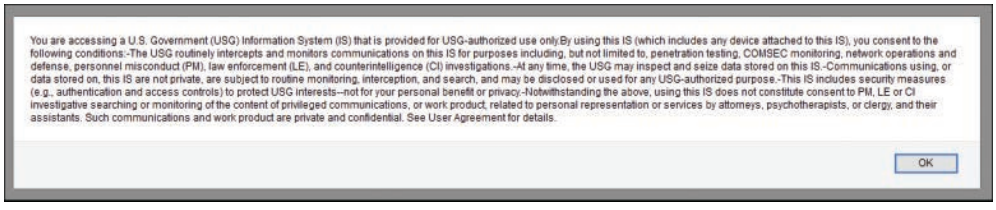


Figure 6-34 *Login Banner*

https://t.me/learningnets
@cisco_in_persian_channel

Enabling System Monitoring Using SNMP and Syslog

System monitoring is important to system security in that it lets system administrators know how the system is performing. With monitoring, administrators can identify unsuccessful login attempts, configuration changes, and resource utilization spikes.

Configurating and Using SNMP for System Monitoring

Simple Network Management Protocol (SNMP) is an application layer protocol that facilitates the exchange of management information among network devices. SNMP lets administrators remotely monitor and maintain network performance, identify and resolve network problems, and provide a baseline for network health. Two concepts related to SNMP should be defined:

- **Management Information Base (MIB):** MIBs are a collection of defined aspects of a managed object within a device that is being managed. Every device managed keeps a table of sorts, with the values for each definition contained in the MIB. Additionally, MIBs use object identifiers (OIDs) to uniquely define the managed objects within an MIB. Typically, the OID is represented in numeric form like 1.3.6.1.4.1.9.9.156.1.9.2, which is the OID for ccmPhoneFailedAlarmInterval.

- **Traps:** An SNMP agent sends traps to an SNMP Manager to indicate some event occurred. These SNMP traps are simply notifications.

The Cisco Unified Serviceability GUI is used to configure the SNMP settings, such as community strings, users, and notification destinations for SNMP versions 1, 2c, and 3. Configured settings apply to the local node; however, if the system supports clustering, you can apply the configuration settings to all servers in the cluster by selecting the **Apply to All Nodes** option in the SNMP configuration window.

SNMP uses these three main components:

- A managed device is configured on the network with an SNMP agent. Managed devices collect and store system information and events that are then made available using SNMP.

- An SNMP agent is a software module or service on the managed device that can access the local management information and convert it into a format that SNMP can utilize. Cisco UC applications have a master agent and subagent components to support SNMP.

  The master agent is the protocol engine; it performs the authentication, authorization, access control, and the privacy functions related to SNMP requests. The master agent listens on port 161, forwards SNMP packets for vendor MIBs, and connects and disconnects the subagents as they complete their requisite tasks.

  The SNMP subagent interacts with the local host to send trap and information messages to the SNMP master agent; then the master agent communicates directly with the SNMP notification destination.

https://t.me/learningnets

@cisco_in_persian_channel

- Network Management System (NMS) is an application that performs the bulk processing of collected information and system resources required for network management. NMS allows system administrators to monitor and manage controlled devices.

Cisco UC applications, on the whole, support SNMP versions 1, 2c, and 3. Using SNMP version 3 is recommended because this version provides advanced security features that the other two versions do not. SNMP versions 1 and 2c function in a similar fashion with the Structure of Management Information (SMI) operating over UDP and IP. Version 2c provides an optimized feature set over version 1. Neither of these SNMP versions offers security features such as authentication (outside of the community string), privacy, and authorization.

A quick packet capture from the Cisco Unified CM can show that the community string is sent in the clear (see Figure 6-35).

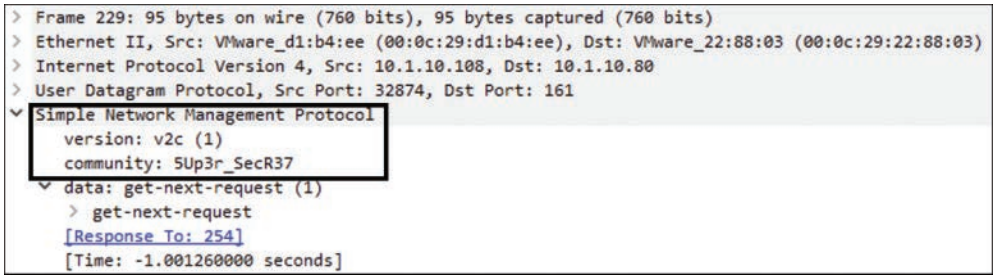


Figure 6-35 *SNMPv2c Packet Capture*

SNMP version 3 does support these security features, which is why it is recommended. SNMP version 3 supports authentication, authorization, and privacy using usernames, passwords, and SHA/AES128. Starting with Cisco Unified CM version 12.5(1) SU1 and higher, MD5 or DES encryption options are not supported. In these later versions, you can only select SHA/AES. Cisco Unified CM versions prior to 12.5(1) SU1 allowed the use of MD5 or DES unless the system was configured to support FIPS140-2 or Enhanced Security Mode. The current 12.5 versions of Cisco Unity Connection, IM&P, and Cisco Emergency Responder follow the direction of Cisco Unified CM and only support the encryption options of SHA or AES128.

A packet capture from the Cisco Unified CM shows that while the username is sent in plaintext, the rest of the request is sent encrypted (see Figure 6-36).

To take this a step further, the packet capture can be decrypted using Wireshark. Figure 6-37 shows the decrypted response to a request for the RemoteEngineID.

Note The decryption in this example using Wireshark is possible because the secrets are already known. Since SNMP is a User Datagram Protocol (UDP) that uses the secrets for encryption and authentication, the decryption process is simple.

https://t.me/learningnets

@cisco_in_persian_channel

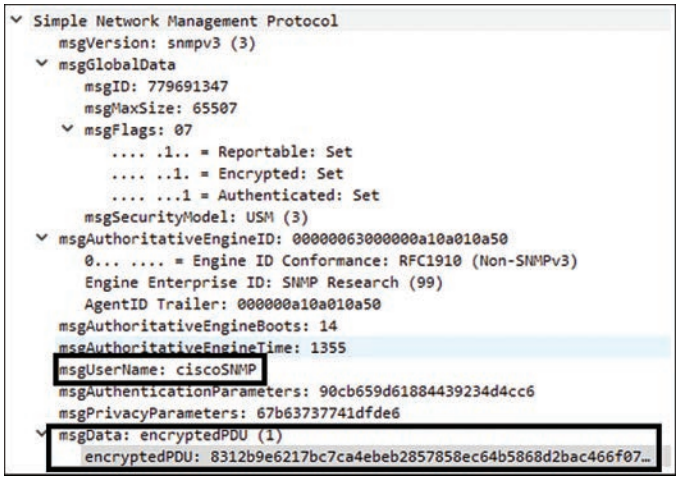


Figure 6-36 *SNMPv3 Packet Capture Encrypted*

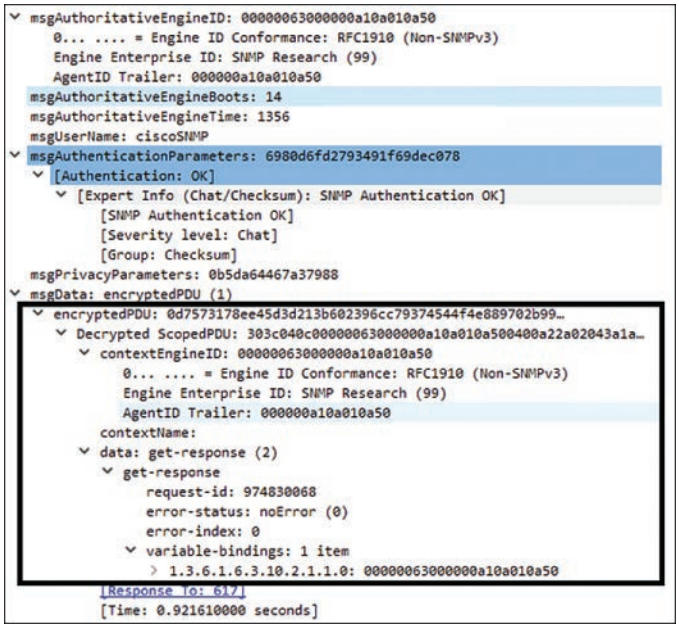


Figure 6-37 *SNMPv3 Packet Capture Decrypted*

For more information on the features, functions, and MIBs for the Cisco UC applications, see www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/admin/12_5_1SU1/cucm_b_serviceability-admin-guide-1251su1/cucm_b_serviceability-admin-guide-1251su1_chapter_01000.html.

https://t.me/learningnets

@cisco_in_persian_channel

The process to enable SNMP versions 1, 2c, and 3 first starts with deciding which version you will run. The discussion here covers configuring SNMPv3 because SNMPv2c configuration is already considered commonplace. Out of the box, Cisco UC applications do not have SNMP enabled. The process to configure it starts with first activating the SNMP service (SNMP Master Agent) in the Cisco Unified Serviceability GUI on each node that will run the service.

Next, within the Serviceability GUI, you go to **SNMP > SNMPv3 > User** to configure the user that will be used for authentication and privacy. For this configuration to be successful, you need the SNMPv3 user settings information shown in Table 6-12.

Table 6-12 *SNMPv3 User Settings*

SNMPv3 User Settings	
Server	This is the server that is being configured to support SNMP.
User Name	In this field, you enter the name of the user for which you want to provide access. The name can contain up to 32 characters and can contain any combination of alphanumeric characters, hyphens (-), and underscore characters (_).
Authentication Required	Select the checkbox and populate the Password boxes. SHA is the default option available starting with the Cisco Unified CM version 12.5(1)SU2 or if FIPS is enabled. The minimum password length is 8 characters.
Privacy Required	Select the checkbox and populate the Password boxes. AES128 is the default option available starting with the Cisco Unified CM version 12.5(1)SU2 or if FIPS is enabled. The minimum password length is 8 characters.
Accept SNMP Packets from Any Host	This is an acceptable option but not recommended. When it is selected, any host can communicate using SNMP with the system. Care should be taken when this option is used.
Accept SNMP Packets Only from These Hosts	This is the recommended option. When it is selected, only the hosts specified can communicate with the system using SNMP. Some practical examples of systems that would be defined here are ■ NMS servers that query the servers using SNMP to monitor them ■ Cisco Emergency Responder querying the system for information on identified endpoints

https://t.me/learningnets

@cisco_in_persian_channel

SNMPv3 User Settings	
Access Privileges	Choose one of the following options for the access level: ReadOnly: You can only read the values of MIB objects. ReadWrite: You can read and write the values of MIB objects. ReadWriteNotify: You can read and write the values of MIB objects. The server is also able to send MIB object values for a trap and inform messages to the defined hosts. NotifyOnly: You can only send MIB object values for trap and inform messages. ReadNotifyOnly: You can read values of MIB objects and also send the values for trap and inform messages. None: You cannot read, write, or send trap information.
Apply to All Nodes	

Now that the SNMP user is configured, you need to configure the Notification Destination (Traps). For this, two types can be configured. The first type is a simple Trap that is used to send information to the NMS without waiting for any acknowledgment. The second type is Inform. When Inform is used, the system uses the Remote SNMP Engine ID when the data is sent and expects an acknowledgment that it was received.

If the Remote SNMP Engine ID of the NMS is unknown, you can use the **utils snmp walk 3** command at the CLI to collect it, as shown in Figure 6-38. You do need to know the SNMPv3 user and passwords that will be used to connect.

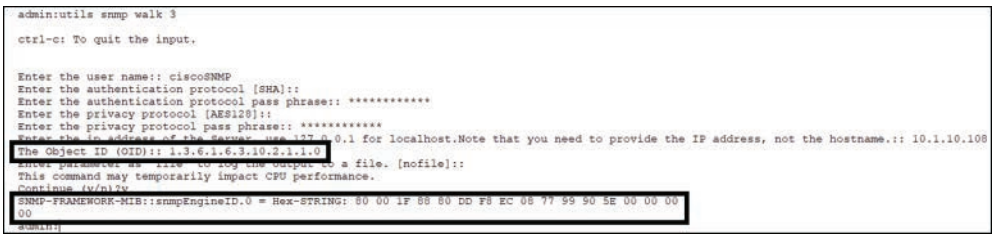


Figure 6-38 *Using utils snmp walk*

Regardless of whether you are running any of the versions of SNMP, it is important to provide the contact information and location. This information can be pulled in the event of an incident to identify who needs to be contacted and where the node is located. To configure the system contact and location, navigate to **Serviceability GUI > SNMP > SystemGroup > MIB2 System Group**. Just like the rest of the options for SNMP, you can configure the location and contact on one node and have it replicate to the rest of the nodes in the cluster.

https://t.me/learningnets
@cisco_in_persian_channel

Defining the Alerting Types and Configuring Logging

There are two methods for generating alerts into the logging facilities with the Cisco UC applications. The first is through alarms, which provide runtime information on the current status and state of the system. The second method is with audit logs, which provide logging capabilities for system configuration changes. It is important that alarms and auditing are configured so that events are logged and appropriately acted upon. Some good examples of alarms to alert on are a spike in devices unregistering or a backup failure. As with alarms, logging and alerting on auditing events is important because they provide a record of configuration changes and unsuccessful logins. One potentially overlooked aspect of alarms and logging in general is that they should be offloaded to a centralized location. One of the benefits of offloading them is the ability to more easily correlate data across multiple devices to identify the root cause of an issue. A second benefit is as simple as being able to store more of the data. It is quite possible that once a problem has been identified, the logs from the time the problem initially began have already been overwritten.

Alarms

Alarms provide information on the current health of the system. This information helps you identify issues and aids in troubleshooting the events as they occur. The information provided by an alarm contains a brief explanation of the alarm with recommended remediation tasks. Additionally, to aid in identification of where the event occurred, the alarm provides the application name, machine name, and so on to aid in troubleshooting.

Alarms can be configured to send notifications to multiple locations, and each location can have a different alarm level configured. The system administrator can configure the forwarding of alarms to the Syslog Viewer (local syslog), Syslog file (remote syslog), SDL trace log file (for Cisco CallManager and CTIManager services only), or to all of them.

Note When a remote syslog service is configured, it cannot be another Cisco UC application, such as Cisco Unified CM or Cisco Unity Connection.

The Cisco Unified Real-Time Reporting Tool is best suited to collect and view alarms that are sent to the SDL trace log or local syslog. The CLI can be used to view some of the log files and collect logs, but it is usually not the best method for reviewing log files.

You should configure alarms for services, such as Cisco CallManager, in Cisco Unified Serviceability. Then, you can configure where you want the logs sent, such as Syslog Viewer (local syslog). Doing so enables you to do the following:

- Configure alarms for services on a specific server or on all servers
- Configure different remote syslog servers for the configured services or servers
- Specify different alarm event-level settings for different destinations

https://t.me/learningnets

@cisco_in_persian_channel

Cisco Syslog Agent enterprise parameters in the different Cisco UC applications allow you to forward all alarms that meet or exceed the configured threshold to a remote syslog server with two settings: remote syslog server name and syslog severity. To access these Cisco Syslog Agent parameters, go to the applicable window for your configuration shown in Table 6-13.

Table 6-13 *Syslog Agent Parameters*

Cisco UC Application	Location of Cisco Syslog Agent Parameters
Unified Communications Manager	In Cisco Unified Communications Manager Administration, choose **System > Enterprise Parameters**.
Cisco Unity Connection	In Cisco Unity Connection Administration, choose **System Setting > Enterprise Parameters**.
Cisco IM and Presence	In Cisco Unified Communications Manager IM and Presence Administration, choose **System > Enterprise Parameters**.

Note Cisco Emergency Responder does not support RTMT, so you can view logs from the Cisco ER Serviceability GUI.

One item to note is that enabling the alerting of alarms at different locations can result in multiple alerts being sent for the same event. For example, if the Cisco Syslog Agent alarm enterprise parameters and service alarms under Cisco Unified Serviceability are configured, this could result in duplication of alarms being sent.

The event-level/severity settings provide a filtering mechanism for the alarms and messages that the system collects. This setting helps prevent the syslog and trace files from becoming overloaded. The system forwards only alarms and messages that exceed the configured threshold.

Table 6-14 describes the alarm severity.

Table 6-14 *UC Alarm Severity Levels*

Alarm Event Level	Alarm Description
Emergency	Designates the system as unusable
Alert	Indicates that immediate action is needed
Critical	Detects a critical condition
Error	Signifies that an error condition exists
Warning	Indicates that a warning condition is detected
Notice	Designates a normal but significant condition
Informational	Designates information messages only
Debug	Designates detailed event information that Cisco Technical Assistance Center engineers use for debugging

https://t.me/learningnets

@cisco_in_persian_channel

When enabling the alarm levels and specifying where the alarms should be directed, refer to the default settings listed in Table 6-15.

Table 6-15 *Default Alarm Locations*

	Local Syslogs		Remote Syslogs	SDI Trace	SDL Trace
Enable Alarm	Checked		Unchecked	Checked	Checked
Alarm Event Level	Error		Disabled	Error	Error
Exclude End Point Alarms	**Local Syslog**	**Alternate Syslog**	**Remote Syslog**	**Syslog Severity and Strangulate Alert**	**Syslog Traps**
Checked	No	Yes	No	No	No
Unchecked	No	Yes	Yes	Yes	Yes

Audit Logs

The system is configured for standard audit logging by default. With standard audit logging, configuration changes to the system get logged in separate log files for auditing. To access the Cisco Audit Event Service, you navigate to **Cisco Unified Serviceability GUI > Control Center - Network Services.** This service monitors and logs configuration changes to the system by user or user interaction. By default, audit logs are configured to rotate.

Audit logs consist of these two parts:

- **Audit logging framework:** The framework comprises an API that uses an alarm library to write audit events into audit logs. An alarm catalog defined as GenericAlarmCatalog.xml is applied to these alarms. Because the different system components provide their own logging, the referenced GenericAlarmCatalog.xml is different from the alarm catalog available via Unified CM Serviceability.

  You can find the referenced alarm catalog in the respective UC applications Serviceability settings.

  The following example displays a Unified Communications Manager component alarm:

```
Date: 04/08/2020   22:25:46.250
UserID: ucadmin
ClientAddress: 10.1.10.152
Severity: Notice
EventType: GeneralConfigurationUpdate
ResourceAccessed: CUCMAdmin
EventStatus: Success
CompulsoryEvent: No
AuditCategory: AdministrativeEvent
```

https://t.me/learningnets

@cisco_in_persian_channel

```
ComponentID: Cisco CUCM Administration  CorrelationID :
AuditDetails: record in table applicationuserdevicemap with key
  field name = ciscoUnityAXL added
App ID: Cisco Tomcat
Cluster ID:
Node ID: hq-cucm-pub
```

- **Audit event logging:** An audit event represents any event that is required to be logged. The following audit event is provided for reference:

```
22:25:46.250 |LogMessage   UserID : administrator  ClientAddress :
10.1.10.152  Severity : 5  EventType : GeneralConfigurationUpdate
ResourceAccessed: CUnified CMAdmin  EventStatus : Success
CompulsoryEvent : No  AuditCategory : AdministrativeEvent
ComponentID : Cisco CUnified CM Administration  CorrelationID :
AuditDetails :  record in table applicationuserdevicemap with key
field name = ciscoUnityAXL added  App ID: Cisco Tomcat Cluster ID:
Node ID: hq-cUnified CM-pub
```

For a more detailed review of logged events that are captured with the audit logs, you can review the Cisco Unified Serviceability Administration Guide – Audit Logs for your release version.

Continuing with audit logging is the optional feature to enable detail audit logging. This feature logs additional configuration changes that are not stored in standard (default) audit logs. In addition to the standard information stored in standard audit logs, detailed audit logging includes configuration items that were added, updated, or deleted, including the modified values. Detailed audit logging is disabled by default. There are two types of detailed audit logging to consider. The first is the system audit logs that track the creation, modification, or deletion of Linux OS users, log tampering, and any changes to file or directory permissions. The system audit logs can only be enabled by using the **utils auditd** command at the CLI and must be enabled on a node-by-node basis. The second detailed audit logging is for application audit logging, which tracks and logs any configuration changes to the system that were made by a user or user interaction. To enable application audit logs, you can navigate to **Cisco Unified Serviceability GUI, Tools > Audit Log Configuration**. The logs can be replicated to the other nodes in a cluster.

After you enable the system audit log feature, you can collect, view, download, or delete logs using the Real-Time Monitoring Tool under Trace & Log Central. System audit logs are contained in a file named vos-audit.log. Other than using the RTMT to download and view the audit logs, you cannot change the settings for the events collected in the logs.

Cisco recommends removing the Standard Audit Users role from the UC Administrator accounts. By default, the application administrator configured during installation is the only user with this role. Cisco recommends creating a separate Auditor account that has access to the Standard Audit User role and then removing it from the standard UC administrator accounts. Only a user with an audit role can change the audit log settings.

https://t.me/learningnets

@cisco_in_persian_channel

Disaster Recovery Planning and Best Practices

An often-overlooked aspect of security is having a valid backup that is ready and available in the unfortunate event you need to restore a system. If you don't have a backup or the backup is not current, valuable time is lost while you try to recover the system. Cisco's implementation of the Disaster Recovery Framework (DRF) provides complete backup and restore functionality for the Cisco UC applications. Depending on the application, the whole UC cluster is backed up from a single location. In the case of Cisco Unity Connection, however, if you run the system in high availability mode, you must configure the backup on both nodes in the HA cluster. Either way, though, the backup and restore process provides a full-featured and complete process to safeguard the application in the event of a failure.

Ultimately, the key reason for a backup strategy is for disaster recovery: you hope you won't need it, but when you do, you're glad you have it. A good backup strategy depends on several things, and the key to a successful backup strategy is to ensure you have valid and current data. What defines this strategy varies from company to company based on the size of the implementation, uptime requirements, and even federal regulations that might require consideration. The standard backup tools built in to the UC applications provide for an administrator-defined number of backups to be retained. This allows for streamlined backups because the backups are offloaded to a remote FTP/SFTP server and rotated automatically.

Another part of a successful backup strategy is to not just have the backups but to validate them. What is accomplished by validating a backup? Restoring the backup validates the restore phase of a backup strategy. This serves two purposes: first, it familiarizes support personnel with the tasks involved with performing a backup. This lessens the angst that comes along with the pressure to restore services in a critical situation. The second purpose is that it validates that you have all the information needed to perform the restoration. An important piece of information needed during a restore is the cluster security passphrase. This passphrase is used during a backup to encrypt the data being backed up. During a restore process, if you do not know the proper passphrase, you are prompted to enter it. If you do not have it, you are not able to restore because the backups are encrypted using the cluster security passphrase.

The Cisco Disaster Recovery Service (DRS) has the following capabilities:

- A user interface for performing backup and restore tasks
- A distributed system architecture for performing backup and restore functions
- Scheduled backups
- Archived backups to a remote SFTP server

https://t.me/learningnets

@cisco_in_persian_channel

The Disaster Recovery System has two main services: Master Agent (MA) and Local Agent (LA).

- **Master Agent:** The system automatically starts the Master Agent service on each node of the cluster, but the Master Agent is functional only on the publisher/primary node (for Cisco Unified CM/IM&P). The Master Agent on the subscriber nodes does not perform any function.
- **Local Agent:** Each server has a Local Agent to perform backup and restore functions. Each node in a Cisco UC cluster, including the node that contains the Master Agent, must have its own Local Agent to perform backup and restore functions.

Summary

This chapter covered several different topics that relate to securing the Cisco UC application GUIs. This is accomplished through controlling access to the system with properly implemented user accounts and user account controls. We explained the differences between application users and end users along with the best practices for each type of account. Implementing user account controls through either the local credential policies or integrating with LDAP and proper configuration of user account groups reflects the best practices that should be used. We also covered implementing a common PIN and credential change service to show ways in which passwords and PIN management can be streamlined as UAC requirements increase complexity.

Lastly, we highlighted how monitoring, logging, and having a backup strategy can be used to track system health and security events. Monitoring and logging are the cornerstone for understanding the baseline health and activities of the UC systems. Logs provide the information needed so that you can identify when something out of the ordinary happens. Having a backup strategy and using the Cisco Disaster Recovery Framework enables you to prepare for the worst-case scenario when things don't go as planned.

Additional Resources

Security Guide for Cisco Unified Communications Manager, Release 12.5(1)SU3 – Credential Policies https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/security/12_5_1SU3/cucm_b_security_guide_1251SU3/cucm_m_credential-policies.html

Cisco Collaboration System 12.x Solution Reference Network Designs (SRND) – Directory Integration and Identity Management https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/srnd/collab12/collab12/directry.html#59256

Security Guide for Cisco Unified Communications Manager, Release 12.5(1)SU3 – Identity Management https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/security/12_5_1SU3/cucm_b_security_guide_1251SU3/cucm_m_identity-management.html

https://t.me/learningnets

@cisco_in_persian_channel

System Configuration Guide for Cisco Unified Communications Manager, Release 12.5(1) SU3 Configure Single Sign-On https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/admin/12_5_1SU3/systemConfig/cucm_b_system-configuration-guide-1251su3/cucm_m_configure-single-sign-on.html

Cisco Collaboration System 12.x Solution Reference Network Designs (SRND) – Network Management https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/srnd/collab12/collab12/netmgmt.html?bookSearch=true

Administration Guide for Cisco Unified Communications Manager, Release 12.5(1)SU3 - https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/admin/12_5_1SU3/adminGd/cucm_b_administration-guide-1251su3/cucm_b_test-adminguide_chapter_010110.html

https://t.me/learningnets

@cisco_in_persian_channel

This page intentionally left blank

https://t.me/learningnets

@cisco_in_persian_channel

Chapter 7

Encrypting Media and Signaling

This chapter steps through the process to enable secure signaling and media. A structured approach to enabling the signaling and media is recommended to reduce the chances of causing an outage. This structured approach is made up of six main tasks:

- Apply an encryption license or enable export restrictions on Smart Account.
- Configure third-party signed certificates (if required).
- Set the Cisco Unified Communications Manager cluster security mode (mixed mode) to enabled.
- Deploy Locally Significant Certificates to endpoints that do not support SIP OAuth.
- Enable and configure SIP OAuth.
- Apply secure profiles to endpoints, DSP resources, trunks, and gateways.

To explain this process, we start with a short discussion on what licensing is required and then follow up with what certificates are used to secure the signaling and media. Next, we explain the concept of security by default, including how it has evolved since Cisco implemented it with Cisco Unified CM 8.x. Next, we explain what it means to enable mixed mode and when to use it and what features it enables. Lastly, to wrap up the discussion, we cover the process to update endpoints and SIP trunks to utilize a secure connection.

Over the past couple of chapters Anthony has been securing the Unified Communications infrastructure one aspect at a time. The time has come to secure the signaling and media. Anthony has done some research and read the configuration guides, but working with the required certificates is daunting. After doing his homework on the subject and speaking with management, he's gotten the approval to move forward.

https://t.me/learningnets

@cisco_in_persian_channel

Questions that you should ask:

1. Do the UC applications currently deployed support strong encryption and SRTP?
2. Is there a need to use a Public certificate authority to sign digital certificates needed or will an internal certificate authority suffice?
3. Is it necessary to secure the signaling and encrypt the media for communication or is it acceptable to only secure the signaling?

Licensing (Encryption License) and Allowing Export Restrictions Requirements

Encrypting media and securing signaling starts with licensing and software versioning. The software version being used must allow Secure Real-time Transport Protocol (SRTP) to be enabled if the goal is to secure both the signaling and media between the different devices. As discussed in Chapter 5, "Hardening the Core Cisco UC Appliance Operating Systems," to enable SRTP, you must use an export-restricted version of Cisco Unified CM or Cisco Unity Connection. Building on this requirement, Cisco began enforcing the license requirement for encryption with Cisco Unified CM 11.5(1) SU3, with the licensing requirement later backported to Cisco Unified CM 10.5(2) SU6. Cisco Unity Connection follows suit and enforces the same licensing requirements with 11.5(1) SU3 and was backported to the 10.5(2) versions. This requirement means that if you are using either version of those products, a co-resident deployment of Prime License Manager (PLM) must be, at a minimum, version 11.5(1)SU3, or if PLM is deployed as a standalone application, it needs to be at least version 11.5(1)SU2 if it is supporting an environment that is either entirely 11.x or a mixed environment of 10.x and 11.5.

Starting with the 12.x versions of the Cisco UC applications, most have moved to using Smart Licensing, which allows for centralized management and tracking of licenses. A discussion of Smart Licensing is beyond the scope of this book, but the requirement for additional permissions to allow strong encryption changes slightly. With Smart Licensing, whether the UC application is registered to the Cisco Smart Software Manager in the cloud or on-premises, the Smart Account needs to be enabled for export-controlled functionality, which is required to enable SRTP.

You can verify that encryption is enabled using various methods. The first method is to use Smart Licensing with version 12.5 of Cisco Unified CM and Cisco Unity Connection via the command-line interface (CLI). You use the **show license usage** command, as shown in in Example 7-1, to output the licensing currently used by the system and, by extension, whether or not encryption is allowed.

A second and more common method is to use the GUI. For Cisco Unified CM, you log in to the Cisco UC Administration GUI and navigate to **System > Licensing > License Usage Report**. From here, you can verify that the encryption license is installed.

https://t.me/learningnets

@cisco_in_persian_channel

Figure 7-1 shows that the encryption license has been installed on a system registered to a Cisco Prime License Manager.

Example 7-1 *Verifying Export Restrictions (CLI)*

```
admin:show license usage

License Authorization Status: AUTHORIZED as of Jul 21 05:39:38 2020 EDT
Output omitted
 (CUCM_Export_Restricted_Authorization_Key)
  Description: <empty>
  Count: 0
  Version: 1.0
  Status: NOT IN USE
  Export status: RESTRICTED_ALLOWED
  Feature Name: CUCM_Export_Restricted_Authorization_Key
  Feature Description: Export Authorization Key for Export Restricted Customers
  running CUCM 12.5 onwards Restricted Software
```

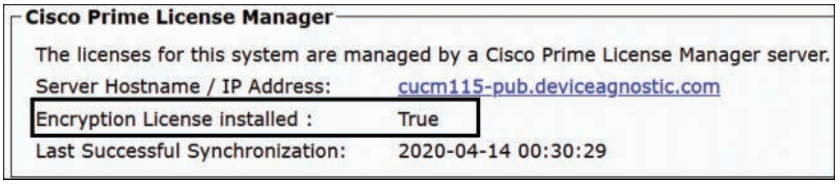


Figure 7-1 *Cisco Unified CM Encryption License*

You can verify the Smart License–enabled versions of the UC applications using the GUI in a similar manner. To verify that the encryption license is installed, you use the following methods. For Cisco Unified CM, you log in to the Cisco UC Administration GUI and navigate to **System > Licensing > License Management**. Then you verify that Export Controlled Functionality is allowed. Figure 7-2 shows that the encryption license has been installed.

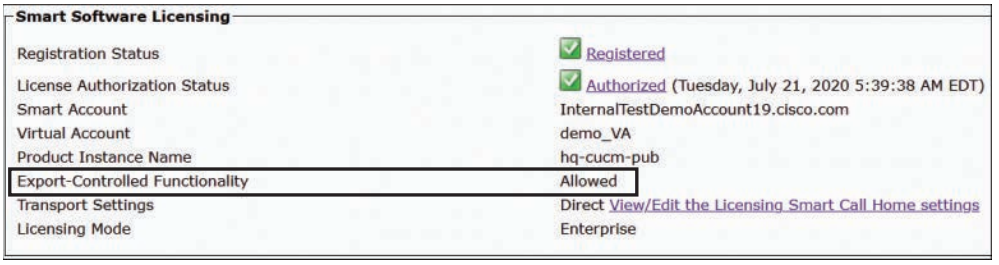


Figure 7-2 *Cisco Unified CM Encryption License*

https://t.me/learningnets

@cisco_in_persian_channel

For Cisco Unity Connection, you log in to the Cisco Unity Connection Administration GUI and navigate to **System Settings > Licenses**. There, you can verify that the encryption license is available for Cisco Unity Connection, as shown in Figure 7-3.

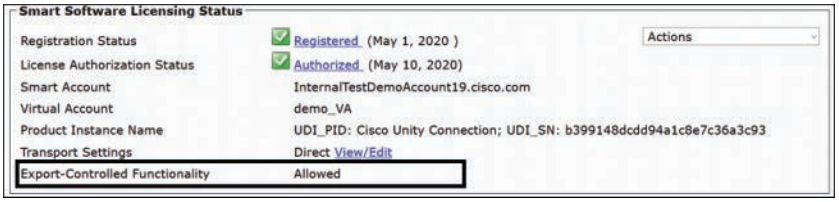


Figure 7-3 *Cisco Unity Connection Encryption License*

If an encryption license is not currently applied to PLM, obtaining one is a simple process. With the UC software versions that use PLM for license tracking, two things must take place. The first is that an appropriate version of software has been installed. This version is the restricted version that was discussed in Chapter 5. The second is that an encryption license (SKU= "CUCM-PLM-ENC-K9=") is obtained. The easiest way to get the license is through the My Cisco Entitlements portal located at https://mce.cisco.com/mce/#/version, as shown in Figure 7-4. Once logged in, you can search based on your associated contracts and then assign the license to your virtual account.

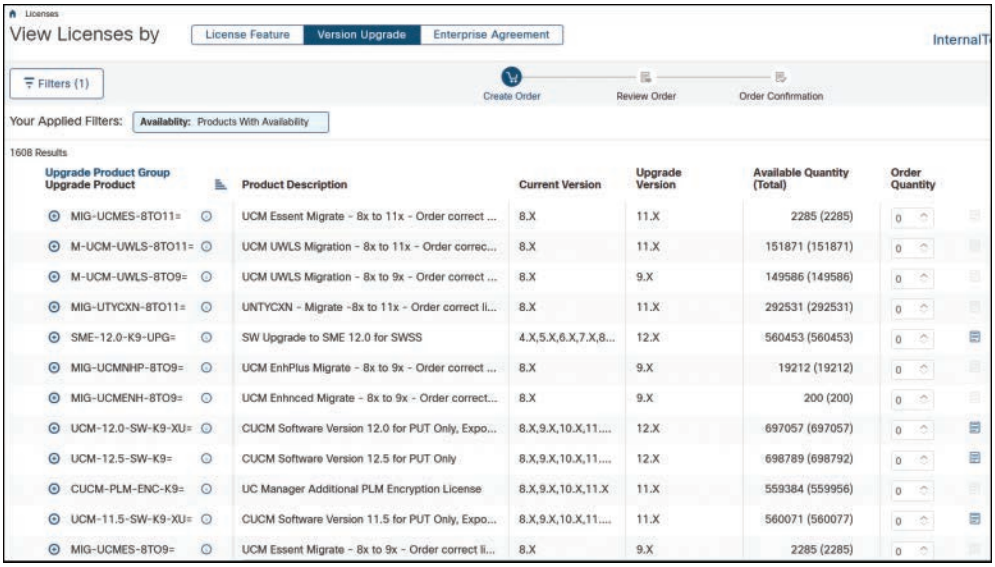


Figure 7-4 *My Cisco Entitlements: Assigning a License*

Once the license is assigned, you can specify the eDelivery method, as shown in Figure 7-5. After it is approved, an email is sent with instructions on how to download the license file.

https://t.me/learningnets

@cisco_in_persian_channel

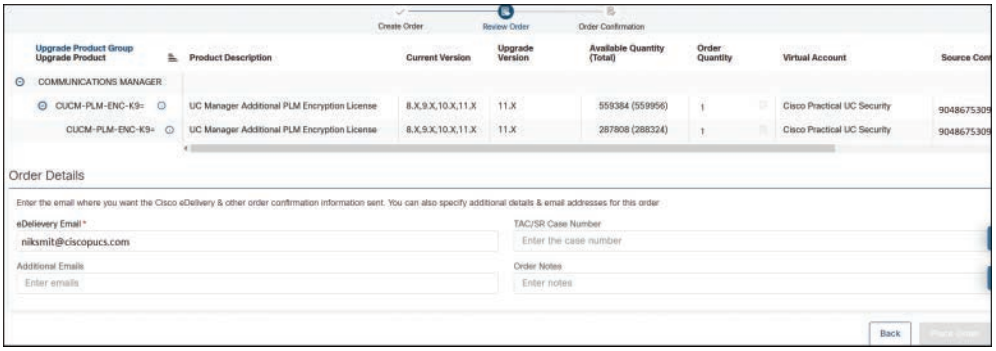


Figure 7-5 *My Cisco Entitlements: License Delivery*

With versions of the UC applications that use Smart Licensing to track their entitlements, a similar set of requirements applies just as it did with versions that use PLM. First, the application—Cisco Unified CM, for example—must use the restricted version; otherwise, the application is not able to support features like strong encryption and SRTP. The second requirement is that the Smart Account and Virtual Account used are configured to allow export-controlled functionality. When this feature is enabled, the **Allow export-controlled functionality** checkbox is present when the registration token is created. When you select this option, the registration token generates a restricted registration token, enabling the strong encryption capabilities within the application. Figure 7-6 shows the options available when creating the registration tokens used with Smart Licensing. If your Smart Account is not enabled for Export Restrictions, you need to open a case with Cisco Licensing to have it enabled.

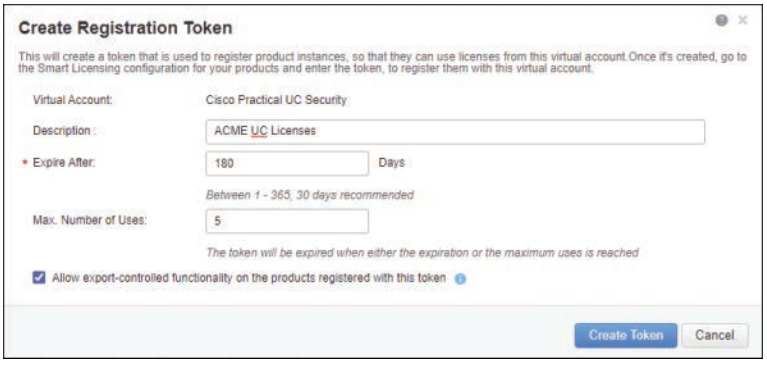


Figure 7-6 *Smart License Restricted Registration Token*

Chapter 8, "Securing Cisco Unified Communications Manager (Cisco)," presents additional details on how Smart Licensing works and is configured. The information in this chapter is provided solely as a reference to the interworking of licensing so that you can enable the strong encryption capabilities of the applications.

https://t.me/learningnets
@cisco_in_persian_channel

FIPS Considerations When Enabling Secure Signaling and Media Encryption

Chapter 5 discussed how to enable the Federal Information Processing Standard (FIPS) and the specific requirements that are imposed on the system when FIPS is enabled. Enabling FIPS is not a requirement to secure the signaling and media for the Cisco UC applications. A limitation to having FIPS enabled is that in some instances certain features cannot be enabled while FIPS is configured. When we discuss these features, we call out the fact that they are not available currently with FIPS enabled.

Public Key Infrastructure Overview

Public key infrastructure (PKI) utilizing X.509 certificates is used extensively across the Internet to secure communications. The methods that a financial institution uses to secure your online communications, for example, are the same methods that Cisco uses to secure Unified Communications.

PKI can be very complex, but in its simplest form, it is a method to authenticate entities by a trusted party. This authentication ensures that the identity the party presents is accurate and provides a basis on which secure communications channels can be established. In this section we discuss the implementation of PKI and provide a primer on how PKI functions to secure communication channels. In later sections we discuss how Cisco uses PKI to secure the signaling and media within a Unified Communications environment. One topic we do not delve into is the complex mathematical operations that go on in the algorithms used to encrypt the data.

PKI is nothing more than a method of authenticating entities to ensure they are who they say they are. An entity, as it relates to PKI, can be a device, person, service, or application. Essentially, an entity can be just about anything. PKI uses X.509 digital certificates containing the identity information of the entity, its public key, and Public Key Cryptography (asymmetric cryptography) to provide authentication and encryption. Because PKI uses asymmetric cryptography, two keys are generated: the first is the private key, and the second is the public key. The private key is never transmitted and should be kept secure, local to the entity. If the private key is compromised, the authentication and data encryption cannot be considered secure. The public key, as the name implies, is "public" and can be shared across the Internet. What makes these keys special is that they are mathematically related, so when one key (e.g., public key) is used to encrypt data, the other key (e.g., private key) can decrypt that data. Also, the key that is used to encrypt the data (public key) cannot then decrypt the data that it just encrypted. This point is important because, in this instance, the key performing the encryption is freely shared. Likewise, the private key can be used to encrypt data that then can be decrypted using the public key.

https://t.me/learningnets

@cisco_in_persian_channel

Why is this capability necessary? How does it provide authentication and encryption? Let's examine a few simple examples to illustrate the process before defining specific attributes of digital certificates or outlining what a chain of trust is.

First, let's look at a scenario where a web browser is connecting to a website using HTTPS. In this case, the website sends its public key back to the web browser. The web browser then uses the website's public key to encrypt a symmetric key for the session and send it back to the website. At this point, the website decrypts the session key using the website's private key. Finally, the web browser and website can communicate securely using the symmetric key for the session. Why are asymmetric and symmetric cryptography both used? The asymmetric keys are used to provide a secure method of exchanging the symmetric keys. Asymmetric cryptography uses larger keys that are not as fast computationally when compared to symmetric keys. Symmetric keys are essentially preshared keys and lack a mechanism to securely transport them from one entity to another, which is why asymmetric cryptography is used to encrypt the communications channel used to share the symmetric key. A breakdown of this communication is illustrated in Figure 7-7.

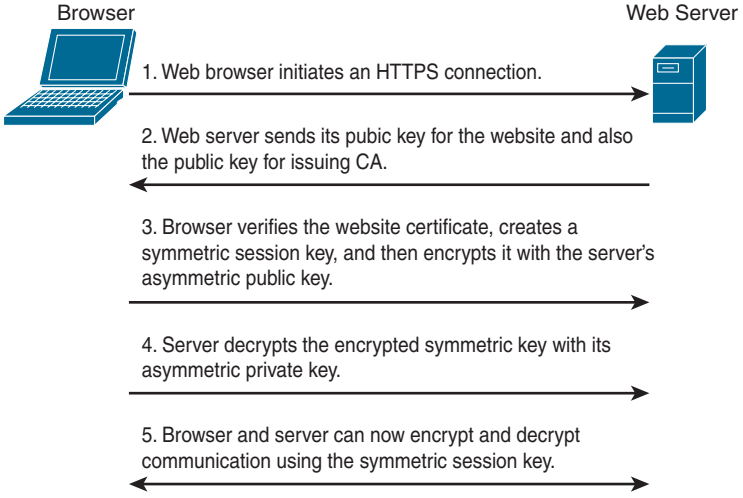


Figure 7-7 *Secure Communication Using Asymmetric Cryptography*

Is this method completely secure? No, it's not. In the preceding example, although the communication between the browser and the website is encrypted, the authenticity of the entities is not verified. A usual indication of this is the warning banner shown in Figure 7-8 stating that the connection isn't private or is untrusted.

Because the authenticity of the website isn't verified, the certificate presented from the entity is just accepted as being valid. In this instance, the communication can fall victim to a man-in-the-middle attack, in which a third party intercepts the communication between the browser and the website. Figure 7-9 illustrates this scenario.

https://t.me/learningnets
@cisco_in_persian_channel

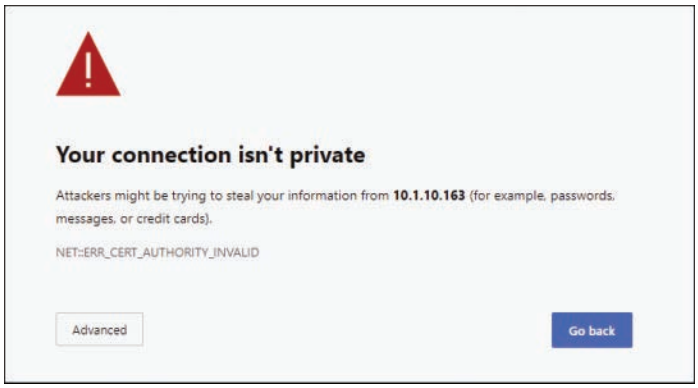


Figure 7-8 *Untrusted Digital Certificate Banner*

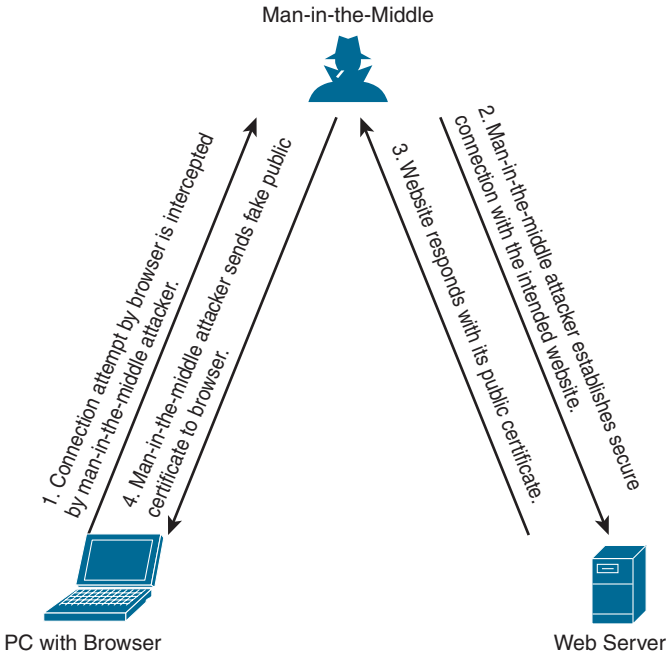


Figure 7-9 *PKI Man-in-the-Middle Attack*

The way to mitigate this type of man-in-the-middle attack with PKI is to use trusted third parties called certificate authorities (CAs). By using CAs, you can build a chain of trust; because the CA is considered trusted, it is used to issue the digital certificates used by PKI. The CA performs the vetting to ensure the identity of the entity that is being issued the certificate is correct. Browsers and operating systems in general have many public CAs configured that are considered trusted by default. They can also add additional CAs and digital certificates to the trusted certificate stores as required.

https://t.me/learningnets
@cisco_in_persian_channel

Digital certificates do not have to be signed by a CA, and depending on the use, it may make sense for a digital certificate to not be signed by a CA. A self-signed digital certificate is created locally by the service, device, and so on and has the same issuer and subject. Figure 7-10 shows a self-signed certificate—in this case, a CAPF certificate in Cisco Unified CM. Because the certificate is primarily used internally, having it signed by a CA is usually not necessary.

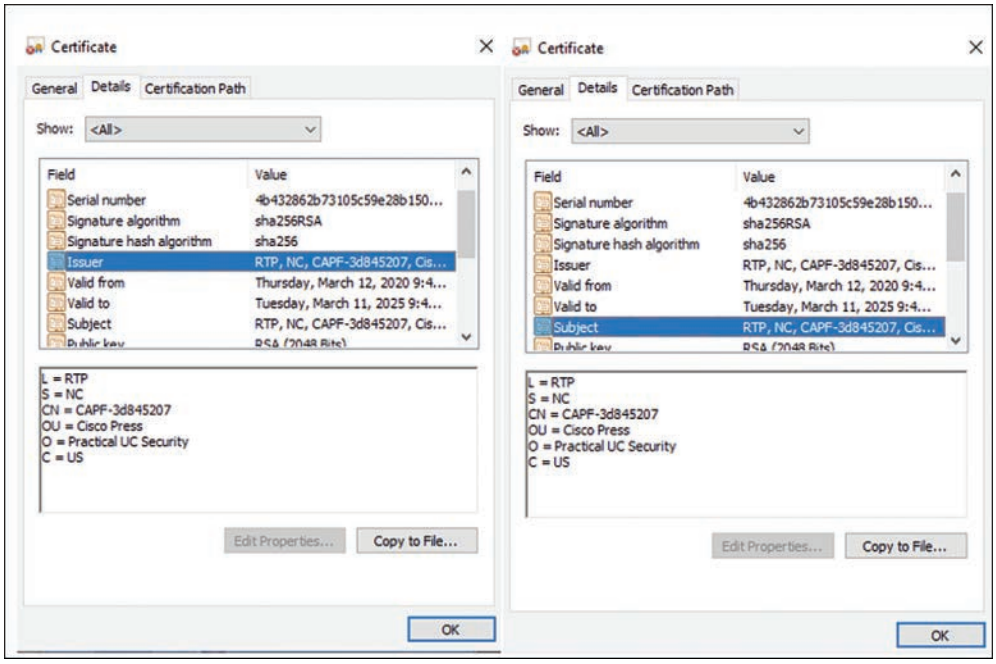


Figure 7-10 *Self-Signed Certificate*

When you're working with a chain of trust, the root CA is always a self-signed certificate. Three common hierarchies for deployments of CAs are

- **Single tier:** In a single-tier hierarchy, the root CA and the issuing CA are the same device. This deployment works for labs and testing but is not the best choice for a production environment. An example of a single-tier CA hierarchy in a UC environment happens when locally significant certificates (LSCs) are configured for endpoints using the default Certificate Authority Proxy Function (CAPF) service. In this hierarchy, the chain of trust for a certificate that is issued by the CA goes from the issued certificate to the root CA. Understanding the concept of chain of trust is important. We discuss the role of the CAPF service later in this chapter and provide a more detailed explanation regarding how it plays into securing the signaling and media within a Cisco UC environment. Figure 7-11 illustrates the generation of a digital certificate in a single-tier CA hierarchy.

https://t.me/learningnets
@cisco_in_persian_channel

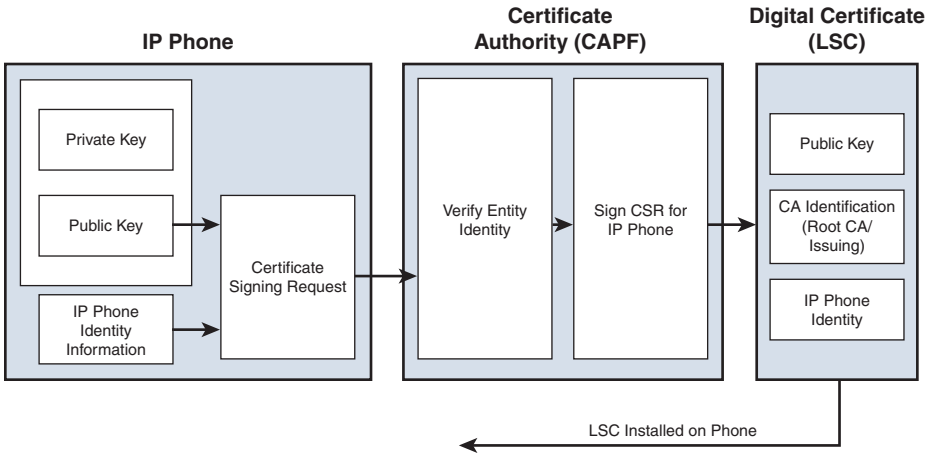


Figure 7-11 *Single-Tier CA Hierarchy*

In the single-tier hierarchy, when a digital certificate is signed by a CA, the CA signs the public key of the entity requesting the certificate with the private key of the CA issuing the digital certificate. When it is used to secure communication, two certificates are used. The first is the issuing CA certificate (the public key of the CA that signed the certificate), and the second is the certificate of the entity attempting to communicate securely. First, the issuing CA certificate is verified against the trusted root CAs in the trust store. If the root/issuing CA used to sign the digital certificate is present in the trust store, the public key of the issuing CA is used to decrypt the signing certificate. If this process is successful, the digital certificate used to sign the digital certificate can be considered trusted. After the issuing certificate is verified as trusted, the digital certificate that was signed is verified using the same process. The public key of the issuing CA is used to decrypt the signed digital certificate. If this process is successful, at this point, the digital certificate is considered trusted.

- **Two tier:** The two-tier hierarchy builds on the single-tier hierarchy by separating the roles of the root CA and the issuing CA. The issuing CA can also be called an intermediate CA or subordinate CA. In this model the root CA is used to sign the digital certificate of the issuing CA. The issuing CA then is used to sign certificates for the environment, and the root CA is typically kept offline. The root CA can be used to sign the certificates for multiple issuing CAs that can then be used to sign certificates based on specific criteria like region, device type, or certificate function. The point here is that the root CA is no longer used to sign certificates in day-to-day operations; this function is offloaded to the issuing CA. In the two-tier hierarchy, the chain of trust extends from the issued certificate to the issuing CA to the root CA. A good example of this type of hierarchy is the certificate used to secure the Cisco Unified CM Administration GUI. For this situation, the Cisco Tomcat service has

https://t.me/learningnets
@cisco_in_persian_channel

its public key signed by the intermediate CA. For the certificate to be trusted, each certificate in the chain of trust needs to be present. Figure 7-12 shows the two-tier hierarchy and the extended chain of trust.

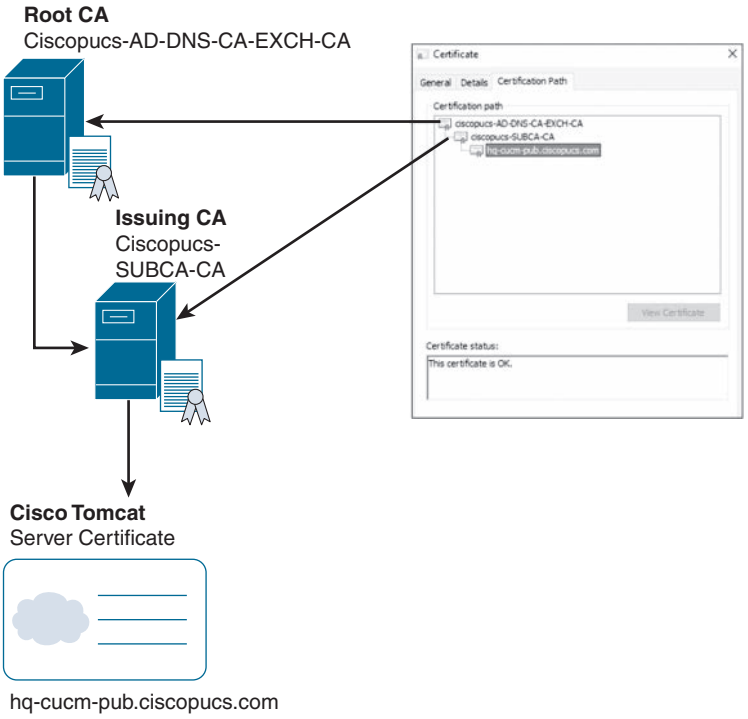


Figure 7-12 *Two-Tier CA Hierarchy*

- **Three tier:** A three-tier CA hierarchy is just an extended version of the two-tier hierarchy. The advantage to this hierarchy is that the issuing CAs can be further subdivided. The practice of keeping only the issuing CAs online continues with this deployment model, illustrated in Figure 7-13.

Now a certificate that is signed by a CA is used to establish a chain of trust. This chain of trust is essential in secure communication channels because the CA provides the authenticity of the parties involved. Let's look back to the example of the web browser connecting to a secure website. If the browser trusts the CA used to sign the digital certificate of the website, it can safely accept that the website is valid. If the browser doesn't trust the CA used to sign the certificate, the communication with the website should be considered suspect. That doesn't mean the traffic can't be trusted; it's just that the chain of trust is not available for the certificate.

https://t.me/learningnets
@cisco_in_persian_channel

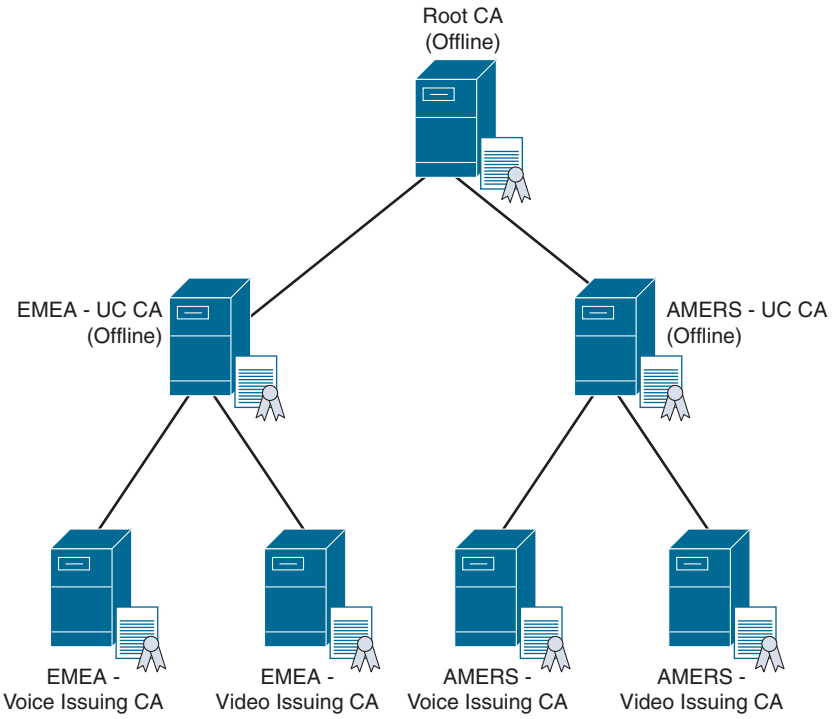


Figure 7-13 *Three-Tier CA Hierarchy*

Now let's expand on the example of a browser accessing a website securely. When a complete chain of trust is present, the communication is secure because both the browser and the website trust the CA that signed the certificates. In this scenario, when the browser attempts to connect to the website securely, the website returns two certificates: the certificate of the website and the certificate of the CA that signed the certificate. The browser inspects the certificate of the issuing CA to verify whether it trusts the signer of this certificate. Because the issuing CA certificate was signed using the private key of the root CA, the browser can verify the issuing CA certificate if it has the root CA public key. Next, the browser inspects the website's digital certificate. Because the website's digital certificate was signed using the private key of the issuing CA, the browser uses the public key provided in the issuing CA's certificate to verify the website's digital certificate. This trust is transitive; because the browser trusts the root CA that signed the issuing CA's certificate, the browser trusts the issuing CA. Since the browser trusts the issuing CA's certificate, it trusts the website's digital certificate because it was signed by the issuing CA's private key. A simplified version of this communication is shown in Figure 7-14.

https://t.me/learningnets

@cisco_in_persian_channel

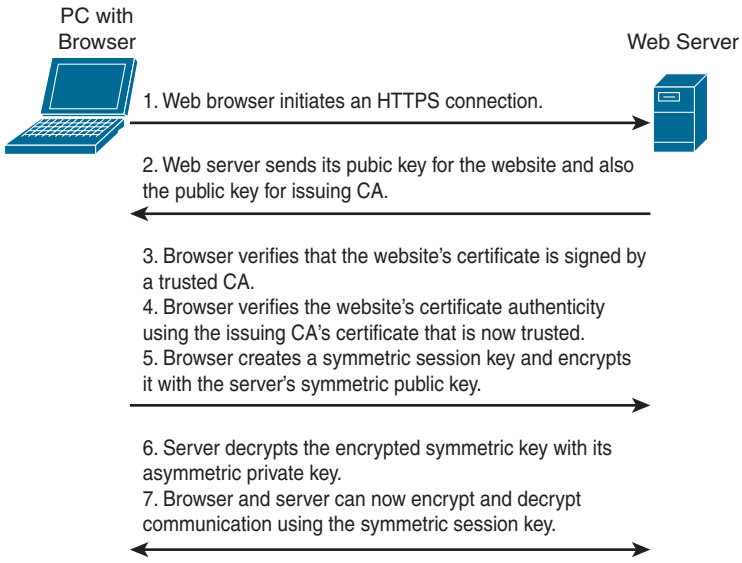


Figure 7-14 *Secure Communications Using CA-Signed Digital Certificates*

Utilizing Public Key Infrastructure with Cisco Unified Communications

Securing the media and signaling within Cisco UC applications is similar to the previous example of a web browser communicating securely over the Internet. PKI and Public Key Cryptography using certificates, discussed here, have a narrow focus on how they are utilized to secure communications within a Cisco UC environment. For example, when a self-signed certificate is not used to secure a service, the server/service generates a key pair that consists of a set of public and private keys and a certificate signing request (CSR). The CSR is a block of encoded text that contains the public key, organization name, common name (FQDN/domain name), locality, and country. The CSR is submitted to a CA to be signed, and the CA verifies the identity of the submitter as part of the signing process. After the certificate is signed, it can then be configured on the server/service where the private key is used to validate the authenticity that the proper public key was used in the creation of the certificate.

The three main sections of a digital certificate are

- **Basic constraints:** This section defines whether the certificate is a CA certificate or an end entity certificate.
- **Key usage:** This section provides the constraints for what cryptographic operations the certificate can perform. Examples could be that the certificate can be used for a digital signature and/or key encipherment. Specific requirements for the common Cisco services are shown in Table 7-1.

https://t.me/learningnets

@cisco_in_persian_channel

Note Table 7-1 and Table 7-2 are current as of CSR 12.0. Future releases of the UC applications are expected to reduce and consolidate the number and types of certificates needed within a UC environment.

Table 7-1 *Cisco UC Certificate Key Usage Requirements for CSR 12.0*

	Multi server	Digital Signature	Key Encipherment	Data Encipherment	Key Cert Sign	Key Agreement
Cisco Unified Communications Manager						
CallManager CallManager-ECDSA	Y	Y	Y	Y	N	N
CAPF (publisher only)	N	Y	Y	N	Y	N
ipsec	N	Y	Y	Y	N	N
tomcat tomcat-ECDSA	Y	Y	Y	Y	N	N
TVS	N	Y	Y	Y	N	N
Cisco Unity Connection						
ipsec	N	Y	Y	Y	N	N
tomcat tomcat-ECDSA	Y	Y	Y	Y	N	N
Cisco IM & Presence						
cup cup-ECDSA	N	Y	Y	Y	N	Y
cup-xmpp cup-xmpp-ECDSA	Y	Y	Y	Y	N	Y
cup-xmpp-s2s cup-xmpp-s2s-ECDSA	Y	Y	Y	Y	N	Y

https://t.me/learningnets

@cisco_in_persian_channel

	Multi server	Digital Signature	Key Encipherment	Data Encipherment	Key Cert Sign	Key Agreement
Cisco IM & Presence						
ipsec	N	Y	Y	Y	N	N
tomcat tomcat-ECDSA	Y	Y	Y	Y	N	N
Cisco VCS & Expressway						
Server Certificate	N	Y	Y	Y	N	N
Cisco Meeting Server						
CallBridge	N	Y	Y	N	N	N
WebBridge	N	Y	Y	N	N	N
WebAdmin	N	Y	Y	N	N	N
TURN	N	Y	Y	N	N	N
XMPP	N	Y	Y	N	N	N
Database Client	N	Y	Y	N	N	N
Database Server	N	Y	Y	N	N	N

- **Extended key usage:** This section specifies the certificate extensions that list the purpose the certificate public key can be used for. A web server, for example, has server authentication listed under extended key usage. These extensions can be either called out as the object short name, like "Client Authentication," or use the dotted numerical form of the Object ID (OID), such as 1.3.6.1.5.5.7.3.2.

Specific requirements for the common Cisco services are shown in Table 7-2.

Table 7-2 *Cisco UC Certificate Extended Key Usage Requirements for CSR 12.0*

	Multi server	Server Authentication (1.3.6.1.5.5.7.3.1)	Client Authentication (1.3.6.1.5.5.7.3.2)	IP Security End System (1.3.6.1.5.5.7.3.5)
Cisco Unified Communications Manager				
CallManager CallManager-ECDSA	Y	Y	Y	N
CAPF (publisher only)	N	Y	N	N

https://t.me/learningnets

@cisco_in_persian_channel

	Multi server	Server Authentication (1.3.6.1.5.5.7.3.1)	Client Authentication (1.3.6.1.5.5.7.3.2)	IP Security End System (1.3.6.1.5.5.7.3.5)
Cisco Unified Communications Manager				
ipsec	N	Y	Y	Y
tomcat tomcat-ECDSA	Y	Y	Y	N
TVS	N	Y	Y	N
Cisco Unity Connection				
ipsec	N	Y	Y	Y
tomcat tomcat-ECDSA	Y	Y	Y	N
Cisco IM & Presence				
cup cup-ECDSA	N	Y	Y	Y
cup-xmpp cup-xmpp-ECDSA	Y	Y	Y	Y
cup-xmpp-s2s cup-xmpp-s2s-ECDSA	Y	Y	Y	Y
ipsec	N	Y	Y	Y
tomcat tomcat-ECDSA	Y	Y	Y	N
Cisco VCS & Expressway				
Server Certificate	N	Y	Y	N
Cisco Meeting Server				
CallBridge	N	Y	Y	N
WebBridge	N	Y	Y	N
WebAdmin	N	Y	Y	N
TURN	N	Y	Y	N
XMPP	N	Y	Y	N
Database Client	N	N	Y	N
Database Server	N	Y	N	N

https://t.me/learningnets

@cisco_in_persian_channel

Next-Generation Encryption Support Using Elliptical Curve Cryptography

With CSR 11.0, Cisco introduced support for next-generation encryption (NGE) using Elliptic Curve Digital Signature Algorithm (ECDSA) certificates. Elliptic curve cryptography (ECC) is public key cryptography based on the algebraic structure of elliptic curves over finite fields. The main benefit in comparison with non-ECC cryptography is that the same level of security is provided using keys of a smaller size.

These certificates are stronger than the standard RSA-based certificates and are required for products supporting Common Criteria certifications. The ECDSA certificates are available along with the existing RSA certificates in these areas:

- Certificate Management:
  - Services that support ECDSA have a trailing "-ECDSA" at the end of the certificate purpose.
  - The certificates using ECC algorithms require the host portion of the common name of the certificate to end in "-EC." This prevents them from having the same common name as the RSA-based certificate for the same service.
  - For services such as Tomcat and CallManager that support a multiserver SAN certificate, the host portion of the common name ends in "-EC-ms."

Figure 7-15 shows a CSR generated for the Tomcat service to use ECC.

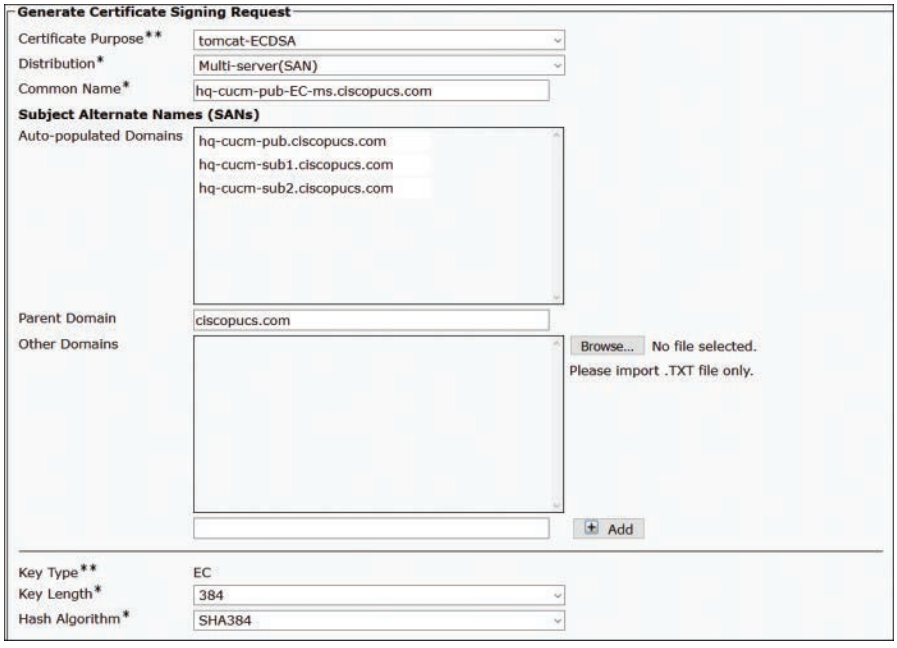


Figure 7-15 *ECDSA Certificate CSR*

https://t.me/learningnets

@cisco_in_persian_channel

- Both the self-signed certificate request and the CSR request limit the hash algorithm choices depending on the EC key size:
  - EC 256 key size: The hash algorithm can be SHA256, SHA384, or SHA512.
  - EC 384 key size: The hash algorithm can be SHA384 or SHA512.
  - EC 521 key size: The only option is SHA512.

  The default key size is 384 and the default hashing algorithm is SHA384, which can be changed. The options available are based on the chosen key size.
- Certificate Authority Proxy Function (CAPF):
  - CAPF version 3.0 enables support for EC key sizes and enables you to specify both EC key size and RSA key size.
  - CAPF options added to the existing CAPF fields are Key Order and EC Key Size (bits). The first allows you to specify the key type to use (RSA/EC/Both), and the second allows you to specify the EC key size if used.
  - If the Key Order option RSA Only is selected, only RSA Key Size is available to be selected. In turn, when EC Only is selected, only EC Key Size can be selected. The option EC Preferred, RSA backup allows both RSA and EC Key Size to be selected.

Note Not all endpoints support ECDSA-based certificates. The Cisco Unified Reporting Tool (CURT) can be used to verify the endpoints supporting ECDSA-based certificates.

- Transport Layer Security (TLS):
  - The Enterprise Parameter TLS ciphers support ECDSA ciphers and set the ciphers for SIP Line, SIP Trunk, and CTI Manager.
- Secure Session Initiation Protocol (SIP) Connections:
  - Cisco CSR 11.0 and higher include ECDSA support for SIP lines and SIP trunk interfaces.
  - The connection between Cisco Unified CM and an endpoint phone or video device is an SIP line connection, whereas the connection between two Cisco Unified CMs is an SIP trunk connection.
- HTTP:
  - HTTP provides support for the secure download of configuration files. If required, both client and server use mutual TLS authentication. Clients and endpoints that are configured to use ECDSA LSCs and TFTP configuration files are required to present their LSC.
  - Port 6971 is used for authentication of the CallManager and CallManager-ECDSA certificates used by phones.

https://t.me/learningnets

@cisco_in_persian_channel

- Port 6972 is used for authentication of the Tomcat and Tomcat-ECDSA certificates used by Cisco Jabber.

- Entropy:
  - Entropy is a measure of randomness of data, and for strong encryption, a robust source of entropy is required. If a strong encryption algorithm, such as ECDSA, uses a weak source of entropy, the encryption can be easily broken.

IP Phone Certificates Types

Cisco uses identity certificates with IP phones as a key component in the secure registration process between the endpoints and Cisco Unified CM. These same certificates also play a role in secure network access for the devices. We discuss these certificates in greater depth later in the chapter, but this section provides a foundational understanding of the roles these certificate play. Cisco Unified IP phones can have two different identity digital certificates: the manufacturing installed certificate (MIC) and the locally significant certificate (LSC). The MIC is installed during manufacturing and cannot be modified or deleted by UC administrators. The CA certificates used to build the chain of trust with the MIC are

Cisco_Manufacturing_CA

ACT2_SUDI_CA

Cisco_Root_CA_M2

Cisco_Manufacturing_CA_SHA2

Cisco_Root_CA_2048

The purpose of the MIC is to provide a mechanism to show that the device is a genuine Cisco phone and has been signed by the Cisco manufacturing CA. A benefit of using the MIC is to prevent a nonlegitimate client from spoofing the MAC address of an existing phone that is configured in the Unified CM cluster. Although the MIC being present on an endpoint does not prove the endpoint is part of your deployed phone base, it does prove that the phone is a genuine Cisco device. It's not advised to use authentication based on the MIC for 802.1x or virtual private network (VPN) because any Cisco endpoint, even devices that are not part of your organization, are then able to authenticate.

The locally significant certificate contains the public key for the IP phone and is created using one of three methods. Although we discuss these methods in more detail later in the chapter, we present them here at a high level:

- **Certificate Authority Proxy Function (CAPF):** This service is used to sign the IP phone's public key when the LSC is generated.

https://t.me/learningnets

@cisco_in_persian_channel

- **Offline CA:** Using this method, the IP phone's certificate signing requests are bundled into a tarball so that they can be signed by a third-party CA. Once signed, the LSCs can be uploaded into Unified CM and installed in the IP phones. Using this method is usually not desirable because it is a largely manual process.
- **Online CA:** This method of deploying LSCs to IP phones uses the Unified CM to forward the request to an enterprise CA (currently only supported for Microsoft).

The following are some of the instances in which MICs and LSCs are used:

- **Installing/upgrading, deleting, or troubleshooting between CAPF and phone for LSC operations:** Depending on the authentication mode configured, the phone certificate is used to establish a secure connection to the CAPF service. The two modes that use the MIC and LSC to establish communications are By Existing Certificate (Precedence to LSC) and By Existing Certificate (Precedence to MIC).
- **Secure signaling using mTLS between Cisco Unified CM and IP phone:** The phone uses a phone certificate to establish an mTLS connection with Unified CM. Unified CM validates the phone certificate against the CallManager-Trust certificate store. The MIC CAs, CAPF certificate, and/or the certificate chain (if a third-party CA is used to sign the phone certificates) must be present in the CallManager-Trust certificate store.
- **802.1x authentication for IP phone and network access:** When 802.1x is used for network authentication, the MIC CAs, CAPF certificate, or full certificate chain of the LSC (if it is signed by a third-party CA) should be uploaded to the authentication server, such as the Identity Services Engine (ISE). This way, the authentication server can authenticate the phone certificate when presented for network access.

It is not possible to specify which phone certificate gets used when setting up the mTLS connection between the IP phone and Unified CM. Nor is it possible to specify the phone certificate used for network authentication. In these instances, the LSC is prioritized over the MIC when the LSC is present.

The general recommendation is to use the MIC during the first CAPF enrollment to generate the first LSC on the endpoint. Once the endpoint has an LSC, the recommendation is always to use the LSC rather than the MIC for authentication when renewing the LSCs. For endpoints that do not have or expose an MIC (for example, Jabber), the first CAPF enrollment authentication can be based on an authentication string or null string. Authentication based on an authentication string is more secure but requires the user to enter a string manually on the endpoint. If this approach is not practical, you can choose authentication based on a null string, but this method effectively bypasses any endpoint authentication during the first CAPF enrollment. When Jabber has an LSC, as with the rest of the endpoints, authentication based on the LSC is recommended for any LSC renewals.

https://t.me/learningnets

@cisco_in_persian_channel

TFTP File Encryption

Cisco Unified CM supports two methods of key distribution supporting the ability to encrypt configuration files. The methods used are either manual key distribution or symmetric key distribution using the public key of the phone's MIC or LSC. Both methods require that the phone security profile assigned to the endpoint is enabled for TFTP Encrypted Config downloads.

When the symmetric key is manually distributed to endpoints, either the authentication string can be manually defined, or Unified CM can generate the string to be used. After a key is created, it is copied to the Unified CM database, and the administrator or user needs to enter the key in the phone. When the key is entered in the phone, the phone stores the key in flash and resets. After resetting, the phone requests the encrypted TFTP file. The Cisco Unified Reporting Tool can generate a report to determine the phone's encryption supported by different model phones.

The encrypted configuration file downloaded by the phone is signed by the TFTP server and, depending on the phone model, may or may not be validated. The phone decrypts the configuration file using the symmetric key that was previously entered manually. If the file is decrypted successfully, it is loaded; otherwise, the configuration is not loaded, and the phone attempts to boot using the old configuration. A downside to using the manual method is that if TFTP Encrypted Config is disabled, the manually entered symmetric key must be deleted from the phone GUI. Until this step is completed, the phone will request an encrypted configuration file.

Most deployments currently use the phone's public key to securely transmit the symmetric key used to encrypt the configuration files. This method of encrypting the configuration files uses a similar approach to a web browser securely connecting to a website. This method uses PKI where the phone initially requests a signed configuration file using the naming format **SEP<macaddress>.cnf.xml.sgn**. This file is signed by the TFTP server's private key. When the phone downloads the configuration file, it parses the configuration file for a few pieces of information.

1. The phone verifies the signer of the configuration file using the public key contained in the CTL or ITL. The configuration file contains the subject name of the certificate that signed the configuration file. The phone uses the certificate's corresponding public key contained in the CTL or ITL to verify the signer.
2. The phone verifies whether the configuration file downloaded is the full or partial configuration. Specifically, the phone verifies whether the parameter **<fullConfig>** is set to **True** or **False**. Because the configuration file is configured to be encrypted, it is set to False and only a minimal configuration is provided.

Note If a phone requests an encrypted configuration file but it is not enabled for this file, Cisco Unified CM returns a "File Not Found" error. The phone then requests an unencrypted file when it encounters this message.

https://t.me/learningnets

@cisco_in_persian_channel

3. The phone identifies whether the configuration file should be encrypted or not. The **<encrConfig>** parameter set to **True** means that the configuration should be encrypted, and the phone should request an encrypted configuration.

4. The phone compares the MD5 hash of its certificate (MIC/LSC) to the value contained in the **<certHash>** parameter. If the values match, the phone requests an encrypted configuration.

5. If the verification of the configuration fails or encounters an error, the partial configuration downloaded contains the CAPF configuration that should be used to synchronize the phone's public key with Cisco Unified CM.

Note If a public key for a phone does not exist in the Unified CM database, the phone initiates a connection to CAPF unless the CAPF authentication mode is set to "By Authentication String." The CAPF authenticates the phone to Unified CM and extracts the public key from the LSC or MIC and then generates an MD5 hash. This hash is stored in the database. The phone then resets and requests a new configuration file. Figure 7-16 shows a breakdown of the different portions of a partial configuration file.

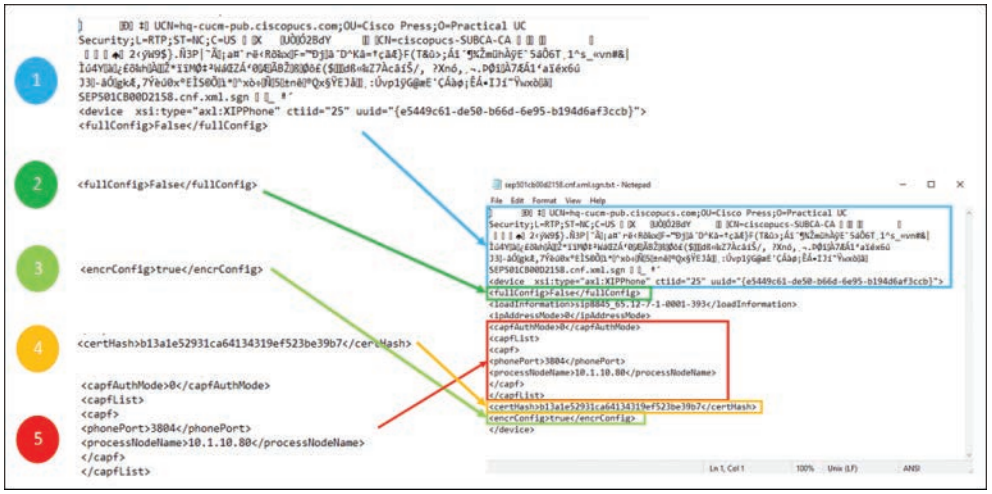


Figure 7-16 *IP Phone Signed Configuration File*

If the phone can successfully verify the signer, the phone requests an encrypted configuration file using the naming format **SEP<macaddress>.cnf.xml.enc.sgn** from the Unified CM. The phone uses the private key (either MIC or LSC) to secure communication between the phone and Unified CM. The Unified CM TFTP server generates a 128-bit symmetric key and encrypts the configuration file. The phone's public key is used to encrypt the symmetric key, which is included in the signed header of the configuration file. When the phone receives the configuration file, it uses its private key to decrypt the symmetric key; then the decrypted symmetric key is used to decrypt the configuration file.

https://t.me/learningnets

@cisco_in_persian_channel

Overview of the Endpoint Registration Process

When endpoints register to Cisco Unified CM, a process known as Security by Default is utilized to harden the registration process. This process is applicable only to Cisco endpoints registering to Cisco Unified CM. Endpoints registering to Cisco VCS Expressway do not use this methodology.

Security by Default

Cisco implemented Security by Default (SBD) in Cisco Unified CM 8.0 to address concerns about compromising the endpoint registration process. With SBD, Cisco introduced several key features that lay the framework for a secure endpoint registration process.

The core features of SBD are

- Default authentication of TFTP downloaded files (configuration, ringlist, and so on) by using a signing key
- Encryption of TFTP configuration files using a signing key
- HTTPS connection verification using certificate verification from a remote certificate trust store in Cisco Unified CM using the Trust Verification Service (TVS)

The introduction of the Initial Trust List (ITL) file allows a supported endpoint to authenticate configuration files downloaded using TFTP or HTTP. The ITL is a trust list that contains several certificates used to verify the Cisco Unified CM cluster that the phone is attempting to register to. Depending on the version of Cisco Unified CM implemented, the trust anchor (signer) of the ITL file varies. Because the ITL file is a trust list, it is signed by the System Administrator Security Token (SAST). The role of SAST is assigned to the ITLRecovery certificate and the CallManager certificate of the TFTP server that the phone is configured for. Only certificates with the SAST role can sign the trust lists in the phone.

When an endpoint initially tries to register to a cluster, it is in one of three states:

- **Valid ITL file:** With a valid ITL file present, the phone verifies the cluster it is attempting to register with using the certificates in the ITL file. During the phone bootup process, it first requests a Certificate Trust List (CTL) from the Cisco Unified CM cluster. We discuss this CTL in greater depth in a following section, but for now you just need to know that the CTL is used when the Cisco Unified CM cluster is placed in mixed mode, enabling TLS and strong encryption. If mixed mode is not currently enabled, the Unified CM TFTP server responds with a "404 Not Found" message. Next, the phone attempts to download the ITL file. The ITL file that is downloaded is verified using the existing ITL file and the trust anchor used to sign it or using the Trust Verification Service to attempt to verify the ITL signer. At this point, because the ITL file is trusted, it is installed and the configuration file is

https://t.me/learningnets

@cisco_in_persian_channel

downloaded. The configuration files that are downloaded are also signed, and the phone verifies them using the certificates in the ITL file.

- **Invalid ITL file:** When a phone has an invalid ITL, it attempts to register using the same process as a valid ITL. It requests the CTL file, and if the Cisco Unified CM cluster is not in mixed mode, it is not present, so the Cisco Unified CM cluster responds with a "404 Not Found" message. Next, the phone attempts to download the ITL file. In this example because the ITL is not valid, it means that the signer of the file (trust anchor) cannot be verified. Because the trust anchor is not valid, the phone uses the TVS to try to verify the signer of the file.

  There are two possible outcomes to using the TVS. If it fails, the phone does not load the ITL file but retains the existing ITL file. This outcome is called loss of trust, and depending on the cluster configuration, the phone may or may not be able to register. Additionally, in a loss of trust situation, the ability to make updates to the phone is limited.

  However, if the TVS can verify the signer, the ITL file is installed in the phone. The use of TVS is important because it allows the phone to offload some of the requirements for certificate validation to Cisco Unified CM, thereby reducing the size of the certificate trust stores on the phone.

  If the TVS service can verify the ITL file, after it is installed, the phone continues the boot process and downloads its configuration files and verifies them with the newly installed ITL file.

- **No ITL file:** If the phone does not contain an ITL file, either because it is new and has never registered to a Cisco Unified CM cluster, or it has registered and the ITL is either empty or was erased, the Security by Default process changes slightly. The difference when an ITL is not present or is empty involves a leap of faith. The boot process does not change for the phone when compared to the previously discussed ITL states until the new ITL file is downloaded. Because the ITL file is not present in the phone, it does not have a trust list to verify the new ITL file. Without an existing trust list to use as a reference, the phone blindly accepts the new ITL and makes a leap of faith to accept the ITL file and install it. This leap of faith is important because it allows new phones to register to Cisco Unified CM without an existing ITL file. After the new ITL is successfully installed in the phone, it continues its boot process by downloading its configuration files.

The ITL file is provided to all endpoints attempting to register to Cisco Unified CM. If an endpoint does not support using an ITL file, it does not request one but instead requests an unsigned configuration file. An additional file considered when an endpoint registers or changes it configuration is the Certificate Trust List. We discuss the CTL later in the section on enabling mixed mode in a Cisco Unified CM cluster. The takeaway is that if a phone does not have a CTL file, it automatically trusts the first ITL file it receives or uses the existing ITL (if present) to verify the new ITL file. If a CTL file is present, it is used to verify the signature of the ITL file.

https://t.me/learningnets

@cisco_in_persian_channel

The ITL file contains a server certificate, public key, serial number, signature, issuer name, subject name, server function, DNS name, and IP address for each server in the Cisco Unified CM cluster that performs the following functions:

- **Certificate Authentication Proxy Function (CAPF):** The CAPF certificate is included in the ITL file to authenticate the LSC. It is also a prerequisite to support enabling encryption of TFTP configuration files.
- **Trust Verification Service (TVS):** The TVS certificate is used by the endpoint for authentication requests for certificates that it does not trust to Cisco Unified CM.
- **System Administrator Security Token:** Certificates with this function can be used to verify the signature of the ITL and CTL files. Most deployments now use the TFTP server's CallManager certificate or ITLRecovery file in versions of Unified CM 12.5(1) SU1 and higher. The ITLRecovery file is used to verify the signature of the ITL file. Hardware USB tokens are available for use with the CTL client to create SAST certificates, but they are being phased out in favor of using the ITLRecovery file.
- **ITL Recovery:** This is the first of the two certificates typically used to verify the signature of the ITL and CTL files. In versions of Cisco Unified CM 12.1(SU1) and higher, this is the primary certificate used to perform the verification of the ITL and CTL files. In prior versions, the CallManager certificate was used to sign the ITL. The ITLRecovery file was introduced in version 10.5(2) as a mechanism to regain the trust between an endpoint and the Unified CM cluster in the event it was broken. The ITLRecovery file is a long-lived (5-year expiry) certificate currently in Cisco Unified CM 12.5. If the system has been upgraded from a prior release and the ITLRecovery file has not been updated, the lifetime of the certificate is 20 years.
- **TFTP / CallManager:** This is the second of two certificates that can be used to verify the signature of the ITL and CTL files. It is included to allow the endpoint to authenticate the downloaded files, signaling, and call control of the Cisco Unified CMs the endpoint is attempting to register to. Each Unified CM node running the TFTP service has its own ITL file that it provides to endpoints.

There are a few key concepts about the ITL file that you should be aware of. The ITL file is only served by the TFTP servers and can be viewed with the **show itl** command in the CLI. The ITL file is displayed as one file but is broken into two distinct sections: the standard ITL file and the extended ITL file. The standard ITL file contains the digital certificates using RSA algorithms as well as the EC certificate of the local CCM+TFTP server. The extended ITL contains the digital certificates built using the next-generation encryption algorithms as well as the RSA certificate of the local CCM+TFTP server.

How can you tell which ITL file a phone will use? There are two ways. The first is to verify if **Enterprise Parameters > Endpoint Advanced Encryption Algorithms Support** is set to **Enabled** rather than the default Disabled. Changing the setting, as shown in Figure 7-17, requires a restart of the Cisco TFTP service and a reset of the endpoints.

https://t.me/learningnets

@cisco_in_persian_channel

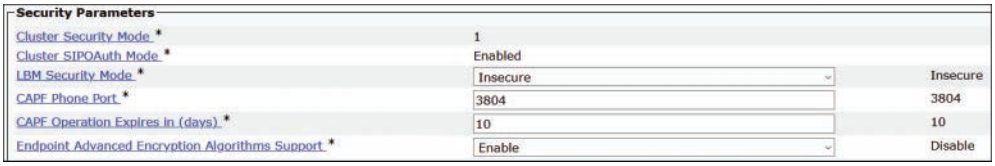


Figure 7-17 *Enabling EC Encryption and Extended ITL*

You can also verify the ITL used from the CLI when you issue the **show itl** command and then verify that signerversion is 1.1, as shown in Figure 7-18.

```
admin:show itl
The checksum value of the ITL file:
f1d5a11c3665a43cd8a4ce2b41516e43(MD5)
449a7b30973daca6f0bb9e9d345cc205c7c92e1e(SHA1)


The checksum value of the Extended ITL File with EC entries:
34cdb0f865153c63d33b54ee944b231d(MD5)
ef3c69b324cb0b210db52dd363d8a5574816fd1a(SHA1)


================================================================================================
Showing Regular ITL File.
================================================================================================
Length of ITL file: 10844
The ITL File was last modified on Mon Jul 27 13:12:00 EDT 2020

        Parse ITL File
        ----------------

Version:         1.2
HeaderLength:    536 (BYTES)

BYTEPOS TAG                     LENGTH  VALUE
------- ---                     ------  -----
28      SIGNERVERSION           2       1.1
3       SIGNERID                2       219
```

Figure 7-18 *Verify ITL SignerVersion*

The ITL is built dynamically as the content for the certificates it contains is modified. Depending on the version of Cisco Unified CM deployed, the private key associated with the ITLRecovery file or the TFTP server's CallManager private key is used to sign it.

