EC-Council

Building A Culture Of Security

C|ND

Certified Network Defender

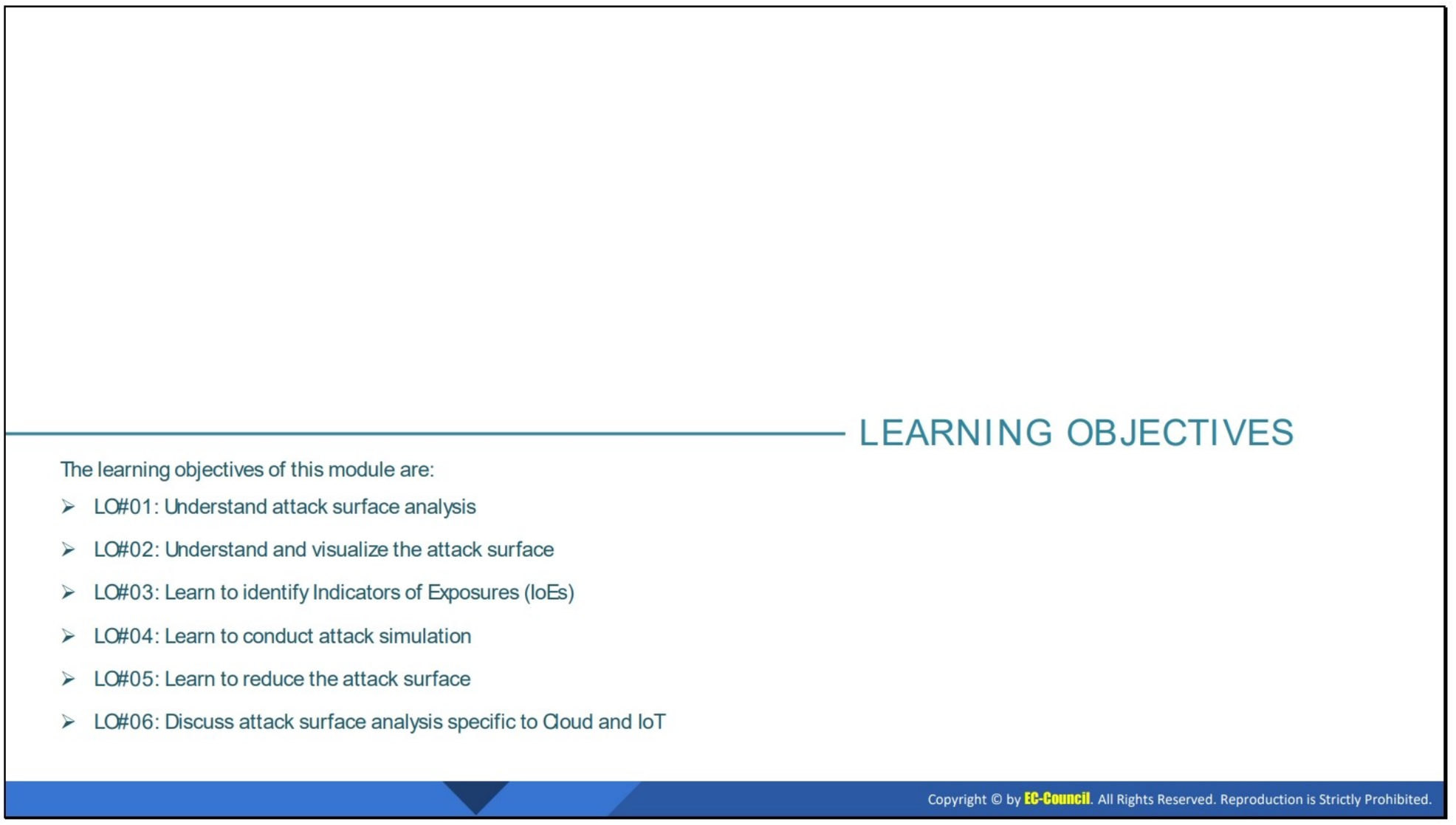
Certified Network Defender v3

MODULE 19

THREAT ASSESSMENT WITH ATTACK SURFACE ANALYSIS

EC-Council Official Curricula

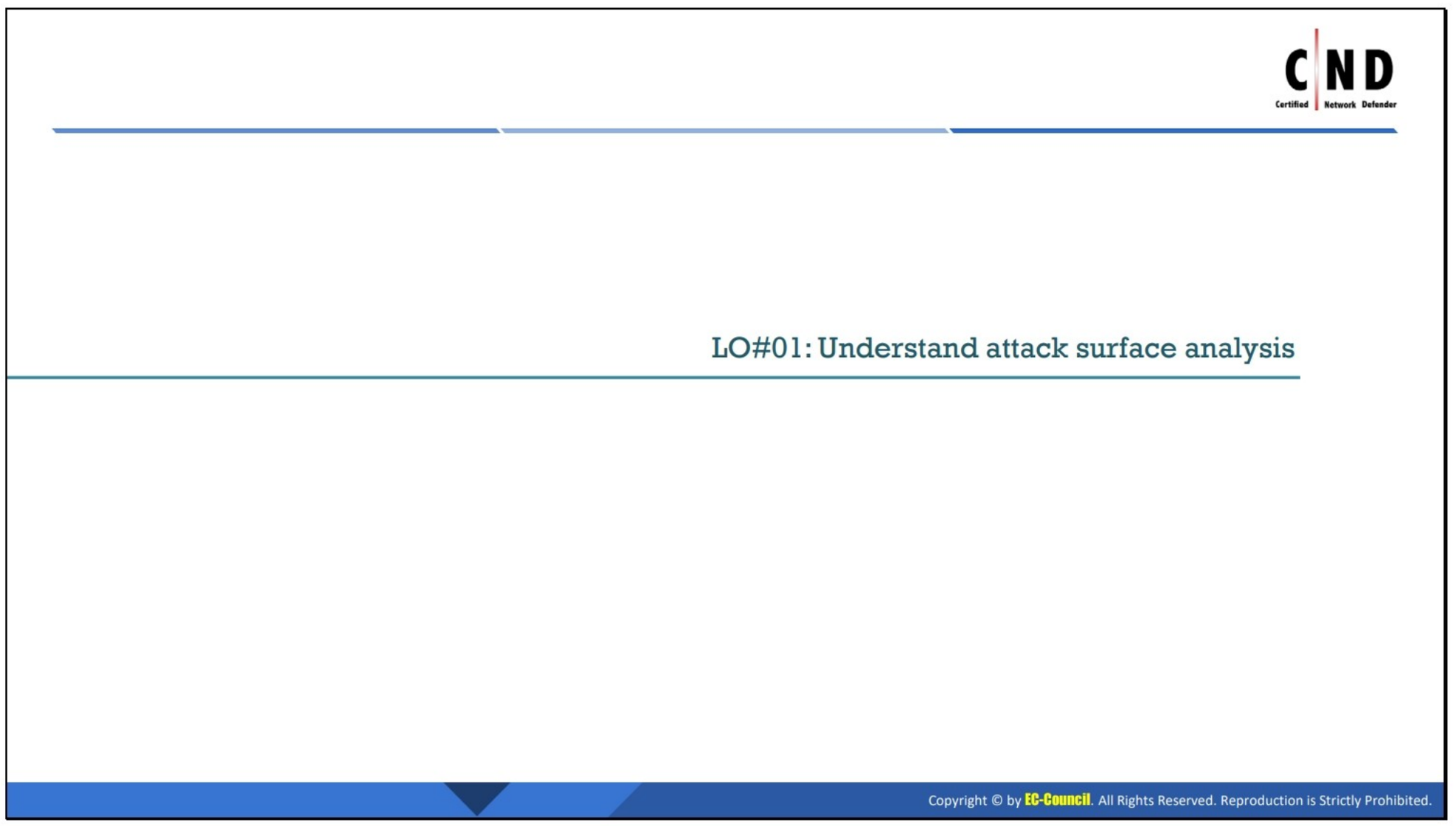
This page is intentionally left blank.