When you're upgrading from a version of Cisco Unified CM prior to version 12.5(1)SU1, the ITL file is signed by the CallManager certificate's private key of the TFTP server. Although this was the standard in previous versions, it is a best practice to update the ITL and CTL files, so they are signed by the ITLRecovery file. The process to perform this has two parts: the first, which is optional, is to regenerate the ITLRecovery file; the second is to have it sign the new ITL and CTL files. Figure 7-19 shows the difference between trust anchors pre and post-Unified CM version 12.5 SU1.

https://t.me/learningnets

@cisco_in_persian_channel

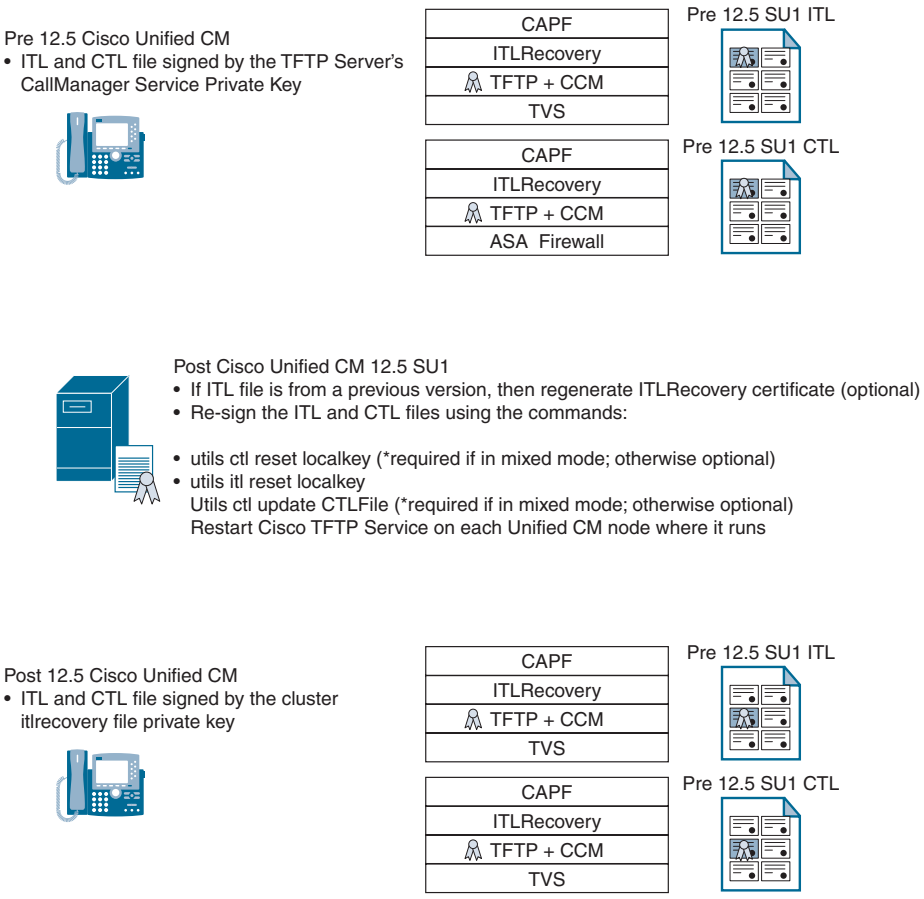


Figure 7-19 *ITL and CTL File Signing*

Note You should verify the existing ITL file is valid on all the phones in the environment. Do not proceed with updating the ITLRecovery file until the existing ITL file is trusted by the phones.

Note We cover the process to regenerate the ITLRecovery file in the event it is expired or close to expiring. This step is not required but is provided for reference.

Note When updating the ITLRecovery file, you need to regenerate this certificate on each node that runs the Cisco TFTP service.

https://t.me/learningnets

@cisco_in_persian_channel

For brevity, we show the process to regenerate the ITLRecovery file using the CLI. The first step in the process to regenerate the ITLRecovery file is to connect to the Cisco Unified CM CLI and run the **set cert regen ITLrecovery** command, as shown in Example 7-2.

Example 7-2 *Regenerating ITLRecovery File*

```
admin:set cert regen ITLRecovery
WARNING: The ITLRecovery Certificate is used to sign the CTL, ITL and other files.
  Regeneration of the ITLRecovery certificate will impact the trust between the
  Phones and Cisco UCM, which may result in phones not being able to register to
  UCM. If this cluster is in mixed-mode you must perform the following steps:
1. Run 'utils ctl reset localkey' form the CLI.
2. Run 'utils itl reset localkey' from the CLI.
If the cluster is in non-secure mode run 'utils itl reset localkey' from the CLI.
After running these commands reset all phones to ensure they receive the updated
  CTL and ITL files which are now signed using the TFTP server's CallManager
  certificate.


After all phones are reset, follow the steps below to update the CTL file with the
  ITLRecovery as signing and allow the TFTP server to sign the ITL file with
  ITLRecovery certificate:
1. Run 'utils ctl update CTLFile' from the CLI
2. Restart tftp server to generate new ITL file.
Reset all phones so that they receive the updated CTL and ITL files.


Proceed with the regeneration (yes|no)? yes


Successfully Regenerated Certificate for ITLRecovery.
You must follow the steps listed above to ensure phones can properly update CTL and
  ITL files.
```

When the command completes successfully, the CTL (if the cluster is in mixed mode) and ITL files must be re-signed by the new ITLRecovery file. For now, the assumption is that the cluster is not currently enabled for mixed mode. The command to re-sign the ITL is **utils itl reset localkey.** This command re-signs the ITL file using the secondary SAST (CallManager) certificate and updates the ITLRecovery file that is contained in the ITL file, as shown in Example 7-3.

Example 7-3 *Re-signing ITL File*

```
admin:utils itl reset localkey
ITL File on all the active TFTP nodes will be reset now and signed using
  CallMananger certificate key...
ITL file Reset.
Please reset all the phones to pickup the new ITL File.
```

https://t.me/learningnets

@cisco_in_persian_channel

You can run the **show itl** command from the CLI to verify that the ITL file has been signed by the Secondary SAST certificate (TFTP Server's CallManager certificate). Figure 7-20 and Figure 7-21 show how to verify which certificate acting as SAST signed the ITL file.

```
        ITL Record #:6
        ----
BYTEPOS TAG             LENGTH  VALUE
------- ---             ------  -----
1       RECORDLENGTH    2       2206
2       DNSNAME         2
3       SUBJECTNAME     85      CN=hq-cucm-pub.ciscopucs.com;OU=Cisco Press;O=Practical UC Security;L=RTP;ST=NC;C=US
4       FUNCTION        2       System Administrator Security Token
5       ISSUERNAME      22      CN=ciscopucs-SUBCA-CA
6       SERIALNUMBER    19      58:00:00:00:07:55:D2:1B:D3:32:42:64:59:00:00:00:00:00:07
7       PUBLICKEY       270
8       SIGNATURE       256
9       CERTIFICATE     1521    F4 09 7A A6 D7 2E 4D C2 EB F4 93 90 CF 69 2B 68 62 FA EA 94 (SHA1 Hash HEX)
This etoken was used to sign the ITL file.
```

Figure 7-20 *ITL Signed by Secondary SAST Certificate*

```
        ITL Record #:3
        ----
BYTEPOS TAG             LENGTH  VALUE
------- ---             ------  -----
1       RECORDLENGTH    2       1786
2       DNSNAME         2
3       SUBJECTNAME     97      CN=ITLRECOVERY_hq-cucm-pub.ciscopucs.com;OU=Cisco Press;O=Practical UC Security;L=RTP;ST=NC;
C=US
4       FUNCTION        2       System Administrator Security Token
5       ISSUERNAME      97      CN=ITLRECOVERY_hq-cucm-pub.ciscopucs.com;OU=Cisco Press;O=Practical UC Security;L=RTP;ST=NC;
C=US
6       SERIALNUMBER    16      6F:E4:89:6A:11:50:32:01:9D:A1:03:BF:D9:AF:54:0E
7       PUBLICKEY       270
8       SIGNATURE       256
9       CERTIFICATE     1017    17 94 A8 95 B8 B4 DC 56 02 9A 73 9E 9F C7 1E FB FE F6 58 30 (SHA1 Hash HEX)
This etoken was not used to sign the ITL file.
```

Figure 7-21 *ITL Not Signed by Primary SAST*

To complete an extra level of verification, you verify the checksum value of the ITL file matches the checksum displayed on the phones. After you verify the ITL file, the next step is to restart the Cisco TFTP service on the nodes in the cluster where it runs. This step forces the ITL file to be re-signed by the ITLRecovery file. With the file re-signed, you need to restart the phones to download the new ITL file. The process to re-sign a CTL file is similar in nature to the ITL file with a couple of caveats, which we explain in the following section.

What It Means to Place a Unified CM Cluster into Mixed Mode

A Cisco Unified CM cluster can run in one of two modes. The first mode, which is non-secure, is the default mode. While Cisco Unified CM is running in this mode, it is unable to secure media using SRTP between the endpoints. The second mode, called mixed mode, requires two tasks. The first task is licensing based and includes either the installation of an encryption license for PLM-based licensing or enabling the Smart Account for export restrictions when Smart Software–based licensing is used. The second task is

https://t.me/learningnets

@cisco_in_persian_channel

the enablement of mixed mode from the CLI of the Unified CM Publisher using the **utils ctl set-cluster mixed-mode** command. In mixed mode, a Cisco Unified CM cluster can exchange encryption keys between endpoints to facilitate the negotiation of encrypted media using SRTP and use mTLS to secure the signaling between endpoints and Cisco Unified CM.

Before attempting to place a Cisco Unified CM cluster into mixed mode, you need to activate two services across the nodes in the cluster:

- **Cisco CTL Provider Service:** The Cisco Certificate Trust List Provider Service is used to change the security mode of the cluster from nonsecure mode to mixed mode. It also facilitates the transporting of server certificates to the CTL file when hardware-based tokens are used. Finally, it replicates the CTL file to all Cisco Unified CMs and TFTP servers. This service is enabled across all the nodes in a cluster.
- **Cisco CAPF:** This service can perform various tasks, depending on the cluster configuration. It can issue locally significant certificates to supported Cisco Unified IP phones. It also can upgrade existing locally significant certificates on the phones and provide a methodology to retrieve phone certificates for viewing and troubleshooting. This service is enabled and available only on the Cisco Unified CM Publisher node.

There are two methods in which mixed mode can be enabled within a Cisco Unified CM cluster. The first is to use a hardware token with the CTL client; the second is to use a software token. The method of using the hardware token and CTL client, although still available, is not widely used anymore and isn't covered to any real extent. The second—and more common—method of using a software token is covered in depth.

Versions of Cisco Unified CM prior to 10.0(1) required the use of two USB hardware tokens to be present for a cluster to be placed in mixed mode. Although this approach worked fine from a technology perspective, day-to-day operational considerations made their use problematic. The most common issue was keeping track of the USB tokens. Because the CTL file that is generated when the cluster is placed in mixed mode does not update often, the tokens could get misplaced. Starting with version 10.0(1), Cisco introduced a "tokenless" method of putting a Cisco Unified CM cluster in mixed mode. This method relies on executing commands from the CLI rather than physical tokens and an external application running to secure the cluster.

You can enable mixed mode from the CLI of the Unified CM Publisher by using the **utils ctl set-cluster mixed-mode** command. When prompted, enter **y** to place the cluster in mixed mode.

Note Typically, the answer to placing the cluster in mixed mode is not displayed until after you send the command (that is, press Enter).

While not required on newer versions of Unified CM, it is advised that, when this step is successful, you restart the Cisco TFTP service on the nodes in the cluster where it is running. Then restart the Cisco CallManager server on all the nodes where that service is

https://t.me/learningnets

@cisco_in_persian_channel

running. The reason you restart Cisco TFTP first is to allow the TFTP server to serve the CTL file to devices attempting to register.

Now that the cluster is in mixed mode, you can verify it either by using the Cisco CM Administration GUI or the CLI.

To verify via the GUI, log in with an administrative account and navigate to **System > Enterprise Parameters**. Under **Security Parameters**, verify that the **Cluster Security Mode*** is set to 1, as shown in Figure 7-22.

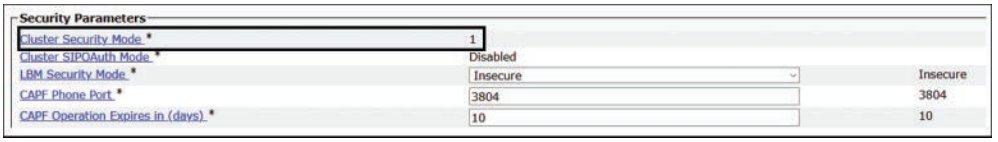


Figure 7-22 *Verify Mixed Mode Using GUI*

Another way to verify that mixed mode is enabled is to use the CLI and execute a SQL query against the Unified CM database, as shown in Example 7-4.

Example 7-4 *Verifying Mixed Mode Using CLI*

```
admin:run sql select paramname,paramvalue from processconfig where
  paramname='ClusterSecurityMode'
paramname                paramvalue
=============== =========
ClusterSecurityMode 1
admin:
```

There are three commands that you can use to work with a Unified CM cluster that is in mixed mode:

- **utils ctl set-cluster mixed-mode:** Creates the CTL file and places the cluster in mixed mode
- **utils ctl set-cluster non-secure-mode:** Removes the CTL file and sets the cluster to the default nonsecure mode
- **utils ctil update CTLFile:** Updates the CTL file on each node in the cluster

CTL Files

Mixed mode creates a file called the Certificate Trust List that is pushed to phones that support it during their registration. The sole purpose of the CTL file is to enable device, file, and signaling authentication.

https://t.me/learningnets

@cisco_in_persian_channel

Like the ITL file mentioned previously, the contents of the CTL file are nothing more than a collection of certificates used by the phones to register and secure their communications. The CTL file contains the following certificate types:

- **System Administrator Security Tokens (SAST):** Like the ITL file before it, the CTL file uses the ITLRecovery file as the primary SAST certificate to sign the CTL, and the TFTP server's CallManager certificate is used as the secondary SAST certificate in Unified CM versions 12.5 SU1 and higher.
- **Cisco CallManager and Cisco TFTP services:** These are used for device registrations as well as to verify the signature of the configuration files.
- **Certificate Authority Proxy Function (CAPF):** The CAPF certificate is a built-in certificate authority within the Cisco Unified CM cluster. This certificate is used in the deployment of locally significant certificates to the phones.
- **ASA Firewalls:** These firewalls support TLS connections with endpoints.

The CTL file contains specific information from the server certificates: public key, serial number, signature, issuer name, subject name, server function, DNS name, and IP address for each server.

CTL files, unlike ITL files, are static in nature and must be updated manually. To do so, you use the **utils ctl upgrade CTLFile** command when any of the following events occur:

- After you change the host name of a Cisco Unified CM node
- After you add or remove a security token (re-signing an existing CTL with the ITLRecovery file)
- After you add or remove an ASA firewall
- If you change the IP address or host name for any configured ASA firewall
- After you add or remove a TFTP server
- After you add or remove a Cisco Unified CM node
- After you upload a third-party CA-signed certificate contained in the CTL
- After you regenerate a self-signed certificate contained in the CTL

When you're upgrading to Cisco Unified CM version 12.5(1) SU1 and higher, it is recommended that you re-sign the CTL file using the new ITLRecovery certificate. This process is like re-signing the ITL file, but because the CTL file is static, you need to complete a couple of extra steps for a successful operation.

First, you can optionally regenerate the ITLRecovery certificate if it hasn't been already.

https://t.me/learningnets

@cisco_in_persian_channel

You use the **utils ctl reset localkey** command to re-sign the CTL file using the secondary SAST (CallManager) certificate and update the ITLRecovery certificate in the CTL file, as shown in Example 7-5.

Example 7-5 *Re-signing the CTL File*

```
admin:utils ctl reset localkey
This operation will reset the CTLFile. Do you want to continue? (y/n): y


Resetting CTL file
CTL file Resetting Please restart Cisco CallManager service and Cisco CTIManager
  services on all the nodes in the cluster that run these services.
```

You use the **show ctl** command from the CLI, as shown in Figure 7-23 and Figure 7-24, to verify that the CTL file has been signed by the secondary SAST certificate.

```
        CTL Record #:1
        ----
BYTEPOS TAG             LENGTH  VALUE
------- ---             ------  -----
1       RECORDLENGTH    2       2223
2       DNSNAME         12      hq-cucm-pub
3       SUBJECTNAME     85      CN=hq-cucm-pub.ciscopucs.com;OU=Cisco Press;O=Practical UC Security;L=RTP;ST=NC;C=US
4       FUNCTION        2       System Administrator Security Token
5       ISSUERNAME      22      CN=ciscopucs-SUBCA-CA
6       SERIALNUMBER    19      58:00:00:00:07:55:D2:1B:D3:32:42:64:59:00:00:00:00:00:07
7       PUBLICKEY       270
8       SIGNATURE       256
9       CERTIFICATE     1521    F4 09 7A A6 D7 2E 4D C2 EB F4 93 90 CF 69 2B 68 62 FA EA 94 (SHA1 Hash HEX)
10      IPADDRESS       4       10.1.10.80
This etoken was used to sign the CTL file.
```

Figure 7-23 *CTL Signed by Secondary SAST Certificate*

```
BYTEPOS TAG             LENGTH  VALUE
------- ---             ------  -----
1       RECORDLENGTH    2       1803
2       DNSNAME         12      hq-cucm-pub
3       SUBJECTNAME     97      CN=ITLRECOVERY_hq-cucm-pub.ciscopucs.com;OU=Cisco Press;O=Practical UC Security;L=RTP;ST=NC;C=US
4       FUNCTION        2       System Administrator Security Token
5       ISSUERNAME      97      CN=ITLRECOVERY_hq-cucm-pub.ciscopucs.com;OU=Cisco Press;O=Practical UC Security;L=RTP;ST=NC;C=US
6       SERIALNUMBER    16      6F:E4:89:6A:11:50:32:01:9D:A1:03:BF:D9:AF:54:0E
7       PUBLICKEY       270
8       SIGNATURE       256
9       CERTIFICATE     1017    17 94 A8 95 B8 B4 DC 56 02 9A 73 9E 9F C7 1E FB FE F6 58 30 (SHA1 Hash HEX)
10      IPADDRESS       4       10.1.10.80
This etoken was not used to sign the CTL file.
```

Figure 7-24 *CTL Not Signed by Primary SAST*

After you verify the CTL file, the next step is to restart the phones so that they can pick up the CTL file signed by the secondary SAST certificate.

After the phones register back to Unified CM, the next step is to update the CTL file so that it will be signed by the new ITLRecovery file; here, you use the **utils ctl update CTLFile** command.

Next, you restart the Cisco CallManager and CTI Manager services and Cisco TFTP services on the nodes in the cluster where they run. With the file re-signed and services restarted, the phones should be restarted once again to download the new CTL file.

https://t.me/learningnets

@cisco_in_persian_channel

If locally significant certificates have been deployed to the phones, the last configuration item to complete is to update those certificates. This step is as simple as using BAT to update the phones by device type or going to each device and setting the **Certificate Operation** under **Certificate Authority Proxy Function (CAPF) Information** to **Install/Upgrade**. Ensure that the **Operation Completes By** is a future date (a default of 10 days is good) because this is how long the process will try to complete, as shown in Figure 7-25.

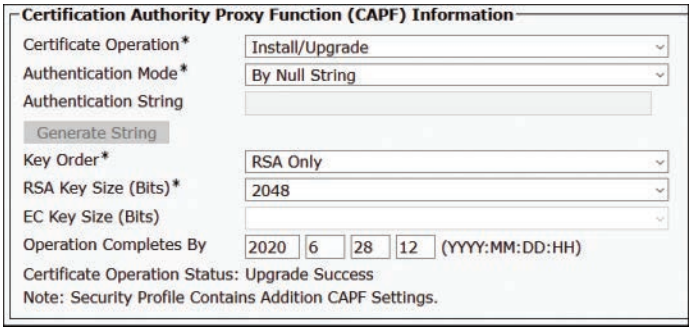


Figure 7-25 *LSC Update per Device*

Using SIP OAuth to Secure Signaling and Encryption

Cisco has begun the process of introducing an alternative method of encrypting signaling and media in a Unified Communications environment that does not require the use of LSCs beginning with the release of Cisco Unified CM 11.5(1) SU3 and CSR 12. With these releases, Cisco has begun to introduce the ability to use OAuth 2.0 (referred to as OAuth) with the end goal of providing always-secure phones. This section covers the basics of what OAuth is, how it functions, and how to configure it.

In the current CSR 12.5, only Cisco Jabber is supported for SIP OAuth on-premises, which allows Jabber to be deployed in an always-secure manner. Specific phone models, namely, the 78XX and 88XX, are supported along with Jabber when deployed using Cisco Mobile and Remote Access (MRA). This section does not cover MRA with SIP OAuth enabled; that topic is left to Chapter 11, "Securing the Edge."

Table 7-3 shows the minimum software requirements for enabling OAuth for soft clients.

Table 7-3 *OAuth Minimum Software Releases for Soft Clients*

Product	OAuth with Refresh	+ SIP OAuth Encryption
Unified CM / IM&P	115.(1) SU3	12.5(1) SU1
Unity Connection	115.(1) SU3	12.5(1) SU1
Expressway	X8.11.4	X12.5
Jabber for Windows, Mac, iOS, Android	11.9	12.5

https://t.me/learningnets
@cisco_in_persian_channel

Product	OAuth with Refresh	+ SIP OAuth Encryption
Webex Teams (Unified CM Calling) Win, Mac	3.0.11421.0	3.0.11421.0
Webex Teams (Unified CM Calling) Android	4.0.17	4.0.17
Webex Teams (Unified CM Calling) Android	4.0.62	4.0.62

Additional prerequisites for using OAuth are in line with enabling mixed mode within a Unified CM cluster. For OAuth to be enabled, the Unified CM cluster must be running a restricted version and have a valid encryption license installed (pre 12.0), or the Smart Account that is used for synchronization must have export controls enabled. A key difference with OAuth is that the Unified CM cluster is not required to have mixed mode enabled.

The following subsections start by explaining why OAuth is used. That discussion is followed by how SAML and OAuth interact to perform authentication and authorization. Last, we describe the steps to enable OAuth. These three concepts are central to understanding how OAuth is implemented and used to secure communications without the need to deploy locally significant certificates with CAPF.

Why Is OAuth Used to Secure Signaling and Media?

To start the conversation on why you should use OAuth, let's look at some key benefits it provides:

- **Limited trust:** The application being used may not be fully trusted. Because OAuth doesn't provide authentication, this task must be handed off to a trusted party. OAuth prevents the application from having access to your password because OAuth defines the permissions in tokens that are requested/issued to restrict their access.
- **Password synchronization:** This task is controlled with an authentication (AuthN) server whether it is an SAML single sign-on (SSO) identity provider (IdP), LDAP server, or even local authentication to a resource. Because tokens are used for OAuth when the password is changed, the authorization is not affected.
- **Reduced risk of compromise due to phishing:** OAuth redirects authentication requests to an AuthN server. This process centralizes authentications, so once logged in, users are not prompted to continue entering passwords.
- **Stronger authentication:** Methods like MFA are promoted over standard password authentication because authentication is centralized at the AuthN server rather than that application.

https://t.me/learningnets

@cisco_in_persian_channel

Why does using OAuth over CAPF make sense? A driving reason for the use of OAuth is due to the deployment of soft clients in the environment. With a soft client in the mix, Cisco does not control the trust stores used by the underlying operating system like it does with a hardware endpoint. With an operating system, many trusted root certificates are already installed by default, so it makes little sense to force updates to these trust stores unnecessarily. A second reason for using OAuth over CAPF to deploy LSCs relates to easing the burden of management. With a soft client like Jabber, if the application needs to be reset or reinstalled, the process to redeploy an LSC must take place in order to secure communications again. From a day-to-day operational perspective, OAuth obviates the monitoring that must happen to ensure that expiring LSCs are updated prior to the expiry dates. In both of those cases, a loss of communication can occur when a valid LSC is not present. When OAuth is used, this is not the case because a TLS connection uses the Cisco Tomcat service to open a TLS connection between Cisco Jabber rather than an mTLS connection between Cisco Jabber and the CallManager service. A final example is that it encourages the use of single sign-on with authorization tokens to reduce the number and frequency of logins for a user. This benefit, by itself, provides added security because it allows for stronger authentication mechanisms like multifactor authentication and keeps users from having to share their passwords with the application.

OAuth is an authorization framework (RFC 6749) that allows third-party applications to access a service or application on behalf of a user. This method of authorization is commonplace on the Internet today and is the same methodology used when signing in to your favorite website and using your Facebook credentials to authenticate you to the website. It is important to reiterate that OAuth only provides a framework for access authorization for a client (application) to a resource on behalf of a user. OAuth does not provide authentication, and it does not specify which authentication services should be used. OAuth requires that authentication happen outside its framework. Although OAuth doesn't dictate a specific type of authentication take place, the example we use here focuses on the use of Security Assertion Markup Language (SAML) for SSO. The specific steps to enable SAML SSO for Unified CM are covered in Chapter 8, "Securing Cisco Unified Communications Manager (Cisco)," and Unity Connection in Chapter 9, "Securing Cisco Unity Connection."

Using SAML for Authentication with OAuth

This section provides a foundational level of understanding about SAML as it relates to how OAuth is used to secure communications. SAML was developed by the Security Service Technical Committee at the Organization for the Advancement of Structured Information Standards (OASIS) and is intended to provide a framework for exchanging security and identity information between different systems. Like OAuth, SAML does not perform the authentication of services, but unlike OAuth, it does provide the method for interacting with the authentication servers. The current version of SAML that is used within Cisco Unified Communications is version 2, which is not backward compatible with version 1.0 or version 1.1.

https://t.me/learningnets

@cisco_in_persian_channel

The components that make up SAML ultimately are reliant on a web browser; they are illustrated in Figure 7-26.

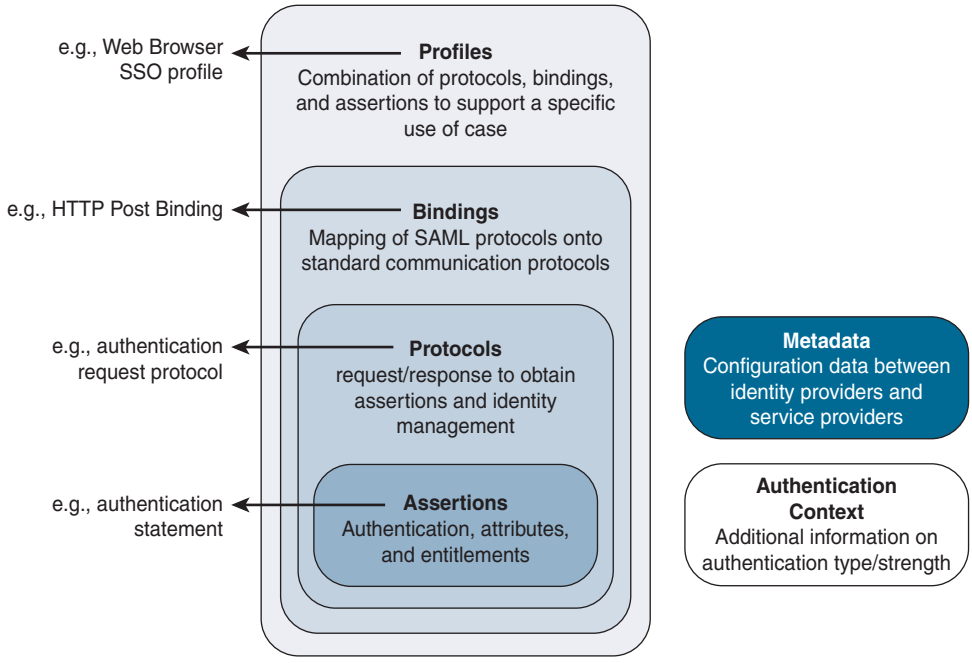


Figure 7-26 *SAML Components*

SAML consists of these three main concepts:

- **Identity provider (IdP):** This entity is otherwise known as an asserting party and it verifies that you are who you say you are. In other words, it is the AuthN server and uses an Identity Management System like LDAP to perform the actual authentication. The IdP can also verify whether the access request to a resource was previously approved or not.
- **Service provider (SP):** This entity can be called the relying party because the SP relies on the IdP to trust whether access should be granted.
- **User:** This entity makes the request for access and provides the credential for authentication (username/password).

Note In the explicit trust relationship between the IdP and SP, the agreements can be completed offline where metadata is exchanged.

https://t.me/learningnets

@cisco_in_persian_channel

The interaction between the different SAML concepts is illustrated in Figure 7-27.

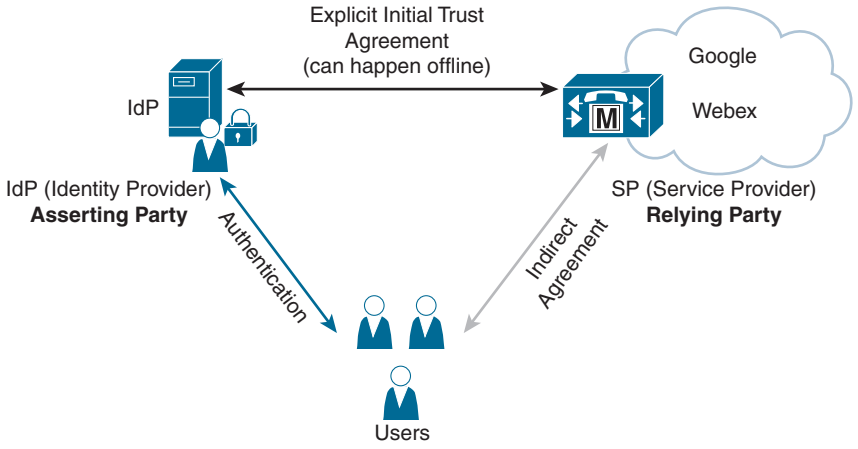


Figure 7-27 *SAML Authentication Framework*

Let's use a simplified scenario of a user logging in to the Cisco Unified CM Administration GUI as an example. The following actions take place. First, a UC administrator attempts to log in to Cisco Unified CM Administration. The Unified CM responds with a redirect and an authentication request to the user's web browser, which redirects to the IdP. The location of the IdP is contained in the metadata that was initially shared when Unified CM was configured for single sign-on (this configuration process is detailed in Chapter 8). Once redirected, the user is prompted to authenticate. The IdP uses the configured Relying Party Claim Issuance Policy that is specified for the service provider to perform that actual authentication. With a successful authentication, the IdP sends an SAML response with an assertion that is signed by the IdP with a cookie to the user's web browser. Based on the URL response, the browser sends an HTTP Post with an SAML assertion back to the SP. Because the metadata that was shared previously between the IdP and Unified CM contained the public certificates, the SP can validate the signature of the SAML assertion. At this point, the SP can provide access based on the verified SAML assertion and send it back to the browser.

Figure 7-28 illustrates a service provider–initiated web browser flow with SAML SSO. The steps occur as follows:

0. Metatdata exchange happens during initial configuration between IdP and SP.

1. A user wants access to a restricted resource (SP), such as Unified CM.

2. The SP resource issues a redirect with authentication request to the user's web browser to the IdP. Metadata is used to identify where to send the request.

3. Authentication happens between the IdP and user, but neither the IdP nor the SP controls authentication. The authentication can be username/password, MFA, and so on, and is specified via the configured IdP policy.

https://t.me/learningnets
@cisco_in_persian_channel

4. Successful authentication sends an SAML response with assertion (signed by IdP) with a session cookie to the user.
5. Based on the URL in the response, the web browser sends a POST with SAML assertion to the SP. The SP has the public certificates that are contained in the metadata to verify the SAML assertion.
6. The SP can provide access based on the verified SAML assertion and send it back to the browser.

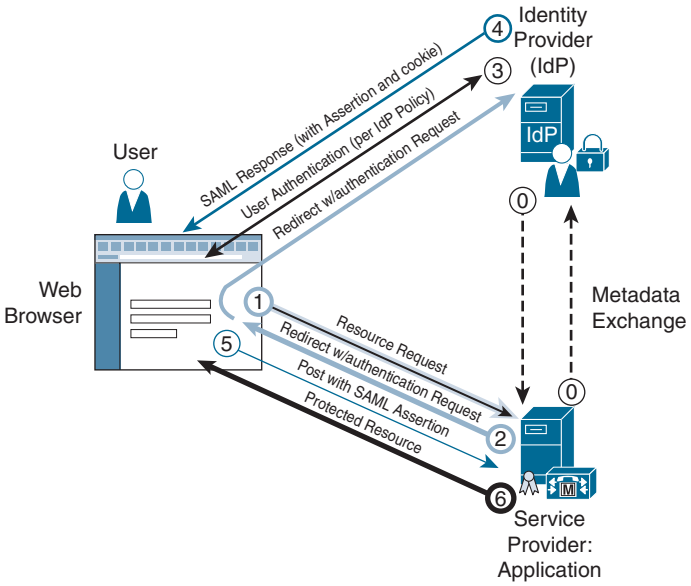


Figure 7-28 *SP-Initiated Web Browser SSO Flow*

Utilizing OAuth for Authorization

After having covered why OAuth should be used and explained how SAML can be used to provide authentication for OAuth, we can now cover how OAuth provides authorization within a Cisco UC environment. OAuth is a standard framework (RFC 6749) that enables third-party applications to obtain access to a service or API on behalf of a user. With OAuth, users can authorize a client application like Cisco Jabber to access a protected resource (Unified CM) without sharing their credentials using access tokens. These tokens come in two forms: an access token and a refresh token. The short-lived access token is used to provide access to a resource. This token is issued to a specific user with specific scopes and expiry. The optional refresh token is used by the OAuth client to request a new access token when the existing access token is about to expire. The refresh token is used to keep the user from having to reauthenticate with the IdP every time the access token expires. When the refresh token requests a new access token, it requires the

https://t.me/learningnets
@cisco_in_persian_channel

OAuth server (AuthZ) to verify with the IdP that the user is active before the new access token is issued. Tokens are like digital certificates in the following ways:

- Tokens must be kept confidential in transit and in storage (TLS only). This is like the public/private key pair used by digital certificates.
- Authorization servers must ensure the security of tokens so that they cannot be generated, modified, or guessed by third parties.
- Authorization servers must verify the binding between a refresh token and client identity. While not the same, the concept of verifying the entity presenting the token is like the chain of trust concept with PKI.

The refresh token is implemented in different ways depending on whether a soft client or physical endpoint is used. For a soft client, the refresh token and the client secret are presented to the authorization server to refresh the access token. This refresh takes place when the access token is due to or has expired. When the refresh token is about to expire, the user is prompted to refresh (log in) the session. This happens on the following schedule when the default 60-day expiry is used for the refresh token:

- First prompt appears when the refresh token has 3 days left to expire.
- Second prompt is shown 24 hours later.
- Third prompt is shown 24 hours later.
- Fourth prompt is shown 1 hour before the token expires.

Once the lifetime of the refresh token has expired, it requires the user to reauthenticate before issuing a new set of tokens to the soft client. A physical phone differs in that once the phone obtains the tokens, they are automatically refreshed without the need for the user to refresh/reauthenticate the device. If the endpoint is offline and not communicating with the authorization server and the refresh token expires, the device needs to be reactivated in order to register.

When OAuth issues tokens, it uses two specific grant types: implicit grant and authorization code grant. Previous versions of Unified CM and Jabber used the implicit grant, but in current releases Cisco has implemented the authorization code grant. Benefits of using the authorization code grant are that the refresh token is enabled, and the tokens are never exposed to the user agent (web browser). With the implicit grant, the refresh token was not used, and this meant that when the access token expired, the user was required to authenticate again. By using the refresh token to request a new access token, the user must reauthenticate only when the refresh token expires. Additionally, because the implicit grant exposed the tokens to the user agent (web browser) and resource owner (user), this required the protected resource to programmatically extract the token from the AuthZ response and pass it to the client (Cisco Jabber). Using the authorization code

https://t.me/learningnets

@cisco_in_persian_channel

grant, the authorization server sends the access token and optional refresh token directly to the client.

Before we delve into the enrollment process, one last aspect to address is that OAuth requires applications to register with the authorization server. Registering ensures that that API requests are correctly identified by providing the application with client credentials consisting of a client ID and client secret. The application provides the client credentials to the authorization server when it requests an access token. For some products like Cisco Jabber, the client credentials are hard-coded into the authorization server. This does not degrade the security of OAuth because communication between Cisco Jabber and the authorization server is secured via TLS. Also, the authentication of the user is handled outside OAuth and secured by the IdP.

Using Cisco Jabber registering on-premises to Unified CM as an example, it is possible to provide always-on phone security with the use of OAuth. For enrollment, the Cisco Jabber client attempts to access a protected resource (Unified CM). The authorization server redirects the user. If the user is not currently authenticated, the process called out earlier in Figure 7-28 commences and the user is authenticated. The authorization server sends a redirect URI to the client with an AuthZ code (one-time code). The client then uses this code to send an access token request to the authorization server that contains the AuthZ code and redirect URI. Finally, the authorization server verifies the code and redirect URI and responds back with the access token and refresh token.

What does this process look like in practice? Figure 7-29 illustrates the OAuth enrollment for a soft client such as Cisco Jabber.

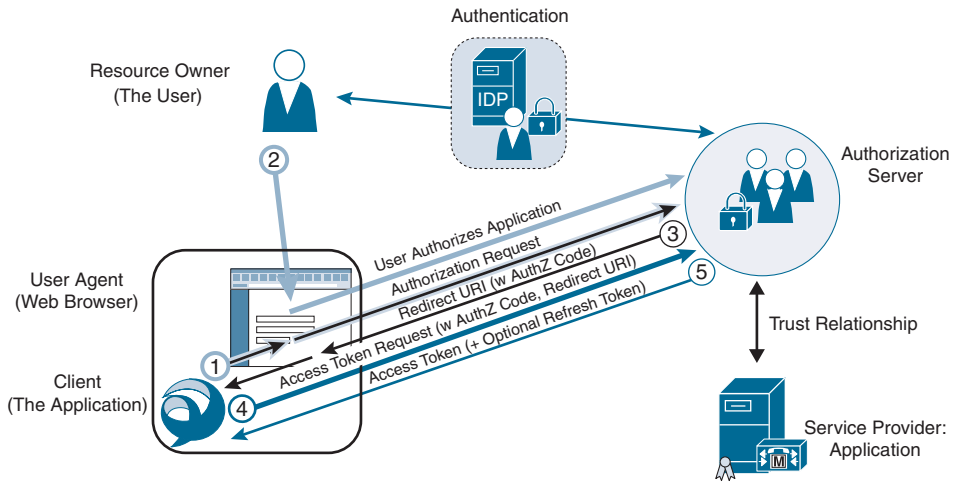


Figure 7-29 *OAuth Enrollment Flow for Soft Client*

https://t.me/learningnets

@cisco_in_persian_channel

Note Currently, only Cisco Jabber is supported for on-premises use, and Cisco MRA is always phone-secure, but the next major release of Unified CM is expected to support this functionality on select phone models.

Figure 7-29 diagrams the enrollment of Cisco Jabber for OAuth roles and access to Cisco Unified CM where the user has been previously authenticated. The steps are as follows:

1. A client requests authorization. Unified CM sends a redirect to the authorization server (in Unified CM, this is SSOSP).
2. A user grants access (an app is authorized to access resources on the user's behalf).
3. The client requests a token from the authorization server.
4. The client requests a resource using the access token from the resource server.
5. The resource server sends the protected resource.

Enabling OAuth on Unified CM and Unity Connection

Enabling the new OAuth features is reliant on meeting the minimum software requirements and enabling them on the different UC applications in a specific order.

You begin by setting the Enterprise parameter in Unified CM Administration to enable OAuth with Refresh Login Flow, as shown in Figure 7-30. By default, this setting is disabled, so enabling this login flow changes the OAuth grant type from Implicit to Authorization Code Grant. The expiry timers for the OAuth access token, Jabber OAuth refresh token, and physical phone OAuth refresh tokens can be modified. The default settings are 60 minutes for the access token and 60 days for the refresh tokens.

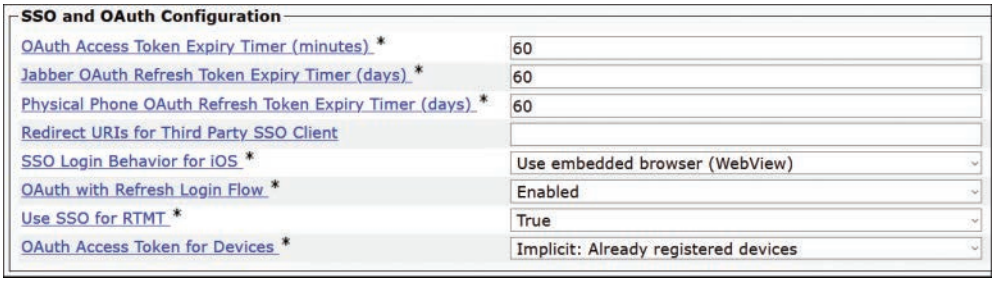


Figure 7-30 *Enabling OAuth Refresh Login Flow*

The default port used for SIP Phone OAuth is 5090, and the SIP Mobile and Remote Access Port is 5091; both are configurable in the event of a conflict. When SIP Phone OAuth is used, registrations for devices utilize TLS and require that the Cisco Tomcat certificate (ideally a multiserver certificate) needs to be signed by a trusted CA. This

https://t.me/learningnets
@cisco_in_persian_channel

changes slightly when MRA is used; instead of TLS being used, mTLS is utilized so that both legs of the communication are verified because the connection is happening over an untrusted network.

Next, you must run the **utils sipOAuth-mode enable** command from the CLI of the Unified CM Publisher:

```
admin:utils sipOAuth-mode enable
SIP OAuth mode enabled.
Please restart the Cisco CallManager service on all nodes in the clus-
ter where it is running.
```

When this step is complete, you need to restart the Cisco CallManager service on all nodes where endpoints register using SIP OAuth. At this point, it is a best practice to restart the Cisco CallManager on all nodes in the Unified CM cluster where it is running, if nothing more than for consistency across the cluster. It does not hurt the cluster if you restart the Cisco CallManager service only on the nodes where devices register using SIP OAuth. However, if you forget and later try to register a device to a node where the service was not restarted, you'll likely spend unneeded time troubleshooting.

The preferred way of verifying that the command was successful is shown in Figure 7-31; in this case, you access **System > Enterprise Parameters** in Cisco Unified CM Administration.

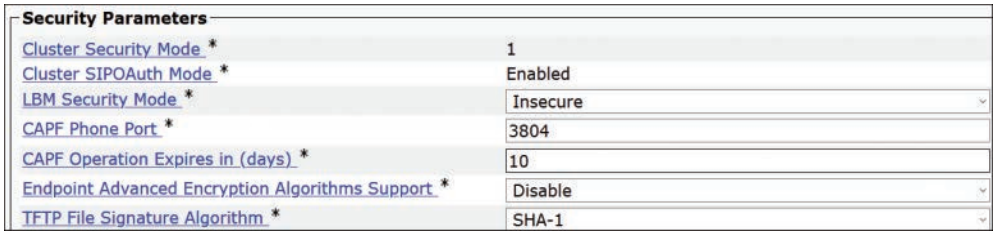


Figure 7-31 *Verifying SIP OAuth-mode Enabled*

Now that Unified CM is enabled to use OAuth with a Refresh Login Flow, you can configure Unity Connection. The process starts in a similar manner to Unified CM. You log in to Cisco Unity Connection Administration and navigate to **System Settings Enterprise Parameters**. The same options present within the Enterprise Parameters for Unified CM are present in Unity Connection under **System Settings > Enterprise Parameters**, as shown in Figure 7-32. Here you need to set OAuth with Refresh Login Flow to Enabled. The default, as with Unified CM is, Disabled.

Cisco Unity Connection, unlike Unified CM, does not have a built-in authorization server, so it must be configured to use one. For this task to be successful, you configure the Unified CM as the AuthZ server.

https://t.me/learningnets

@cisco_in_persian_channel

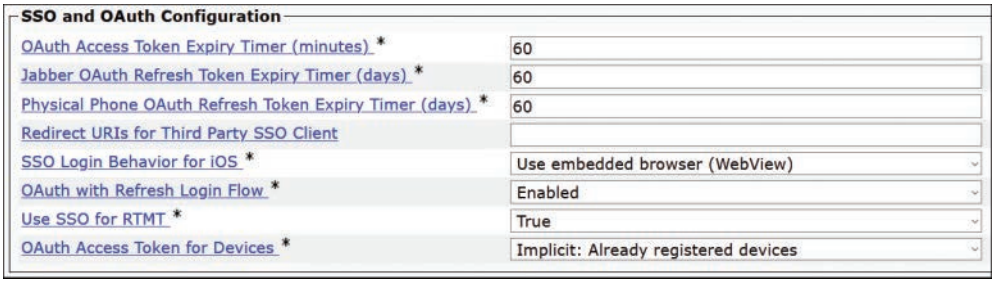


Figure 7-32 *Unity Connection Enabling OAuth with Refresh Login Flow*

While logged in to Unity Connection Administration, navigate to **System Settings > Authz Servers** and select **Add New**. The settings are shown in Table 7-4. Figure 7-33 shows the Unified CM Publisher successfully added as an AuthZ server.

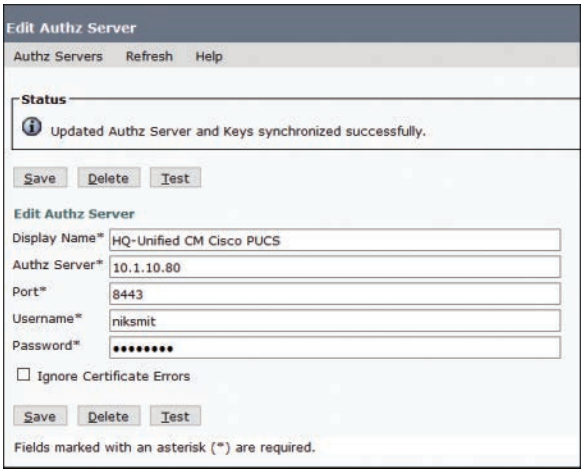


Figure 7-33 *Unity Connection AuthZ Server Addition*

Table 7-4 *Unity Connection AuthZ Server Configuration*

Field	Description
Display Name	Enter the display name for an AuthZ server.
Authz Server	Enter the host name, IP address, or fully qualified domain name of the AuthZ server that Unity Connection connects to.
Port	Enter the port through which Unity Connection connects to the AuthZ server. Default value: 8443

https://t.me/learningnets
@cisco_in_persian_channel

Field	Description
Username	Enter the username that Unity Connection uses to sign in to the AuthZ server.
Password	Enter the password that Unity Connection uses to sign in to the AuthZ server.
Ignore Certificate Errors	Select the checkbox to ignore the certificate validation errors for AuthZ servers. If the checkbox is not checked, you must upload the Tomcat root certificate of Cisco Unified CM to Tomcat trust of Cisco Unity Connection to validate the certificates for the AuthZ server. Default setting: Checkbox is not checked. For more information on certificates, see the "Security" chapter of Cisco Unified Communications Operating System Administration Guide for Cisco Unity Connection Release 12.x at www.cisco.com/c/en/us/td/docs/voice_ip_comm/connection/12x/os_administration/b_12xcucosagx.html.

Configuring Secure Phone Profiles to Enable Secure Signaling and Media Encryption

Cisco Unified CM uses the concept of phone security profiles to enable and disable the security-related settings for signaling and media. By default, Cisco Unified CM has nonsecure phone security profiles created for each supported Cisco endpoint. Universal device templates are a second type of preconfigured phone security profile within Cisco Unified CM. This means that UC administrators can configure either a product-specific phone security profile that can be applied per device type, or universal templates can be configured and applied across all device types. The choice on which to use comes down to preference and operational requirements. One operational reason for utilizing the universal templates is to reduce the number of SAN entries required on the certificate to support MRA (MRA is discussed in Chapter 11). Another reason is simply endpoint management. If there is not a pressing reason to use multiple secure phone profiles within the environment, the use of universal templates encourages consistency in configuration.

Product-specific phone security profiles provide a more granular set of options than the universal phone security templates, as shown in Figure 7-34. This granularity is obviated once the profiles are configured to communicate securely.

Notice that the SIP phone port and transport type are not present in the universal template in Figure 7-34. This does not reduce the security of the profile when the device security mode is set to Encrypted or Authenticated because TLS is required to secure the signaling and media. Also, universal device templates are preconfigured with both nonsecure and secure profiles, unlike the product-specific profiles that are nonsecure by default.

https://t.me/learningnets

@cisco_in_persian_channel

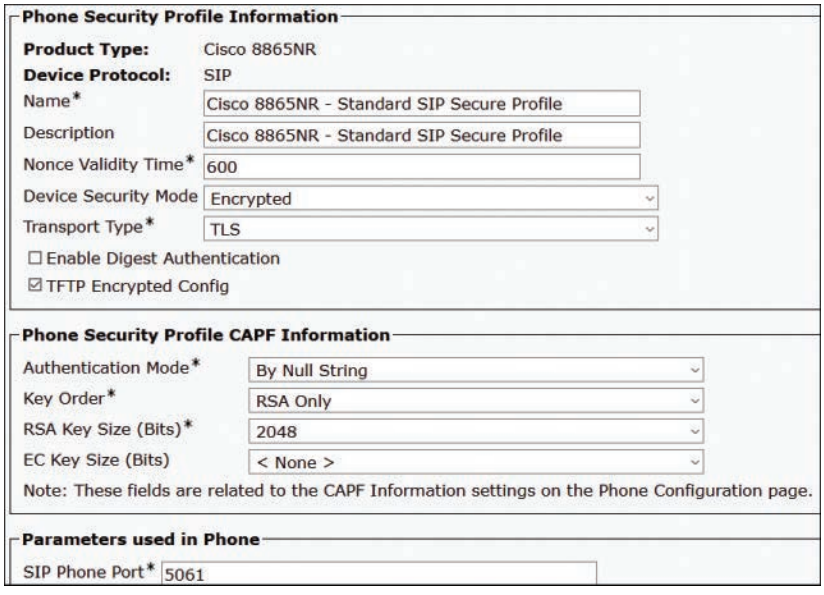


Figure 7-34 *Product-Specific Phone Security Profile*

Let's look at a couple of notes on the phone security profiles in general. The default product-specific profiles cannot be updated. To update them, you must create a copy of the profile. A profile that is assigned to a device cannot be deleted, and changes made to a template that is applied to a device replicate to the device configuration.

When you're deciding on whether to use MIC versus LSC to authenticate TLS connections with Cisco Unified CM, using the LSC is advised. It is possible that the MICs could be compromised, and Cisco recommends utilizing the LSC when possible for this reason.

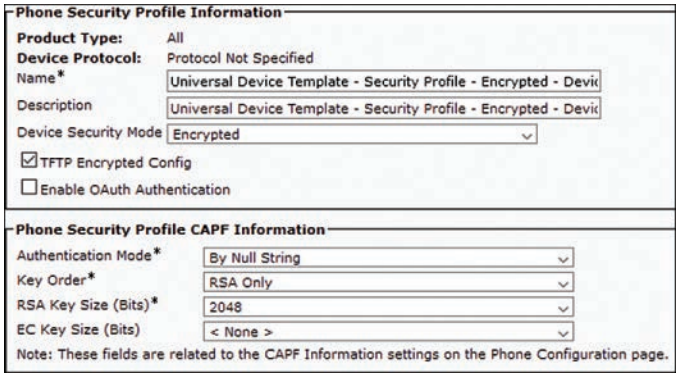


Figure 7-35 *Universal Phone Security Profile*

https://t.me/learningnets

To enable the phone to secure the signaling or media, you need to create or identify a secured profile and apply it to the phone. Figure 7-35 shows what settings these profiles can configure:

- **Device Security Mode:** You can select from the following three options:
  - **Non-Secure:** No security features except image, file, and device authentication exist for the phone. Phones open a nonencrypted TCP connection to Unified CM.
  - **Authenticated:** Unified Communications Manager provides integrity and authentication of the signaling for the phone. An mTLS connection that uses NULL/SHA for signaling.
  - **Encrypted:** Unified Communications Manager provides integrity, authentication, and encryption for the phone. An mTLS connection that uses AES128/SHA for signaling, and SRTP carries the media for all phone calls on all SRTP-capable hops.
- **Transport Type:** A couple of options are provided here. For the nonsecure security modes, the transport type is either TCP, UDP, or both. If the security mode is set to Authenticated or Encrypted, TLS is used.
- **Enable OAuth Authentication:** SIP OAuth authentication allows you to use OAuth refresh tokens for authentication in secure environments. OAuth allows SIP line-secure signaling and media without the need to use CAPF to deploy LSCs by using the MIC to authenticate the endpoint within Unified CM. This does not eliminate the possible need to deploy LSCs to support features like 802.1x where using the MIC may not be desirable.
- **Encrypted Configuration Files:** When this setting is enabled, Cisco Unified CM encrypts the endpoint configuration files that are downloaded from the TFTP server. This option applies only to Cisco endpoints. These configuration files are encrypted using one of two identity certificates installed on the phone. This setting is disabled if OAuth authentication is used.
- **CAPF Settings:** Under CAPF settings, you have the option to select the authentication mode used during CAPF operations. There are four options to choose from here:
  - **By Authentication String:** When you choose this option, the process to install, upgrade, or troubleshoot a locally significant certificate commences only after an authentication string is entered on the phone.
  - **By Null String:** This option installs or upgrades and allows troubleshooting a locally significant certificate without any user interaction.
  - **By Existing Certificate (Precedence to LSC):** Similar to the method using the authentication string, this method uses a certificate installed on the phone. As the name implies, precedence is given to the LSC if installed and will fall back to the MIC.

https://t.me/learningnets

@cisco_in_persian_channel

- **By Existing Certificate (Precedence to MIC):** Like the method using the authentication string, this method uses a certificate installed on the phone. As the name implies, precedence is given to the MIC if installed and will fall back to the LSC.

The CAPF settings also offer the ability to select the key types used: RSA, elliptical curve (EC), or EC preferred with RSA as a backup. Currently, the more common key type used is RSA because legacy endpoints do not support ECDSA certificates. Although newer model phones such as the 78XX and 88XX series phones do support the ECDSA certificates, the Cisco Unified Reporting Tool should be used to verify the phone models that support EC ciphers before enabling them.

- **SIP Phone Port:** This option applies only to Cisco phones running the SIP firmware that are in a nonsecure mode. Phones listen to this UDP port number for messages from Cisco Unified CM. If the transport type for the phone security profile is set to TCP or TLS, this setting is ignored.

Applying the Secure Phone Profiles and LSC to the Phones

In this section, we discuss the methods of deploying an LSC and then explain the nonsecure and secure communication indicators.

With the Cisco Unified CM cluster in mixed mode and the phone security profiles that will be applied identified, the next step is to deploy the LSC to the phones. Cisco has several options available with varying levels of involvement. The first and most common method, enabled by default, is called Cisco Certificate Authority Proxy Function, just like its namesake service. When CAPF is used to deploy the LSC to a phone, the phone authenticates itself to CAPF using the configured authentication method in the phone security profile. The phone generates a public key and private key pair and forwards its public key to the CAPF server in a signed message. The private key remains in the phone and never gets exposed externally as part of the process to deploy an LSC. CAPF signs the phone certificate and sends the certificate back to the phone in a signed message, at which point the phone installs the certificate. This process can take up to 30 minutes to complete depending on the phone model, key type, and key size being used.

Let's use a single phone as an example. In this case, you log in to Cisco Unified CM Administration GUI and navigate to **Device > Phone**. Then you search for and select the phone to be secured. On the device configuration page for the phone, two sections are updated. The first section, **Protocol Specific Information**, has the **Device Security Profile** updated to **Universal Device Template – Security Profile – Encrypted – Device_TFTP**, as shown in Figure 7-36.

https://t.me/learningnets

@cisco_in_persian_channel

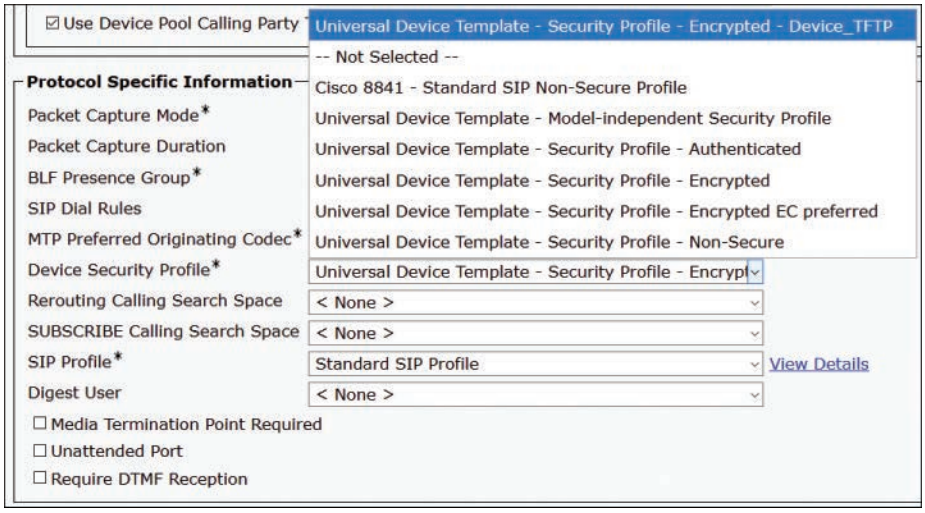


Figure 7-36 *Protocol-Specific Information*

The second section to be updated is **Certificate Authority Proxy Function (CAPF) Information**, and it has the **Certificate Operation*** changed to **Install / Upgrade**. The **Operation Completes By** section has a date in the future configured, as shown in Figure 7-37. The system defaults to 10 days in the future, and this timer indicates how long the system will try to install or upgrade the LSC to the phone before giving up.

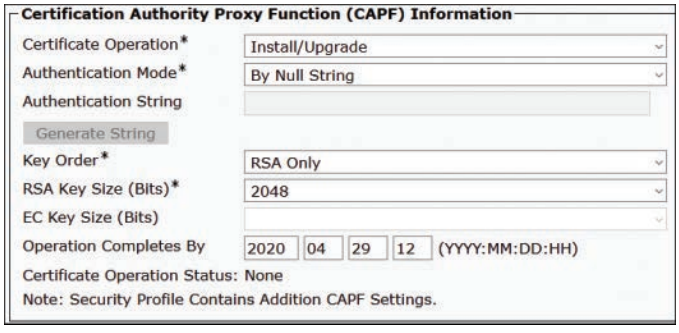


Figure 7-37 *Certification Authority Proxy Function*

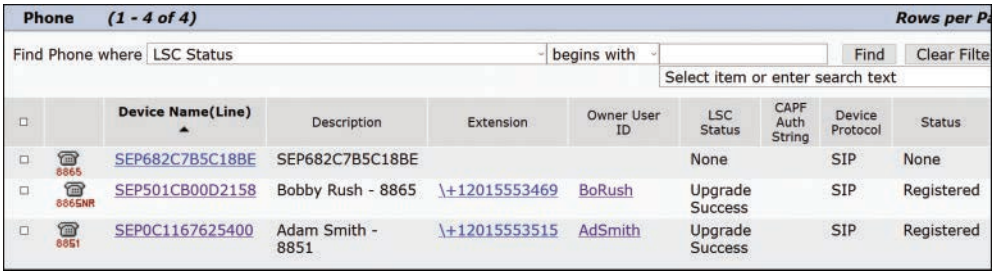
Then you need to save and apply the configuration. The phone reboots a couple of times and registers with a new LSC installed. You can verify the LSC on the phone (physical device) by navigating to **Admin Settings > Security Setup** (see Figure 7-38). The current security status of the phone is displayed.

https://t.me/learningnets

@cisco_in_persian_channel

Figure 7-38 *Phone Security Setup*

If the phone does not reboot, or you are unsure of the status of the LSC install, you can use Cisco Unified CM Administration GUI to gather the current status. Within the Cisco Unified CM Administration GUI, navigate to **Device > Phones.** From the search options, select **Find Phone Where LSC Status** and perform an empty search to return a list of the phones and their current LSC status, as shown in Figure 7-39.



		Device Name(Line)	Description	Extension	Owner User ID	LSC Status	CAPF Auth String	Device Protocol	Status
□	8865	SEP682C7B5C18BE	SEP682C7B5C18BE			None		SIP	None
□	8865NR	SEP501CB00D2158	Bobby Rush - 8865	\+12015553469	BoRush	Upgrade Success		SIP	Registered
□	8851	SEP0C1167625400	Adam Smith - 8851	\+12015553515	AdSmith	Upgrade Success		SIP	Registered

Figure 7-39 *LSC Status Verification*

One drawback to using CAPF to deploy the LSC is that the CAPF certificate is typically a self-signed certificate. Depending on the network environment where the LSCs are being deployed, these certificates could be flagged during a security audit. A typical complaint from the security department is that the LSC is an unmanaged certificate authority because it can provide X.509 certificates to network devices (IP phones). The remaining two methods of deploying LSCs solve this problem.

The first of these methods, Online CA, is a feature introduced in Cisco Unified CM 12.5(1). The Online CA consists of several components that enable the CAPF service to forward requests to a registration authority that then brokers the request with the enterprise CA to finally issue the certificate.

https://t.me/learningnets
@cisco_in_persian_channel

The main conceptual components within Unified CM that facilitate this method to deploy the LSCs are

- **Certificate Authority Proxy Feature (CAPF):** The Unified CM service that phones interact with when performing certificate enrollment requests. CAPF interacts with the CES on behalf of the phones. In Figure 7-40, CAPF implements libEST in client mode to enroll the phones' certificates through CES.
- **Certificate Enrollment Services (CES):** The service in Unified CM that acts as the RA between the CAPF service and the CA. CES is also referred to as CiscoRA, or simply RA.
- **Registration Authority (RA):** The service that verifies user requests within the network for digital certificates and tells the CA to issue the certificate.

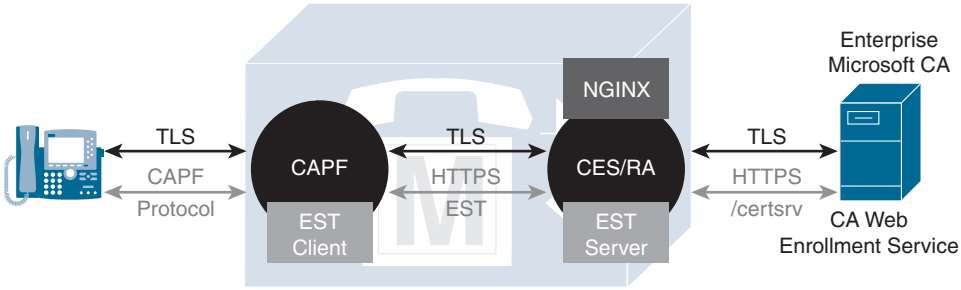


Figure 7-40 *Online Certificate Authority Message Flow Diagram*

The process to deploy an LSC using the Online CA mode can be simplified down to a couple of phases. The first phase begins with the phone sending a certificate request to CAPF. When this happens, CAPF communicates with Certificate Enrollment Services, acting as a registration authority, which processes and responds to CAPF's request. The CES/RA process communicates with the CA's Web Enrollment Service via HTTPS to issue the certificate to the phone.

You can find a full explanation of the message flow and components involved when Online CA is used for LSC deployment in the Troubleshooting Technote – Troubleshooting CAPF Online CA located on the Cisco.com website.

Note Currently, the Online CA method supports only a Microsoft enterprise certificate authority.

The last method of deploying LSCs to phones is an older method named Offline, as shown in Figure 7-41. With this method, the CSRs for the phones are downloaded from the Cisco Unified CM cluster and submitted to a CA to be signed. Once signed, the certificates are uploaded to Unified CM and sent to the phones to complete the LSC installation.

https://t.me/learningnets
@cisco_in_persian_channel

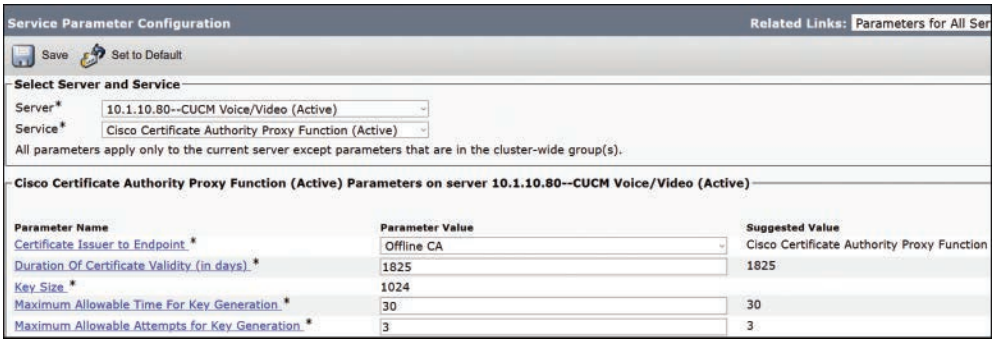


Figure 7-41 *Offline CA Service Parameters*

This method, like the Online CA, allows the phones to have a CA-signed certificate as their LSC. While the Online CA method is automated like the CAPF process, it currently supports using only a Microsoft CA. The Offline method, however, requires more manual effort but works with other third-party certificate authorities.

To start using the Offline process, you set the **Certificate Issue to Endpoint** value to **Offline** by logging in to Cisco Unified CM Administration GUI and navigating to **System > Service Parameters**. Once there, select the **Cisco Unified CM Publisher** and the **Cisco Certificate Authority Proxy Function** services. The CAPF service needs to be restarted after the service parameter is updated.

To generate a CSR to be signed by an external CA, select a phone that needs an LSC installed or upgraded. From the phone's **Certificate Authority Proxy Function (CAPF) Information**, update the **Certificate Operation*** to **Install/Upgrade** and set the **Operation Completes By** to a date in the future, as shown in Figure 7-42. When this step is complete, save and apply the configuration.

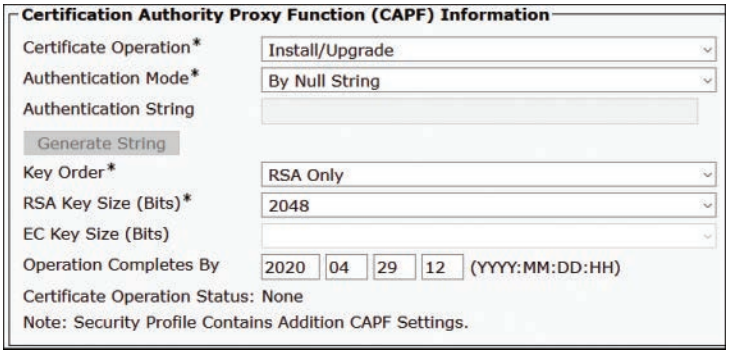


Figure 7-42 *Offline Device CSR Generation*

Next, you connect to the Cisco Unified CM Publisher via SSH and execute the **utils capf csr count** command to verify that the expected CSRs have been generated, as shown in Example 7-6.

https://t.me/learningnets
@cisco_in_persian_channel

Note The phones cannot be reset after the keys are generated. If the phones are reset before the LSCs are installed, the key pairs are regenerated and no longer valid for the CSRs that were previously downloaded.

Example 7-6 *Using utils capf csr count*

```
admin:utils capf csr count

Count CSR/Certificate files.
Valid CSR   : 3
Invalid CSR : 0
Certificates: 0
```

When the expected number of CSRs is present, you can run the **utils capf csr dump** command to download the compressed CSRs to an FTP or TFTP server, as shown in Example 7-7.

Example 7-7 *Using utils capf csr dump*

```
admin:utils capf csr dump

Dump CSR files.
CSR File tarred successfully...

Destination:

 1) Remote Filesystem via FTP
 2) Remote Filesystem via TFTP
 3) Local Download Directory
 q) quit

Please select an option (1 - 3 or "q" ): 2
File Path: /
Server: 10.1.10.152

File exported successfully

checksum value for tmp_capf_csr.tgz:
SHA512
fab926902e0b78f2b08bb79a04bf334692fbe116f98b8be109567cbe1d7be5f0a01fe1f3f732550036
  c0009c3250e565545952805 2b26d215fcb358c89a050c6
```

With the compressed CSRs downloaded, you submit them to the enterprise CA to be signed. After the certificates are returned, they need to be compressed once again into

https://t.me/learningnets

@cisco_in_persian_channel

a tar.gz file before they can be imported. After the files have been tarred and gzipped, you can use the **utils capf cert import** command to upload the signed certificates to the Cisco Unified CM Publisher, as shown in Example 7-8.

Example 7-8 *Using utils capf cert import*

```
admin:utils capf cert import

Importing files.

Source:

 1) Remote Filesystem via FTP
 2) Remote Filesystem via TFTP
 q) quit

Please select an option (1 - 2 or "q" ): 2
File Path: SEP88908D72FADF.csr.tgz
Server: 10.1.10.152
Certificate file imported successfully
Certificate files extracted successfully.
Please wait. Processing 1 files

admin:
```

Finally, you need to verify that the LSC was deployed successfully by logging in to the Cisco UC Administrator GUI, navigating to **Device > Phones**, and searching for a phone that had the certificate installed or upgraded. A successful upgrade should reflect **Upgrade Success**, as shown in Figure 7-43.

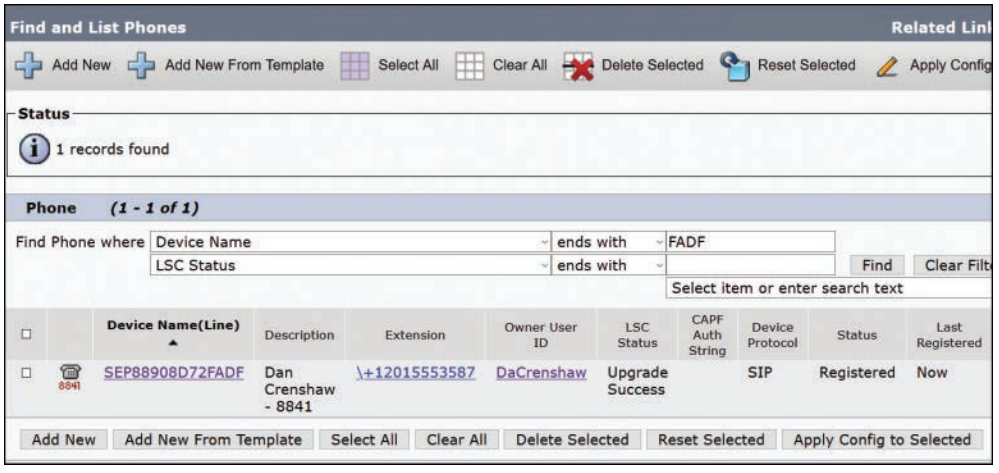


Figure 7-43 *Verifying Offline CA LSC Installation*

https://t.me/learningnets

@cisco_in_persian_channel

Summary

This chapter covered how to secure the signaling and media within the Core UC applications with a focus on certificates. You now know how to verify that a valid encryption license has been installed and recognized by the UC applications. The chapter also provided a primer on PKI and Public Key Cryptography to explain how PKI is used to secure the communications channels that build the framework for secure communications. Cisco's Security by Default methodology, implemented in Cisco Unified CM version 8.0 with enhancements in 10.5, 11.5, and up to the current version of 12.5, shows the relevant enhancements to increase security and ease of use. Using Security by Default, Cisco has built a secure registration process that reduces the methods in which an attack can compromise a system. The use of certificates in this process allows for out-of-the-box compatibility with network authentication servers like ISE to provide network access. These same certificates provide a secure method of establishing a secure communication channel to ensure that only trusted devices can request and install trusted endpoint identity certificates. Finally, you learned the three methods of deploying certificates to endpoints to show when one should be used over another.

Additional Resources

Cisco IP Phone Certificates and Secure Communications - https://tools.cisco.com/security/center/resources/ip_phone_certificates

Configure Automatic Certificate Enrollment and Renewal Via CAPF Online CA - https://www.cisco.com/c/en/us/support/docs/unified-communications/unified-communications-manager-callmanager/214501-configure-automatic-certificate-enrollme.html

Troubleshooting CAPF Online CA - www.cisco.com/c/en/us/support/docs/security-vpn/certificate-authority-ca/214396-troubleshooting-capf-online-ca.html

Q/A for CUCM Phone Certificates (LSC/MIC) - www.cisco.com/c/en/us/support/docs/unified-communications/unified-communicationsmanager-callmanager/200860-Q-A-for-CUCM-PHONE-CERTIFICATES-LSC-MIC.pdf

Preferred Architecture for Cisco Collaboration 12.x Enterprise On-Premises Deployments, CVD - www.cisco.com/c/en/us/td/docs/solutions/CVD/Collaboration/enterprise/12x/120/collbcvd/security.html

https://t.me/learningnets

@cisco_in_persian_channel

This page intentionally left blank

https://t.me/learningnets

@cisco_in_persian_channel

Chapter 8

Securing Cisco Unified Communications Manager (Cisco)

Securing a Cisco Unified Communications Manager involves more than just enabling Secure Real-time Transport Protocol (SRTP) and Transport Layer Security (TLS) to secure the signaling and media between endpoints or configuring and enforcing user access controls. This chapter explains some specific aspects to securing endpoints, conferencing, and Smart Licensing with a specific focus on Cisco Unified CM.

At ACME, Anthony believed he had secured the Unified Communications environment until he received some trouble tickets. The IT security group had been doing scans again and found ports open on the different UC endpoints. They also found instances where RTP packets were traversing the network, contrary to what they had been told about using SRTP and TLS. Once again, Anthony needed to hit the books and identify what was going on so that he could assure the security group that the UC infrastructure complied with policy.

Questions that you should ask:

1. What are the steps to harden an endpoint that registers to Cisco Unified CM?
2. Where in the configuration hierarchy should the configuration settings be made?
3. When you're securing conferencing resources, is enabling SRTP between endpoints enough to secure the communications?
4. What are the different types of conferences within Cisco Unified CM, and do the methods of securing them differ?
5. What is Smart Licensing, and why should you use it? Should you synchronize with Cisco Smart Software Manager or Smart Software Manager On-Prem, or should you use License Reservation?

https://t.me/learningnets

@cisco_in_persian_channel

Endpoint Hardening

Hardening endpoints is the same as hardening an OS or other network device. Endpoint hardening is nothing more than removing and disabling unused features and services. When dealing with endpoints, you need to make some considerations to properly lock down devices and still allow them to function. An example is disabling the PC port on an IP phone. Using the example of disabling a PC port also lets us address some additional caveats to hardening endpoints. When looking at disabling a PC port, you need to consider the location and function of the phone that is having its PC port disabled. Disabling the PC port on a lobby phone makes sense, because in most instances a lobby phone does not have a PC connected to it. However, disabling the PC port on a user's desk phone would mean there would need to be two separate network connections: one for the PC and one for the phone. Some security policies might require separate voice and data network connections. These are infrequent in most environments because one of the benefits of using VoIP is the reduced network infrastructure when voice and data networks are consolidated. This section covers the basics of hardening endpoints to illustrate where the settings are configured and what common settings are used to secure the endpoints.

Where to Configure the Settings

The settings to harden an endpoint can be configured in several different places. Cisco uses a layered approach to applying configuration settings within Cisco Unified CM. This way UC administrators can provide default endpoint profiles systemwide that can then be updated to be either site or device specific.

For example, within Cisco Unified CM, you can define a clusterwide endpoint security policy by configuring the settings under **System > Enterprise Phone Configuration**. By default, these settings are not applied, but as the UC administrator, you do have the ability to configure default settings for all endpoints. At this top level in configuration options, you can configure many options, such as enabling FIPS on the endpoint, disabling the PC port, and configuring 802.1x authentication. Although not every configurable option is available at this level, it does enable you to define a baseline security posture for endpoints.

Diving a little deeper, you can refine the configuration options for groups of phones using Common Phone Profiles. These profiles, located at **Device > Device Settings > Common Phone Profile**, enable you to configure groups of phones with configuration settings that override the Enterprise Phone Configuration settings. The same options configurable with Enterprise Phone Configuration can be configured here.

The previous two methods for configuring endpoints provide a way to ensure endpoints are hardened by default. This can be at the enterprise level where all devices that register to Cisco Unified CM have a specific set of features enabled. Or with the use of Common Phone Profiles, groups of phones at a site or function level can be configured with specific features that override the Enterprise settings.

https://t.me/learningnets

@cisco_in_persian_channel

The last method is at the device level. Configuration settings at the device level take priority over both the Enterprise Phone Configuration settings and the Common Phone Configuration settings. These settings should be configured for one of two reasons: either it is a one-off situation, or the setting does not exist at the Enterprise Phone Configuration or Common Phone Profile levels. For example, one such feature that is available only at the device is the ability to disable the speakerphone.

Features and Services to Consider

There is no hard-and-fast rule when it comes to which features and services need to be configured so that an endpoint can be considered hardened. It is important to consider what the endpoint is used for when defining the hardening criteria. A sample scenario to consider is whether to enable or disable the built-in video camera on an IP phone. In most cases, this feature could be left enabled, with no change to the Enterprise Phone Configuration. However, if the phone is used in a secure environment or where personally identifiable information (PII) could be viewed by unauthorized parties during a video call, disabling it might be prudent. There are a couple of issues to consider here. The first is whether deploying a video-enabled phone is the right choice for that location. If it is and there is a pressing reason to deploy this type of endpoint, it could still be an indicator that these devices should have their video disabled, for example, at the Common Phone Profile. A second example is the PC port. Should it be enabled or disabled? For most endpoints, this port should be enabled because most users connect their workstations to this port for network access. If, however, the endpoint is a public area, like a lobby or elevator, likely the PC port should be disabled.

The following are some of the common settings used to harden endpoints:

- **Gratuitous ARP:** You should disable Gratuitous ARP to reduce the likelihood of ARP spoofing.
- **Web Access:** If the web services are disabled, the phone does not open HTTP port 80 or HTTPS port 443 for monitoring purposes and blocks access to the phone's internal web pages. This is the easiest approach, but it reduces the ability to monitor and maintain the endpoint, which means it is not a feasible solution for some environments. Another option is to restrict access to use HTTPS rather than HTTP. Using this secure protocol and ensuring CA-signed certificates are deployed on the endpoints ensure the traffic is trusted within the enterprise. Last, rather than disabling web access to the phones, a more common approach is to apply access control lists (ACLs) to the phone IP subnets to restrict access to approved networks.
- **PC Voice VLAN:** Cisco phones, by default, forward all traffic received on the phone's network switch port to the PC port. This setting can be disabled if desired. Disabling it can cause problems with applications connected via the PC port that use the voice VLAN.
- **Setting Access:** Setting access can be either Enabled, Disabled, or Restricted. The default, Enabled, allows the end user to access the phone network configuration,

https://t.me/learningnets

@cisco_in_persian_channel

ring type, and volume on the phone. When this setting is set to Restricted, only access to the user preferences and volume settings is available. When it is set to Disabled, setting access is completely disabled; no options appear when Settings is selected. Additionally, the ringer volume cannot be adjusted and volume settings cannot be saved. An option to restrict the network configuration settings of the endpoint while setting access is enabled is to configure the Common Phone Profile with a Local Phone Unlock Password. With this password configured (see Figure 8-1), end users can access the user preferences, but a password is required to access the Administrator settings on the device.

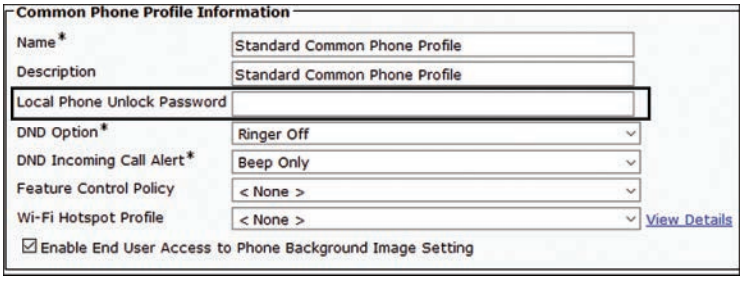


Figure 8-1 *Local Phone Unlock Password*

- **PC Port:** Disabling the PC port should be common practice on endpoints where it is not needed; typically, this would be a lobby or courtesy phone. An extra step to take with devices like this and in general is to use 802.1x for network authentication or even a restricted data VLAN that does not allow network connectivity. This way, if someone unplugs the phone and attempts to connect, that user still does not have network access.

Other settings can be considered when identifying what to configure to lock down an endpoint, such as video calling, speakerphone, and SSH access. Consider where the endpoint will be located and what features are required. If video calling is enabled, is there any sensitive information that might be inadvertently shared? If so, disable video calling or educate the end users to obscure or remove sensitive information when using video. The same process goes for speakerphone.

Secure Conferencing

In Chapter 7, "Encrypting Media and Signaling," we described how to secure the signaling and media between endpoints, but what happens when you begin to use system features and services? Conferencing within Cisco Unified CM is handled by different components, depending on the conference type being invoked. Endpoints can use built-in conferencing resources when features like barge are used. Cisco Unified CM itself has built-in software-based conferencing resources; however, this resource type cannot use SRTP, and it supports a combination of wideband or G.711 a-law and mu-law. External conferencing resources can be configured within Cisco Unified CM as either hardware or

https://t.me/learningnets
@cisco_in_persian_channel

software based. This section covers the behavior and configuration of secure conferencing when hardware-based DSPs and software-based Cisco Meeting Server are used as external conferencing resources.

For a complete listing of the conferencing resources supported by Cisco Unified CM, review the relevant chapter in the System Configuration Guide for Cisco Unified Communications Manager by going to **System Configuration > Configure Conference Bridges** (see www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/admin/12_5_1SU2/systemConfig/cucm_b_system-configuration-guide-1251su2/cucm_b_system-configuration-guide-for-cisco-1251su2_chapter_01100.html).

In a secure conference, all participating endpoints have encrypted signaling and media. Just like between endpoints, a secure conference uses SRTP encryption over a TLS or IPsec connection. Common conference types are ad hoc, Meet-Me, and barge. Depending on the conference type invoked, the behavior of the conference can change based on the security status of the participants.

An encrypted conference can revert to authenticated or nonsecure if an authenticated or nonsecure participant connects to the call. To illustrate this behavior, a secure conference that includes three encrypted connections and one authenticated connection has a conference security status of authenticated. If the authenticated connection drops, the conference transitions to being encrypted. A nonsecure participant who connects to a conference call renders the conference nonsecure. A conference status can also change in several scenarios like when participants chain conferences together, when a held conference call is resumed on another device, when a conference call gets barged, or when a transferred conference call completes to another device.

When an encrypted phone connects to a secure conference bridge, the media streaming between the device and the conference bridge gets encrypted; however, the icon for the conference can be encrypted, authenticated, or nonsecure depending on the security levels of the other participants. A nonsecure status indicates that one of the parties is not secure or cannot be verified.

Cisco endpoints indicate the security status of a conference by displaying a security icon for the security level of the entire conference. The icon is the same status icons used for a secure two-party call.

- A lock icon is displayed when the conference bridge is secure and all participants in the conference are encrypted.
- Older phone models like the 79XX display a shield icon when the conference bridge is secure and at least one of the participants is authenticated. The newer model phones like the 78XX and 88XX do not display this icon, and the call appears nonsecure.

Cisco enables you to verify and maintain the security level of a secure conference through two methods when traditional conferencing methods (DSPs, Unified CM Software Conference Bridges) are used. The first is with ad hoc conference lists. Using conference lists, the conference participants can see the list of participants and their

https://t.me/learningnets

@cisco_in_persian_channel

associated security status. Figure 8-2 shows the display that a conference participant would see when a conference list is used. Participants in the conference with an undesired security status can be removed from the conference. The second method is with Meet-Me conferences with a minimum-security level.

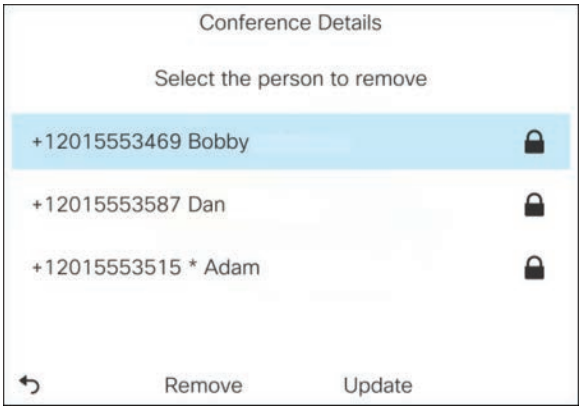


Figure 8-2 *Conference List Security Level*

Ad Hoc Conferencing

An ad hoc conference is initiated when a user invokes the conference feature key or a call using a built-in bridge exceeds more than three participants and is escalated to use either Cisco Meeting Server or hardware DSPs. For this process, the digital certificates that are used should all be signed by the same enterprise CA to simplify the configuration steps.

Secure Conferencing Using Hardware-Based DSPs

Enabling secure conferencing with DSPs involves several steps:

1. Generate a certificate signing request (CSR) on the Cisco IOS gateway.
2. Sign the CSR by the enterprise certificate authority.
3. Import the certificate chain and signed device certificates.
4. Create the secure conferencing resource.
5. Associate the resource with the SCCP CCM group.
6. Configure the secure conference bridge in Cisco Unified CM.

For specific restrictions and prerequisites on enabling media and signaling encryption on DSP farm conferencing, refer to www.cisco.com/c/en/us/td/docs/ios/12_4t/12_4t15/itsdsp.html.

https://t.me/learningnets
@cisco_in_persian_channel

The first step in the process is to create the crypto key that will be used as part of generating the certificate. You accomplish this with the **crypto key generate rsa modulus 2048 label <key>** command. You can add the **exportable** keyword to the end of the command string to allow the certificates (public and private keys) to be exported from the system. It is recommended to export the keys that will have their public key certificates imported into other devices because in the event of a device failure, the exported keys can be imported back into the original device. By doing this, you do not have to request new certificates that then need to be uploaded to other devices. One thing to keep in mind is that if the keys are exported, it is imperative that they are stored in a secure location to prevent them from being compromised.

```
(config)# crypto key generate rsa modulus 2048 label Secure-DSP
The name for the keys will be: Secure-DSP

% The key modulus size is 2048 bits
% Generating 2048 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 8 seconds)
```

With the key generated, the next task is to configure a trust point (see Example 8-1) and create the CSR. This trustpoint is used to store the digital certificates used on the DSP farm. The sequence of commands used is as follows:

- **crypto pki trustpoint:** This command defines the trustpoint that the certificates will use and enters the CA-trustpoint configuration.
- **enrollment terminal:** Enrollment of the certificates takes place at the command line.
- **serial-number none:** This command specifies that a serial number is not included in the certificate request.
- **fqdn none:** This command specifies a fully qualified domain name (FQDN) that is included as unstructuredName in the certificate. This is the FQDN of the router. The **none** keyword specifies that the FQDN is not included in the request.
- **ip-address none:** This command specifies that an IP address is not to be included in the certificate request.
- **subject-name <x.500-name>:** This command specifies the subject name in the certificate request.
- **revocation-check none:** Certificate checking is not required.
- **rsakeypair <key>:** This is the label used when the crypto key was generated.
- **hash {md5 | sha1 | sha256 | sha384 | sha512}:** This command specifies the hash function for the signature that the Cisco IOS client uses. The use of **md5** and **sha1** is no longer recommended.

https://t.me/learningnets

@cisco_in_persian_channel

Example 8-1 *Create Secure CFB Trustpoint*

```
(ca-trustpoint)# crypto pki trustpoint Secure-CFB
(ca-trustpoint)# enrollment terminal
(ca-trustpoint)# serial-number none
(ca-trustpoint)# fqdn none
(ca-trustpoint)# ip-address none
(ca-trustpoint)# subject-name CN=secure-cfb,OU=Cisco Press,O=Practical UC
  Security,L=RTP,ST=NC,C=US
(ca-trustpoint)# revocation-check none
 (ca-trustpoint)# rsakeypair Secure-DSP
(ca-trustpoint)# hash sha256
```

Next, you generate the CSR using the **crypto pki enroll to generate** command and display the CSR to the terminal session (see Example 8-2).

Example 8-2 *Generate Secure CFB Certificate Signing Request*

```
(ca-trustpoint)# crypto pki enroll Secure-DSP
% Start certificate enrollment ..

% The subject name in the certificate will include: CN=secure-cfb,OU=Cisco
  Press,O=Practical UC Security,L=RTP,ST=NC,C=US
% The fully-qualified domain name will not be included in the certificate
Display Certificate Request to terminal? [yes/no]: yes
Certificate Request follows:

MIIC2jCCAcICAQAwdDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAk5DMQwwCgYDVQQH
EwNSVFAxHjAcBgNVBAoTFVByYWN0aWNhbCBVQyBTZWN1cml0eTEUMBIGA1UECxML
Q2lzY28gUHJlc3MxFDASBgNVBAMMC3NlY3VyZS1jZmJ+MIIBIjANBgkqhkiG9w0B
AQEFAAOCAQ8AMIIBCgKCAQEAu4y6PF4SOQ5Zs7Vs+fTQC8AoSo1w2cB/pOokHA4F
oiCc3cbUkjmovgThKw0qFHfamv3QVHgG2ln0PBqugsWe9NycbyjkJBjDwvhf/kWp
jW34bdZJBEOx9NMgGMifORH1tllrm6au2eY9kWSZxsa7ek2vP6oR3sKsBVMgGcHC
5QJu2YeKvwy1IIfdShhhNXBu14gKNq52RH3gXkt/s8eqhtAH8uRltjgVIbLovHIG
6mYfOVVtveVno+zOU4gFtmaTl7ZWKnTvJqa8D8WIp8+WOv5VlaW1yHrOHRjp3y6w
yeN+ls1dSLwlTMjTAKpXtdl8ST/01W5uoe/xAIw2aTZxkQIDAQABoCEwHwYJKoZI
hvcNAQkOMRIwEDAOBgNVHQ8BAf8EBAMCBaAwDQYJKoZIhvcNAQELBQADggEBAHme
KAdNqSASJ/4+NoaQXBm01thPEFtfBJkVhu0aPdIzQJ3dcgflNheU09yO8jrzogZY
sCxxSkWtWS+XNRaGPhJXJHrrk/y54U5w1GiiSWQY98zrd+MSpoaRzX4TTpHXUk4t
hwuTbEYnaj5z9M3Qgut+qnct43HHwDpcCEknKiIEIrhDb9zfQB+4J9wgz++ZZDgi
Uh+EmPkXUuDX+VxSBIslsSbrFRjMfHEbLPnq5TkCcPnHH+1ob77oFF9i6TTJAAzO
ubrpQPBOK2A1GG7bmNxx9+pGa5XSwq0/LlsSlMAHmBGGY5/iOerODtEEX+Qe5NF3
/s6QyFqzzPdhaCWxc0A=

---End - This line not part of the certificate request---

Redisplay enrollment request? [yes/no]: no
```

https://t.me/learningnets

@cisco_in_persian_channel

You can copy the text that comes after **Certificate Request follows:**.

Cisco IOS does not include the standard **begin** and **end** certificate request lines. It is recommended that you add those headers when saving the file. Example 8-3 shows the final output of the CSR with the header and footer additions.

Example 8-3 *Certificate Signing Request Example*

```
-----BEGIN CERTIFICATE REQUEST-----
MIIC2jCCAcICAQAwdDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAk5DMQwwCgYDVQQH
EwNSVFAxHjAcBgNVBAoTFVByYWN0aWNhbCBVQyBTZWN1cml0eTEUMBIGA1UECxML
Q2lzY28gUHJlc3MxFDASBgNVBAMMC3NlY3VyZS1jZmJ+MIIBIjANBgkqhkiG9w0B
AQEFAAOCAQ8AMIIBCgKCAQEAu4y6PF4SOQ5Zs7Vs+fTQC8AoSo1w2cB/pOokHA4F
oiCc3cbUkjmovgThKw0qFHfamv3QVHgG2ln0PBqugsWe9NycbyjkJBjDwvhf/kWp
jW34bdZJBEOx9NMgGMifORH1tllrm6au2eY9kWSZxsa7ek2vP6oR3sKsBVMgGcHC
5QJu2YeKvwy1IIfdShhhNXBu14gKNq52RH3gXkt/s8eqhtAH8uRltjgVIbLovHIG
6mYfOVVtveVno+zOU4gFtmaTl7ZWKnTvJqa8D8WIp8+WOv5VlaW1yHrOHRjp3y6w
yeN+ls1dSLwlTMjTAKpXtdl8ST/01W5uoe/xAIw2aTZxkQIDAQABoCEwHwYJKoZI
hvcNAQkOMRIwEDAOBgNVHQ8BAf8EBAMCBaAwDQYJKoZIhvcNAQELBQADggEBAHme
KAdNqSASJ/4+NoaQXBm01thPEFtfBJkVhu0aPdIzQJ3dcgflNheU09yO8jrzogZY
sCxxSkWtWS+XNRaGPhJXJHrrk/y54U5w1GiiSWQY98zrd+MSpoaRzX4TTpHXUk4t
hwuTbEYnaj5z9M3Qgut+qnct43HHwDpcCEknKiIEIrhDb9zfQB+4J9wgz++ZZDgi
Uh+EmPkXUuDX+VxSBIslsSbrFRjMfHEbLPnq5TkCcPnHH+1ob77oFF9i6TTJAAzO
ubrpQPBOK2A1GG7bmNxx9+pGa5XSwq0/LlsSlMAHmBGGY5/iOerODtEEX+Qe5NF3
/s6QyFqzzPdhaCWxc0A=
-----END CERTIFICATE REQUEST-----
```

Now you need to save this output to a text file and submit it to the enterprise CA to request the digital certificate for the DSP farm. When the digital certificate is signed by the CA, ensure that it is collected in Base 64 format. This allows the digital certificates to be easily imported into the Cisco router housing the DSP farm. Additionally, ensure that the root and all the intermediate digital certificates are downloaded because they need to be uploaded to build the chain of trust for the DSP farm's digital certificate.

One way to verify the signing and root certificates for your certificate is by opening the certificate. In Windows, you can save the certificate with a .cer extension, and after it is opened, navigate to the tab titled Certification Path. The certificates used to sign the device's digital certificate are shown on this tab (see Figure 8-3).

Next, you need to create the trustpoint to be used by the enterprise CA signing certificates.

```
(config)# crypto pki trustpoint CiscoPUCS-CA-Root
(ca-trustpoint)# enrollment terminal
(ca-trustpoint)# revocation-check none
```

https://t.me/learningnets

@cisco_in_persian_channel

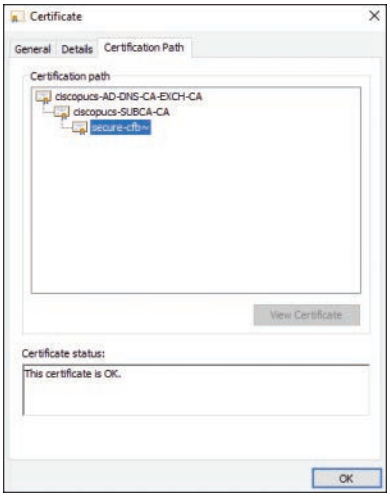


Figure 8-3 *Digital Certificate > Certification Path*

Once the trustpoint is created, the Intermediate certificate (issuer of the device certificate) must be authenticated to the trustpoint that was initially created (Secure-DSP). This step should happen before authenticating the root certificate to the enterprise CA trustpoint that was just configured (CiscoPUCS-CA-Root).

Note After each of the next three steps where the root, intermediate certificates, and device certificates are imported, you need to verify the operation completes successfully before proceeding to the next step.

Now you can start by authenticating the enterprise CA intermediate certificate that signed the actual certificate for the secure DSPs by using the **crypto pki authenticate <trustpoint name>** command, as shown in Example 8-4. A successful import reflects "% Certificate successfully imported" at the end of the command output.

Example 8-4 *Authenticate Intermediate CA into the IOS Trustpoint*

```
(ca-trustpoint)# crypto pki authenticate Secure-DSP

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
MIIFezCCBGOgAwIBAgITHAAAAANnNT673RXs3wAAAAAAAzANBgkqhkiG9w0BAQsF
ADBWMRMwEQYKCZImiZPyLGQBGRYDY29tMRkwFwYKCZImiZPyLGQBGRYJY2lzY29w
dWNzMSQwIgYDVQQDExtjaXNjb3B1Y3MtQUQtRE5TLUNBLUVYQ0gtQ0EwHhcNMTkx
```

https://t.me/learningnets

@cisco_in_persian_channel

```
MTI4MDI1MzQ3WhcNMjExMTI4MDMwMzQ3WjBNMRMwEQYKCZImiZPyLGQBGRYDY29t
MRkwFwYKCZImiZPyLGQBGRYJY2lzY29wdWNzMRswGQYDVQQDExJjaXNjb3B1Y3Mt
U1VCQ0EtQ0EwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCffj2/PZar
0jPzhgvhs3BFe8SrD1ip0nFqhXyVqG74rc9WWv7/Uirx7AWiafIOqDiGrm1TFif4
00Tio/FeRmyyB0ESJqt10oJ8oQoi1T904ixrscV3z1Z2YMVcCFZMdFHuHF6Cfqkx
wlOy35ZE0yaVUyb/VXmWs7zQvqlC4zvkmbkfO0YYhnYKMlRcmj93TG6mmxZvP7Na
CIX2UyzBXJTvoByMNCa28W7AiL6YapeWvi+zhZjEnJxUyWw8+KYy6ciYPgyDgYBl
SVxdBpxYx0+KZvXhvlhZ8621aDF2Lo5/4BytPqq8c4KwzR0pmlG3a3B5uUXGlDgC
5pSyBRa3u0CNAgMBAAGjggJJMIICRTAQBgkrBgEEAYI3FQEEAwIBADAdBgNVHQ4E
FgQUBTp/bxDf6thaIRyE8TWjxthdkN4wGQYJKwYBBAGCNxQCBAweCgBTAHUAYgBD
AEEwDgYDVR0PAQH/BAQDAgGGMA8GA1UdEwEB/wQFMAMBAf8wHwYDVR0jBBgwFoAU
4FH/CNl0I4JIkCo8SrvQVRDFsqQwgeIGA1UdHwSB2jCB1zCB1KCB0aCBzoaBy2xk
YXA6Ly8vQ049Y2lzY29wdWNzLUFELUROUy1DQS1FWENILUNBLENOPUFELUROUy1D
QS1FeGNoLENOPUNEUCxDTj1QdWJsaWMlMjBLZXklMjBTZXJ2aWNlcyxDTj1TZXJ2
aWNlcyxDTj1Db25maWd1cmF0aW9uLERDPWNpc2NvcHVjcyxEQz1jb20/Y2VydGlm
aWNhdGVSZXZvY2F0aW9uTGlzdD9iYXNlP29iamVjdENsYXNzPWNSTERpc3RyaWJ1
dGlvblBvaW50MIHPBggrBgEFBQcBAQSBwjCBvzCBvAYIKwYBBQUHMAKGga9sZGFw
Oi8vL0NOPWNpc2NvcHVjcy1BRC1ETlMtQ0EtRVhDSC1DQSxDTj1BSUEsQ049UHVi
bGljJTIwS2V5JTIwU2VydmljZXMsQ049U2VydmljZXMsQ049Q29uZmlndXJhdGlv
bixEQz1jaXNjb3B1Y3MsREM9Y29tP2NBQ2VydGlmaWNhdGU/YmFzZT9vYmplY3RD
bGFzcz1jZXJ0aWZpY2F0aW9uQXV0aG9yaXR5MA0GCSqGSIb3DQEBCwUAA4IBAQBQ
cDozmao6c+W4CxWZ9eSmZeAQJ5tL85UI3PLeb6MCMuFpdXguXksGnE2/P2p1P+qI
5VSEb3PbTsndu69u/kv30QsvqICI1h8EXtp93Q/wbsPoSuwxHVDTN6LVGKjwnZbi
NbNN5pLwjArZvQBL4DFHOybiK2vGK86RgcVcMbSpXL2fx2lkZKGUqvb0w/80hW7u
2L2iXUn4u4YSljPd/bak/fcwv+PBFonRKJ9CpRvrrOIN1Z055l1Zxiu7BJfuIthr
5guHuCYWzxB7ZejiMF/V37j0EalvPkHeFvDBHI8mfKph4BGjgoUd2pbCumd7aRJK
ldeGtC4Hu3g2H4oVv8aG
-----END CERTIFICATE-----

Certificate has the following attributes:
       Fingerprint MD5: BAC09690 17AB5389 723B540D 77CE2FF5
      Fingerprint SHA1: 037EE4BB 4547C089 CD36F95C B9A805DC F139CC6C
Certificate validated - Signed by existing trustpoint CA certificate.

Trustpoint CA certificate accepted.
% Certificate successfully imported
```

Now you authenticate the enterprise CA root certificate by using the **crypto pki authenticate <trustpoint name>** command, as shown in Example 8-5.

https://t.me/learningnets

@cisco_in_persian_channel

Example 8-5 *Authenticate Root CA in to IOS Trustpoint*

```
(ca-trustpoint)# crypto pki authenticate Secure-DSP

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
MIIFezCCBGOgAwIBAgITHAAAAANnNT673RXs3wAAAAAAAzANBgkqhkiG9w0BAQsF
ADBWMRMwEQYKCZImiZPyLGQBGRYDY29tMRkwFwYKCZImiZPyLGQBGRYJY2lzY29w
dWNzMSQwIgYDVQQDExtjaXNjb3B1Y3MtQUQtRE5TLUNBLUVYQ0gtQ0EwHhcNMTkx
MTI4MDI1MzQ3WhcNMjExMTI4MDMwMzQ3WjBNMRMwEQYKCZImiZPyLGQBGRYDY29t
MRkwFwYKCZImiZPyLGQBGRYJY2lzY29wdWNzMRswGQYDVQQDExJjaXNjb3B1Y3Mt
U1VCQ0EtQ0EwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCffj2/Pzar
0jPzhgvhs3Bfe8SrD1ip0nFqhXyVqG74rc9WWv7/Uirx7AwiafIOqDiGrm1Tfif4
00Tio/FeRmyyB0ESJqt10oJ8oQoi1T904ixrscV3z1Z2YMVcCFZMdFHuHF6Cfqkx
wlOy35ZE0yaVUyb/VXmWs7zQvqlC4zvkmbkfO0YYhnYKMlRcmj93TG6mmxZvP7Na
CIX2UyzBXJTvoByMNCa28W7AiL6YapeWvi+zhZjEnJxUyWw8+Kyy6ciYPgyDgYBl
SvxdBpxYx0+KZvXhvlhZ8621aDF2Lo5/4BytPqq8c4KwzR0pmlG3a3B5uUXGlDgC
5pSyBRa3u0CNAgMBAAGjggJJMIICRTAQBgkrBgEEAYI3FQEEAwIBADAdBgNVHQ4E
FgQUBTp/bxDf6thaIRyE8TWjxthdkN4wGQYJKwYBBAGCNxQCBAweCgBTAHUAYgBD
AEEwDgYDVR0PAQH/BAQDAgGGMA8GA1UdEwEB/wQFMAMBAf8wHwYDVR0jBBgwFoAU
4FH/CNl0I4JikCo8SrvQVRDFsqQwgeIGA1UdHwSB2jCB1zCB1KCB0aCBzoaBy2xk
YXA6Ly8vQ049Y2lzY29wdWNzLUFELUROUy1DQS1FWENILUNBLENOPUFELUROUy1D
QS1FeGNoLENOPUNEUCxDTj1QdWJsaWMlMjBLZXklMjBTZXJ2aWNlcyxDTj1TZXJ2
aWNlcyxDTj1Db25maWd1cmF0aW9uLERDPWNpc2NvcHVjcyxEQz1jb20/Y2VydGlm
aWNhdGVSZXZvY2F0aW9uTGlzdD9iYXNlP29iamVjdENsYXNzPWNSTERpc3RyaWJ1
dGlvblBvaW50MIHPBggrBgEFBQcBAQSBwjCBvzCBvAYIKwYBBQUHMAKGga9sZGFw
Oi8vL0NOPWNpc2NvcHVjcy1BRC1EtlMtQ0EtRVhDSC1DQSxDTj1BSUEsQ049UHVi
bGljJTIwS2V5JTIwU2VydmljZXMsQ049U2VydmljZXMsQ049Q29uZmlndXJhdGlv
bixEQz1jaXNjb3B1Y3MsREM9Y29tP2NBQ2VydGlmaWNhdGU/YmFzZT9vYmplY3RD
bGFzcz1jZXJ0aWZpY2F0aW9uQXV0aG9yaXR5MA0GCSqGSIb3DQEBCwUAA4IBAQBQ
cDozmao6c+W4CxWZ9eSmZeAQJ5tL85UI3Pleb6MCMuFpdXguXksGnE2/P2p1P+qI
5VSEb3PbTsndu69u/kv30QsvqICI1h8Extp93Q/wbsPoSuwxHVDTN6LVGKjwnZbi
NbNN5pLwjArZvQBL4DFHOybiK2vGK86RgcVcMbSpXL2fx2lkZKGUqvb0w/80hW7u
2L2iXUn4u4YsljPd/bak/fcwv+PBFonRKJ9CpRvrrOIN1Z055l1Zxiu7BjfuIthr
5guHuCYWzxB7ZejiMF/V37j0EalvPkHeFvDBHI8mfKph4BgjgoUd2pbCumd7aRJK
ldeGtC4Hu3g2H4oVv8aG
-----END CERTIFICATE-----

Certificate has the following attributes:
       Fingerprint MD5: BAC09690 17AB5389 723B540D 77CE2FF5
      Fingerprint SHA1: 037EE4BB 4547C089 CD36F95C B9A805DC F139CC6C
```

https://t.me/learningnets

@cisco_in_persian_channel

```
Certificate validated - Signed by existing trustpoint CA certificate.

Trustpoint CA certificate accepted.
% Certificate successfully imported
```

The last step is to import the device's certificate by using the **crypto pki import <trustpoint name> certificate** command. The command output is shown in Example 8-6.

Example 8-6 *Import Identity Certificate for the Secure DSPs*

```
(config)# crypto pki import Secure-DSP certificate
% The fully-qualified domain name will not be included in the certificate

Enter the base 64 encoded certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
MIIFeDCCBGCgAwIBAgITWAAAABrCsLGV0iRyUQAAAAAAGjANBgkqhkiG9w0BAQsF
ADBNMRMwEQYKCZImiZPyLGQBGRYDY29tMRkwFwYKCZImiZPyLGQBGRYJY2lzY29w
dWNzMRswGQYDVQQDExJjaXNjb3B1Y3MtU1VCQ0EtQ0EwHhcNMjAwNjA1MDMzODQ2
WhcNMjExMTI4MDMwMzQ3WjB0MQswCQYDVQQGEwJVUzELMAkGA1UECBMCTkMxDDAK
BgNVBAcTA1JUUDEeMBwGA1UEChMVUHJhY3RpY2FsIFVDIFNlY3VyaXR5MRQwEgYD
VQQLEwtDaXNjbyBQcmVzczEUMBIGA1UEAwwLc2VjdXJlLWNmYn4wggEiMA0GCSqG
Sib3DQEBAQUAA4IBDwAwggEKAoIBAQC7jLo8XhI5DlmztWz59NALwChKjXDZwH+k
6iQcDgWiIJzdxtSSOai+BOErDSoUd9qa/dBUeAbaWfQ8Gq6CxZ703JxvKOQkGMPC
+F/+RamNbfht1kkEQ7H00yAYyJ85EfW2Wwubpq7Z5j2RZJnGxrt6Ta8/qhHewqwF
***Output Omitted***
gcUwgcKggb+ggbyGgblsZGFwOi8vL0NOPWNpc2NvcHVjcy1TVUJDQS1DQSxDTj1z
dWJjYSxDTj1DRFAsQ049UHVibGljJTIwS2V5JTIwU2VydmljZXMsQ049U2Vydmlj
ZXMsQ049Q29uZmlndXJhdGlvbixEQz1jaXNjb3B1Y3MsREM9Y29tP2NlcnRpZmlj
YXRlUmV2b2NhdGlvbkxpc3Q/YmFzZT9vYmplY3RDbGFzcz1jUkxEaXN0cmlidXRp
b25Qb2ludDCBxgYIKwYBBQUHAQEEgbkwgbYwgbMGCCsGAQUFBzAChoGmbGRhcDov
Ly9DTj1jaXNjb3B1Y3MtU1VCQ0EtQ0EsQ049QUlBLENOPVB1YmxpYyUyMEtleSUy
MFNlcnZpY2VzLENOPVNlcnZpY2VzLENOPUNvbmZpZ3VyYXRpb24sREM9Y2lzY29w
dWNzLERDPWNvbT9jQUNlcnRpZmljYXRlP2Jhc2U/b2JqZWN0Q2xhc3M9Y2VydGlm
aWNhdGlvbkF1dGhvcml0eTAhBgkrBgEEAYI3FAIEFB4SAFcAZQBiAFMAZQByAHYA
ZQByMBMGA1UdJQQMMAoGCCsGAQUFBwMBMA0GCSqGSIb3DQEBCwUAA4IBAQBHsNpF
BpiIuUPJfuFQrJegU2zMUKTvSk2OksTmvskhuYaQ1c1gPYX4KpXmqclkJ9EyjtLc
tuj+lelBAsfT16gInmXm2dUjh8kJTgj8eB6Dj5810SxupLQvRcrUMcjRwT3owBRd
LvPh1LobG3GrPvw8Q78QD9ErsUHuDbFk9dvVmYR83KgVLDWyPPyyRTxNXg86sr2V
8m31zss6uziXkhGNjP89idnVHkJa7Qcjt2A0RaMHMBYJPNaec27/1CixcM4T7WQ0
qxmhs74m1Ib888ll2wYKtbw/aAm3G5p1xJ4j0gf87CfzokviisBWcQL02NxzIZqn
2L5HSJmcJSWUFmro
-----END CERTIFICATE-----

% Router Certificate successfully imported
```

https://t.me/learningnets

@cisco_in_persian_channel

The certificates can be verified with the **show crypto pki certificates <trustpoint name>** command shown in Example 8-7. For troubleshooting purposes, this command would provide useful information to verify that the proper certificates are being used (serial number) and that the name used to register the conference bridge is correct (Subject Name).

Example 8-7 *Show Certificate Chain of Trust*

```
# show crypto pki certificate Secure-DSP
Certificate
  Status: Available
  Certificate Serial Number (hex): 580000001AC2B0B195D224725100000000001A
  Certificate Usage: General Purpose
  Issuer:
    cn=ciscopucs-SUBCA-CA
    dc=ciscopucs
    dc=com
  Subject:
    Name: secure-cfb
    cn=secure-cfb
    ou=Cisco Press
    o=Practical UC Security
    l=RTP
    st=NC
    c=US
  CRL Distribution Points:
    ldap:///CN=ciscopucs-SUBCA-CA,CN=subca,CN=CDP,CN=Public%20Key%20Services,
  CN=Services,CN=Configuration,DC=ciscopucs,DC=com?certificateRevocationList?base?
  objectClass=cRLDistributionPoint
  Validity Date:
    start date: 23:38:46 est Jun 4 2020
    end   date: 22:03:47 est Nov 27 2021
  Associated Trustpoints: Secure-DSP

CA Certificate
  Status: Available
  Certificate Serial Number (hex): 1C0000000367353EBBDD15ECDF000000000003
  Certificate Usage: Signature
  Issuer:
    cn=ciscopucs-AD-DNS-CA-EXCH-CA
    dc=ciscopucs
    dc=com
  Subject:
    cn=ciscopucs-SUBCA-CA
    dc=ciscopucs
    dc=com
```

https://t.me/learningnets

@cisco_in_persian_channel

```
CRL Distribution Points:
  ldap:///CN=ciscopucs-AD-DNS-CA-EXCH-CA,CN=AD-DNS-CA-Exch,CN=CDP,CN=Public%
20Key%20Services,CN=Services,CN=Configuration,DC=ciscopucs,DC=com?certificate
RevocationList?base?objectClass=cRLDistributionPoint
Validity Date:
  start date: 21:53:47 est Nov 27 2019
  end   date: 22:03:47 est Nov 27 2021
Associated Trustpoints: Secure-DSP
```

Next, you need to configure the secure DSP resources. Currently, although IOS-based DSPs support secure transcoders and media termination points, they are not supported by Cisco Unified CM.

The first step in configuring the secure DSPs is to enable them. Using Example 8-8 as a reference, you use the **voice-card <slot>** and then **dspfarm** commands to enable the DSPs. The **show dspfarm dsp** command can be used to show the DSP status after they have been enabled.

- **Voice-card:** Configure a specific voice card.
- **dspfarm:** Enable the DSP farm feature for this voice card.

Example 8-8 *Enable DSPFarm*

```
(config)# voice-card 0
(config-voicecard)# dspfarm
```

Note If the conferencing resources already exist but are not configured with security, you need to delete them from the IOS configuration and add them back with the **security** keyword.

A best practice when working with DSP resources is to include only the codecs that are used in the environment. This helps provide more sessions because the number of sessions available depends on the number of DSPs and the complexity of the enabled codecs. When secure resources are enabled, the maximum number of sessions is reduced to account for this feature being enabled.

You create the secure conference DSP resource by using the **dspfarm profile <profile number> conference security** command. The DSP farm profile specifies the codecs that the conference bridge supports, the maximum number of sessions, and the trustpoint used to secure the signaling and media. Most importantly, DSP farm profiles are shut down by default, so you need to enable them; otherwise, they never register.

```
(config)# dspfarm profile 1 conference security
(config-dspfarm-profile)# trustpoint Secure-DSP
```

https://t.me/learningnets

@cisco_in_persian_channel

```
(config-dspfarm-profile)# maximum sessions 10
(config-dspfarm-profile)# associate application SCCP
(config-dspfarm-profile)# no shutdown
```

The last portion of the IOS configuration when enabling secure conferencing is to enable and configure SCCP. A reference for this is provided with Example 8-9. The configuration here is the same as a normal conference bridge where SCCP is bound to a local interface, the Unified CMs that SCCP communicates with are defined, and the DSP farm profile is associated with an SCCP CCM group.

The sequence of commands used is as follows:

- **sccp local interface-type interface-number:** Selects the local interface that SCCP applications (transcoding and conferencing) use to register with Unified CM.
- **sccp ccm {ip-address | dns} identifier identifier-number:** Specifies the Unified CM node that is configured within the Unified CM Group applied to the conference bridge. There should be an entry for each Unified CM node in the Unified CM group that is applied.
- **sccp:** Enables the SCCP protocol and its related applications.
- **sccp ccm group group-number:** Creates a Unified CM group and enters SCCP Unified CM configuration mode.
- **bind interface-type interface-number:** Binds an interface to a Cisco Unified Communications Manager group.
- **associate ccm identifier-number priority priority-number:** Associates a Unified CM with a Unified CM group and establishes the Unified CM's priority within the group.
- **associate profile profile-identifier register device-name:** Associates a DSP farm profile with a Unified CM group. The **profile-identifier** should be the same as the subject name (CN) field in the trustpoint.

Example 8-9 *Secure Conference Bridge SCCP Configuration*

```
(config)# sccp local GigabitEthernet0/0
(config)# sccp ccm 10.1.10.81 identifier 1 version 7.0
(config)# sccp ccm 10.1.10.82 identifier 2 version 7.0
(config)# sccp
(config)# sccp ccm group 3
(config-sccp-ccm)# associate ccm 1 priority 1
(config-sccp-ccm)# associate ccm 2 priority 2
(config-sccp-ccm)# associate profile 1 register secure-cfb
```

https://t.me/learningnets

@cisco_in_persian_channel

The next portion of the configuration does not differ from the standard configuration to enable DSP-based resources. The only caveat is that the name used to register the conference bridge should be the common name (CN) of the device certificate. This is the same name used in Cisco Unified CM to register the conference bridge.

Now that the IOS voice gateway has been configured, the next phase is to configure the conference bridge in Cisco Unified CM. In Cisco Unified CM, you navigate to the Cisco Unified CM Administration GUI and select **Media Resources > Conference Bridge**. The new secure conference bridge is created as an IOS Enhanced Conference Bridge.

When the conference bridge is configured, the device security mode should be set to Encrypted Conference Bridge. This setting allows a conference to be encrypted, authenticated, or nonsecure based on the lowest security level of the participants.

When it is saved, the conference bridge should reflect a valid registration status (see Figure 8-4).

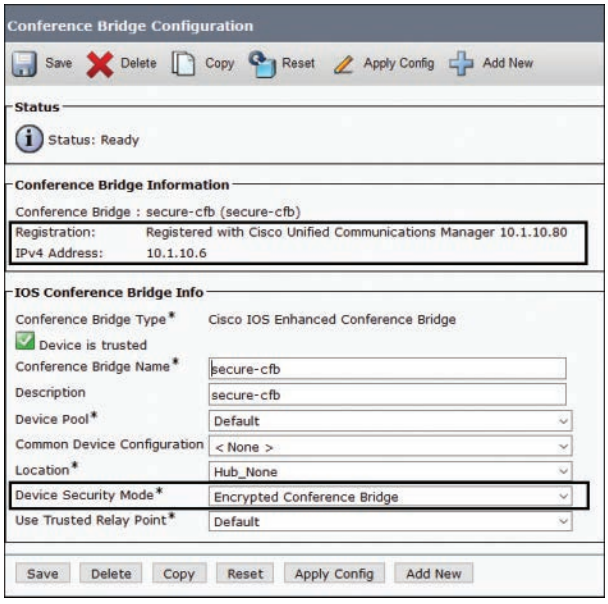


Figure 8-4 *Cisco Unified CM Secure Conference Bridge Creation*

The **show sccp** command can also be run from the IOS voice gateway to verify the current registration status of the conference bridge and whether it is encrypted (see Example 8-10).

https://t.me/learningnets

@cisco_in_persian_channel

Example 8-10 *Verify SCCP State and CFB Status*

```
# show sccp
SCCP Admin State: UP
Gateway Local Interface: GigabitEthernet0/0
        IPv4 Address: 10.1.10.6
        Port Number: 2000
IP Precedence: 5
User Masked Codec list: None
Call Manager: 10.1.10.82, Port Number: 2000
                Priority: N/A, Version: 7.0, Identifier: 3
                Trustpoint: N/A
Call Manager: 10.1.10.81, Port Number: 2000
                Priority: N/A, Version: 7.0, Identifier: 2
                Trustpoint: N/A
Call Manager: 10.1.10.80, Port Number: 2000
                Priority: N/A, Version: 7.0, Identifier: 1
                Trustpoint: N/A

Conferencing Oper State: ACTIVE - Cause Code: NONE
Active Call Manager: 10.1.10.80, Port Number: 2443
TCP Link Status: CONNECTED, Profile Identifier: 1
Security
 Signaling Security: ENCRYPTED TLS
Media Security: SRTP
Supported crypto suites: AES_CM_128_HMAC_SHA1_32
Reported Max Streams: 80, Reported Max OOS Streams: 0
Supported Codec: g711ulaw, Maximum Packetization Period: 30
Supported Codec: g729ar8, Maximum Packetization Period: 60
Supported Codec: g729abr8, Maximum Packetization Period: 60
Supported Codec: g729r8, Maximum Packetization Period: 60
Supported Codec: g729br8, Maximum Packetization Period: 60
Supported Codec: rfc2833 dtmf, Maximum Packetization Period: 30
Supported Codec: rfc2833 pass-thru, Maximum Packetization Period: 30
Supported Codec: inband-dtmf to rfc2833 conversion, Maximum Packetization Period: 30
TLS : ENABLED
```

Secure Conferencing Using Cisco Meeting Server

Starting with Cisco Unified CM version 10.5, the ability to leverage Cisco's Meeting Server platform for ad hoc resources has reduced the cost of implementing ad hoc resources and greatly expanded the capacity of available conferencing resources.

This section covers enabling Meeting Server as a secure ad hoc conferencing resource. Although the design and clustering of Meeting Server are beyond the scope of this book,

https://t.me/learningnets

@cisco_in_persian_channel

Chapter 10, "Securing Cisco Meeting Server," does provide the foundational information needed to secure Cisco Meeting Server that is deployed in a clustered environment. For this example, Meeting Server is configured as a single-server deployment.

There are two phases to configuring Meeting Server to act as an ad hoc conferencing resource in Cisco Unified CM. The first step is to configure Meeting Server with a valid API user and certificates signed by the enterprise CA for call bridge and web admin. The second phase is to configure the Meeting Server in Cisco Unified CM as a conference bridge with a SIP trunk from Cisco Unified CM to Meeting Server.

To start the configuration of Meeting Server, you need to generate CA-signed certificates that have both server and client authentication for enhanced key usage. Depending on the specific IT security policy, separate certificates for each service (call bridge/web admin) may be required. In most instances, though, a single combined certificate suffices and reduces the complexity of the deployment. These steps are provided for reference because the use of certificates to secure the signaling, media, and traffic is considered a prerequisite for this configuration. Chapter 10 provides a more robust explanation of the certificate generation process use cases and deployment for Meeting Server.

The first step here is to connect to the Mainboard Management Processor (MMP) in Meeting Server and issue the **pki csr <file name> CN:<common name> subjectAltName:<subject alternative names>** command. The common name should be the FQDN of the server that is resolvable via the DNS and IP addresses of the Meeting Server, and the subject alternate names should be the domain name and can contain other addresses or urls such as join.domain.com (assuming both web admin and call bridge use the same interface).

```
hq-adhoc-cms> pki csr secureconf CN:hq-adhoc-cms.ciscopucs.com
subjectAltName:ciscopucs.com,10.1.10.94
..............
.....
Created key file secureconf.key and CSR secureconf.csr
CSR file secureconf.csr ready for download via SFTP
```

This command creates .key and .csr files that are used to create the certificates for the services.

Next, you copy the .csr file from the Meeting Server using SFTP and submit it to the enterprise CA so that it can be signed.

Note You must ensure that the template used to sign the CSR contains both the Server Authentication and Client Authentication roles.

Once the CSR has been signed, you collect the new digital certificate as a Base 64 along with the root and intermediate certificates in the certificate path. The services certificate should have .cer as its file extension for consistency.

https://t.me/learningnets

@cisco_in_persian_channel

The root and intermediate certificates must be combined (certbundle.cer) so that they can be uploaded to the Meeting Server with the services certificate. You accomplish this by opening the root and intermediate certificates with a text editor and copying and pasting the contents into a new file. The root certificate should be first, followed sequentially by the remaining intermediate certificates. The contents of the new bundled certificate should look like Example 8-11.

Example 8-11 *CMS Combined Cert Bundle*

```
-----BEGIN CERTIFICATE-----
MIIDhzCCAm+gAwIBAgIQMAe3B21axIZGiqHmifSuNjANBgkqhkiG9w0BAQsFADBW
MRMwEQYKCZImiZPyLGQBGRYDY29tMRkwFwYKCZImiZPyLGQBGRYJY2lzY29wdWNz
MSQwIgYDVQQDExtjaXNjb3B1Y3MtQUQtRE5TLUNBLUVYQ0gtQ0EwHhcNMTkxMTI3
MDMwMzEyWhcNMjQxMTI3MDMxMzEyWjBWMRMwEQYKCZImiZPyLGQBGRYDY29tMRkw
FwYKCZImiZPyLGQBGRYJY2lzY29wdWNzMSQwIgYDVQQDExtjaXNjb3B1Y3MtQUQt
RE5TLUNBLUVYQ0gtQ0EwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC1
z73qVQqKAoZ2MjBHWw2gUDG5rO6xMIqHi4/IFTX2QGWC7cRtXMK3P02pTQa9pleO
5XdBO1jLUPytzo48kwJVO7Oq2gdd9uz8PxWgiblLxXGldKjD9X+Yl/6Hu70w+811
912RQ10Ae3IncTA+LbF8j8VNGPlQlzvSGNkRco4R/JAJNiKFw0LfccssTR+BIR8u
QylU5dwQbAIoeQo6H+thSjN+VU3kjG8ve35z2hO8woucSz99vfkCHDw0FB3Xsg0T
o0byehjZYKO+ZIrNTYZflfiMB3c0iCSi5udvniZm+lnVkmZD448/Vma6gA==
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIFezCCBGOgAwIBAgITHAAAAANnNT673RXs3wAAAAAAAzANBgkqhkiG9w0BAQsF
ADBWMRMwEQYKCZImiZPyLGQBGRYDY29tMRkwFwYKCZImiZPyLGQBGRYJY2lzY29w
dWNzMSQwIgYDVQQDExtjaXNjb3B1Y3MtQUQtRE5TLUNBLUVYQ0gtQ0EwHhcNMTkx
MTI4MDI1MzQ3WhcNMjExMTI4MDMwMzQ3WjBNMRMwEQYKCZImiZPyLGQBGRYDY29t
MRkwFwYKCZImiZPyLGQBGRYJY2lzY29wdWNzMRswGQYDVQQDExJjaXNjb3B1Y3Mt
U1VCQ0EtQ0EwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCffj2/PZar
0jPzhgvhs3BFe8SrD1ip0nFqhXyVqG74rc9WWv7/Uirx7AWiafIOqDiGrm1TFif4
00Tio/FeRmyyB0ESJqt10oJ8oQoi1T904ixrscV3z1Z2YMVcCFZMdFHuHF6Cfqkx
wlOy35ZE0yaVUyb/VXmWs7zQvqlC4zvkmbkfO0YYhnYKMlRcmj93TG6mmxZvP7Na
NbNN5pLwjArZvQBL4DFHOybiK2vGK86RgcVcMbSpXL2fx2lkZKGUqvb0w/80hW7u
2L2iXUn4u4YSljPd/bak/fcwv+PBFonRKJ9CpRvrrOIN1Z055l1Zxiu7BJfuIthr
5guHuCYWzxB7ZejiMF/V37j0EalvPkHeFvDBHI8mfKph4BGjgoUd2pbCumd7aRJK
ldeGtC4Hu3g2H4oVv8aG
-----END CERTIFICATE-----
```

You can save this file with any name, but the extension should be .crt, .cer, or .pem.

Now you upload the certificates to Meeting Server using SFTP and then apply the certificates to the two services, as shown in Example 8-12. First, you use the **webadmin certs <key-file> <crt-file> <cert-bundle>** command to apply the certificates to the web admin service. Then you use the **callbridge certs <key-file> <crt-file> <cert-bundle>** command to apply the certificates to the call bridge service.

https://t.me/learningnets

@cisco_in_persian_channel

Example 8-12 *CMS Configure Webadmin PKI*

```
hq-adhoc-cms> webadmin disable
hq-adhoc-cms> webadmin certs combined.key combined.cer certbundle.cer
hq-adhoc-cms> webadmin enable
SUCCESS: TLS interface and port configured
SUCCESS: Key and certificate pair match
SUCCESS: certificate verified against CA bundle
hq-adhoc-cms> callbridge certs combined.key combined.cer certbundle.cer
hq-adhoc-cms> callbridge restart
SUCCESS: listen interface configured
SUCCESS: Key and certificate pair match
SUCCESS: certificate verified against CA bundle
```

Cisco Unified CM dynamically creates ad hoc conferences in Meeting Server using APIs. Consequently, you need to create a new account that has the API role assigned to it:

```
hq-adhoc-cms> user add apiadmin api
Please enter new password:
Please enter new password again:
Success
```

To ensure that the signaling and media of the ad hoc conference is secured, you need to enable SIP media encryption in Cisco Meeting Server (see Figure 8-5). You do so under **Configuration > Call Settings** and by selecting either **Allowed** or **Required**. If all endpoints will use SRTP, select the option for Required. If, however, you expect to have a mix of endpoints using SRTP and RTP, set the option to Allowed. The default setting of Disabled does not allow the media to be encrypted.

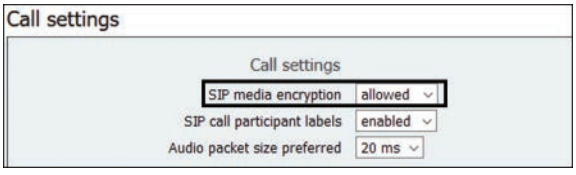


Figure 8-5 *CMS SIP Media Encryption*

The next phase of the configuration takes place in Cisco Unified CM and begins with uploading the services certificate that was just created in Cisco Meeting Server so that a chain of trust can be established. This step is optional, however, if the root and intermediate certificates are already present in the CallManager-trust certificate store. When the root and possibly intermediate certificates are present in Unified CM, you need to create an SIP trunk between Unified CM and Meeting Server. Lastly, you need to define the Meeting Server as a conference bridge in Cisco Unified CM.

https://t.me/learningnets
@cisco_in_persian_channel

Note If the Meeting Server services certificate is not self-signed and it is signed by a certificate authority, it does not need to be uploaded to the Unified CM CallManager-Trust store. At a minimum in this instance, the root CA of the issuing CA needs to be present.

Now you can log in to the Cisco Unified OS Administration and navigate to **Security > Certificate Management**. From here, you select **Upload Certificate/Certificate Chain** and upload the root and intermediate enterprise CA certificates that are not present in the CallManager-Trust store. You may need to repeat this process on each node in the Cisco Unified CM cluster if the certificates are not replicated from the Publisher to remaining nodes in the cluster.

After the services restart, log in to the Cisco Unified CM Administration GUI and create a new SIP Trunk Security Profile (**System > Security > SIP Trunk Security Profile**). This profile is applied to the SIP trunk used to communicate with the Meeting Server for ad hoc conferencing. Table 8-1 provides the recommended settings to configure on the SIP Trunk Security Profile.

Table 8-1 *SIP Trunk Security Profile Configuration*

SIP Trunk Security Profile Information	
Name*	Meaningful profile name (e.g., Ad-Hoc-CMS)
Description	Useful profile description (e.g., Ad Hoc Conferencing Resources)
Device Security Mode	Encrypted
Incoming Transport Type*	TLS
Outgoing Transport Type	TLS
Secure Certificate Subject or Subject Alternate Name	FQDN of CMS callbridge
Incoming Port*	5061 (default value)
Accept replaces header	checked

Figure 8-6 reflects the standard SIP Trunk Security Profile that is configured to support using Meeting Server as an ad hoc conference resource.

Next, you navigate to **Device > Trunk** and select **Add New** to create a new SIP Trunk that points to Meeting Server. Select **SIP Trunk** from the Trunk Type drop-down. The default options **SIP** for Device Protocol and **None** for Trunk Service Type* do not need to be changed. In an enterprise UC environment, the standard SIP trunk configuration settings can be and should be very specific. Configuration settings like Device Pool, Location, and Incoming CSS are important, so you must ensure that these settings are consistent with the existing UC environment.

https://t.me/learningnets

@cisco_in_persian_channel

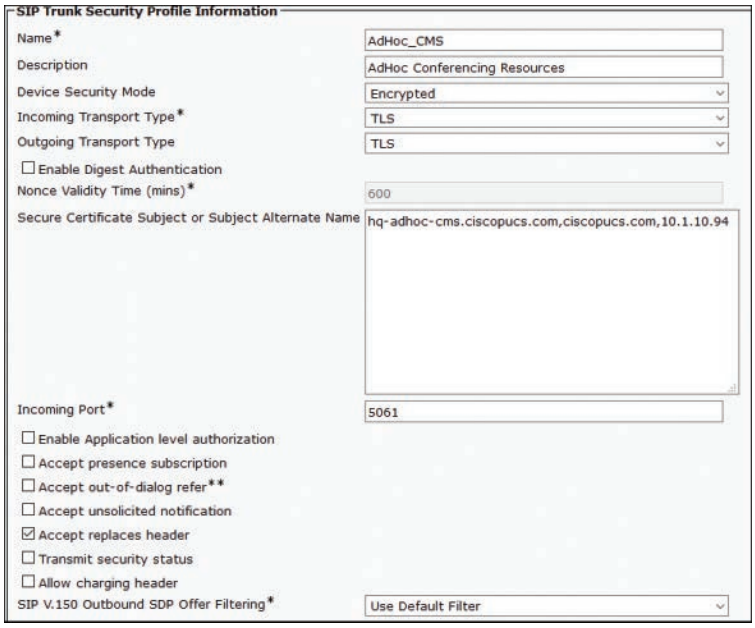


Figure 8-6 *SIP Trunk Security Profile*

Figure 8-7 illustrates the areas of interest on the SIP Trunk to Meeting Server. For our purposes, the important settings are

- **SRTP Allowed:** When this box is checked, Encrypted TLS needs to be configured in the network to provide end-to-end security. Failure to do so exposes keys and other information.
- **Consider Traffic on This Trunk Secure*:** Set to **When using both sRTP and TLS**.
- **Destination Address and Destination Port:** This setting specifies the Meeting Server FQDN that will be used for ad hoc conferencing connecting on port 5061.
- **SIP Trunk Security Profile:** This setting specifies the profile previously created that specifies encryption will be used on a TLS connection.
- **SIP Profile:** This setting specifies SIP-specific settings and options to use.
- **Normalization Script:** This setting provides a method of normalizing the SIP packets sent to Meeting Server. Current versions of Unified CM use the normalization script called cisco-meeting-server-interop. Depending on the version of Unified CM that you are working with, you may not have this normalization script. If this is the case, you can use the normalization script called cisco-telepresence-conductor-interop.

https://t.me/learningnets

@cisco_in_persian_channel

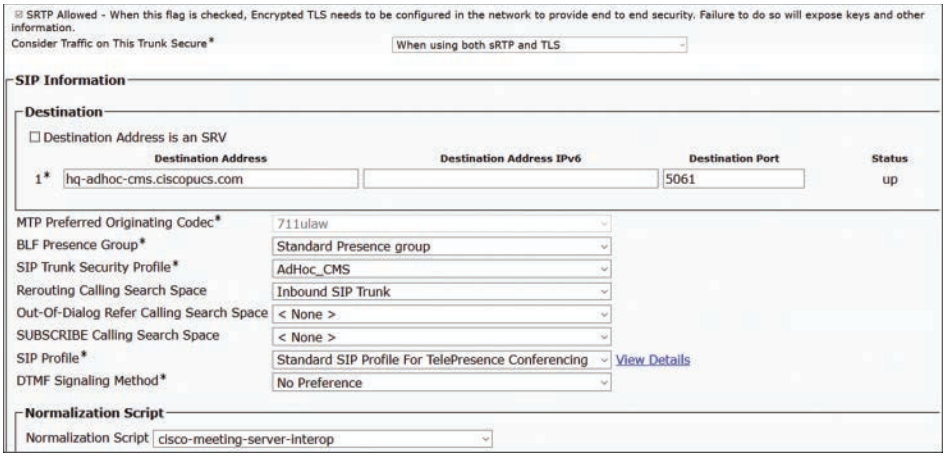


Figure 8-7 *Cisco Unified CM SIP Trunk Configuration*

The Meeting Server ad hoc conference bridge is created by navigating to **Media Resources > Conference Bridge** and selecting **Add New**. The conference bridge type being created should be **Cisco Meeting Server**. The conference bridge is then configured with the recently created SIP Trunk and uses the web admin interface with the API user account that was created previously. The configuration here brings up an interesting point. When Cisco Meeting Server is used as an ad hoc conference resource, there are two secure connections between Unified CM and Cisco Meeting Server. The first is the SIP trunk itself, because the SIP Trunk Security Profile that was applied is configured for encryption. This means that the signaling is authenticated (TLS) and the data is encrypted (SRTP). The second is the API traffic that is used by the Cisco Meeting Server Conference Bridge is configured to communicate using HTTPS. Figure 8-8 shows the configuration of the Cisco Meeting Server Conference Bridge.

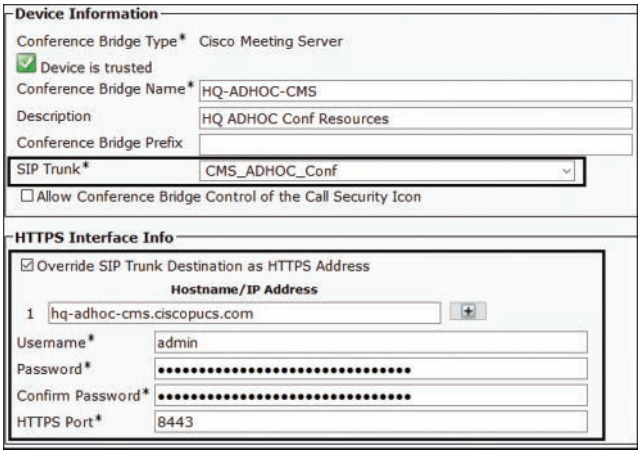


Figure 8-8 *Configured Cisco Meeting Server Ad Hoc Conference Bridge*

https://t.me/learningnets

@cisco_in_persian_channel

This next step is to identify the port that should be used for the conference bridge API. Depending on whether webrtc (web bridge) is enabled on the Meeting Server, you may need to use a different port for HTTPS. This is common practice to simplify the login process for the Cisco Meeting Web App or Cisco Meeting App. You can verify the port that is used by running the **webadmin status** command from the Meeting Server MMP.

```
hq-adhoc-cms> webadmin status
Enabled                 : true
TLS listening interface : a
TLS listening port      : 8443
Key file                : combined.key
Certificate file        : combined1.cer
CA Bundle file          : certbundle.cer
HTTP redirect           : Disabled
STATUS                  : webadmin running
```

Now that the Meeting Server conference bridge is configured, it can be added to the existing media resource groups and media resource group lists within the enterprise UC environment.

Meet-Me Secure Conferencing

Meet-Me conferences are static on-demand conferences that can be configured so that they require a minimum level of security. The device that initiates the Meet-Me conference must meet the required security level; otherwise, the conference will fail. If an endpoint tries to join the conference and does not meet the security level that is configured, the call will fail. A Meet-Me conference that is configured as nonsecure accepts all calls and has a status of nonsecure. Figure 8-9 shows the configuration of a Meet-Me conference with the minimum security level configured.

- **Ad hoc conference:** If a nonsecure or an authenticated ad hoc conference attempts to join an encrypted Meet-Me conference, the call will fail.
- **Barge:** When a barge is attempted on a participant in a secured Meet-Me conference, the barge caller must meet the minimum-security requirements of the Meet-Me conference. When a nonsecure phone attempts to barge a call that is in authenticated Meet-Me conference, the barged device is downgraded, and both the device and the barged caller are dropped from the Meet-Me conference.
- **Shared line:** A caller on a shared line that has connected to a Meet-Me conference with a configured minimum-security level restricts the other devices with the shared line. When a Meet-Me call is placed on hold, it cannot be resumed from a shared line unless the phone resuming the call meets the minimum-security level configured.

https://t.me/learningnets

@cisco_in_persian_channel

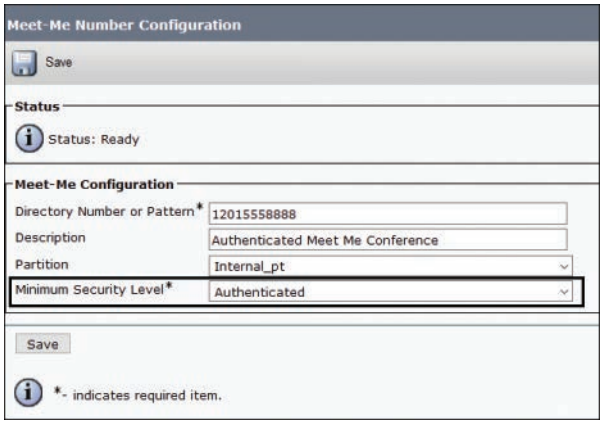


Figure 8-9 *Authenticated Meet-Me Conference*

Conference Now

Conference Now can utilize secured conference bridges and can dynamically upgrade and downgrade the security level of the conference based on the participants. Unlike ad hoc conferencing, Conference Now does not provide a method to list the conference participants, but the overall conference status is displayed like a normal endpoint-to-endpoint call (see Figure 8-10).

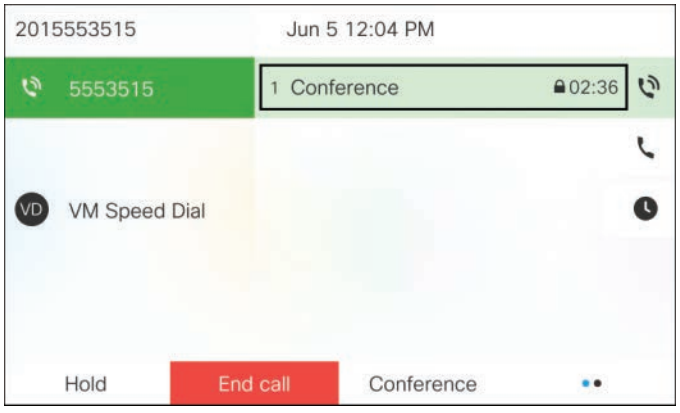


Figure 8-10 *Conference Now Security Level*

Smart Licensing

Cisco has begun implementing Smart Licensing across its product portfolio to ease the burden that licensing customers have voiced over the years. This section covers some

https://t.me/learningnets
@cisco_in_persian_channel

of the security considerations surrounding the information that Smart Licensing shares with Cisco and the methods available to implement Smart Licensing. Consult the following references to build a foundational understanding of Smart Software Licensing before continuing:

- Cisco Smart Software Licensing: Overview and Deployment Options White Paper: www.cisco.com/c/en/us/products/collateral/software/smart-accounts/white-paper-c11-741659.html
- Care and Feeding of Smart Licensing BRKARC-2034: www.ciscolive.com/global/on-demand:library.html?ssoToken=y9z15986237578780019dr8nn2&showMyInterest=true#/session/1542224299905001rWD6
- Transitioning to Smart Licensing: Cisco Unified Communications Manager and Cisco Unity Connection 12.x BRKUCC-2725: www.ciscolive.com/global/on-demand-library.html?ssoToken=y9z15986237578780019dr8nn2&showMyInterest=true#/session/1542224335413001rk8J

From a simplified point of view, you can think of Smart Accounts as containers where Smart Licenses or traditional licenses can be deposited. This means that instead of licenses being issued to an individual Cisco Connection Online ID (CCO Id), they are issued at an organizational level. Because all Cisco products are moving to Smart Licensing, Smart Accounts will be required for all licensing eventually. Smart Accounts also allow customers to manage, move, store, and view software assets that they are entitled to.

How do Smart Accounts address security?

- **Utilization visibility:** Smart Accounts enable you to see where licenses are being used and provide a way to audit usage to ensure licenses are used in the proper manner.
- **Improved self-management:** Reduced overhead is needed to interact with the Cisco Global Licensing Organization when working with common tasks related to licenses.
- **Increased control:** With Virtual Accounts, the customer can group licenses by technology, department, region, or other business requirement.
- **Improved compliance controls:** Virtual Accounts enable you to verify license usage and break down the enabled features along virtual domains of control.

By default, a Smart Account has one Virtual Account named default. Virtual Accounts are used to organize licenses and can be defined to fit a company's specific requirements. Virtual Accounts can be defined by region, line of business, or technology so that licenses can be grouped together to allow for easier management and utilization tracking.

User Access Control to a Smart Account and Virtual Account is controlled with four different user account types:

- **Smart Account Administrator:** Able to edit the Smart Account, manage users, and perform overall license management

https://t.me/learningnets

@cisco_in_persian_channel

- **Smart Account User:** Able to view licenses in the Virtual Account(s) but cannot create Virtual Accounts
- **Virtual Account Administrator:** Able to edit Virtual Accounts and perform license management
- **Virtual Account User:** Able to view and perform licensing activities for a specific Virtual Account

An example of these accounts in practical application is an individual or a group that is a Smart Account Administrator. This group can create Virtual Accounts, move licenses between Virtual Accounts, accept the EULA, and finally grant others rights within a Smart Account. Depending on the security policies and company size, it is possible that all users may be Smart Account Administrators. A more common scenario is that Virtual Account Administrators and users are also utilized. With these account types in use, a more granular UAC is available to the Smart Account Administrators. This allows them to provide users with only the access they require to perform their job functions.

What makes up a Smart Account, and how do devices communicate with Cisco? The key piece of software that devices communicate with for Smart Licensing is the Cisco Smart Software Manager (CSSM). The CSSM can be located either in the cloud (direct deployment) or an SSM located within the enterprise network using SSM On-Prem (mediated deployment). A final method, License Reservation (SLR), is also available; however, this method still requires interaction with CSSM upon initial deployment.

- **Direct deployment:** This is the simplest of the deployment models for Smart Licensing. This method requires the end device to have access to the Internet directly or through a proxy. This method provides full functionality for Smart Licensed devices because they connect back to Cisco.com for license management. The infrastructure supporting the CSSM is located within the US, and user access is controlled through the management of the Smart Account and Virtual Account administrators.
- **Mediated deployment:** For environments where the IT security policy doesn't allow the UC infrastructure to communicate directly with the cloud SSM, Cisco has developed an on-premises version that provides the same functionality as the cloud version (see Figure 8-11). With SSM On-Prem, the communication with the cloud is centralized to the SSM On-Prem, and the internal UC infrastructure can in kind be configured to communicate with the SSM On-Prem. This scenario provides several advantages to user provisioning and the mapping of the local Virtual Accounts to the Virtual Accounts in the cloud. By using the SSM On-Prem, the accounts (users and Virtual Accounts) are local to the SSM On-Prem; they do not have to reside in the Cloud SSM. These users can create additional Virtual Accounts within the SSM On-Prem that are not visible to the Cisco Cloud SSM. By using this method, you can effectively hide the network information from Cisco and send limited information relating to license usage requirements.

https://t.me/learningnets

@cisco_in_persian_channel

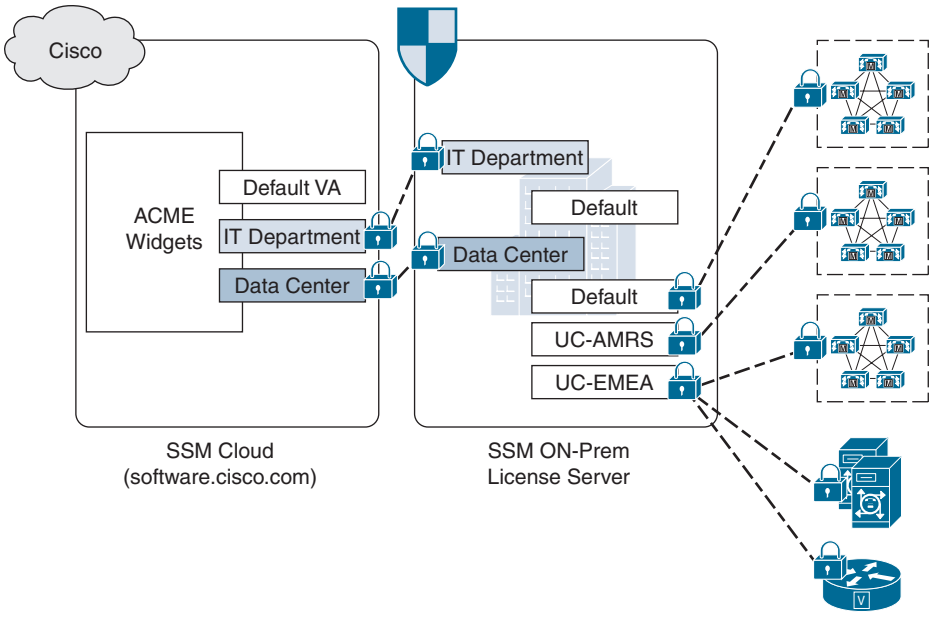


Figure 8-11 *SSM On-Prem Account to Cisco Virtual Account Relationship*

- **License Reservations:** License Reservation comes in two flavors: Specific License Reservation (SLR) and Permanent License Reservation (PLR). Cisco discourages you from using this type of licensing because it removes the software and license management capabilities that Smart Licensing makes available and node-locks the license to the device. License Reservation is expected to be used by governmental agencies and the military. Specifically, PLR is made available only to the military because it enables all functions available to the device permanently. While the other methods of licensing provide a self-service methodology, the License Reservation method removes that capability, so Cisco's Global Licensing Organization must be engaged.

Regardless of the method used to communicate with CSSM, the connection is secured using TLS to ensure end-to-end encryption of the communication channel. The choice on which SSM to use is completely up to customers and their security requirements.

How does the use of SSM On-Prem help secure the UC environment? One easy answer is that it helps address common concerns about what data Cisco has access to when Smart Licensing is used. Cisco can collect seven different data sets but requires only three pieces of information to license a device:

- **Trusted Unique Identifier:** This is the identifier used by both SSM and SSM On-Prem during the licensing process. It is a combination of the serial number and UUID.
- **Licenses Consumed:** These are the license types and counts for each license type required for operation.

https://t.me/learningnets

@cisco_in_persian_channel

- **Organization Identifier (Token):** This ID is used to associate the product to the specific account and Virtual Account.

The following data sets are included by default but can be disabled. Cisco does not track this information, and it is provided for customer reporting. The recommendation here is that when SSM On-Prem is used, this information is reported internally so that the data can be used in reports but disabled when connecting to CSSM.

- Host name (Optional)
- IP address (Optional)
- MAC address (Optional): used in reporting for the customer
- Other Smart Call home information (Optional)

Another benefit that SSM On-Prem provides is the ability to add local users to manage licenses that are separate from the users in CSSM. With SSM On-Prem, deployed users can be configured locally or imported from LDAP. Authentication for imported users can be handled through LDAP, OAUTH2 ADFS, and single sign-on (SSO).

While the CSSM never initiates communication with SSM On-Prem or a device directly registered to it, the communication must happen for the licenses to be synchronized and remain in compliance. With SSM On-Prem used as the registration point for the enterprise, it can either communicate automatically with CSSM or use a disconnected manual method. This manual method uses a sneaker net file transfer to synchronize with CSSM. This process works by generating a sync request from SSM On-Prem and then uploading the file to CSSM. CSSM then generates an account authorization file that is uploaded to the SSM On-Prem.

Ultimately, for those concerned with the information sent to Cisco to allow licensing to work, SSM On-Prem can perform basic network hiding from Cisco. This is accomplished with multiple local Virtual Accounts created on the SSM On-Prem; the local Default Virtual Account can be used to synchronize license usage between CSSM and the SSM On-Prem. When these aspects are taken into account, SSM On-Prem should be the primary tool used for monitoring and tracking license usage.

Summary

To recap, this chapter covered endpoint hardening, securing of conferencing resources, and Smart Licensing. Endpoint hardening is a layered approach where only the features and services necessary for an endpoint to function are enabled. The remaining unused features are either disabled or restricted. When you are hardening endpoints, start with Enterprise Phone Configuration settings to make configurations at a global level and then move to the Common Phone Profile and then the device to apply more specific settings at a site or device level. To secure conferences, you need to understand the different conference types. For example, you need to know when the built-in bridge within an IP phone is used to conference versus when a software or hardware conference resource is

https://t.me/learningnets

@cisco_in_persian_channel

used. Simply enabling SRTP and TLS between endpoints does not provide a secure conference if the conference resource is not secure. The connections between the endpoint and the conference might be secure, but without the proper configuration, the conference itself might not be secured. In this chapter, you learned how the use of Smart Licensing affects the security of Cisco Unified CM and by extension the other Smart Licensed UC products. For Smart Licensing, you learned what components make it up and, in addition, you learned who needs a Smart Account and what Virtual Accounts are and when they should be used. The chapter also discussed which method of synchronization with CSSM makes sense for the UC environment: whether a direct connection to CSSM, a mediated deployment using SSM On-Prem, or something more traditional like License Reservation. Regardless of the method used to synchronize license usage, the connections always use TLS to encrypt the connections. Finally, you learned what information Cisco requires for a successful synchronization and how you can limit this information only to the required information.

Additional Resources

Security Guide for Cisco Unified Communications Manager, Release 12.5(1)SU2

Phone Hardening - https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/security/12_5_1SU2/cucm_b_security-guide-1251SU2/cucm_b_security-guide-1251SU2_chapter_01111.html

Secure Conference Resources Setup - https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/security/12_5_1SU2/cucm_b_security-guide-1251SU2/cucm_b_security-guide-1251SU2_chapter_010000.html?bookSearch=true

Cisco Collaboration System 12.x Solution Reference Network Designs (SRND) – Cisco Rich Media Conferencing - https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/srnd/collab12/collab12/confernc.html?bookSearch=true

https://t.me/learningnets

@cisco_in_persian_channel

This page intentionally left blank

https://t.me/learningnets

@cisco_in_persian_channel

Chapter 9

Securing Cisco Unity Connection

In this chapter we discuss the process to secure Cisco Unity Connection and center it around four core areas:

- Securing user access
- Securing the phone system integration to Cisco Unified CM
- Securing messages
- Preventing toll fraud

In securing user access, we review how to secure user and administrative access to the system. This review is a rehash of some of topics previously discussed in Chapter 6, "Core Cisco UC Application Lockdown," and how they specifically relate to Cisco Unity Connection. We discuss the roles that administrators can assign to provide an understanding of their different use cases. This review enables administrators to properly provide access to the system based on end-user requirements. We also cover the types of user account controls available in Unity Connection in addition to how they differ from those in Cisco Unified CM.

We then look at the two main methods (RSA and next-generation encryption) of integrating Cisco Unity Connection securely with a phone system like Cisco Unified CM using SIP. The goal here is to explain the process used to secure the signaling and provide media encryption with a focus on using the standard X.509 digital certificates.

Next, we discuss how to best secure voicemail messages within Cisco Unity Connection. We address the impact of marking messages private and/or secure. Then we discuss the system settings for message aging and others that should be considered for message security to explain how they relate to voicemail security.

https://t.me/learningnets

@cisco_in_persian_channel

Last, we talk about toll fraud and what tools Cisco Unity Connection provides to limit the impact of potential toll fraud.

Returning to our case study company, ACME was in the process of reconciling its monthly billing records and noticed a large number of outbound calls during nonbusiness hours. When the senior IT manager, Jonah, brought in Anthony to help find out who was placing the calls, they discovered that several of the voicemail accounts had been compromised. Anthony started to dig and found that a few of the voicemail accounts for the company were not being used and that some of them were compromised. Additionally, he found several configuration settings that would allow users to dial in to their voicemail and then place outbound calls. The unrestricted ability to dial out from Unity Connection was what had been exploited and resulted in cases of toll fraud.

For this chapter we discuss the process that Anthony goes through to ensure that his Unity Connection server is secure and provide some insight into how the steps taken to secure other UC applications can be leveraged to help secure Cisco Unity Connection.

Questions that you should ask:

1. What can be done to secure voicemail access, and how is voicemail being used?
2. How does the voicemail system integrate with the call processing system?
3. When you are securing voicemail, what are some of the call types you should expect to see, and what should you allow or attempt to block?

Baseline Security Considerations Overview

If you haven't already read Chapter 5, "Hardening the Core Cisco UC Appliance Operating Systems," and Chapter 6, "Core Cisco UC Application Lockdown," we advise you to do that now. The first few sections of this chapter rely heavily on understanding tasks that were performed to build a baseline level of security outside of what is built in by default.

To begin, let's review a few of these building blocks. First, the version of Cisco Unity Connection that is deployed matters. There are two versions. The first is export restricted and can have SRTP and TLS enabled to secure the signaling and media. The second version is export unrestricted and does not have the ability to use SRTP and TLS to secure the signaling and media. Although on face value this distinction may not seem very important, you need to be aware of some things if you're working with the two versions. Table 9-1 outlines the functionality that is available to the security modules of the two versions.

https://t.me/learningnets

@cisco_in_persian_channel

Table 9-1 *Security Module Functionality*

Functionality	Restricted Version	Unrestricted Version
SSL for IMAP connections used to access voice messages	Allowed	Disallowed
Secure SCCP, SIP, and SRTP for call signaling and media	Allowed	Disallowed
Communications among networked Unity Connection servers or clusters (over secure MIME)	Allowed	Disallowed
SSL for Comet notification (Jetty SSL command)	Allowed	Disallowed

Additionally, the difference in version can adversely affect upgrade paths because although you can upgrade from a restricted version to an unrestricted version, the opposite is not the case. Upgrading from an unrestricted version to a restricted version is not supported. This concept also applies to disaster recovery backups: you cannot restore an unrestricted version of Unity Connection on a restricted install. Luckily, with Unity Connection, you do have the option of using Consolidated Object Backup and Restore Application Suite (COBRAS) Export and Import to perform this type of migration.

When you're considering upgrades to Unity Connection, if you are upgrading from a version prior to 12.0(1), the system will be in evaluation mode until it registers with either the Smart Software Manager or Smart Software Manager On-Prem that uses a token to allow export control functionality. This move to Smart Software Manager is mandatory and requires a valid Smart account and Virtual account, both of which are available free from Cisco.

Lastly, regarding an install running on a restricted version, even though encryption is available, it is not enabled by default. You must manually enable it by using the **utils cuc encryption <enable/disable>** command in current versions of Unity Connection. For tools like COBRAS Import and Export and a secure SIP phone system integration to function properly, encryption must be enabled. One caveat to this is that COBRAS can be used with an unrestricted version of Unity Connection when it is run with the **/UseCUMI** argument. Using this option forces COBRAS to use the REST-based CUMI API rather than the standard IMAP API. The reason for this change in API usage is that the IMAP API works only over TLS/SSL, which is not available in the unrestricted versions of Unity Connection. For more information on using COBRAS for migrations, see the COBRAS reference document titled "Connection Migration Strategies" at www.ciscounitytools.com/Documents/Connection/MigrationOptions/MigrationStrategies.htm.

https://t.me/learningnets

@cisco_in_persian_channel

Another baseline security feature that was covered in previous chapters is Federal Information Processing Standard (FIPS) 140-2, which is a US and Canadian government certification mandating the requirements that cryptographic modules must adhere to. Typically, most customers outside of governmental organizations like the US federal government and Department of Defense do not enable FIPS due to the strict requirements it places on the cryptographic algorithms that can be used. Like the unrestricted and restricted versions of the software, FIPS brings nuance to a UC environment.

Some of the aspects to consider when enabling or disabling FIPS is that it will cause the certificates being used to authenticate and secure the services provided by Unity Connection to be regenerated. If these certificates are self-signed, this might not be an issue. If the certificates are signed by a certificate authority, you will need to request new certificates. Regardless of whether the digital certificates are signed by a certificate authority or are self-signed, they potentially may need to be uploaded elsewhere in the network. For example, if FIPS results in new digital certificates in the environment, updates would likely be required to a Security Assertion Markup Language (SAML) agreement if SSO is enabled or remote nodes if IPsec is configured.

Since FIPS compliance is version dependent, it is recommended that you first disable FIPS when performing an upgrade to a version that is not FIPS compliant. This step in and of itself will require additional planning to ensure that the digital certificates that are regenerated do not cause an outage.

With FIPS enabled, certain encryption and hashing algorithms are disabled. For example, if you are configuring SNMPv3, you cannot use MD5 and DES. You need to use SHA1 and AES.

The nuances to how Unity Connection is configured when FIPS is enabled are not meant to steer you away from using it. The purpose is to encourage you, as a UC engineer, to read the relevant Unity Connection Design Guide and Unity Connection Security Guide for a more in-depth explanation of these behaviors.

The last security baseline aspect to discuss is monitoring and logging. These topics are covered in Chapters 5 and 6 to provide a consistent configuration process across the core UC applications. It is important to configure logging such as audit logs, which provide a method of tracking changes within Unity Connection. These logs provide relevant information as to what was changed, who changed it, and when it was changed. Configuring trace settings allows for the collection of application-specific logs to be available in the event of an outage or error. When logging is configured, does it help at all if logs are not collected? Not really. The collection of logs allows monitoring to take place. With syslog and SNMP, you can collect relevant data for Unity Connection to provide a baseline of the system. From this baseline you can identify any deviances that are identified and act accordingly, whether it is a service down event or failed login attempts. Lastly, you can use the Real-Time Monitoring Tool (RTMT) to collect trace logs and counter information for use in troubleshooting.

Unity Connection has built-in reports that can be generated and reviewed to ascertain system usage, failed logins, and unused voicemail accounts, among others. Table 9-2 describes some of the more useful built-in reports for Unity Connection.

https://t.me/learningnets

@cisco_in_persian_channel

Table 9-2 *Unity Connection Built-in Reports*

Report Name	Description of Output
Phone Interface Failed Logon	Includes the following information for every failed attempt to sign in to Unity Connection by phone: ■ Name of user, alias, caller ID, and extension of user who failed to sign in ■ Date and time the failed logon occurred ■ Whether the maximum number of failed sign-ins has been reached for the user
Users	Includes the following information for each user: ■ Last name, first name, and alias ■ Information that identifies the Unity Connection or Cisco Business Edition server associated with the user ■ Billing ID, class of service, and extension ■ Whether the account is locked ■ Whether the user has enabled personal call transfer rules
Port Activity	Includes the following information for voice messaging ports: ■ Name ■ Number of inbound calls handled ■ Number of outbound message waiting indicators (MWI) calls handled ■ Number of outbound Audio Message Interchange Specification (AMIS-a) calls handled ■ Number of outbound notification calls handled ■ Number of outbound Telephone Record and Playback (TRaP) calls handled ■ Total number of calls handled
User Phone Login and MWI	Includes the following information about phone logins, MWI activity, and message notifications to phone devices per user: ■ Name, extension, and class of service ■ Date and time for each activity ■ The source of each activity ■ Action completed (e.g., Login, MWI On or Off, and Phone Dialout) ■ Dialout number and results (applicable only for message notifications to phone devices) ■ The number of new messages for a user at time of login

https://t.me/learningnets

@cisco_in_persian_channel

Report Name	Description of Output
User Lockout	Includes user alias, the number of failed logon attempts for the user, credential type (a result of "4" indicates a logon attempt from the Unity Connection conversation; a result of "3" indicates a logon attempt from a web application), and the date and time that the account was locked. (Date and time are given in Greenwich Mean Time.)
Unused Voicemail Accounts	Includes user alias and display name, and the date and time that the user account was created. (Date and time are given in Greenwich Mean Time.)
Transfer Call Billing	Includes the following information for each call: ■ Name, extension, and billing ID of the user ■ Date and time that the call occurred ■ The phone number dialed ■ The result of the transfer (connected, ring-no-answer [RNA], busy, or unknown)
Outcall Billing Detail	Includes the following information, arranged by day and by the extension of the user who placed the call: ■ Name, extension, and billing ID ■ Date and time the call was placed ■ The phone number called ■ The result of the call (connected, ring-no-answer [RNA], busy, or unknown) The duration of the call in seconds
Outcall Billing Summary	Arranged by date and according to the name, extension, and billing ID of the user who placed the call and is a listing of the 24 hours of the day, with a dialout time in seconds specified for each hour span.
Call Handler Traffic	Includes the following information for each call handler, in rows for each hour of a day: ■ Total number of calls ■ Number of times each key on the phone keypad was pressed ■ Extension ■ Invalid extension ■ Number of times the after greeting action occurred ■ Number of times the caller hung up

https://t.me/learningnets

@cisco_in_persian_channel

A couple of these reports should be periodically reviewed for abnormal behavior. The first is the Phone Interface Failed Logon. This report is useful in identifying failed login attempts and can provide insight into specific accounts that could be compromised or which attempts to compromise could be in progress. A follow-on report is the User Lockout report, which provides a listing of the currently locked-out accounts. This report is useful in identifying accounts that are currently locked out and would be particularly useful if you have received several notifications that people are locked out of their voicemail accounts. This report gives you a more global view to identify the scope of users being locked out, which could be indicative of a larger effort to compromise the system. A couple of reports to also review periodically are the Port Activity and Billing reports. They are useful in building a baseline for normal interactions with voicemail and outcalls. Say normal business hours are between 9 a.m. and 5 p.m. with little voicemail activity after 6 p.m., but recently you have noticed a spike in calls to voicemail after midnight with a high number of transfers. You would need to look into this type of deviation from the base because it could indicate attempts to compromise the system and potentially commit toll fraud.

How does enabling encryption, logging, and reporting affect the security baseline? Enabling encryption provides the ability to authenticate and encrypt the signaling and media between Unity Connection and other UC applications and endpoints. The ability to enable encryption is predicated on having the proper version of Unity Connection deployed. Certain cryptographic algorithms are not available if FIPS is enabled and this can impact features and services that you plan to provide to end users. If there is a requirement that FIPS is enabled, being knowledgeable about what it can affect can potentially avert an outage. Lastly, with regards to logging and monitoring, simply enabling logging will not help you identify a deviance in the behavior of the system. You must review the logs and the outliers to see if they need additional attention. These security baselines focus on confidentiality (encryption), integrity (data validity), and availability (system uptime).

Securing User Access to the Unity Connection

In previous chapters, we discussed securing user access. As with the other VOS-based UC applications, it is necessary to secure both the operating system users and the application users. The operating system users are secured via configuration using the same commands as other VOS-based applications. Typical configuration parameters for the different accounts are

- Setting the minimum credential length
- Enabling password complexity
- Configuring the use of stored passwords to reduce repeated passwords
- Setting credential expiry to force passwords to be reused
- Disabling unused accounts

These password policies should be defined in consultation with the IT security group to ensure they align with existing policy and the lines of business to ensure they can be met.

https://t.me/learningnets

@cisco_in_persian_channel

Some additional practices with regards to securing user access are a "separation of church and state" when it comes to administrator accounts. This means that system administrators should use their administrator accounts only for day-to-day administration of Unity Connection and not for checking voicemail. The practice of least privilege should be applied to the administrator accounts that are created. It is recommended that users have only the access needed to perform their functions. Unity Connection provides user access control groups in the form of roles, which can be used to provide granular access to the system. However, depending on the size of the support staff, the use of roles could be impractical. A small support staff might see an environment where they are all full administrators, whereas a large deployment for a company with a multitiered support staff likely would see different roles for the support staff being applied. Some administrator accounts should be protected in a similar manner to a root account in Linux. These specific accounts are the OS administration account, Application administrator account, and Unified Messaging accounts. The latter two accounts are created during installation, and the former is used with Unified Messaging. The administrator accounts created during installation should be secured and have backup accounts created and assigned to individual system administrators. The main purpose for this is accountability for the actions of an account. The built-in accounts are generic, so identifying who performed what actions with an account can be a difficult task. Unified Messaging accounts should be used in a similar manner as the built-in administrator accounts. While the Unified Messaging account is not able to reboot a mail server, it can, however, read and send messages on the mail server.

The other type of account is an end user. End users are the user accounts used to access voice and video mail within Unity Connection. These accounts are created with a mailbox, so they have two types of passwords available to them. The first is a web application password (used to log in to Cisco Personal Communications Assistant [PCA], Unity Connection Conversation, etc.), and the second type is a PIN that is used to log in to the telephone user interface (TUI) when accessing voicemail from a phone.

Regardless of the type of user, they are imported using a user template. User templates allow you to configure preset attributes for the users when they are created. They also include passwords and password settings. Typically, a standard password is applied to these templates, and when a new account is created, the users log in and are prompted to change the password to something new. While that approach works well, it is not encouraged, mainly because the same or similar password/PIN combination is always given out. The recommended approach is to populate those fields with 12+ character strings and then uniquely assign users their passwords and PINs when the accounts are created. Bulk administration can be used when large numbers of users are created to import these unique password/PIN combinations.

On the topic of passwords, as with Cisco Unified CM, trivial passwords are not allowed by default in Unity Connection. The following trivial password rules determine whether a password/PIN should be considered trivial or not.

https://t.me/learningnets

@cisco_in_persian_channel

A nontrivial phone PIN has the following attributes:

- The PIN cannot match the numeric representation of the first or last name of the user.
- The PIN cannot contain the primary extension or alternate extensions of the user.
- The PIN cannot contain the reverse of the primary extension or alternate extensions of the user.
- The PIN cannot contain groups of repeated digits, such as "408408" or "123123."
- The PIN cannot contain only two different digits, such as "121212."
- A digit cannot be used more than two times consecutively (e.g., "28883").
- The PIN cannot be an ascending or descending group of digits (e.g., "012345" or "987654").
- The PIN cannot contain a group of numbers that are dialed in a diagonal, vertical, or horizontal straight line on the phone keypad (e.g., the user cannot use "159," "159730," "147," "147365," "123," or "123597" as a PIN).

A nontrivial web application password has the following attributes:

- The password must contain at least three of the following four characters: an uppercase character, a lowercase character, a number, or a symbol.
- The password cannot contain the user alias or its reverse.
- The password cannot contain the primary extension or any alternate extensions.
- A character cannot be used more than three times consecutively (e.g., "!Cooool").
- The characters cannot all be consecutive, in ascending or descending order (e.g., "abcdef" or "fedcba").

These and other password/PIN-related settings are defined under the authentication rules that are applied to individual user templates.

Along with authentication rules, the user templates can also have message aging policies applied to them. These policies are useful from a security perspective in that they can play a role in compliance policies. If the company has policies that dictate voicemails or emails should be automatically deleted after six months, the message aging policies should either account for an average number of messages within a six-month period, or a backup-and-restore methodology should be identified that allows for compliance with the policy.

To review message aging, we discuss two components. First, we quickly discuss mailbox quotas followed by aging policies and message expiration. To start with, mailbox quotas are used to limit the size of a user's voicemail box by limiting the space allocated to it. With this quota enabled and configured, users are forced to eventually delete old and unwanted voicemails because when they reach the thresholds, they cannot send or receive any more voicemails. The default mailbox quota at the system level is displayed in Figure 9-1.

https://t.me/learningnets

@cisco_in_persian_channel

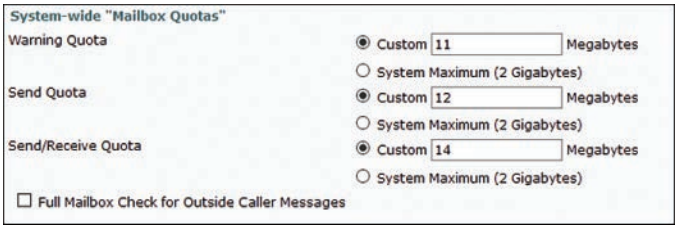


Figure 9-1 *Unity Connection System Mailbox Quota Settings*

You can verify these settings by logging in to the Cisco Unity Connection Administration GUI and navigating to **Message Storage > Mailbox Quotas**. In Figure 9-1 notice that the **Full Mailbox Check for Outside Caller Messages** checkbox is not selected. When this option is not selected, Unity Connection does not check incoming messages against the message quotas if it considers that an outside caller is leaving the message. This setup can lead to accounts exceeding their allocated quotas. It is recommended you select this option as a default for the system.

The quotas can be changed to other values or set to the system maximum of 2 gigabytes. Because these settings are at the global level, they affect all mailboxes configured in the system. Using user templates, you can overwrite these settings when users are configured to allow them to deviate from the system standard. You can edit the message quotas for user templates by going to **Templates > User Templates**, selecting the template to update, and then selecting **Edit Mailbox**. From here (see Figure 9-2), you can select the Message Aging Policy, the Mailbox Store Information, as well as the different Mailbox Quotas available.

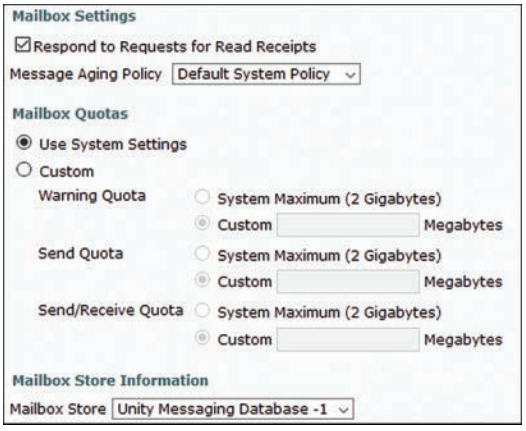


Figure 9-2 *Message Quotas User Template Settings*

You can complete the same process at the individual user level where the quotas can be updated. It is recommended that you use the system settings for the users and use the user templates or edit individual users only as one-offs.

One issue to put into perspective is the amount of time allocated for the default quotas (see Table 9-3).

https://t.me/learningnets
@cisco_in_persian_channel

Table 9-3 *Mailbox Size Quotas Defaults in Unity Connection*

Quota Level	Quota Size Limit	Action taken when quota reached	G.711 Mu-Law	G.711 A-Law	PCM 8 kHz	PCM 16 kHz	G.729a
Warning	12 MB	The user is warned that the mailbox is reaching the maximum size allowed.	18 min 11 KB/sec	18 min 11 KB/sec	9 min 22 KB/sec	4.5 min 44 KB/sec	122 min 1.67 KB/sec
Send	13 MB	The user is prevented from sending any more voice messages.	20 min 11 KB/sec	20 min 11 KB/sec	10 min 22 KB/sec	5 min 44 KB/sec	132 min 1.67 KB/sec
Send/ Receive	14 MB	The user is prevented from sending or receiving any more voice messages.	21 min 11 KB/sec	21 min 11 KB/sec	10 min 22 KB/sec	5 min 44 KB/sec	143 min 1.67 KB/sec

Setting message quotas is not enough though, and message aging policies should be configured. The default system policy for message aging in Unity Connection is to permanently delete messages in the deleted items folder only every 15 days. This policy would fail to meet the compliance requirements to delete messages older than six months, which means additional settings would need to be configured. Within the Cisco Unity Connection Administration GUI, navigate to **Message Storage > Message Aging > Aging Policy**, as shown in Figure 9-3. One example to accommodate this would be to create a policy to perform the following:

- Moving new messages to saved messages 30 days after receipt
- Moving saved messages to deleted messages every 60 days
- Permanently deleting messages in the deleted items folder every 90 days

https://t.me/learningnets

@cisco_in_persian_channel

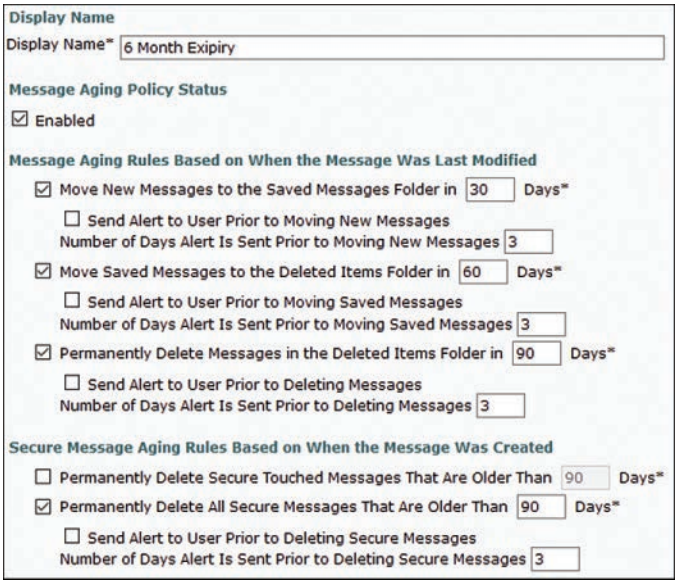


Figure 9-3 *Message Aging Policy*

A less granular approach of accomplishing the same task for compliance would be to enable message expiration, as shown in Figure 9-4. This approach enforces Message Expiration at a system level rather than a user level.

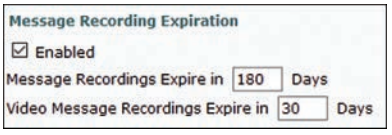


Figure 9-4 *Message Expiration*

Securely Integrating Unity Connection with Unified CM

The integrations between Unity Connection and Unified CM are focused on those using SCCP or SIP. If the traffic between Unity Connection and Unified CM is not secured, several points of exposure to threats exist. These threats are not unlike those faced by Cisco Unified CM cluster when it is not configured to secure signaling and media.

https://t.me/learningnets

@cisco_in_persian_channel

These threats include

- Man-in-the-middle attacks
- Network traffic sniffing
- Modification of the signaling between Unity Connection and Cisco Unified CM
- Modification of the media between Unity Connection and the endpoint (an IP phone or gateway)

The types of security afforded Unity Connection are the same as you would see when you secure a Cisco IP phone. That is, you can encrypt the signaling and media or authenticate the signaling and media between Unity Connection and either Cisco Unified CM or an endpoint. The same types of indicators are present when connections between Cisco IP phones are made—that is, a lock icon or a shield to indicate encrypted traffic or authenticated traffic, respectively.

The default integration type for Unity Connection is nonsecure. This type provides no integrity and privacy of call signaling for messages because the signaling messages are sent in the clear (unencrypted) to Cisco Unified CM. When Unity Connection is configured for Authenticated mode, the integrity of call-signaling messages is ensured to Cisco Unified CM through an authenticated TLS connection. However, call signaling and media are sent as clear (unencrypted) text. Configuring encryption provides integrity and privacy of call-signaling messages; the communication for the integration to Cisco Unified CM uses an authenticated TLS port, and the signaling messages are encrypted. This also allows the media to be encrypted. For the media to be encrypted, the devices in the media path between Unity Connection and the endpoint must support encryption.

Note The signaling and media encryption discussed here protect only the call to Unity Connection. The recorded messages are not stored in an encrypted format but can be protected within Unity Connection using private secure messaging.