Learning Objectives

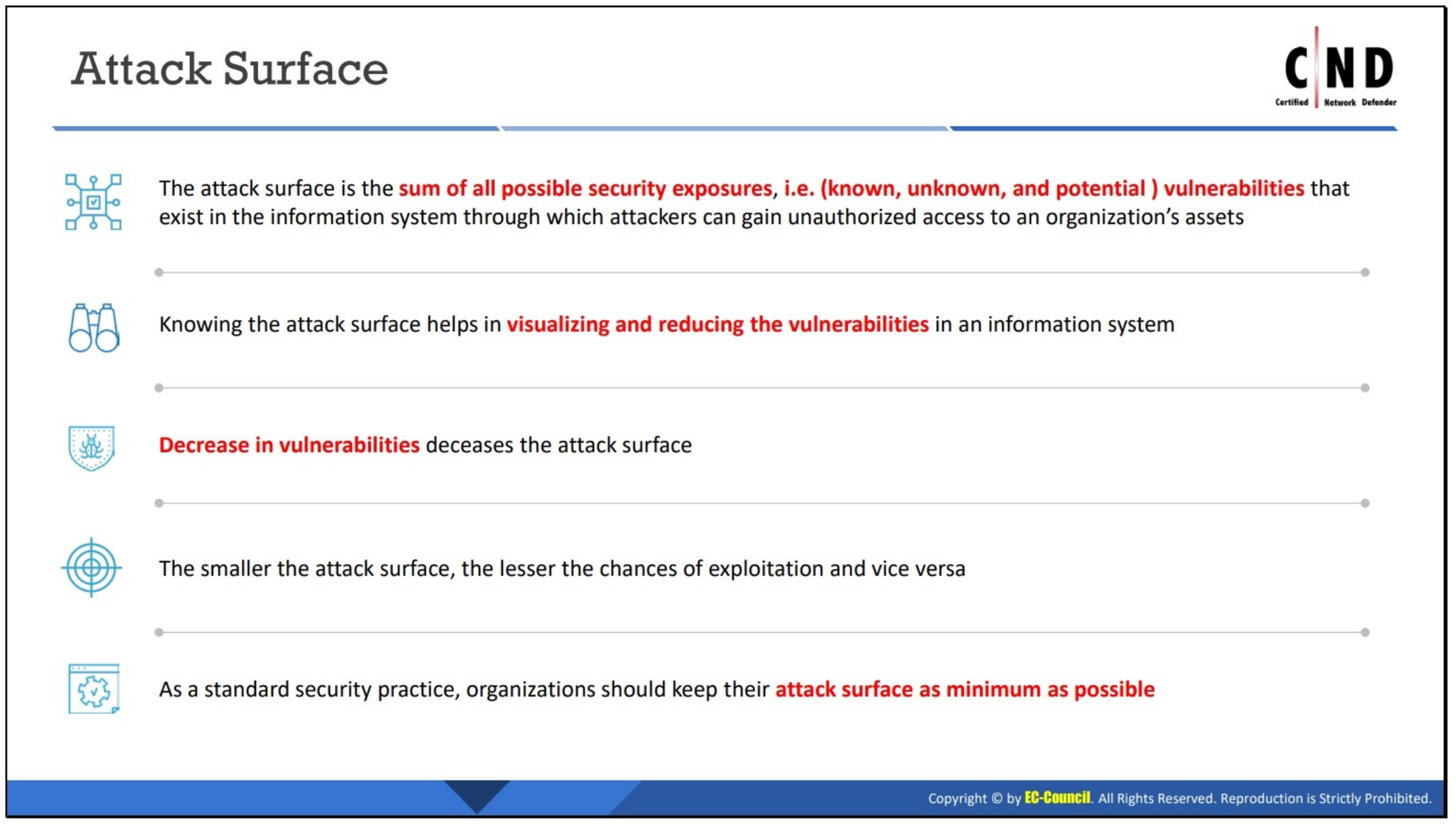
To implement appropriate security controls, network defenders need to have an understanding of an organization's attack surface. This will help them take appropriate measures to reduce the attack surface and prevent attackers from gaining access to restricted data/resources. This module will help you analyze and visualize the attack surface. It will enable you to understand various security measures to reduce the existing attack surface. The learning objectives of this module are as follows:

- Attack surface analysis
- Attack surface
- Identify Indicators of Exposures (IoEs)
- Conduct attack simulation
- Reduce the attack surface
- Attack surface analysis specific to Cloud and IoT



LO#01: Understand Attack Surface Analysis

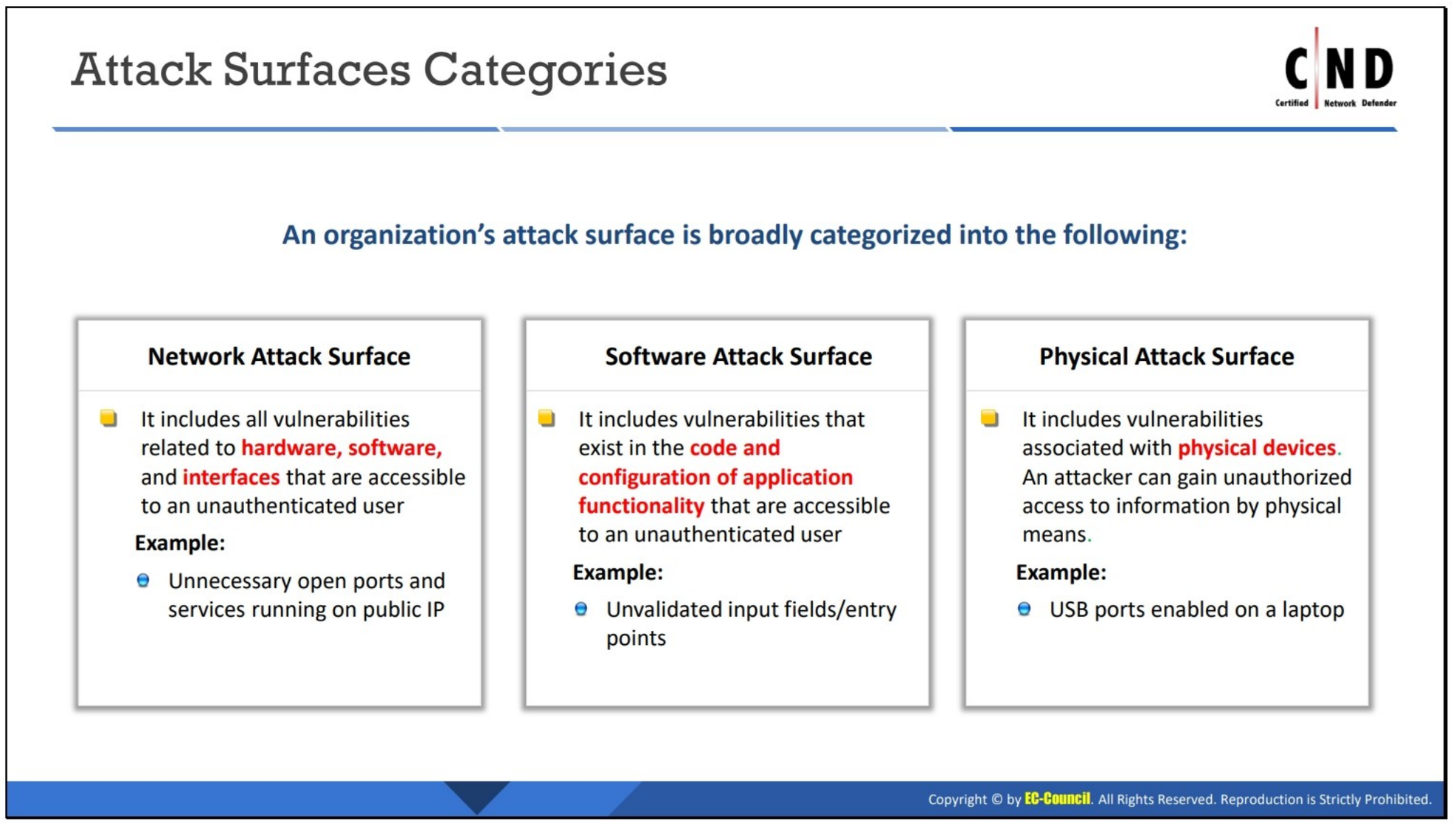
Understanding the attack surface is important for deploying the right security tools to defend the network. Conducting an attack surface analysis helps protect the network by implementing appropriate methods to protect the attack surface. This section will help you understand attack surfaces such as the network attack surface, software attack surface, physical attack surface, human attack surface, and system attack surface. You will learn the various steps involved in performing an attack surface analysis.

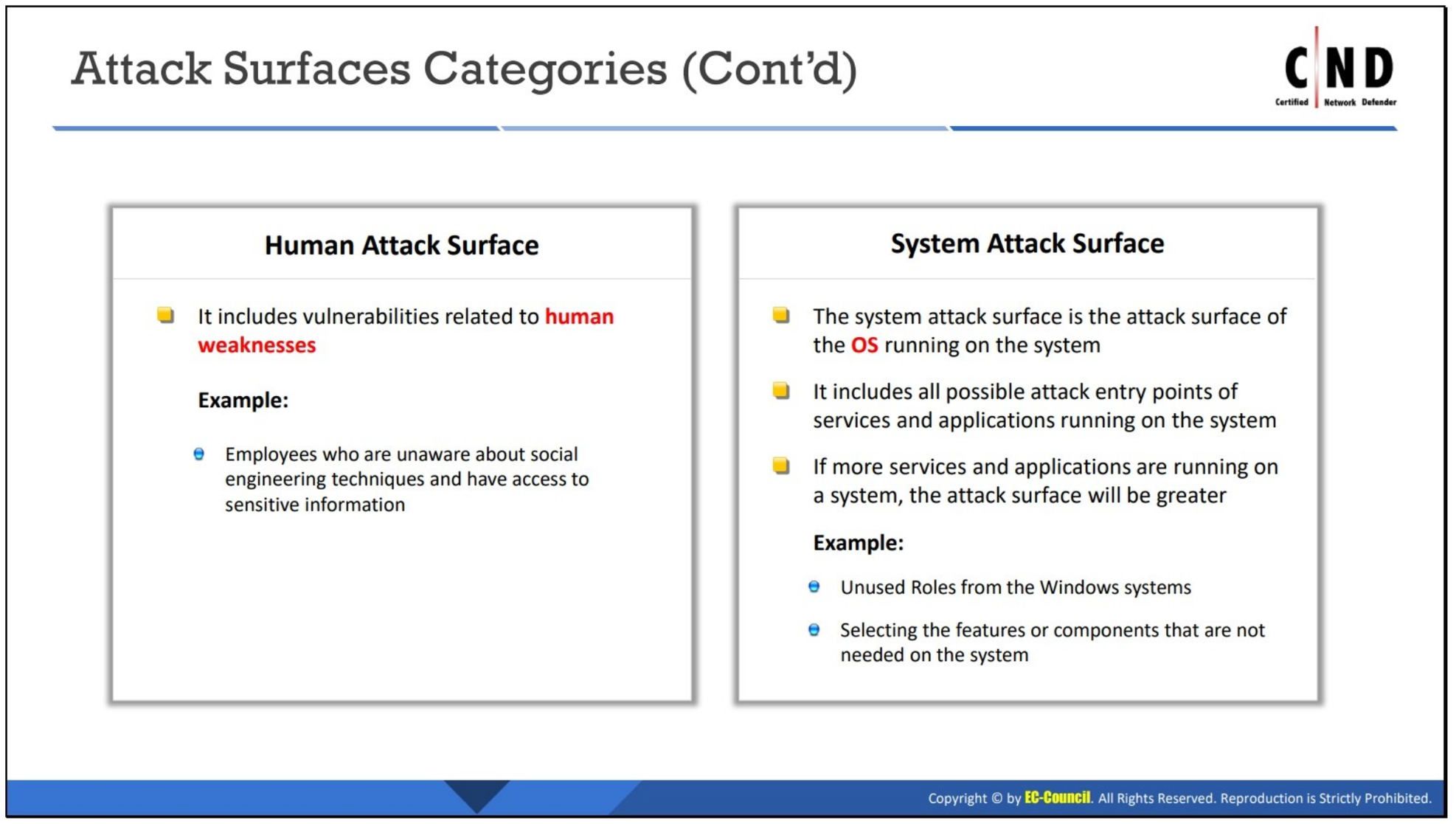


Attack Surface

The attack surface is the sum of all possible exposures (known, unknown, and potential) that exist in the information system through which an unauthorized user or attacker can access the organization's assets. Some examples of possible exposures include protocols, interfaces, user input fields, and services. An organization's attack surface includes unpatched vulnerabilities, open ports, misconfigured networks, excess privileges granted to users, improper network segmentation, and employees who are unaware of security controls.

A lesser number of vulnerabilities in an information system results in a smaller attack surface. A smaller attack surface makes the organization less exploitable and reduces risk, whereas a greater attack surface makes the organization more vulnerable to attacks, which increases the risk. Therefore, as a standard security practice, organizations should keep their attack surface as minimum as possible.





Attack Surfaces Categories

Attack surfaces are categorized into five types, and they pose different types of threats.

Network Attack Surface

The network attack surface comprises all possible vulnerabilities related to hardware (ports protocols or devices), software (services or network applications), and firmware interfaces that can be accessed by an unauthorized person.