Integrating with Cisco Unified CM Securely

Unity Connection supports integration with Cisco Unified CM using either SCCP or SIP, and both support nonsecure, authenticated, and encrypted communication. The recommended method of integration is to use SIP because it is an industry standard protocol and easier to configure and troubleshoot. Figure 9-5 illustrates the signaling flow between Unity Connection and Unified CM when setting up a secure connection.

https://t.me/learningnets

@cisco_in_persian_channel

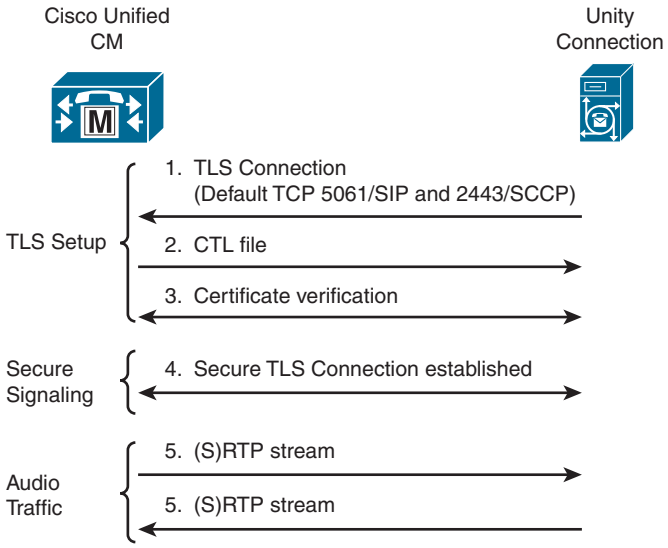


Figure 9-5 *Secure Phone System Integration Signaling*

1. Cisco Unified CM sets up a secure Transport Layer Security (TLS) connection to CUC server either on port 2443 Skinny Call Control Protocol (SCCP) or 5061 Session Initiation Protocol (SIP) based on the protocol used for integration.
2. Unity Connection server downloads the Certificate Trust List (CTL) file from the TFTP server (a one-time process), extracts the CallManager.pem certificate, and stores it.
3. Cisco Unified CM server offers the CallManager.pem certificate, which is verified against the CallManager.pem certificate obtained in the preceding step. In addition, the Unity Connection certificate is verified against the Unity Connection root certificate stored in Cisco Unified CM.
4. If verification of the certificates is successful, a secure mTLS connection is established. This connection is used to exchange encrypted SCCP or SIP signaling.
5. Audio traffic can be exchanged either as Real-time Transport Protocol (RTP) or SRTP.

Applying Certificates Against the SIP Trunk Integration

The process to enable encryption is broken down into several steps, starting in Unity Connection and creating the SIP certificates and ending with updating the SIP Trunk Security Profile on the trunks in Cisco Unified CM. The process to secure the integration between Unity Connection and Unified CM contains six tasks, which are outlined in Figure 9-6.

https://t.me/learningnets
@cisco_in_persian_channel

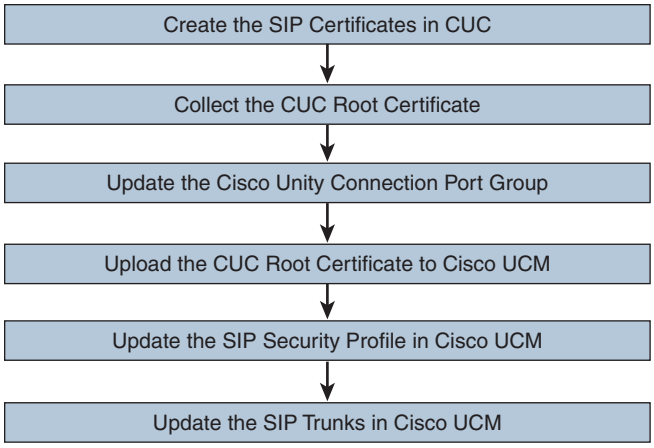


Figure 9-6 *Secure SIP Configuration Steps*

The first step in enabling encryption is to create the X.509 SIP certificate that will be applied to the port group used for the integration. This certificate will be verified in Cisco Unified CM by the CUC Root certificate. Log in to Unity Connection Administration and navigate to **Telephony Integrations > Security > SIP Certificate**. Select **Add New** and populate the Display Name and Subject Name (see Figure 9-7). The Subject Name used here will be specified in the SIP Trunk Security Profile in Cisco Unified CM.

Figure 9-7 *Unity Connection SIP Trunk Certificate*

Note If Unity Connection has had FIPS enabled or disabled, you need to regenerate the Root certificate before creating the SIP certificate.

https://t.me/learningnets

@cisco_in_persian_channel

Download the Unity Connection root certificate located at **Telephony Integrations > Security > Root Certificate** and save it using a .pem extension (see Figure 9-8). Shortly, you will upload this certificate to the Cisco Unified CM Callmanager-trust store on each of the nodes in the Unified CM cluster.

Figure 9-8 *Unity Connection Root Certificate*

Next, you need to update the port group used for the SIP integration to Cisco Unified CM to use the secure SIP port (5061/TLS), the newly created SIP certificate, Security Mode, and if media encryption will be enabled, then Secure SRTP. The Security mode can be set to Authenticated or Encrypted, as shown in Figure 9-9; the setting here needs to match the configuration within Cisco Unified CM.

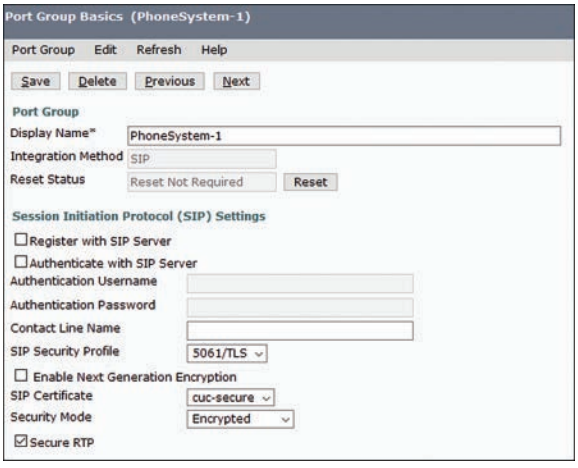


Figure 9-9 *Secure SIP Port Group*

https://t.me/learningnets

@cisco_in_persian_channel

The servers configured for the port group need to be validated. The TLS port should be set to use 5061 by default, and the Unified CM TFTP nodes should also be configured on the port group so that the CTL file can be downloaded. The port group or groups need to be reset after making these updates (see Figure 9-10).

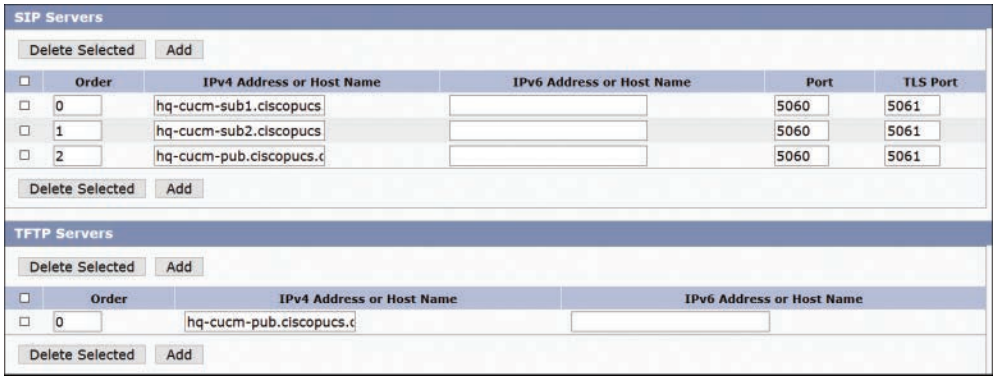


Figure 9-10 *Update Port Group Servers*

The updates to Unity Connection should be complete now. The configuration within Cisco Unified CM can now be updated to connect securely to Unity Connection.

The Unity Connection root certificate that was previously saved must be uploaded to the Callmanager-trust certificate store on each call processing node in the Unified CM cluster. Although it's not required on every node, it is best practice to upload the certificates across the cluster for consistency. The remaining tasks left to be completed in Cisco Unified CM are the creation of a Secure SIP Trunk Security Profile that can be applied to the Unity Connection trunks and the updating of the Unity Connection Trunks to enable the security.

Within Cisco Unified CM Administration, update the SIP Trunk Security Profile located at **System > Security > SIP Trunk Security Profile** (see Figure 9-11). If there is not already a Secure Unity Connection SIP Trunk Profile, you can copy and update the existing Unity Connection SIP Trunk Profile. The device security mode should be set to either **Authenticate** (for mTLS connection verification) or **Encrypted** (to enable signaling encryption and SRTP); this setting should match the configuration of the port group in Unity Connection. The Subject Name that was configured on the SIP certificate in Unity Connection must be configured under **Secure Certificate Subject or Subject Alternate Name**. You need to select two additional options so that message waiting indicators (MWIs) and transfers from Unity Connection are functional. The first is selecting the checkbox for Accept Unsolicited Notification. The second feature is selecting the option for Accept Replaces Header. This allows "REFER w/replaces" to be passed, allowing Unity Connection to initiate supervised transfers.

https://t.me/learningnets

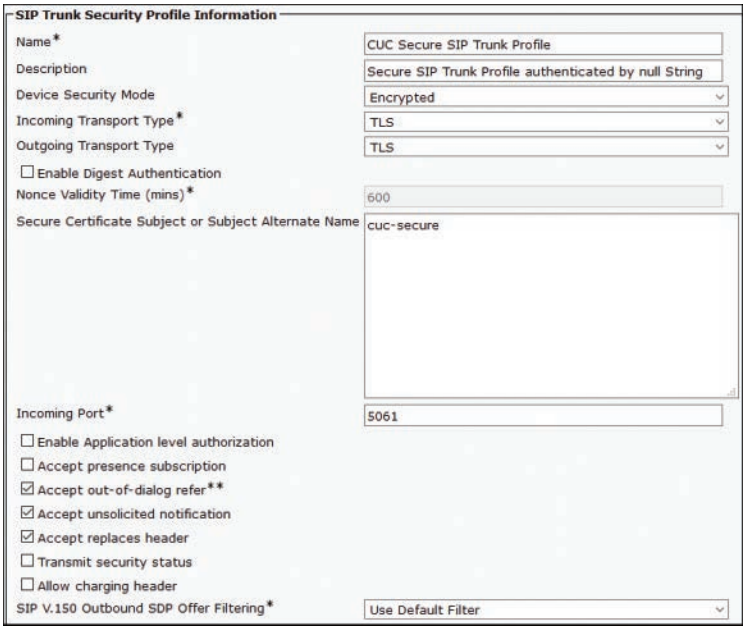


Figure 9-11 *Secure SIP Trunk Security Profile*

The last step of the process is to configure the existing SIP trunks to Unity Connection with the newly configured SIP Trunk Security Profile, update the SIP port, and enable SRTP. To do so, navigate to **Device > Trunk** and select the first Unity Connection SIP trunk. To enable the SIP trunk to support SRTP, you must select the option for SRTP (see Figure 9-12). If this option is not selected, only the signaling will be encrypted; the media will not be encrypted.

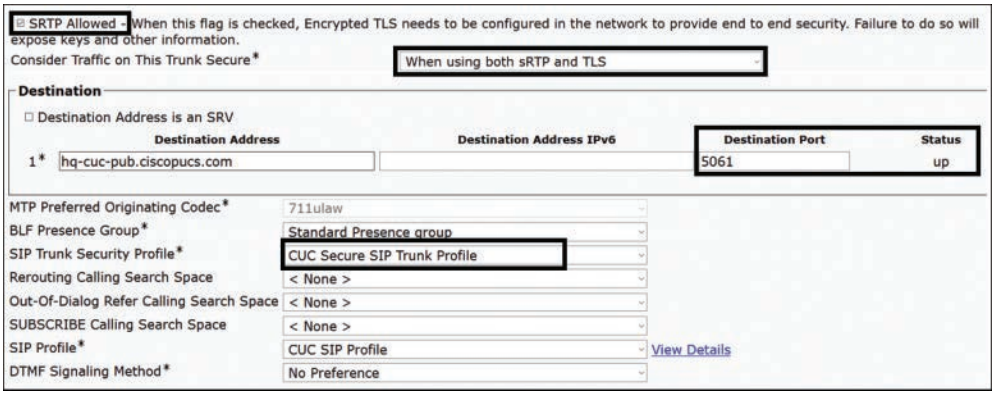


Figure 9-12 *SIP Trunk Updates*

https://t.me/learningnets
@cisco_in_persian_channel

The inbound CSS that is applied to this SIP trunk should be restricted to contain only the required route partitions necessary to complete expected call transfers from Unity Connection. For a practical application, you would typically want to ensure the inbound CSS is configured to reduce the risk of toll fraud and allow similar dialing habits reflected in the restriction tables configured in Unity Connection.

After saving these settings, reset the SIP trunk and repeat the updates on any remaining SIP trunks for the remaining Unity Connection pair if it is running in high availability.

Securing Messages

Securing users' voicemail messages takes more than simply enabling security on the phone system integration. The messages themselves are stored unencrypted within Unity Connection, and there are a few ways to restrict how those messages can be accessed. The first of these methods is to enable message sensitivity. This approach allows system administrators and users to control who can have access to a voice message, how long they have access to it, and whether it can be redistributed to others. In general, there are two types of message sensitivity within Unity Connection:

- **Private Messages:** Private messages cannot be forwarded and saved locally when an IMAP client is used to access the messages unless additional configuration is done. A message marked as private, if replied to or forwarded, has those replies and forwards marked as private. The end users cannot change the message sensitivity after it is set. Most important is that Unity Connection relays a message marked as private as a regular message with the private flag set if it has message actions configured to relay messages to an SMTP relay address. You can restrict this behavior to disallow this ability by navigating to **System Settings > Advanced > Messaging** and deselecting **Allow Relaying of Private Messages**.
- **Secure Messages:** Secure messages are a bit more nuanced. These messages can be forwarded by other Unity Connection users and are only stored on the Unity Connection server with the option to download the message disabled. As with private messages, if a message is marked secure and is replied to or is forwarded, the message sensitivity does not change. Additionally, by default, only Unity Connection subscribers homed in the local networking site can receive secure messages. Other contacts and remote networking site users can receive secure messages but only if the VPIM location or intersite link is configured to allow secure message delivery. However, once the message leaves Unity Connection, the security of the message cannot be ensured.

  When Unified Messaging is enabled, secured messages do not get forwarded to the relay. A nondelivery receipt is sent to the defined address for the mailbox, alerting the recipient that a secured message is present.

https://t.me/learningnets

@cisco_in_persian_channel

Message sensitivity can be enabled or disabled depending on where the message is left or with whom. Effectively all messages can be marked secure by enabling the following:

- Each class of service (CoS) can be configured to mark messages as secure. Users who sign in to their mailboxes to send messages have their CoS determine how messages are marked.
- Individual users and user templates can be configured to mark all messages left by outside callers as secure.
- Call handlers and call handler templates can be configured to mark all outside caller messages as secure.

When you are considering whether to configure message sensitivity, it is important to take into account the business requirements of the users. If the users deal with PII or other sensitive information and there is a remote chance that a message might be either left in a voicemail or overheard via background noise, it would be wise to enable message sensitivity to those types of calls, even if that type of information shouldn't be left in a voicemail.

The ability to shred messages is available for deployments that require additional security when deleting messages that have not been deleted by user or those messages that have not aged out of the system. Message shredding is available as a systemwide setting with Unity Connection using the standard Linux shred tool behind the scenes to perform the message shredding. The configuration ranges from 0 through 10, where the value entered is the number of times the message will be overwritten with random bits of data. Entering a number other than 0 for Message File Shredding Level located in Unity Connection Administration at **Advanced Settings > Messaging Configuration** enables Unity Connection message shredding (see Figure 9-13). The recommended maximum number of overwrites is three due to system performance, and by default, the shredding service runs every 30 minutes when the Clean Deleted Messages task runs.

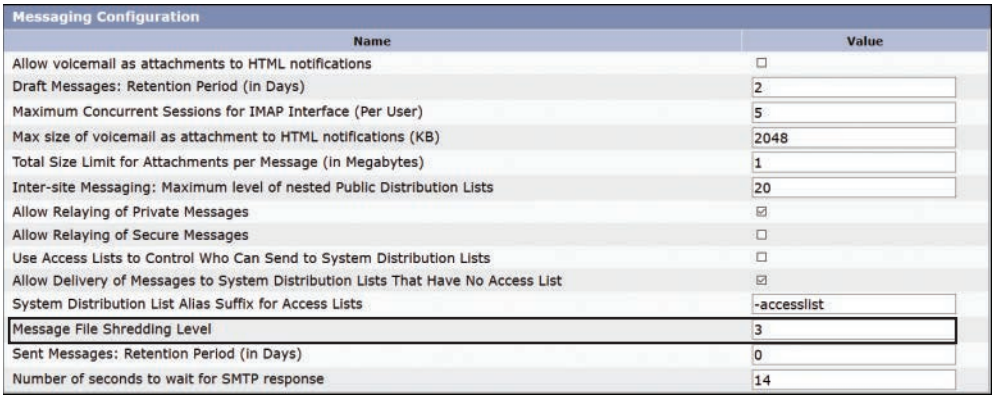


Figure 9-13 *Unity Connection File Shredding*

https://t.me/learningnets

@cisco_in_persian_channel

There are some caveats to using this service while the shredding occurs, mostly surrounding temporary files. For more information on these caveats, refer to the Cisco Unity Connection Security Guide.

Preventing Toll Fraud in Unity Connection

Toll fraud is committed when a compromised phone service places unauthorized long-distance or other types of toll calls. In extreme cases the cost of these calls can amount to thousands of dollars within a few hours, and usually the end user is responsible for repayment for the cost of the toll fraud. The perpetrators of toll fraud target networks with known vulnerabilities, and typically, provider networks are not targeted directly. Unfortunately, the provider is not liable for compensating customers for attacks outside their control. Simple configuration updates can often be completed to mitigate and reduce the likelihood of being a victim of toll fraud through Unity Connection.

Take a moment and think about how Unity Connection is deployed in your environment. Is remote access available to your voicemail system? If it is, have you considered how your subscriber accounts are configured? How about IVRs or ACDs? Does Unity Connection handle these call types? If Unity Connection does handle these types of calls, have you considered what access the system call handlers have?

These are the areas we review in this section. Specifically, we address how to harden your voicemail subscriber accounts and system call handlers to make them more resistant to toll fraud.

Do Not Skip PIN Logins

The first of the areas to discuss is user access to voicemail boxes. It is rarely a good idea to trust the caller ID of a call into a mailbox for authentication. While not enabled by default, this option is available for users with a mailbox and allows users to sign in to their mailbox solely using the caller ID of the call leg that is attempting to access the mailbox. With the rise of low-cost IP telephony, the ability to spoof the caller ID of someone is commonplace. The only time this type of feature is enabled is on very tightly controlled and monitored mailboxes, so in the event it is compromised, it can be detected and secured quickly.

Restriction Tables

Access to voicemail that allows transfers to outside extensions can be used to compromise and harass other systems or result in toll fraud, costing a company large amounts of money. Unity Connection uses restriction tables to control outgoing calls by allowing UC administrators to specify numbers and patterns that Unity Connection can dial for tasks such as

- Transferring calls
- Sending faxes to a fax machine

https://t.me/learningnets

@cisco_in_persian_channel

- Sending message notifications
- Creating user-defined alternate extensions

We focus on two areas of interest for restriction tables: transferring calls or sending message notifications and creating user-defined alternate extensions. What are the differences between each of these? The first, transferring calls, can be abused in several ways, most notably toll fraud where a voicemail box is set up to transfer to a high toll number. In another type of abuse, the system is used to transfer to another system in an attempt to compromise it or even simply harass the other party. The second use of restriction tables to look at is creating user-defined alternate extensions. With user defined-alternate extensions, the system administrator can determine whether a user can define an alternate number, and if allowed, you can restrict the types of numbers that can be defined.

Restriction tables are applied to a class of service, one each for call transfers, message notifications, and fax deliveries for each member assigned to the class of service. These restriction tables get applied when a subscriber attempts to change an extension used to transfer calls or update the extension for message notification; in addition, when signed-in subscribers attempt to specify a number, they define a system transfer. However, these restriction tables are not applied when an administrator makes the updates via the Cisco Unity Connection Administration GUI.

Restriction tables are similar in that they are a set of rules, consisting of rows of dial strings. These dial strings, just like the rules in an ACL, control whether a number, pattern, or URI is allowed when Unity Connection attempts to complete a transfer or delivery. In the case of caller system transfers, which allow unauthenticated callers to transfer to a number that they specify, Unity Connection checks the specified number against the Default System Transfer table. By default, this table blocks all numbers to protect against toll fraud and unauthorized use. A good rule of thumb when working with restriction tables is to limit dial strings you allow while denying everything else and also to be specific in the transfers. Unity Connection comes with predefined restriction tables that you can edit (including changing their names) but not delete, as shown in Table 9-4. By default, each restriction table prevents access to long-distance phone numbers and is based on using a site access code of 9.

Table 9-4 *Default Restriction Tables*

Table Name	Description
Default Fax	Restricts numbers for fax delivery.
Default Outdial	Restricts numbers for message notifications. Also restricts the user extensions that Unity Connection dials when the phone is selected as the recording and playback device in the Media Player.

https://t.me/learningnets

@cisco_in_persian_channel

Table Name	Description
Default System Transfer	Restricts numbers that can be used for caller system transfers, which allow unidentified callers to transfer to a number that they specify. For example, callers may want to dial a lobby or conference room phone that is not associated with a Unity Connection user. By default, the table does not allow Unity Connection to dial any numbers.
Default Transfer	Restricts numbers for call transfers.
User-Defined and Automatically Added Alternate Extensions	Restricts the numbers the users can use to create alternate extensions for themselves through interfaces such as the Cisco Personal Communications Assistant or via an API call. It also restricts numbers from being offered as alternate extensions. For example, you might block a lobby or conference room extension so that users who frequently call Unity Connection from those shared phones are not automatically prompted to add the number as an alternate extension.
Excluded Extensions for Automatically Added Alternate Extensions	Restricts numbers from being offered as alternate extensions. For example, you might add a lobby or conference room extension so that users who frequently call Unity Connection from those shared phones are not automatically prompted to add the number as an alternate extension.

You can edit the predefined restriction tables, and you can create up to 100 new ones. You can also add up to 50 dial strings to a table. When new dial strings are inserted into the restriction table, they are inserted at dial string 0. If you compare restriction tables to ACLs, the order of the dial strings is very important because Unity Connection sequentially compares a phone number to the call patterns in the restriction table, starting with dial string 0. If a number matches more than one call pattern, the number is handled according to the first call pattern it matches.

The dial strings used in the restriction tables allow the special characters listed in Table 9-5 as wildcards.

Table 9-5 *Unity Connection Wildcard Characters*

Wildcards	Actions
*	Matches zero or more alphanumeric strings.
?	Matches exactly one digit or character. Uses ? as a placeholder for a single digit.
#	Corresponds to the # key on the phone.

Let's use the Default Transfer restriction table 9-4 as a reference. In this case, all calls that attempt to transfer to an extension that begins with "+" or "9" are blocked by default, as shown in Figure 9-14.

https://t.me/learningnets

@cisco_in_persian_channel

The last dial string in a restriction table is the * wildcard, which cannot be deleted or edited. This dial string is important because it allows the administrator to use restriction tables as either a white list or black list. With a white list, only approved patterns can be dialed and everything else is blocked. The default configuration is that of a black list, however, because explicit patterns are blocked and everything else allowed. Ultimately, the correct configuration is the one that fits your needs in the most secure manner.

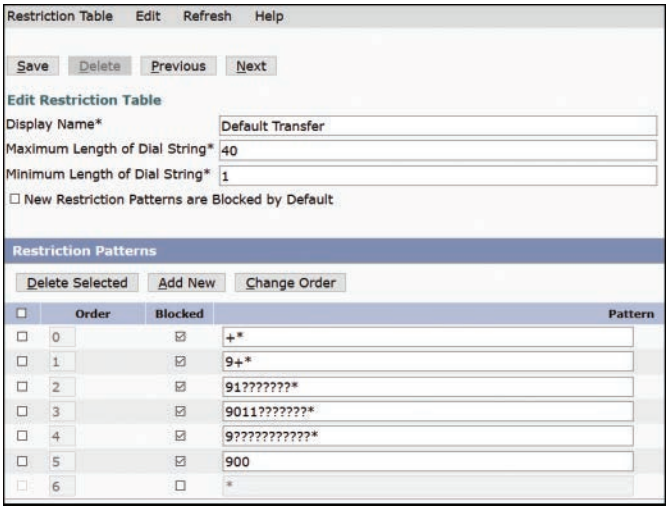


Figure 9-14 *Default Transfer Restriction Table*

If the site uses a different access code than 9 or no access code at all, those transfers would succeed. Successful transfers could be a problem if not accounted for. Therefore, it's important to look at the restrictions applied to the subscribers and the system itself to ensure they securely meet the needs of the deployment.

The restriction table in Figure 9-14 could be updated to allow transfers to internal extensions and local external numbers if the phone system integration supports the transfer with the following update (illustrated in Figure 9-15).

- Internal extension range 201555[1-9]XXX
- Internal dialing via 5-digit extension
- Local calling uses access code 9 and 7- or 10-digit dialing.
- Long-distance calls require access code 9 followed by the US country code 1 + 10 digits.

https://t.me/learningnets
@cisco_in_persian_channel

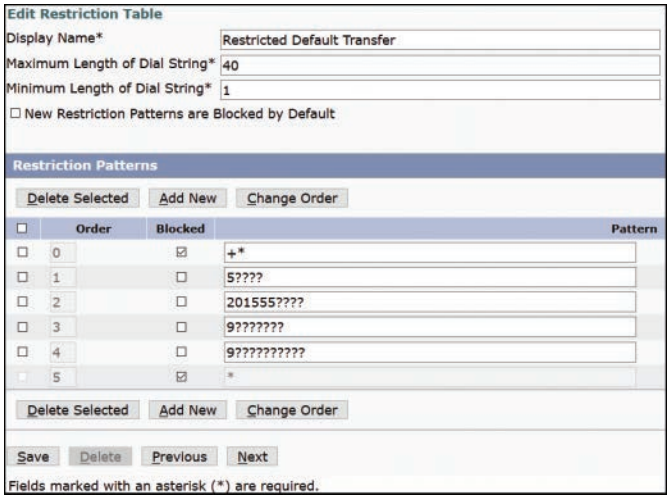


Figure 9-15 *Updated Default Transfer Restriction Table*

Another feature available to subscribers with ties to restriction tables is the ability to add new alternate extensions to their mailboxes. Unity Connection keeps a record of the caller IDs of the extensions that sign in to a subscriber's voicemail box. Understanding this capability is important because if the feature is enabled, after five logins from the same extension, subscribers will be prompted if they would like to add it as an alternate extension. What does this feature do, and how is it affected by the restriction tables? When enabled, the feature uses the recognized alternate extension to ease the login process and bypasses the need to enter a subscriber ID when logging in to the Unity Connection TUI. An issue arises if the alternate extension is added outside the control of an administrator. For example, if lobby phones have access to the voicemail pilot or voicemail in general, they could have their extension logged as an alternate extension for a legitimate subscriber. This is where some planning of the Excluded Extensions for Automatically Added Alternate Extensions restriction table can be used to block out specific ranges of numbers that belong to common area phones, like those in a lobby or kitchen area.

The parameter to set the **Sign-in Count for a Number Before It Is Offered as an Alternate Extension** is configured in **Cisco Unity Connection Administration > System Settings > Advanced > Conversations** (see Figure 9-16).

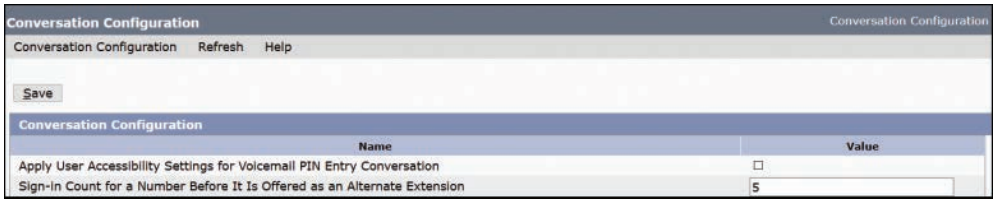


Figure 9-16 *Sign-in Count for Alternate Extension*

https://t.me/learningnets
@cisco_in_persian_channel

The feature is enabled at the class of service that is applied to the end user located in **Cisco Unity Connection Administration > Class of Service**. Select the class of service to enable the feature and update the setting under Alternate Extensions (see Figure 9-17). By default, this feature is not enabled for users with a mailbox.

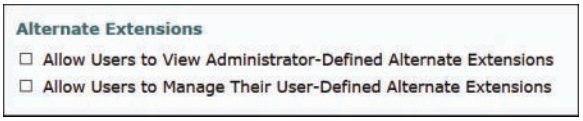


Figure 9-17 *Class of Service, Alternate Extensions*

Hardening Access to the TUI/GUI

Within this section, we discuss updates to common points of access into Unity Connection to illustrate some common items to review when looking to secure ways of access to the system.

TUI Voicemail Restricting Alternate Contact Numbers

With a focus on toll fraud, you can change the keymapping for end users and restrict the ability to set alternate contact numbers. This, in conjunction with the remote side of the telephony integration, can reduce the chances of toll fraud by removing the ability to add an alternate contact number via TUI. Why would this be done? One common method of toll fraud using voicemail is to find unused, weakly secured voicemail boxes and then use the alternate contact number to transfer them to a toll number where another call can be made. The charge for this call is then passed on to the owner of the compromised system. Removing this ability from the TUI negates it but still allows end users to set the alternate contact number using Cisco Unity Connection Personal Communication Assistant.

There are two steps to this: first, you update one of the Custom Keymapping settings to remove the ability to access the settings via the TUI to change the alternate contact number. Then you apply this updated custom keymapping to the user templates.

In Cisco Unity Connection Administration GUI, navigate to **Tools > Custom Keypad Mapping**. Because the default keypad mapping assigned to voicemail subscribers cannot be edited, you need to select one of the "custom" keypad mappings and update it to remove the option to update the alternate contact number listed under Settings. For this, you select the **Custom Keypad Mapping 1**, select the **Settings Menu** tab, and then remove the key assignment for **Alternate Contact Numbers**, as shown in Figure 9-18.

If, however, you need to leave the ability to update the alternate contact number available, you can enable the option to remove it from the audible menu by selecting the **Option Voiced in Menu** checkbox.

https://t.me/learningnets

@cisco_in_persian_channel

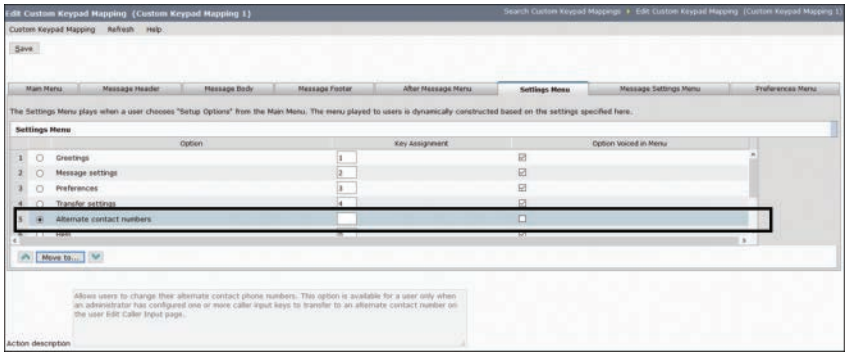


Figure 9-18 *Custom Keypad Mapping Update*

Next, the custom keymap needs to be applied to a user template (and ultimately the subscribers). Updating the user template ensures that new users created will have the newly updated custom keymap configured. As an example, this scenario updates the **voicemailusertemplate**. Navigate to **Templates > User Template** and select **voicemailusertemplate**; then select **Edit > Phone Menu** (see Figure 9-19). Here you can make two updates. The first is to change the **Conversation Style** from the default **Classic Conversation** to the newly updated custom keymap mapping. Additionally, you should update the setting for **When Exiting the Conversation** from using the opening greeting to using the **Goodbye** call handler. The reason for this change is simple: you want to control who has access to the opening greeting. From the opening greeting, callers can, depending on the configuration, access the directory of all voicemail subscribers; they also can attempt to sign in to other mailboxes. By setting this option to use the **Goodbye** call handler, you force Unity Connection to end the call after the interaction with the user voicemail has completed.

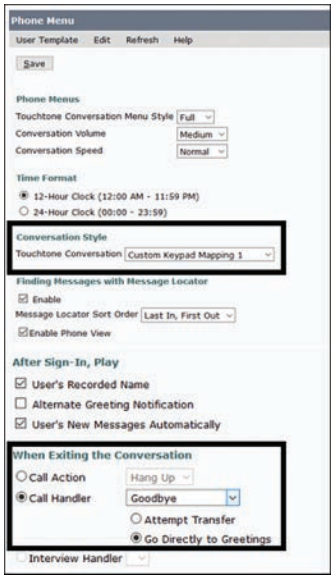


Figure 9-19 *User Template Updates*

https://t.me/learningnets
@cisco_in_persian_channel

Next, you can update the individual voicemail subscribers using Bulk Edit. There are two options for Bulk Edit: the first is a manual method of selecting users, and the second is to create and use a CSV file to select the users. The first option is better when you have only a few accounts to update or accounts that can be searched for and returned based on a simple search. The second is better for updating a larger number of accounts that cannot be filtered easily using the search criteria provided by Unity Connection. The second method is accomplished using a simple CSV file consisting of a header row showing "Alias," with a subsequent row for each user's Unity Connection Alias to be updated.

Note Be aware of two things when using Bulk Edit (see Figure 9-20). First, when specifying the alias of the voicemail subscribers to update, be aware they are case sensitive. The second is that the characters specified when creating the CSV file must be in either UTF-8 or UTF-16 encoding.

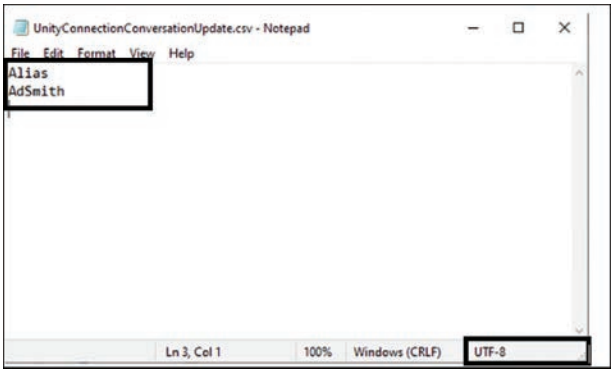


Figure 9-20 *Bulk Edit by CSV: Sample File*

From the upper-right corner of the main user search screen, select **Bulk Edit by CSV**, and when prompted, select the CSV file created, followed by **Bulk Edit.** If the file is created successfully, the screen should change to **Edit User Basics** and reflect a proper number of objects to update, as shown in Figure 9-21.

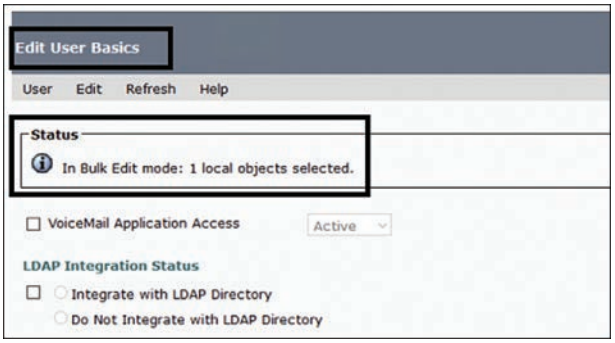


Figure 9-21 *Bulk Edit by CSV: Edit User Basics*

https://t.me/learningnets
@cisco_in_persian_channel

From here, select **Edit > Phone Menu**. The options to update are the same as when the user template was updated. The difference is that the checkbox next to each option that needs to be updated is selected as well (see Figure 9-22). This process is like using the Bulk Administration Tool in Cisco Unified CM. With the options selected, select **Run Now** and then **Submit**.

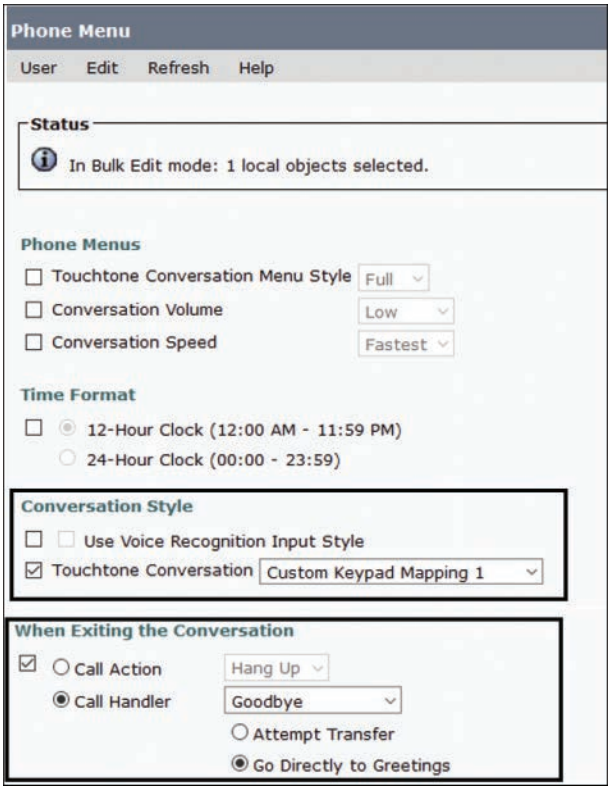


Figure 9-22 *Bulk Edit by CSV: Updating Users*

System Transfers from Call Handlers and Nonsystem Numbers

The last few settings to review involve controlling user input and restricting access to system settings via the Unity Connection TUI. While these settings do not specifically tie in to preventing and reducing the risk of toll fraud, they are avenues to gain more access to the system rather than just leaving a voicemail or accessing an IVR hosted by Unity Connection.

The Opening Greeting within Unity Connection is the gateway into Unity Connection. From here, you can

- Leave a voicemail if you know the party's extension
- Dial another party and have the system call the user

https://t.me/learningnets
@cisco_in_persian_channel

- Sign in to your own voicemail box
- Access the system distribution list for all voicemail users in the system

One problem is that, by default, all users are included within a systemwide distribution list. If an end user or attacker knows the extension of this distribution list, that person can send messages to *all* of the voicemail subscribers on the system. This message can be helpful, such as "The system will be upgraded this coming weekend; please expect intermittent services." Or it could be a more malicious, phishing-type of message asking users to visit a specific site to update their credentials.

You can configure the user templates and individual users to be excluded from this distribution list, and if this feature is not used, it is recommended that you use one of two options. The first option is to configure the user templates so that the option **List in Directory** is not enabled and then to use **Bulk Edit** to update the existing users to have the feature disabled. The second option is to restrict this option to only administrator accounts with a voicemail box (these should be limited). This setting allows the feature to be available to administrative users so that systemwide messages can be sent while restricting who can send the messages. The process to restrict access is detailed in the following paragraphs.

The first step is to create a new partition, which is then configured on the system distribution list. In Unity Connection Administration, navigate to **Dial Plan > Partition** and select **Add New**. From here, create a new partition named Restricted_PT. This new partition is added to a new search space called AdminVM_SS. Create this new search space by navigating to **Dial Plan > Search Spaces** and selecting **Add New**. In this example, shown in Figure 9-23, the Restricted_PT is added to the AdminVM_SS. The default partition for the cluster (<Hostname> Partition) is not added to this search space because none of the administrator voicemail accounts should have the ability to send voicemail directly to voicemail subscribers. Remember that there is a difference between an account used to administer the system and a user account intended to send and receive voicemails.

Now, update the partition that is configured on the system distribution list allvoicemailusers with the new Restricted_PT partition, as illustrated in Figure 9-24. An additional systemwide distribution list, allvoicemailenabledcontacts, should be considered as well. If the Unity Connection deployment has contacts imported or configured for VPIM as an example, it would be prudent to update this distribution list as well to disable it or restrict access to it.

https://t.me/learningnets

@cisco_in_persian_channel

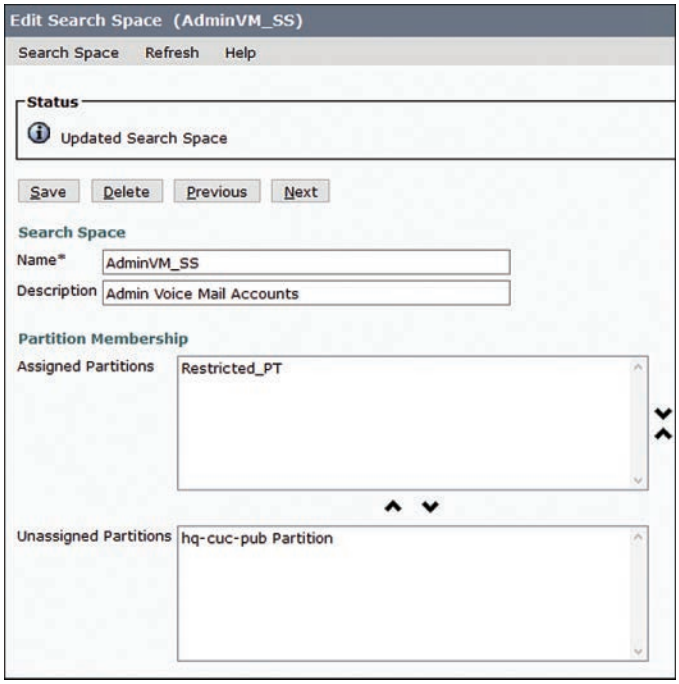


Figure 9-23 *AdminVM_SS Creation*

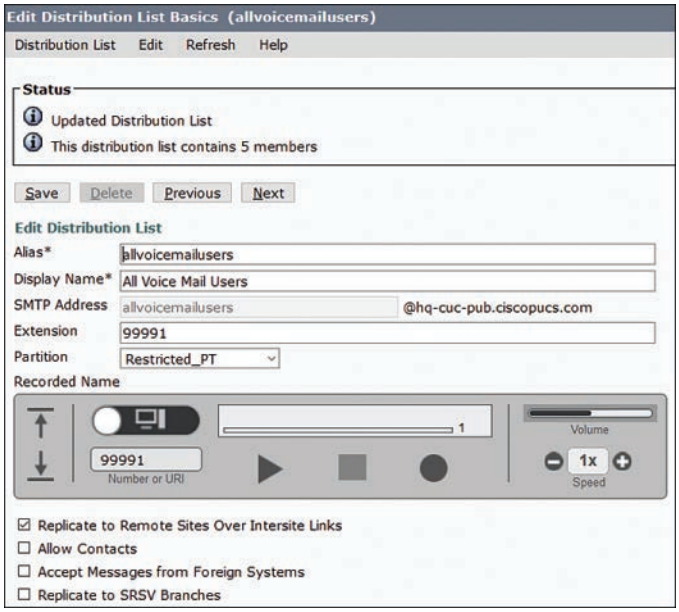


Figure 9-24 *System Distribution List Update*

https://t.me/learningnets

@cisco_in_persian_channel

Next, with the opening greeting, a systemwide directory handler is configured; it should have its access restricted as well. This directory handler is configured as option 4 on the opening greeting, which means anyone who has access to the opening greeting can search for users listed in the directory and then either leave a message or attempt to be transferred to those users. This useful feature comes in handy; however, the voicemail policy should recognize that not all users may need to be in this list. Users can either opt out of being listed in the directory by themselves, or an administrator can act to either remove individual users or update the user templates for new users to not include them. The recommended action is to remove it as an option from the Opening Greeting call handler if the feature is not used, or if it is used, limit the scope of the directory handler used to a subset of users who can redirect calls to the proper parties.

The last item to discuss is caller input within call handlers and voicemail subscribers. While the greetings are playing, callers can perform different actions like attempting to sign in to another voicemail box or being transferred to the opening greeting. There is no hard-and-fast rule for what to update here, but it is recommended to remove unused options, set the unused input keys to ignore, and then lock them so that they do not accept any additional input.

Summary

In this chapter on securing Unity Connection, we started with ensuring the right version of Unity Connection is deployed so that it can support the features you need. There are two different versions available for Unity Connection: restricted and unrestricted. If your deployment requires TLS and SRTP to be enabled, you should deploy the restricted version. We also discussed the restrictions placed on the different versions. Of note was whether you could upgrade from an unrestricted version to a restricted version. You cannot directly upgrade between these versions, but it is possible to upgrade from a restricted version to an unrestricted version. We also briefly discussed user access control, and it was recommended that there be a clear delineation between administrator accounts and end-user accounts. The two common methods of integrating Unity Connection with Cisco Unified CM (SIP, SCCP) were discussed, along with how they are secured when integrated. We walked through the specific process to secure an SIP integration reflecting secured signaling with TLS and SRTP-enabled media between the endpoint and Unity Connection. Following that, we discussed toll fraud and covered ways to mitigate the exposure to it. The methods discussed for toll fraud included securing individual voicemail subscriber accounts, like not skipping the PIN for authentication for known calling numbers. In addition, we discussed restriction tables and explained how to use them to restrict what calls could be configured for transfers and notification. Lastly, we covered updates to the various interfaces available via the TUI. System distribution lists and corporate voicemail directories accessible from outside callers were outlined, along with why you would want to restrict this access. The purpose was to illustrate how these system resources can be used to gain further access and potentially cause more toll fraud.

https://t.me/learningnets

@cisco_in_persian_channel

Additional Resources

Security Guide for Cisco Unity Connection Release 12.x - www.cisco.com/c/en/us/td/docs/voice_ip_comm/connection/12x/security/b_12xcucsecx.html

Command Line Interface Reference Guide for Cisco Unified Communications Solutions, Release 12.5(1)SU2 - www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/cli_ref/12_5_1SU2/cucm_b_command-line-interface-reference-guide-1251Su2.html

Unified Messaging Guide for Cisco Unity Connection Release 12.x - www.cisco.com/c/en/us/td/docs/voice_ip_comm/connection/12x/unified_messaging/b_12xcucumgx/b_12xcucumgx_chapter_00.html

Unity Connection System Administration Guide (Custom Keypad Mapping Tool) - www.cisco.com/c/en/us/td/docs/voice_ip_comm/connection/12x/administration/guide/b_12xcucsag/b_12xcucsag_chapter_010010.html#ID-2418-000000cd

https://t.me/learningnets

@cisco_in_persian_channel

This page intentionally left blank

https://t.me/learningnets

@cisco_in_persian_channel

Chapter 10

Securing Cisco Meeting Server

In this chapter, we discuss the various aspects of securing Cisco Meeting Server (CMS), which is unique in many ways because the operating system, system configurations, and security features are all set up differently than many of Cisco's other Unified Communication applications that use Cisco's voice operating system. One big reason for the differences is that Cisco Meeting Server is an application that was acquired from a company formerly known as Acano.

To address the various strategies for securing Cisco Meeting Server, we cover configuration options and architectural considerations. Initially, we discuss security that is built into the operating system and which options exist for strengthening the security of the CMS platform. At this point, we can discuss the techniques that are needed to secure core services with X.509 certificates and to initialize the services. Because use of X.509 certificates is key for providing security connectivity between individual components and external infrastructure, our goal is to help you feel comfortable using certificates within CMS. Finally, we cover the security parameters that can be leveraged to establish secure connectivity to external infrastructure such as directory servers and Cisco Unified Communications Manager so that users can access meeting resources in a secure manner.

ACME's Conferencing Integration

As part of ACME's initiative to increase collaboration and innovation, ACME project management teams have created a series of recurring meetings for project team members. As part of this initiative, ACME would like team members across the organization to be able to join meetings from any device, at any time, and from any place. Meetings must be secured to help protect the information that will be discussed in a number of upcoming meetings. The worst fear that ACME leadership has is that the company may lose a competitive advantage regarding one of its new products it is planning to release called the widget 360. Therefore, ACME leadership has asked Anthony Starke, who is ACME's newly promoted UC manager, to take any and all precautions to make sure that meeting information is kept confidential so that it is not inadvertently leaked to the wrong personnel.

https://t.me/learningnets

@cisco_in_persian_channel

There is a short timeline for several of ACME's projects, which has project teams distributed globally across several time zones. Each member of the project team has different work habits, so Anthony is encouraging his team to support the main use cases, such as voice- and video-based calling into CMS meetings and the use of browser-based connections so that guests and end users without a client can easily join CMS meetings. To do this, ACME's video team needs to integrate CMS servers with ACME's collaboration environment, which has not been done previously. This is also the best way of aligning to the requirements of the leadership team, which involves allowing any user to call in to CMS from any device and from any location.

CMS Overview and Deployment Modes

CMS is a lightweight UC application that can be deployed in small UC environments while also providing the ability to scale up to and meet the high availability requirements of a large enterprise or hosted UC conferencing environment. When deploying CMS to support rich-media conferencing, organizations have the option of offering meeting services to users and devices that are located internally or externally to organizational boundaries. Devices that can be used to join CMS meetings include UC devices such as Cisco Jabber, IP phones, and video endpoints that are registered to call control platforms such as Unified CM and/or Expressway-C. Clientless web-based connections can join CMS meetings as well.

CMS uses a modern operating system that utilizes a series of Docker containers to provide functionality. The following are CMS components:

- **Database:** This component stores information about CMS meetings; it is managed by the Call bridge.
- **Call bridge:** This core calling component is used for bridging audio and video streams so that meeting participants can collaborate with each other. It is used for connectivity to external call control platforms such as Unified CM.
- **Web bridge:** This component is used for providing clientless connections to CMS via WebRTC and/or the CMS web app for accessing a CMS meeting. The Web bridge requires connectivity to a Call bridge.
- **XMPP server (optional):** This component provides connectivity to a CMS meeting for users who have the Cisco Meeting application in versions CMS 2.9 and earlier. It also handles the XMPP signaling between the Call bridge and XMPP clients. XMPP clients include the Cisco Meeting application and the WebRTC client.
- **Recording server (optional):** This component is used to record meetings.
- **Streaming server (optional):** This component is used to stream meetings.
- **Uploader (optional):** This component is used with vBrick to easily identify and download recordings.

https://t.me/learningnets

@cisco_in_persian_channel

We should point out that Cisco has removed CMS's reliance on XMPP services as of 3.0 software, so we discuss the use of XMPP services within CMS only minimally. Starting in CMS 2.9, Cisco introduced the Call bridge to Web bridge protocol (C2W), which provides connectivity between the Call bridge and the new Web Bridge 3 component. There is still a clientless option, but Cisco refers to this as the web app. For organizations that still want to use XMPP clients, Cisco recommends use of Cisco Jabber.

Figure 10-1 depicts how CMS components communicate with each other at a high level to exchange information between internal CMS components and externally between UC applications. For more information regarding a deployment that is more specific to your organization, Cisco offers deployment guides for single-server deployments and scalable and resilient deployments. These guides also provide specific information about which network ports are utilized between CMS components. We discuss securing the primary CMS components later in this chapter.

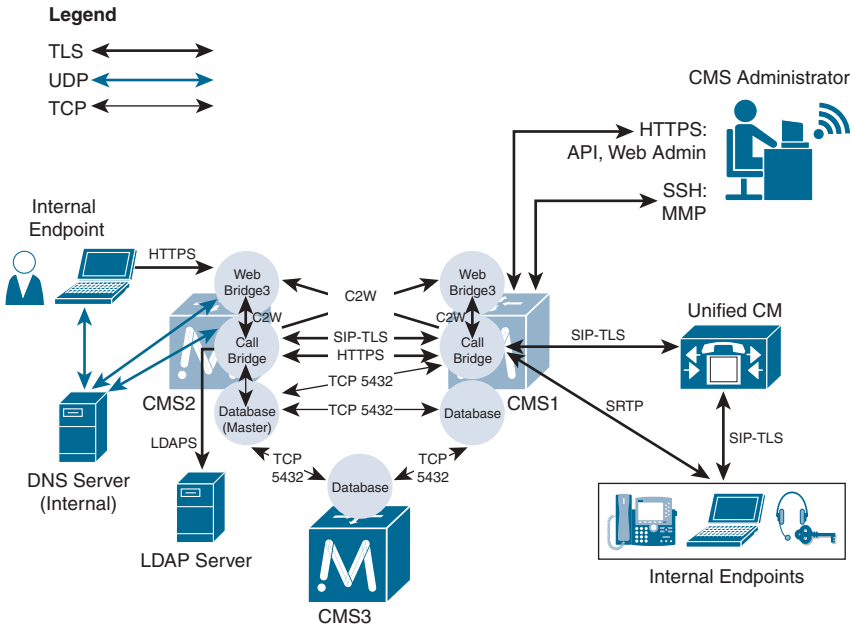


Figure 10-1 *Connectivity Between Internal and External Components*

When providing meeting services to users that are external to an organization's network boundary, Cisco's Collaboration Edge architecture is used with Expressway-C and -E to help provide firewall traversal of clientless web connections. CMS servers can also be configured to support Edge functionality, but for the sake of this discussion, we focus on using Expressway-C and -E because Cisco's preferred architecture recommends use of Expressway. When users attempt to access CMS meetings across public networks, such as the Internet, Expressway-E provides a web proxy to secure web traffic. Because Cisco recommends use of firewalls to provide security from external users, this architecture

https://t.me/learningnets
@cisco_in_persian_channel

provides two different layers of security (firewalls and web proxy) to secure CMS meeting resources from unauthorized participants or external attackers.

To support rich media for external CMS users, Expressway uses Traversal Using Relays around NAT (TURN). The Expressway TURN servers act as an anchor point for secure media that is trusted by the organization's firewall. Figure 10-2 depicts the security and the network connectivity that are needed to provide secure web-based connections outside of an organization's enterprise boundary to CMS meetings. We discuss Cisco's Collaboration Edge architecture further in Chapter 11, "Securing the Edge."

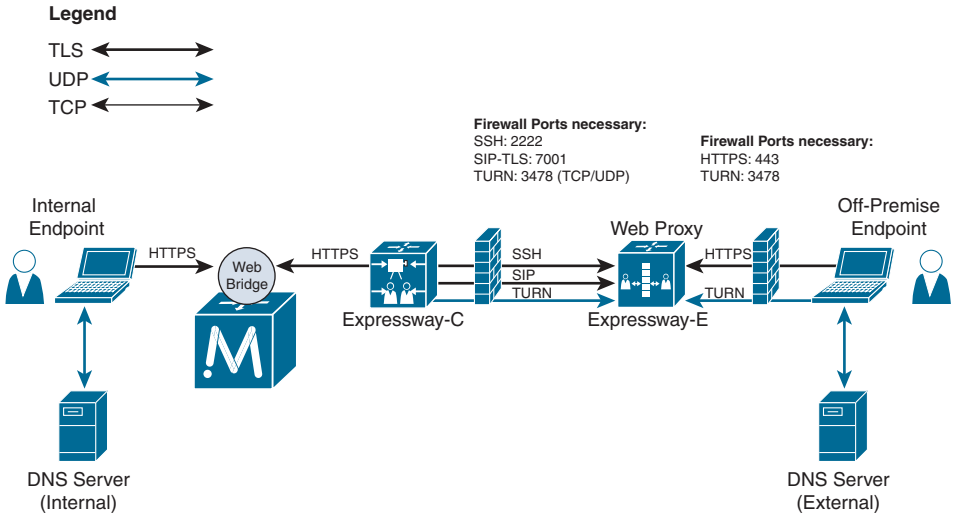


Figure 10-2 *Connectivity Between Internal and External Web-Based Connections to CMS*

Operating System Hardening

In Chapter 5, "Hardening the Core Cisco UC Appliance Operating Systems," we established how an operating system supporting UC applications is different from a standard Linux server. Although the operating system supporting Cisco Meeting Server is different from what supports the core UC applications (e.g., Unified CM, Unity Connection, IM/P), this principle stays the same. CMS uses an operating system that is based on the Linux operating system, which has been customized to support rich media-based conferencing. The CMS platform was also built with security in mind. Because Cisco designed a platform with security requirements in mind up front, CMS is able to comply with the most stringent of government and industry standards for rich-media conferencing. As an example, CMS currently is certified with the Federal Information Processing Standard (FIPS), which is a set of security standards that describes encryption algorithms, the secure design and implementation of cryptographic modules, message authentication,

https://t.me/learningnets
@cisco_in_persian_channel

mitigation of attacks, and so on. CMS is also certified with the Department of Defense (DoD) Joint Interoperability Test Command (JITC) requirements to ensure that interoperability is possible while deploying CMS in a secure fashion.

From a configuration perspective, three options are available for configuring a CMS environment:

- **Command-line interface (CLI):** This is often referred to as the Mainboard Management Processor (MMP) from the legacy Acano documentation.
- **Web GUI:** This interface is also known as the web admin interface that provides a configuration menu for specific features or access to API objects.
- **Application programming interface (API):** This interface enables you to use the CMS web admin or REST API tools and commands.

During the initial configuration of CMS and while hardening the OS, an administrator must use the CLI. The CLI can be accessed via the virtual machine console interface or via SSH after the Ethernet interface has been assigned with an IP address. The built-in administrator account (admin) is used to log in to the server for initial configuration.

Similar to a virtual operating system (VOS), some of the key takeaways that make the CMS operating system different from a standard Linux server include the following:

- Reduced number of default installed packages and applications
- No root access
- No standard shell access
- No third-party applications or packages installed independently without Cisco's approval
- IP Tables enabled by default for firewall usage

As we compare and contrast the CMS OS to VOS, there are some security enhancements. As discussed, CMS uses a modern OS that utilizes a series of Docker containers. Use of containers allows CMS to provide isolation of CMS components. The CMS OS also provides role-based access control (RBAC) for administrative and security purposes within its OS. This RBAC feature allows organizations to have granular security controls down to the OS layer and up to the application layer, as you will later see when we discuss use of the API. CMS supports the following administrative roles and associated permissions within CMS:

- **admin:** This role permits all tasks with the exception of auditing.
- **crypto:** This role permits all cryptography-related tasks.
- **audit:** This role permits an administrator to configure auditing; it sends logs to external syslog servers.

https://t.me/learningnets

@cisco_in_persian_channel

- **appadmin:** This role permits all application-level configuration through the web GUI's administrator interface.
- **api:** This role permits access to the API for obtaining information and making configuration changes.

Note As a built-in security mechanism, the CLI is preconfigured to time out after three minutes of inactivity. Currently, this parameter cannot be adjusted.

Like many other operating systems, the CMS OS enables administrators to align to an organization's security policy for password complexity. Password complexity is beneficial because it prevents all types of attackers, ranging from script kiddies to sophisticated attackers, from password guessing and/or launching brute-force attacks to gain access to the OS. Brute-force attacks, which use computers to guess username and password combinations, are simple to launch, and they're much more effective when an OS is configured with weak passwords. By default, CMS requires that all passwords be reset after six months. The following are some of the rules that can be implemented:

- **Max history:** Prevents password reuse
- **Password age:** Enforces the maximum age of a password (e.g., days)
- **Minimum length:** Specifies the minimum length of a password
- **Minimum amounts:** Specifies minimum amounts of characters (e.g., special, uppercase, and lowercase characters)
- **Prevention of usernames:** Prevents a password based on a username
- **Maximum failed login attempts:** Sets the number of failed logins allowed before a 15-minute lockout timer

For additional security, a text-based dictionary file can be uploaded into CMS to prevent dictionary attacks. With this mechanism in place, any attempts to create a new password that matches an entry in the dictionary file are rejected. The following password policy configuration can be used to secure access to the OS:

```
cms1a> user rule max_history 5
cms1a> user rule password_age 90
cms1a> user rule min_length 8
cms1a> user rule min_special 1
cms1a> user rule min_uppercase 1
cms1a> user rule min_lowercase 1
cms1a> user rule no_username true
cms1a> user rule max_failed_logins 5
```

https://t.me/learningnets

@cisco_in_persian_channel

As an additional layer of security for user accounts, CMS administrators can restrict access to the system based on expected work hours. The following configuration permits access to the CMS CLI only during weekdays and standard work hours such as 0900 to 1700 in 24-hour clock format and verifies the permissions of the user:

```
cms1a> user duty astarke Wk0900-1700
cms1a> user info astarke
astarke role: appadmin
Last login 2020-Aug-23 14:45:50 using web remote host 172.16.179.1
Failed authentications since last login: 0
Password last changed 2020-Aug-23
Password expires 2021-Feb-23
Duty hours: Wk0900-1700
cms1a>
```

For government and Department of Defense (DoD) organizations, CMS supports the use of the Common Access Card (CAC), which is used as a two-factor authentication mechanism for securely accessing information systems. CAC authentication to the SSH and web admin interface is possible only if LDAP integration has been performed beforehand, and if LDAP mappings include the **userPrincipalName** attribute to match the user identity located on the CAC. CAC authentication must be configured by using the CMS API because the **authenticationIdMapping** attribute is available only on the API.

When CAC is enabled, certificate-based login can be used in addition to local administrator accounts that use username and password combinations. If required, strict mode can be enabled so that only certificate-based logins are allowed into the CLI or web admin interface. Although using two-factor authentication is not necessarily unique to CAC cards, the CMS solution to use CACs was designed for the US government. Other certificate-based smart card systems are not supported for two-factor authentication into CMS because they have not been tested by Cisco. The following configuration of CAC-based authentication can be used to strengthen the authentication mechanism into the OS:

```
cms1a> cac enable
cms1a> cac issuer certbundle.crt
cms1a> cac
Enabled                 : true
Issuer certificate      : certbundle.crt
Strict (no passwords)   : false
OCSP Enabled            : false
OCSP Signing key        : none
OCSP Signing certificate: none
OCSP responder          : none
cms1a>
```

When using CAC-based logins, a certificate-enabled SSH client, such as puttyCAC, must be used in order to support smart card authentication with certificates. PuttyCAC is maintained independently from the US government by the open-source community. Online Certificate Status Protocol (OCSP) can optionally be enabled to check the validity

https://t.me/learningnets

@cisco_in_persian_channel

of a CAC-based certificate. When this setting is enabled, OCSP ensures that a certificate has not been revoked before allowing an administrator to sign in to CMS.

Similar to VOS, CMS provides a software-based firewall solution based on IP Tables. The internal CMS firewall is not intended to be a substitute for standalone firewall appliances, which is why we previously recommended use of firewalls to provide security for external connections. When IP Tables is configured, it provides organizations with the possibility to restrict access from certain networks (e.g., a management network) and services that exist with CMS to maintain and operate CMS securely and efficiently. Unlike VOS, which does not allow administrators to change or update the firewall settings without assistance from the Technical Assistance Center, these tasks are possible from CMS.

To add firewall rules, you just need to specify the interface, whether traffic should be allowed or denied followed by the port number, and a source IP subnet. When you're creating firewall rules, it is generally recommended that you use a console connection so that an inadvertent error or misconfiguration does not prevent you from being locked out from accessing the CLI when connected via an SSH session. The following configuration shows how to add a rule that permits SSH and SFTP from a specific IP subnet (10.10.1.0/24) and specifically tells CMS to deny all other traffic by default within the CMS firewall:

```
cms1a> firewall a allow 22 from 10.10.1.0/24
cms1a> firewall a default deny
cms1a> firewall a enable
```

Within CMS, each firewall rule is defined by a unique tag number. To view the status of the firewall rules that you have created along with the CMS firewall rules that are created by default when services are enabled, you, as administrator, can leverage the **firewall <interface>** command, as shown in Example 10-1. One thing of importance is that you need to understand that the CMS OS requires additional services than what we initially have listed here in order to properly function. If the CMS firewall is enabled, and rules have not been created to permit these services, an organization may cause a self-inflicted DoS attack against itself. We discuss these services and how to enable them in upcoming sections to help prevent this issue.

Example 10-1 *Firewall Rules That Can Be Used to Secure the CMS Operating System*

```
cms1a> firewall a
Interface      : a
Enabled        : true
Default policy : deny

Tag     Rule     Service
---     ----     -------
68      allow 443 webbridge3
69      allow 443 webbridge3
```

https://t.me/learningnets

@cisco_in_persian_channel

```
70      allow 80 webbridge3
71      allow 80 webbridge3
72      allow 9999 webbridge3
73      allow 9999 webbridge3
74      allow 8443 webadmin
75      allow 8443 webadmin
76      allow 5060 callbridge
77      allow 5060 callbridge
78      allow 5061 callbridge
79      allow 5061 callbridge
80      allow 22 from 10.10.1.0/24
cms1a>
```

We should point out that for management access to infrastructure, CMS does not permit use of plaintext configuration protocols such as Telnet and FTP, so firewall rules do not apply to these protocols. Instead, SSH and SFTP are used to provide secure management access. As previously discussed, CMS is FIPS certified. This means that within CMS, administrators also have the option of enabling strong cryptography to prevent against cryptographic attacks. The implication for enabling FIPS mode is that weak authentication and encryption mechanisms are not allowed because they do not comply with FIPS-approved algorithms. We discuss this topic further in the Management and Visibility section later on. To enable FIPS, you use the following commands:

```
cms1a> fips enable
System reboot required to effect changes

cms1a> fips test
Test passed
```

For additional safeguards and syntax options that can be used to secure the CMS OS, we recommend that you review the CMS MMP Command-Line Reference. This documentation from Cisco helps you better understand what other controls can be used to secure the operating system, password rules, firewall rules, and so on.

Now that we've discussed some of the options that can be used to secure CMS's operating system, we must discuss which dependencies CMS has on the network's infrastructure to gain full functionality and also how network infrastructure can be used to secure connectivity to CMS.

Infrastructure Considerations

Due to the functionality that CMS provides (e.g., rich-media-based conferencing involving voice, video, content sharing) inside meeting "spaces," CMS is high-value target for attackers. Meetings involve collaboration and information sharing, some of which may be confidential in nature. If attackers are able to gain access to (or eavesdrop) a CMS

https://t.me/learningnets

@cisco_in_persian_channel

meeting, they can collect a wealth of information about an organization and perhaps the business decisions that the organization is making. To align to defense-in-depth principles, we need to look beyond the OS's internal security controls and into the network infrastructure to provide the best security for a CMS deployment.

To prevent against attacks related to eavesdropping, the infrastructure is critical. At the transport layer, CMS leverages encrypted TLS connections to provide security between internal and external components. Because we have discussed use of X.509 certificates throughout this book, it should be no surprise that certificates are required to support TLS connectivity. As discussed in previous chapters, use of X.509 certificates requires integration with existing infrastructure such as NTP and DNS services. The use of DNS is generally a best practice because many of the other services that organizations deploy do not function easily or reliably without DNS. In the case of X.509 certificates that are used within CMS, DNS is required for resolving the fully qualified domain names (e.g., "A" records) that are located within the certificates that are used. CMS also uses DNS to find infrastructure services, which are known SRV records to establish connectivity to Microsoft Lync/Skype services.

Similar to DNS, using NTP is also a best practice for troubleshooting so that timestamps can be used to identify issues and to prevent session replay attacks. In the case of certificates, NTP services are required so that clients can verify that certificates are valid and can be trusted. To add an NTP server(s) to CMS, you use the **ntp server** command from the CLI. NTP services use port 123 (TCP or UDP). The following **ntp status** command can be used to validate the status of NTP in your environment:

```
cms1a> ntp server add [ip address of NTP server]
cms1a> ntp server add [ip address of secondary NTP server]
cms1a>  ntp status
remote        refid        st t when poll reach delay  offset  jitter
==============================================================================
 10.0.10.101 .INIT.        16 u  -   64   0     0.000  0.000   0.000
*10.0.10.100 10.81.254.131 2  u  48  64   177   0.378  0.094   0.191
 LOCAL(0)    .LOCL.        10 l  452 64   200   0.000  0.000   0.000
cms1a>
```

To add primary and redundant DNS servers, you must use the **dns add** command from the CLI. DNS services use port 53 (TCP or UDP). The following **dns lookup** command can be leveraged to validate functionality:

```
cms1a> dns add forwardzone . [ip address of DNS server]
cms1a> dns add forwardzone . [ip address of secondary DNS server}
cms1a>
cms1a> dns lookup A cms1a.ciscopucs.com
Trying cms1a.ciscopucs.com
10.140.2.51 (IS NOT DNSSEC SECURE)
```

https://t.me/learningnets

@cisco_in_persian_channel

During the DNS lookup validation process, you may have noticed the response from the DNS server: **IS NOT DNSSEC SECURE**. DNS Security Extensions (DNSSEC) is a suite of extensions that is added to the DNS protocol for added security, such as authentication of DNS records. Because the traditional DNS protocol offers very few security safeguards, attackers have managed to forge DNS records to trick clients to connect to a fraudulent and/or malicious host that is impersonating the actual host. This type of attack is known as DNS cache poisoning. Once connected to a malicious host, attackers can collect personal information such as login credentials, credit card numbers, and a number of other items by using programs such as key loggers. For this reason, the risk of operating traditional DNS-based services is a growing concern. Figure 10-3 depicts how a DNS poisoning attack occurs.

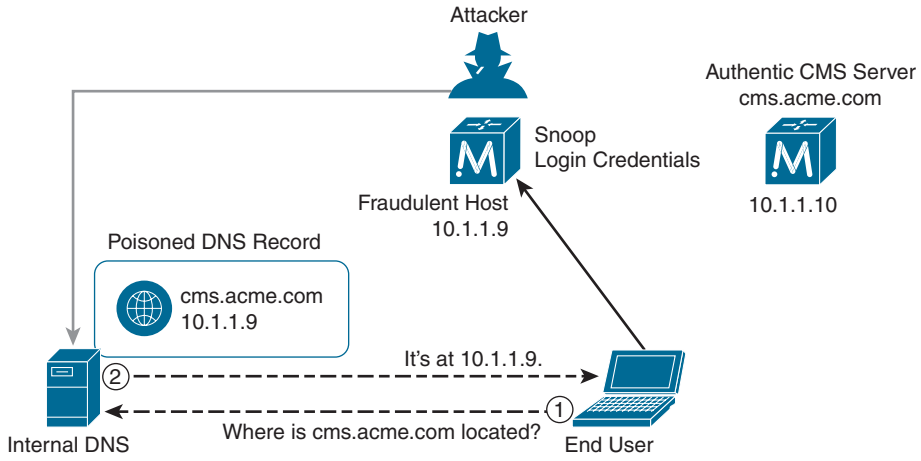


Figure 10-3 *How a DNS Poisoning Attack Can Impact a CMS Environment*

Unlike protocols that leverage TLS for security, DNSSEC responses are not encrypted over the network. Instead, DNSSEC responses are authenticated via digital signatures. Because CMS has a high reliance on DNS for users and devices to connect to its services, it can benefit from organizations that have DNSSEC deployed. DNSSEC ensures that users are not tricked into connecting to malicious hosts that could be impersonating a service, such as a CMS Web bridge for attackers to steal personal information like login credentials for a meeting. By implementing DNSSEC within CMS, an organization can be sure that components such as the CMS Web bridge are always directed to an authentic CMS resource, such as the CMS Call bridge from the "real" DNS server.

Before we talk about how to implement DNSSEC within CMS, let's discuss how DNSSEC provides organizations with increased security when compared to DNS. At a high level, DNSSEC provides a set of new DNS record types that is used to ensure integrity and trust by leveraging asymmetric cryptography to sign DNS records. By signing all DNS

https://t.me/learningnets
@cisco_in_persian_channel

records within a DNS zone with security keys to create cryptographic signatures, a client is able to verify that DNSSEC records are authentic. The following are key DNSSEC terms:

- **RRSet:** Resource Record Set. A grouping of DNS resource records inside a DNS zone.
- **Zone Signing Key (ZSK):** Private ZSKs are used to digitally sign RRSets.
- **RRSig:** Resource Record Signature. A digital signature of RRSets; a result after being signed by the Zone Signing Key.
- **DNSKEY:** A public key that resolvers use to verify RRSigs.
- **Key Signing Key (KSK):** Key used to sign the public ZSK. A root KSK acts as a trust anchor to validate DNSKEY records.
- **Delegation Signer (DS):** A record containing a hash/digest of a child domain's/zone's public KSK. It allows "delegation" of trust from a parent zone to a child DNS zone that is enabled for DNSSEC.

To establish a chain of trust and authenticate DNS queries, a PC (DNSSEC resolver) compares the validity of a child zone's public KSK to the DS record. If they match, trust is established because the resolver can assume that there has not been any tampering with records.

To implement DNSSEC within CMS, a Delegation Signer (DS) trust anchor is used. As specified in RFC 7958, "in order to obtain secure answers from the root zone of the DNS using DNSSEC, a client must configure a suitable trust anchor." As defined in RFC 4033, a trust anchor is "a configured DNSKEY RR or DS RR hash of a DNSKEY RR." In terms of the CMS OS, a trust anchor is simply a trustpoint by which CMS can authenticate DNS responses from existing DNS infrastructure. The syntax for creating a trust anchor is **dns add trustanchor <anchor>**. Within the "anchor" syntax, you must include the DNS zone name, the "DS" anchor type, the DNSSEC zone's key tag, the algorithm (8 = RSA/SHA-256), digest (2=SHA-256), followed by the hash of the public key used to sign the DNS zone. This information should be obtained from the DNSSEC administrator. Figure 10-4 depicts how DNSSEC can be used to authenticate DNS records in an organization. The following is a sample configuration for a DNSSEC trust anchor followed by a DNS lookup to verify DNSSEC functionality:

```
cms1a> dns add trustanchor "sec.ciscopucs.com. IN DS 11537 8 2
D470E2924EEAF3EACE2524E64824EA7D238DEDCAA8A82BC408F044BA7F27DA72"
cms1a>
cms1a> dns lookup a cms1a.sec.ciscopucs.com
Trying cms1a.sec.ciscopucs.com
10.140.2.51 (IS DNSSEC SECURE)
```

https://t.me/learningnets

@cisco_in_persian_channel

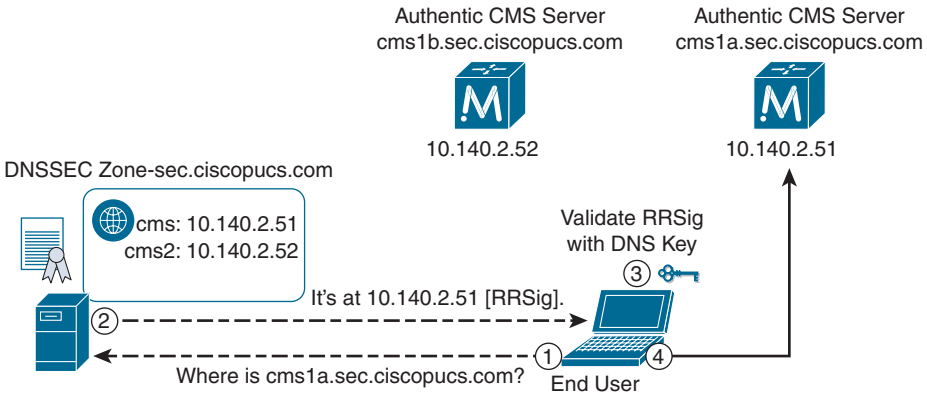


Figure 10-4 *Using DNSSEC to Authenticate Digitally Signed Resource Records*

Although DNSSEC is an extremely powerful tool for modernizing an organization's DNS environment to increase the security, the biggest challenge is organizational adoption. Unfortunately, DNSSEC is simply seen as an optional security upgrade to existing DNS environments. When DNS infrastructure is already functional, it can be challenging to initiate projects that involve optional upgrades that may also require an increase in operational workload, such as periodically changing keys used for DNSSEC. UC administrators who see the benefits of implementing DNSSEC need to work with management and/or server administrators to convince them to help implement it. For more general information on DNSSEC, see these IETF publications: RFC 3833, "Threat Analysis of the Domain Name System," and RFC 4033, "DNS Security Introduction and Requirements."

Securing CMS Connections

To provide secure connectivity between internal CMS components and external applications such as Unified CM and/or Expressway, X.509 certificates are required. Like many other UC applications, CMS supports self-signed certificates, private certificate authority (CA)-signed certificates, and public CA-signed certificates. As we start discussing use of different types of certificates for security connectivity, it makes sense to examine the considerations and caveats when certificates are used with CMS. One consideration is that while other Cisco UC applications install self-signed certificates at the time of installation by default, CMS does not. Another consideration is that the only time self-signed certificates are supported is with single-combined server deployments, which may be useful to quickly secure connectivity for a lab or inside a small UC environment.

One of the most important considerations is that depending on the services that CMS supports, using self-signed certificates may impact the user experience because end users are also required to establish TLS connections to CMS. As an example, when users attempt to connect to a CMS meeting via a web browser, CMS uses self-signed certificates. Either the user's browser blocks the connection because it doesn't trust the

https://t.me/learningnets
@cisco_in_persian_channel

self-signed certificate, or the web browser presents the user with an "untrusted" certificate error message. To establish a secure connection, users have to accept the risk of connecting to an untrusted service before accessing a CMS meeting. For this reason, the use of self-signed certificates is not typically recommended. Lastly, the CMS database does not support self-signed certificates when in a clustered environment. For this reason, Cisco strongly recommends a public key infrastructure (PKI) with CA-signed certificates. The rest of this section focuses on using CA-signed certificates. When working with CA-signed certificates, organizations have the option of using an internal CA server or an external CA server. These CMS components typically use internally CA-signed certificates:

- **Database cluster:** Uses certificates to secure database replication between nodes
- **Call bridge:** Uses certificates to support secure connections to internal CMS components and external call control platforms such as Unified CM
- **Web bridge:** Uses certificates for securing web-based connections into CMS
- **Web admin:** Uses certificates for securing access to the web admin interface

These CMS components may be required to use public CA-signed certificates:

- **Web bridge:** Uses public certificates to improve the user experience for WebRTC and web app connections
- **Call bridge:** Uses public certificates when directly connected to SIP trunks on a public network or federated with Microsoft

Because use of a private CA fits the requirements for most organizations and is the easiest to leverage, it is the general recommendation for organizations to use internal/private CAs to the extent possible. Use of a private CA also allows an organization to obtain X.509 certificates without burdening the organization with additional costs to acquire certificates from a public CA. Although this use is supported, you don't need to create individual certificates for individual CMS components like what is done in other Cisco UC applications (e.g., Tomcat, CallManager, CUP, CUP-XMPP, IPsec). The next portion of this section will focus on creating certificates that can be reused for multiple services for the sake of simplicity.

When an entity such as a CMS server requires a certificate, it first must generate a public/private key pair. It then must create a certificate signing request (CSR), which is signed by the CMS server's private key and then shared with a CA for signing. When CSRs are created, they can contain several different attributes:

- Common name (CN)
- Organization (O)
- Organizational unit (OU)
- Location (L)
- State (ST)

https://t.me/learningnets

@cisco_in_persian_channel

- Country (C)
- Email address
- Subject alternative name (SAN)

Although many of these attributes are available, only the common name is considered mandatory because it includes the FQDN, which specifies the location of the CMS server inside the network by using DNS resolution. While the SAN field is optional, it is equally beneficial for organizations to use. Per RFC 2459, SSL certificates are allowed to specify multiple names that the certificate should match. This means that organizations that would like to simplify how certificates are used within CMS can specify different FQDNs for CMS components (e.g., database, Call bridge, Web bridge) in the SAN field and reduce the number of certificates that are needed within CMS.

When server certificates are reused across CMS components, the CN field should contain the domain name, while the SAN field should contain the FQDNs of all the components that use the certificate. In this regard use of the SAN field is beneficial for simplifying the security that is implemented in a CMS environment.

The following example shows how to create a CSR to allow organizations to share certificates across services from the CLI:

```
cms1a> pki csr cms1a CN:cms1.ciscopucs.com
subjectAltName:cms1a.ciscopucs.com,
ciscopucs.com,call.ciscopucs.com,join.ciscopucs.com.....
.....
Created key file cms1a.key and CSR cms1a.csr
CSR file cms1a.csr ready for download via SFTP
```

After CSRs are created, you, as administrator, must download the CSRs from CMS so that they can be signed by a CA. You can download CSRs by logging in to CMS using administrator privileges with an SFTP client to simplify the download process. Popular SFTP clients include Putty SFTP, Filezilla, and WinSCP. Figure 10-5 shows how the SAN field maps to individual CMS components when CSRs are created.

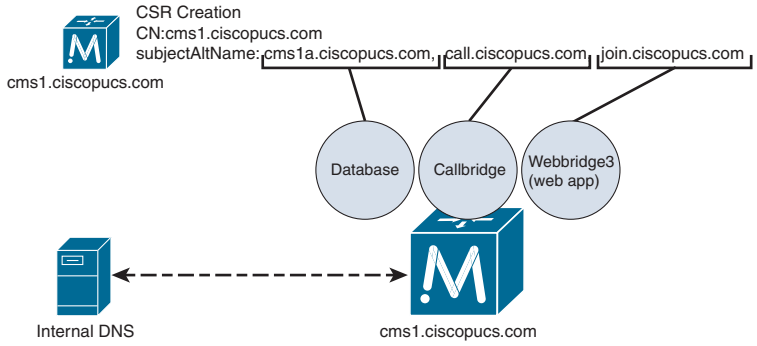


Figure 10-5 *Subject Alternative Name Attribute Within CMS*

https://t.me/learningnets

@cisco_in_persian_channel

Database Security

The CMS database uses PostgreSQL, which is an open-source, object-relational database. Because one of the most important concepts of security is availability of services, organizations should be aware of resilient designs and security implications necessary to ensure high availability of services. This is especially true when considering the CMS database. While it is technically possible to create a database cluster that consists of only two CMS nodes, doing so isn't recommended because this type of design actually reduces resiliency. The reason is that the slave node is not able to check whether it is safe to promote itself to master status to support database requests. Cisco currently recommends having an odd number of database cluster nodes for the purposes of master selection and active failover mechanisms. In this regard, database clusters should consist of at least three or five nodes in a database cluster and ideally with database nodes located at different locations to account for site failure scenarios. These types of considerations are helpful for creating a resilient deployment with predictable and secure database operations.

The CMS database supports clustering with or without using certificates; however, without certificates, a CMS database cluster lacks privacy, access control, and/or authentication of the CMS database between cluster members. For this reason, Cisco strongly recommends the use of certificates to provide security for the CMS database. When database clustering uses certificates, certificates provide authentication of database cluster members and ensure that the database has privacy because of public/private key encryption that is used. For organizations that have a resilient CMS design, administrators should understand that if there are network issues, at any time, one of the database cluster members may be promoted to become the master, or the master could be demoted to a slave.

In this regard, the CMS database has a unique requirement for separate client and server certificates. These certificates must then be shared with each server that is connected to a database in the cluster. These certificates must also be signed by the same CA. It is important to keep this issue in mind when performing CSR requests because the SAN field needs to contain the FQDNs of the other database nodes. The following example shows how to create a CSR from the CLI for a three-node database cluster:

```
cms1a> pki csr dbclusterserver CN:cms1a.ciscopucs.com
subjectAltName:cms1b.ciscopucs.com,cms1c.ciscopucs.com
.......................................... .........
Created key file dbclusterserver.key and CSR dbclusterserver.csr
CSR file dbclusterserver.csr ready for download via SFTP
```

To use certificates that provide authentication for database clients, a CA template needs to have Extended Key Usage enabled with Client Authentication for database clients to participate in the database cluster. Another example of how database certificates are unique is that the CMS database is hard-coded to use "postgres" to perform certificate

https://t.me/learningnets

@cisco_in_persian_channel

validation. Therefore, when you're creating a CSR for this certificate, the CN for the database client must equal "postgres."

For organizations that do not permit use of "postgres" as the CN, because they require something else such as an FQDN or a distinguished name, Cisco CMS provides an internal CA starting in CMS 2.7 software, which can be used to sign database certificates. Using an internal CA within CMS is also an approach to providing increased security by creating a private PKI for the CMS database. When UC administrators are able to closely control and monitor CA-signed certificates through their own private PKI, they can reduce the attack surface on the CMS database and tightly control the information that is replicated across CMS database cluster members. To use the internal CA, you can use the **pki selfsigned** and **pki sign** commands to create a self-signed certificate for an internal CMS root CA and then also to sign database certificates with the internal CMS CA. The following example shows how to generate a self-signed certificate for the internal CA server:

```
cms1a> pki selfsigned dbca CN:ciscopucs.com
.................................................
..........
Created key file dbca.key and selfsigned certificate dbca.crt
```

When you're using the internal CA, you also need to create CSRs for the database server and for the database clients. Similar to creating CSRs with an external, private CA, you need to specify the FQDNs of the other CMS database nodes in the SubjectAltName field as shown:

```
cms1a> pki csr dbserver CN:cms1a.ciscopucs.com
subjectAltName:cms1b.ciscopucs.com,cms1c.ciscopucs.com
....
...
Created key file dbserver.key and CSR dbserver.csr
CSR file dbserver.csr ready for download via SFTP
cms1a>
cms1a> pki csr dbclient CN:postgres
..............

Created key file dbclient.key and CSR dbclient.csr
CSR file dbclient.csr ready for download via SFTP
cms1a>
```

The last step is to sign the CSRs with the internal CA, which we have named dbca. The first CSR was called dbserver. The CSR for the database was called dbclient. The following provides an example:

```
cms1> pki sign dbserver dbca
Signature ok
```

https://t.me/learningnets

@cisco_in_persian_channel

```
subject=/CN=cms1a.ciscopucs.com
Getting CA Private Key
Created certificate dbserver.crt signed with dbca
Created certificate chain file dbserver-fullchain.crt
cms1>
cms1a> pki sign dbclient dbca
Signature ok
subject=/CN=postgres
Getting CA Private Key
Created certificate dbclient.crt signed with dbca
Created certificate chain file dbclient-fullchain.crt
cms1a>
```

After the certificates are downloaded and then distributed to the other CMS nodes, they can securely join the CMS cluster and become CMS database cluster members. We discuss downloading and distributing certificates later in this chapter.

Note The database cluster master listens on TCP port 5432 for connections from slave nodes. If there are ACLs or firewalls between database cluster nodes, you need to make sure that this port is open and that the maximum round-trip time does not exceed 200 ms between any cluster members.

Certificate Verification

When working with certificate administrators and CAs, there is a possibility that signed CA certificates were created with the wrong certificate template, they have had the incorrect certificate attributes applied, or certificate attributes were mangled. From a security perspective, it is a best practice to verify that certificate attributes are correct and that the certificate has not been tampered with to create a rogue certificate.

Rogue certificates are almost indistinguishable by a web browser because they appear to be signed by a trusted CA server. However, if an attacker is able to breach a CA server and its private keys that are used to sign certificates, rogue certificates can be easily distributed. When this occurs, users are redirected to unauthorized sites so that the attacker can perform a man-in-the-middle attack by snooping credentials, which can possibly be used to infiltrate an organization or one of its meetings. This type of attack is similar to DNS poisoning, which we previously discussed.

For these reasons, it is recommended that you verify that the private key matches the certificate file and a certificate chain is fully trusted before assigning certificates to CMS components. To validate the PKI from within CMS, you can leverage the **pki match**

https://t.me/learningnets

@cisco_in_persian_channel

command. The following example shows how to validate that the private key for cms1a matches the CA-signed certificate (cms1a.cer):

```
cms1a> pki match cms1a.key cms1a.cer
Matching certificate and private key
cms1a>
```

In some PKI environments, a certificate may be signed by a certificate authority, and the CA provides a certificate bundle of intermediate CA certificates and perhaps a CA certificate in its own file. To check that the CMS certificate is signed by an authentic CA and that the certificate bundle can be used to assert this, you can use the **pki verify** command:

```
cms1a> pki verify cms1a.cer cms1abc.cer cms-rootCA.cer
Success
cms1a>
```

With CMS, a certificate bundle is a file that holds a copy of multiple individual certificates in a single file. External clients, such as web browsers and XMPP clients, require a server certificate and certificate bundle to be presented by the Web bridge and XMPP server, when setting up a secure connection. In many cases, organizations have both root and intermediate CAs in the trust chain, so a single bundled certificate should be used to establish a trust chain between CAs involving the intermediate CA and root CA certificates. This idea of a CA bundle file will be more evident when we start assigning certificates to CMS components later in the chapter. We also show how to use certificate bundles to trust connections between the Call bridge components. The requirements for bundling certificates and then assigning them for use within CMS include

- Certificates must have an extension of .cer, .pem, or .crt.
- Certificates must be in a specific sequence with the root CA being the last certificate listed inside the bundle.

To bundle certificates together, you must create a file that contains all of the certificates that should be bundled together. As an example, to combine individual CMS server certificates (e.g., cma1a.cer, cms1b.cer, cms1c.cer) into one certificate bundle of trusted certificates, you can use the following options from each operating system:

- **Any OS:** Use a text editor to copy/paste certificates into a single file.
- **Windows command prompt:** Use **copy cms1a.cer + cms1b.cer + cms1c.cer cms1abc.cer**.
- **Linux-based operating system:** Use **cat cms1a.cer cms1b.cer cms1c.cer > cms1abc.cer**.

https://t.me/learningnets

@cisco_in_persian_channel

A bundled certificate is shown in Figure 10-6.