Some network attack surface examples are as follows:

- Unnecessary open ports and services running on public IP
- Network protocols (Telnet, File Transfer Protocol (FTP), Hypertext Transfer Protocol (HTTP), and Simple Mail Transfer Protocol (SMTP)) that pass unencrypted data over the network
- Network file systems (Network File System (NFS) and Server Message Block (SMB)) that pass unencrypted information.
- Remote memory dump services (netdump) that pass the unencrypted contents of memory
- Network printers

Software Attack Surface

The software attack surface includes the vulnerabilities that exist in the code, configuration, or complete profile of functions in an application that are accessible to an authorized user.

It comprises vulnerabilities related to the following:

- Different kinds of code (applications, OSes, mobile apps, Dynamic-Link Libraries (DLLs), databases, web pages, executables, email services, or configurations)
- Unpatched software (Java, Adobe Reader, or Adobe Flash).

Some software attack surface examples are unvalidated input fields/entry points.

Physical Attack Surface

The physical attack surface includes all security vulnerabilities that exist in the hardware system. An attacker gains unauthorized access to information by physical means. A physical attack surface (physical access) is a direct attack surface. An attacker who gets access to the physical device can explore the entire network and perform the following malicious activities:

- Scan network, ports, and service to create a network map.
- Create a digital map of all network, ports, and services.
- Access the running databases and the stored information.
- Upload malware to infect the OS.
- Crack credentials to access privileged areas.
- Copy sensitive data to removable devices or send them to remote servers.

The physical attack surface can be exploited through the following two types of threats:

- **Insider threats** such as employees with malicious intentions who steal and leak sensitive information, negligent employees who are easy targets for phishing attacks or social engineering attacks.
- **External threats** such as extracting sensitive information from discarded hard disks.

Physical attack surface examples are access to all endpoint devices (desktops, laptops, mobiles, USB ports, and improperly discarded hard drives).

Human Attack Surface

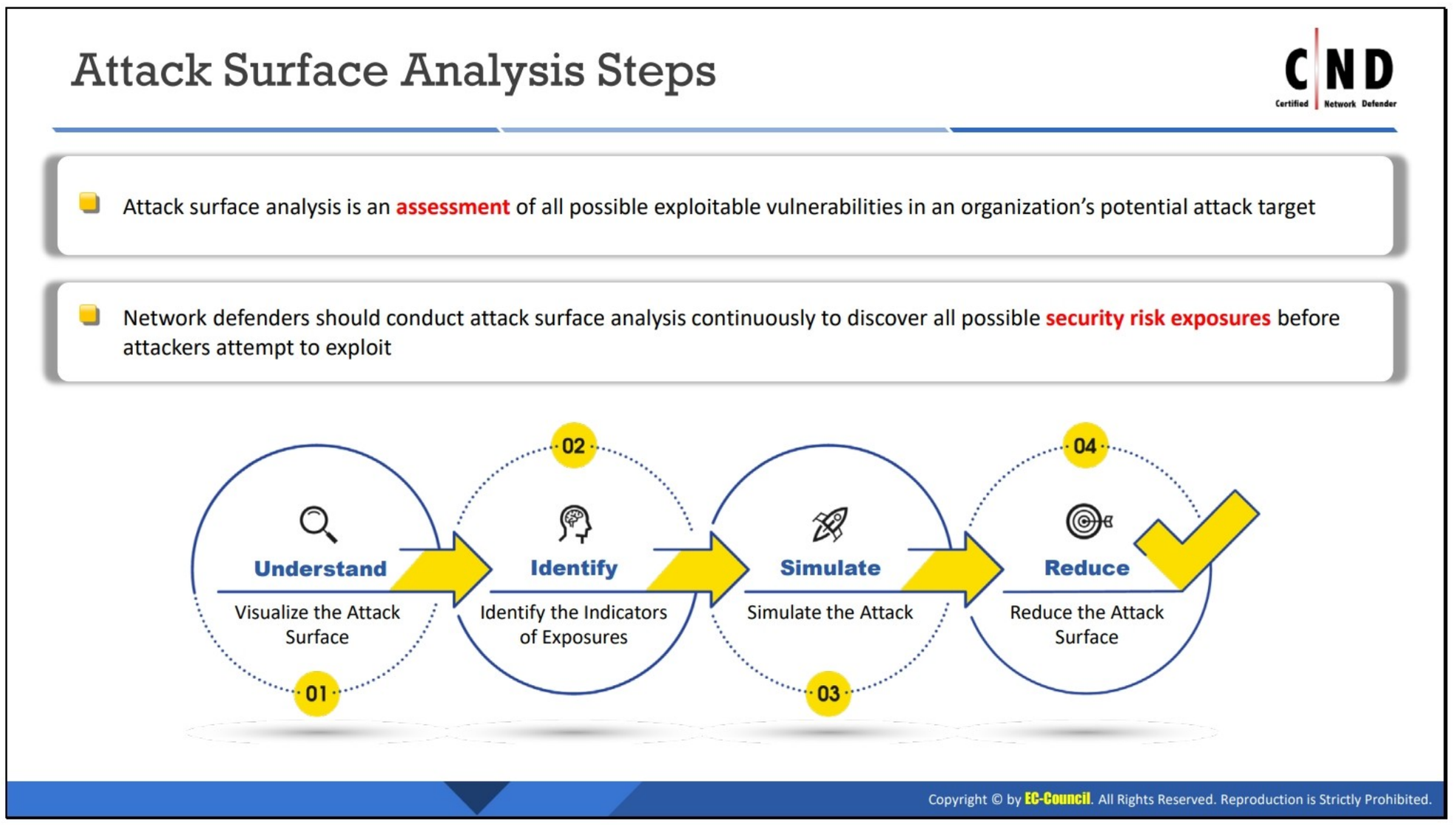
The human attack surface/social engineering attack surface includes vulnerabilities related to human weaknesses, and it is the weakest point in any cybersecurity. Employees/users who wish to gain something for free, those who are ready to help anyone for goodwill, and those who can be easily manipulated are the main targets of social engineering attacks.

Examples of human attack surface include fake calls resulting in giving up passwords, losing removeable devices containing sensitive information, or accessing legitimate websites with trojans or viruses.

System Attack Surface

The system attack surface represents the attack surface of the OS running on the system**.** It includes all possible attack entry points of services and applications running on the system. The more the number of services and applications running on the system, the higher its attack surface.

Some system attack surface examples include unused roles from the Windows systems and selecting all features or components that are not needed on the system.



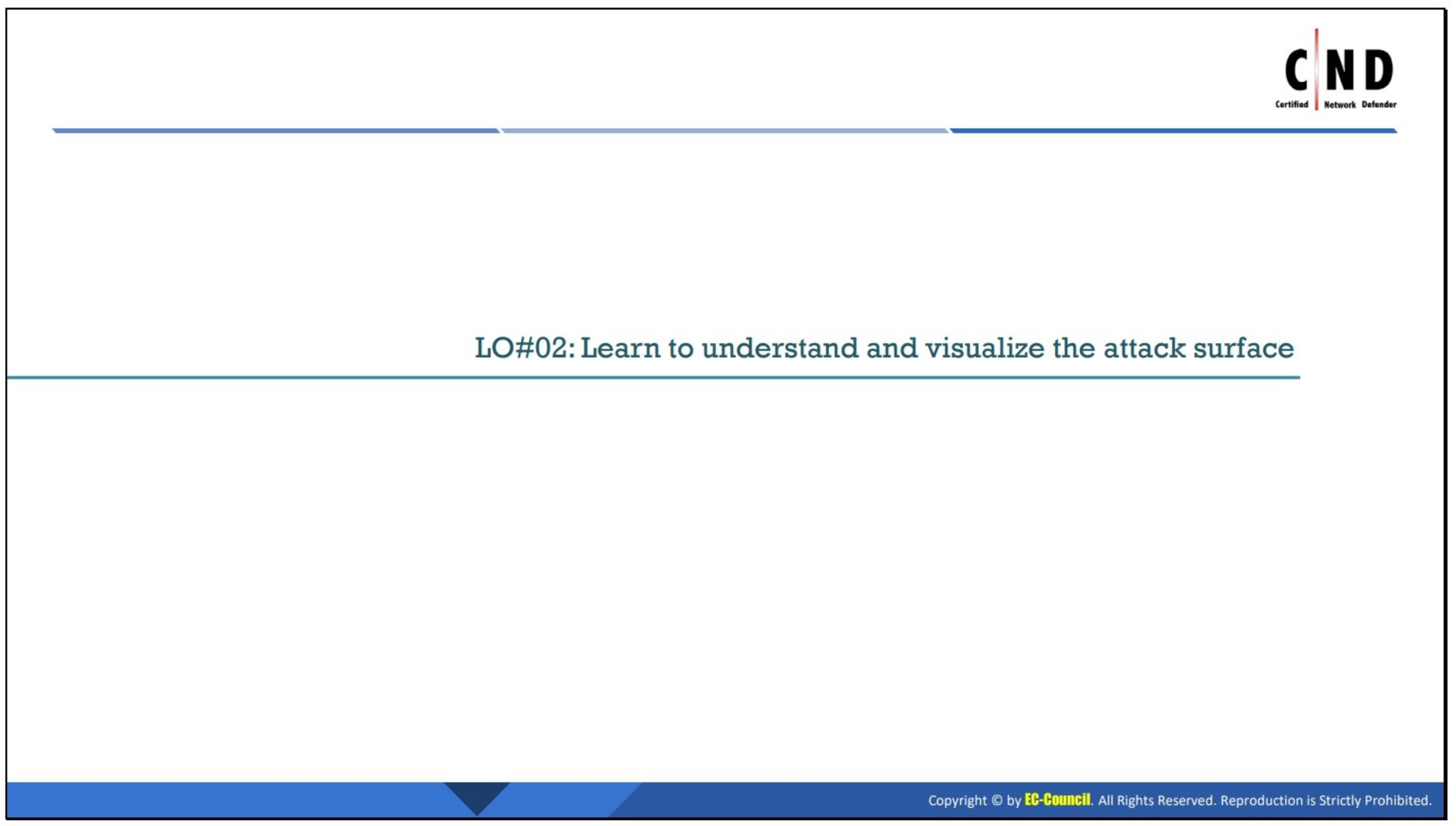
Attack Surface Analysis Steps

The attack surface analysis is an assessment of all possible exploitable vulnerabilities in an organization's potential attack target. The attack surface analysis helps in the following:

- Identifying the functions and the parts of the system that must be reviewed/tested for security vulnerabilities
- Identifying high-risk areas of code that need defense-in-depth protection
- Identifying when the attack surface has changed and what kind of threat assessment is needed
- Filtering the issues speedily without causing a financial disaster in the organization
- Continuous discovery of all possible security risk exposures before being exploited by attackers

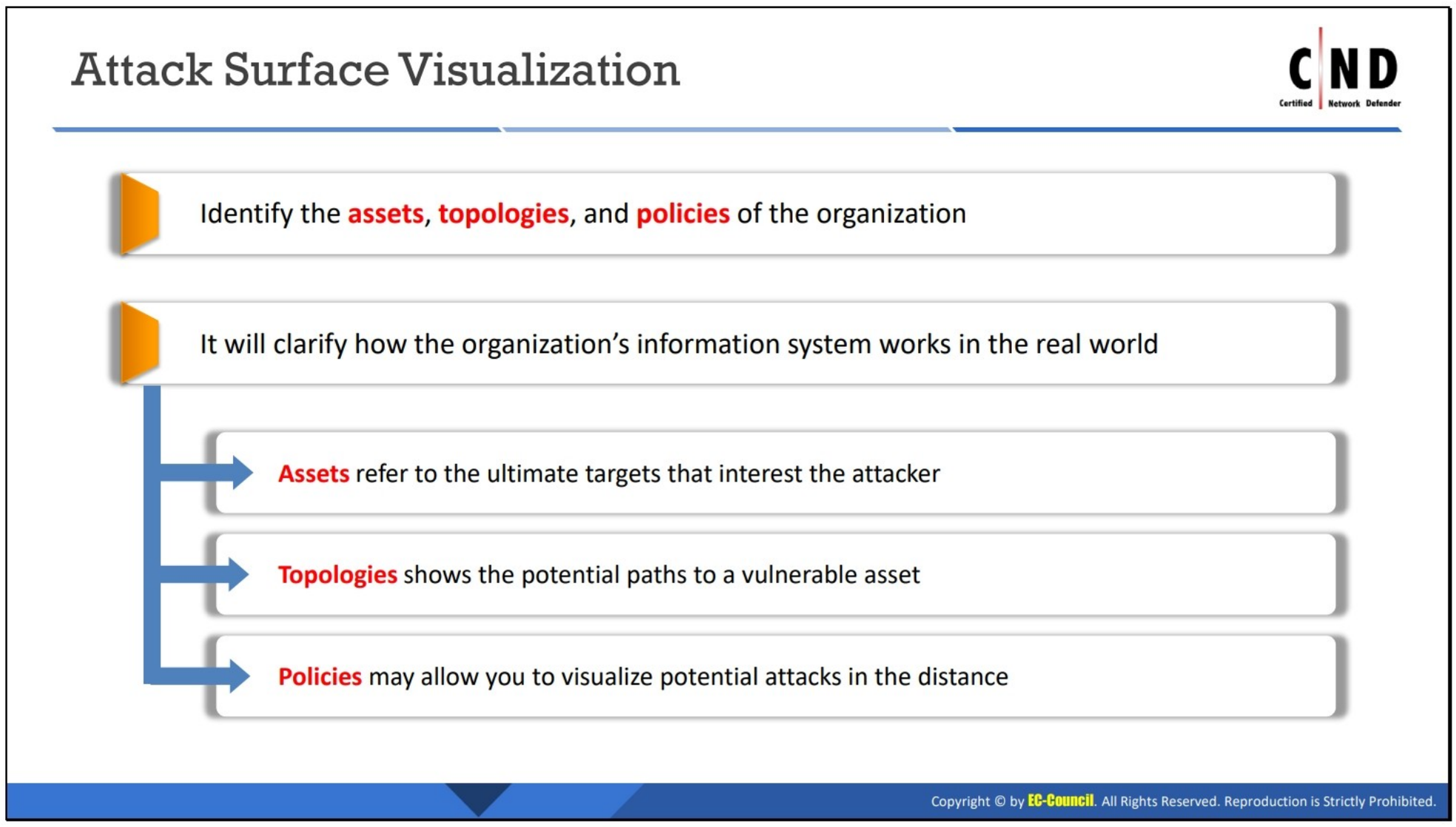
Steps to analyze the attack surface include the following.