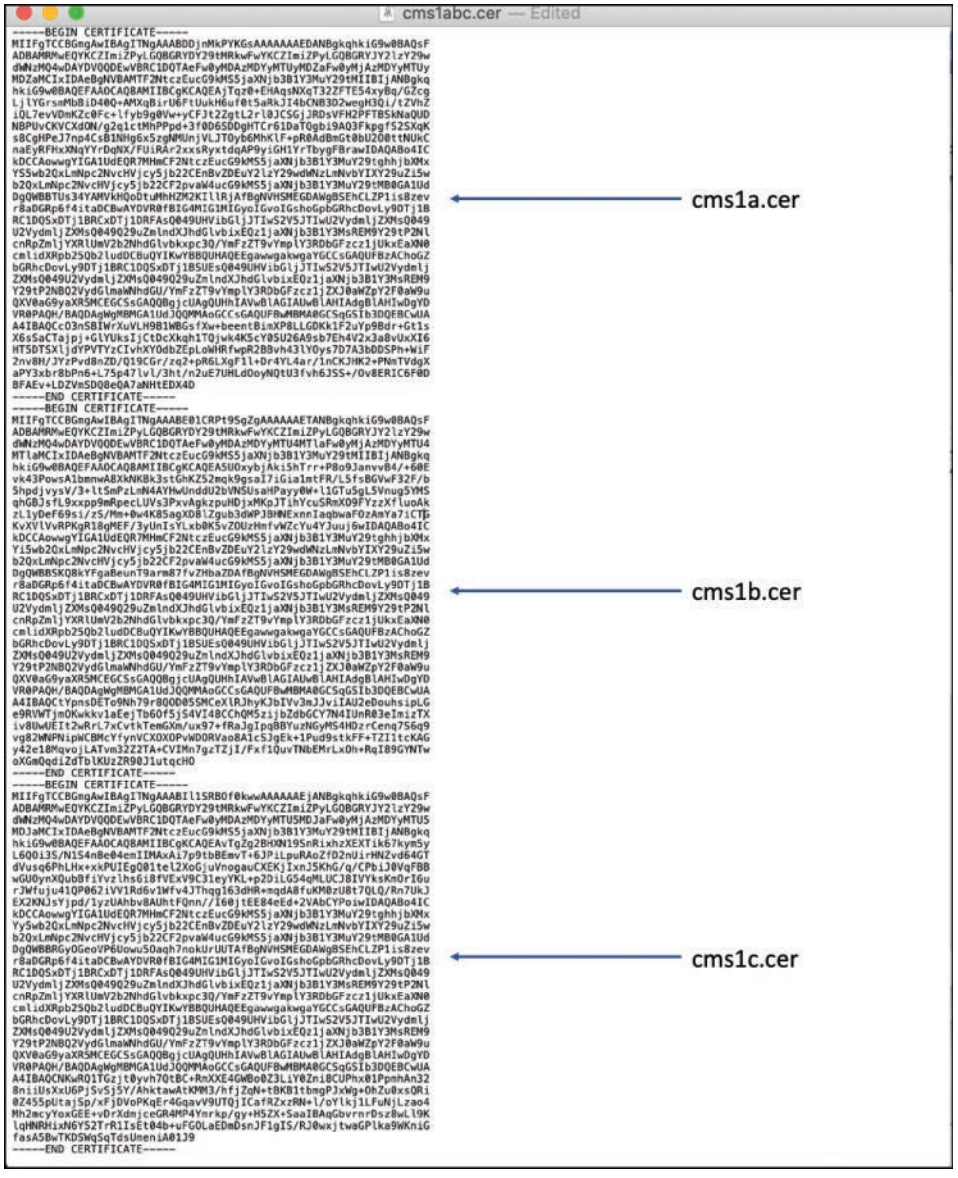


Figure 10-6 *Certificate Information Contained Inside a Certificate Bundle*

After verifying that CA-signed certificates contain all of the correct information from the certificate administrator(s), they must be installed onto each CMS node in the

https://t.me/learningnets

@cisco_in_persian_channel

UC cluster. To install CMS server certificates, certificate bundles, and associated CA server certificates that are needed for TLS security, you must upload certificates to CMS via SFTP using administrator credentials. Figure 10-7 shows how to use the Filezilla client to upload certificates via SFTP onto a single CMS server.

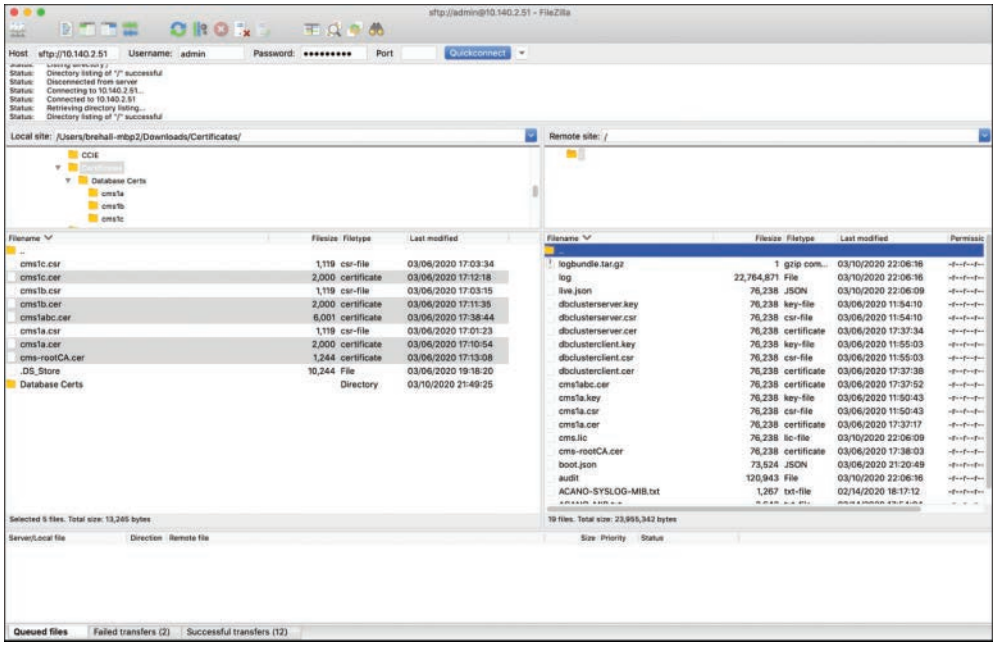


Figure 10-7 *Using Filezilla to Upload Signed Certificates into CMS Servers*

Transport Layer Security

Over the course of this chapter and book, we have recommended use of TLS security for encrypted communications. Since version 2.3, the default level of TLS encryption for CMS is TLS 1.2 and DTLS 1.2 encryption for all services. A primary reason of this enhancement is due to the weaknesses in previous versions of TLS. Examples of significant exploits against weak versions of TLS include Padding Oracle On Downgraded Legacy Encryption (POODLE) and Browser Exploit Against SSL/TLS (BEAST).

Although this approach is not necessarily recommended due to security weaknesses, using previous versions of TLS is more secure than not using TLS at all. Some organizations may be required to leverage earlier versions of TLS for interoperability with older UC endpoints, IP gateways, or infrastructure. As an example, Unified CM versions prior to 11.5(1)SU3 do not support TLS 1.2; therefore, TLS 1.2 encryption is not supported between Unified CM and CMS's Call bridge component. If this is the case and organizations require interoperability, they can either specify a different version of TLS within CMS or upgrade Unified CM.

https://t.me/learningnets

@cisco_in_persian_channel

CMS provides backward compatibility of TLS versions for SIP, LDAP, and HTTPS. The other two options are to either disable TLS verification or not to use TLS connections in general. For added security for SIP and LDAP services, CMS provides the ability to perform mutual authentication between services. This security feature is called TLS verification. When this feature is enabled, the Call bridge requests a certificate from a remote entity. If CMS cannot verify that the certificate from the connecting system matches the certificate inside its trust store, the connection is refused. We discuss use of the trust store in a later section.

Example 10-2 shows how you, as administrator, can specify minimum versions of TLS for LDAP and SIP connections. We also provide the syntax needed to disable TLS verification for LDAP if needed. For a complete list of TLS commands, see the CMS MMP command reference.

Example 10-2 *Specifying Minimum Versions of TLS Encryption*

```
cms1a> tls
Usage:
    tls
    tls (sip|ldap|dtls|webadmin)
    tls (sip|ldap) trust <crt bundle>
    tls (sip|ldap) verify (enable|disable|ocsp)
    tls sip ciphers <cipher string>
    tls (sip|ldap|webadmin) min-tls-version <minimum version string>
    tls min-dtls-version <minimum version string>
cms1a>
cms1a> tls sip min-tls-version 1.1
cms1a> tls ldap min-tls-version 1.0
cms1a> tls ldap verify disable
cms1a>
```

For additional information on TLS compatibility, review Cisco's TLS 1.2 Compatibility Matrix for Cisco Collaboration products.

Certificate Assignment

Now that you understand which TLS versions can be customized for specific protocols and that all of the relevant X.509 certificates have been uploaded into CMS, you can now specify which certificates should be used for securing the various components of CMS. To ensure that configurations are consistent across all CMS nodes, you start with the CMS database. The CMS database is initialized from the CLI. Example 10-3 shows how to associate CMS database certificates, and Example 10-4 shows how to initialize the database cluster.

https://t.me/learningnets

@cisco_in_persian_channel

Example 10-3 *Assigning Certificates to the Database*

```
cms1a> database cluster
Usage:
    database
    database cluster join <hostname/IP>
    database cluster connect <hostname/IP>
    database cluster remove
    database cluster initialize
    database cluster localnode <interface>
    database cluster status
    database cluster upgrade_schema
    database cluster clear_error
    database cluster certs none
    database cluster certs <client_key> <client_crt> <ca_crt>
    database cluster certs <server_key> <server_crt> <client_key> <client_crt>
      <ca_crt> cms1a>
cms1a> database cluster certs dbclusterserver.key dbclusterserver.cer
  dbclusterclient.key dbclusterclient.cer cms-rootCA.cer
Certificates updated
cms1a>
```

Example 10-4 *Initializing a Database Cluster*

```
cms1a> database cluster localnode a
Localnode updated
cms1a>
cms1a> database cluster initialize
WARNING!!! Are you sure you wish to initialize this node as a new database
  cluster? (Y/n) Y
  The contents of this node's database will become the master version of the
   database in the new cluster.
  The callbridge and web administration will restart at the end of this procedure.
   Server certificate/key validated..
   Client certificate/key validated..
Please wait...
Initialization started...
```

Depending on the size of your database cluster, you can join additional CMS nodes to the database, for a total of five database servers. To join additional servers, you can use the **database cluster join <hostname/IP>** command. As shown in Example 10-5, the **database cluster status** command can be used to verify the status of the cluster and also to determine whether the additional CMS nodes are in sync with the database.

https://t.me/learningnets

@cisco_in_persian_channel

Example 10-5 *Using the* ***database cluster status*** *Command to Check Health of a CMS Cluster*

```
cms1a> database cluster status
Status : Enabled

Nodes:
    10.140.2.51 (me)      : Connected Master
    10.140.2.52           : Connected Slave ( In Sync )
    10.140.2.53           : Connected Slave ( In Sync )
Node in use               : 10.140.2.51
Interface                 : a

Certificates
Server Key                : dbclusterserver.key
Server Certificate        : dbclusterserver.cer
Client Key                : dbclusterclient.key
Client Certificate        : dbclusterclient.cer
CA Certificate            : cms-rootCA.cer
Last command : 'database cluster initialize'
(Success)
```

The web admin component of CMS enables you, as administrator, to configure basic tasks and monitoring of calls using a GUI. Web admin should be enabled on all of the CMS servers because it also helps you gain access to the CMS API, which is useful for CMS configuration tasks and is used for enabling capabilities such as ad hoc conference bridging with Unified CM. The process for assigning X.509 certificates to the web admin service is similar to the process that we just discussed for the database server. By assigning certificates to the web admin service, you can securely access the GUI to perform administration. To understand options for the syntax, you can use the **help** keyword to provide optional configuration items for web admin, as shown in Example 10-6.

Example 10-6 *Assigning Certificates to Web Administration*

```
cms1a> help webadmin
Configure webadmin

Usage:
    webadmin
    webadmin restart
    webadmin enable
    webadmin disable
    webadmin listen <interface> [<port>]
    webadmin certs <key-file> <crt-file> [<cert-bundle>]
    webadmin certs none
```

https://t.me/learningnets

@cisco_in_persian_channel

```
    webadmin http-redirect <enable/disable>
    webadmin status
cms1a>
cms1a> webadmin certs cms1a.key cms1a.cer cms-rootCA.cer
cms1a>
```

The process that is needed to initialize the web GUI is also similar to the database cluster. A subtle difference arises if you plan to host web-based, user-facing services. It is a best practice to reserve well-known port numbers such as TCP 443 for user-facing services such as WebRTC and the CMS web app that are supported by Web Bridge/Web Bridge 3. We discuss use of the Web Bridge 3 shortly.

To ensure that administrative services do not conflict with user-facing services at the transport layer, the recommendation is to specify a unique, nondefault port number for the web GUI such as TCP 8080 or TCP 8443. The following example shows how to use a customized port number for the web admin interface:

```
cms1a> webadmin listen a 8443
cms1a>
cms1a> webadmin restart
SUCCESS: TLS interface and port configured
SUCCESS: Key and certificate pair match
SUCCESS: certificate verified against CA bundle
cms1a>
```

As shown in the following output, the **webadmin status** command can be used to verify the status of the service. One thing that you may notice is that HTTP redirect is currently disabled. Although it is typically a best practice to enable HTTP redirect for security purposes, it is important to remember that you are reserving TCP 443 for end-user access. Attempts to enable HTTP redirect also result in a transport layer conflict between the web admin and Web bridge, so this feature can be left disabled.

```
cms1a> webadmin status
Enabled                       : true
TLS listening interface       : a
TLS listening port            : 8443
Key file                      : cms1a.key
Certificate file              : cms1a.cer
CA Bundle file                : cms-rootCA.cer
HTTP redirect                 : Disabled
STATUS : webadmin running
```

As shown in Figure 10-8, after using these commands, you can verify that you have secure access to the web admin GUI. Although CMS is accessible from an IP address, remember you previously configured FQDNs in the SAN field of the CSR. To verify TLS

https://t.me/learningnets

@cisco_in_persian_channel

connectivity, you should browse to the FQDN and the custom TCP port that was previously defined. The lock icon verifies that the TLS connection is trusted and has been established.

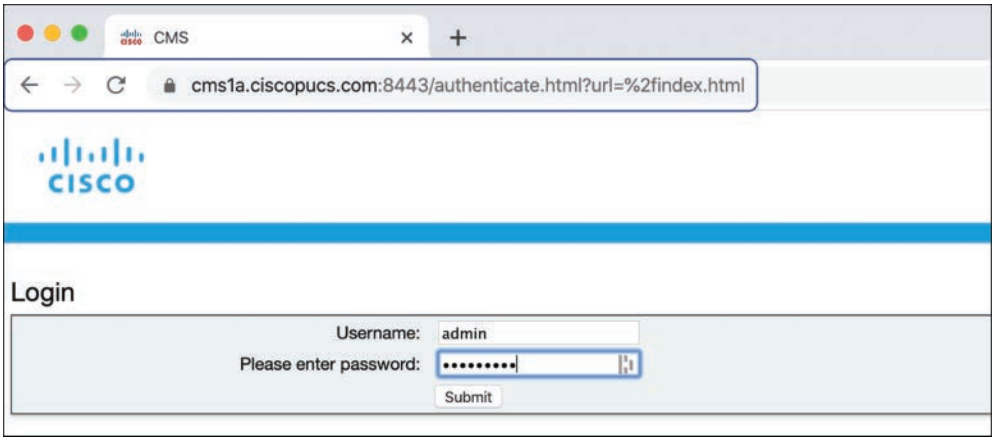


Figure 10-8 *Verifying Access to the Web Admin GUI After Assigning Certificates and Initializing the Service*

The CMS Call bridge is the main component of CMS, which provides audio and video conferencing services. The Call bridge also provides connectivity to external call control platforms, such as the Cisco Unified CM or Expressway. In a scalable and resilient deployment, Call bridges can be clustered together so that all Call bridges operate in unison as a single entity. The way that Call bridges connect with each other is already secure since Call bridges use HTTPS to encrypt traffic, but CMS does not require use of certificates to authenticate the identities of Call bridge servers within a cluster by default. Organizations have the option of adding an additional layer of security to the Call bridge cluster by using the Call bridge trust store to authenticate the identities of Call bridges within the cluster. In order to establish secure connections, a certificate bundle holding the certificates of the web admin servers for the clustered Call bridges must be created.

When we first started discussing all of the various CMS components, we provided a diagram showing how CMS components communicate with each other. In the diagram, the C2W protocol was shown as the mechanism that facilitates connectivity between the Call bridge and the Web Bridge 3. CMS allows the network port for C2W connections to be customized on an as-needed basis at the transport layer (e.g., 9999). Cisco recommends that administrators configure this connection between the Call bridge and Web Bridge 3 so that it is only accessible from the Call bridge. How we do this is with certificate-based authentication.

Starting in CMS 3.0, CMS provides organizations with added flexibility for how they can choose to use certificates to secure connections between the Call bridge and Web Bridge 3. One example is that the C2W trust store no longer requires root certificates.

https://t.me/learningnets
@cisco_in_persian_channel

This provides organizations with better flexibility with how to establish trusted connections using certificates. Using an approach called certificate pinning, organizations can transition from simply securing connections to securing connections with specific certificates that the CMS is expecting. A benefit of using this type of custom approach for securing connections is that it helps prevent attacks by perpetrators attempting to intercept traffic to conduct man-in-the-middle attacks. An example is shown in Figure 10-9.

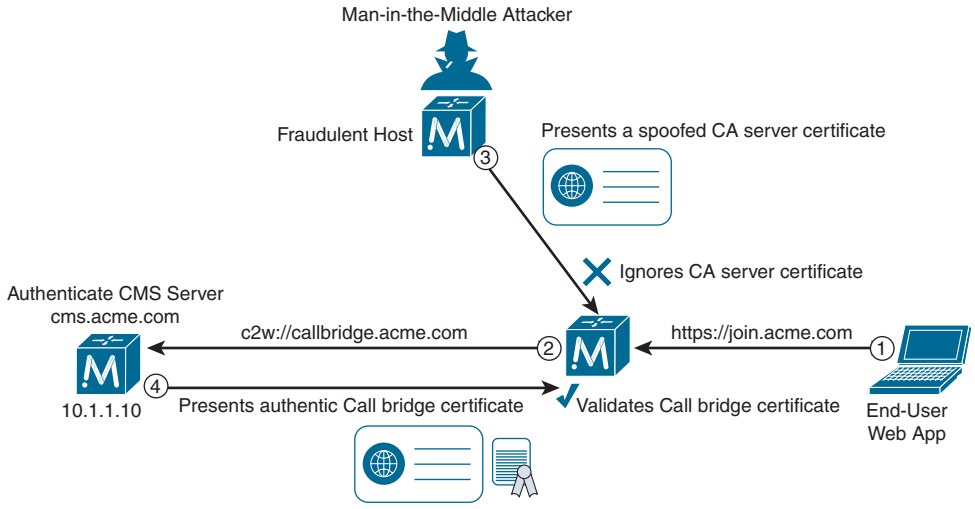


Figure 10-9 *Using Custom C2W Certificates to Mitigate Man-in-the-Middle Attacks*

Up to this point, we have only really discussed securing the internal connections between the Call bridge and CMS components such as Web Bridge 3. Signaling connections between external infrastructure such as Unified CM and/or Expressway can be secured by using TLS over SIP trunks. In order to facilitate a trusted connection between Unified CM and CMS, Unified CM requires that CAs that sign Call bridge certificates use templates that support both Server Authentication and Client Authentication.

As UC endpoints connect to CMS meetings, media streams are secured using SRTP. Starting in CMS 2.9, CMS supports stronger encryption ciphers such as

- AED_AES_128_GCM
- AED_AES_256_GCM
- AES_CM_128_HMAC_SHA1_80
- AES_CM_128_HMAC_SHA1_32

In order to determine the cipher suite used for inbound/outbound SRTP calls, administrators can use the CMS API to query specific call legs by querying the /callLegs/[call leg ID] object. An example of output from the API interface provided by the Web admin GUI is included in Figure 10-10.

https://t.me/learningnets
@cisco_in_persian_channel

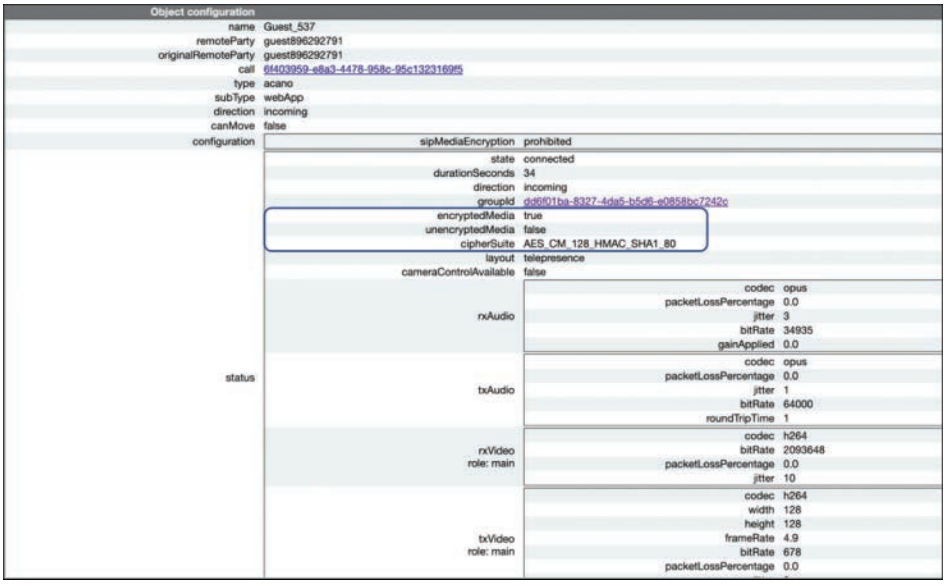


Figure 10-10 *Determining the Cipher Suite Used for a CMS Call Leg Using the Web Admin's API Screen*

Example 10-7 shows the configuration of the items that we have discussed while assigning certificates and initializing the Call bridge.

Example 10-7 *Assigning Certificates and Initializing the CMS Call Bridge*

```
cms1a> callbridge listen a
cms1a> callbridge certs cms1a.key cms1a.cer cms-rootCA.cer
cms1a> callbridge trust c2w cms1abc.cer
cms1a> callbridge trust cluster cms1abc.cer
cms1a>
cms1a> callbridge restart
SUCCESS: listen interface configured
SUCCESS: Key and certificate pair match
SUCCESS: certificate verified against CA bundle

Verifying status of the call bridge:
cms1a> callbridge
Listening interfaces    : a
Preferred interface     : none
Key file                : cms1a.key
Certificate file        : cms1a.cer
Address                 : none
C2W trusted certs                : cms1abc.cer
Callbridge cluster trusted certs : cms1abc.cer
```

https://t.me/learningnets

@cisco_in_persian_channel

The CMS web app that allows users to join CMS meetings through a web browser uses the new Web Bridge 3 component. We already discussed use of C2W protocol from a Call bridge perspective. Web Bridge 3 provides additional certificate requirements to ensure that the C2W connection is secure. As an example, Web Bridge 3 requires a bundle of certificates, which includes a full certificate chain. Lastly, you can enable HTTP redirection to ensure secure web app connections and also specify the C2W port to listen on. Example 10-8 shows how to configure the Web Bridge 3 for secure connections.

Example 10-8 *Assigning Certificates and Initializing the CMS Web Bridge 3*

```
cms1a> webbridge3 https listen a:443
cms1a> webbridge3 http-redirect enable
cms1a> webbridge3 https certs cms1a.key cms1a-fullchain.cer
cms1a> webbridge3 c2w listen a:9999
cms1a> webbridge3 c2w certs cms1a.key cms1a-fullchain.cer
cms1a> webbridge3 c2w trust cms1abc.cer
cms1a> webbridge3
cms1a> webbridge3 enable
SUCCESS: HTTPS Key and certificate pair match
SUCCESS: HTTPS full chain of certificates verifies correctly
SUCCESS: C2W Key and certificate pair match
SUCCESS: C2W full chain of certificates verifies correctly
SUCCESS: Webbridge3 enabled
cms1a> webbridge3
Enabled                          : true
HTTPS Interface whitelist        : a:443
HTTPS Key file                   : cms1a.key
HTTPS Full chain certificate file : cms1a-fullchain.cer
HTTP redirect                    : Enabled, Port:80
C2W Interface whitelist          : a:9999
C2W Key file                     : cms1a.key
C2W Full chain certificate file  : cms1a-fullchain.cer
C2W Trust bundle                 : cms1abc.cer
Beta options                     : none
```

At this point, one thing to notice is that the whitelist feature works with the internal CMS firewall to automatically open TCP ports 443 and 9999.

Application Programming Interfaces (APIs)

Up to this point, we have had a strong focus on using the CMS CLI to perform initial configurations. We have also used various parameters to increase the security of the OS and configurations and to establish secure connections between components using certificates. CMS provides a Representational State Transfer (REST) API that allows for intermediate configuration of CMS features that can provide both basic and advanced

https://t.me/learningnets

@cisco_in_persian_channel

functionality along with security. There are many benefits of using APIs for CMS configuration, some of which include programmability, automation, best practice sharing, and platform scalability. For the sake of this discussion, we remain focused on the features that organizations can use to provide security controls accessing and using the CMS API.

Organizations have many different options to choose from when configuring CMS using the API. Starting in CMS 2.9 software, Cisco provides the ability to use the CMS API from the web admin interface without a third-party API client such as Postman. To access the CMS API tab from the web admin GUI, you, as administrator, can navigate to **Configuration > API**. This feature provides a simpler workflow for configuration tasks. It is simpler because Cisco provides the API structure to make it easier to navigate within the web GUI instead of always cross-referencing the CMS API guide for instruction. When you need to perform a task, you can simply browse to the web admin and apply configurations. An example of a security feature that can be configured through the API is call leg profiles. A call leg profile defines a set of in-call behaviors for CMS meetings. If needed, you can enable encryption for specific CMS spaces via the call leg profile setting within CMS using the web admin and the newly created API tab. An example is provided in Figure 10-11, and an example of how this call leg profile can be applied to a space is included in Figure 10-12.

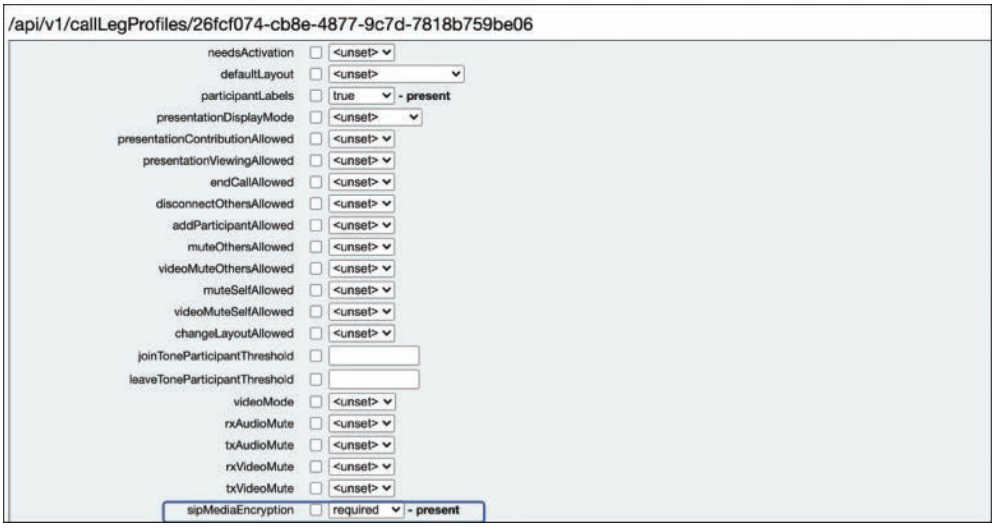


Figure 10-11 *Using the Web Admin Interface to Require Encryption on a Call Leg Profile*

Standalone tools such as Postman, RESTer, RESTED, curl, and Powershell are also still viable options for configuring CMS through the API. To ensure a secure connection to CMS, CMS uses HTTPS as the underlying transport protocol to the API interface. To use the CMS API, you need to connect via HTTPS with the TCP port that has previously been specified for the web admin GUI. To authenticate to the CMS API, a username and password that have administrative access to the API are required. The most simple and

https://t.me/learningnets
@cisco_in_persian_channel

secure way of providing access to the API is to add one or more specific administrator accounts with API privileges via the CLI. You do this with **user add <username><role>**, while being sure to specify **api** as the assigned role. Because CMS uses HTTPS, the authentication credentials are obfuscated from external entities.

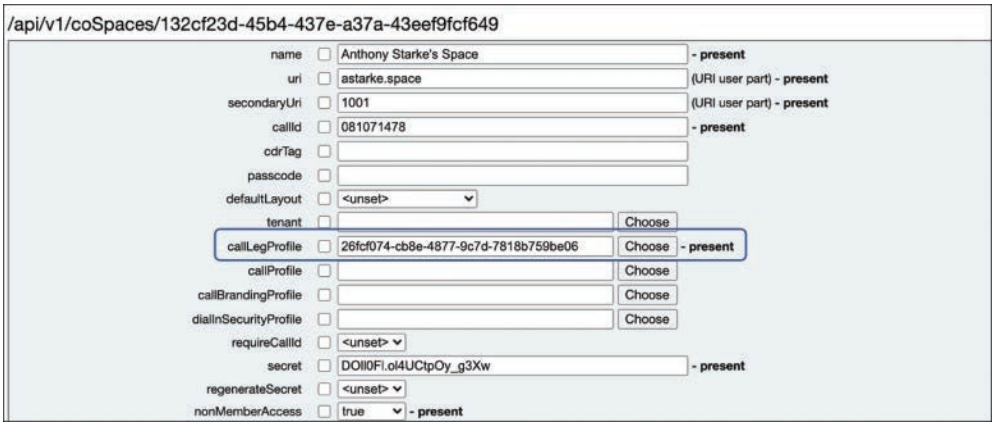


Figure 10-12 *Assigning a Call Leg Profile to a CMS Space to Require Encrypted Calls*

LDAP integration also can be configured via the CMS API to provide increased security. Integrating CMS with LDAP servers is a security benefit so that users can log in to CMS meetings with domain credentials. CMS is able to integrate with the Microsoft Active Directory, OpenLDAP, or Oracle Internet Directory servers to import users into the CMS database.

Although many organizations use LDAP for directory synchronization, there is a push for organizations to migrate toward Secure LDAP (LDAPS) so that synchronizations are not as susceptible to man-in-the-middle attacks. Manufacturers such as Microsoft are currently in the process of updating the security requirements to establish connections to Active Directory over Secure LDAP to make their systems more secure. CMS supports the use of LDAPS over TCP 636 when connecting to an LDAP server such as Microsoft Active Directory. Figure 10-13 shows how to configure a secure LDAP server using CMS's API parameters (e.g., address, port number, username, password), while using the Postman client.

If the parameters are specified correctly and the API POST method is successful, you receive a "200 OK" response from CMS. If there is an error or mistake in the API request, CMS responds with a 4xx or 5xx status code. To complete the secure LDAP integration, you need to create an LDAP mapping and LDAP source with the API. When this task is complete, and LDAP synchronization is initiated, users are able to perform secure LDAP authentication into CMS. For more information about the parameters needed to accomplish these tasks, review the CMS API guide for your version of software.

https://t.me/learningnets

@cisco_in_persian_channel

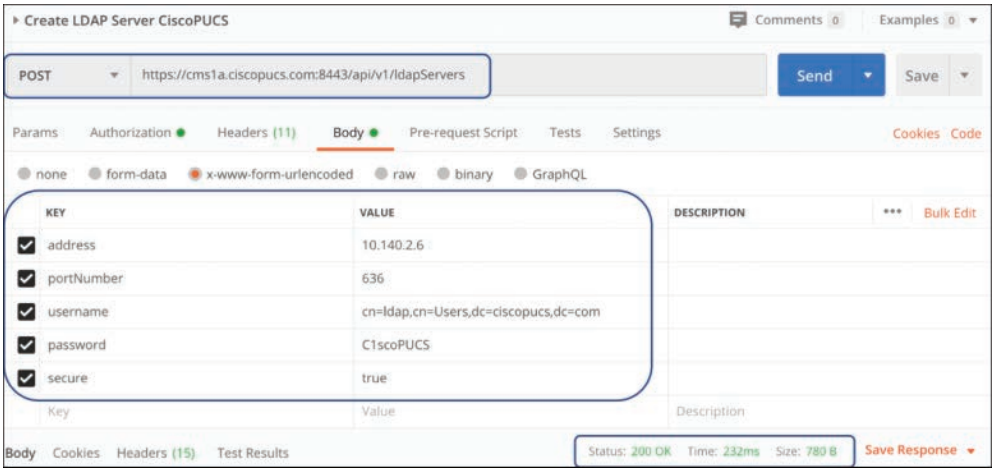


Figure 10-13 *Successfully Adding a LDAP Server into CMS Using Postman and the API Interface*

Inbound and Outbound Calling

For organizations to fully utilize CMS to provide rich-media conferencing, they need to establish connectivity to external call control platforms. For example, CMS needs to integrate with Cisco Unified CM or Expressway so that the UC devices that they are managing are able to place inbound calls into CMS meetings or also to be called from within a CMS meeting. When this integration is done, users and endpoints that are interested in joining a CMS meeting (e.g., Cisco Jabber client, IP phone, video endpoints) are able to communicate with native CMS clients such as the Cisco Meeting App or WebRTC-based clients to collaborate within a CMS meeting. This next section discusses how to configure CMS to accept inbound calls in a secure manner.

CMS supports inbound calls that are either unencrypted or encrypted. For the most part, we have been advocating for all things to be secured using TLS. Understandably, this isn't always possible due to use cases that require unencrypted media, such as older UC endpoints, for call recording or streaming media. With that said, since we have already discussed how to modify call leg profiles, this next section will discuss how to configure CMS for encrypted media so that all inbound calls can be secured. In this section we cover how to provide encrypted media so that inbound calls can be secured.

Because you have already performed the requisite certificate work for the CMS Call bridge component, the CMS environment is ready to support SIP-TLS signaling and secure RTP (SRTP) media. To review the certificate that is currently in use by Call bridge, you leverage the **callbridge** command to identify the Certificate file and then use **pki inspect** to identify the common name used in the certificate. This way, you can make sure that the CN is in FQDN format and that it matches the CMS server's Call bridge component so that you can start configuring SIP-TLS connections from Unified CM.

https://t.me/learningnets
@cisco_in_persian_channel

Example 10-9 shows how to check which certificate is in use and what is inside the certificate.

Example 10-9 *Checking CMS Certificates and the Information Inside the Certificates*

```
cms1a> callbridge
Listening interfaces  : a
Preferred interface   : none
Key file              : cms1a.key
Certificate file      : cms1a.cer
Address               : none
CA Bundle file        : cms-rootCA.cer
XMPP trusted certs                  : none
Callbridge cluster trusted certs    : none
cms1a>
cms1a> pki inspect cms1a.cer
Checking ssh public keys...not found
Checking user configured certificates and keys...found
File contains a PEM encoded certificate
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            36:00:00:00:1a:29:07:fa:ba:7d:81:e7:9b:00:00:00:00:00:1a
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: DC=com, DC=ciscopucs, CN=AD-CA
        Validity
            Not Before: Mar 23 17:32:19 2020 GMT
            Not After : Mar 23 17:32:19 2022 GMT
        Subject: CN=cms1a.ciscopucs.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
```

Note The rest of the **pki inspect** output in Example 10-9 has been omitted for brevity.

To enable media encryption on CMS, you can use CMS's web admin interface. To enable SIP media encryption from the CMS web admin, you can navigate to **Configuration > Call Settings** and specify that SIP media encryption is **Required**, as shown in Figure 10-14. When enabled, this parameter requires that SRTP be negotiated in the SIP offer for a call to succeed.

https://t.me/learningnets

@cisco_in_persian_channel

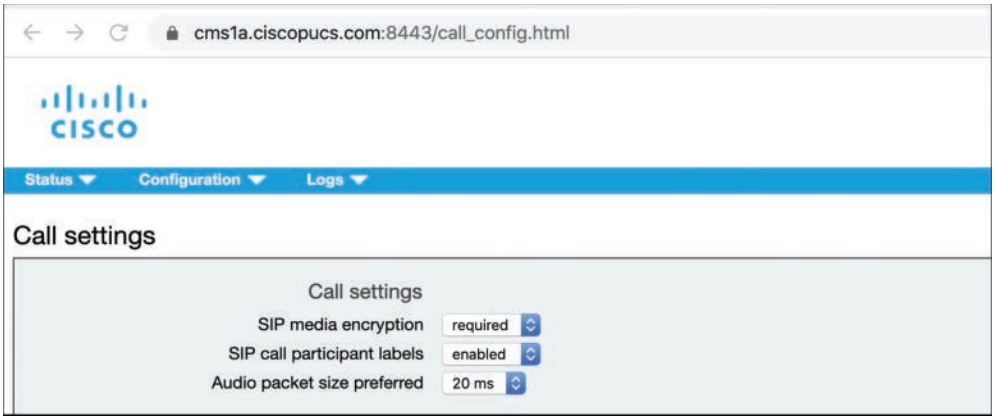


Figure 10-14 *Specifying SIP Media Encryption Requirements Within CMS*

The last step within CMS that is required to support secure inbound calls is to create rules for which domain(s) should be referenced for the incoming calls. You can do this by browsing to **Configuration > Incoming Calls** or by leveraging the CMS API. This step is important because due to security purposes, if the domain parameter is not configured, the Call bridge refuses the call. With external call control solutions such as Unified CM, the destination of an SIP trunk (to CMS) may be the IP address or host name of the server. With CMS, it prioritizes the highest-priority domain as the main domain. Therefore, the proper domain name needs to be specified to accept the inbound call. If calls can possibly be routed to the IP addresses of the CMS servers, it is a good idea to also specify the IP addresses of the CMS servers just in case. An example is shown in Figure 10-15.

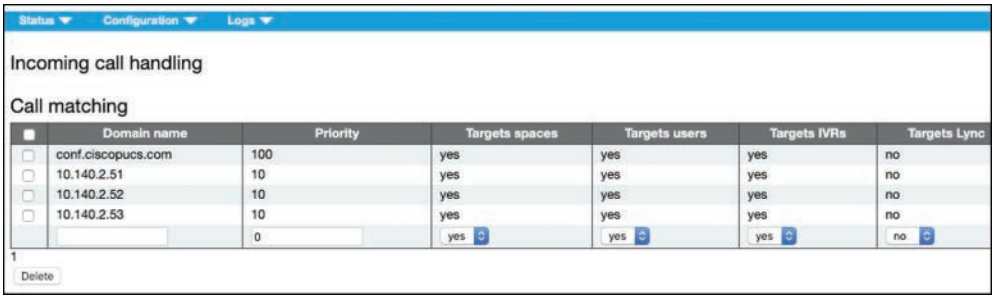


Figure 10-15 *Specifying Inbound Call Handling Rules Within CMS*

Outbound calling is a powerful feature that is provided by CMS, but it could also be used to inadvertently cause issues. Two specific issues involve toll fraud and call routing loops. The first issue around call fraud is pretty simple to understand. Because it is possible to originate an outbound call from CMS, you should consider whether the UC architecture supports outbound dialing from CMS to the PSTN. Any outbound access

https://t.me/learningnets
@cisco_in_persian_channel

to the PSTN from CMS may actually result in toll fraud. Typically, CMS should be able to dial outbound only to external call control platforms to communicate with other UC endpoints or possibly to integrate with third-party platforms such as Skype for Business when CMS is acting as a gateway.

When integrating with Unified CM/SME, administrators can leverage the ability to apply Calling Search Spaces (CSSs) on SIP trunks to limit access to the PSTN. When you're considering use of CSSs to prevent toll fraud, it may be best to provide an analogy based on the concept of least privilege. CSSs contain a list of partitions that are assigned to individual lines of a phone and also route patterns that are used by Unified CM to route a call to a PSTN gateway. To limit attempts at toll fraud, the SIP trunk that connects to CMS should be configured with only the partitions internal to the network that belong to UC endpoints. The CSSs should not have access to any partitions that may be used to route outbound calls to the PSTN. An example is shown in Figure 10-16.

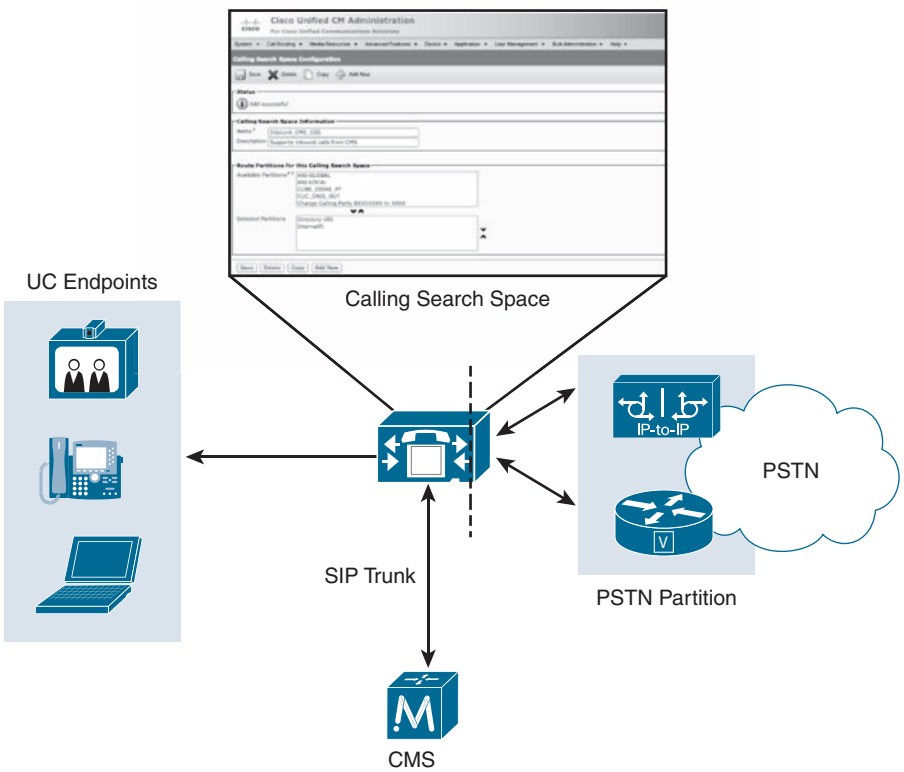


Figure 10-16 *Using CSSs Within Unified CM to Mitigate the Risk of Toll Fraud*

https://t.me/learningnets

@cisco_in_persian_channel

Additionally, if CMS needs to integrate with multiple different call control platforms (e.g., Unified CM and Expressway), administrators should take care that outbound forwarding rules do not inadvertently match an outbound calling rule and create an issue with call loops. Organizations that integrate with Expressway have the option of configuring a "source" value within Expressway's search rules to address this topic.

Unified CM Configuration

The most common external call control platform for CMS is Cisco Unified CM because it is at the center of the UC architecture. This is also the recommendation in Cisco's Solution Reference Network Design Guide for CSR 12.0 software. The same process and configuration can be used when leveraging Cisco's Session Manager Edition (SME) solution as well because the SME software is essentially the same as Unified CM but is responsible for aggregating SIP trunks instead of providing line side services. Alternatively, use of Expressway-C and -E software is possible also if organizations need to extend conferencing to the edge of their environment. These options, along with network connectivity and protocols that provide security, are shown in Figure 10-17.

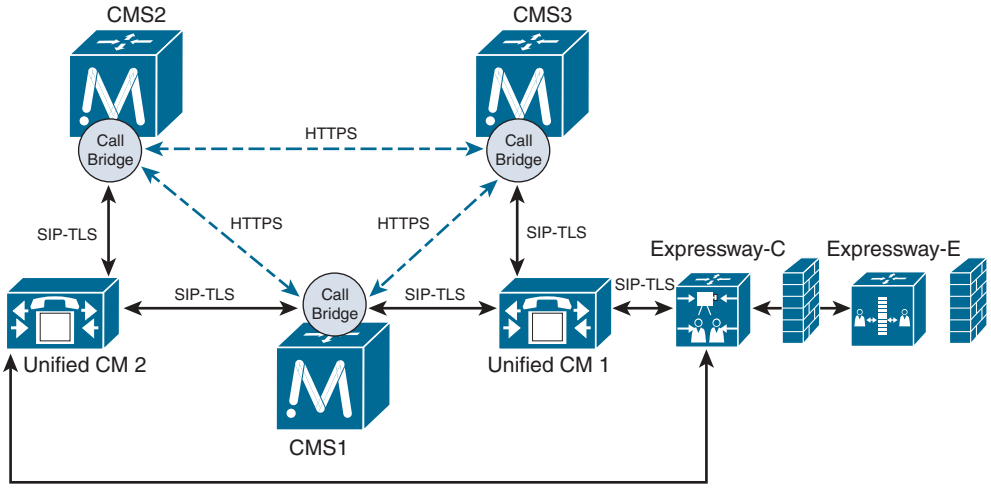


Figure 10-17 *Integrating with Unified CM to Support Secure Inbound/Outbound Calling*

If Unified CM has not been configured for certificates yet, the root CA and any other intermediate certificates must be uploaded to the certificate store as a CallManager-Trust certificate. This page can be found under **Cisco Unified OS Administration > Security > Certificate Management > Upload Certificate/Certificate chain.** The next step is to create an encrypted SIP trunk profile, as shown in Figure 10-18, which can be found under **Cisco Unified CM Administration > System > Security > SIP Trunk Security Profile.** Be sure that the Device Security Mode is Encrypted, incoming and outgoing transport

https://t.me/learningnets
@cisco_in_persian_channel

types are both TLS, the Secure Certificate Subject or Subject Alternative Name matches the CN of the CMS server, and the incoming port is 5061.

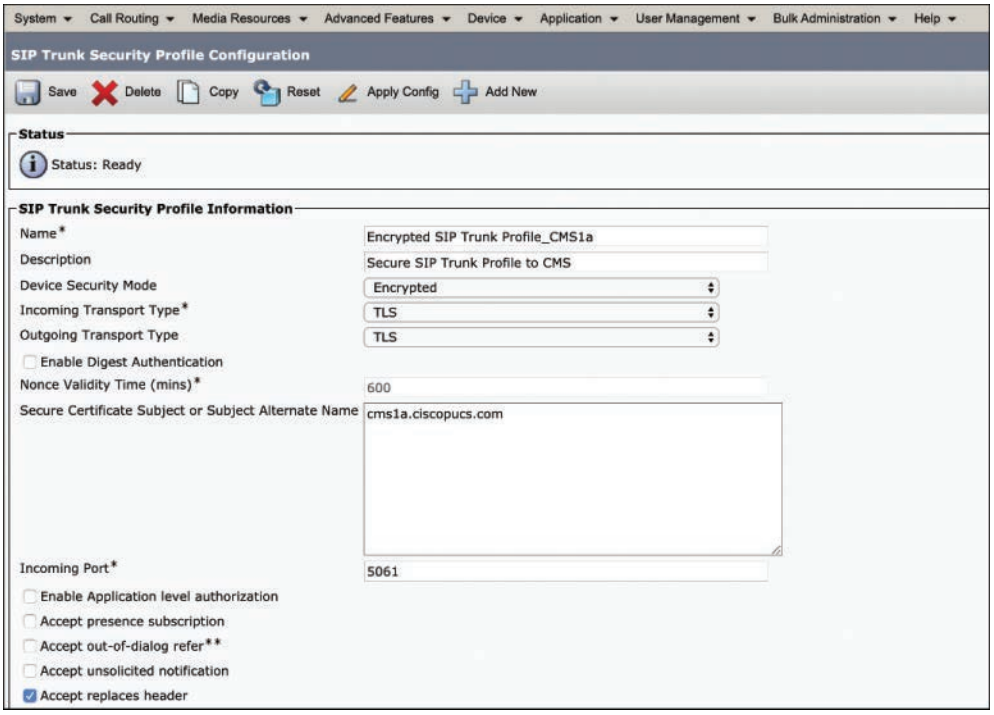


Figure 10-18 *Creating an SIP Trunk Security Profile Within Unified CM*

If Unified CM/SME needs to integrate with more than one CMS server, you need to add the Subject Alternative Name for the additional CMS servers, with the unique CNs in the SAN field of the SIP Trunk Security Profile.

After taking this step, you can create one or more SIP trunks that point to CMS nodes. Be sure to apply the unique SIP Trunk Security Profile on each SIP trunk that connects to each CMS server. Also, be sure to check the **SRTP Allowed** checkbox to get encrypted calls to work. An example of this configuration is shown in Figure 10-19.

When the SIP trunk is fully operational, Cisco Unified CM's routing logic can be followed to provide call routing into the CMS Space. To verify functionality, you can browse to **Status > Calls** from the CMS web admin, as shown in Figure 10-20.

https://t.me/learningnets

@cisco_in_persian_channel

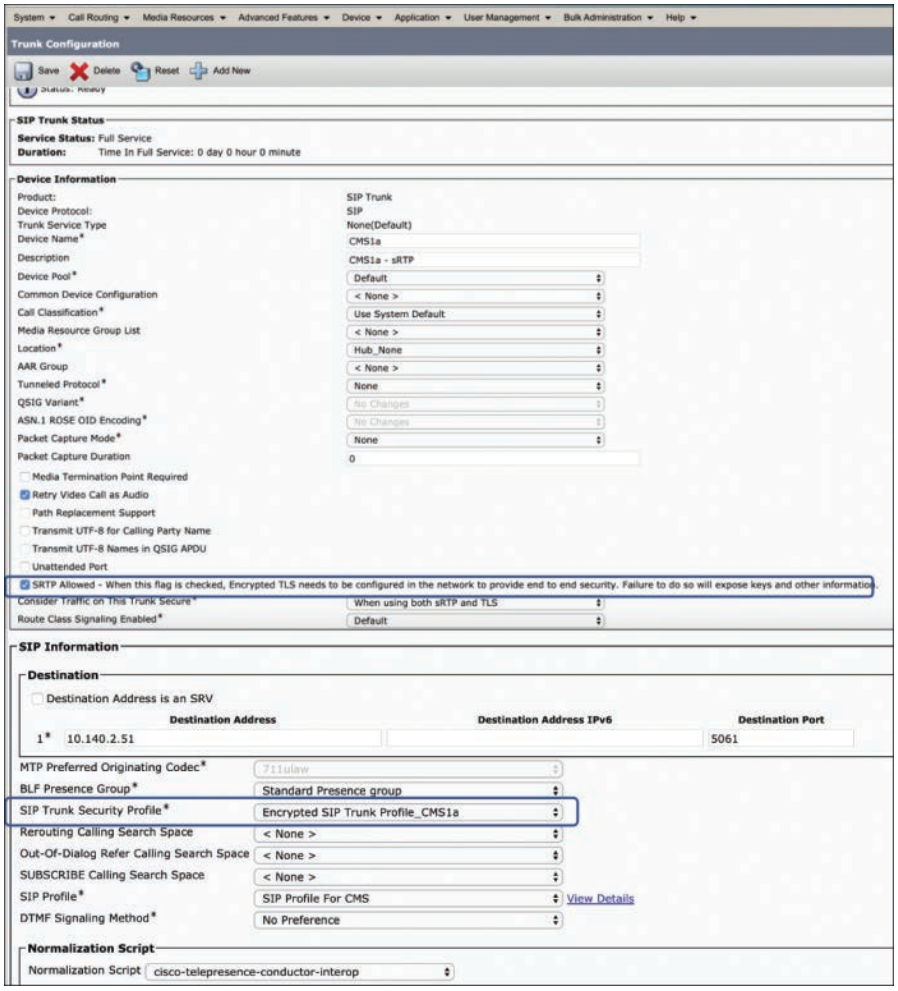


Figure 10-19 *Specifying Encrypted Media and an SIP Trunk Security Profile Within Unified CM*

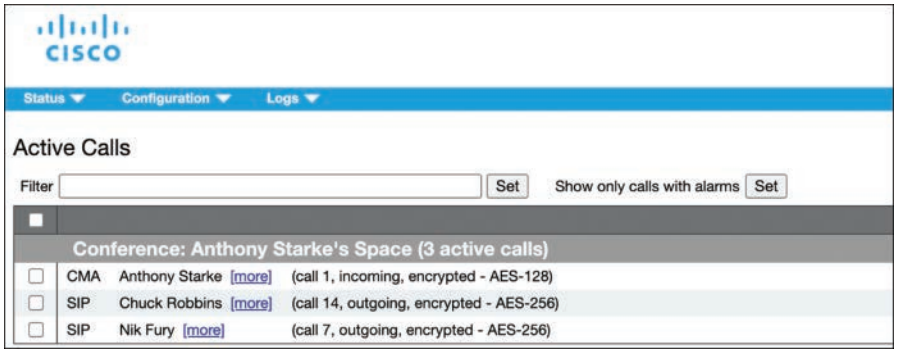


Figure 10-20 *Monitoring Active Calls on CMS Servers*

https://t.me/learningnets
@cisco_in_persian_channel

Securing CMS Spaces

A final layer of security can be added to restrict unauthorized users from accessing meetings with strong PINs or passwords and to help organizations comply with security policies. Any scheduled meeting or permanent meeting can have a PIN set so that all participants are challenged to enter the PIN before being allowed to connect. Starting in CMS 3.0 software, administrators can leverage the /dialInSecurityProfiles API object to require minimum passcode lengths. Once created, this parameter can be assigned to the /system/profiles object to enforce the policy across the entire CMS platform. Additional options for assigning this parameter include the tenant level (/tenants), for coSpaces (/coSpaces/[coSpace ID]), or as an access method to a coSpace (/coSpace/access-Methods). Figure 10-21 shows how to configure the dialInSecurityProfiles object.

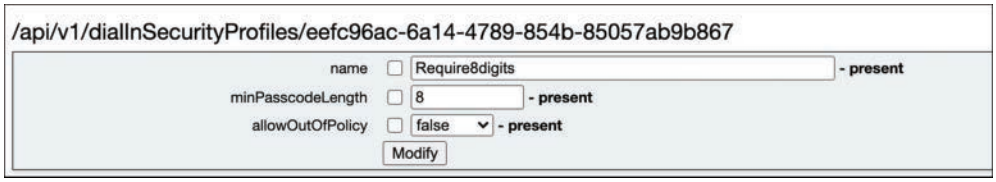


Figure 10-21 *Creating a dialInSecurityProfile to Require Strong Passcodes for CMS Meetings*

Management and Visibility

Up to this point, we have spent a significant amount of time with the CMS CLI to show how administrators can work with X.509 certificates to set up secure connections. Aspects of security that cannot be overlooked include management tasks, meeting utilization, and visibility. One aspect of management and compliance involves gathering log information. The audit log can be downloaded from CMS via SFTP or exported to external servers. When downloaded, the audit log can provide information related to the traditional "accounting" aspects of configuration management. Additional information that the audit log can provide includes

- Configuration changes and significant low-level events and the username of the administrator who performed the change
- Participants joining/leaving events
- CDR receivers being added, modified, or deleted

Exporting logs from CMS is possible by configuring syslog and auditing. This option is an effective way for increasing visibility and for proactively detecting incidents. By using a management system such as a Splunk, which can correlate messages and generate alarms, administrators can avoid the tedious nature of manually sifting through logs in an attempt to detect security issues.

https://t.me/learningnets

@cisco_in_persian_channel

As previously discussed, CMS provides separation of duties with RBAC. Syslog servers can be configured by any user with the admin role, but audit configuration can only be configured by a user with the audit role. By default, syslog traffic uses TCP port 514, but secure syslog traffic with TLS can be configured to use TCP port 6514. Example 10-10 provides an audit configuration, and Example 10-11 provides a syslog configuration.

Example 10-10 *Audit Configuration Performed by User with the Audit Role*

```
cms1a> syslog audit add tls:192.168.1.100 6514
cms1a> audit http enable
cms1a> audit firewall enable
```

Example 10-11 *Syslog Configuration Performed by User with the Admin Role*

```
cms1a> syslog server add tls:192.168.1.100 6514
cms1a> syslog enable
cms1a> syslog
Sending log messages to the following servers:
  192.168.1.100                  6514

Sending audit logs to the following servers:
  192.168.1.100                  6514
cms1a>
```

Simple Network Management Protocols can also be configured within CMS to support polling and offloading information to a network management server using SNMP traps. Although CMS supports SNMP versions 1, 2, and 3, versions 1 and 2 do not support authentication, so anyone who is aware of the community string would be able to query CMS to collect information. For this reason, we cover use of SNMPv3. When you're configuring SNMPv3, a username is required to provide authentication before being allowed to poll CMS. When you're adding SNMP users, CMS supports weak algorithms such as MD5 and DES. Earlier, when we discussed hardening the CMS OS, we discussed how use of FIPS mode could help prevent use of weak algorithms. This feature is beneficial with SNMPv3 because, by default, it blocks administrators from inadvertently trying to configure these weak algorithms that are more susceptible to man-in-the-middle attacks. The following example shows how CMS provides a safeguard against these weak algorithms:

```
cms1a> snmp user add md5user C1sc0123 MD5 DES
MD5 authentication is not allowed in FIPS mode
cms1a>
cms1a> snmp user add md5user C1sc0123 SHA DES
DES encryption is not allowed in FIPS mode
cms1a>
```

https://t.me/learningnets

@cisco_in_persian_channel

The following example provides a secure SNMPv3 configuration:

```
cms1a> snmp user add cmspriv C1sc0123 SHA AES
cms1a> snmp community add R34d0n1y 192.168.1.100/24
cms1a> snmp trap enable NMS R34d0n1y
cms1a> snmp
SNMP is enabled

The following SNMPv1 communities are defined:
  R34d0n1y        192.168.1.100

The following SNMPv3 users are defined:
  cmspriv

Sending traps to NMS with community string R34d0n1y
cms1a>
```

Now that we've explored use of the CLI to configure CMS for monitoring logs and alarms, we can explore how to monitor previous and active meetings. Although the web admin GUI can be used to provide basic monitoring of current calls, it doesn't provide the ability to collect call detail records (CDRs) from previous meetings that can be used to identify CMS utilization. Instead, CDR receivers can be configured for external services such as Cisco Meeting Management (CMM) server.

CMM, which is provided by Cisco as a free management tool, enables organizations to collect logs and to monitor and manage CMS meetings. As shown in Figure 10-22, CMM allows administrators to provide central management of meetings, which includes the ability for muting participants or for providing meeting security by preventing unauthorized participants from joining current CMS meetings by leveraging the lock icon. Starting with CMS 3.0 software, CMM 3.0 (or later) is also required for licensing purposes. Without CMM, organizations have limited meeting and security functionality, such as not being able to establish encrypted calls to/from SIP endpoints.

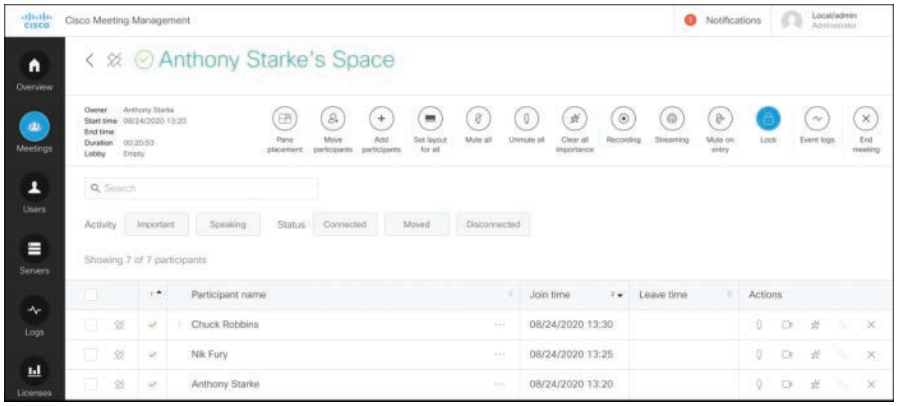


Figure 10-22 *Leveraging Cisco Meeting Manager to Manage and Secure CMS Meetings*

https://t.me/learningnets

@cisco_in_persian_channel

CMM supports different authentication options, such as integration to directory servers via LDAP/LDAPS or use of local username/password combinations. When organizations use local CMM accounts, organizational password security policies can be enforced (e.g., minimum password length, restricted password reuse). CMM also includes a built-in passphrase generator to help create strong passwords from a combination of words from a dictionary. For more specific management and security features, CMS also supports use of ecosystem partner products such as VQ Conference Manager and/or Vyopta.

An example of a specific integration that provides increased security is two-factor authentication or SAML 2.0 single sign-on (SSO) into management tools. VQ Conference Manager is currently the only CMS management application that provides strong authentication using two-factor authentication. VQ Conference Manager supports SAML 2.0-based SSO by integrating with solutions such as Microsoft Active Directory Federated Services (ADFS), Okta, Duo, and OneLogin. VQ Conference Manager can provide unique security capabilities for CMS meetings. As an example, instead of using a personal meeting space to collaborate, users can use VQ to create a new CMS space that can be used only once with their own PIN/passcode. When the meeting is over, the CMS space (and any associated data) is removed. VQ provides organizations with a way to comply with authentication policies when joining CMS meetings. With VQ Conference Manager, administrators can require a PIN/passcode for meetings and define the minimum length and complexity of a PIN/passcode for a CMS space. For management of meetings and overall administration, VQ also provides customizable dashboards for organizations to analyze historical data and by collecting CDRs. Data that is collected within VQ can be exported and shared with management or integrated into other business intelligence/analytics tools. It can also provide email-based alerting.

Vyopta provides historical analytics and real-time monitoring for UC solutions. Vyopta's goal is to help meet or exceed industry standards for the security, availability, and confidentiality with analytics and monitoring platforms. From a compliance perspective, Vyopta is Service Organization Control (SOC) 2 compliant. Reporting capabilities have been developed to help organizations comply with principles that govern operational and organizational controls, including monitoring for poor connections that may lead to a poor user experience (e.g., choppy voice/video, packet loss), along with analytics to track activity and utilization of conference rooms.

Last but not least, management tasks include disaster recovery services and scenarios. This includes how organizations are able to back up and restore the CMS database. Two CLI commands allow you to back up the CMS system configuration in a file. Similar to CSRs, backups are downloaded from CMS server using SFTP. Snapshots are used to back up the CMS database. When you leverage **backup snapshot [name of backup]**, CMS creates a full system snapshot of the CMS configuration that includes information such as configurations, IP addresses, passwords, and certificates. When you need to search for files, you can identify backup names with the .bak extension. To restore CMS backups, you use the **backup rollback [name of backup]** command. The **rollback** command overwrites the existing configuration and also licensing files, certificates, and private keys on the system. Cisco recommends using this command with caution because it overwrites any configuration, certificates, and private keys on a system and is immediately followed by a reboot.

https://t.me/learningnets

@cisco_in_persian_channel

The following example shows the **backup snapshot** command:

```
cms1a> backup snapshot cms1a
cms1a.bak ready for download
cms1a>
```

Because manually performing tasks such as backing up the CMS database or importing new users via LDAPS may not always be practical, you can use automated scripts on a regular basis. As previously discussed, one of the benefits of having an API is that programmers who are familiar with programming languages such as Powershell and Python can automate menial tasks. This topic is beyond the scope for this book, but you can find more information by reviewing public repositories such as GitHub that contain scripts and software code that can be used to automate tasks within CMS. GitHub can be located by browsing to https://github.com.

Summary

Because CMS is a rich media conferencing system where valuable information is shared, it can be a high-value target for attackers. Whether an attacker's goal is to eavesdrop on a meeting or to join a meeting and be a nuisance, CMS provides security controls that align with the most stringent government and industry standards, such as two-factor authentication into the CLI or web admin interface and AES-256 encryption for ensuring confidentiality and privacy of media streams.

To align to defense-in-depth principles, organizations have multiple options for configuring CMS for different layers of security. The layers that you should be more familiar with now are within the OS, through integrating with network infrastructure services such as DNS and NTP and also PKI to obtain X.509 certificates. CMS configuration options include the CLI, web admin interface, API, and programming languages that communicate with Cisco Meeting Server's API. To address the threat landscape, Cisco recommends use of TLS encryption whenever possible to provide operational security. Leveraging techniques that provide both authentication and encryption organizations have the best chance at securing internal and external connections.

The last area in which organizations can increase their security is through proactive management and monitoring. Organizations are able to manage CMS meetings in a number of ways. Using Cisco-based tools such as CMM, organizations can securely manage conferences (and participants) while also collecting CDRs to achieve visibility to historical data. Cisco supports additional management tools that are part of Cisco's ecosystem for managing CMS environments. These third-party tools include the ability to provide two-factor authentication, SSO, scheduled meetings, historical analysis, monitoring, alerting, and integration with additional systems that can be used for business intelligence or for responding quickly to issues.

https://t.me/learningnets

@cisco_in_persian_channel

Additional Resources

www.cisco.com/go/ucsrnd

www.vqcomms.com/

www.vyopta.com/

https://t.me/learningnets

@cisco_in_persian_channel

Chapter 11

Securing the Edge

Business Requirements for the Collaboration Edge

Recent studies highlight a surge in working from home (WFH), which is also called telework. Studies show that more employers are offering telecommuting options for their workforce. Additionally, studies from Globalworkplaceanalytics.com show that the average worker would consider accepting less pay for this option because of the increased quality of life that is associated with working from home. Cisco's Unified Communications (UC) architecture has been adapted to support these types of new business models by helping provide access to UC services while workers are either inside or outside the traditional workspace. In turn, employee satisfaction is increasing because many organizations are developing work-from-home policies, and organizations have seen productivity increases due to reduction in sick days and break time.

In practice, management may still hesitate to allow employees to work from home due to concerns about lack of communication or work ethic. However, if the work environment is a good fit, implementation of work-from-home solutions, such as Cisco's Collaboration Edge, can benefit both the company and employees.

Security Considerations

While telecommuting is an attractive work option for organizations to offer their employees, the question becomes, What types of services do employees need to be as productive from home as in the office? Assuming that services such as telephony, messaging, and voicemail (UC services) are needed, decision makers should ask themselves three questions:

- What is the most practical solution for providing remote access to remote workers?
- Will the solution that is provided meet all of the requirements dictated in the organizational security policy?
- Can the solution provide a quality experience so that employee productivity doesn't suffer?

https://t.me/learningnets

@cisco_in_persian_channel

Solving ACME's Telework Challenges

In response to ACME's urgent remote work initiative, Anthony has been tasked with implementing solutions that help teleworkers connect to their on-premises UC environment so that they can continue to work remotely without impacting the user experience. ACME's COO, Dr. Nicholas Fury, also believes that increasing the teleworkers' capabilities will help increase the productivity levels of ACME employees. ACME already has a telework policy that allows certain employees to telework, but employees who are teleworking do not have the same user experience or access to the same UC services when teleworking as they do when they are in the office.

One of the challenges that Anthony believes he will run into is getting approval from ACME's Information Assurance (IA) office to deploy UC services within ACME's DMZ environment. One of the challenges that Anthony sees with IA is that the notion of extending UC services beyond the perimeter may seem to pose a security risk. He has been told that the best path to get IA approval for deploying something in the DMZ is to follow the IA lead and let them think that they are making the decisions to support the teleworkers.

Anthony realizes that he needs to increase his understanding of the types of attacks that can be launched from attackers across the Internet and be prepared to recommend ways to mitigate risk to get IA on his side. The approach that Anthony is planning is to educate IA about the security features that are built into the architecture and how encryption is used with existing protocols. As Anthony works to convince IA, he also has to minimize any costs associated with increasing teleworker capabilities.

One of the benefits that Anthony sees with Cisco's Collaboration Edge architecture is that there doesn't appear to be a requirement for VPN hardware, so it seems to streamline implementation timelines with minimal cost. The one exception seems to be the cost of obtaining certificates from a public certificate authority. He is not sure where to start, so he is trying to decide whether he should ask the server team for help. Another challenge is that nobody from ACME has experience with the core components in Cisco's Collaboration Edge architecture. Anthony is always up for a challenge, but he has some concerns about how long it will take him to install and configure Expressway in a secure fashion.

Questions that you should ask:

1. What features do teleworkers need to be provided to have a quality experience?
2. What is the IA policy for deploying applications inside a DMZ?
3. What security controls exist within the DMZ?

Cisco's Collaboration Edge Architecture

Cisco's Collaboration Edge is an architecture that organizations can leverage to consume IP-based services from service providers and also provide secure and remote access to Unified Communication services to end users. The Collaboration Edge architecture builds on the traditional core UC applications that are contained within the corporate

https://t.me/learningnets

@cisco_in_persian_channel

environment and extends the traditional on-premises UC infrastructure beyond the corporate perimeter. Two key UC components of the Collaboration Edge architecture are Cisco Unified Border Element (CUBE) and Cisco Expressway (Edge and Core).

In total, Cisco's Collaboration Edge architecture provides the following four options to help organizations obtain or provide services beyond the corporate perimeter, while providing customers with the most flexibility:

- IP-based access to the PSTN
- VPN-based access to UC services
- VPN-less access to UC services
- Business-to-business communications

A high-level view of the architectural components in Cisco's Collaboration Edge architecture is shown in Figure 11-1.

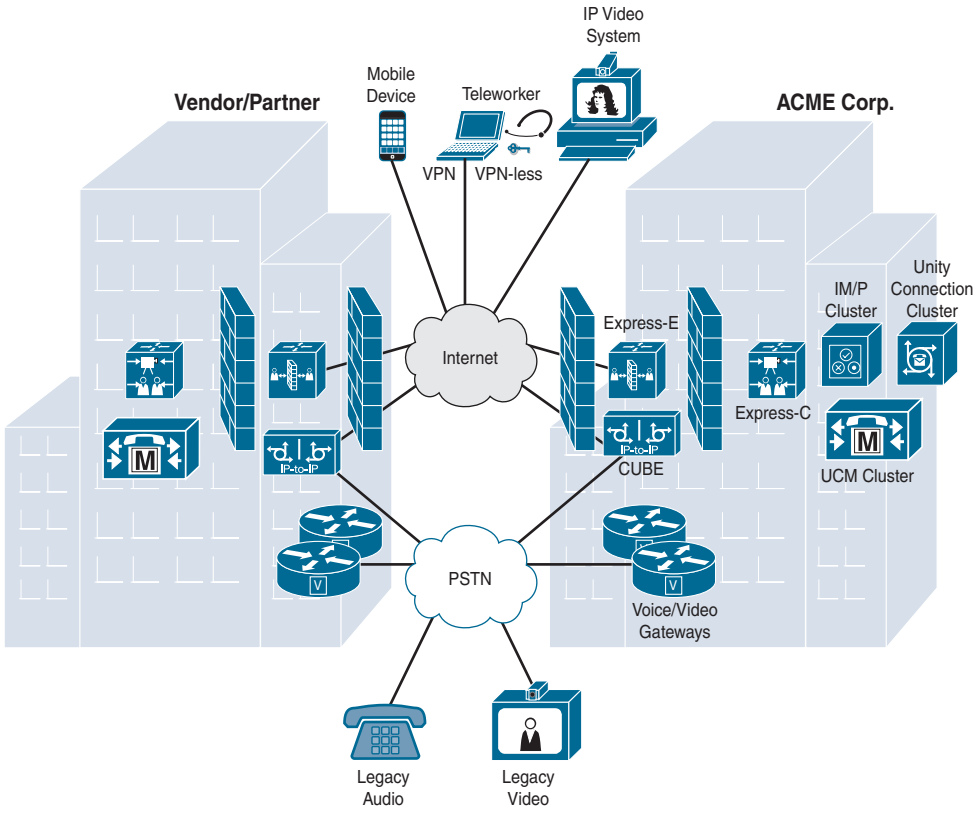


Figure 11-1 *High-Level View of Cisco's Collaboration Edge Architecture for ACME*

https://t.me/learningnets

@cisco_in_persian_channel

IP-Based PSTN Access

As organizations migrate away from time-division multiplexing (TDM)-based connections to the public switched telephone network (PSTN) and start connecting via SIP to IP PSTN service providers, a session border controller, such as Cisco Unified Border Element (CUBE), is recommended. CUBE is essentially a feature that is enabled on a Cisco routing platform running IOS/IOS-XE software and enabled with the Unified Communications technology package. Once enabled, it can provide protection against attacks from layer 3 up to layer 7 of the OSI model to provide application layer security for voice and video traffic. For this reason, CUBE can help organizations maintain compliance with industry security standards and facilitate integration to adjacent services such as Unified Communications Manager inside the network and a service provider located outside the network.

In many cases, CUBE is also used to provide interoperability with different systems. As an example, it allows you to manipulate SIP parameters or to provide direct route integration from a Microsoft Phone System environment to the PSTN. CUBE is also able to provide management and visibility over SIP trunk connections so that organizations can easily troubleshoot connections and monitor use of resources as calls are routed to the PSTN.

CUBE can provide the following features to simplify connectivity and security when connecting to the IP PSTN:

- **Interworking:** Provides normalization of SIP signaling; Codec selection (e.g., Opus, G.711, G.729, G.722), DTMF interworking (e.g., RFC 2833, Key Press Markup Language [KPML]), and so on.
- **Demarcation:** Terminates and re-originates SIP signaling between internal and external services. This allows IP phones that have non-publicly routable addresses (e.g., using private RFC 1918 based IP addresses) to call out to the IP PSTN. This capability also helps prevent network reconnaissance attacks by hiding an organization's UC topology from network scans, thus hiding the internal network from the rest of the world.
- **Session Control:** Allows organizations to terminate SIP sessions and route IP calls securely.
- **Security:** Prevents telephone denial-of-service (TDoS) and toll fraud attacks; enforces authentication and encryption of signaling and media

The two most common models for deploying CUBE are centralized and distributed deployment models. Centralized deployments help simplify connectivity to the IP PSTN and help provide cost-effectiveness because they minimize how many CUBEs must be deployed across an organization.

Distributed connectivity to the IP PSTN is also an effective model but can be more costly to deploy because of the additional hardware and management costs. Although we do not intend to spend much time discussing software licensing models, Cisco's new Smart

https://t.me/learningnets

@cisco_in_persian_channel

Licensing model allows organizations to flatten out costs for CUBE software and licensing. This model offers customers flexibility because it provides a mechanism to route outbound calls locally in the event that users are unable to traverse the WAN due to a loss of service or the WAN is not effective due to performance issues. In the event that neither the centralized nor distributed model is a great fit for a customer, a hybrid connectivity model can be used. The diagram in Figure 11-2 depicts how CUBE provides secure connectivity to the IP PSTN and how it can be used to prevent against various UC attacks.

Note Customers should make sure that they can reach 911-based emergency services through centralized SIP Trunk services, especially during times of congestion or during an emergency WAN outage. See Chapter 2, "Physical Security and Life Safety," for more information on this topic.

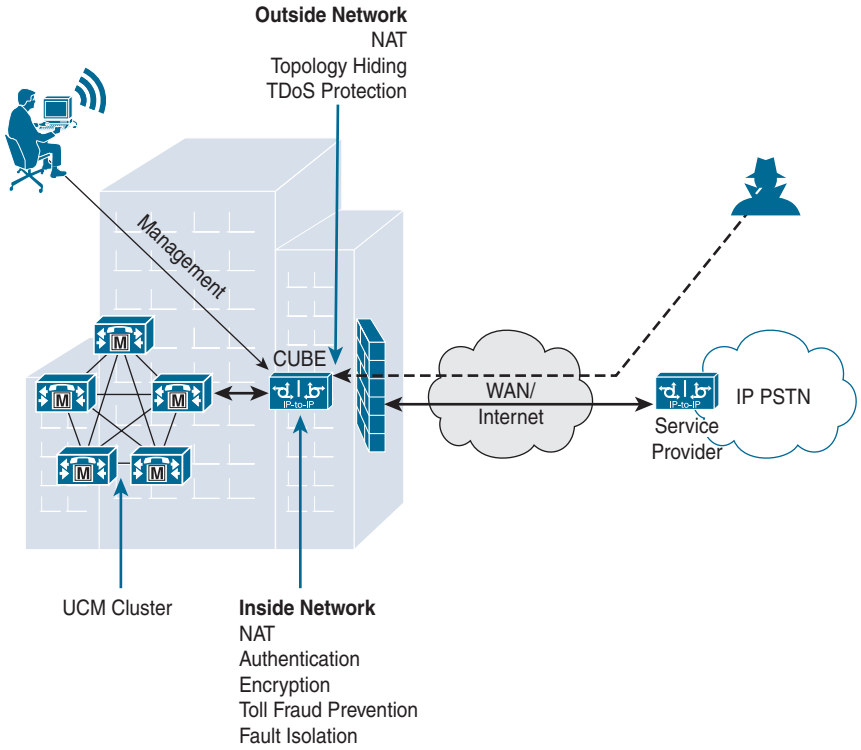


Figure 11-2 *CUBE Providing Session Control, Border Protection, and Topology Hiding*

CUBE can be deployed on either a virtual machine environment (vCUBE) or on traditional Cisco UC-enabled routing platforms running IOS or IOS-XE. Because CUBE is software that is enabled on a router, it can be easily enabled on an existing platform, such

https://t.me/learningnets
@cisco_in_persian_channel

as a router that is providing WAN connectivity or already acting as a PRI gateway to the PSTN. When an existing platform is reused, it can provide organizations with a cost-efficient method of connecting to IP PSTN services.

Deploying CUBE

Previously, we introduced the capabilities and benefits of the Cisco Unified Border Element. In this section we focus on the process to enable and then apply relevant security features to reduce the attack surface against an organization acquiring IP-based PSTN connectivity by deploying CUBE at the edge of the network. Starting with the network layer, Cisco recommends that organizations use security controls such as access control lists (ACLs) to permit traffic from trusted IP addresses (e.g., UCM, a service provider's SBC) before traffic reaches the application layer. To do this, you can create a simple access control list that specifies a specific source IP address (IP address of IP PSTN provider) that is destined to a specific destination IP address, such as the external interface of CUBE, as in this example:

```
cube(config)# access-list 100 remark PERMIT TRAFFIC FROM IP PSTN
cube(config)# access-list 100 permit ip host [IP ADDRESS OF IP PSTN
PROVIDER] any
```

After this, you must apply the access control list to an external interface of the device in the direction that IP traffic is headed:

```
cube(config)# int gig0/0
cube(config-if)# ip access-group 100 in
```

Network Based Application Recognition (NBAR) is another feature that can be enabled to gain visibility of traffic flows across the network and specifically in or out of the network through CUBE. Cisco's NBAR provides classification of IP packets. Thus, UC administrators can have an improved situational awareness by monitoring traffic flows passively from the command-line interface (CLI) or through network management software, such as LiveAction's LiveNX software, which provides active monitoring and analytics on historical performance. The NBAR feature is not necessarily unique to UC but can provide benefits to UC administrators when it is enabled on CUBE at the edge of the network. One specific benefit of using NBAR to inspect traffic flows is proactive detection of anomalies, such as an increase in traffic that may lead to a DoS/TDoS condition, so that the organization can take steps to limit the damage of this kind of attack and reduce the organization's attack surface.

NBAR can also be used to detect traffic that is entering and exiting CUBE and, if needed, apply preferential treatment to mission-critical UC traffic if bandwidth congestion occurs at the edge. Because NBAR maintains per-protocol statistics on an individual interface, it can be helpful to detect the total number of input packets that are entering into CUBE so that the organization can plan to increase scalability on an as-needed basis.

https://t.me/learningnets

@cisco_in_persian_channel

To enable NBAR, you should enter the following commands from a router that is running IOS/IOS-XE on a per interface basis.

```
cube(config)# int gig0/0
cube(config-if)# ip nbar protocol-discovery ipv4
```

Starting with IOS-XE 3.14S, NBAR provides two levels of application recognition: coarse-grain and fine-grain. By default, NBAR operates in coarse-grain mode because it provides better performance. Cisco recommends use of fine-grain mode when detailed layer 7 metrics need to be extracted to support critical applications or to leverage NBAR's full application recognition capabilities, such as UC traffic. To enable fine-grain mode, you must boot the IOS-XE router to appxk9 license level. After this is done, you can use the following commands to recognize normal and SIP-TLS signaling:

```
cube(config)# ip nbar classification granularity fine-grain protocol sip
cube(config)# ip nbar classification granularity fine-grain protocol
sip-tls
```

Figure 11-3 shows an example of the UC applications that NBAR is able to monitor and provides insights to and from the CLI.

```
cube#sh ip nbar protocol-discovery

 GigabitEthernet0/0/0

 Last clearing of "show ip nbar protocol-discovery" counters 00:00:49

                              Input                  Output
                              -----                  ------
 Protocol                     Packet Count           Packet Count
                              Byte Count             Byte Count
                              5min Bit Rate (bps)    5min Bit Rate (bps)
                              5min Max Bit Rate (bps) 5min Max Bit Rate (bps)
 ---------------------------- ---------------------- ----------------------
 cisco-collab-video           3648                   3741
                              2024185                2070234
                              50000                  51000
                              50000                  51000
 rtp-video                    3741                   3648
                              2070234                2024185
                              51000                  50000
                              51000                  50000
 cisco-collab-audio           5000                   5000
                              1070000                1070000
                              32000                  32000
                              32000                  32000
 ssh                          130                    90
                              10412                  16972
                              0                      0
                              1000                   2000
 cisco-collab-control         33                     34
                              4446                   5215
                              1000                   1000
                              1000                   1000
 rtcp                         9                      9
                              1098                   1098
                              0                      0
                              0                      0
 ntp                          1                      1
                              90                     90
                              0                      0
                              0                      0
 Total                        12562                  12523
                              5180465                5187794
                              134000                 134000
                              135000                 136000
cube#
```

Figure 11-3 *The Visibility Provided by NBAR Once Enabled on CUBE*

https://t.me/learningnets

@cisco_in_persian_channel

For further information about NBAR, such as creating a policy on the command line to provide preferential treatment for UC, review the NBAR configuration guide for the version of software running on CUBE.

Up to this point, we have not really addressed the voice inspection features contained within the CUBE application. A common theme across this chapter and the rest of this book is deploying security in layers. Many organizations do not deploy security this way. By understanding which IOS and IOS-XE features are available and how they work, an organization can leverage traditional software features that work in unison with CUBE features to proactively detect and prevent attacks at very little cost. This also allows organizations to take a layered approach that is able to increase an organization's visibility and security in a simple but powerful way. The next several sections focus on enabling CUBE and preventing specific VoIP attacks. To start using CUBE, an organization needs to have the UCk9 technology package (or its equivalent) loaded on the IOS/IOS-XE device. This package allows the UC administrator to have access to the necessary commands on the router's CLI to enable UC features and the relevant UC security features designed for a session border controller such as CUBE. The syntax required from the CLI is as follows:

```
CUBE# config t
Enter configuration commands, one per line. End with CNTL/Z.
CUBE(config)# voice service voip
CUBE(config-voi-serv)# mode border-element license capacity
[cube licenses]
CUBE(config-voi-serv)# allow-connections sip to sip
CUBE(config-voi-serv)# end
```

Note Starting in IOS-XE 17.2.1r, the license capacity option has been deprecated because Cisco CUBE uses dynamically assigned licenses.

Toll Fraud Protection

Toll fraud has been and continues to be a type of attack that negatively impacts organizations. Recently, experts have associated toll fraud attacks with organized crime rings. In 2018, the Communications Fraud Control Association (CFCA) estimated that toll fraud amounted to approximately $28 billion in damages to organizations. Toll fraud attacks present themselves in many different ways. A somewhat innocent act of toll fraud is a scenario in which someone such as a family member places a phone call into a work location. The family member has a discussion but then, instead of hanging up the phone when the conversation is over, transfers the call to a different family member who lives in a different, long-distance area code. Unaware of the impact to the organization, the office worker is costing the organization long-distance toll charges. This type of incident could easily be prevented.

A more overt form of toll fraud occurs when a hacker is able to gain access to an organization's UC application and/or dial plan contained within Unified CM. Once a hacker

https://t.me/learningnets

@cisco_in_persian_channel

has access to the system (or makes calls across the system), the perpetrator can achieve financial gain at the expense of the organization in the form of toll charges. As customers build out a strategy to prevent toll fraud, CUBE is a logical place in the network to provide security controls because it provides both inbound and outbound call routing at the edge of the network.

One of the simplest ways to protect against which systems are allowed to place inbound and outbound calls is to specify which IP addresses CUBE trusts by using the **ip address trusted list** command followed by "trusted" IPv4 or IPv6 addresses. When this feature is configured, it adds an internal CUBE application named TOLLFRAUD_APP to the call control stack. As its name implies, the application is intended to prevent toll fraud by checking the source IP address used during call setup before routing the call. In this approach, CUBE authenticates the source IP address and verifies whether a source IP address is a trusted VoIP source. If the source is trusted, the call proceeds. If not, the call is rejected. The necessary syntax and a brief explanation of the commands are shown in Example 11-1.

Example 11-1 *Preventing TDoS and Toll Fraud Attacks Through CUBE*

```
cube(config)# voice service voip
cube(conf-voi-serv)# ip address trusted authenticate (Enables IP address
  authentication on incoming H.323 or Session Initiation Protocol (SIP) trunk
  calls for toll fraud prevention support.)
The command ENABLES the ip address authentication on incoming
H.323 or SIP trunk calls for toll fraud prevention supports.
Continue? [confirm]
cube(conf-voi-serv)# ip address trusted list
cube(cfg-iptrust-list)# ipv4 10.130.2.2 (Specifies list of trusted IP addresses
  such as UCM)
cube(conf-voi-serv)# sip
cube(conf-serv-sip)# silent-discard untrusted (enabled by default. This
  feature discards SIP requests from untrusted sources on an incoming SIP
  trunk to prevent against TDoS attacks)
cube(conf-serv-sip)# end
```

CUBE-Based TDoS Protection

In Chapter 1, "The Importance of Practical UC Security," we introduced the threat of a telephone denial-of-service attack and also provided an example illustrating how social media could be used to initiate a TDoS. As calls are flooding into an organization's PSTN gateways, little can be done to stop this type of attack. Instead, organizations should take precautions to limit the disruption so that they can perform as many normal duties as possible. An approach to limit the disruption of a TDoS attack using CUBE is to use Call Admission Control (CAC). When this feature is enabled, CUBE monitors the call arrival rate over a moving window of time. When call levels "spike" above the expected threshold rates, they are rejected. You can configure this feature globally or on a per dial-peer

https://t.me/learningnets

@cisco_in_persian_channel

level. When it is configured globally, you can easily configure call spikes with the following commands:

```
cube# conf t
Enter configuration commands, one per line. End with CNTL/Z.
cube(config)# call spike 50 steps 5 size 100
cube(config)# call treatment on
cube(config)# end
cube#
```

Session Control and Protection

To route a VoIP call to the IP PSTN, dial-peers are necessary. A dial-peer within CUBE is essentially a static route that maps a dialed phone number to a destination IP address. A call leg is a logical connection between a CUBE and a VoIP endpoint or between two different network devices, such as two different SBCs. Cisco currently recommends use of two dial-peers for each call that is established. One dial-peer is used for establishing a connection on the inbound call-leg, and another dial-peer is used for the outbound call-leg. Dial-peers are matched to call legs based on configured parameters, such as the caller ID (ANI) or the dialed number (DNIS). An outbound dial-peer is meant to establish call legs for calls that originate from CUBE to an external entity, such as the UCM (LAN side) or the IP PSTN's IP address (WAN side). Example 11-2 depicts a sample dial-peer configuration.

Example 11-2 *Sample Dial-Peer Configuration*

```
dial-peer voice 100 voip
 description OUTBOUND DIAL-PEER TO UNIFIED CM
 destination-pattern 9195551...$              (matches dialed number)
 session protocol sipv2                       (specifies SIP protocol)
 session target ipv4:10.130.2.2:5061          (destination IP address over SIP-TLS)
 voice-class sip bind control source-interface GigabitEthernet0/1 (specifies
  interface to bind signaling to)
 voice-class sip bind media source-interface GigabitEthernet0/1   (specifies
  interface to bind media to)


dial-peer voice 200 voip
 description INBOUND DIAL-PEER FROM PSTN
 session protocol sipv2                        (specifies SIP protocol)
 incoming called-number 9195551...$            (matches Dialed number)
 voice-class sip bind control source-interface GigabitEthernet0/0  (specifies
  interface to bind signaling to)
 voice-class sip bind media source-interface GigabitEthernet0/0    (specifies
  interface to bind media to)
```

https://t.me/learningnets

@cisco_in_persian_channel

Now that we've introduced the basic purpose of dial-peers, we need to further discuss the implications of improperly configured dial-peers, especially as they pertain to call loops and toll fraud. Dial-peers should be configured to ensure that all calls received from a particular destination can be routed only to destinations where you expect that traffic to go. For example, inbound calls from the PSTN should always be routed to an internal call agent such as UCM and should never route back out toward the PSTN without first being routed to UCM for processing. When dial-peers are not explicitly configured to provide this degree of specificity, it is possible for unexpected call flows to occur. These unexpected calls flows may involve toll-fraud attacks, so it is important to protect against them.

A general configuration that could cause unexpected call flows occurs when organizations use wildcards on dial-peer statements. When a call comes into CUBE and matches a wildcarded dial-string (e.g., incoming called-number 9T), the initial call leg is established on CUBE. By configuring (and matching) a wildcarded destination pattern (e.g., destination-pattern .T), a VoIP call is routed out to the IP PSTN. If your organization's configuration resembles this type of wildcarded configuration, an attacker may able to exploit poorly configured dial-peers to hairpin calls out to the IP PSTN. Fortunately, Cisco CUBE has features to mitigate this type of threat by using the **outbound dial-peer group** feature. When this feature is configured, it provides a mechanism for administrators to provide more specific and predictable configurations on CUBE so that calls cannot be hairpinned back out to the PSTN. The next section provides more detail about this feature.