- **Understand and Visualize the Attack Surface**: This step involves mapping out all devices, paths, and networks.
- **Identify the Indicators of Exposures (IoEs)**: In this step, potential risk exposures that attackers can use to breach the security of an organization are identified.
- **Simulate the Attack:** This step involves recognizing how the identified IoE could turn to exploit.
- **Reduce the Attack Surface:** This step involves implementing appropriate security controls and countermeasures to reduce the attack surface of a system.



LO#02: Attack Surface

Visualizing the attack surface helps network defenders identify possible risk exposures beforehand that pose a potential threat to an organization's security and reduce the attack surface. This section will help you understand the concept of attack surface visualization and the tools used for attack surface visualization.



Attack Surface Visualization

Attack surface visualization refers to monitoring the attack surface constantly. This strategy improves information security by minimizing untrusted user access and unnecessary functionalities.

Understanding and visualizing the attack surface can help a network defender reduce the vulnerabilities in an information system. Improper understanding of the attack surface leads to the following:

- Data breaches
- Unidentified threats
- Difficulty in forecasting security investments
- Increased audit cost
- Poor reaction to the violation of policies and rules

It is recommended that organizations possess a clear understanding of their attack surface to meet the best outcomes in the attack surface visualization. To ensure this and to take preventive measures, organizations must be aware of all possible ways in which a hacker might try to exploit the attack surface.

Visualization helps to do the following:

- **Identify the assets** that show the stock of everything the network possesses and the things for which that attacker may interested in.
- **Identify the topologies** of the organization. Visualizing involves mapping all systems, devices, and network segments in the organization and the paths between them where

data can flow. These topologies show the potential ways in which vulnerable assets can be accessed. The following elements should be mapped:

- **Servers** comprise web servers, application servers, and database servers, among others, as well as a massive storage of the company.
- **Endpoints** are the individual electronic devices used by the employees in the organization. These include laptops, desktops, and mobile devices.
- **Networks** comprise network segments and private and public Clouds. Organizations transact through this medium.
- **Networking devices** comprise routers, switches, and load balancers. The organization's networking occurs through these mediums.
- **Security devices** comprise firewalls, Intrusion Prevention Systems (IPSs), and Virtual Private Network (VPN) concentrators, among others. These can prevent intrusions into the company's server.

- **Create Policies** to implement better security controls and reduce the attack surface.

Challenges in Understanding an Attack Surface

Organizations believe that they can keep themselves secure always by following different approaches – deploying vulnerability scanners to identify weaknesses, patching all systems and applications, and installing strong security controls. However, several challenges continue to exist in understanding the organization's attack surface.

- **Vast security data**

  To deal with vulnerabilities, organizations always embed policies rules in firewalls and use IPSs and other security controls. New vulnerabilities are always introduced when organizations make many changes to ensure their security. For example, adding/removing servers and devices, reconfiguring networks, modifying applications, and deploying new technologies. All these actions generate vast amounts of security data that is too difficult to analyze.

- **Security silos**

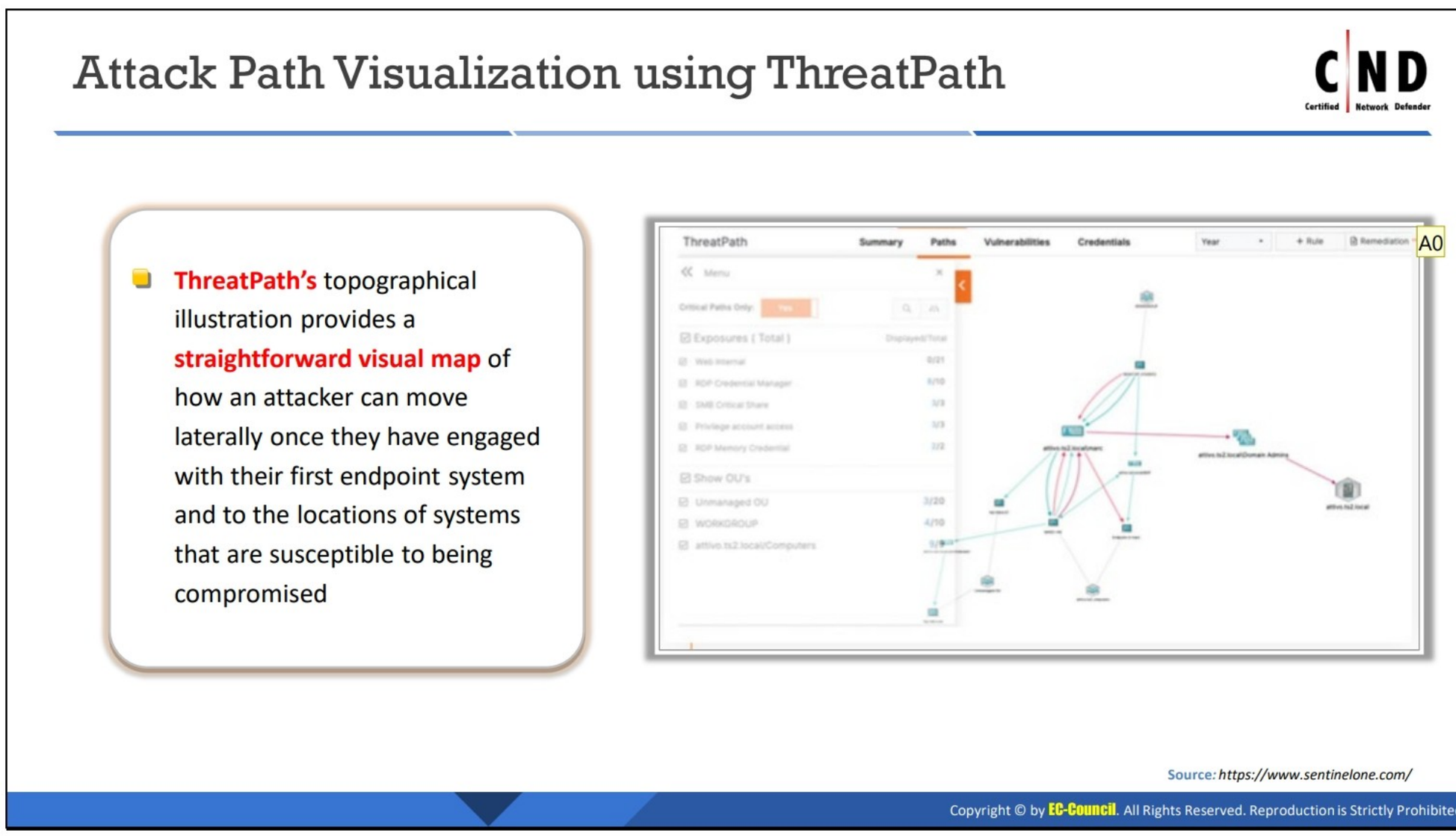
  IT organizations work in silos (security, network, applications, and system operations teams), and each silo section is responsible for different sections of the IT infrastructure and different classes of vulnerabilities in an organization. Each silo section uses different solutions based on its requirement and produces different pools of security data that offer little visibility into areas that overlap or are outside the scope of their responsibility. Similarly, with regard to teams from different geographical regions and different business units, they do not have clear visibility about how resources and information flow.