Introduced in IOS 15.4(1)T and IOS XE 3.11S, the dial-peer group feature allows organizations to define a specific list of prioritized dial-peers to use for outbound calls while matching an inbound dial-peer. In doing so, it simplifies administration of dial-peers to support expected call flows. The dial-peer group feature also allows administrators to reduce the number of outbound dial-peer statements that are needed to provide redundancy. To provide more detail, we have included an example of the syntax used for outbound dial-peer grouping and also a visual representation of how the topics for using explicit dial-strings and dial-peer groups work together (see Figure 11-4).

Step 1. Match the inbound dial-peer.

```
dial-peer voice 1 voip
description Inbound dial-peer
incoming called-number 9195551…$ (specific range of directory
numbers on corporate network)
destination dpg 1000
```

Step 2. Use the dial-peer group [**dpg 1000**].

Step 3. Select the outbound dial-peer (see Example 11-3).

https://t.me/learningnets

@cisco_in_persian_channel

Example 11-3 *Outbound Dial-Peer*

```
voice class dpg 10000
description Outbound dial-peer group for Unified CM
dial-peer 100 preference 1
dial-peer 101 preference 2

dial-peer voice 100 voip
description Unified CM subscriber 1
destination-pattern 9195551…$
session protocol sipv2
session target ipv4:10.1.1.1
!
dial-peer voice 101 voip
description Unified CM subscriber 2
destination-pattern 9195551…$
session protocol sipv2
session target ipv4:10.1.1.2
```

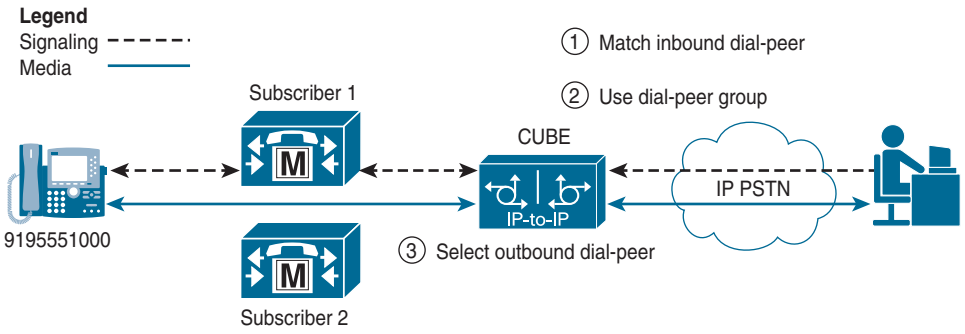


Figure 11-4 *Use of Dial-Peer Groups to Help Ensure Proper Routing of VoIP and Video Calls*

For more information about dial-peer matching and/or dial-peer grouping, refer to the IOS/IOS-XE Configuration Guide for these topics.

By this point, we have shown several options that an organization has to deploy security in layers with CUBE. As previously discussed, when layers of security are deployed, a perpetrator would need to bypass multiple layers of security checks prior to reaching CUBE's call routing logic in order to be able to attack an organization's CUBE at the edge of the network. As discussed, these layers of security can include an external firewall to filter traffic, use of ACLs within the router platform supporting CUBE, and then CUBE-based application security (e.g., IP address trust lists) before using dial-peers and dial-peer groups (DPGs) to avoid call loops and unauthorized attempts at toll fraud.

https://t.me/learningnets

@cisco_in_persian_channel

In some cases, organizations may be required to provide secure VoIP calling between different corporate locations that are already using UCM. As shown in Figure 11-5, when different sites require secure VoIP calling, CUBE can be configured to provide session controls between different corporate locations and also ensure that TLS-encrypted signaling and media (SRTP) are used to provide end-to-end confidentiality. By leveraging Transport Layer Security, organizations can ensure that VoIP calls are not intercepted and potentially used for a playback attack or eavesdropping.

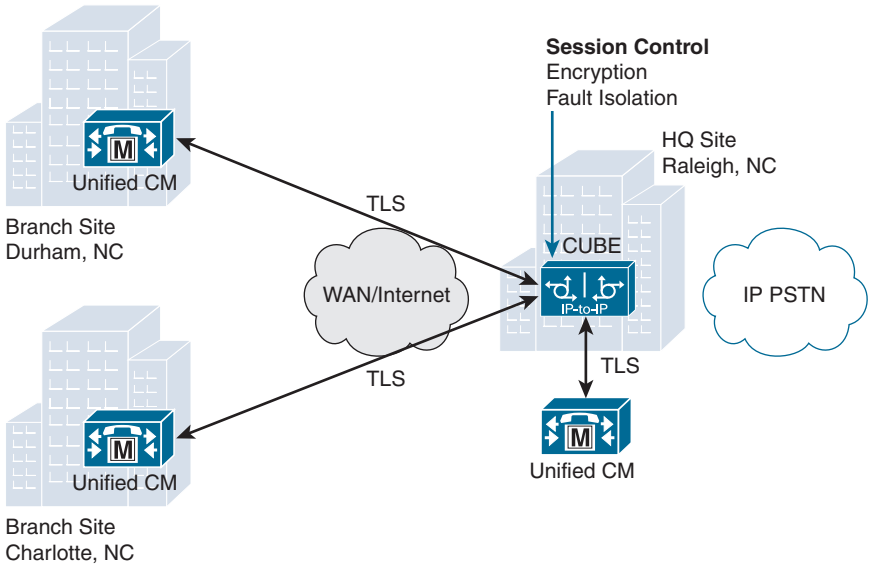


Figure 11-5 *Using CUBE to Provide Session Control and Encryption Between UCM Locations*

With TLS configured on CUBE, CUBE encrypts SIP signaling messages. Media streams are secured by leveraging Secure Real-time Protocol (SRTP) on dial-peers. To configure TLS, you once again work with certificates, but this time it is between UCM and CUBE. Cisco provides the option of using either self-signed or CA-signed certificates within CUBE. Because many organizations prefer CA-signed certificates, our examples include the necessary configurations for this approach.

Enabling CUBE for TLS Connectivity

The process for enabling CUBE to support TLS is fairly straightforward. Here's an overview of the necessary steps:

Step 1. Generate RSA keys for a certificate signing request (CSR).

```
cube(config)# crypto key generate rsa modulus 2048 exportable
label cubetls
```

https://t.me/learningnets

@cisco_in_persian_channel

Step 2. Create trustpoints to store certificate data within IOS/IOS-XE (see Example 11-4).

Example 11-4 *Create Trustpoints*

```
cube(config)# crypto pki trustpoint rootca (This will be the Trustpoint for the Root
  CA Certificate)
cube(ca-trustpoint)# enrollment terminal pem
cube(ca-trustpoint)# revocation-check none
cube(ca-trustpoint)# serial-number none
cube(ca-trustpoint)# ip-address none
cube(ca-trustpoint)# subject-name CN = CA,DC = ciscopucs,DC = com
cube(ca-trustpoint)# exit

cube(config)# crypto pki trustpoint cubetls  (This will be the Trustpoint for the
  CUBE's CA signed Certificate)
cube(ca-trustpoint)# enrollment terminal pem
cube(ca-trustpoint)# serial-number none
cube(ca-trustpoint)# fqdn cube.ciscopucs.com
cube(ca-trustpoint)# ip-address none
cube(ca-trustpoint)# subject-name cn=cube.ciscopucs.com,ou=DOD,o=Cisco,l=RTP,st=NC,c=US
cube(ca-trustpoint)# subject-alt-name cube.ciscopucs.com

cube(ca-trustpoint)# revocation-check none
cube(ca-trustpoint)# rsakey
cube(ca-trustpoint)# rsakeypair cubetls
cube(ca-trustpoint)# end
```

Step 3. Create a CSR for CUBE, as shown in Example 11-5.

Example 11-5 *Create a CSR*

```
cube(config)# crypto pki enroll cubetls (this should match the previously created
  trustpoint)
% Start certificate enrollment ..

% The subject name in the certificate will include: cn=cube.ciscopucs.com,ou=DOD,o=Cisc
  o,l=RTP,st=NC,c=US
% The subject name in the certificate will include: cube.ciscopucs.com
Display Certificate Request to terminal? [yes/no]: yes
Certificate Request follows:

-----BEGIN CERTIFICATE REQUEST-----
MIIC7TCCAdUCAQAwgYYxCzAJBgNVBAYTAlVTMQswCQYDVQQIEwJOQzEMMAoG
CSqGSIb3DQEBBQUAA4IBAQCQ21Db0fuciEnuZ/dTNhgEtI6Vr229r68aF+F3k31S
xN6XohaEaZdCDc5qWaakjS72yar/NP8SrAALLV7FryhS
-----END CERTIFICATE REQUEST-----
Note: The certificate output has been truncated for brevity
```

https://t.me/learningnets

@cisco_in_persian_channel

To obtain a CA-signed certificate, you must copy the output starting with **-----BEGIN CERTIFICATE REQUEST-----** and ending with **-----END CERTIFICATE REQUEST-----**. At this point, it can be submitted to be signed by the CA server. The CA signer should also be able to provide root and intermediate CA certificates at this time so that they can be imported into the IOS trustpoints.

Step 4. Obtain the CA (Root and Intermediate) certificates, as in Example 11-6, and import them into IOS/IOS-XE trustpoint(s), as shown in Example 11-7.

Example 11-6 *Importing the Root CA Certificate into the rootca Trustpoint*

```
cube(config)# crypto pki authenticate rootca

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
MIIDXzCCAkegAwIBAgIQC3cKA1lnX4hJBLCDuPYKIzANBgkqhkiG9w0BAQUFADBC
MRUwEwYKCZImiZPyLGQBGRYFbG9jYWwxFDASBgoJkiaJk/IsZAEZFgRoYWxsMRMw
uc06YY/Tp/pAHD3OvXOHmDIiOkzOQBOJMJQaRV8i8MiTBHO3fvviRyYEmySeebGJ
-----END CERTIFICATE-----
quit
Certificate has the following attributes:
       Fingerprint MD5: 9B75726F 4169A727 96DCE60E 70F503C8
      Fingerprint SHA1: DC3C46D2 B9A06147 3FD40B6F 72D8DD72 98128775

% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported
Note: The certificate output has been truncated for brevity
```

Example 11-7 *Importing the Root CA Certificate into the cubetls Trustpoint*

```
cube(config)# crypto pki authenticate cubetls

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
MIIDXzCCAkegAwIBAgIQC3cKA1lnX4hJBLCDuPYKIzANBgkqhkiG9w0BAQUFADBC
MRUwEwYKCZImiZPyLGQBGRYFbG9jYWwxFDASBgoJkiaJk/IsZAEZFgRoYWxsMRMw
uc06YY/Tp/pAHD3OvXOHmDIiOkzOQBOJMJQaRV8i8MiTBHO3fvviRyYEmySeebGJ
-----END CERTIFICATE-----
```

https://t.me/learningnets

@cisco_in_persian_channel

```
quit
Certificate has the following attributes:
       Fingerprint MD5: 9B75726F 4169A727 96DCE60E 70F503C8
      Fingerprint SHA1: DC3C46D2 B9A06147 3FD40B6F 72D8DD72 98128775

% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported
Note: The certificate output has been truncated for brevity
```

Step 5. Obtain CUBE's CA-signed (server) certificate and import it into IOS/IOS-XE (see Example 11-8).

Example 11-8 *Importing CUBE's CA-Signed Server Certificate into the cubetls Trustpoint*

```
cube(config)# crypto pki import cubetls cert


Enter the base 64 encoded certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
MIIFOjCCBCKgAwIBAgIKWhxCdgAAAAAAHDANBgkqhkiG9w0BAQUFADBCMRUwEwYK
CZImiZPyLGQBGRYFbG9jYWwxFDASBgoJkiaJk/IsZAEZFgRoYWxsMRMwEQYDVQQD
Hz8yL/73JEZnOCM8DZAQ774AwyXKwi5g/cnh2fslnk2FO97mS4TOKRR2MRU9wUoU
-----END CERTIFICATE-----
quit
% Router Certificate successfully imported
Note: The certificate output has been truncated for brevity
```

Step 6. Import CUBE's CA-signed certificate into the CallManager-Trust certificate store.

To upload CA-signed certificates into Communication Manager, navigate to **Cisco Unified OS Administration > Security > Certificate Management > Upload Certificate/Certificate Chain**. Next, select **CallManager-trust** and then CUBE's CA-signed certificate and finally select **Upload**, as shown in Figure 11-6. When the upload is successful, you are instructed to restart CallManager and TFTP services.

Note If using the same CA on CUBE and UCM this step is not necessary

https://t.me/learningnets

@cisco_in_persian_channel

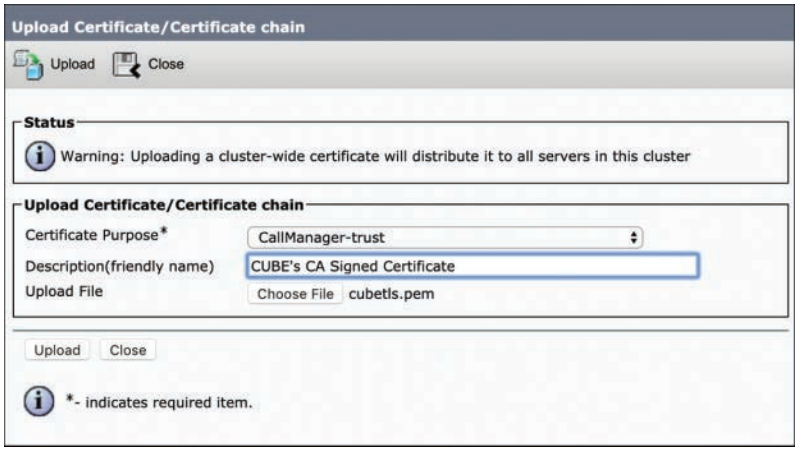


Figure 11-6 *Importing CUBE's CA Certificate into the CallManager-trust Certificate Store*

After completing the necessary work with CA-signed certificates, you need to create a secure SIP trunk profile within Unified CM and then apply the profile to the SIP trunk within Unified CM, as shown in Figure 11-7. The newly created SIP trunk security profile needs to include the fully qualified domain name of CUBE, as used when creating CUBE's certificate within IOS.

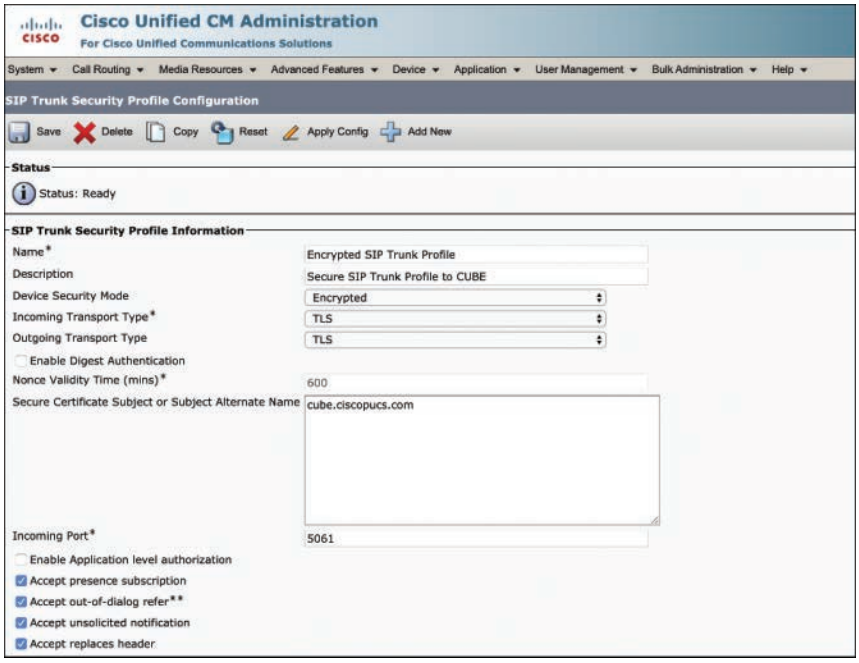


Figure 11-7 *Creating a Secure SIP Trunk Profile Within Cisco Unified CM*

https://t.me/learningnets

@cisco_in_persian_channel

The last step within UCM is to apply the security profile to the SIP trunk and to select the **SRTP Allowed** checkbox on the trunk to allow encrypted media. Figure 11-8 shows where to do this within UCM.

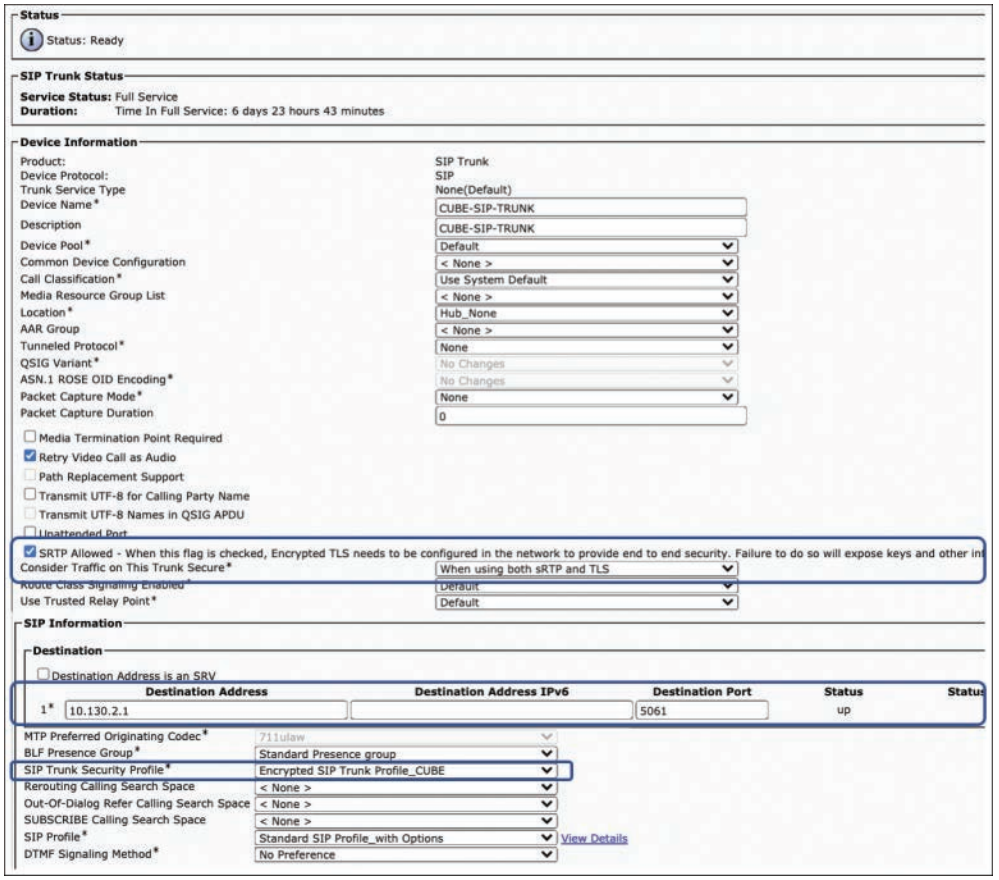


Figure 11-8 *Applying a SIP Trunk Security Profile to a SIP Trunk Within UCM*

To complete the configuration within CUBE, you need to instruct it to use the proper certificate for TLS connections. Because you have already created dial-peers, you can just add the necessary features to enable SIP-TLS and SRTP encryption for secure call routing, and then mirror configurations for each remote site or for future scenarios in which the IP PSTN provider supports TLS connections.

Step 1. Configure CUBE to use its certificate for TLS signaling.

```
cube(config)# sip-ua
cube(config-sip-ua)# crypto signaling remote-addr 10.130.2.2
255.255.255.255 trustpoint cubetls
```

https://t.me/learningnets
@cisco_in_persian_channel

Step 2. Configure CUBE dial-peers.

```
dial-peer voice 100 voip
session transport tcp tls   (specifies TLS transport for signaling)
srtp                        (specifies SRTP media)
```

Perhaps you've wondered why you don't need to import UCM's certificate into CUBE. Because the root CA certificate has been imported into the CUBE trustpoints, it is able to validate the entire certificate chain and trust UCM.

Step 3. Verify that a TLS connection has been successfully established by issuing the **show sip-ua connections tcp tls detail** command from the CLI, as shown in Example 11-9.

Example 11-9 *Verifying the TLS Connection*

```
cube# show sip-ua connections tcp tls detail
Total active connections       : 1
No. of send failures           : 0
No. of remote closures         : 0
No. of conn. Failures          : 0
No. of inactive conn. ageouts  : 0
TLS client handshake failures  : 0
TLS server handshake failures  : 0

---------Printing Detailed Connection Report---------
Note:
 ** Tuples with no matching socket entry
    - Do 'clear sip <tcp[tls]/udp> conn t ipv4:<addr>:<port>'
      to overcome this error condition
 ++ Tuples with mismatched address/port entry
    - Do 'clear sip <tcp[tls]/udp> conn t ipv4:<addr>:<port> id <connid>'
      to overcome this error condition

Remote-Agent:10.130.2.2, Connections-Count:1
  Remote-Port Conn-Id  Conn-State   WriteQ-Size   Local-Address  TLS-Version
  =========== =======  ===========  ===========   =============  ===========
        45703       3  Established      0            10.130.2.1      TLSv1.2


-------------- SIP Transport Layer Listen Sockets ---------------
  Conn-Id                Local-Address
 ===========    ==============================
   0                     [0.0.0.0]:5061:
   2                     [10.130.2.1]:5061:
```

https://t.me/learningnets

@cisco_in_persian_channel

VPN-Based Solutions

Many IT organizations, including Cisco IT, have developed "virtual office" solutions, which are based on IPsec VPN technology. The purpose of these types of solutions is to provide connectivity to the corporate resources while also ensuring privacy of data communications over an unsecure Internet connection. Virtual office solutions are based on either small-office/home-office (SoHo) hardware or software. If they're software based, a VPN client such as Cisco AnyConnect is used, along with clients such as Cisco Jabber that provide Unified Communications.

In a VPN-based solution, a user must establish an encrypted tunnel (to ensure privacy) while connected to a VPN termination point that is located inside a corporate data center(s). If hardware is used to originate a VPN connection, users gain the flexibility of choosing from various types of VoIP and IP video endpoints, as well as software-based endpoints, based on which provide the best user experience. A challenge with this option is that a management burden is placed on corporate IT because an individual (or team of individuals) must be able to manage the VPN hardware, software configurations, and software versions deployed at all of the various remote locations, not to mention the costs associated with obtaining the VPN hardware.

In certain circumstances, additional challenges may arise depending on an end user's workflow, such as when a teleworker tries to conduct a VoIP or video call over the VPN. Because real-time traffic rides over a network, such as the Internet, the quality of a conversation cannot be guaranteed. Typical issues that arise are due to suboptimal routing, latency, and/or delay. For non-real-time traffic, such as instant messaging, there is little impact to the user experience. From a security perspective, there is risk when using VPN-based solutions because these solutions are vulnerable to unauthorized users connecting to corporate resources. When unauthorized users are able to connect to corporate networks, they can conduct many types of nefarious activities that include but are not limited to exfiltrating data, planting Trojan horses, spreading malware, and more. To mitigate these risks, a network-based approach is recommended, as we discussed in Chapter 3, "Security Through Network Fundamentals," such as through the use of 802.1x authentication and NAC.

VPN-less Solutions

A modern way of providing teleworkers secure access to UC services is by leveraging protocols that have security mechanisms, such as Transport Layer Security, already built into them. This option is beneficial and cost-effective because it simplifies the management and onboarding processes without necessarily requiring an additional procurement of VPN hardware for the teleworker locations. Leveraging TLS 1.2 encryption, VPN-less solutions are just as secure as traditional IPsec VPN-based solutions.

As defined in RFC 5246, TLS 1.2 allows client/server applications to communicate in a way that is designed to prevent eavesdropping, tampering, or message forgery. One of the differences between TLS 1.2 and TLS 1.1 is that TLS 1.2 mandates the use of SHA-256. Another benefit of TLS 1.2 is support for elliptic curve cryptography (ECC), which is

https://t.me/learningnets

@cisco_in_persian_channel

defined in RFC 4492. ECC is a public-key cryptosystem for mobile devices and mobile environments because it offers equivalent security with smaller key sizes. Smaller key sizes lead to less effort for computation, which helps with power savings, memory, and bandwidth usage.

Cisco's Expressway series is the key functional component to providing VPN-less connectivity. As of X8.1.1 software, the Mobile and Remote Access (MRA) feature allows customers to connect to enterprise UC services, even if that endpoint is not connected to the local network, and without VPN connectivity. When Expressway is deployed, it facilitates traversal of perimeter firewalls, which are in place to protect the enterprise network from untrusted traffic. By default, MRA uses encryption to secure UC signaling and media. At this time, the option for VPN-less connectivity with MRA comes with some of the same risks to the user experience due to the nature of traversing the Internet. Cisco is currently working on optimizing the media path to help alleviate user experience issues by leveraging Interactive Connectivity Establishment (ICE). We provide more information about ICE later in the chapter.

Business-to-Business Communication

As organizations look to working smarter and more effectively, effective communication both within organizations and to their business partners has become a priority. Connecting businesses over the Internet has many benefits, including but not limited to the ability to conduct transactions more rapidly while reducing costs. As organizations work through the complexities of integrating systems, taking orders, and sharing information, technology that provides Unified Communications is critical to success.

Cisco Expressway enables organizations to provide secure business-to-business connectivity. Expressway can support either SIP or H.323 trunks to upstream organizations or IT service providers for voice and/or video calling. As of X8.2 software, Expressway adds additional B2B support by providing XMPP and SIP Federation for instant messaging between organizations. Federation though Expressway supports on-site or off-premises deployment types, such as to Skype for Business in a cloud location. Expressway software is the key component to providing VPN-less connectivity to UC services as well as providing secure B2B connectivity. The rest of this chapter examines the security features that are contained within Expressway and also the security mechanisms that are used to provide secure connectivity without use of a VPN.

Security Features Within Expressway

It is recommended that organizations use an external perimeter firewall to provide filtering of traffic that is destined to Expressway-E. When Expressway-E is deployed in an organization's DMZ environment, behind a perimeter firewall, it can still be vulnerable to well-known attacks such as network reconnaissance attacks, denial-of-service attacks, and unauthorized attempts to gain administrative access to Expressway-E. To defend against these types of attacks, organizations should enable Expressway's built-in security features to protect it from both network and application layer attacks. This is a way for

https://t.me/learningnets

@cisco_in_persian_channel

organizations to deploy security in layers to help prevent attacks against Expressway. These two architectural components provide a modular approach for how policy can be enforced and provide protection for other types of services that may need to be accessed from the Internet. To protect against unwanted traffic entering through the inside of the corporate IT environment, organizations should leverage an internal firewall. The next section provides more information about some of the security features built into Expressway.

Network-level protection within Expressway is provided by two main components: firewall rules and automated intrusion protection. Firewall rules enable an organization to secure Expressway by controlling the ability to

- Specify the source IP address subnet from which to allow or deny traffic
- Choose whether to drop or reject denied traffic
- Configure well-known services such as SSH and HTTP/HTTPS, or specify customized rules based on transport protocols and port ranges
- Configure different rules for the LAN 1 and LAN 2 interfaces on Expressway-E

Expressway uses the Linux-based IP Tables software to enforce firewall rules. An example of a firewall rule that can be configured is shown in Figure 11-9.

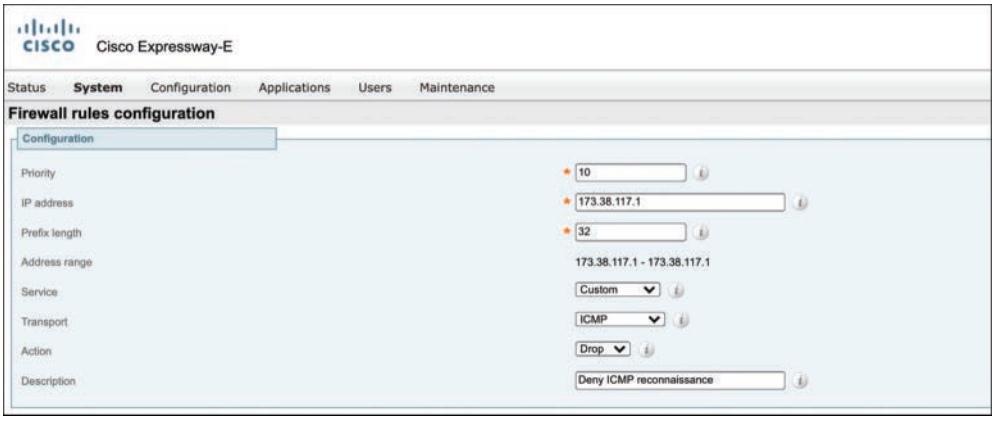


Figure 11-9 *Firewall Rule Within Expressway-E, Denying ICMP Traffic from a Specific Host*

Automated Intrusion Prevention (AIP) is enabled by default starting with X8.9 software. When enabled, AIP uses the Linux-based Fail2ban software to detect and block malicious traffic that may be used to gain administrative access to Expressway by automatically

- Detecting repeated failures to access specific service categories, such as SIP, SSH, and web/HTTPS

https://t.me/learningnets

@cisco_in_persian_channel

- Preventing attempts to breach login security by automatically blocking the source host IP address (the intruder) and destination port for a specified period of time

While AIP protects against the most common vectors of an attack (e.g., unauthorized attempts to access the system by SSH), Cisco recommends enabling these other specific features on Expressway-E:

- HTTP proxy authorization failure
- HTTP proxy protocol violation
- XMPP protocol violation

Similar to what we discussed in Chapter 3, with VLAN hopping, sophisticated attackers have tools, programs, and scripts at their disposal to launch attacks against Expressway. As defined in RFC 4732, Internet Denial-of-Service Considerations, potential denial-of-service attacks can be launched against any protocol that is used, such as SIP, HTTP, and XMPP systems leveraging protocol violations in protocols to exploit software quality, such as to create a buffer overflow. Additionally, attackers may attempt to target a weakness in system resources such as memory, CPU, disk space, or with infrastructure such as DNS. Because SIP is not enabled by default, AIP does not automatically monitor for SIP protocol violations, SIP registration failures, or SIP authentication failures.

When MRA is enabled, organizations should consider enabling AIP for these additional categories (and protocols such as SIP) to monitor. To do this, as administrator, you can navigate to **System > Protection > Automated Protection > Configuration**. As shown in Figure 11-10, we also recommend that you take the time to investigate intrusion attempts and review the relevant log messages that are associated with each category. When AIP detects repeated attempts, you should consider adding a permanent firewall rule within Expressway-E or on an external filtering device such as a firewall. By default AIP automatically unblocks a host address after a time period so as not to lock out any genuine hosts that may have been temporarily misconfigured. Cisco recommends using this feature on Expressway-E because it is located at the edge of the network and disabling it on Expressway-C.

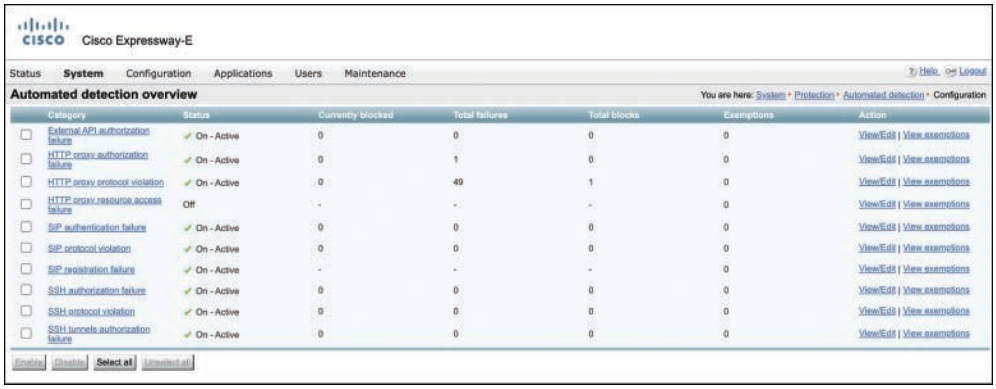


Figure 11-10 *Verification of Automated Detection Configuration Within Expressway-E*

https://t.me/learningnets
@cisco_in_persian_channel

Cisco recommends not enabling the HTTP proxy resource access failure category because IP phones and Jabber clients often request files that are not always accessible over MRA. In this case, a false positive may be created and a self-inflicted DoS can occur, blocking end-user access to legitimate users. If the organization suspects that AIP is causing a self-inflicted DoS, rather than turn off an entire category, you can make exemptions for specific networks and categories. An example is shown in Figure 11-11.

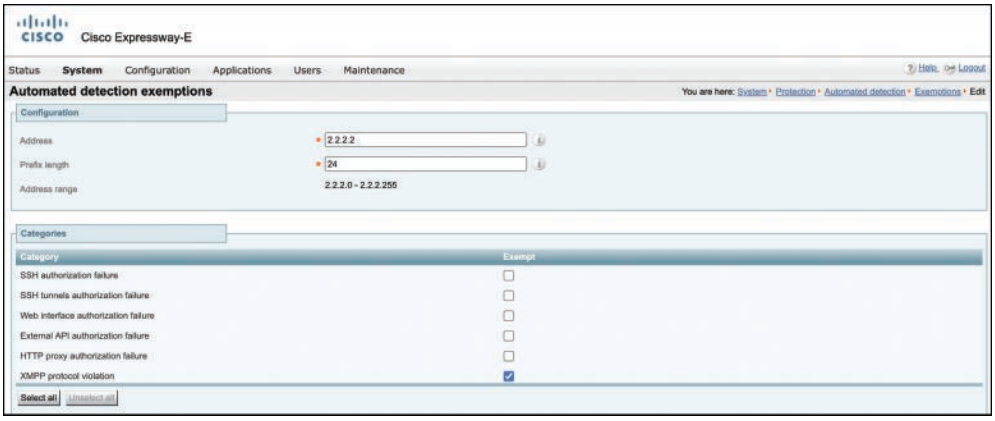


Figure 11-11 *Exemptions Within Expressway's Automated Detection to Eliminate False Positives*

When using the MRA feature within Expressway, all traffic between the remote-access client and Expressway-C is always encrypted. Leveraging the security parameters that are built into existing protocols such as TLS, customers are able to obtain VPN-less connectivity to UC services. In the next section, we identify the protocols that are used to provide security when Expressway is deployed with the intention of providing MRA support for the user community that is located at off-premises locations.

Deploying Mobile and Remote Access

Organizations can achieve VPN-less connectivity while working remotely by using the Mobile and Remote Access solution provided by Cisco. To support MRA, the Expressway series application is divided into two different roles, providing both Core and Edge connectivity. Expressway-C (aka Core) sits inside the enterprise boundary, typically within a data center. Expressway-E (aka Edge) is typically deployed at the edge of the enterprise boundary, typically in a DMZ between firewalls. Together, the Expressway pair of C/E work in conjunction to minimize the number of ports that a firewall needs to open to communicate with the internal UC infrastructure.

From a design perspective, it is important to limit the number of ports that need to be opened through a firewall because having fewer ports reduces the attack surface for applications that are deployed on the edge of the network. A reduced attack surface helps reduce the likelihood that an organization will be susceptible to certain types

https://t.me/learningnets
@cisco_in_persian_channel

of attacks, such as a denial of service. As shown in Figure 11-12, Expressway-E (Edge) requires only four specific TCP and UDP ports/port ranges opened to communicate with Expressway-C (Core), which is located adjacent to Unified CM on the internal network. The Assent protocol is used in the traversal zone between Expressway-C and -E to minimize the number of ports that are needed to be opened on the internal firewall. To reduce the number of ports that need to be opened, Assent works to multiplex/demultiplex real-time media.

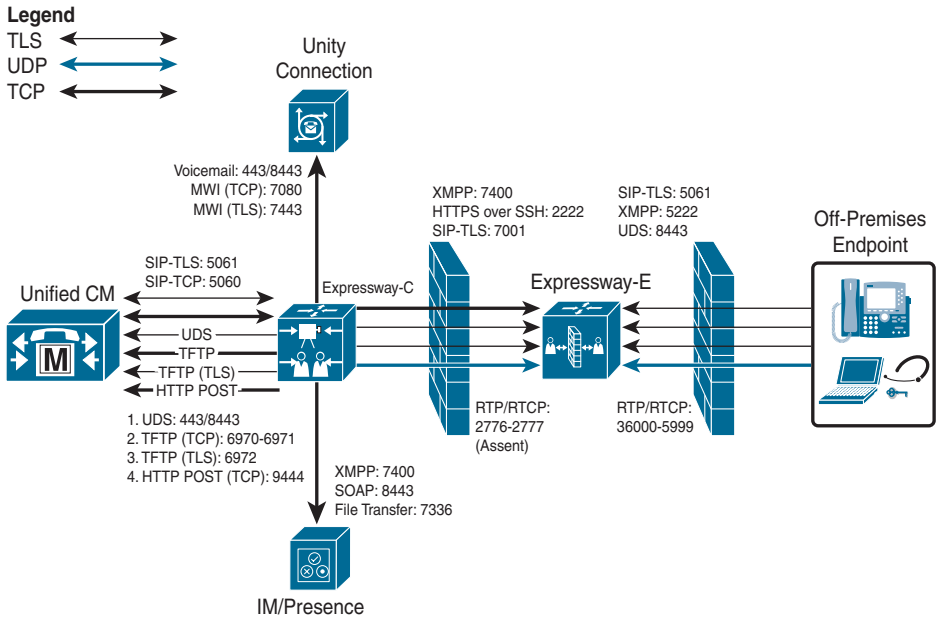


Figure 11-12 *TCP and UDP Ports Used by Expressway When Enabled for Mobile and Remote Access*

To start configuring MRA, the organization must meet some basic infrastructure requirements. Cisco requires use of NTP services that are synchronized with a reliable time service and DNS. The next section provides additional details for how UC clients and Expressway servers leverage DNS in support of MRA.

DNS

When an organization is deploying Mobile and Remote Access, the Cisco recommendation is to have a single domain (e.g., cisco.com) but with a split DNS architecture. This configuration allows clients that reside on different networks in different locations to resolve DNS names differently. This split DNS architecture implies that server infrastructure such as UCM and Expressway-C need to have the ability to resolve (and be resolved) host names using one or more internal DNS servers. To support external UC endpoints, Expressway-E needs a host name that can be resolved through publicly available DNS servers and have the ability to resolve external addresses.

https://t.me/learningnets
@cisco_in_persian_channel

Service discovery is a reason that split DNS is beneficial to an organization. Split DNS simplifies how endpoints are able to find services, especially when an endpoint is located off-premises. As an example, when a device is located off-premises, it can use external DNS services (DNS SRV) to find an organization's MRA capabilities that are being offered through Expressway. A DNS SRV record typically contains the name of a service, priority, weight, and host name that is offering a service.

When a client tries to find UC resources, it first issues DNS queries for internal resources. The DNS SRV record that must be present inside the internal DNS server can be as follows:

- **_cisco-uds._tcp.example.com**—Used to obtain full UC services by resolving to UCM.
- **_cuplogin._tcp.example.com**—Used to obtain instant messaging services by resolving to IM/P server.

  If present, a _cisco-uds request always takes precedence over _cuplogin. When a device is external, it is not able to find services using either of these DNS SRV queries. When this is the case, a third and final attempt to find services is to search for Collaboration Edge services. For a client to find Collaboration Edge services, a DNS SRV record must resolve to an Expressway-E server. The following SRV record is needed in a publicly available DNS server to help an endpoint find Collaboration Edge services:

- **_collab-edge._tls.example.com**—Used to obtain full UC services by resolving to Expressway-E.

To support Expressway clustering and redundancy, Cisco recommends use of multiple SRV records. Figure 11-13 shows how endpoints use DNS infrastructure and discover services.

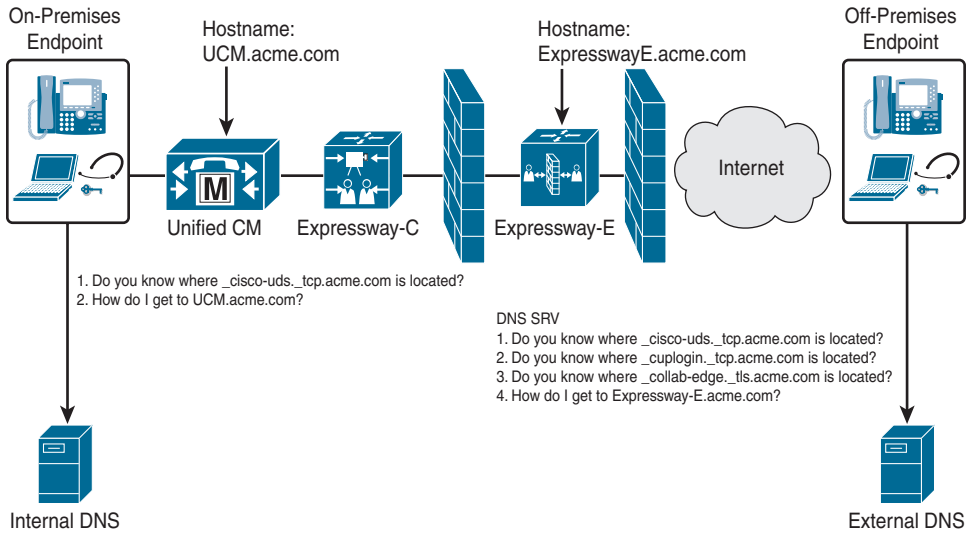


Figure 11-13 *Split DNS Architecture Showing Internal and External DNS Servers*

https://t.me/learningnets
@cisco_in_persian_channel

To perform an initial configuration for MRA, follow these steps:

Step 1. Enable SIP on Expressway-C and -E. To do this, navigate to **Configuration > Protocols > SIP**.

Step 2. Enable Mobile and Remote Access. To do this, navigate to **Configuration > Unified Communications > Configuration** and change the Unified Communications mode to Mobile and Remote Access on both Expressway-C and -E.

Step 3. Add UCM, IM/P, and Unity connection servers into Expressway-C. To do so, navigate to **Configuration > Unified Communications** and select the specific server type.

Step 4. Add the SIP domain into Expressway-C and select services that you would like to offer over MRA. You can do this by navigating to **Configuration > Domains**. An example is provided in Figure 11-14. Because we are discussing use of MRA, you can select SIP registrations and provisioning on UCM. Doing so allows UCM to provide all endpoint registration, call control, and instant messaging for the SIP domain. In this solution, Expressway simply acts as a UC gateway to proxy user registrations and to securely traverse the internal firewall as traffic connects from the edge to the inside of the network.

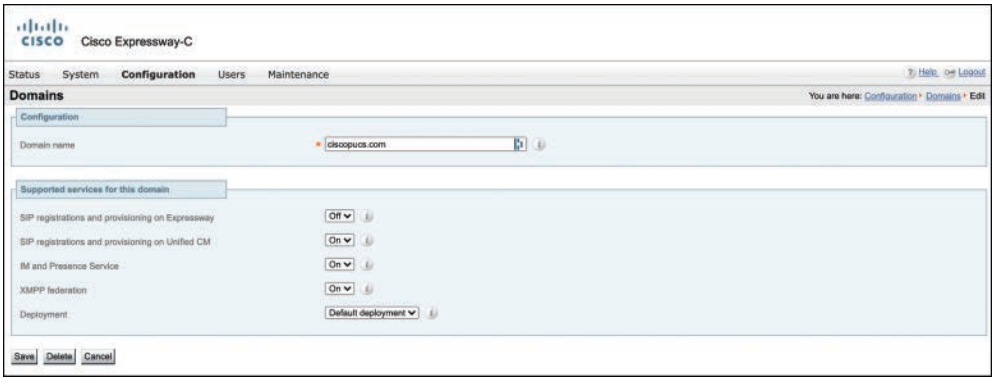


Figure 11-14 *Sample Configuration to Configure the SIP Domain and UC Services Within Expressway-C*

X.509 certificates are required when using MRA to provide secure and encrypted UC services across an insecure network such as the Internet. The next section goes into further detail to help provide information for which options an organization has to apply certificates on Expressway-C and -E.

https://t.me/learningnets

@cisco_in_persian_channel

Certificate Requirements for Mobile and Remote Access

As discussed in previous chapters, X.509 certificates are needed to set up trusted TLS connections. This does not change when establishing a TLS connection between Expressway-C and Expressway-E. With Expressway, organizations can take advantage of some nice enhancements when using TLS to establish trust. As an example, Expressway supports use of Online Certificate Status Protocol (OCSP) for SIP-TLS connections. Organizations can use OCSP to query the status of a CA-signed certificate without having to download and parse through Certificate Revocation Lists (CRLs).

An organization can use multiple options to obtain signed certificates for Expressway-C and -E servers. Although self-signed certificates are supported, CA-signed certificates are recommended. The only server that is really required to be signed by a public CA is Expressway-E. One reason that this requirement exists is for supporting UC endpoints which support MRA users that reside off-premises. When certificates are signed by a public CA, they can establish a trusted connection to Expressway-E. Registration of 7800 and 8800 series phones over MRA also requires use of a publicly signed CA certificate on Expressway-E because the certificate trust list on an IP phone cannot be modified to trust privately signed CA certificates.

CA-signed certificates for Cisco's Core UC applications require a mixture of certificates that support the Server Authentication and Client Authentication extensions. To verify which CA-signed certificates are necessary, review the administration guide for your version of software. Use of CA-signed certificates within Expressway must include both Server and Client Authentication extensions. Otherwise, the operating system does not allow you to upload the server certificate when Unified Communications features are enabled.

Expressway-E leverages a public key infrastructure (PKI) to ensure secure connectivity across untrusted networks, such as the Internet. A PKI consists of a certificate authority that is used to store, issue, and sign certificates. Several major publicly available CAs, such as GoDaddy, Verisign, and others, can provide digitally signed certificates to help ensure TLS connection across the Internet.

For a full and inclusive list of Cisco trusted CAs, you can download the following document: www.cisco.com/c/dam/en/us/td/docs/voice_ip_comm/dx/series/ca/CA-Trust-List.docx.

One reservation that an organization may have with using publicly signed X.509 certificates to establish a TLS connection is the ongoing costs and operational overhead of updating certificates. For this reason, Cisco has developed support for obtaining free CA-signed certificates from Let's Encrypt (https://letsencrypt.org/). Let's Encrypt is a nonprofit organization that provides free X.509 signed certificates for customers and integrates into Expressway when running X12.5 software or later. Certificates from Let's Encrypt are valid only for 90 days, but if you use the Automated

https://t.me/learningnets

@cisco_in_persian_channel

Certificate Management Environment (ACME) protocol, certificates can be automatically renewed if the automated scheduler is enabled. Automatically renewing certificates helps ease the burden of IT administrators having to remember to renew their CA-signed certificates as they approach their expiration date. The page to configure the ACME certificate service is shown in Figure 11-15.

Figure 11-15 *Expressway-E Server Certificate Using the Certificate Authority from Let's Encrypt*

If the automated scheduler setting is not enabled, Let's Encrypt provides a bot service that provides an email reminder in advance of certificate expiry to make sure that administrators renew all relevant certificates on Expressway-E. Let's Encrypt certificates can be manually renewed at any time. To support the ACME protocol, perimeter firewalls need to allow inbound TCP ports 80 and 443.

For MRA deployments, and for Expressway-C to trust the Core UC applications such as Cisco Unified CM and IM&P, Expressway-C must be able to trust the Tomcat certificates. For encrypted signaling, trust must also be established by exchanging the certificates between Communications Manager and Expressway-C. The recommended configuration is to use a certificate authority to sign certificate requests for all of the Core UC application services (such as Tomcat). For a clustered Expressway environment, the certificate trust is simplified because the IT administrator needs to upload only one to two certificates: the root CA certificate and intermediate certificate (if applicable) of the certificate issuer. Figure 11-16 shows Expressway-C's Trusted CA certificate store, which includes certificates from the Public CA—Let's Encrypt (Intermediate and Root), along with the privately signed CA certificates.

https://t.me/learningnets

@cisco_in_persian_channel

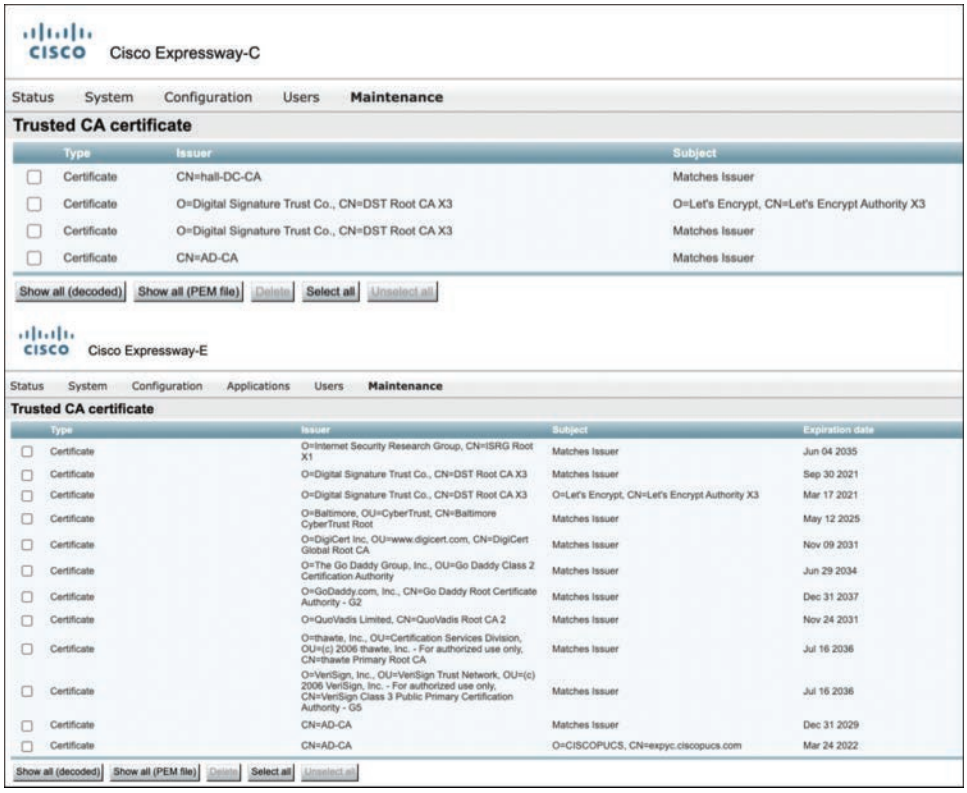


Figure 11-16 *Trusted CA Certificates Uploaded into Expressway-C and -E*

When an organization uses certificates on Expressway-C and -E (and also with other Unified Communications infrastructure), attributes within the certificates such as common name (CN) and subject alternative name (SAN) use fully qualified domain names. These FQDNs must be resolvable through DNS to establish connectivity and trust between certificates.

Firewall Traversal

Previously, we explained that Expressway is used to traverse an organization's firewall to support MRA. The feature that is configured to support this capability is called a traversal zone, which is a mutual TLS (MTLS) connection between Expressway-C and Expressway-E. A traversal zone is based on a client/server model, with Expressway-C acting as the client and Expressway-E acting as the server.

An MRA traversal zone must be configured with a traversal zone type of Unified Communications traversal, which uses SIP-TLS and TLS verify mode set to on. TLS verify mode helps ensure that a secure connection can be made by examining the common name from the X.509 certificate. Because Expressway can act as both a client and a server, Expressway's certificate must be valid for both a client and a server. A sample configuration is provided in Figure 11-17.

https://t.me/learningnets
@cisco_in_persian_channel

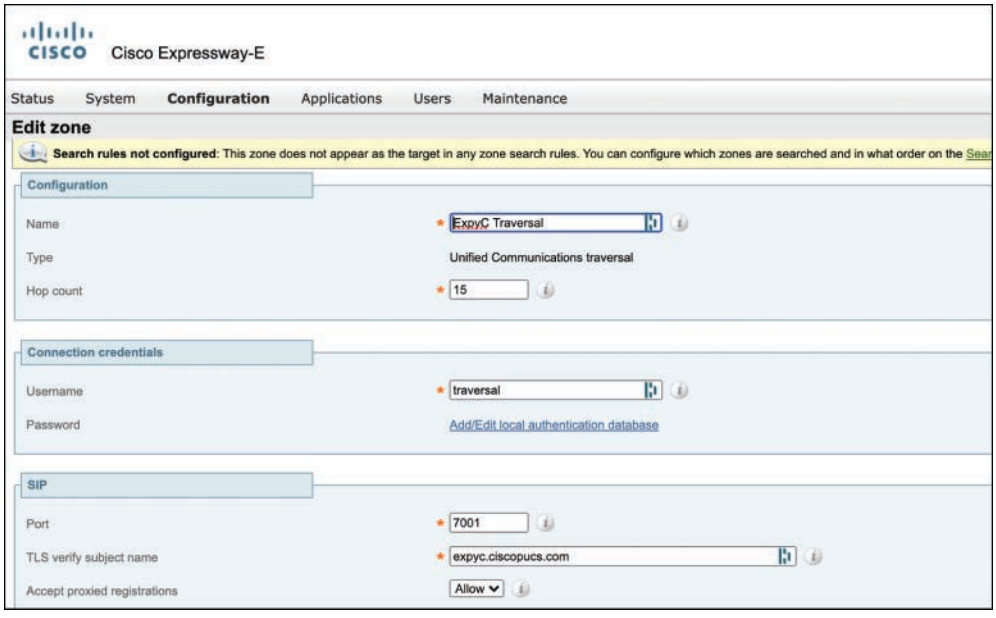


Figure 11-17 *Configuration for a UC Traversal Zone on Expressway-E*

Authentication and Authorization for MRA

An area in which Cisco has made some of the most significant improvements for customers is authentication and authorization. Leveraging a modern framework for authentication and authorization, Cisco enables organizations to implement security on a more granular level and on an as-needed basis. In many ways, the new framework that Cisco provides is analogous to a zero-trust architecture. This is based on the concept that assumes an end user is somehow compromised, so therefore privileges should be granted on an as-needed basis and security controls must be layered across each area of the UC infrastructure to prevent a security incident.

Cisco's new authentication framework, released as of Collaboration Systems Release (CSR) 12.0, also provides customers with a greater degree of simplicity. As previously discussed, typically X.509 certificates and a PKI are needed to support encrypted services. This means that to deploy MRA prior to CSR 12.5, an organization would need to put its Unified CM into mixed-mode, deploy a strict PKI with CA-signed certificates on, and then push out certificates to endpoints using CAPF or CAPF enrollment with authentication strings to establish a secure connection at the Collaboration Edge. Enabling OAuth across a UC infrastructure that supports MRA (e.g., Unified CM, IM/P, Unity Connection, Expressway) allows customers to have the flexibility of using a mixed-mode cluster with CAPF or to use OAuth tokens to provide secure access to services without a VPN. This increased flexibility allows organizations to rapidly deploy UC services at the edge of the network in minutes so that Jabber users can establish secure connectivity to internal UC services.

https://t.me/learningnets

Note As of CSR 12.5, IP Phones and video endpoints running CE software do not support SIP OAuth.

Because we already discussed how to enable OAuth in Chapter 7, the next few sections address how to enable OAuth within Expressway and describe the MRA endpoint. When MRA is enabled, you have several authentication options, including

- **Unified CM/LDAP basic authentication:** Authentication is performed by Unified CM against LDAP credentials.
- **SAML SSO authentication:** Authentication is provided by a SAML 2.0 Identity Provider (IdP).
- **SAML SSO and Unified CM/LDAP:** These options allow either method.

For the most flexible authentication and to take full advantage of OAuth-based authorization, you should select the SAML SSO and Unified CM/LDAP authentication option. The authentication path should include Unified CM/LDAP authentication to support IP phones, Telepresence endpoints, and any Jabber clients homed to a UCM cluster not configured for SSO. Expressway also provides an option to "Check for internal authentication availability." This option should be set to No because when set to Yes, it makes organizations more vulnerable for inbound authentication requests from rogue and/or malicious attackers.

Because OAuth is a framework that provides authorization to services, you have multiple options to choose from. These options, described next, can be used to authorize an endpoint to UC services:

- **Authorize by OAuth token with refresh:** This option enables OAuth with refresh tokens. This is the Cisco recommended option for authorization when UC infrastructure versions are supported. One of the benefits over the OAuth without refresh tokens authorization is that if users authenticate to resources locally, they don't have to reauthenticate when they move off-premises. This OAuth option can be used in conjunction with SAML SSO, UCM, and LDAP authentication. This mode is required with Jabber clients configured with SIP OAuth and IP phones onboarded through an activation code.
- **Authorize by OAuth token:** This option applies only to UC services prior to CSR 12.0 using SSO-based authentication and OAuth Implicit Flow.
- **Authorize by user credential:** This option applies to Jabber clients not using SSO or OAuth token with refresh, supported IP phones (e.g., 7800, 8800 series), not using activation code onboarding and Telepresence devices.

Note The Webex Teams client also supports Authorize by OAuth token with refresh. We will discuss more about Webex Teams in Chapter 12.

https://t.me/learningnets

@cisco_in_persian_channel

Figure 11-18 shows the authentication and authorization policy just created within Expressway-C. Figure 11-19 shows how clients authenticate and receive authorization to use UC services. The last step to configure secure access to MRA is to configure UCM to support OAuth with a new phone security profile, which we discuss in more detail in the next section.

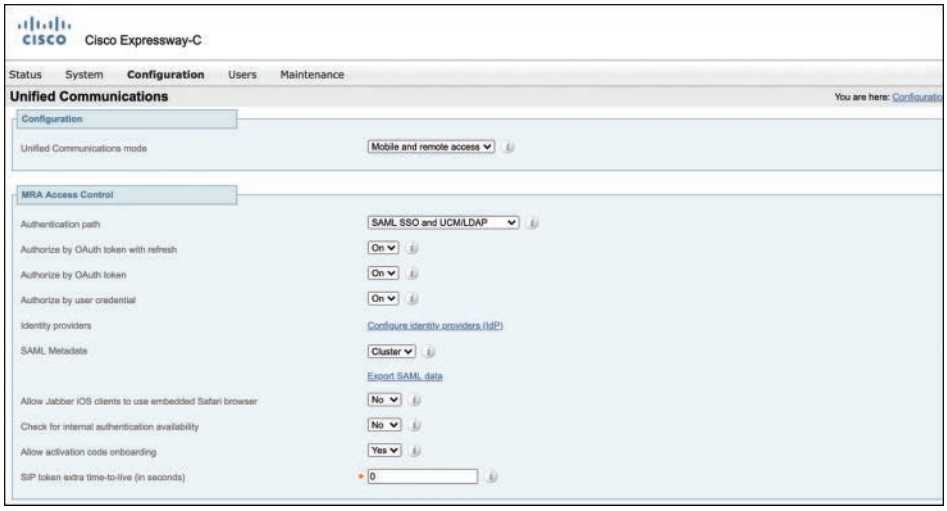


Figure 11-18 *Configuration for Authentication and Authorization Policy on Expressway-C*

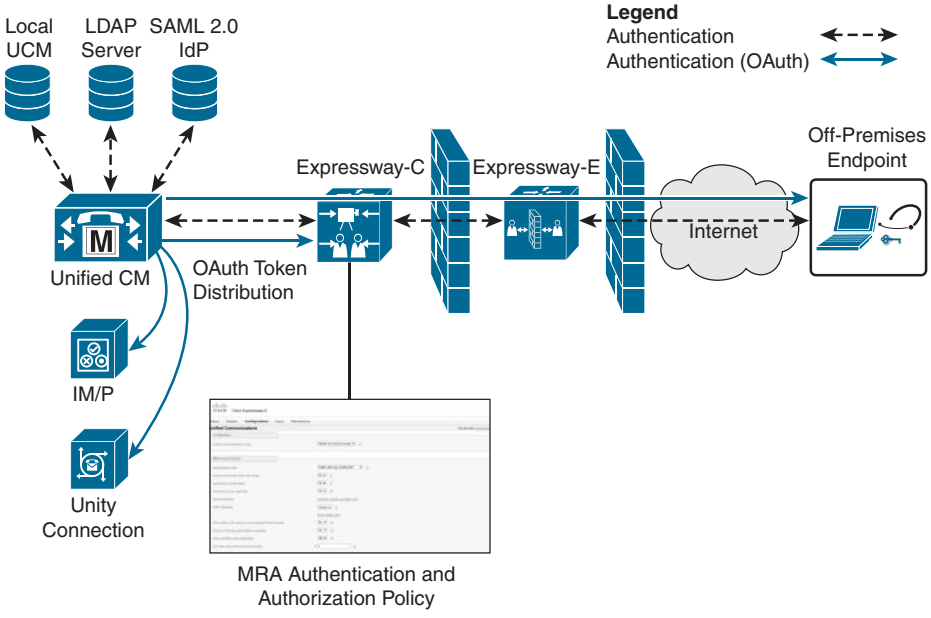


Figure 11-19 *MRA Login Flow with Authentication and Authorization (OAuth) Enabled*

https://t.me/learningnets

Phone Security Profiles

With an MRA deployment, Cisco forces encryption of media between Expressway-E and the devices located beyond the corporate perimeter using TLS. Cisco also forces media encryption between Expressway-C and -E using TLS. Another function of Expressway-E is to proxy client registrations to the UCM cluster through Expressway-C. To support secure registrations to Unified CM, a trusted TLS connection must be established between Expressway-C and Unified CM. MRA uses phone security profiles configured on UCM that support TLS encryption and the SAN field from Expressway-C's server certificate. In a traditional MRA deployment, phone security profiles must be in the form of a fully qualified domain name to match the FQDN in Expressway's server certificate. If a phone security profile is not configured correctly to support encryption, a SAN field was not created to support phone security profiles on Expressway-C during the CSR process, or the parameters do not match, a TLS connection will fail. When using SIP OAuth with Jabber, the SAN field does not need to contain the name of the phone security profile.

As shown in Figure 11-20, the phone security profile name is jabber.secure.ciscopucs.com, and it provides support for OAuth Authentication. When the Enable OAuth authentication box is checked, UCM allows the device that is associated with the phone security profile to register with the SIP OAuth port. By default, the SIP OAuth port is 5090 when connecting to Unified CM internally or 5091 when connecting to Unified CM via MRA. To use the phone security profile, it must be assigned to a device (e.g., Jabber CSF device). As shown in Figure 11-21, the SAN field of the certificate within Expressway-C contains the same name from the phone security profile (jabber.secure.ciscopucs.com). When these two items are configured correctly, you can make secure registrations to Unified CM. An example of how this creates a TLS connection is shown in Figure 11-22.

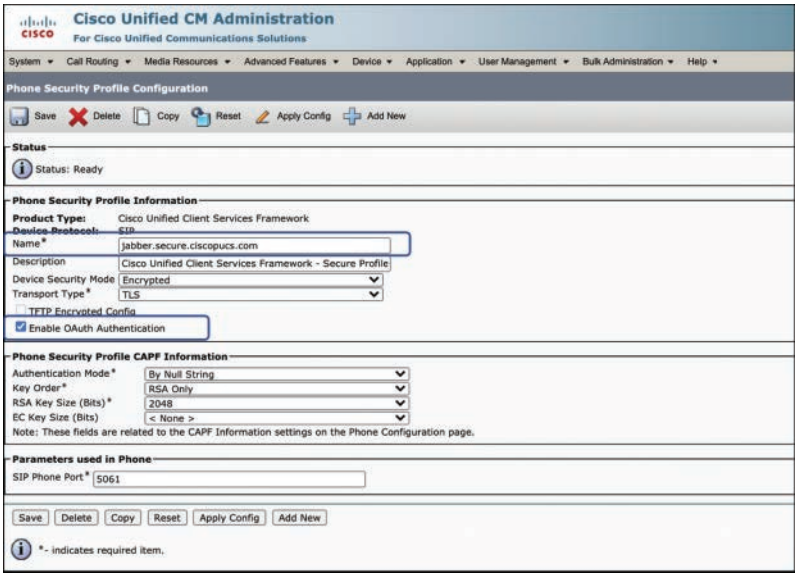


Figure 11-20 *Phone Security Profile Configuration for TLS Encryption*

https://t.me/learningnets
@cisco_in_persian_channel

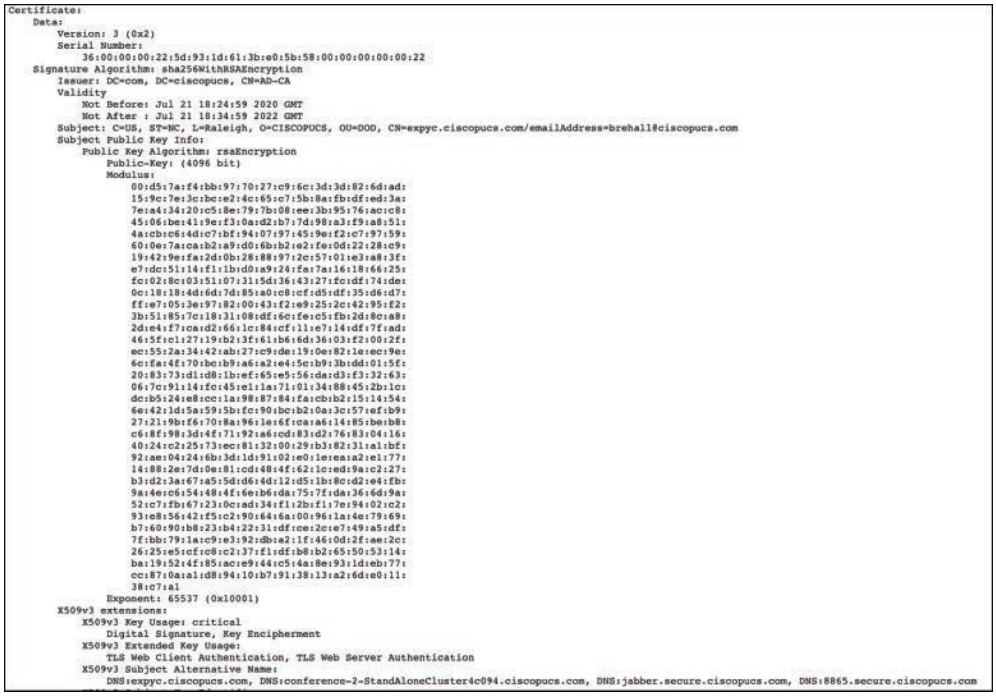


Figure 11-21 *SAN Field from Expressway-C's Server Certificate Containing the Phone Security Profile Name*

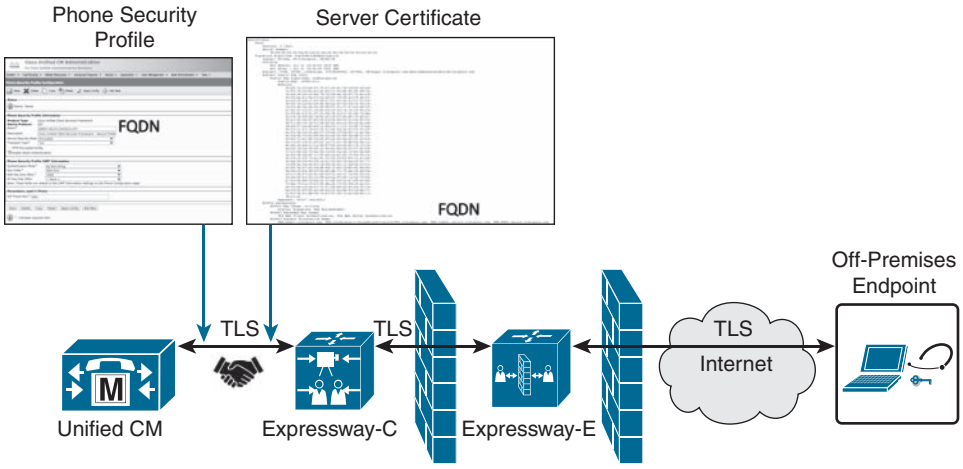


Figure 11-22 *How Phone Security Profiles Are Used to Establish a TLS Handshake and Provide Secure Registrations*

https://t.me/learningnets

@cisco_in_persian_channel

Token Scopes and Revocation

As previously discussed, Cisco's new framework allows for varying levels of privileges to be assigned to users. The reason is that the OAuth access tokens contain a scope element. Beginning in CSR 12.0, the scope element can be used to control access to the MRA service provided by Cisco Expressway and also define which UC services the token is valid for. Within UCM, you can create a user profile that defines the privileges that an MRA user can access (e.g., IM/P, voice, video calling) by device type. To configure this setting, navigate to **User Management > User Settings > User Profile** within UCM. Once created, the user profile must be assigned to an end user(s). An example is shown in Figure 11-23.

Figure 11-23 *Configuration of MRA Policy from Unified CM to Authorize Full UC Services with OAuth Token Scopes*

This policy is enforced when a user with a Cisco Jabber client tries to access the UC environment via MRA. Once authenticated, the access token scope is checked to determine which services the Jabber user is authorized for.

When an organization uses OAuth, a refresh token is used to reauthorize a user to UC services. By default, a refresh token lasts for five years (60 months). If at any time a device is lost or an employee leaves the organization, it is beneficial for the organization to have the ability to revoke the OAuth token, which would force the user to reauthenticate. To do this, you can use AXL administrator credentials to issue the following command to Unified CM's REST-based API:

```
https://<IP_Address_of_UCM>:8443/ssosp/token/revoke?user_id=<ID_of_
end_user>
```

MRA Troubleshooting

If you need to validate and troubleshoot MRA configurations, Cisco has developed a tool called the CollabEdge Validator tool to simulate the MRA login process. The CollabEdge Validator tool can be accessed at https://cway.cisco.com/csa/ and is available for anyone who has a Cisco.com login. The tool is helpful because it simulates every step of the login process toward UCM and IM/P servers through Expressway-E. Based on MRA configurations, the tool provides feedback about whether a feature is correctly configured or whether a check fails. If a failure occurs, corrective actions are provided. An example of the tool is shown in Figure 11-24.

https://t.me/learningnets

@cisco_in_persian_channel

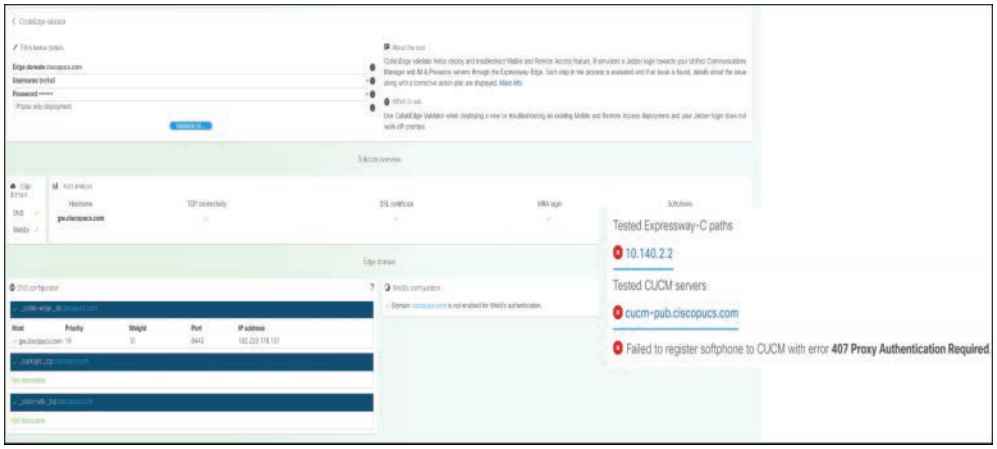


Figure 11-24 *Cisco's CollabEdge Validator Tool, Which Can Be Used to Troubleshoot Connectivity Problems*

Interactive Connectivity Establishment (ICE)

We previously discussed how the traditional IPsec-based VPN-based solutions could cause suboptimal routing of VoIP and/or video traffic. Within a VPN, the UC signaling and media traffic are encapsulated by a VPN tunnel, which connects to a VPN concentration device, which is usually located in a centralized location. By leveraging the ICE architecture, Cisco helps simplify VPN-less connectivity and optimize media flows for Expressway-based Mobile and Remote Access (MRA) users. ICE, as defined in RFC 5245, provides a best-effort mechanism for SIP client NAT traversal, which means that ICE allows remote-access clients (e.g., MRA users) to discover if direct connectivity exists between endpoints, so that real-time media can flow directly between endpoints instead of a central location, such as a VPN concentrator, to provide the most optimum user experience.

ICE combines two other protocols: STUN and TURN. Session Traversal Utilities for NAT (STUN) is based on RFC 5389. Using STUN, endpoints and clients can create and maintain UDP connections through firewalls and NAT devices. If outbound UDP traffic is not allowed, STUN does not work, and Traversal Using Relays around NAT (TURN) is leveraged. TURN, as defined in RFC 5766, allows for a host to use the services of an intermediate node, such as Cisco Expressway, to provide a relay point for media streams so that multiple hosts can communicate using a single relay address.

As shown in Figure 11-25, when ICE is enabled, and two different MRA users would like to call each other, there are different options for negotiating the media paths between UC endpoints:

- **Standard media path:** Calls are anchored on Expressway-C.
- **TURN server media path:** Calls are anchored on Expressway-E.
- **Peer-to-peer media:** Media flows directly between ICE compatible endpoints.

https://t.me/learningnets

@cisco_in_persian_channel

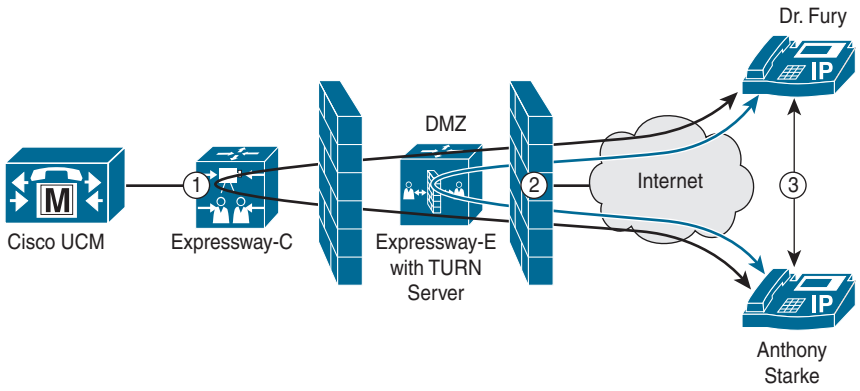


Figure 11-25 *Media Flows Between Mobile and Remote Access Clients with ICE and TURN Services Enabled*

When ICE is used with MRA, a VoIP call must be encrypted from end to end. As previously discussed, Cisco Jabber clients can achieve the end-to-end encryption requirement that is needed by leveraging SIP OAuth with UCM clusters that are either in nonsecure or mixed-mode configurations.

ICE requires Cisco Unified CM 12.5(1) and Cisco Expressway X12.5 or later. Cisco IP phones that support ICE include the 7800 and 8800 series of IP phones running 12.5(1) SR2 firmware or later, Cisco Jabber clients using version 12.5 or later, and Telepresence devices that use CE software that is 9.6.1 or later, where ICE configurations can be provisioned through UCM. For additional information about ICE, review the MRA through Cisco Expressway Deployment guide.

Defending Against Attacks at the Edge

Because the results of an attack against UC infrastructure can be so damaging, it is important to have the mentality that an attack will happen. Former Cisco CEO John Chambers once said, "There are two types of companies: those that have been hacked, and those who don't yet know they have been hacked." Specific attacks that an organization should focus on preventing at the edge of the network typically include TDoS and toll fraud, but certainly there are other types of more complex attacks to be aware of also.

One specific use case that organizations should be aware of is when they have Expressway used for B2B connectivity to support open video federations. A benefit of having an open video federation is that it simplifies B2B connectivity and provides organizations with business agility. Technically, this is because it allows organizations to establish communications without any predefined peering agreements. The only technical requirement is that a business partner is using standards-based SIP video

https://t.me/learningnets
@cisco_in_persian_channel

infrastructure. An attack against this type of environment involves scanning and enumerating the network (e.g., establishing connections to Expressway in order to discover potential attack vectors). Potential attack vectors include the ability to conduct toll fraud or to effectively turn Expressway into an open SIP relay that consumes an organization's bandwidth, licensing, and compute resources. To protect against this type of attack, you can leverage call policy rules to provide a layer of security to the dial-plan configured on Expressway. As an example, call policy rules can prevent unauthorized access to PSTN gateways, conference bridges, voicemail systems, and so on and also ensure that calls that come in from an Internet connection do not go back out to unauthorized locations.

There are two different approaches to using Cisco's Call Processing Language (CPL) rules to block unwanted calling attempts from the Internet. Similar to network-based ACLs, these design options include

- **Allow-based policy:** This policy allows calls only if they match the numeric range or the alphanumeric URI format for an internal domain that belongs to users, meeting rooms, or conference systems. Similar to network-based access control lists, which have a Deny IP Any/Any at the end, the last CPL rule blocks all other inbound calls.
- **Deny-based policy:** This policy denies calls to specific services such as PSTN gateways, conference systems, and voicemail. Everything else that matches a destination within the corporate domain is allowed.

Because CPL rules are processed in order from the top down, organizations can use a blended approach to block prefixes that may be used to call out to the PSTN, while allowing inbound calls to internal systems. An example of call policy rules that support this approach from unauthenticated users is shown in Figure 11-26.

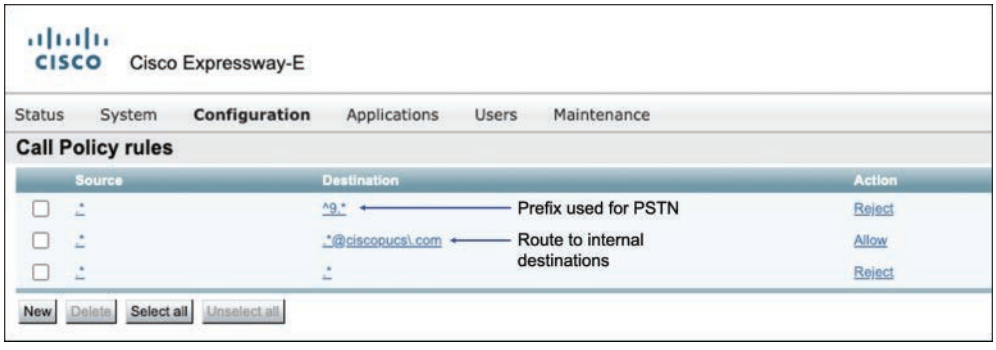


Figure 11-26 *Use of Call Processing Language Rules to Protect Against Toll Fraud and Use of Expressway as an Open Relay*

https://t.me/learningnets
@cisco_in_persian_channel

The use of transforms and search rules is another layer of dial-plan security that can be applied on Expressway to prevent toll fraud. Because a call cannot be routed without matching a search rule within Expressway, search rules can be used to prevent routing calls inbound to Unified CM. Last but not least, UCM can leverage Calling Search Spaces (CSSs) to prevent unauthenticated call attempts from being routed. To use CSSs, you need to develop an ordered list of resources that a device or trunk should have access to. Within Unified CM, these resources are assigned specific partitions. A CSS within Unified CM should allow access to corporate directory numbers, directory URIs, and conference resources. To prevent toll fraud, the inbound CSS should not have access to a partition that belongs to route patterns that could be used to route an inbound call to the PSTN.

The best solution for defending against attacks is to provide defense in depth. An example of the layers of security that we have discussed in this chapter, starting with ACLs to defend against attacks originating at the edge of the network, is included in Figure 11-27. The next section provides more detail about securing B2B connectivity.

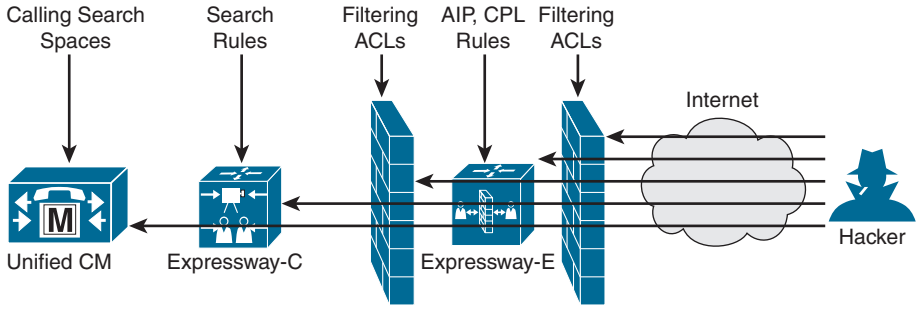


Figure 11-27 *Security Features That Can Be Enabled in Layers Across the Collaboration Edge Architecture*

B2B Connectivity

When using Expressway to provide B2B traffic for specific business partners, organizations can use dedicated neighbor zones to establish secure connectivity. This type of design is different from having an open federation for anyone to use for inbound calls because organizations can work with their business partners to secure their connection to the Expressway environment. Leveraging neighbor zones, organizations can apply another layer of security by configuring TLS and TLS verification to authenticate and encrypt media between locations, thus eliminating attacks involving privacy concerns such as eavesdropping and session replay attacks. An example of a secure neighbor is shown in Figure 11-28.

https://t.me/learningnets
@cisco_in_persian_channel

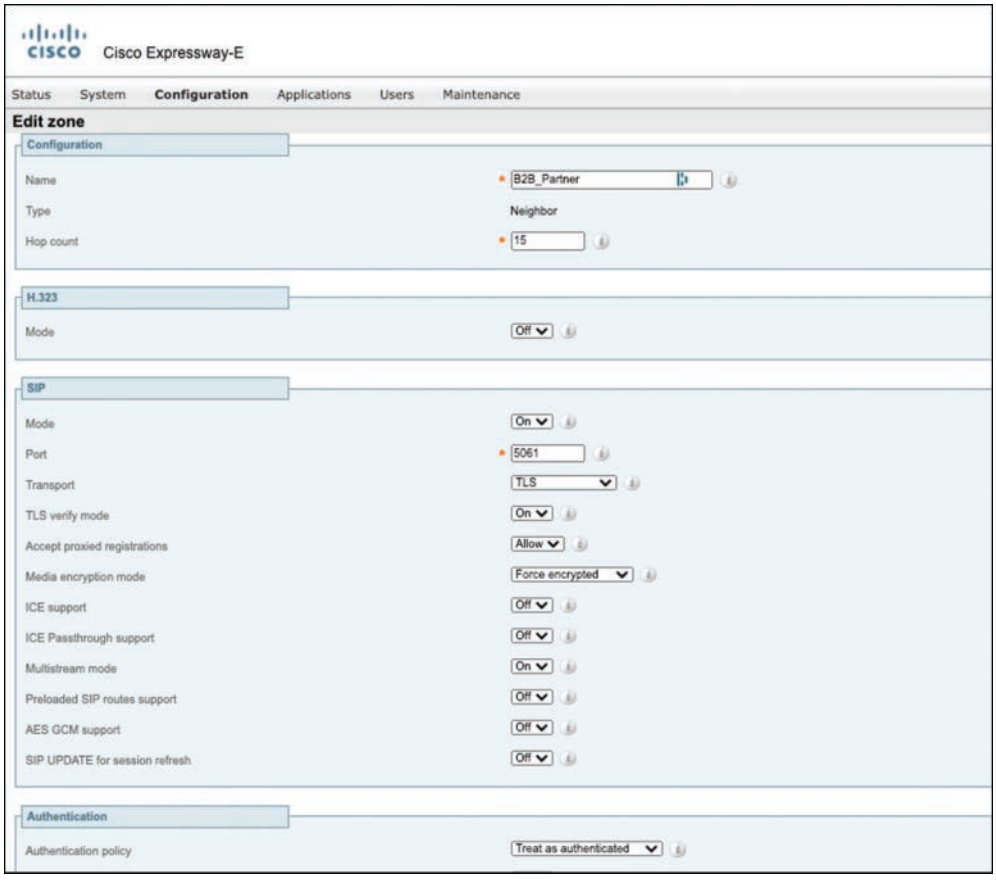


Figure 11-28 *A Secure B2B Neighbor Zone Configured on Expressway-E*

Monitoring and Compliance

A common theme that we have mentioned starting with Chapter 1 is the need for visibility of traffic to monitor for suspicious events and to proactively detect security issues. Call processing systems such as CUBE, UCM, and Expressway all provide options for monitoring call details in real time and also to export call detail records (CDRs) to external databases for compliance purposes. In IOS/IOS-XE-based systems, we have already discussed use of NBAR to provide visibility. Additionally, Cisco recommends use of an external authentication, authorization, and accounting (AAA) server by adding the relevant configurations to support RADIUS. UCM and Expressway both provide options to collect CDRs and export them to external systems. UCM currently supports the ability to export CDR data to three servers, while Expressway supports the ability to export to four.

https://t.me/learningnets
@cisco_in_persian_channel

When CDRs are exported into systems such as Splunk that can detect anomalies and then provide alerting, you do not have to worry about sifting through logs or generating reports to check for unusual activity. Unfortunately, many customers that do not have the ability to export CDR data do not take the time to sift through logs because the process is often too time-consuming, and they may therefore be more vulnerable to financial loss due to toll fraud attacks. Additional benefits that organizations may gain from exporting CDR data to external systems include the ability to generate billing reports, obtain diagnostic information about calls, and determine the quality of the call (e.g., jitter, latency, lost packets). Figure 11-29 shows how to export CDR and system metrics from Expressway.

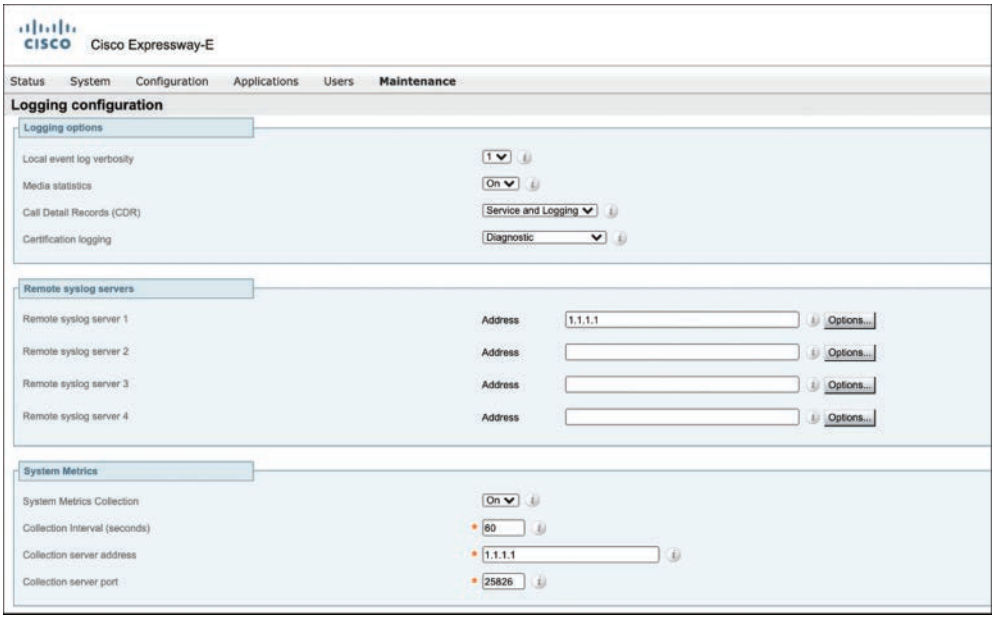


Figure 11-29 *An Expressway Configuration to Export CDR Data to an External System*

Summary

An effective UC architecture should consider the types of services that employees need to be productive. With the growth of telework and business-to-business communications, organizations should consider deploying services at the "Collaboration Edge," to provide users with the services they need to be productive. With that said, cybersecurity attacks are on the rise against traditional IT systems as well as UC systems. Organizations should adopt a mindset that it is not a matter of *if* they will ever be attacked but a matter of *when*. Therefore, when deploying UC services at the edge, organizations should take care

https://t.me/learningnets
@cisco_in_persian_channel

to secure the edge by enabling features that protect against the types of attacks most likely to occur and to mitigate the amount of damages or financial loss incurred.

Cisco offers many features that are included inside solutions such as Cisco Unified Border Element and Expressway that can be used to secure connectivity to the IP PSTN and for mobile and remote users. The most practical security solutions involve disabling unnecessary connections and protocols so that the productivity of the user community is not negatively impacted. The most common types of attacks that occur at the edge of the network originate from attackers who are attempting to enumerate systems and services for financial gain in the form of toll fraud attacks or ransom in the form of TDoS. Although less prevalent, some attacks can occur in the form of revenge.

Cisco provides numerous security features that can be enabled in layers to help organizations provide UC services in an agile, secure, and robust manner to defend against these attacks. Now that you are more aware of the risks of these types of attacks and understand some of the techniques that can be used to protect against the attacks, you should have the confidence to support growing business needs and industry trends.

Additional Resources

https://tools.ietf.org/html/rfc5246

https://tools.ietf.org/html/rfc4492

https://tools.ietf.org/html/rfc1918

https://tools.ietf.org/html/rfc4732

https://tools.ietf.org/html/rfc5245

https://tools.ietf.org/html/rfc8445

https://tools.ietf.org/html/rfc5389

https://tools.ietf.org/html/rfc5766

www.cisco.com/go/cube

www.cisco.com/go/expressway

https://t.me/learningnets

@cisco_in_persian_channel

This page intentionally left blank

https://t.me/learningnets

@cisco_in_persian_channel

Chapter 12

Securing Cloud and Hybrid Cloud Services

The convenience of consuming cloud and hybrid cloud services has many organizations interested in transitioning to their use. The purpose of this chapter is to help organizations that are interested in making the transition understand what mechanisms are available so that they do not have to trade the confidentiality, integrity, and availability of data and services for the convenience of a cloud platform. In a recent survey conducted by International Data Corporation (IDC), 80 percent of IT decision makers said that they have completed or are in the process of a cloud repatriation process. This means that their organizations are moving applications and associated data out of public cloud environments and back to an on-premises location, such as a data center. One of the biggest reasons for this repatriation is lack of security. To be fair, not all of the repatriated workloads are UC related, but this is something that an organization needs to consider beforehand because undergoing a repatriation can be very costly in both time and resources, not to mention embarrassing when things don't go as well as planned.

By the end of this chapter, you should be more comfortable with the security controls that are provided by Cisco when moving UC workloads to a cloud/hybrid cloud environment. Additionally, the reader should be able to help their organization ensure that cloud-based UC solutions have been architected with security and protection of data in mind. By considering security up front, an organization will have increased confidence that they will not have to trade security or the risk of data loss for the convenience and/or business benefits of cloud. To increase confidence, we first will need to address the general security considerations that an organization should make when moving UC workloads to Cisco's cloud and hybrid cloud services. Because many organizations have differing business and security requirements, we then discuss the protections that are enabled by default and also highlight various security features that can be enabled to help increase confidentiality, integrity, and availability of the UC environment.

https://t.me/learningnets

@cisco_in_persian_channel

ACME's Cloud Initiative

To take advantage of innovative new technologies such as artificial intelligence (AI) and facial recognition, ACME leadership has developed a cloud initiative that includes consumption of UC services from a cloud environment. As part of this initiative, the company would like UC administrators to help them understand the risks and benefits that they are faced with if they decide to transition a subset of ACME's workforce and conference rooms to cloud platforms. ACME leadership would also like to explore the capability of rapidly scaling up UC services from off-premises environments, such as the cloud. This initiative is a result of a recent ACME acquisition that specializes in widget manufacturing and the need to onboard new staff, some of which are responsible for providing widget support as teleworkers. As a secondary goal, ACME wants to know the implications for moving directory services to a cloud environment, such as Microsoft Azure, so that the company can provide centralized authentication to users who sign in to ACME's domain.

ACME's leaders understand the business implications if any customer data is lost or compromised, so they are looking for solutions that provide protection of data while in transit and while at rest. They are also interested in how to ensure compliance for users and devices using cloud platforms. The legal and security team believes that this will help minimize any objections to their customers using workloads that are contained in cloud environments and, if necessary, help articulate the security controls that maximize protection of their data.

Although some UC administrators at ACME are concerned about their job security, Anthony Starke is excited about this initiative. Anthony believes that by integrating ACME's on-premises UC architecture to cloud services, ACME will benefit from the innovation provided by cloud environments and see an increased level of productivity, which will take some of the stress away from him and his UC team. As part of this increased productivity, Anthony believes that his boss Jonah and the rest of ACME leadership will recognize what he and his team have been working on so diligently. Personally, Anthony hopes that he will also be recognized for the vision that he has provided for designing a Unified Communications architecture that supports cloud services. Anthony is also hopeful that ACME leadership will provide increased investment into his UC infrastructure by replacing some of the older conference room systems with some of the latest models.

ACME leaders have asked Anthony to report back to them within a week with a plan to brief them on the appropriate security controls that can be implemented to ensure privacy if and when they decide to transition some of their UC services to a cloud environment.

Business Drivers for Cloud and Hybrid UC Services

Before addressing the methods available to provide security and compliance for organizations consuming UC services from a cloud environment, we must provide additional detail behind the trend of moving UC workloads to the cloud. The market for UC services

https://t.me/learningnets

@cisco_in_persian_channel

is shifting from on-premises-based infrastructure to cloud-based UC-as-a-Service (UCaaS) offers based on various business drivers and use cases such as

- Reduced capital investments to consume services
- Ease of access to UC services for teleworkers
- Offloading management of UC services
- Simplified security
- Innovation and integration with third-party environments

If we take a closer look at each of these use cases, they all provide a unique value to organizations. Further, it has been proven that when end users and teleworkers have access to the services that they need, their communications are more effective, and ultimately, they are more productive.

Perhaps contrary to the belief of many information assurance specialists within an organization, a cloud UC environment may provide better security than their on-premises UC environment. A primary reason for this is that by enabling security on-premises, an organization can create management complexities. For this reason, many organizations simply choose to leave many of the available security features disabled. Similarly, an additional perspective is that an attack against an organization's UC infrastructure is not an immediate threat; therefore, projects that require enabling security get de-prioritized by an organization. By now, everyone should believe that an assumption that an organization isn't vulnerable or prone to attacks on UC infrastructure is false. Further, when an attack does occur, the results can be disastrous.

When consuming a cloud service, organizations are able to offload the burden of potentially adding management complexity when enabling security features. Also, the burden to secure UC services from attackers belongs to the cloud service provider (CSP). By offloading management and security concerns, organizations can focus on other critical areas such as increasing the user experience, application integration, user adoption, application development, compliance to security standards and so on. Consuming UC services from the cloud as a subscription can reduce the out-of-pocket expenditures that an organization needs to make, thus also allowing the organization to reduce the risk of costs or projects that exceed expected cost in order to maintain and/or secure UC services while investing in other areas of the business.

Currently, Cisco has several publicly available cloud-based UC platforms, including Webex, which offers cloud calling, cloud messaging, cloud meetings, and a cloud contact center. Additional cloud offers include Unified CM cloud, Hosted Collaboration Services (HCS), and HCS for Government. The delivery models for these services can be grouped into these three main categories:

- **Native cloud:** UC services hosted from a cloud environment that possess characteristics of multitenancy, microservices, and the ability to easily integrate with other cloud platforms.

https://t.me/learningnets

@cisco_in_persian_channel

- **Hosted:** UC services hosted from a managed service provider's (MSP's) location. UC applications in this environment are often based on traditional on-premises architectures.
- **Hybrid:** UC services that integrate an organization's on-premises infrastructure with a native cloud environment.

For the purposes of this chapter, we focus on Cisco's Webex platform because it provides both native and hybrid cloud architectures that can be integrated with on-premises infrastructure. The Webex platform also represents an area in which many integrations are taking place with other cloud environments such as Microsoft 365, Google G Suite, and additional cloud platforms that provide content management. Areas in which rapid innovation is taking place with cloud UC include integration with these other types of cloud environments and the applications inside of them. In many cases, these integrations create an environment for an organization to integrate with business applications to provide business intelligence, to use virtual assistants in order to simply workflows, and/or to integrate with other cloud security solutions to further protect an organization's data.

Coordinating for Governance and Compliance

When an organization makes the decision to consume UC services from a cloud service provider, the burden of providing stability, maintenance, and secure access to services is on the CSP, such as Cisco. As we have discussed previously, different levels of security can be provided to any UC environment. Native cloud environments provide dozens of security features to mitigate various types of threats for the services that they provide. Because they are often developed and deployed independently of each other, there are little dependencies of creating second- or third-order issues when security features are enabled. This type of service-oriented platform allows for continuous innovation, integration, and deployment of advanced features in a custom yet secure fashion.

While CSPs provide the necessary security controls, organizations focused on security should implement governance and oversight to ensure that the right security features are turned on and also to make sure that they are in compliance with legal requirements surrounding data protection and data retention. The primary services that Cisco provides (and which we focus on) from the Webex platform include calling, messaging, and meetings. To provide security across these services, Cisco has developed a multilayered security model based on data center security, application security, and security established across the development life cycle. By doing this, Cisco aligns to defense-in-depth principles.

Contrary to the opinion of many, a product or cloud offering is not secure without the proper organizational model in place to provide governance, oversight, and compliance. Cisco's organizational model for providing governance, guidance, oversight, and compliance management as a CSP to its cloud platforms is a coordinated effort that involves

https://t.me/learningnets

@cisco_in_persian_channel

multiple various teams that are focused on security. These teams include but are not limited to the following:

- Security and Trust Organization (STO)
- Cisco's Information Security (InfoSec) cloud team
- Product Security Incident Response Team (PSIRT)

STO is chartered to provide guidance on the procedures, processes, and security tools to ensure security, privacy, trust, and resiliency to Cisco's customers. Cisco's InfoSec cloud team works with Cisco IT to ensure that the products it builds and the infrastructure it operates are secure. The PSIRT investigates vulnerabilities across the cloud UC portfolio and then assigns impact levels while also coordinating resolutions to the issues and transparently communicating the threats to consumers. Just as Cisco has created these dedicated security teams, the most successful organizations across the industry have adopted a similar operational model.

As we discussed in Chapter 1, "The Importance of Practical UC Security," a key aspect of security is availability. Webex Calling targets 99.99 percent availability, which is equal to less than five minutes of downtime per month. To provide security while maintaining 99.99 percent availability, Cisco leverages a Secure Development Operations (SecDevOps) approach for providing continuous penetration testing with internal and external teams. As a CSP, Cisco is also certified and maintains compliance with industry security standards. Following are some of Cisco's certifications and a brief explanation of their purposes:

- **International Organization for Standardization (ISO) 27001 / 27017 / 27018:** These certifications ensure that an Information Security Management System (ISMS) is in place. The goal of the ISMS is to identify and analyze information security risks and then mitigate them.
- **SOC 2 type II and SOC 3:** These certifications ensure that CSPs have "System and Organizational Controls." The purpose of the SOCs is to ensure CSPs are handling customer data in a safe and secure manner.
- **Federal Risk and Authorization Management Program (FedRAMP) Moderate:** Certified for Webex Teams, Webex Meetings, and Unified CM Cloud for Government services. The FedRAMP certification program provides authorization for federal agencies to consume services from a cloud environment. FedRAMP is a governmentwide program that ensures a standardized approach to security assessment, authorization, and continuous monitoring for cloud products and services.

These certifications show that Cisco is committed to providing organizations with highly available and secure services. These certifications also show the broad and balanced approach that Cisco has taken to align with security concerns and policies of their customers.

https://t.me/learningnets

@cisco_in_persian_channel

Transport Security and Compliance

The Cisco Webex architecture is designed to be secure while allowing organizations to choose from different transport options for how UC services can be consumed and routed. As an example, Cisco provides organizations (and associated endpoints) with an option to connect over an insecure Internet transport using SIP with TLS security. This approach is commonly referred to as over-the-top (OTT) connectivity. To increase reliability and security, Cisco has also developed another approach that has been optimized for real-time media that is called the Webex Edge Connect architecture.

Leveraging Cisco's global backbone network for Webex and Cisco's Webex Edge Connect architecture, organizations have the ability to bypass an insecure transport such as the Internet while also improving the user experience while securely routing media to Webex Meetings environments. This option essentially provides organizations with a private transport to Cisco's Webex cloud over private and secure connections. To provide organizations with this private transport, Cisco has chosen to leverage Equinix's cloud exchange to directly peer with Cisco Webex data center locations and customer sites. With this option, organizations increase business continuity options by allowing end users and devices to consume calling services from Cisco, join meetings from either traditional, cloud registered UC endpoints, and/or also the Webex Meetings application while in the office or working remotely. If desired, OTT connectivity can be used as a backup connection to private transport connectivity options. Lastly, Cisco provides hybrid options that we will later discuss regarding anchoring and routing media locally so that organizations can take a hybrid approach to securing connectivity.

Many of Cisco's endpoints and applications leverage HTTPS channels to communicate securely with Cisco Webex cloud services. To comply with security policies, many organizations are required to route web-based traffic through a web proxy and/or firewall before leaving an organization. These proxies typically perform security functions such as inspection of traffic, URL blacklisting (deny traffic), and URL whitelisting (permit traffic). Cisco's endpoints and applications can be configured to use web proxies in different ways; they include

- **Automatic Configuration:** Using Web Proxy Auto Discovery (WPAD); Proxy Auto Config (PAC) files
- **Manual Configuration:** Configured via the Operating system or GUI

When using WPAD, web browsers use DHCP and DNS to locate a Proxy Auto-Config (PAC) file automatically. DHCP servers must be configured to support DHCP option 252 in order to obtain a URL that specifies the exact location of the PAC file. DNS is used to resolve the URL so that a PAC file can be downloaded. Otherwise, modern web browsers can be configured to point to the location of a PAC file.

Many organizations require authentication to proxies in order to block requests to content until a user can provide valid credentials to display that they are authorized access to the content. Cisco endpoints and applications support authentication to proxies using the following authentication mechanisms: basic, digest, Windows NTLM, Kerberos, and

https://t.me/learningnets

@cisco_in_persian_channel

negotiate (Kerberos with NTLM fallback). When proxy authentication is being used, valid credentials must be configured in the operating system of the Webex device/application. Since authentication to proxy servers is not mandatory, many organizations choose this option, which is also supported by Cisco. When using proxy servers without authentication, there are security benefits with how network firewalls are configured. As an example, an administrator can create an object group for the IP address of the proxy server(s) and easily create ACLs that permit web traffic that originates from the web proxy as it traverses a firewall and routes out to the Internet.

When using firewalls to filter traffic destined to the cloud, filtering based on IP addresses that may be used for signaling traffic is not supported. This is because IP addresses used for signaling are dynamic and may change. Organizations that have strict ACL policies should use URL filtering techniques instead. Similarly, if using proxy servers to whitelist traffic, organizations need to verify that they have access to URLs used for Cisco's Webex cloud services. Cisco's ecosystem partners (Amazon Web Services and Microsoft Azure) have reserved IP subnets for media traffic, so if necessary, ACLs may be used to permit/deny media traffic. Cisco recommends use of UDP to support real-time media, in alignment with RFC 3550. Because Cisco Webex platforms support TCP for media as a fallback mechanism to UDP-based media streams, care should be taken that proxy servers do not intercept TCP media traffic. If this occurs, the user experience may be degraded based on the performance of the proxy server. For more information about configuration of proxy servers and firewalls, including a list of URLs that Webex services use, please review the Cisco Network Requirements for Webex Teams Services article.

Now that we've discussed Cisco's overarching approach for providing governance, compliance, and basic connectivity for its cloud environment, we can move to the services that Cisco provides from a cloud and hybrid cloud architecture.

Considerations for Secure Calling

The Cisco native cloud calling solution, Webex Calling, is a platform that provides registration for phones and soft clients (Webex Teams). Webex Calling supports internal calling and outbound calling to the PSTN. Cisco's vision for calling from a cloud environment is "Cisco cloud first, not cloud only." Therefore, Cisco allows organizations to decouple PSTN connectivity from the Webex Calling service. This means that organizations can connect to the PSTN from a local gateway, at their location, or from a cloud-connected PSTN (CCP) partner. CCP partners include Intelepeer, MNF Enterprise, NTT U.S., NTT EU, OneStream Networks, and Veracity Networks. Webex Calling does provide E911 services for emergency calling through integration with Redsky.

The Webex Calling service is provided by a value-added reseller (VAR). This means that additional architectural options may exist based on how the VAR delivers the Webex Calling service to the customer. An example of a value-added service may be to host a CUBE SBC in the VAR's data center to provide secure connectivity to the Webex Calling service. VARs also have access to Cisco Webex Control Hub, which is a web portal that Cisco provides so that a VAR and/or an organization can manage its cloud and hybrid

https://t.me/learningnets

@cisco_in_persian_channel

cloud services. Inside Control Hub, administrators can configure services, onboard users, onboard devices, assign user privileges, enable security features, gather analytics, generate reports, and troubleshoot their environment. You can access Control Hub by navigating to https://admin.webex.com. We will be discussing how to use Control Hub to provide functionality as it pertains to security in the upcoming sections. An example of how Control Hub can be used for both functionality and security is by onboarding a local gateway (CUBE) within Control Hub. When a local gateway such as CUBE is used, it is possible to provide SIP-based interworking with existing on-premises calling infrastructure, such as Unified CM, and also to provide security connectivity to the Webex Calling service. A diagram to show the Webex Calling architecture is shown in Figure 12-1.

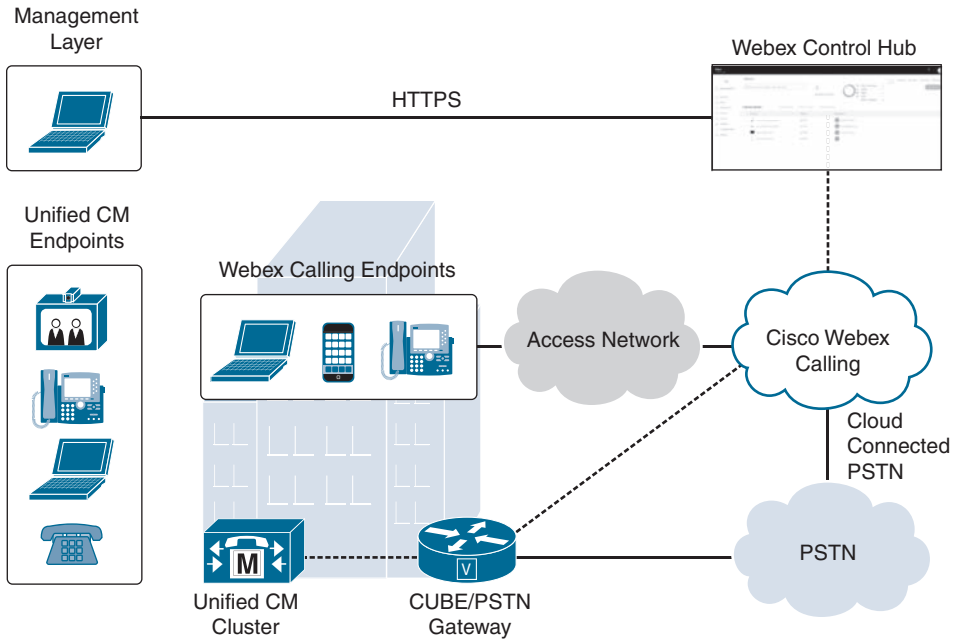


Figure 12-1 *High-Level Architecture of Webex Calling Service*

After a local gateway has been added from Webex Control Hub, you can establish a SIP trunk to the Webex Calling environment. With Webex Calling, all SIP trunks between a local SIP gateway and the access SBC located with the Webex data center are always secure. Specifically, SIP trunk connections are encrypted with TLS 1.2 ciphers to ensure privacy of UC signaling traffic destined for the Webex Calling service. To simplify how an organization provisions a SIP gateway, configurations for local gateways are provided by Cisco.

These configurations are very specific because mutual TLS connections between local gateways and Webex Calling must be established. As an example, Webex Calling supports only router platforms that support IOS-XE due to the configurations needed for security and registration to the Webex Calling service. Specifically, a CUBE running IOS-XE must be updated with the Cisco CA root bundle in order to validate the certificate presented by

https://t.me/learningnets
@cisco_in_persian_channel

the Webex Calling access SBC. CUBE also needs a set of SIP digest credentials (provided by Cisco) in order to register to the Webex Calling environment.

Media is always encrypted for Webex Calling registered devices. Endpoints that use Webex Calling employ AES-128 SRTP encryption in order to provide confidentiality, and SHA1 generates a keyed hash message authentication code (HMAC) for each encrypted message to provide integrity and authenticity. Additionally, video calls, screen sharing, call recordings, and voicemails are all protected by AES-128 media encryption. These safeguards are helpful in securing any environment to protect against snooping or replay attacks.

One point of interest is that media must be decrypted and then reencrypted for call routing between call legs by default because each call leg uses a unique pair of keys for a media stream. The ability to decrypt and then reencrypt traffic provides benefits, such as the ability to provide transcoding, network-based recording, and routing to the PSTN. For some customers, this is a risk because it increases the chances that an attacker could exploit the cloud environment and use a man-in-the-middle attack to gain access to a conversation. In later sections we will discuss which services support end-to-end encryption.

For a local gateway to connect to the Webex Calling service, perimeter firewalls need to allow outbound connections to the service so that the service can provide organizations with optimal functionality. Some connections to the Webex Calling service use different destination ports than what are typical with on-premises UC services, such as TCP 8934 for SIP-TLS.

A port reference for connecting to the Webex Calling service is provided in Table 12-1. All other transport that is not listed uses HTTPS (TCP 443) to protect connections to and from the Webex Calling service. Figure 12-2 depicts the secure connections between a local gateway and the Webex Calling service.

Table 12-1 *Destination Ports Used to Connect to the Webex Calling Service*

Purpose	Source	Protocol	Destination Port
Signaling (SIP-TLS)	Devices, Applications, Interface of PSTN Gateway	TCP	8934
Media (SRTP)	Devices, Applications, Interface of PSTN Gateway	UDP	19560–65535
Device configuration and firmware upgrades	Calling Devices	TCP	80, 443
Cloud upgrader	Calling Devices	TCP	*443, *6970
Application configuration	Applications	TCP	80, 443, 1081, 2208, 8443, 5222, 5280–5281, 52644–52645
Calling Scan (CScan)	Devices	TCP	80, 443, 8934, 19569–19760

*Only used when migrating from Enterprise firmware on UC endpoints to Webex Calling firmware

https://t.me/learningnets

@cisco_in_persian_channel

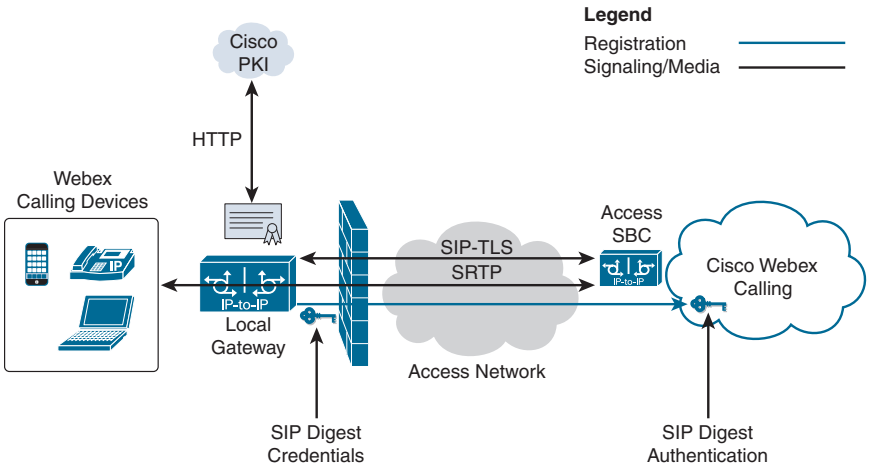


Figure 12-2 *Secure Connections Between a Local Gateway, Endpoints, and Webex Calling*

To provide troubleshooting assistance for network-related issues, Cisco provides customers with the CScan utility to identify issues connecting to the Webex Calling platform. You can access the CScan utility by browsing to https://cscan.webex.com/. The CScan utility provides insights into whether the necessary network ports are open and whether the network meets Cisco's requirements for bandwidth, latency, packet loss, and jitter. Sample test results provided by the CScan utility are shown in Figure 12-3.

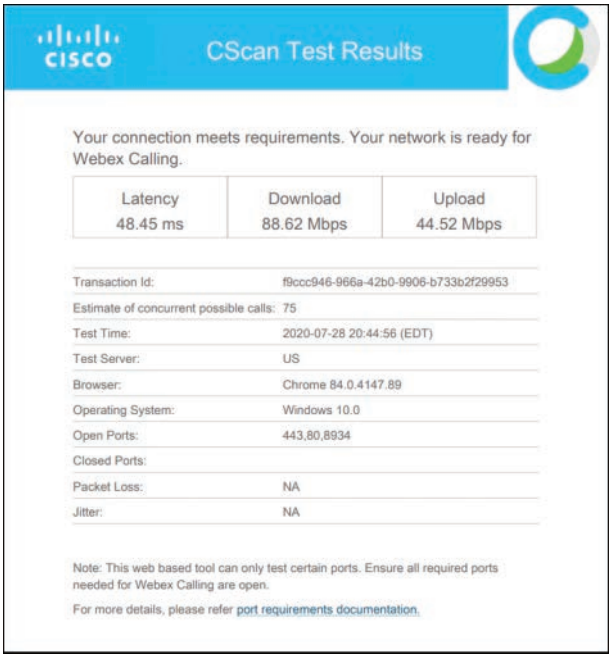


Figure 12-3 *Report from CScan Test Utility*

https://t.me/learningnets
@cisco_in_persian_channel

The Webex Calling service bundles calling together with the Webex Teams client for messaging with the option to add meetings so that organizations can consume a complete UC solution. The next few sections discuss critical aspects of security, such as identity when using software clients such as Webex Teams to access Webex Calling services. More important than the identity itself is how users authenticate to the Webex platform and how they are authorized to access services.

Who's Who and What Privileges?

One of the biggest security threats that organizations are faced with is identity theft. In 2019, Microsoft revealed that over 44 million users of its Azure and Microsoft Service accounts were using credentials that were leaked to the public. In the same year, Facebook databases containing 419 million user accounts were discovered and leaked to the world, representing a tremendous security risk for those Facebook users whose usernames and phone numbers were listed within the accounts.

Within Cisco's Webex platform, a flexible built-in identity service provides organizations (and its users) secure access to services to help mitigate identity theft concerns. By default, local usernames and passwords are used for authentication and authorization. If organizations have security requirements beyond username and password authentication (to mitigate the risk of identity theft), Webex supports single sign-on (SSO) and/or multifactor authentication (MFA) to strengthen authentication mechanisms. Single sign-on is possible by integrating with third-party identity providers (IdPs) and using the Security Assertion Markup Language (SAML) 2.0 protocol to provide authentication with on-premises or cloud-based directory services.

In the next section we explain how the Cisco Webex platform is able to securely integrate with local directory servers, such as Active Directory, and how the Webex identity service leverages modern protocols that have helped strengthen authentication. Specifically, we discuss SAML 2.0 and Open Authorization (OAuth) 2.0, which is used for authorization. As we have discussed previously, these protocols are commonly supported with on-premises environments but are also commonly used with cloud-based platforms.

User Onboarding and Role-Based Access

As we discuss identity, it makes logical sense to discuss the different types of options and protocols that organizations can use to create a simple but secure way of user authentication and authorization into Cisco's Webex environment. This also includes provisioning of users and the user onboarding process. In total, there are five ways to provision users into Control Hub:

- Manually
- Using a CSV template
- Using Webex APIs
- Synchronizing from Active Directory (AD) using Directory Connector
- Synchronizing from Azure AD and Okta using SCIM

https://t.me/learningnets

@cisco_in_persian_channel

After users are imported or provisioned inside Control Hub, they can be assigned to different services and also be assigned various privileges. For the subset of users that need administrative privileges, role-based access controls can used to limit privileges based on which administrative rights are needed. By default, Control Hub provides organizations with a predefined set of roles, each with different various privileges. This role-based access allows organizations to provide a layer of security by assigning roles that are based on least privilege to be able to perform duties and responsibilities for the organization.

Some of the roles and associated privileges are as follows:

- **Device Administrator:** Is responsible for device management
- **Support Administrator:** Has access to analytics, reports, and troubleshooting tools
- **User and Device Administrator:** Is responsible for adding/deleting users, device management
- **Full Administrator:** Is responsible for adding/deleting users, managing devices, assigning roles/licenses, managing devices, handling data retention policy, and configuring security settings
- **Read-Only Administrator:** Has read-only access to administrator tabs
- **Compliance Officer:** Has access to user-generated content and the ability to put user-generated content into a legal hold state

For a full listing and explanation of these roles, you can research user roles in the Webex Control Hub. This next section will cover use of synchronizing a local directory into the Cisco Webex identity service using Directory Connector.

Directory Connector

Many organizations must maintain a local directory for various reasons such as 802.1x authentication of local endpoints into the network. Additionally, it is not always desirable for organizations to have to manage separate directories located in separate environments (local and cloud). Integration of local directory services allows organizations to maintain a single directory structure when moving UC workloads to the cloud. Integration with local directory services also allows organizations to leverage their existing investments when moving UC services to the cloud. Cisco's Directory Connector application can be used to securely import user credentials from a local directory server into Control Hub for use with the Webex identity service. From a user perspective, this capability helps simplify the experience because it allows users to use the same username and password when authenticating into the network locally or when accessing cloud resources. It also ensures that all users and Webex devices are searchable from the cloud to help simplify how devices can be called.

Directory Connector software can be downloaded from the Cisco Webex Control Hub and installed on a trusted Windows server that integrates with Microsoft Active

https://t.me/learningnets

@cisco_in_persian_channel

Directory services on Windows server versions including 2019, 2016, and 2012. To provide security for an organization and its associated users, Cisco requires that the account used with Directory Connector be the same account with the same type of administrator privileges to be able to synchronize with a directory server and also in Control Hub. To synchronize user accounts into Control Hub, Directory Connector first uses LDAP or LDAP over SSL (LDAPS) to connect to a local domain controller. After a user with an administrator role authenticates into Webex, Directory Connector uses HTTPS (port 443) with TLS 1.2 encryption to synchronize user accounts into Control Hub, as shown in Figure 12-4. If required, Directory Connector can connect to Webex via a web proxy through one of the following three options:

- Explicit web proxy through Internet Explorer
- Explicit web proxy through a .pac file
- Transparent proxy that works with the connector without any change

As previously discussed, many organizations require proxy authentication. Directory Connector supports NTLM authentication. The following URLs will need to be whitelisted and accessible from the web proxy:

- **cloudconnector.webex.com:** Used for synchronization
- **idbroker.webex.com:** Used for authentication

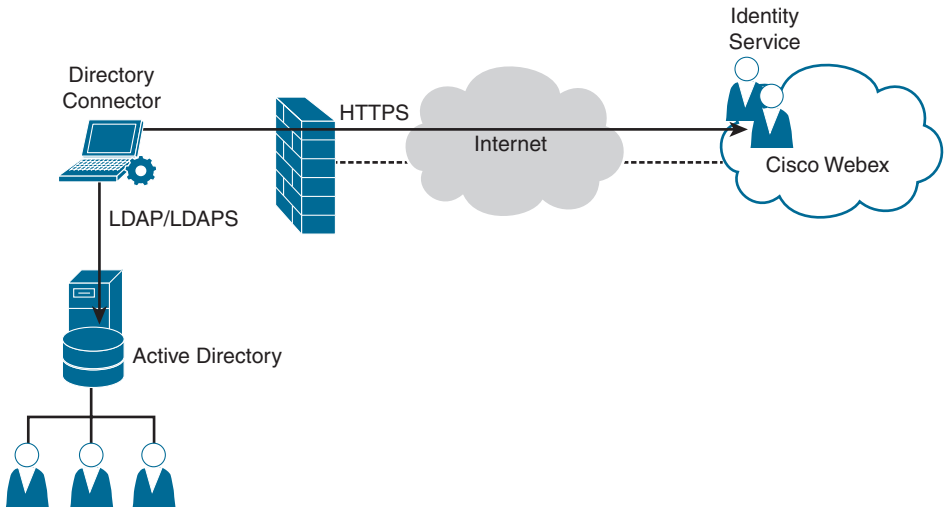


Figure 12-4 *Directory Connector Can Be Used to Import Contacts into Control Hub*

Organizations that desire single sign-on for their users can configure this capability within Control Hub, using a SAML 2.0 integration. The next section explains the authentication and authorization capabilities available for organizations, such as single sign-on using SAML 2.0, and OAuth 2.0.

https://t.me/learningnets

@cisco_in_persian_channel

SAML 2.0

Security Assertion Markup Language 2.0 is an open standard protocol for authentication. The SAML 2.0 standard, as defined in RFC 7522, is an XML-based framework for automating the exchange of identity information across domains and application platforms. The primary purpose of SAML 2.0 in a UC environment is to simplify user authentication across multiple applications and to enhance security by providing single sign-on capabilities. SAML also provides some additional benefits:

- Provides an ability to create a federated identity across domains
- Prevents requirement for reauthentication into services because SAML tokens can be stored within a web browser as a cookie

Cisco supports any IdPs that have been tested for SAML 2.0 compliance to provide SSO integration. At the time of this writing, Cisco has validated nine different IdPs. To enable SSO for Webex within Webex Control Hub, you should browse to Settings > Authentication. Clicking Modify opens an option for integration with a SAML 2.0 IdP; it establishes a trusted relationship between IdP and the Cisco Webex cloud by exchanging metadata from the two services. For the specific steps required to complete a SAML agreement with an IdP, review Cisco's documentation on how to perform single sign-on integration within Cisco Webex Control Hub.

After SSO is configured within Control Hub (and the Webex identity service), an organization is able to provide its users with a single, common set of credentials for Webex Meetings, Webex Teams, and other applications across the organization. Figure 12-5 depicts the protocol interactions between an end user, the Webex service, and a SAML 2.0 compliant IdP to provide SSO authentication.

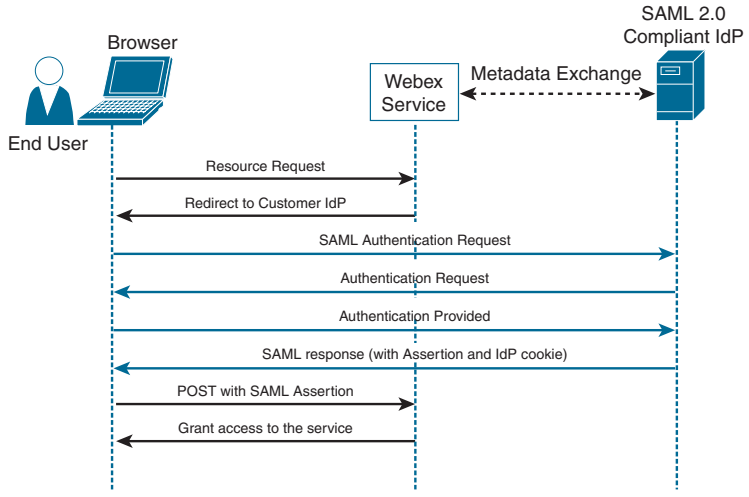


Figure 12-5 *How SAML 2.0 Authentication Works for Webex Calling*

https://t.me/learningnets
@cisco_in_persian_channel

The Webex identity service also provides an option for organizations to use multifactor authentication with or without the need for an IdP. Cisco recommends Duo Mobile and/or Google Authenticator, but Webex supports most MFA solutions. When this service is used with Duo, security policies can be defined and then enforced during login.

OAuth 2.0

OAuth 2.0 is the industry standard protocol for providing authorization to services. The OAuth 2.0 framework, as defined in RFC 6749, "enables a third-party application to obtain limited access to an HTTP service, either on behalf of a resource owner by orchestrating an approval interaction between the resource owner and the HTTP service, or by allowing the third-party application to obtain access on its own behalf."

With OAuth 2.0, authorization to resources is provided by access tokens and refresh tokens. These two types of tokens make API requests on behalf of a user in the following manner:

- **Access token:** Represents the credentials used to access a protected resource
- **Refresh token:** Obtains updated access tokens and continuous authorization to resources

Before OAuth was implemented, if an administrator wanted to allow an integration or web application to access data or resources from another account, that administrator would have to share administrative credentials with the third-party application. This process created security issues, particularly related to password storage and the threat of identity theft because there were limited ways to control how third parties secured the passwords. OAuth not only provides authorization to services but also allows you to revoke a token when access is no longer desired instead of requiring a change in user credentials.

The OAuth 2.0 framework provides a number of different methods that a client can use to gain access to resources through an access token. The most common grant types to resources are as follows:

- **Authorization code:** Used by user-agent-based clients. Supports use of refresh tokens. Uses the Proof Key for Code Exchange (PKCE) extension to ensure that authorization codes are not intercepted. This is the most common type of OAuth grant.
- **Client credentials:** Typically used for application access when a user is not present.
- **Device code:** Typically used by browserless devices to exchange a previously obtained device code.
- **Refresh token:** Used by clients to exchange a refresh token for an access token when an access token is expired.

https://t.me/learningnets

@cisco_in_persian_channel

These legacy grants are no longer recommended:

- **Implicit:** Used with user-agent-based clients but without exchange of intermediate credentials, such as an authorization code for an access token.
- **Password:** Used to obtain an access token with a username and password.

The IETF no longer recommends implicit grants due to security reasons. Simply put, implicit grants are vulnerable to a number of attacks including: token exfiltration, token replay, and man-in-the-middle attacks. Additionally, many services do not support implicit grant types due to the inherent risks involved with the type of flow.

With Cisco Webex Teams, authorization code grant flows are used to mitigate security threats with access tokens and to align with industry best practices. With authorization code grant flows, tokens are kept confidential in transit and storage. Authorization code grants also provide simplicity of use; refresh tokens can be used to obtain new access tokens.

With the Cisco Webex Teams application, OAuth tokens uses a JSON web token (JWT) format. Tokens are signed using JSON web signature (JWS) in accordance with RFC 7515 and are encrypted using JSON web encryption (JWE) in accordance with RFC 7516. By default, access tokens have a duration of 12 hours. When OAuth tokens reach 75 percent of their lifetime (9 hours), Webex attempts to present an OAuth Refresh token to the Webex identity service to get a new access token. Refresh tokens have a duration of 60 days, but the refresh process happens forever, as long as a user logs in once every 60 days. Figure 12-6 depicts the process that OAuth uses to provide continuous authorization to services.

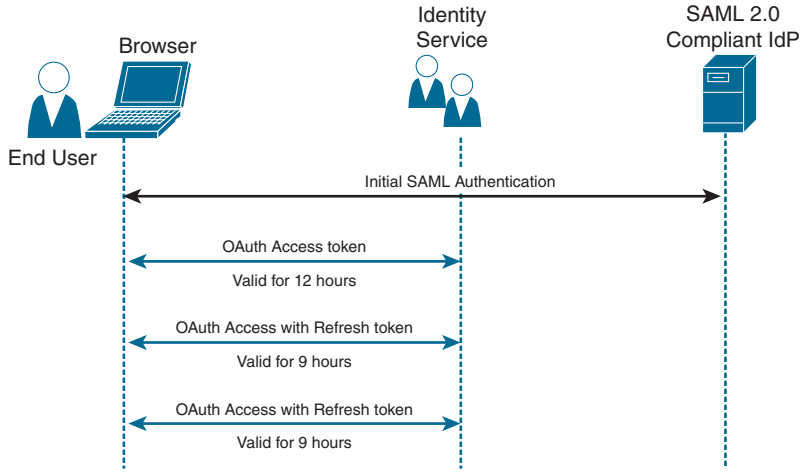


Figure 12-6 *How OAuth 2.0 Provides Authorization to Services*

Although we have described some security features that are built into the OAuth 2.0 framework, organizations need to be aware of some inherent risks. As defined in RFC

https://t.me/learningnets
@cisco_in_persian_channel

6819, there are some additional security considerations beyond the safeguards listed in the OAuth 2.0 specification. As an example, a malicious application could attempt to phish for end-user passwords by misusing an embedded browser in the end-user authorization process. Another attack related to password phishing includes using a counterfeit server to provide unauthorized access to services. This type of attack is possible by DNS poisoning and/or Address Resolution Protocol (ARP) spoofing, which would be instrumental for intercepting a client request and returning misleading or otherwise incorrect responses. These attacks can be mitigated using secure DNS solutions, such as Umbrella, which we discussed in Chapter 3, "Security Through Network Fundamentals." A secondary option is a cloud access security broker (CASB), which understands which OAuth-connected applications are being used and enables you to detect and prevent malicious use. We discuss CASB in more detail later in the chapter. For additional information on OAuth 2.0 specifications, visit https://oauth.net/2/.

SCIM

The System for Cross-Domain Identity Management (SCIM) is an HTTP-based open standard protocol designed for simplifying identity management. As defined in RFC 7644, "SCIM's intent is to reduce the cost and complexity of user management operations by providing a common user schema, an extension model, and a service protocol." This protocol helps simplify identities across multiple domains and multiple cloud environments. Additional information is provided in RFC 7643, which defines the core schema for SCIM. SCIM was created to help complement traditional identity protocols, such as LDAP, which "do not easily traverse firewalls and/or are not easily layered onto existing web protocols." In practice, SCIM allows for automatic provisioning/deprovisioning of users with a REST-based API, between different cloud-based applications and services such as Azure Active Directory, Microsoft 365, Okta, Google G Suite, and so on. When SCIM is used between other cloud-based identity systems, it is useful for linking users in Webex Teams, updating user attributes, deactivating users, and so on.

Device Onboarding

To onboard devices into Control Hub, administrators have the choice of generating a 16-digit activation code or providing the MAC address of a device (IP phone only), which is similar to registering devices to Unified CM on-premises. The 6800, 7800, and 8800 series multiplatform phones and Webex video series devices such as DX, Webex board, Webex room, Webx Share, and others also support use of activation codes for ease of onboarding. IP phones that are onboarded to Control Hub for Webex Calling must be running multiplatform software version 11.0 or higher. If a phone uses Enterprise Unified CM software, Cisco provides a Cloud Upgrader utility that can be used to convert enterprose software to multiplatform software. To access Cisco's Cloud Upgrader utility, visit https://upgrade.cisco.com. For IP phones to use activation codes, they must have a firmware load of 11.2.3 MSR1 or later to display the activation code screen. Video devices and room systems must be on CE 8.3.4 or higher in order to use activation codes. The next section provides more detail about use of activation code security.

https://t.me/learningnets

@cisco_in_persian_channel

Activation codes are used as temporary passwords to authenticate a new device into the Webex Calling service. Activation codes also come with a QR code that can be scanned by devices that have a camera, as shown in Figure 12-7. Activation codes are provided by Cisco's Global Discovery Service (GDS), which is a cloud service that is accessible from Control Hub. This option allows Cisco partners and customers to deliver phones to end-user locations without having to keep track of physical attributes such as the MAC address or prestaging beforehand.

Figure 12-7 *Activation Codes Can Be Entered on Devices or Scanned for Simplicity*

Previously we alluded to the fact that not all Webex Calling devices support use of activation codes with GDS. Only devices that currently support embedded trust anchors to establish a TLS connection with the GDS can use activation codes. Cisco activation codes can be used only once per device and expire after seven days by default. These security measures help prevent man-in-the-middle and brute-force attacks during device activation.

For authentication and authorization into the identity service, Cisco uses a variant of the Diffie-Hellman key exchange algorithm, called password-authenticated key exchange (PAKE). PAKE uses the 16-digit authentication code as a password to establish a TLS connection to Cisco's Collaboration cloud and the Global Discovery Service. After this, endpoints are able to authenticate with the Cisco Webex identity service with the OAuth token and onboard to the Webex environment. A sample flow is shown in Figure 12-8.

https://t.me/learningnets
@cisco_in_persian_channel

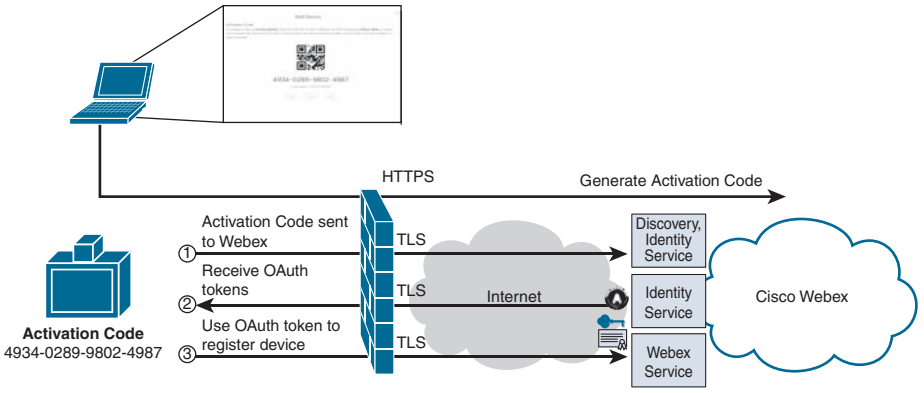


Figure 12-8 *Flow of Onboarding Webex Devices Using Activation Codes*

Once the devices are onboarded into Control Hub, an administrator has visibility into the state of those devices and is able to modify configurations.

As previously discussed, with refresh tokens, a user can reauthorize access to services forever until a device is logged out. In some cases, organizations may determine that a device is no longer trusted, such as when a device is lost, or a device belongs to an employee who does not work for the organization any longer. In this case, Cisco allows administrators to either delete the endpoint or manually revoke the OAuth tokens for any devices that they have previously provisioned. To reset access to the Webex Teams (desktop, mobile, and web application), from within Control Hub, as administrator, you can navigate to **Users [User] > Roles and Security > Security** and select **Reset Access**. This process can take up to six hours, so if immediate revocation is required, you should delete the user to revoke access immediately. Tokens can also be revoked from an API.

End users or developers who are security focused also can revoke OAuth tokens for devices that have been registered and active within an account for the past 30 days. To do this, a user or administrator can open Webex Teams and navigate to General > Recent sessions. After authenticating to Webex, a user can navigate to devices and choose **END SESSION**, as shown in Figure 12-9.

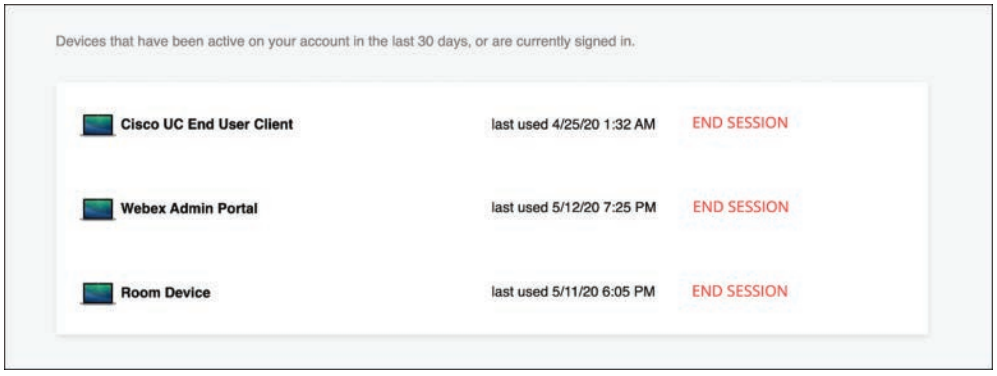


Figure 12-9 *User-Based Web Portal That Allows Revocation of OAuth Tokens*

https://t.me/learningnets
@cisco_in_persian_channel

As previously discussed, the Webex Calling service is bundled together with the Webex Teams client for messaging services. The next section will discuss critical aspects of security for messaging services when using software clients such as Webex Teams.

Securing Messaging Services

Messaging services have become a very popular form of communication for both personal and business use due to convenience and ease of integration with other environments. As we discuss and define the capabilities of Cisco's cloud messaging service, we may need to temporarily forget about traditional on-premise messaging services and how they are secured. Within Cisco's messaging platform, there is certainly a capability to send someone a message, but the platform itself was designed for increasing collaboration amongst a team, so the capabilities are expanded and very different. As an example, Cisco's messaging platform includes the ability to share items such as files, pictures, URLs, and whiteboard drawings and can easily be used to collaborate with users located at external organizations. It also allows team members to edit files that have been shared to a Webex Teams space in real time.

While the convenience of using messaging to communicate and share information is key to a quality experience, it is also one of the potential dangers that exist for introducing attack vectors into an organization. The next few sections will provide some insights to the general threat landscape for organizations that use cloud messaging platforms. One area of concern is that messaging platforms, especially such as those that are installed in an unauthorized fashion (shadow IT), can leave confidential information on the server(s) after messaging sessions are ended, which increases the risk of data leakage and data exfiltration.

When we consider the consumption of messaging services from a cloud environment, the security challenges can get quite complicated. As an example, traditional security solutions involving proxy servers and/or firewalls may not have the ability to inspect content that is embedded inside secured HTTPS messages, which can be used by unauthorized or insecure messaging platforms. This secure channel can be used for attacks involving data exfiltration without allowing an organization to have any insight to potential or ongoing attacks. Additionally, attackers can use messaging platforms to attack an organization by embedding an attachment into a message that has been infected with malware such as a Trojan horse, spyware, worms, or viruses. When this occurs, it is possible for an attacker to use messaging clients as a way to gain backdoor access into an organization while bypassing perimeter-based firewall protection.

Given these examples, it is easy to understand why messaging platforms need to have security built into them. We have only begun to scratch the surface of the security considerations that an organization should have. Fortunately, the Cisco Webex platform has already considered these potential threat vectors and has implemented security features to allow organizations to consume messaging services in a secure fashion. Although the rest of this section will focus on security controls, a holistic approach for safeguarding an organization and its data from messaging platforms such as Webex Teams is a

https://t.me/learningnets

@cisco_in_persian_channel

two-pronged approach based on user training and technical controls. Employees should be trained not to share data and content with others who do not need to know it, and they should be trained not to open files that seem to be suspicious or that come from unknown individuals. While user training is an important topic to discuss, it is out of scope to discuss this topic in much more detail. Instead, we will focus on the technical controls that an organization can leverage within the Webex Teams messaging platform to further reduce an organization's attack surface against well-known attacks.

At this point, we have already discussed some of the components that can be used to provide an element of secure messaging, such as authentication, authorization, and provisioning. The next section focuses on the various technical controls that help provide privacy and prevent against data loss.

End-to-End Message Encryption

Privacy is one of the primary security benefits that organizations gain by using Webex Teams for messaging services. All messaging traffic is encrypted by default and from end to end for users of Webex teams, for traffic that is in transit and while it is at rest. If an organization (or attacker) is able to decrypt a TLS session used for a message sent within Webex Teams, they would find that the messages and files have been separately encrypted with a key unique to the Webex Teams space. Cisco's Key Management Service (KMS), located in Cisco's Webex cloud by default, is the service that is responsible for managing encryption keys that are necessary for providing end-to-end encryption of messages. When an organization leverages Cisco Webex Teams for messaging, it is also the KMS that ensures that content is decrypted only when necessary. Operationally, when a user running Webex Teams needs to send a message to a different Webex Teams space, KMS requests a conversation encryption key via an Elliptic Curve Diffie-Hellman (ECDH) exchange tunnel from the KMS. The message can then be encrypted with an advanced AES-256-GCM cipher. Members of a Webex Teams space can decrypt encrypted messages by accessing the conversation encryption key. To access the encryption key, a URL is included in an encrypted message in order to provide a pointer to the encryption key. This flow is shown in Figure 12-10.

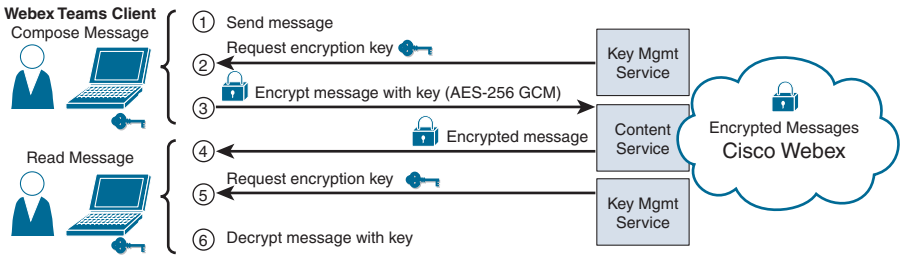


Figure 12-10 *Key Management Service Sends and Receives Encrypted Messages*

When a Webex Teams space is created, the KMS also creates a data structure that allows the KMS to track the keys for a given Webex Teams space and also the people authorized

https://t.me/learningnets
@cisco_in_persian_channel

to receive the keys. Because KMS relies on tokens from the authorization service, only authenticated (and authorized) users are given the unique keys that are used to encrypt and/or decrypt content located in a space.

For some customers, the storage of encryption keys in a cloud environment may not be permitted based on security policies. If this is the case, organizations can leverage Cisco's Hybrid Data Security (HDS) architecture, which allows organizations to install and manage their own KMS on-premises. When organizations host KMS on-premises, encrypted data and content remain stored in Webex Teams data centers, so the organizations must still provide connectivity to Webex cloud environments. In an HDS deployment an organization is responsible for key management and storage. Since keys are stored within a local database, customers are responsible for backing up this database because if keys are lost, then Cisco does not have a way of replacing them. As you can see, there are some security benefits, but there are also some additional responsibilities required with a HDS architecture. Before an organization takes on these responsibilities, they should be aware of the risks that come with owning encryption keys and be committed to performing the necessary tasks to maintain availability of services. If there is a failure in the HDS cluster, users are unable to create or request new encryption keys, but fortunately Webex Teams clients cache keys, so therefore they can still send and receive messages. A limitation is that users will not be able to create new Webex Teams spaces and/or be able to join or leave existing spaces.

When customers use the on-premises HDS architecture, they are also moving additional functions that are located in the security realm, such as indexing and compliance, to an on-premises location. The specific features are as follows:

- **KMS:** Provides management and storage of encryption keys
- **Search indexing:** Provides encrypted search indexing for Webex Teams content
- **Compliance:** Provides an on-premises E-discovery engine to allow services on encrypted messages for information assurance and compliance purposes

For more information, refer to Cisco's deployment guide for Cisco Webex Hybrid Data Security.

External Communications and Content Management

Many organizations want to easily federate with external organizations, and that capability is natively provided by Cisco's messaging architecture. A risk of federating with external organizations is that it exposes an organization to the threats that we previously discussed around messaging: from attackers that are external to an organization. With Webex Messaging, organizations can block external communications to new spaces within **Webex Control Hub > Settings**. As shown in Figure 12-11, this feature provides the ability to restrict users to invite external contacts to Webex Teams spaces and also to prevent users from joining Webex Teams spaces external to the organization to hinder insider attacks, such as data exfiltration from a corporate network. If communications with certain external organizations, such as a business partner, are permitted, whitelist exceptions to their external domains can be added.

https://t.me/learningnets

@cisco_in_persian_channel

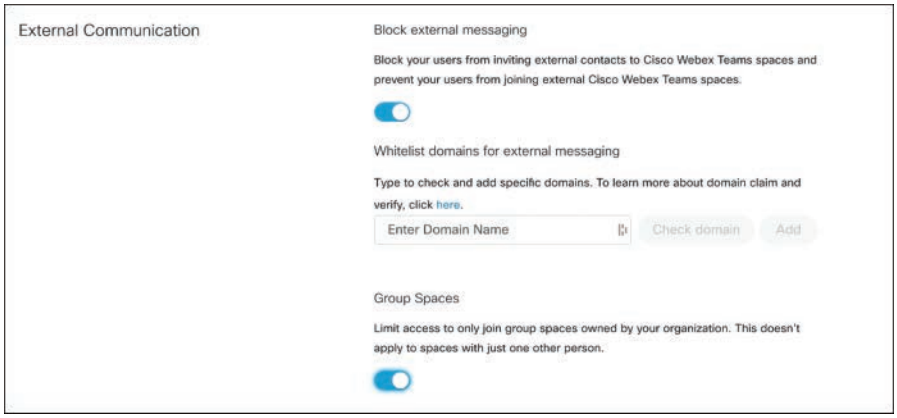


Figure 12-11 *Control Hub Allows Organizations to Block External Communications*

If additional security controls are required due to security policies, you, as administrator, can browse to **Services > Message > Settings** within the Webex Control Hub. From this page, you can restrict users' ability to upload, download, or preview content from certain device types (e.g., desktop, mobile, web, bot). If whiteboard devices are deployed across an organization, you can apply restrictions to new spaces also.

Note In the future, the ability to configure additional file sharing controls will be possible by leveraging Microsoft Active Directory security groups.

Content Management is another approach to reducing an organization's attack surface when sharing files by mitigating concerns with data loss and spread of malware. Embedded content management configurations allow native Webex Teams file storage to be disabled so that organizations can leverage IT-approved repositories in order to help with security and also to eliminate risks associated with shadow IT. To streamline access to IT-approved repositories, Webex Teams allows for integration with other cloud-based storage options. When Content Management is enabled, Cisco supports these storage repositories, which support a number of security features to be enabled to

- **Webex Teams native file storage:** Enable/Disable whiteboard drawings, screen captures, annotations, file uploads, file preview
- **Microsoft OneDrive and SharePoint Online:** Enable/Disable SharePoint linked folders. Allows users to share, view, and work on content stored within OneDrive and SharePoint from Webex Teams
- **Box:** Allows users to share, view, and work on content stored within Box from Webex Teams

To configure Enterprise Content Management, you should navigate to **Services > Message > Settings > Content Management > Settings**. Content Management can be enabled globally for all users within an organization or manually enabled for individual

https://t.me/learningnets
@cisco_in_persian_channel

users. In the future, Cisco will provide organizations with the ability to define labels based on governance of data and security policies. Webex Teams spaces will be able to be classified with these labels. This will help ensure that external users are not allowed to join a Webex Teams space in which classified information is being shared.

From within a content management approach, Cisco provides another layer of security granularity by providing the ability to police files that have been shared inside of spaces. This control provides protection against an insider or outsider sharing malware or confidential information within a file. The ability to police files comes from Anti-Malware File Scanning, Data Loss Prevention (DLP) solutions, and Cloud Access Security Brokers (CASB). We will discuss more about DLP and CASB in upcoming sections.

Anti-Malware File Scanning is powered by Cisco Talos ClamAV. This feature helps mitigate the threat of data leakage (accidental or malicious) to unauthorized individuals through malware, virus, and phishing attacks. It provides in-line scanning of files and links that are uploaded inside Webex Teams spaces to prevent the spread of malware and viruses. This feature is useful for protecting an organization from users so that they cannot download an infected file or document.

Anti-Malware File Scanning works by temporarily placing a file/document inside a quarantine area until it is scanned. If a file is infected, a user cannot download the infected document. As shown in Figure 12-12, Cisco's anti-malware protection also provides a customizable view of historical scans that have occurred in the past day, week, or month across the organization. Anti-malware protection also provides the ability to scan for malicious URLs in order to protect against malware and phishing threats.

Note Files that are uploaded to Webex Teams' spaces before Anti-Malware File Scanning is enabled are not scanned.

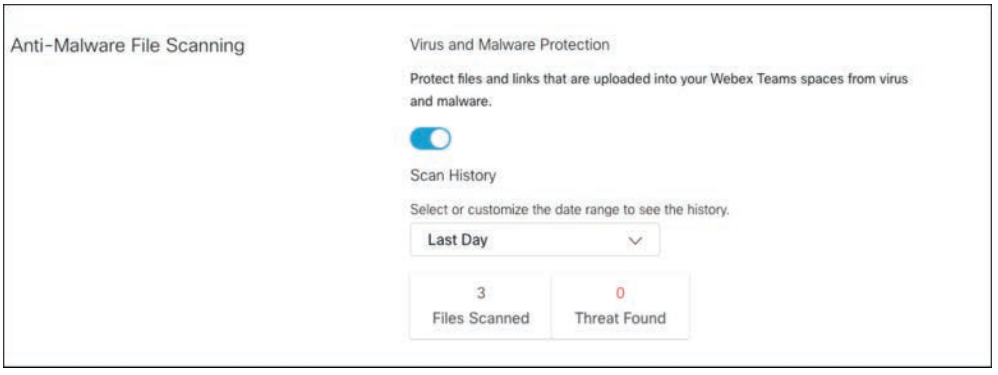


Figure 12-12 *Control Hub Allows Organizations to Scan Content When Uploading into Webex Teams Spaces*

https://t.me/learningnets
@cisco_in_persian_channel

In some cases, the longer that content resides within a Webex Teams space, the more risk it presents to an organization for being leaked or lost to external entities. We discuss this topic in greater detail in the next section. To reduce the amount of time that sensitive information is stored within Webex Teams spaces, Cisco enables organizations to configure data-retention policies. By default, Cisco stores data indefinitely, but you, as administrator, can customize the data-retention interval by browsing to **Settings** within Control Hub.

The minimum retention period is 1 month, and the maximum is 120 months. At the end of the retention period, user data is automatically removed by the system and cannot be retrieved. To provide forensics and compliance capabilities, compliance officers can add exceptions to retention policies and put users on a legal hold if, for example, a user or employee needs to be placed under investigation for unauthorized content sharing. This capability helps ensure that users' content can be preserved and not purged by an organization-wide retention policy during investigations. Webex Teams can also be connected with external providers for enhanced archiving services.

Data Loss Prevention

The loss of sensitive data is a major risk for organizations. Whether data loss involves intellectual property (IP), secrets from a government agency, or personal information, an organization often is faced with severe implications if or when this happens. Data loss prevention (DLP) is meant to help protect the loss of data using technical security controls and processes to protect against intentional data leakage or unintentional data loss. As an example, DLP tools and processes may have prevented information from being shared with Julian Assange, a notable hacker who went on to start WikiLeaks.

To apply DLP techniques, organizations must be able to monitor user behavior and apply corrective actions when a user's behavior is out of compliance with policy. As previously discussed, Webex Control Hub provides role-based access to users who require administrative privileges. Users who have been assigned the specific role of compliance officer within Webex Control Hub can monitor that the data shared within Webex Teams spaces remains in compliance with acceptable use policies, professional standards, and the relevant legal requirements that pertain to the organization. To ensure separation of duties, Cisco does not allow administrators to assign themselves with the compliance officer role.

To accomplish these tasks, Cisco provides compliance officers with access to the eDiscovery search and extraction console. This console enables compliance officers to perform activity searches across specific Webex Teams spaces as well as to create reports to ensure messages and content being shared are in compliance with organizational policies. The eDiscovery console provides contextual data such as time, user IDs, space names, times, and dates. As shown in Figure 12-13, if an employee is out of compliance with a legal matter and needs to be further investigated, the compliance officer may preserve this employee's content so that it is not purged by the organization-wide retention policy; this can be done by creating a "matter" and placing users (custodians) on legal hold. In a scenario in which an employee tries to remove evidence against them by

https://t.me/learningnets

@cisco_in_persian_channel

deleting messages or attachments, it doesn't matter because the Webex platform preserves a user's data on the back end once they are placed on legal hold. Reports are generated in .eml (email) format. Cisco recommends use of a third-party application in order to effectively groom and render data to meet organizational reporting requirements.

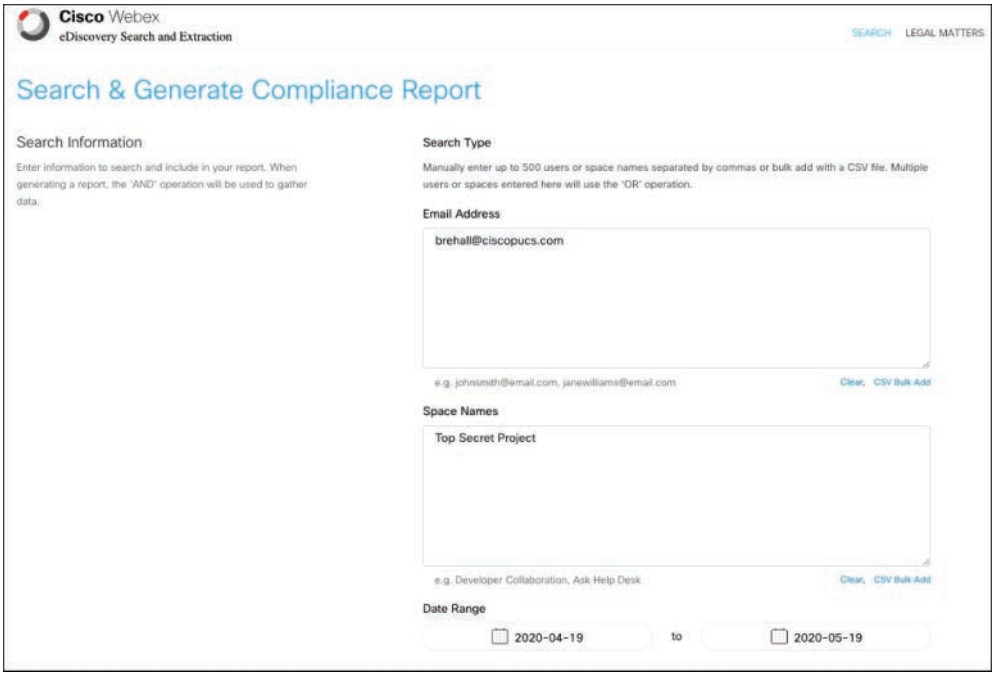


Figure 12-13 *eDiscovery Allows Compliance Officers to Search for Information Located in Webex Teams Spaces and Generate Compliance Reports*

Organizations desiring to increase visibility for messaging traffic in order to provide compliance and monitoring can leverage the EventsAPI. As a RESTful API, the EventsAPI allows authorized third-party software such as DLP and/or archival solutions to access, monitor, and archive (and search) the data that has been generated by users within a Webex Teams organization. Organizations that have developers and/or programmers can use the EventsAPI to create a custom data loss prevention practice as they develop homegrown security tools. The EventsAPI can be accessed from the Cisco developer web portal as shown or from a third-party tool such as Postman. This way, application developers can better familiarize themselves with the API and develop security tools that leverage it.

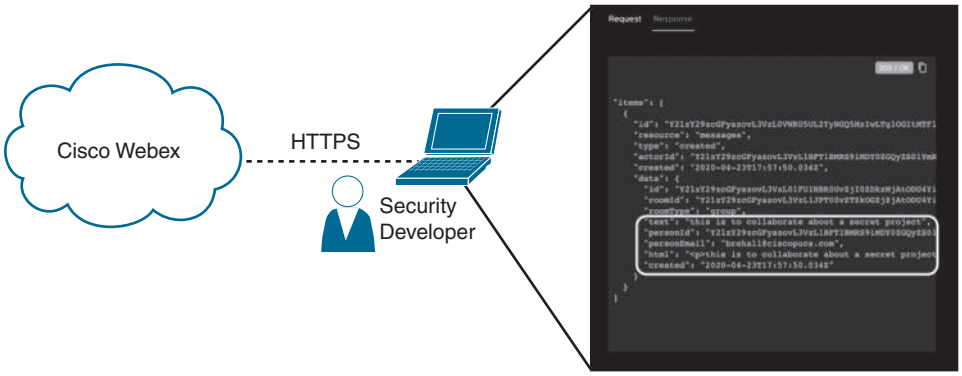
Leveraging the EventsAPI to query specific parameters from Webex Teams to monitor and detect issues with data leakage or exfiltration attempts provides compliance officers with a response that includes the following parameters:

- **Resource:** The type of resource in the event
- **Type:** The action that took place in the event

https://t.me/learningnets
@cisco_in_persian_channel

- **actorId:** The ID of the person who performed the action
- **From/to:** A list of events that occurred from/to specific dates and times

Figure 12-14 shows a response that could be obtained by querying the EventsAPI. For additional information on using the EventsAPI for security and compliance purposes, see the API reference guide on Cisco's Webex developer portal at https://developer.webex.com/.



Events API–Cisco Developer Portal https://developer.webex.com/.

Figure 12-14 *Developers Can Use the EventsAPI Provided by Cisco's Developer Portal to Develop Homegrown Compliance Tools*

In many circumstances, organizations may find it necessary to implement safeguards that automatically prevent information from being shared rather than taking a reactive approach that is manual. In a manual approach, an organization is simply doing damage control after an employee is misusing or failing to protect an organizations data. An automatic approach can be beneficial if there is a threat of a user sharing information that may be considered confidential-based intellectual property, which may cause embarrassment to an organization, or cause them to lose a competitive advantage, such as before releasing a new product. Other examples involving loss of data belonging to an organization's workforce include social security numbers, credit card numbers, financial information, and/or medical conditions/treatment plans. For this type of enhanced functionality, Webex Teams can integrate with multiple DLP providers to identify policy violations and take immediate remediation actions.

When Webex Teams are integrated with cloud-based DLP providers, such as a cloud access security broker (CASB), enhanced security compliance can be performed by

- Monitoring messages and Webex Teams spaces
- Providing auto-remediation (deletion) for scenarios that violate policies
- Providing alerts to users and administrators upon policy violations
- Automatically removing users who trigger policy violations from the Team Space

https://t.me/learningnets
@cisco_in_persian_channel

An example of a cloud-based DLP provider/CASB is Cisco Cloudlock. To integrate Webex Teams with Cloudlock, you must enable the EventsAPI in your Webex Teams instance. Figure 12-15 shows an incident flagged by Cisco Cloudlock in order to prevent the leakage of personal information, such as employee salary data.

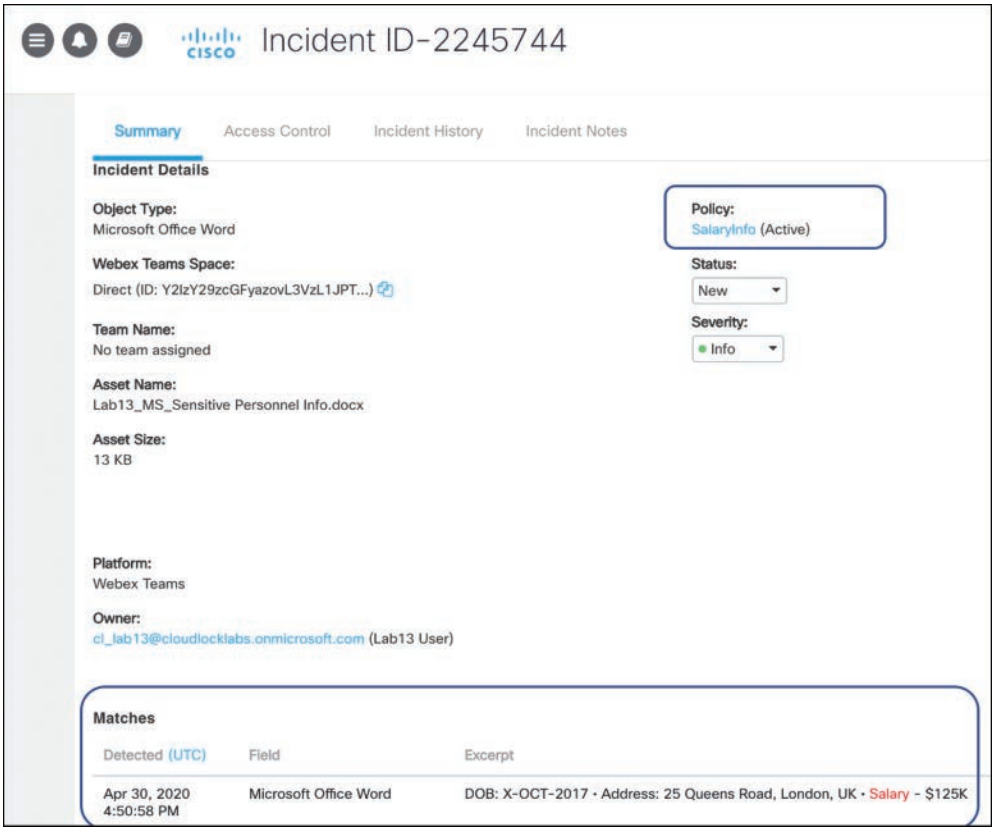


Figure 12-15 *An Incident Generated by Cloudlock for Policy Violation About Sharing Salary Information Within Webex Teams*

Examples of third-party applications that can be used to effectively groom to meet organizational reporting requirements along with archiving solutions include Smarsh (Actiance), AGAT, Verba Verint, and Global Relay (Cisco Advanced Services). Additionally Cisco supports the following eDiscovery tools that can also be used to ingest data created by a compliance office (in .eml format) in order to support advanced indexing and searching: Recommind, Veritas, LogikCull,Relativity, and Exterro. For a full list of Cisco partners that specialize in DLP and archiving solutions, review the article on Cisco Webex Teams and integration with archiving and data loss prevention solutions. For additional questions about integrating any of these solutions with Webex Teams, Cisco provides an email alias that organizations may use to submit questions: ask-teams-compliance@cisco.com.

https://t.me/learningnets
@cisco_in_persian_channel

Mobility Management

According to Global Market Insights, bring your own device (BYOD) initiatives are becoming more common and are estimated to reach a market value of almost $367 billion by 2022. Survey results conducted by Verizon, a service provider for mobile devices, points out that 45 percent "of respondents admitted that their defenses were falling behind attackers' capabilities." When Control Hub is used to secure the Webex Teams messaging environment, Cisco offers several security features that provide basic enterprise mobility management controls such as

- Pin lock enforcement for iOS and Android devices
- File sharing controls for iOS and Android devices
- Token revocation and remote wipe of cached content
- Automatically logging users out of idle sessions
- Data at rest encryption on mobile devices

To protect against the risk of other types of attacks on mobile devices, specialized Mobile Device Management (MDM) solutions can allow you to disable access from a rooted device, disable copy/paste, disable document sharing, and disable backups. One challenge that organizations are faced with is that in order to leverage a MDM to support security features for a mobile device, a device must be onboarded to a MDM first. In some cases, even if a device is onboarded to a MDM, some security features cannot be deployed without an Enterprise Mobility Management (EMM) suite. Gartner refers to EMM suites as the "glue that connects mobile devices to their enterprise infrastructure" because EMM includes functions such as provisioning and auditing of mobile devices.

Another approach for providing UC security on mobile devices is with Mobile Application Management (MAM), which is often included in discussions of an EMM suite. Cisco supports MAM for Webex Teams, which allows wrapped versions of Webex Teams applications to be developed and hosted internally within an organization's application store. This option allows an organization to deploy a secure version of Webex Teams without previously onboarding a device to an MDM beforehand. As part of Cisco's MAM program for Webex Teams, Cisco supports AppConfig, which is a community that provides a standard schema for developers to implement application configuration and management on mobile devices. This standard makes it easier for developers (and organizations) to implement security controls on mobile devices. Examples of security controls that can be implemented include security features such as SSO and remote wipe of a mobile device. It allows organizations to deploy Webex Teams on mobile devices while supporting bring your own device (BYOD) initiatives in a secure manner.

An example of a solution that can be used with a hybrid MDM/MAM solution is through a Software Development Kit (SDK). In a future release of software, Webex Teams will have an SDK integration with Microsoft Intune. This integration is beneficial because it also allows organizations to enforce application policies without requiring device enrollment into a corporate MDM beforehand.

https://t.me/learningnets

@cisco_in_persian_channel

As previously discussed, Control Hub also provides the capability to revoke user access as described in RFC 7009 if it is perceived that a device is compromised, if a device is misplaced, or when a user is terminated but not yet deprovisioned from Webex. To do this, you can navigate to **Users> (User) >Security** and click **Reset Access.** This process resets the user's access and also remotely wipes all cached content on the mobile devices that the user is authenticated into. As shown in Figure 12-16, access to a cached file has been revoked from a mobile device that is signed into a Webex Teams space named "Secret Project."

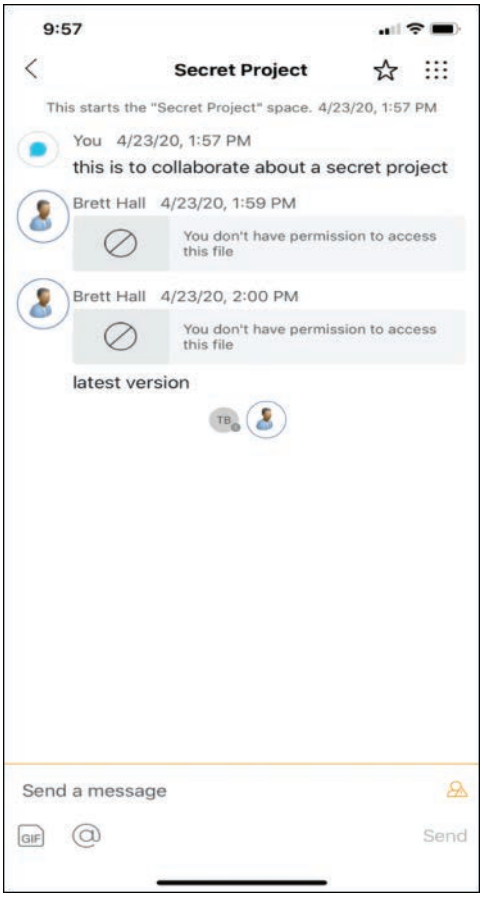


Figure 12-16 *Removal of Cached Content from Within Webex Teams on an iPhone*

Meeting Management and Security Controls

Cisco's native cloud meetings solution, Webex Meetings, is a platform that provides web and video conferencing for up to 1000 participants and is accessible from desktops, mobile devices, IP voice/video devices, and the PSTN. Optionally, participants have

https://t.me/learningnets
@cisco_in_persian_channel

in-meeting chat functionality and meeting hosts can choose to record meetings. Similar to the other Calling and Messaging platform, the Meeting platform provides security by default by leveraging TLS 1.2 secured channels. Encrypted media can be transported over UDP, TCP, or TLS. Similar to our discussion in Chapter 10, since much of an organization's collaborative activity such as sharing of information and decision making takes place across meeting environments, it can be a high-value target for attackers or bad actors who are impersonating an individual. Therefore, an organization must carefully identify which attacks they are most vulnerable to and what security controls need to be implemented across their Meetings environment.

Due to social issues and industry trends, the increased use of meeting platforms by teleworkers has created a renewed focus for strong security inside meetings. Over the years Cisco has developed numerous security features that can be toggled on and off on a global level and also for meetings hosted inside of personal meeting rooms (PMRs). This section will discuss some of the various security features that can be implemented to protect against threats related to privacy such as eavesdropping while also discussing a different type of social attack aimed at disruption of Meetings. We have discussed this throughout many of the chapters, so this chapter is not any different. Security features can be enabled within Meetings while ensuring that an organization can create and maintain a collaborative experience. The areas of security that we will be focusing on include the scheduling of meetings, authentication of meeting participants, in-meeting privacy controls, and protection of data at rest.

Un-Scheduled and Scheduled Meetings

There are two primary ways to apply security for scheduled and ad hoc Webex meetings. Because ad hoc meetings can be scheduled on-demand through personal meeting rooms (PMRs), we first examine ways to provide security without using calendar integration. One thing to consider about PMRs is that they have an increased attack surface when compared to scheduled meetings. A reason for this is because the meeting addresses of PMRs typically follow a consistent address scheme, so this can make it easier for bad actors to create a targeted attack by joining PMRs in order to obtain information or to disrupt a meeting.

While using a PMR, the ability to control who is host and/or a presenter can help provide security and a quality meeting experience in a number of ways. From a security perspective, a host can help prevent unauthorized access to content or backdoor access to the network. The reason is that presenters can share presentations, specific applications, or an entire desktop. Presenters also can grant and revoke remote control over the shared applications and desktop to individual attendees. Additionally, PMRs have various security features that the end user can configure before a meeting is started; they include

- Using an alternate host to eliminate the possibility of unexpected host privileges being passed to unexpected or authorized attendees (e.g., the first attendee)
- Automatically locking the room after a meeting starts (e.g., after 10 minutes)
- Limiting attendee privileges such as remote control of a user desktop

https://t.me/learningnets

@cisco_in_persian_channel

Many of these security controls may seem like common sense, but they are still left disabled. As a general rule, once a meeting starts it can be difficult to enable security. By globally pre-configuring PMR meetings to automatically lock after a specified amount of time, care is taken to ensure that meeting rooms are secure once meetings are started. Although auto-lock can be pre-configured globally, Cisco allows attendees to customize this setting under their Preferences>My Personal Room>Automatic lock setting. If guests are late to a meeting, meeting hosts can manually admit personnel who are temporarily placed within a lobby area. If desired, PMRs can be automatically locked for all attendees once a host starts a meeting. To allow guests into their PMR, a host would need to manually admit each attendee. As of the WBS40.9 release, the Meetings platform can be configured to automatically remove people that a host has not manually admitted after a specific amount of time. Configurable values are 5, 10, 15, 20, 30, 60 minutes. This protection helps protect against bad actors and also insider attacks that may attempt to join recurring meetings within PMRs and a host accidentally adding them. Hosts should also be trained to expel users who cannot be identified, are acting suspicious, or are disruptive to a meeting.

Scheduled meetings can be configured directly through the Webex console and/or through Webex's hybrid calendar service integration. When scheduling meetings through Webex, meeting passwords are included by default. In scenarios where there is the risk of the Webex Meeting invite being forwarded indiscriminately, shared inappropriately, or posted on public forums such as social media, it is a good practice to remove the password and the meeting invite string. Although the meeting invite string is hashed one-way and not human extractable, it can allow unauthorized users to login to a meeting by simply clicking "join meeting." When passwords are removed from the meeting invitation, the host will be responsible for delivering the meeting password to intended participants via other secure means such as email, messaging, phone call, and so on.

Through Control Hub, administrators with privileged access can set up the Hybrid Calendar Service integration so that as users schedule meetings the process of adding a link to a PMR is simplified by entering a keyword into the location field (e.g., @webex or @meet) of a calendar invitation.

Through Control Hub, Cisco offers three options for hybrid calendar service integrations:

- On-Premises Microsoft Exchange
- Native cloud Microsoft 365
- Native cloud Google G Suite

Similar to other Webex Hybrid Services, a software connector (e.g., Calendar connector) is used to bridge the Cisco Webex cloud environment to the environment that an organizations calendar service is located at. The Calendar connector resides and can be managed from an Expressway-C server. Once deployed, the connector securely connects to Cisco Webex using HTTPS encryption.

https://t.me/learningnets

@cisco_in_persian_channel

When integrating with Microsoft 365, the Hybrid Calendar Service uses the Microsoft Graph API and Microsoft Graph authorization to access a user's calendar within their environment. When integrating with Google, Cisco's Hybrid Calendar Service uses Google calendar APIs, and the OAuth 2.0 client credentials grant flow to the Google Authorization Server. To encrypt data, the Hybrid Calendar Service uses the same Webex encryption service (KMS), located either in the cloud or on-premises. The administrator doesn't need to do anything within Control Hub to ensure that the Hybrid Calendar Service doesn't store or send unencrypted sensitive data such as a meeting description, meeting body, or email addresses of invitees; data is always encrypted to the Webex cloud. The scheduling flow when integrating with different calendar environments is shown in Figure 12-17.

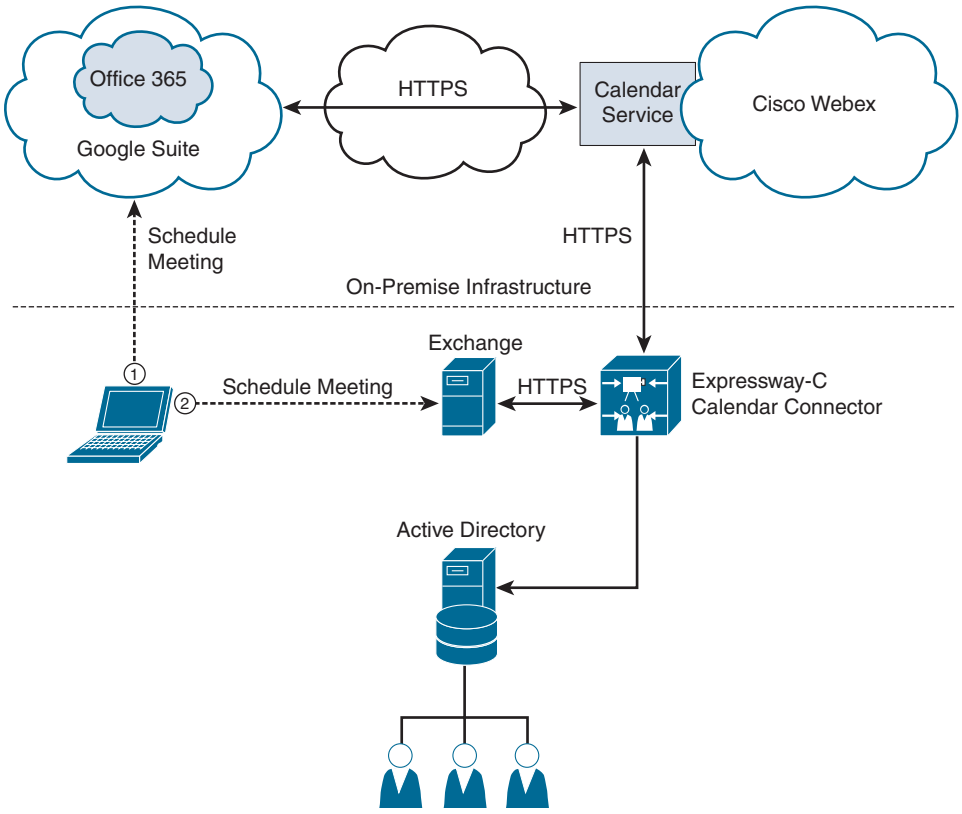


Figure 12-17 *Scheduling Options with Cisco's Hybrid Calendar Architecture*

For prerequisites and detailed information about integrating into these environments, review the deployment guide for Webex Hybrid calendar services at www.cisco.com/go/hybrid-services-calendar.

https://t.me/learningnets

@cisco_in_persian_channel

Meeting Authentication

Previously we alluded to using passwords as a simple way to prevent unauthorized access to meetings. In some meeting environments, administrators have relaxed security standards by not requiring the use of meeting passwords. The lack of security passwords has allowed attackers to barge into meetings and/or to annoy and offend conference participants. With that said, requiring strong passwords is a simple, practical, yet effective way of securing access to meetings. A strong password contains a mix of letters (upper- and lowercase), special characters, and numbers and is eight or more characters long. Such passwords can be configured within Webex Administration by browsing to **Configuration > Common Site Settings > Options**. When possible, passwords for meetings should not be reused because reusing them makes them easier for attackers to guess. Passwords should also not contain words from a dictionary to prevent dictionary attacks.

As previously discussed, Webex Meetings also supports SAML 2.0 for SSO authentication and OAuth 2.0 for authorization. When possible to use SSO authentication, this is Cisco's recommended method of securing access to meetings because it ensures that users have an account on the system before authenticating. As previously discussed, SSO and SSO with dual-factor authentication options are more secure than built-in Webex authentication and/or standalone meeting passwords that can be shared.

A caveat around requiring SSO authentication is that it may limit the ability for guests to join Meetings. Organizations that regularly have guest attendees should develop a guest user policy that dictates how guests can securely authenticate into meetings. When considering a guest user policy for joining meetings, organizations have the following options:

- **Guests not permitted:** Only internal users or those with Webex accounts can login.
- **Guests permitted globally**: This is the most common option but is vulnerable to bad actors who may come across a Webex meeting invitation or are inadvertently forwarded a meeting invite with a password.
- **Guests permitted globally; hosts configure their meetings to require login:** This option permits external users that do not have a Webex account on your site but adds granular security. In this option, users are trained to select "require login" during meeting scheduling from the advanced scheduling tab to restricted guests as needed.

An intermediate option that many organizations have implemented in order to support guest policies that require logins is to use an IdP in order to sideboard guest users. This option entails a process of adding specific guest users into a separate IdP directory so that guests can join meetings securely in a similar way as internal users. This option is beneficial for organizations that are already using an IdP for SSO but simply need to add guest users who are regular attendees to Meetings in order to support security policies that require strong authentication. The downside to this option is that an organization will need to manage multiple user directories.

https://t.me/learningnets

@cisco_in_persian_channel

End-to-End Encryption for Meetings

By default a media stream flowing from a client to a Webex Meeting server is decrypted after traversing the Cisco Webex firewall. This allows for meetings to be recorded and shared with users that are unable to attend meetings originally. Before media is sent to meeting participants, it is re-encrypted in order to provide privacy and security. For organizations that require a higher level of assurance that conversations will be private, Cisco provides end-to-end encryption for meetings. Operationally, this is similar to a Zero Trust based architectural model, which we have already discussed for Messaging services, and how end-to-end encryption is provided using KMS Messaging services with Webex Teams, but in this scenario we are discussing Cisco Webex Meetings instead. Cisco is currently the only cloud-based Meeting provider that supports end-to-end encryption.

When meetings are created to support end-to-end encryption, the Cisco Webex cloud does not have access to encryption keys, so it does not decrypt media streams but is still able to switch media flows by referencing header information. Instead, the meeting host generates a meeting encryption key by using a 2048-bit RSA public and private key pair. The host then transparently distributes the key to meeting participants by encrypting the meeting encryption key with their public keys and using a TLS channel for client/server communications. When end-to-end encryption is enabled, all data that is generated by Cisco Webex clients in a meeting (e.g., voice, video, chat) is encrypted using the shared meeting encryption key using either AES-128-CBC or AES-256-GCM–based ciphers. By default, end-to-end encryption session types are not enabled on all Webex sites, so this feature must be requested. End-to-end encryption is supported only for Cisco Webex Meetings and Cisco Webex Support. As previously discussed, there is an operational trade-off when enabling end-to-end encryption. The following features are not supported:

- Personal room meetings
- Video-device-enabled meetings
- Network-Based Recording (NBR)
- Webex Meetings web application
- Saving session data, transcripts, meeting notes, and so on

End-to-end encryption is supported only by the Webex Application (desktop and mobile apps). An example of why it is beneficial for customers to implement Zero Trust based models for Meetings is better understood while examining the topic of data sovereignty. By using end-to-end encryption, government organizations would not be able to capture information shared within Meetings. Another example involves an employee who believes they are wrongfully terminated by their employer. If the former employee is hostile, they can share links and recording passwords of Meeting recordings in order to exfiltrate information to an organization's competitor. This threat is easily mitigated with end-to-end encrypted Meetings since network-based recordings are not supported.