- **Network topology and configuration complexity**

  The combination of vulnerabilities and misconfiguration of network security devices (firewall) often poses the most serious exposure. These can expose the organizations to attacks that cannot be detected by traditional security software.

- **Lack of a planned approach for attack surface mitigation**

  Organizations that do not have tools to gather and correlate huge vulnerability data, the information of the policy rules and visualizing their attack surface cannot identify the risks, set priorities for remediation, and track progress to enhance their overall security posture.



Attack Path Visualization using ThreatPath

Source: https://attivonetworks.com/

ThreatPath's topographical illustration provides a straightforward visual map of how an attacker can move laterally once they have engaged with their first endpoint system and to the locations of systems that are susceptible to being compromised.

With Attack Path, a network defender can perform the following:

- Visualize the exposed paths an attacker sees.
- Detect the misused and orphaned credentials and misconfigured systems.
- Understand the attack paths to improve the security risk posture.
- Defend with the automated workflows for remediation.

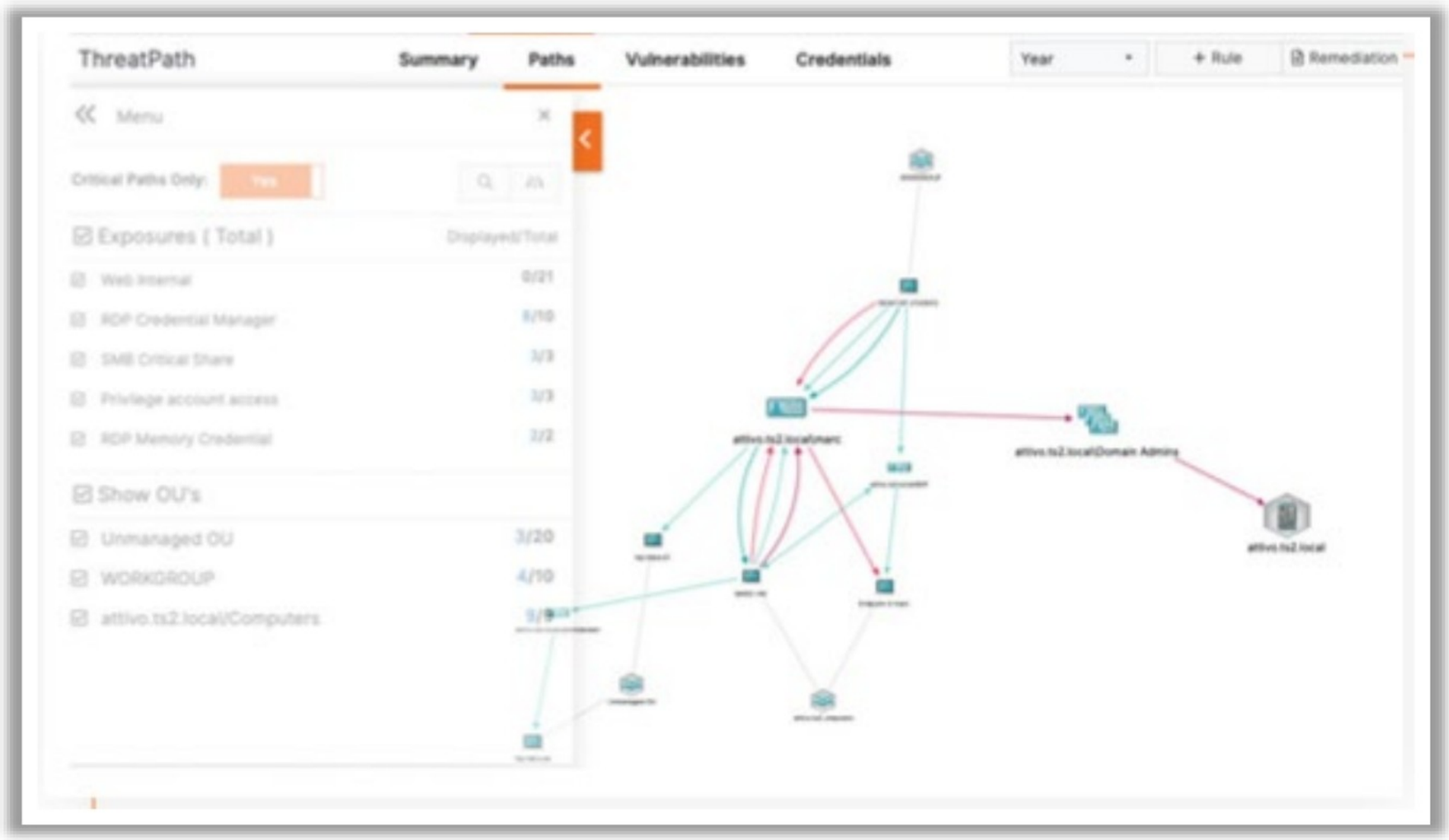
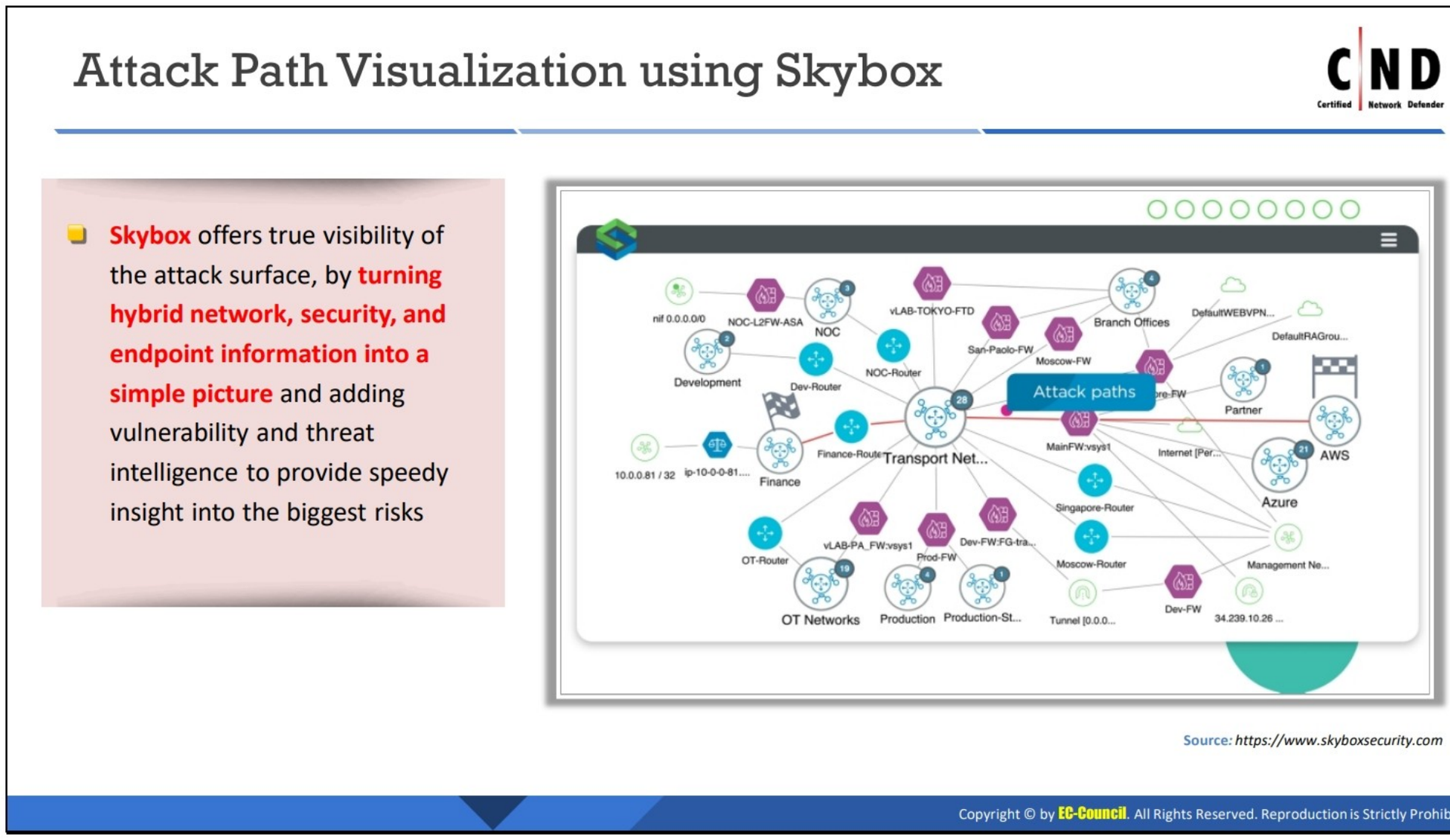