Figure 12-18 provides an operational view of end-to-end encryption.

https://t.me/learningnets

@cisco_in_persian_channel

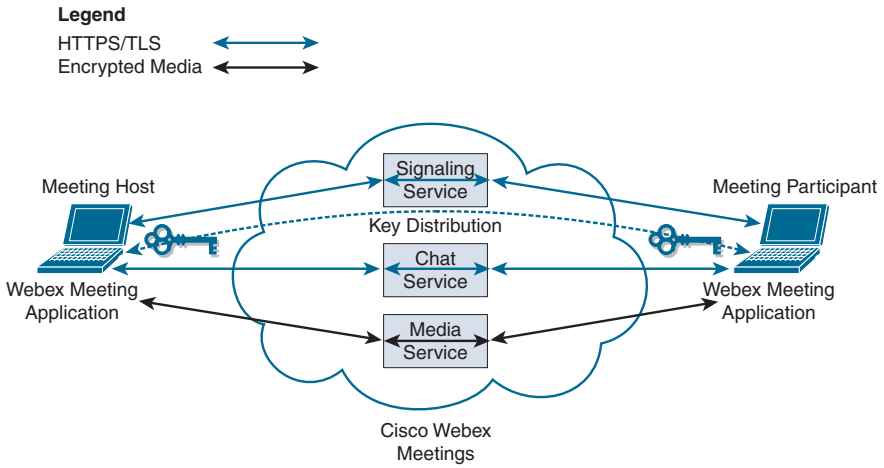


Figure 12-18 *Architecture for End-to-End Meeting Encryption*

Last but not least, organizations have the option of installing Webex Video mesh software on-premises, which effectively allows organizations to move a component of their cloud resources into a local data center environment. The benefit for customers is that they can save bandwidth and anchor their media streams locally and in a secure fashion instead of in a cloud environment in order to provide their users with the best possible experience. Given this information, Webex Video mesh software must register to the cloud and be managed from within Control Hub. This allows local Video mesh nodes to support periods of high demand by cascading resources to the Webex cloud environment. For organizations that do not permit network ports to be opened directly between local Video mesh nodes and Cisco Webex, a proxy server (transparent or explicit) can be used to provide additional inspection. Video mesh software uses a TLS connection over port TCP 444 to secure signaling to the Webex cloud.

In Meeting Privacy Controls

Hosts and administrators are able to help control the security and privacy of data within meetings. By providing entry and exit tones, which provide a beep or announce the name of individuals joining meetings, they can help control meeting attendance by unauthorized attendees, as shown in Figure 12-19.

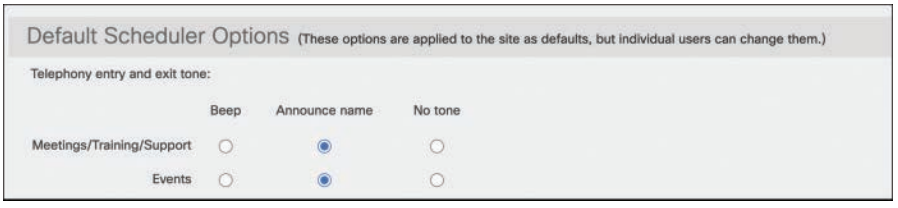


Figure 12-19 *Configuring Telephony Entry and Exit Tones to Minimize Unauthorized Attendees*

https://t.me/learningnets
@cisco_in_persian_channel

If hosts would like to configure personal preferences, a user can navigate to a Webex portal and then navigate to **Preferences >Audio and Video > Entry and exit tone** from the **Classic View**. Options may vary when using the Modern View. Once names or callers have been identified, conducting a roll call is a simple way of verifying that the right personnel are in attendance. If users cannot be identified, they can be expelled until they are able to join from video or identify themselves.

The ability for a host to control which participants can speak and/or share content be an effective safeguard. Although mute is typically considered an option for preventing distracting background noises that originate from meeting participants, it can also be an effective tactic to keep bad actors from hijacking and/or disrupting productive meetings. Therefore, meeting organizers may choose to mute all participants by default for certain meetings and then un-mute participants on an as-needed basis. This type of approach prevents bad actors from joining meetings and using vulgar language that is offensive to attendees. Similarly, allowing anyone to share content is a way to prevent bad actors from grabbing presenter privileges and then sharing content that is vulgar. To disable this option, a Webex site administrator can uncheck the feature "allow participants to share in meetings" in site admin>common site settings>options. Once this is complete, only meeting hosts will be able to pass presenter privileges to meeting attendees in order to share content. If content is being shared that is offensive, hosts have the responsibility and capability of expelling users.

Protection of Data at Rest

By default, Webex recordings are encrypted both at the file level and at the logical volume level using an AES-256 encryption key. When configured by the customer to do so, Cisco Webex Meetings stores meeting and user data that may be critical to your business, such as meeting transcription and people insights. As part of Cisco's transparency policy, Cisco specifies that user-generated information is stored in the Webex data center in the customer's region that is provided during the ordering process. For more information about Cisco's data center locations for Webex Meetings, please review Cisco's privacy data sheet for Webex Meetings.

Any data not encrypted at rest is protected by secure data center protection mechanisms and operational procedures. Administrators can secure recordings through password security mechanisms. Administrators also can prevent download of recordings. As of WBS39.11, administrators have the option of forcing the acceptance of a disclaimer to any attendee prior to viewing or downloading a recording. If a user chooses not to accept the disclaimer, that user does not gain access to the recording. Administrators also can customize their data retention periods. Recordings that are older than the configured retention period are moved to trash automatically.

Security Across Emerging Features

As previously mentioned, rapid innovation is taking place in UC cloud environments. According to *Forbes*, "Artificial Intelligence in the workplace has been identified as an untapped means to improve productivity, efficiency, and accuracy across

https://t.me/learningnets

@cisco_in_persian_channel

an organization." Cisco Webex provides several features that are based on artificial intelligence and machine learning algorithms that are providing innovation:

- Facial recognition
- People insights
- Webex assistant
- Meeting transcription

The integration of these types of features into cloud-provided UC services allows for integration into business applications and workflows to add value to organizations. The goal of this section is to help organizations know which features can be enabled and which security controls can be implemented to ensure that the organization and its associated workforce are operating in the most secure manner possible.

Similar to some of Cisco's advanced security features, these emerging features are not enabled by default. Because organizations and customers do not all have the same requirements, Cisco has provided the flexibility to enable these features and the requisite security on an as-needed basis. If organizations are interested in taking advantage of these features, both organizations and the user community must "opt in" to acceptable use policies that provide information regarding personal privacy, data management, and data retention of personally identifiable information (PII). Now that you understand these considerations, we can take a closer look at the implications for enabling these emerging features and the available ways of providing security when they are enabled.

Facial Recognition

Facial recognition is a feature that is used to easily recognize individuals within Webex meetings. When an administrator enables this feature within Webex Control Hub, an individual can display their name in a label format next to their face so they can be easily identified by other attendees. When this feature is enabled, Cisco uses deep learning, which is a subset of the machine learning algorithms, to generate a mathematical representation of the face. Facial recognition includes measurements between key points in the facial image so that reasonable accuracy can be returned in a variety of formats.

At the time of this writing, many lawmakers in municipalities, cities, and states are introducing legislation regarding the use of facial recognition technology. An example is the Biometric Information Privacy Act (BIPA), which provides restrictions on how private entities can collect and use a person's biometric data. In May 2019, San Francisco became the first city in the US to ban the use of facial recognition technology by city agencies, including the police department. As a privacy control, users are required to provide permission for Cisco to store their photos. Users have control over their facial recognition data. At any time, users may visit their profile settings at https://settings.webex.com to take a new photo or to remove existing facial recognition data. When enabled, facial recognition data is secured with encrypting tokens in Cisco's Webex cloud so that any transient data that is captured is unintelligible to third parties.

https://t.me/learningnets

@cisco_in_persian_channel

People Insights

The People Insights solution provides enriched user profiles of attendees inside Webex meetings and instant messaging clients such as Webex Teams and Jabber. To provide enriched user profiles, this data is gathered by leveraging and integrating information gathered from publicly available sources such as Wikipedia, LinkedIn, Twitter, Facebook, Angel, Crunchbase, and corporate directories (e.g., Active Directory). Once this feature is enabled within **Control Hub > Settings**, the types of information presented to attendees include

- Name and title
- Employment history
- Education history
- A listing of published documents and articles
- Subject matter expertise

A publicly accessible portal (https://people.webex.com) can be used to edit a user's public profile and associated data, as shown in Figure 12-20. To use this feature, directory synchronization must be enabled, and users must be linked to or managed from Webex Control Hub.

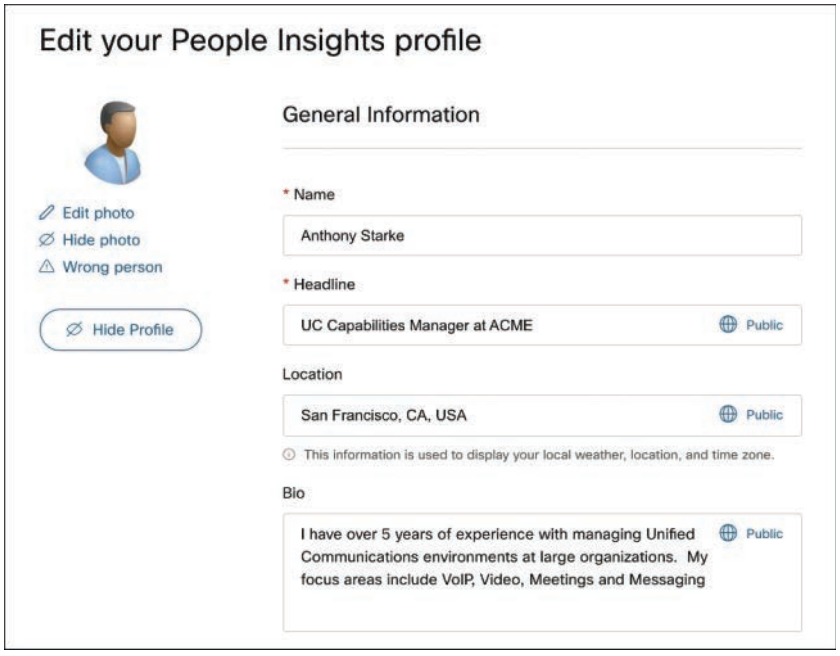


Figure 12-20 *Personal Data Used for People Insights Can Be Changed or Removed Through Webex Settings*

https://t.me/learningnets

@cisco_in_persian_channel

Current legislation regarding people's data includes the General Data Protection Regulation (GDPR), which was passed into law by the European Union in May 2018. When an organization is out of compliance with GDPR privacy and/or security requirements, harsh fines can be issued. Although only currently supported in US-based clusters, People Insights aligns to GDPR standards. When enabled, this feature provides users with an increased awareness of their digital footprint and public presence.

To ensure privacy of data, directory data is displayed only to other members of the same organization. To ensure that there can be no unintentional overlap or integration of the data sources, the different types of data are stored in separate databases in separate virtual private clouds (VPCs). Data is always encrypted whether in transit or at rest from both public and corporate directories. Because information is gathered from public sources, it is stored indefinitely by default. Data based on local directory (e.g., Active Directory) integration, such as what has been imported via Directory Connector, is removed when disabling directory integration.

It is a best practice to closely manage what type of personally identifiable information is stored on publicly available repositories. If necessary, data obtained from publicly accessible sources can be archived (hidden) at the request of a user by opening a case with the Cisco Technical Assistance Center (TAC) or by sending an email message to privacy@cisco.com. If desired, an entire account profile may be removed as well.

Webex Assistant

The Webex Assistant solution provides an AI-powered voice assistant that allows personnel to use their voices to interact with Webex room devices and the services that the devices connect to. When this feature is enabled, users can use their voices to start a scheduled meeting, join a Webex Personal Room, or call individuals who are listed in an organization company's directory by saying their names. Similar to other types of voice assistants provided by Apple (Siri), Amazon (Alexa), or Google, Webex Assistant operates off a *wake word*. To use Webex assistant, a user can speak to supported devices by saying a wake word such as "Hey, Webex" or "Ok Webex" followed by a request, such as "join my personal room." Webex Assistant also allows individuals to take notes, set up future meetings, and capture action items.

Once enabled under Webex Control Hub, Webex Assistant–supported devices leverage Google's speech engine over authenticated TLS connections to provide speech-to-text transcription, as shown in Figure 12-21. Once the text is generated, Webex Assistant uses a conversational AI platform to process the user's intent. From a data retention perspective, speech data that is sent to Google's speech engine is never retained to efficiently process or improve Google's speech accuracy. Similarly, Webex Assistant does not retain data for improving the AI platform or for additional machine learning. Once Webex Assistant is disabled, all relevant data is removed from the Webex Assistant service. From a confidentiality perspective, tokens used to authenticate with Google's speech engine are rotated on an hourly (or better) basis, and keys are rotated on a weekly basis.

https://t.me/learningnets

@cisco_in_persian_channel

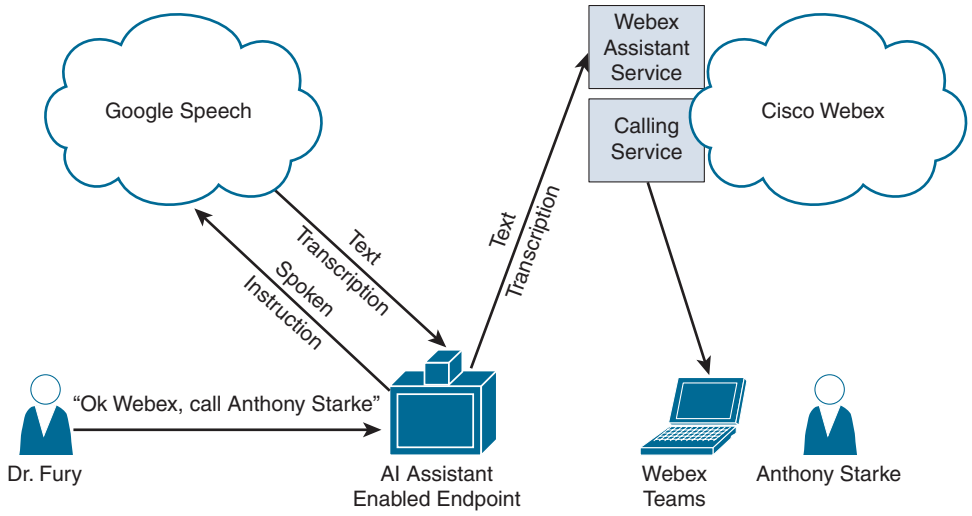


Figure 12-21 *Webex Assistant Integrating with Google Speech APIs*

Meeting Transcription

Webex Meetings provides an optional transcription service for recorded meetings. The Meeting Transcription service is free for customers who have network-based recordings enabled. When enabled, this feature allows users to review the transcript of a recording and navigate to any time period of a meeting that is of most interest, which helps improve productivity by allowing individuals to review and follow up on discussions from meetings faster and more effectively.

When a meeting recording is sent for transcription, the data is encrypted using RSA 2048-bit keys. The resulting transcript is returned and merged with the meeting recording. Cisco does not retain meeting content data at any time. When data is at rest, it is encrypted using 256-bit Advanced Encryption Standard (AES-256). The only resulting artifact is the combined recording and transcript in Webex. To enable meeting transcription for Webex Meetings, you should browse to **Configuration > Common Site Settings > Options**.

IoT Security

The innovation that is taking place is fueling the growth of the Internet of Things (IoT) supporting UC. One example is the ability to use a Webex Share device to wirelessly share content in meetings. Two different types of devices that organizations are using that can be supported in Cisco Webex platforms include the Webex Share device and Cisco Headsets. As the name implies, the purpose of a Webex Share is to share content between the Webex Teams client and a lightweight sharing device that can be transported anywhere in the network and connected to a monitor to easily enable desktop/content sharing.

https://t.me/learningnets

@cisco_in_persian_channel

To address the typical IoT security concerns, Cisco provides security in layers with Webex Share. Starting with network connectivity, Cisco provides enterprise transport and authentication across the network. As an example, authentication into the wireless network using X.509 certificates supports EAP-TLS encryption. The process to register Webex Share devices with the cloud is consistent with the process to onboard devices through Control Hub with activation codes.

Once a Webex Share is registered to Webex, connectivity between a Webex Share and the Webex cloud is secured using TLS. Every 30 seconds, unique tokens are generated by the Webex cloud and shared with the Webex registered device over a TLS connection. These tokens include information about the Webex Share identity; they are emitted using ultrasound from the device speakers to allow sharing of content. Cisco uses Intelligent Proximity with ultrasound and wireless connectivity to share content between a Webex Share (and other video devices) and the Webex Teams application by using security at the application layer with tokens. An example of the protocol interactions is included in Figure 12-22. Another layer of security involves physical infrastructure blocking the transmission of ultrasound waves. Because ultrasound signals do not typically bleed through walls, the range of a pairing token is limited; therefore, an outsider's ability to connect to a device is greatly reduced. If a device that a user is using to share content leaves the room, the content is removed from the screen automatically.

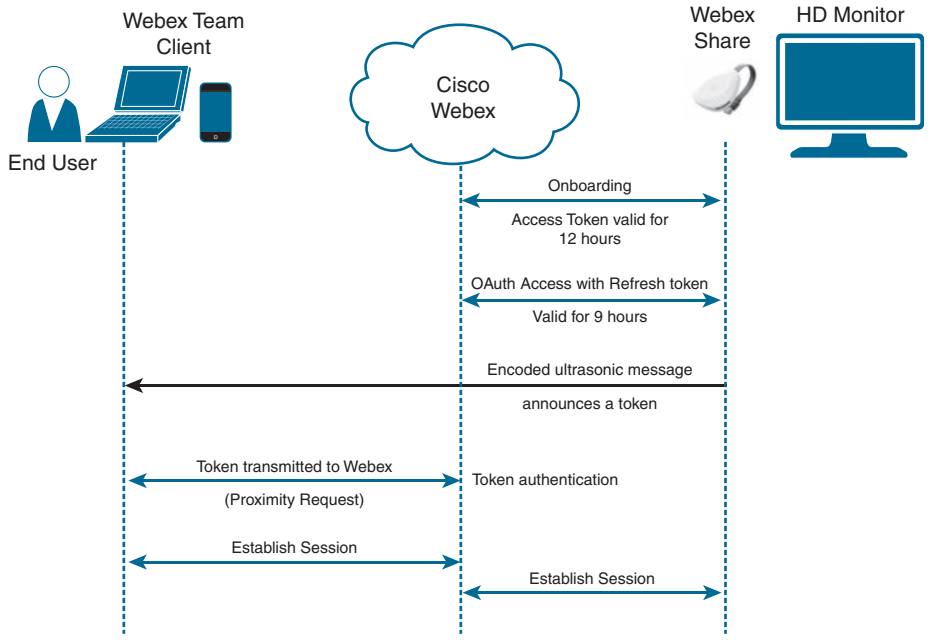


Figure 12-22 *Use of Proximity to Share Content with Webex Share Devices*

For many organizations, security policies may dictate that content sharing to local devices be disabled for security purposes. Control Hub can be used to disable connectivity to on-premises devices. To disable device discovery, you can navigate to **Devices > Settings**.

https://t.me/learningnets
@cisco_in_persian_channel

A nontraditional look at IoT devices involves use of both Bluetooth and Digital Enhanced Cordless Telecommunications (DECT)-based headsets. Although neither of these devices uses Internet Protocol (IP), they connect to IP devices such as IP phones, video devices, and PCs that use a UC client, so they can be considered IoT devices. Furthermore, there are security concerns with using either Bluetooth and/or DECT headsets because of the possibilities for an attacker to intercept communication streams or take over the hardware that is used inside the headset.

Cisco provides Bluetooth security by implementing a feature called *secure boot and image authentication*. This feature ensures that only authentic Cisco images can be installed on the headset and minimizes the possibility of malicious code becoming installed on Cisco Bluetooth headsets. Firmware running on top of Cisco Bluetooth headsets can be centrally managed from both Unified CM and Control Hub. This ensures that headsets are running the latest (and most secure) firmware available. Cisco encrypts Cisco headset firmware with an AES 128-bit key.

Recently, a team of researchers exposed a vulnerability in Bluetooth chips that resulted in an attack called Key Negotiation of Bluetooth (KNOB), which impacts certain devices that comply with Bluetooth specifications. This vulnerability allows an attacker to negotiate weak encryption (1 octet) so link-level encryption can be broken by brute force. Once this done, the attacker can act as a man-in-the-middle when located near a target in order to eavesdrop or inject malicious messages within an encrypted conversation. Fortunately, the chipset and firmeware used in Cisco Bluetooth headsets require a minimum of 56 bits (7 octets), so they are not vulnerable to a KNOB attack.

Since its creation in 1992, Digital Enhanced Cordless Telecommunications technologies have supported voice services on a wireless spectrum dedicated to DECT. DECT standards now have matured to DECT 6.0, which involve a medium to provide a unique, non-overlapping wireless frequency for up to 100 meters. Cisco's headsets align with DECT 6.0 functional and security standards. Similar to Bluetooth headsets, Cisco uses 128-bit encryption for DECT headsets. Unlike Bluetooth headsets, which have technology embedded, DECT headsets require a base station. Be aware that security researchers have previously found flaws within DECT, but security enhancements are ongoing to reduce immediate concerns in a phased process that the DECT form calls "Steps." Step A is the current standard for DECT security certification. Cisco is certified with Step A standards. Current Step A security standards include the following specifications:

- A base station is not kept "open for registration" for longer than 120 seconds.
- A base station and handset support encryption activation, and the base activates it for all calls.
- Encryption activation occurs immediately after connection establishment.
- A base station creates and allocates a 64-bit authentication key (UAK) when the handset is registered.
- A base can authenticate the handset (utilizing its UAK) to ensure it is the genuine handset and not an intruder or an attempt to imitate the real handset.
- A procedure is available for rekeying with a new derived cipher key during a call.

https://t.me/learningnets

@cisco_in_persian_channel

Together, these layers of security should provide organizations the confidence that DECT headsets are genuine, provide privacy, and can avoid brute-force techniques to decipher encryption keys that could be used to intercept a conversation.

Summary

The transition to cloud-based services is underway. There are numerous business benefits for organizations as they make the transition to the cloud, especially to consume UC services. Benefits include new consumption models that make technology more affordable and technical innovation that helps organizations be more effective for how they collaborate. In many cases, a cloud-based UC service may provide more security than what is currently provided by on-premises systems. The reason is that Cisco provides many safeguards by default so UC services can be easily consumed and secured with a minimal effort by consumers of cloud services. Additionally, Cisco provides advanced security features so that consumers who have more stringent security policies can remain in compliance by enabling these features.

These features provide organizations with the confidence that when they make the transition to cloud for UC services, they do not have to trade the confidentiality, integrity, and availability that they require to conduct business operations. When an organization enables certain cloud-based features, there may be implications to personnel data. Current legislation regarding people's data includes the General Data Protection Regulation (GDPR) and the Biometric Information Privacy Act (BIPA). Organizations should investigate whether they need to implement safeguards and messaging to maintain compliance. Lastly, organizations do not have to choose whether to go all in on cloud or keep their on-premises architecture. They can operate a hybrid-cloud UC architecture in a functional and secure manner.

Additional Resources

https://trustportal.cisco.com/

www.cisco.com/c/en/us/about/trust-center.html

www.oasis-open.org/

https://oauth.net/2/

https://tools.ietf.org/html/rfc6749

https://tools.ietf.org/html/rfc7522

https://tools.ietf.org/html/rfc7643

https://tools.ietf.org/html/rfc7644

https://tools.ietf.org/html/rfc7009

https://t.me/learningnets

@cisco_in_persian_channel

Afterword

Reviewing ACME's Progress and Newly Secured UC Architecture

It has been a few grueling months for Anthony Starke. When we first met Anthony, he was new to UC security but he was optimistic and had the support from his management team to take the necessary steps to improve the security of ACME's UC environment. As we walked through Anthony's considerations, experiences, trials, and tribulations, we also leveraged examples that we have taken from real-world scenarios to help identify the real-world challenges that customers are facing. We have also shared many best practices that organizations can use in order to simplify the configuration of security features and solutions to help ease the deployment of various security capabilities that organizations can use in order to reduce the threat landscape.

Although not all organizations may have the luxury of taking a similar progression as Anthony, some may. On countless occasions, we have heard from organizational leadership that their people are their most precious resource. We believe that this is true, which is why we began with discussing physical security and life safety solutions, while also providing detailed configuration guidance on implementing solutions such as Cisco Emergency Responder. As we moved into the realm of network security, we discussed many considerations that the traditional UC engineer may not be familiar with but are important concepts to grasp as the UC engineer works with network engineers to ensure a wholistic approach to network security by implementing features such as 802.1x authentication and network access control (NAC) using the Cisco Identity Service Engine (ISE). It is certainly an important observation that we have made. When an organization is aligned and engineers work together with each other on projects, the outcome typically is faster and much more effective.

https://t.me/learningnets

@cisco_in_persian_channel

Over the rest of the chapters, we spoke about securing the core UC components. Speaking from experience, the use of X.509 certificates and Certificate Authorities is something that can be intimidating to UC engineers at first. More than half of the chapters in this book made mention of TLS security and X.509 certificates to help increase familiarity and ease concerns about working with certificates so that organizations use them to support secure connections. In total, 11 of the 12 chapters were spent discussing on-premise UC applications and highlighted certain security features that are enabled by default. In Chapter 12, "Securing Cloud and Hybrid Cloud Services," we discussed how the cloud is being used and how it can be a different delivery model for customers to consume UC services that are highly available and secure. A compelling fact about the Cisco Webex cloud services is that TLS and other security features are enabled by default. This opens up a range of questions that can be asked regarding the future of UC security for on-premise environments such as

- What type of workflows will Cisco develop to simplify how secure connections can be made over TLS?
- How many organizations will move their entire UC workloads to the cloud for simplicity of security?

First, perhaps the answer to simplify on-premise security is related to a tighter integration of Automatic Certificate Management Environment (ACME) protocol into on-premise applications as we discussed in Chapter 11, "Securing the Edge," with Expressway. ACME itself is a cloud service, so will that have any impact for on-premise UC applications?

Second, it seems that the market is demanding the innovations that can only be offered from cloud platforms such as virtual assistant, facial recognition, and so on. This brings up new questions around the future of security for cloud-based UC services such as

- Will the benefits of the innovative features that are being developed be outweighed by privacy considerations?
- What new security features will Cisco develop to mitigate concerns of privacy?

As discussed, moving UC workloads to the cloud does allow UC engineers to focus on other key areas of the business, such as application integration, user adoption, application development, and compliance to security standards. Perhaps the answer is a hybrid architecture, which our friend Anthony Starke was so excited about.

Circling back to ACME, after all of the chapters and technical solutions that we discussed, Figure 1 depicts the new UC architecture that our friend Anthony Starke was able to help them implement. Tables 1 and 2 list out the various applications and functions that ACME now has added into their UC architecture. When compared to where ACME was when we first began this journey, ACME has been able to improve its focus on UC security and also the health and safety of their employees across the organization. As you can imagine, ACME's COO, Dr. Nicholas Fury, and the rest of ACME leadership have been impressed with the work that Anthony, the UC team, and Jonah have been able to accomplish. Anthony's boss Jonah was recently promoted because of this work,

https://t.me/learningnets

@cisco_in_persian_channel

and Anthony was certainly recognized for his efforts. Therefore, Anthony is next in line to take on Jonah's old role and responsibilities, which is managing ACME's IT division of UC, security, data center, and application development teams. Now that Anthony has finished securing the UC environment, he is wondering which projects he will be asked to take on next. He has heard from ACME's COO that an upcoming project around improving the automation and orchestration of ACME's network is a high priority.

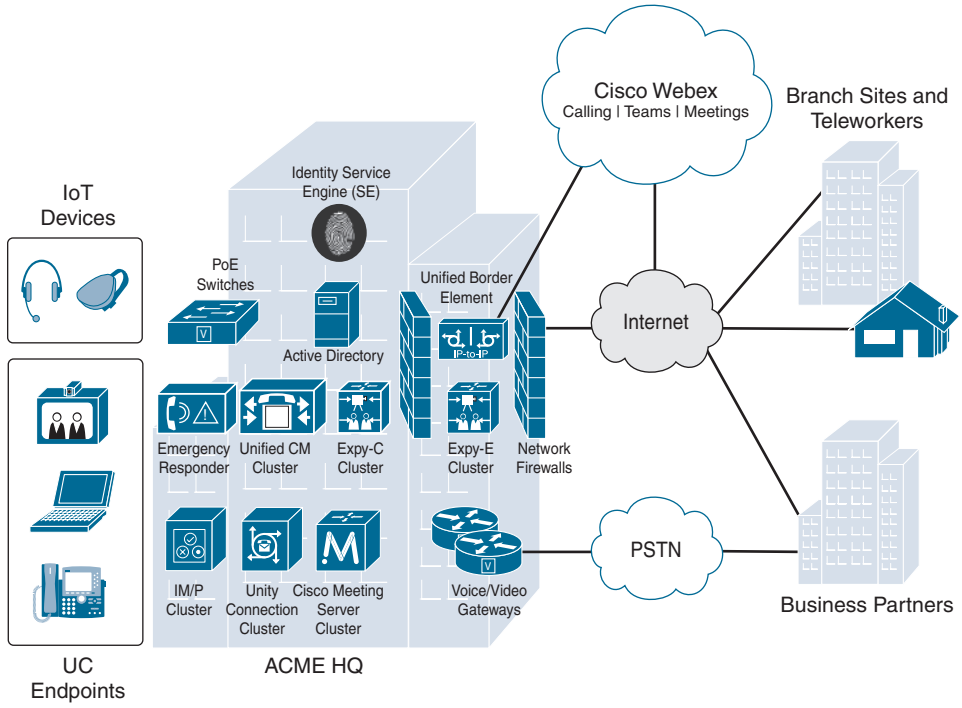


Figure 1 *ACME's Latest Network Topology*

Table 1 *ACME's Current UC Applications and Versions*

Application Name	Application Version	Application Role
Cisco Unified Communications Manager	12.5	Call control: VoIP and video devices, user provisioning
Cisco Unity Connection	12.5	Voicemail, basic ACD
Cisco Instant Messaging and Presence	12.5	Instant messages, presence, calling
Cisco Meeting Server	3.0	Audio/video conferencing
Cisco Emergency Responder	12.5	E911 location support
Cisco Expressway Core and Edge	12.5	Mobile and Remote Access for teleworkers; B2B connectivity

https://t.me/learningnets
@cisco_in_persian_channel

Table 2 *ACME's Current Network Devices and Versions*

Platform Type	Software Version	Platform Role
ISR 4400 router	IOS XE 16.9	PSTN gateway
ISR 4400 router	IOS XE 16.9	Cisco Unified Border Element
Catalyst 3850 and 9000 series switches	IOS XE 16.9	Campus switching, PoE
Identity Service Engine	3.0	Authentication Services, Network Access Control (NAC)

We sincerely thank you for taking the time to read through this book. We hope that you will communicate the principles that we have discussed with your co-workers and friends among your communities. Last but not least, we hope that you will use the principles in this book to go on and do great things like what our friends over at ACME were able to accomplish.

Sincerely,
Brett and Nik

https://t.me/learningnets

@cisco_in_persian_channel

Index

Numbers

802.1x authentication, 67–72, 236

911 services

E911, 28–29, 31

ACME case study, 16–17

ALI DB, 29

ANI, 29

CER, access point association, 38

CER, assigning ERL, 41–42

CER, call flows, 35–36

CER, call routing, 34–35, 43–47

CER, compliance, 49–51

CER, IP subnet tracking, 38

CER, LAN switches, 40

CER, managing, 49–51

CER, manual tracking, 38

CER, SNMPv2, 40–41

CER, SNMPv3, 40–41

CER, switch port tracking, 38

CER, unlocated phone assignments, 38

CER, verifying, 49–51

CER, wireless device location tracking, 42–43

computer-aided dispatch, 52–53

design, 36–38

ELIN, 29, 48–49

ERL, 29, 48–49

ERL, assigning, 41–42

ERL, creating, 38–43

ERL, network discovery, 42–43

InformaCast, 51

Instant Connect solutions, 52–53

Intrado solutions, 51

IP Mulitcast, 51

Kari's Law, 30

native E911 call routing, Cisco Unified CM, 31–34

PS-ALI, 29

PSAP, 28, 29, 47–49

RAY BAUM's Act, 30

RedSky solutions, 51

verifying ERL, 43

VoIP, 30–31

NG911, 28–29

https://t.me/learningnets

@cisco_in_persian_channel

A

access
CAC, CMS authentication, 345–346
IP-based PSTN access, Collaboration Edge, 386–388
MRA, 406–407
authentication, 413–415
authorization, 413–415
certificate requirements, 410–412
DNS, 407–409
firewall traversal, 412–413
ICE, 419–420
secure phone profiles, 416–417
token scopes/revocation, 418
troubleshooting, 418–419
NAC, 73–75
network access, 802.1x authentication, 236
role-based access
control groups, 170–171
Webex identity theft prevention, 437–438
root access, granting, 137–138
secure network access, 64–65
setting access, endpoint hardening, 275–276
Unified CM
access control groups, 171–172, 173–174
user rankings, 173
user access
Cisco Unity Connection, 311–316
endpoint hardening, 275
access point association, CER, 38
accounts
administrator accounts, CLI
changing OS/GUI/Database passwords, 148–151
creating, 149
security passphrases, 150–151
Cisco Emergency Responder user accounts, 163
Cisco Unity Connection user accounts, 162–163, 164–170
end user accounts
importing end users from LDAP directories, 175–176
PIN, 197–199
role assignments, 175
synchronization, 197–199
inactive accounts, disabling, 155–157
locking, 155–157
recovery procedures, 157–159
UAC
local UAC, strengthening, 163–164
order of operations, 163
policy creation, 164–170
Unified CM user accounts, 162–163
Default Credential Policies, 165–166
Enhanced Security Credential Policies, 165–166
Standard End User - Password Credential Policies, 165–167, 169
UAC policies, 164–170
user accounts
Cisco Emergency Responder, 163
Cisco Unity Connection, 162–163, 164–170

https://t.me/learningnets
@cisco_in_persian_channel

credential policies, 167–170
end users, importing from LDAP directories, 175–176
local UAC, strengthening, 163–164
role assignments, 175
Unified CM, 162–163
Unified CM, Default Credential Policies, 165–166
Unified CM, Enhanced Security Credential Policies, 165–166
Unified CM, Standard End User - Password Credential Policies, 165–167, 169
Unified CM, UA policies, 164–170

ACL (Access Control Lists), 85
cluster communications, securing, 100
SGACL, 62

ACME case study, 11–13
application security, 161
call routing, jurisdictional issues, 44–45
Cisco Unity Connection, 306
cloud/hybrid cloud services, 428
CMS, 339–340
conferencing, secure, 339–340
core UC applications, 125–126
life safety assessments, 16–17
media encryption, 217–218
network security, 55–57
physical security assessments, 16–17
secure signaling, 217–218
telecommuting, 384
UAC policies, 164
UC deployments, 89–90, 91–93
Unified CM security, 273

ACS (Assertion Consumer Service), SAML, 188

Ad Hoc conferencing, 278, 296–297

administrative (management) controls, 18

administrator accounts, CLI
changing OS/GUI/Database passwords, 148–151
creating, 149
security passphrases, 150–151

aging
messages, 313–316
passwords, 151–152

alarms, 209
default alarm locations, 211
Syslog Agent, 210

ALI DB (Automated Location Identification Database), 29

alternate contact number restrictions, TUI voicemail, 330–333

ANI (Automatic Number Identification), 29

API (Application Programming Interfaces), CMS, 367–369

application servers, credential change service, 200–201

applications
Cisco Emergency Responder, 127
Cisco IM&P, 127
Cisco Unified CM, 126
Cisco Unity Connection, 126
cluster communications, securing, 98–99
MAM, 455–456
NBAR, 388–390
patch/version management, 136–137
UC core applications, defining, 126–127
UI application passwords, 148

https://t.me/learningnets
@cisco_in_persian_channel

ARP (Address Resolution Protocol)
Gratuitous ARP, 275
inspection, 80
assertions, SAML, 188
asymmetric cryptography, 223
audit logs, 211–212
authentication
802.1x authentication, 67–72, 236
auto-key authentication, 99
caller-ID authentication, 1
CMS, CAC, 345–346
EAP, 67
intermediate CA, authentication into IOS trustpoints, 282–283
LDAP
directories, 184
imported end users, 184–186
MAB, 6, 72–73
meetings, Webex, 460
MRA, 413–415
NTP, 99–100
enabling, 101–102
verifying, 100–101, 102
OAuth
authorization, 255–258
enabling, 258–261
SAML, 252–255
SIP, 250–252
root CA, authentication into IOS trustpoints, 283–285
SAML, 188
symmetric-key authentication, 99
authorization
MRA, 413–415
OAuth, 255–258
AuthZ server configuration, Cisco Unity Connection, 259–261
auto-key authentication, 99

B

B2B (Business-to-Business) communication
Collaboration Edge, 422–423
Expressway, 403–406, 420–423
backups, CMS, 380–381
badges, physical security, 20
banners
login banners, 202–203
untrusted digital certificate banners, 223–224
barges, Meet-Me secure conferencing, 297
bollards, 19
business models, changing, UC security, 2
butt sets, 26

C

CA (Certificate Authorities), 224–225
certificate trustpoints, 112–114
chains of trust, 227–229
intermediate CA, authentication into IOS trustpoints, 282–283
offline CA, 236
online CA, 236
root CA, authentication into IOS trustpoints, 283–285
self-signed certificates, 225
single-tier CA hierarchies, 225–226
three-tier CA hierarchies, 227–228
two-tier CA hierarchies, 226–227
cable plants
butt sets, 26
caller-ID spoofing, 26
PDS, 26

https://t.me/learningnets
@cisco_in_persian_channel

physical security, 26–27
PoE, 26
PoLRE, 26–27
VoIP eavesdropping, 26–27
CAC (Common Access Cards), CMS authentication, 345–346
call bridges, CMS, 340, 366
call handlers
system transfers, 333–336
traffic reports, 310
caller-ID
authentication, 1
spoofing, 1, 26
calls
inbound/outbound calls, CMS, 370–374
routing
CAMA, 44
CER, 43–47
POTS lines, 43–44
PRI, 43–44
SIP, 44
spoofing, 3
Webex Calling, 433–437
CAMA (Centralized Automatic Message Accounting), 44
campus deployment models, 90, 91–92
CAPF (Certificate Authority Proxy Function), 235, 236
CTI, securing, 118–123
mixed mode clusters, 246
CC (Common Criteria, ISO/IEC 15408) compliance, VOS, 144–147
CCTV surveillance, physical security, 20–21
centralized deployment models, 90, 92–95
CER (Cisco Emergency Responders)
access point association, 38
assigning ERL, 41–42
call routing, 43–47
compliance, 49–51
default ERL, 38
E911
call flows, 35–36
call routing, 34–35
ELIN, 48–49
ERL, 48–49
IP subnet tracking, 38
LAN switches, 40
managing, 49–51
manual tracking, 38
SNMPv2, 40–41
SNMPv3, 40–41
switch port tracking, 38
unlocated phone assignments, 38
verifying, 49–51
wireless device location tracking, 42–43
certificates
basic constraints, 229
CAPF, 235, 236
chains, LDAP directories, 177–180
chains of trust, 286–287
CMS certificates
assigning, 360–367
verifying, 356–359
components, 229, 231
extended key usage, 231–232
identity certificates, importing for secure DSP, 285
IP phones, 235–236
key usage, 229–231
lists, IPSec, 105

https://t.me/learningnets
@cisco_in_persian_channel

LSC
applying to phones, 261–264
CAPF, 236
MRA requirements, 410–412
offline CA, 236
online CA, 236
root CA certificates, CUBE, TLS connectivity, 397–398
self-signed certificates, 225
CFB (Call Forward Busy)
CSR, 280–281
trustpoints, 280
chains of custody, 21–22
chains of trust
CA certificates, 227–229
certificates, 286–287
CSR, 104–105
changing
business models, UC security, 2
OS/GUI/Database passwords, 148–151
Cisco Catalyst switches, switchport security, 6
Cisco Emergency Responder, 127
default user accounts, 163
integration, securing, 118–123
JTAPI, enabling, 119–123
Cisco IM&P, 127
Cisco Meeting Server
SIP media encryption, 293
Webadmin PKI, 292–293
Cisco R-Series R42612 Rack, 24
Cisco Unified CM. *See* Unified CM
Cisco Unity Connection, 126
AuthZ server configuration, 259–261
baseline security, 306–311
default user accounts, 162–163
encryption, 307, 311
end user accounts, synchronization, 197–199
FIPS, 308
logging, 308, 311
messages
aging, 313–316
security, 323–325
monitoring, 308
OAuth, 258–261
PIN, 169, 312–313
reports, 308–311
secure Unified CM integration, 316–323
security module functionality, 306–307
system transfers, 333–336
toll fraud, preventing, 325
PIN logins, 325
restriction tables, 325–336
TUI voicemail, alternate contact number restrictions, 330–333
UAC policies, 164–170
upgrades, 307
user access, securing, 311–316
wildcard characters, 327–328
CLI (Command-Line Interface)
administrator accounts
changing OS/GUI/Database passwords, 148–151
creating, 149
export restrictions, verifying, 218–219
mixed mode clusters, verifying, 247
OS lockdowns, 147–148
changing OS/GUI/Database passwords, 148–151
disabling inactive accounts, 155–157
locking accounts, 155–157

https://t.me/learningnets
@cisco_in_persian_channel

password aging, 151–152
password complexity, 152–155
recovering accounts, 148–151
closed mode, 802.1x authentication, 72
cloud/hybrid cloud services, 427
business drivers, 428–430
compliance, 430–431
DLP, 451–454
EMM, 455
facial recognition, 464
FedRAMP Moderate, 431
firewalls, 432–433
governance, 430–431
hosted clouds, 430
hybrid clouds, 430
identity theft prevention, 437
Directory Connector, 438–439
OAuth 2.0, 441–443
onboarding, 437–438, 443–446
role-based access, 437–438
SAML 2.0, 440–441
SCIM, 443–446
IoT security, 467–470
ISO 27001, 431
ISO 27017, 431
ISO 27018, 431
MAM, 455–456
MDM, 455
meeting management/security, 456–457
data at rest, 463
end-to-end encryption, 461–462
meeting authentication, 460
in-meeting privacy controls, 462–463
scheduled/unscheduled meetings, 457–459
security across emerging features, 463–464
Meeting Transcription, 467
messaging service security, 446–447
content management, 448–451
end-to-end message encryption, 447–448
external communications, 448–451
native clouds, 429
PAC files, 432
People Insights, 465–466
SOC 2 type II, 431
SOC 3, 431
transport security/compliance, 432–433
Webex Assistant, 466–467
Webex Calling, 433–437
WPAD, 432
cluster communications, securing
ACL, 100
applications, 98–99
endpoints, 98–99
ICCS, 96–98, 103–110
IPSec, 98–99
NTP, 99–100
clusters, mixed mode, 245–247
CM (Communications Manager)
mixed-mode operations, 8
native E911 call routing, 31–34
CMS (Cisco Meeting Server), 339, 340
administration roles/permissions, 343–344
API, 367–369
backups, 380–381
CAC authentication, CAC, 345–346

https://t.me/learningnets
@cisco_in_persian_channel

call bridges, 340, 366
certificates
assigning, 360–367
verifying, 356–359
Collaboration Edge functionality, 341–342
combined server bundle, 292
components, 340–341
connectivity, 341–342, 351–353
databases, 340, 354–356
dictionary attacks, preventing, 344
DNS, 348–349
DNSSEC, 349–351
Expressway TURN servers, 342
firewalls, 346–347
inbound/outbound calls, 370–374
infrastructure security, 347–351
logging, 377–378
managing, 377–381
meeting space security, 377
MMP Command-Line Reference, 347
NTP, 348
OS
hardening, 342–347
VOS comparisons, 343
passwords, 344
recording servers, 340
secure conferencing, 290–297
SNMP, 378–379
streaming servers, 340
TLS, 359–360
Unified CM configuration, 374–376
uploaders, 340
visibility, 377–381
VQ Conference Manager, 380
Vyopta, 380
web bridges, 340, 367
work hour access restrictions, 345
XMPP servers, 340, 341
Collaboration, design zone for, 8
Collaboration Edge, 383
architecture, 384–385
B2B connectivity, 422–423
CMS functionality, 341–342
compliance, 423–424
CPL, 420–421
CUBE
deploying, 387–390
dial-peers, 392–395
IP-based PSTN access, 386–388
NBAR, 388–390
session control/protection, 392–395
TDoS protection, 391–392
TLS connectivity, 395–401
toll fraud prevention, 390–391
defending against attacks, 420–422
Expressway, 403–406, 420–423
IP-based PSTN access, 386–388
monitoring, 423–424
MRA, 406–420
VPN-based telework solutions, 402
VPN-less telework solutions, 402–403
complexity
passwords, 152–155
UC security, 5–6
high security features, 6, 7
low security features, 6, 7
medium security features, 6, 7
minimizing, 7–10
compliance
CER, 49–51
Collaboration Edge, 423–424

https://t.me/learningnets
@cisco_in_persian_channel

computer-aided dispatch, E911, 52–53
conference bridges, SCCP, 288–289
Conference Now, 298
conferencing, secure, 276–278
Ad Hoc conferencing, 278, 296–297
CMS, 290–297, 339, 340
administration roles/ permissions, 343–344
API, 367–369
backups, 380–381
CAC authentication, 345–346
call bridges, 340, 366
certificate assignments, 360–367
certificate verification, 356–359
Collaboration Edge functionality, 341–342
components, 340–341
connectivity, 341–342, 351–353
databases, 340, 354–356
DNS, 348–349
DNSSEC, 349–351
firewalls, 346–347
inbound/outbound calls, 370–374
infrastructure security, 347–351
logging, 377–378
managing, 377–381
meeting space security, 377
MMP Command-Line Reference, 347
NTP, 348
OS hardening, 342–347
passwords, 344
preventing dictionary attacks, 344
recording servers, 340
SNMP, 378–379
streaming servers, 340
TLS, 359–360
Unified CM configuration, 374–376
uploaders, 340
visibility, 377–381
VQ Conference Manager, 380
Vyopta, 380
web bridges, 340, 367
work hour access restrictions, 345
XMPP servers, 340, 341
Conference Now, 298
DSP, 278–290
Meet-Me secure conferencing, 297–298
smart licensing, 298–302
configuring
AuthZ server configuration, Cisco Unity Connection, 259–261
IPSec, 117
policies, 106–108
transform sets, 116
ISAKMP policies, 115–116
LDAP directories, 180–181
password aging, 151–152
secure phone profiles, 261–264
SIP trunk security profiles, 294–296
Unified CM, 374–376
Unified CM SSO with SAML, 191–197
connectivity
B2B connectivity, Collaboration Edge, 422–423
ICE, MRA, 419–420
content management, messaging, 448–451

https://t.me/learningnets
@cisco_in_persian_channel

continuous monitoring, 86
contracts, SDA, 62
corrective controls, 21
CoT (Circle of Trust), SAML, 188
CPL (Call-Processing Language), Collaboration Edge, 420–421
credential change service, application servers, 200–201
credential policies
default updates, 170
defaults, 169–170
settings, 167–169
crypto maps, IPSec tunnels, 116–117
cryptography, ECC, 233–235
CSR (Certificate Signing Requests)
CFB, 280–281
chains of trust, 104–105
CUBE, TLS connectivity, 395–396
IPSec
CSR generation, 104–105
key usage extensions, 103–104
offline device CSR generation, 268–270
voice gateways, securing, 112
CSR 12.0, certificate key usage requirements, 229–231
CTI (Computer Telephony Integration), Cisco Emergency Responder integration, 118–123
CTL files, 247–250
CTL Provider service, mixed mode clusters, 246
CUBE (Cisco Unified Border Element)
deploying, 387–390
dial-peers, 392–395
IP-based PSTN access, 386–388
NBAR, 388–390
session control/protection, 392–395
TDoS protection, 391–392
TLS connectivity, 395–401
toll fraud prevention, 390–391
custody, chains of, 21–22

D

DAI (Dynamic ARP Inspection), 80
data at rest, protecting with Webex, 463
data centers, physical security, 22–24
ESD, 25
power plants, 24–25
databases, CMS, 340
Default Credential Policies, Unified CM, 165–166
defense-in-depth, physical security, 20
deployments
campus deployment models, 90, 91–92
centralized deployment models, 90, 92–95
cluster communications, securing
ACL, 100
applications, 98–99
endpoints, 98–99
ICCS, 96–98
IPSec, 98–99
NTP, 99–100
CUBE, 387–390
distributed deployment models, 91, 95
E-SRST, 93–94
hybrid deployment models, 91
SRST, 93–94
VM, 91–92
design zone for Collaboration, 8

https://t.me/learningnets
@cisco_in_persian_channel

detection phase, physical security, 20–21
DHCP (Dynamic Host Configuration Protocol)
snooping, 79, 80
starvation attacks, 78–79
dial-peers, CUBE, 392–395
dictionary attacks, CMS, 344
digital certificates
basic constraints, 229
CAPF, 235, 236
CMS certificates, verifying, 356–359
components, 229, 231
extended key usage, 231–232
IP phones, 235–236
key usage, 229–231
key usage requirements for CSR 12.0, 229–231
LSC
applying to phones, 261–264
CAPF, 236
offline CA, 236
online CA, 236
root CA certificates, CUBE, TLS connectivity, 397–398
untrusted digital certificate banners, 223–224
directories, LDAP
authentication, 184
certificate chains, 177–180
configuring, 180–181
importing end users from, 175–176
overview, 177
SSL certificates, 178–180
user attributes, 183–184
Directory Connector, 438–439
disabling
inactive accounts, 155–157
IPSec policies, 110
disaster recovery planning, 213–214
discovery of private information, malicious, 3
distributed deployment models, 91, 95
DLP (Data Loss Prevention), Webex, 451–454
DNA-C (DNA Center), 60, 62–64
DNS (Domain Name System), 83–84
CMS, 348–349
MRA, 407–409
DNSSEC, CMS, 349–351
domain configurations, voice gateway security, 112
DoS attacks, 3, 100
double tagging, VLAN hopping, 77
DRS (Disaster Recovery Service), 213–214
DSP (Digital Signal Processors)
DSPFarm, enabling, 287–288
identity certificates, importing, 285
secure conferencing, 278–290

E

E911 (Enhanced 911), 28–29, 31
ACME case study, 16–17
ALI DB, 29
ANI, 29
call routing (native E911), Cisco Unified CM, 31–34
CER
access point association, 38
assigning ERL, 41–42

https://t.me/learningnets
@cisco_in_persian_channel

call flows, 35–36
call routing, 34–35, 43–47
compliance, 49–51
IP subnet tracking, 38
LAN switches, 35–36
managing, 49–51
manual tracking, 38
SNMPv2, 40–41
SNMPv3, 40–41
switch port tracking, 38
unlocated phone assignments, 38
verifying, 49–51
wireless device location tracking, 42–43
computer-aided dispatch, 52–53
design, 36–38
ELIN, 29, 48–49
ERL, 29, 48–49
assigning, 41–42
creating, 38–43
network discovery, 42–43
verifying, 43
InformaCast, 51
Instant Connect solutions, 52–53
Intrado solutions, 51
IP Mulitcast, 51
Kari's Law, 30
PS-ALI, 29
PSAP, 28, 29, 47–49
RAY BAUM's Act, 30
RedSky solutions, 51
VoIP, 30–31
EAP (Extensible Authentication Protocol), 67
eavesdropping, 3, 26–27
ECC (Elliptical Curve Cryptography), 233–235
Edge (Collaboration), 383
architecture, 384–385
B2B connectivity, 422–423
CMS functionality, 341–342
compliance, 423–424
CPL, 420–421
CUBE
deploying, 387–390
dial-peers, 392–395
IP-based PSTN access, 386–388
session control/protection, 392–395
TDoS protection, 391–392
TLS connectivity, 395–401
toll fraud prevention, 390–391
defending against attacks, 420–422
Expressway, 403–406, 420–423
IP-based PSTN access, 386–388
monitoring, 423–424
MRA, 406–420
VPN-based telework solutions, 402
VPN-less telework solutions, 402–403
ELIN (Emergency Location Identification Numbers), 29, 31–34, 48–49
EMM (Enterprise Mobility Management), 455
employees, remote, 2. ***See also*** **telework**
enabling
CC (Common Criteria, ISO/IEC 15408) compliance, VOS, 144–147
DSPFarm, 287–288
Enhanced Security Mode, VOS, 143–144, 147
FIPS 140–2, VOS, 139–142, 147
JTAPI, Cisco Emergency Responder integration, 119–123
NTP authentication, 101–102

https://t.me/learningnets
@cisco_in_persian_channel

OAuth, 258–261
password complexity, 152–155
encryption
Cisco Unity Connection, 307, 311
end-to-end encryption
meetings, Webex, 461–462
messages, 447–448
encryption, media, 217
asymmetric cryptography, 223
CA, 224–225
chains of trust, 227–229
self-signed certificates, 225
single-tier CA hierarchies, 225–226
three-tier CA hierarchies, 227–228
two-tier CA hierarchies, 226–227
ECC, 233–235
encryption licensing, 218–221
endpoint registration, 239
CTL files, 247–250
ITL, 239–245
mixed mode clusters, 245–247
SBD, 239–245
export restrictions, 218–221
FIPS, 222
man-in-the-middle attacks, 223–224
OAuth
authorization, 255–258
enabling, 258–261
SAML, 252–255
SIP, 250–252
phone profiles, security
applying to phones, 264–270
configuring, 261–264
PKI
overview, 222–229
Unified CM, 229–232
SIP, Cisco Meeting Server, 293
SRTP, 218
TFTP file encryption, 237–238
untrusted digital certificate banners, 223–224
end user accounts
importing end users from LDAP directories, 175–176
PIN, 197–199
role assignments, 175
synchronization, 197–199
endpoints
cluster communications, securing, 98–99
hardening, 274, 275
configuring settings, 274–275
Gratuitous ARP, 275
PC ports, 276
PC Voice VLAN, 275
setting access, 275–276
web access, 275
registration, 239
end-to-end encryption
meetings, Webex, 461–462
messages, 447–448
Enhanced Security Credential Policies, Unified CM, 165–166
Enhanced Security Mode, VOS, 143–144, 147
ERL (Emergency Response Location), 29, 48–49
access point association, 38
assigning, 41–42

https://t.me/learningnets
@cisco_in_persian_channel

creating, 38–43
default ERL, 38
IP subnet tracking, 38
manual tracking, 38
network discovery, 42–43
switch port tracking, 38
unlocated phone assignments, 38
verifying, Cisco Emergency Responder, call flows, 43
ESD (Electrostatic Discharge), 25
E-SRST (Enhanced-Survivable Remote Site Telephony), 93–94
Ethernet
PoE, 26
PoLRE, 26–27
export restrictions, 218–221
Expressway, B2B communication, 403–406, 420–423
Expressway TURN servers, CMS functionality, 342
external communications, securing with Webex, 448–451

F

facial recognition, Webex, 464
FedRAMP Moderate, cloud/hybrid cloud services, 431
file encryption, TFTP, 237–238
FIPS (Federal Information Processing Standard)
Cisco Unity Connection, 308
FIPS 140–2, 139–142, 147, 308
secure signaling/media encryption, 222
firewalls, 84–85
cloud/hybrid cloud services, 432–433
CMS, 346–347
MRA firewall traversal, 412–413
forensic controls, 21–22
full mesh, IPSec policies, 108–109
full-body scanners, physical security, 20

G

granting root access, 137–138
Gratuitous ARP, 275
GUI (Graphical User Interface)
export restrictions, verifying, 218–220
lockdowns
login banners, 202–203
screen timeouts, 201–202
mixed mode clusters, verifying, 247

H

hackers, 1
hardening
CMS OS, 342–347
endpoints, 274, 275
configuring settings, 274–275
Gratuitous ARP, 275
PC ports, 276
PC Voice VLAN, 275
setting access, 275–276
web access, 275
high security features, 6, 7
host names, voice gateway security, 112
hosted clouds, 430
HTTPS, VOS, 134
hybid cloud services, 427

https://t.me/learningnets
@cisco_in_persian_channel

business drivers, 428–430
compliance, 430–431
defined, 430
DLP, 451–454
EMM, 455
facial recognition, 464
FedRAMP Moderate, 431
firewalls, 432–433
governance, 430–431
identity theft prevention, 437
Directory Connector, 438–439
OAuth 2.0, 441–443
onboarding, 437–438, 443–446
role-based access, 437–438
SAML 2.0, 440–441
SCIM, 443–446
IoT security, 467–470
ISO 27001, 431
ISO 27017, 431
ISO 27018, 431
MAM, 455–456
MDM, 455
meeting management/security, 456–457
data at rest, 463
end-to-end encryption, 461–462
meeting authentication, 460
in-meeting privacy controls, 462–463
scheduled/unscheduled meetings, 457–459
security across emerging features, 463–464
Meeting Transcription, 467
messaging service security, 446–447
content management, 448–451
end-to-end message encryption, 447–448
external communications, 448–451
PAC files, 432
People Insights, 465–466
SOC 2 type II, 431
SOC 3, 431
transport security/compliance, 432–433
Webex Assistant, 466–467
Webex Calling, 433–437
WPAD, 432
hybrid deployment models, 91

I

ICCS (Intra-Cluster Communication Signaling), 96–98, 103–110
ICE, MRA, 419–420
identity certificates, importing for secure DSP, 285
identity theft
defined, 3
Webex identity theft prevention, 437
Directory Connector, 438–439
OAuth 2.0, 441–443
onboarding, 437–438, 443–446
role-based access, 437–438
SAML 2.0, 440–441
SCIM, 443–446
inactive accounts, disabling, 155–157
inbound/outbound calls, CMS, 370–374
InformaCast, E911, 51
installation packages, patch/version management, 136–137

https://t.me/learningnets
@cisco_in_persian_channel

Instant Connect, E911 solutions, 52–53
intermediate CA, authentication into IOS trustpoints, 282–283
intermediate certificates, 68
Intrado, E911 solutions, 51
IOS trustpoints
intermediate CA authentication, 282–283
root CA authentication, 283–285
IoT security, Webex, 467–470
IP Mulitcast, E911, 51
IP phones
802.1x authentication, 236
CAPF and LSC operations, 236
certificates, 235–236
LSC, applying to phones, 264–270
mTLS, 236
secure phone profiles
applying to phones, 264–270
configuring, 261–264
IP subnet tracking, CER, 38
IP-based video surveillance, physical security, 20–21
IPSec, 98–99
certificate lists, 105
configuring, 117
CSR
generation, 104–105
key usage extensions, 103–104
ICCS, securing, 103–110
key usage extensions, 115
policies
configuring, 106–108
creating, 108
disabling, 110
full mesh, 108–109
NTP, 110
securing signaling protocols, 111–117
securing voice gateways, 111–117
upgrades, 110
verifying, 109–110
transform sets, configuring, 116
tunnels
certificate configuration, 113
certificate signing requests, 114–115
crypto maps, 116–117
defining interesting traffic, 115
RSA keys, 112
trustpoints, 112–114
ISAKMP policies, configuring, 115–116
ISE (Identity Services Engine), 60–61, 73–75
ISO 27001, cloud/hybrid cloud services, 431
ISO 27017, cloud/hybrid cloud services, 431
ISO 27018, cloud/hybrid cloud services, 431
ITL (Initial Trust Lists), 239–245

J

JTAPI, Cisco Emergency Responder integration security, 118–123

K

Kari's Law, 30

https://t.me/learningnets
@cisco_in_persian_channel

L

LA (Local Agents), DRS, 214
LAN switches, CER, 40
layer 2 segmentation, 58
layer 3 segmentation, 58–59
LDAP (Lightweight Directory Access Protocol)
directories
authentication, 184
certificate chains, 177–180
configuring, 180–181
importing end users from, 175–176
overview, 177
SSL certificates, 178–180
user attributes, 183–184
imported end user authentication, 184–186
synchronization, 182–183
licensing
encryption, 218–221
smart licensing, 298–302
life and safety, 16–17, 28. *See also* physical security
E911, 28–29, 31
ACME case study, 16–17
ALI DB, 29
ANI, 29
CER, access point association, 38
CER, assigning ERL, 41–42
CER, call flows, 35–36
CER, call routing, 34–35, 43–47
CER, compliance, 49–51
CER, IP subnet tracking, 38
CER, LAN switches, 40
CER, managing, 49–51
CER, manual tracking, 38
CER, SNMPv2, 40–41
CER, SNMPv3, 40–41
CER, switch port tracking, 38
CER, unlocated phone assignments, 38
CER, verifying, 49–51
CER, wireless device location tracking, 42–43
computer-aided dispatch, 52–53
design, 36–38
ELIN, 29, 48–49
ERL, 29, 48–49
ERL, assigning, 41–42
ERL, creating, 38–43
ERL, network discovery, 42–43
InformaCast, 51
Instant Connect solutions, 52–53
Intrado solutions, 51
IP Mulitcast, 51
Kari's Law, 30
native E911 call routing, Cisco Unified CM, 31–34
PS-ALI, 29
PSAP, 28, 29, 47–49
RAY BAUM's Act, 29, 30
RedSky solutions, 51
verifying ERL, 43
VoIP, 30–31
NG911, 28–29
Linux servers
UC Appliance comparisons, 127–134
VOS comparisons, 127–134
LMR (Land Mobile Radios), 52

https://t.me/learningnets
@cisco_in_persian_channel

local UAC, strengthening, 163–164
location tracking, wireless devices, 42–43
locking
accounts, 148–151, 155–157
GUI
login banners, 202–203
screen timeouts, 201–202
OS via CLI, 147–148
accounts, 148–151, 155–157
passwords, 148–155
logging
audit logs, 211–212
Cisco Unity Connection, 308, 311
CMS, 377–378
logical (technical) controls, 19
login banners, 202–203
low security features, 6, 7
low-impact mode, 802.1x authentication, 72
LSC (Locally Significant Certificates)
applying to phones, 261–264
CAPF, 236

M

MA (Master Agents), DRS, 214
MAB (MAC Authentication Bypass), 6, 72–73
mailboxes, size quotas, 314–315
malicious discovery of private information, 3
MAM (Mobile Application Management), 455–456
management (administrative) controls, 18
management tools, 10–11
managing
CER, 49–51
CMS, 377–381
meetings, Webex, 456–457
data at rest, 463
end-to-end encryption, 461–462
meeting authentication, 460
in-meeting privacy controls, 462–463
scheduled/unscheduled meetings, 457–459
message content with Webex, 448–451
mobile devices
EMM, 455
MAM, 455–456
MDM, 455
patch/version management, 136–137
man-in-the-middle attacks, 100, 223–224
manual tracking, CER, 38
MDM (Mobile Device Management), 455
media encryption, 217
asymmetric cryptography, 223
CA, 224–225
chains of trust, 227–229
self-signed certificates, 225
single-tier CA hierarchies, 225–226
three-tier CA hierarchies, 227–228
two-tier CA hierarchies, 226–227
ECC, 233–235
encryption licensing, 218–221

https://t.me/learningnets
@cisco_in_persian_channel

endpoint registration, 239
CTL files, 247–250
ITL, 239–245
mixed mode clusters, 245–247
SBD, 239–245
export restrictions, 218–221
FIPS, 222
man-in-the-middle attacks, 223–224
OAuth
authorization, 255–258
enabling, 258–261
SAML, 252–255
SIP, 250–252
phone profiles, security
applying to phones, 264–270
configuring, 261–264
PKI
overview, 222–229
Unified CM, 229–232
SIP, Cisco Meeting Server, 293
SRTP, 218
TFTP file encryption, 237–238
untrusted digital certificate banners, 223–224

media tampering, 3

medium security features, 6, 7

meeting space security
CMS, 377
Webex, 456–457
data at rest, 463
end-to-end encryption, 461–462
meeting authentication, 460
in-meeting privacy controls, 462–463
scheduled/unscheduled meetings, 457–459

Meeting Transcription, Webex, 467

Meet-Me secure conferencing, 297–298

messages
aging, 313–316
content management with Webex, 448–451
end-to-end message encryption, 447–448
mailbox size quotas, 314–315
messaging service security, 446–447
content management, 448–451
end-to-end message encryption, 447–448
external communications, 448–451
voicemail messages, securing, 323–325

metadata, SAML, 188

MIB (Management Information Bases), 204

MIC (Manufacturing Installed Certificates), 68–69

micro segmentation, 59–64

mixed mode clusters, 245–247

mixed-mode operations, 8

MLTS (Multiline Telephone Systems), 30

MMP Command-Line Reference, CMS, 347

mobility management
EMM, 455
MAM, 455–456
MDM, 455

monitor mode, 802.1x authentication, 72

monitoring
Cisco Unity Connection, 308

https://t.me/learningnets

@cisco_in_persian_channel

Collaboration Edge, 423–424
continuous monitoring, 86
system monitoring, SNMP, 204–208
monitoring tools
RTMT, 10–11
UCTM, 10–11
MRA (Mobile and Remote Access), 406–407
authentication, 413–415
authorization, 413–415
certificate requirements, 410–412
DNS, 407–409
firewall traversal, 412–413
ICE, 419–420
phone profiles, security, 416–417
token scopes/revocation, 418
troubleshooting, 418–419
mTLS, Unified CM and IP phones, 236
Multicast, E911, 51

N

NAC (Network Access Control), 73–75
native clouds, 429
native E911 call routing, Cisco Unified CM, 31–34
NBAR (Network-Based Application Recognition), 388–390
NENA (National Emergency Number Association), 28
network
network
networks
access
802.1x authentication, 236
network security, 64–65
security, 55, 57
802.1x authentication, 67–72
access, 64–65
ACL, 85
ARP inspection, 80
continuous monitoring, 86
DAI, 80
DHCP snooping, 79, 80
DHCP starvation attacks, 78–79
DNS, 83–84
firewalls, 84–85
layer 2 segmentation, 58
layer 3 segmentation, 58–59
MAB, 72–73
micro segmentation, 59–64
NAC, 73–75
NTP, 80–82
port security, 65–67
security features (unified), 75–76
VLAN hopping, 77–78
virtual networks, SDA, 62
VLAN
hopping, 77–78
layer 2 segmentation, 58
PC Voice VLAN, 275
VoWLAN, 27
VPN
VPN-based telework solutions, 402
VPN-less telework solutions, 402–403
NFPA (National Fire Protection Association)
NFPA 70: NEC, 24–25
NFPA 110, 24–25
physical security, 24–25

https://t.me/learningnets
@cisco_in_persian_channel

NG911 (Next Generation 911), 28–29
NIST framework, 8
nonsystem numbers, system transfers, 333–336
nontrivial passwords, 168
nontrivial phone PIN, 312–313
NTP (Network Time Protocol), 80–82
authentication, 99–100
enabling, 101–102
verifying, 100–101, 102
CMS, 348
IPSec policies, 110

O

OAuth
secure signaling/media encryption
authorization, 255–258
enabling, 258–261
SAML, 252–255
SIP, 250–252
Webex identity theft prevention, 441–443
offline CA, 236
onboarding, Webex identity theft prevention, 437–438, 443–446
online CA, 236
Operation Desert Storm, tiered security, 9–10
OS (Operating Systems). ***See also*** **VOS**
CMS OS
hardening, 342–347
VOS comparisons, 343
lockdowns via CLI, 147–148
changing OS/GUI/Database passwords, 148–151
disabling inactive accounts, 155–157
locking accounts, 155–157
password aging, 151–152
password complexity, 152–155
recovering accounts, 148–151
outbound/inbound calls, CMS, 370–374
Outcall Billing Detail reports, 310
Outcall Billing Summary reports, 310

P

PAC files, 432
packet captures, SNMP, 205–206
passphrases, security, 148, 150–151
passwords
aging, configuring, 151–152
Cisco Unity Connection PIN, 169
CMS, 344
complexity, 152–155
nontrivial passwords, 168
OS/GUI/Database passwords, changing, 148–151
Standard End User - Password Credential Policies, Unified CM, 165–167, 169
UI application passwords, 148
patch/version management, 136–137
PC ports, endpoint hardening, 276
PC Voice VLAN, 275
PDS (Protective Distribution Systems), 26
People Insights, Webex, 465–466
Phone Interface Failed Logon reports, 309
phone profiles, security
applying to phones, 264–270

https://t.me/learningnets
@cisco_in_persian_channel

configuring, 261–264
MRA, 416–417
physical controls, 19
physical security, 17. *See also* life and safety
ACME case study, 16–17
badges, 20
bollards, 19
cable plants, 26–27
CCTV surveillance, 20–21
corrective controls, 21
data centers, 22–24
ESD, 25
power plants, 24–25
defense-in-depth, 20
detection phase, 20–21
forensic controls, 21–22
full-body scanners, 20
IP-based video surveillance, 20–21
management (administrative) controls, 18
physical controls, 19
piggybacking, 20
preparation phase, 17–19
prevention phase, 19–20
recovery controls, 21
response phase, 21–22
scanners, 20
technical (logical) controls, 19
turnstiles, 20
UC environments, 22
piggybacking, 20
PIN (Personal Identification Numbers)
Cisco Unity Connection, 312–313, 325
end user accounts, 197–199
nontrivial phone PIN, 312–313
toll fraud prevention, 325
PKI (Public Key Infrastructure)
CA, 224–225
chains of trust, 227–229
self-signed certificates, 225
single-tier CA hierarchies, 225–226
three-tier CA hierarchies, 227–228
two-tier CA hierarchies, 226–227
man-in-the-middle attacks, 223–224
overview, 222–229
Unified CM, 229–232
Webadmin PKI, 292–293
PoE (Power over Ethernet), 26
policies
IPSec policies
configuring, 106–108
creating, 108
disabling, 110
full mesh, 108–109
NTP, 110
securing signaling protocols, 111–117
securing voice gateways, 111–117
upgrades, 110
verifying, 109–110
ISAKMP policies, configuring, 115–116
UAC policies, creating, 164–170
Unified CM
Default Credential Policies, 165–166
Enhanced Security Credential Policies, 165–166

https://t.me/learningnets
@cisco_in_persian_channel

user accounts
credential policy default updates, 170
credential policy defaults, 169–170
credential policy settings, 167–169
PoLRE (Power Over Long-Reach Ethernet), 26–27
Port Activity reports, 309
ports
PC ports, endpoint hardening, 276
security, 6, 65–67
tracking, CER, 38
Webex Calling ports, 435
POTS lines, call routing, 43–44
power plants, physical security, 24–25
practical UC security, defined, 2
preparation phase, physical security, 17–19
prevention phase, physical security, 19–20
PRI (Primary Rate Interfaces), 43–44
privacy controls, meetings, 462–463
private information, malicious discovery of, 3
PS-ALI (Private Switch Automatic Location Identification), 29
PSAP (Public Safety Answering Points), 28, 29, 47–49
PSIRT (Product Security Incident Response Teams), 86
PSTN (Public Switched Telephone Networks)
Collaboration Edge, IP-based PSTN access, 386–388
connections, centralized deployment models, 92, 94–95

R

RADIUS servers, MAB, 6
RAY BAUM's Act, 30
recording servers, CMS, 340
recovering accounts, 157–159
recovery controls, 21
RedSky, E911 solutions, 51
registering endpoints, 239
remote employees, 2
replay attacks, 100
reports
Call Handler Traffic reports, 310
Cisco Unity Connection, 308–311
Outcall Billing Detail reports, 310
Outcall Billing Summary reports, 310
Phone Interface Failed Logon reports, 309
Port Activity reports, 309
Transfer Call Billing reports, 310
Unused Voicemail Accounts reports, 310
User Lockout reports, 310
User Phone Login and MWI reports, 309
Users reports, 309
requests, SAML, 188
response phase, physical security, 21–22
Restricted software, VOS, 134–135
restriction tables, 325–326
default restriction tables, 326–327
default transfer restriction tables, 328–329
sign-in counts, 329–330
restrictions, export, 218–221
robocalls, 1, 3

https://t.me/learningnets
@cisco_in_persian_channel

role-based access
control groups, 170–171
Webex identity theft prevention, 437–438
root access, granting, 137–138
root CA
authentication into IOS trustpoints, 283–285
certificates, CUBE and TLS connectivity, 397–398
root certificates, 68
RSA keys, IPSec tunnels, 112
RTMT (Real-Time Monitoring Tools), 10–11

S

SAML (Security Assertion Markup Language)
ACS URL, 188
assertions, 188
authentication, 188
components, 187–188
CoT, 188
metadata, 188
OAuth, 252–255
requests, 188
SSO, Unified CM, 186
Cisco Unified CM Administration GUI, 189–191
configuring, 191–197
Webex identity theft prevention, 440–441
SBD (Security By Default), 239–245
scalable groups, SDA, 62
scanners, physical security, 20
SCCP (Signaling Connection Control Part), 287–288
conference bridges, 288–289
verifying, 289–290
scheduled/unscheduled meeting management, Webex, 457–459
SCIM (System for Cross-Domain Identity Management), Webex identity theft prevention, 443–446
screen timeouts, 201–202
SDA (Software-Defined Access), 60, 61–62
contracts, 62
DNA-C, 60, 62–64
ISE, 60–61
network infrastructures (wired/wireless), 61
scalable groups, 62
virtual networks, 62
security (UC), complexity of, 5–6
high security features, 6, 7
low security features, 6, 7
medium security features, 6, 7
minimizing, 7–10
segmentation
layer 2 segmentation, 58
layer 3 segmentation, 58–59
micro segmentation, 59–64
self-signed certificates, 225
SELinux, VOS, 129–134
server certificates, 68
session replays, 3
setting access, endpoint hardening, 275–276
SFTP, VOS, 134
SGACL (Security Group ACL), 62

https://t.me/learningnets
@cisco_in_persian_channel

shadow IT, 3–4, 5
defined, 4
prevalency, 4–5
range of deployments, 4
statistics, 4–5
targeted devices, 4
SHAKEN (Signature-based Handling of Asserted Information using toKENs), 1
shared lines, Meet-Me secure conferencing, 297
signaling, security, 217
asymmetric cryptography, 223
CA, 224–225
chains of trust, 227–229
self-signed certificates, 225
single-tier CA hierarchies, 225–226
three-tier CA hierarchies, 227–228
two-tier CA hierarchies, 226–227
ECC, 233–235
encryption licensing, 218–221
endpoint registration, 239
CTL files, 247–250
ITL, 239–245
mixed mode clusters, 245–247
SBD, 239–245
export restrictions, 218–221
FIPS, 222
man-in-the-middle attacks, 223–224
OAuth
authorization, 255–258
enabling, 258–261
SAML, 252–255
SIP, 250–252
phone profiles, security
applying to phones, 264–270
configuring, 261–264
PKI
overview, 222–229
Unified CM, 229–232
protocols, 110–111
CA certificates, trustpoints, 112–114
CSR generation, 112
domain configurations, 112
host names, 112
IPSec configuration, 117
IPSec transform sets, 116
IPSec tunnels, 0512–0761
ISAKMP policies, 115–116
SRTP, 218
TFTP file encryption, 237–238
untrusted digital certificate banners, 223–224
sign-in counts, restriction tables, 329–330
single-tier CA hierarchies, 225–226
SIP (Session Initiation Protocol), 44
Cisco Meeting Server, SIP media encryption, 293
OAuth, secure signaling/media encryption, 250–252
trunks
integration, 318–323
security profile configuration, 294–296
smart licensing, 298–302
SNMP (Simple Network Management Protocol)
CMS, 378–379
MIB, 204

https://t.me/learningnets
@cisco_in_persian_channel

packet captures, 205–206
SNMPv3
packet captures, 205–206
user settings, 207–208
system monitoring, 204–208
traps, 204
SNMPv2, CER, 40–41
SNMPv3, CER, 40–41
SOC 2 type II, cloud/hybrid cloud services, 431
SOC 3, cloud/hybrid cloud services, 431
social media, People Insights, Webex, 465–466
software
Restricted software, VOS, 134–135
Unrestricted software, VOS, 134–135
spoofing, 100
call spoofing, 3
caller-ID spoofing, 1, 26
SRND, design zone for Collaboration, 8
SRST (Survivable Remote Site Telephony), 93–94
SRTP, secure signaling/media encryption, 218
SSH (Secure Shell), VOS, 134
SSL certificates, LDAP directories, 178–180
SSO (Single Sign-On), Unified CM and SAML, 186
Cisco Unified CM Administration GUI, 189–191
configuring, 191–197
standard access control groups, 171–172
Standard End User - Password Credential Policies, Unified CM, 165–167, 169
STIR (Secure Telephony Identity Revisited), 1
streaming servers, CMS, 340
switches
LAN switches, CER, 40
port security, 6, 65–67
port tracking, CER, 38
spoofing, VLAN hopping, 77
symmetric-key authentication, 99
synchronization
end user accounts, 197–199
LDAP, 182–183
Syslog Agent, alarms, 210
system monitoring, SNMP, 204–208
system transfers, Cisco Unity Connection, 333–336

T

TCF (Total Certainty Factor), 73–74
TDoS (Telephony Denial of Service) attacks, 3, 391–392
technical (logical) controls, 19
telework
B2B communication, 403–406, 420–423
Collaboration Edge, 383
architecture, 384–385
B2B connectivity, 422–423
CMS functionality, 341–342
compliance, 423–424
CPL, 420–421
CUBE, 386–388
CUBE, deploying, 387–390
CUBE, dial-peers, 392–395
CUBE, NBAR, 388–390
CUBE, session control/ protection, 392–395

https://t.me/learningnets
@cisco_in_persian_channel

CUBE, TDoS protection, 391–392
CUBE, TLS connectivity, 395–401
CUBE, toll fraud prevention, 390–391
defending against attacks, 420–422
Expressway, 403–406, 420–423
IP-based PSTN access, 386–388
monitoring, 423–424
MRA, 406–420
VPN-based telework solutions, 402
VPN-less telework solutions, 402–403
Expressway, security features, 403–406, 420–423
MRA, 406–407
authentication, 413–415
authorization, 413–415
certificate requirements, 410–412
DNS, 407–409
firewall traversal, 412–413
ICE, 419–420
secure phone profiles, 416–417
token scopes/revocation, 418
troubleshooting, 418–419
remote employees, 2
security considerations, 383
VPN
VPN-based telework solutions, 402
VPN-less telework solutions, 402–403
TFTP, file encryption, 237–238
threats
call spoofing, 3
DoS attacks, 3
eavesdropping, 3
Game, The, 2–3
identity theft, 3
malicious discovery of private information, 3
media tampering, 3
robocalls, 1, 3
session replays, 3
shadow IT, 3–4, 5
defined, 4
prevalency, 4–5
range of deployments, 4
statistics, 4–5
targeted devices, 4
TDoS attacks, 3
toll fraud, 3
vishing, 3
three-tier CA hierarchies, 227–228
tiered security, 8–10
timeouts, screen, 201–202
TLS (Transport Layer Security)
CMS, 359–360
CUBE and TLS connectivity, 395–401
toll fraud, 3
Cisco Unity Connection, preventing in, 325
PIN logins, 325
restriction tables, 325–336
CUBE, preventing with, 390–391
Transfer Call Billing reports, 310
transform sets, IPSec, configuring, 116
traps, SNMP, 204

https://t.me/learningnets
@cisco_in_persian_channel

troubleshooting, MRA, 418–419
trust, chains of
CA certificates, 227–229
certificates, 286–287
CSR, 104–105
trustpoints
CA certificates, 112–114
CFB trustpoints, secure, 280
CUBE, TLS connectivity, 396–398
IOS trustpoints
intermediate CA authentication, 282–283
root CA, 283–285
IPSec tunnels, 112–114
TUI voicemail, alternate contact number restrictions, 330–333
TURN servers, CMS functionality, 342
turnstiles, physical security, 20
two-tier CA hierarchies, 226–227

U

UAC (User Account Controls)
local UAC, strengthening, 163–164
order of operations, 163
policies, creating, 164–170
UC (Unified Communication)
business models, changing, 2
certificate key usage requirements for CSR 12.0, 229–231
complexity of UC security, 5–6
high security features, 6, 7
low security features, 6, 7
medium security features, 6, 7
minimizing, 7–10
core applications, defining, 126–127
federal organizations, 1
physical security, 22
practical UC security, defined, 2
remote employees, 2
SHAKEN, 1
standards bodies, 1
STIR, 1
threats, 1, 2–3
UC Appliance, Linux server comparisons, 127–134
UCTM (Unified Communications Threat Management), 10–11
UI application passwords, 148
Unified CM, 273
application server settings, credential change service, 200–201
configuring, 374–376
endpoint hardening, 274, 275
configuring settings, 274–275
Gratuitous ARP, 275
PC ports, 276
PC Voice VLAN, 275
setting access, 275–276
web access, 275
end user accounts, synchronization, 197–199
mixed mode clusters, 245–247
mTLS, 236
OAuth, 258–261
PKI, 229–232
secure Cisco Unity Connection integration, 316–323
secure conferencing, 276–278
Ad Hoc conferencing, 278, 296–297
CMS, 290–297
Conference Now, 298

https://t.me/learningnets
@cisco_in_persian_channel

DSP, 278–290
Meet-Me secure conferencing, 297–298
smart licensing, 298–302
SSO, SAML, 186
Cisco Unified CM Administration GUI, 189–191
configuring, 191–197
Unified CM (Communications Manager), 126
access control groups
advanced role configuration, 173–174
role-based access control groups, 170–171
standard access control groups, 171–172
user rankings, 173
Default Credential Policies, 165–166
default user accounts, 162–163
Enhanced Security Credential Policies, 165–166
native E911 call routing, 31–34
Standard End User - Password Credential Policies, 165–167, 169
UAC policies, 164–170
unified security features, network security, 75–76
unlocated phone assignments, CER, 38
Unrestricted software, VOS, 134–135
unscheduled/scheduled meeting management, Webex, 457–459
untrusted digital certificate banners, 223–224
Unused Voicemail Accounts reports, 310
updating user account credential policy default updates, 170
upgrades
Cisco Unity Connection, 307
IPSec policies, 110
uploaders, CMS, 340
user access, Cisco Unity Connection, 311–316
user accounts
Cisco Emergency Responder, 163
Cisco Unity Connection, 162–163, 164–170
credential policies
default updates, 170
defaults, 169–170
settings, 167–169
end user accounts
importing end users from LDAP directories, 175–176
role assignments, 175
local UAC, strengthening, 163–164
Unified CM, 162–163
Default Credential Policies, 165–166
Enhanced Security Credential Policies, 165–166
Standard End User - Password Credential Policies, 165–167, 169
UAC policies, 164–170
User Lockout reports, 310
User Phone Login and MWI reports, 309
user rankings, Unified CM access control groups, 173
Users reports, 309

https://t.me/learningnets
@cisco_in_persian_channel

V

verifying
CER, 49–51
CMS certificates, 356–359
ERL, 43
IPSec policies, 109–110
mixed mode clusters, 247
NTP authentication, 100–101, 102
SCCP, 289–290
TLS connectivity, CUBE, 401
version/patch management, 136–137
video surveillance, IP-based, 20–21
virtual networks, SDA, 62
vishing, 3
visibility, CMS, 377–381
VLAN (Virtual LAN)
hopping, 77–78
layer 2 segmentation, 58
PC Voice VLAN, 275
VM (Virtual Machines), deployments, 91–92
voice gateways, securing, 110–111
CA certificates, trustpoints, 112–114
CSR generation, 112
domain configurations, 112
host names, 112
IPSec
configuration, 117
transform sets, 116
tunnels, 112–117
ISAKMP policies, 115–116
voice phishing (vishing), 3
voicemail, 311
message aging, 313
PIN logins, 325
restriction tables, 325–330
secure messages, 323–325
system transfers, 333–336
TUI voicemail, alternate contact number restrictions, 330–333
Unused Voicemail Accounts reports, 310
VoIP (Voice over Internet Protocols)
E911, 30–31
eavesdropping, 26–27
VOS (Voice Operating System), 138–139. *See also* OS
CC (Common Criteria, ISO/IEC 15408) compliance, 144–147
CMS OS comparisons, 343
Enhanced Security Mode, 143–144, 147
FIPS 140–2, 139–142, 147
HTTPS, 134
Linux server comparisons, 127–134
password complexity, 152–155
Restricted software, 134–135
root access, granting, 137–138
SELinux, 129–134
SFTP, 134
SSH, 134
Unrestricted software, 134–135
VoWLAN (Voice over WLAN), 27
VPN (Virtual Private Networks)
VPN-based telework solutions, 402
VPN-less telework solutions, 402–403
VQ Conference Manager, 380
VRF (Virtual Route Forwarding), layer 3 segmentation, 58–59
Vyopta, 380

https://t.me/learningnets
@cisco_in_persian_channel

W

WAN, centralized deployment models, 92, 94–95
web access, endpoint hardening, 275
web bridges, CMS, 340, 367
Webadmin PKI, 292–293
Webex
DLP, 451–454
EMM, 455
facial recognition, 464
firewalls, 432–433
identity theft prevention, 437
Directory Connector, 438–439
OAuth 2.0, 441–443
onboarding, 437–438, 443–446
role-based access, 437–438
SAML 2.0, 440–441
SCIM, 443–446
IoT security, 467–470
MAM, 455–456
MDM, 455
meeting management/security, 456–457
data at rest, 463
end-to-end encryption, 461–462
meeting authentication, 460
in-meeting privacy controls, 462–463
scheduled/unscheduled meetings, 457–459
Meeting Transcription, 467
messaging service security, 446–447
content management, 448–451
end-to-end message encryption, 447–448
external communications, 448–451
PAC files, 432
People Insights, 465–466
security across emerging features, 463–464
transport security/compliance, 432–433
Webex Assistant, 466–467
Webex Calling, 433–437
WPAD, 432
WFH (Work From Home)
B2B communication, 403–406, 420–423
Collaboration Edge, 383
architecture, 384–385
B2B connectivity, 422–423
CMS functionality, 341–342
compliance, 423–424
CPL, 420–421
CUBE, 386–388
CUBE, deploying, 387–390
CUBE, dial-peers, 392–395
CUBE, NBAR, 388–390
CUBE, session control/protection, 392–395
CUBE, TDoS protection, 391–392
CUBE, TLS connectivity, 395–401
CUBE, toll fraud prevention, 390–391
defending against attacks, 420–422
Expressway, 403–406, 420–423
IP-based PSTN access, 386–388
monitoring, 423–424
MRA, 406–420
VPN-based telework solutions, 402

https://t.me/learningnets
@cisco_in_persian_channel

Edge (Collaboration), VPN-less telework solutions, 402–403
Expressway, security features, 403–406, 420–423
MRA, 406–407
authentication, 413–415
authorization, 413–415
certificate requirements, 410–412
DNS, 407–409
firewall traversal, 412–413
ICE, 419–420
secure phone profiles, 416–417
token scopes/revocation, 418
troubleshooting, 418–419
remote employees, 2
security considerations, 383
VPN
VPN-based telework solutions, 402
VPN-less telework solutions, 402–403
wildcard characters, Cisco Unity Connection, 327–328
wireless devices, location tracking, 42–43
WireShark, VoIP eavesdropping, 26–27
work hour access restrictions, CMS, 345
WPAD (Web Proxy Auto Discovery), 432

X - Y - Z

x.509 certificates, 68–70
XMPP servers, CMS, 340, 341

https://t.me/learningnets
@cisco_in_persian_channel

Exclusive Offer – 40% OFF

Cisco Press Video Training

livelessons

ciscopress.com/video

Use coupon code **CPVIDEO40** during checkout.

Video Instruction from Technology Experts

Advance Your Skills

Get started with fundamentals, become an expert, or get certified.

Train Anywhere

Train anywhere, at your own pace, on any device.

Learn

Learn from trusted author trainers published by Cisco Press.

Try Our Popular Video Training for FREE!

ciscopress.com/video

Explore hundreds of **FREE** video lessons from our growing library of Complete Video Courses, LiveLessons, networking talks, and workshopscip.

Cisco Press ciscopress.com/video

ALWAYS LEARNING PEARSON

@cisco_in_persian_channel

Photo courtesy of Cisco

Register Your Product at ciscopress.com/register

Access additional benefits and **save 35%** on your next purchase

- Automatically receive a coupon for 35% off your next purchase, valid for 30 days. Look for your code in your Cisco Press cart or the Manage Codes section of your account page.
- Download available product updates.
- Access bonus material if available.*
- Check the box to hear from us and receive exclusive offers on new editions and related products.

**Registration benefits vary by product. Benefits will be listed on your account page under Registered Products.*

Learning Solutions for Self-Paced Study, Enterprise, and the Classroom

Cisco Press is the Cisco Systems authorized book publisher of Cisco networking technology, Cisco certification self-study, and Cisco Networking Academy Program materials.

At ciscopress.com, you can:

- Shop our books, eBooks, practice tests, software, and video courses
- Sign up to receive special offers
- Access thousands of free chapters and video lessons

Visit ciscopress.com/community to connect with Cisco Press

Pearson

Addison-Wesley • Adobe Press • Cisco Press • Microsoft Press • Pearson IT Certification • Que • Sams • Peachpit Press

https://t.me/learningnets

@cisco_in_persian_channel