Figure 19.1: ThreatPath

Benefits of ThreatPath

- Early detection of exposed credential and policy vulnerabilities
- Topographical maps (Visual graphs) to show exposure
- Analysis of the lateral attack paths and movement
- Actionable workflow integrations for fast remediation
- Remediating exposures before they are misused
- Continuous ongoing monitoring of policy adherence



Attack Path Visualization using Skybox

Source: https://www.skyboxsecurity.com

Skybox offers true visibility of the attack surface, by turning hybrid network, security, and endpoint information into a simple picture and adding vulnerability and threat intelligence to provide the capability to respond quickly to major threats and reduce the attack surface.

Key Features

- Visualize and Analyze Indicators of Exposures **(IoEs)** uncovers and filters IoEs based on severity or timeframe. It also allows to export the views of the attack surface and IoEs to enable the organization's security team to discuss the impact of security strategies.
- Attack Surface Modelling and Simulation allows to see the interaction among security controls, network topology, vulnerabilities, and threats. It brings physical, virtual and Cloud networks into one view with its Hybrid Environments feature.

  The analysis that it performs includes the following:

  - Multi-step attack simulations
  - Predictive analysis of proposed network changes
  - Network path analysis and more.
- Risk-Reduction History and Trends provides the following:
  - Insight to the impact security efforts have timely.
  - Shows the progress in achieving strategic security goals/compliance requirements.
  - Contains interactive dashboards that perform the following:

- Track, measure, and report on risk reduction progress.
- Translate diverse and complicated metrics into an easily comprehensible format.

- Focuses on specific sites within the attack surface and compares current and past IoE levels to know whether the security posture is increasing or decreasing.
- Views graphs to know the progress in reducing different IoEs.

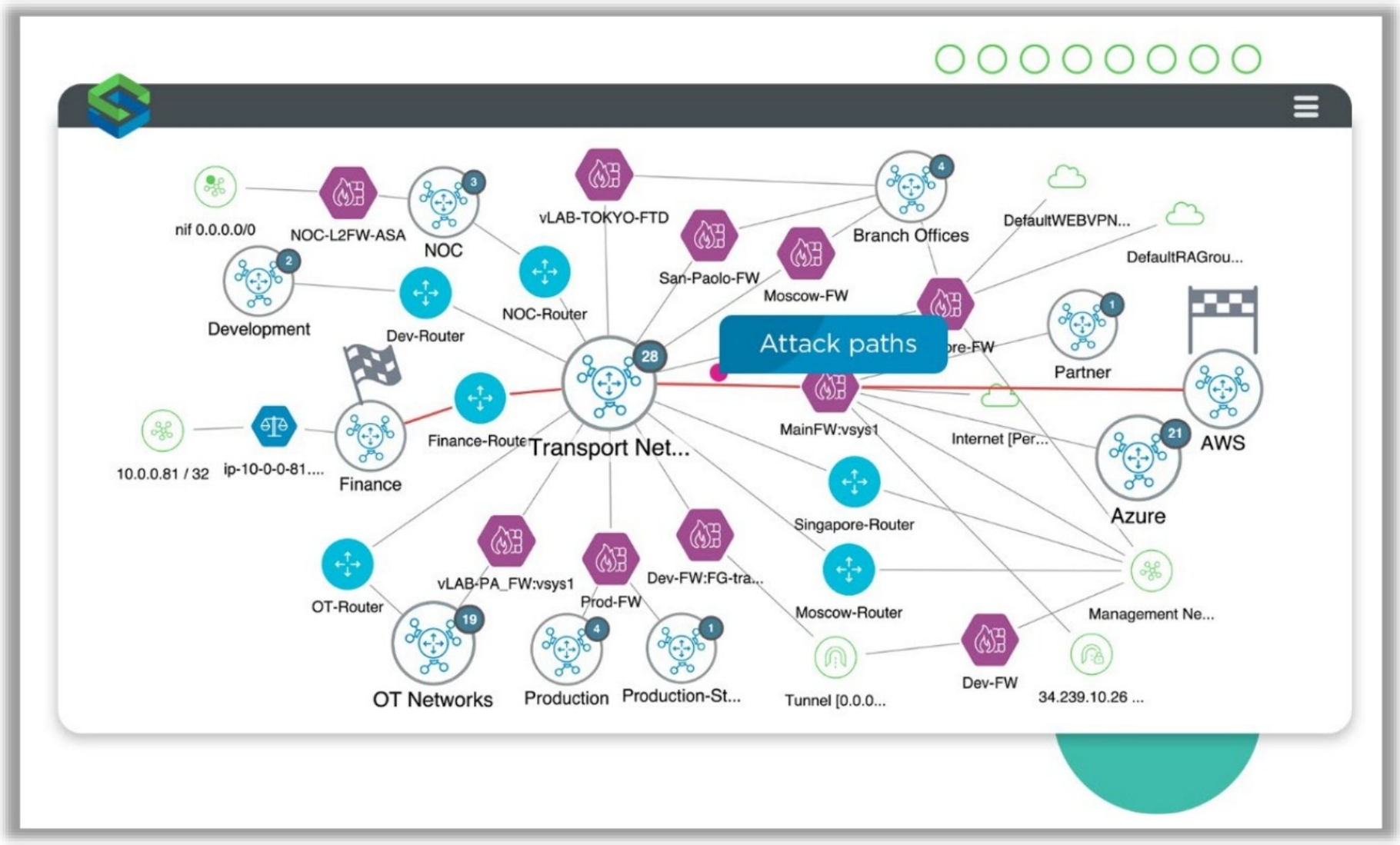
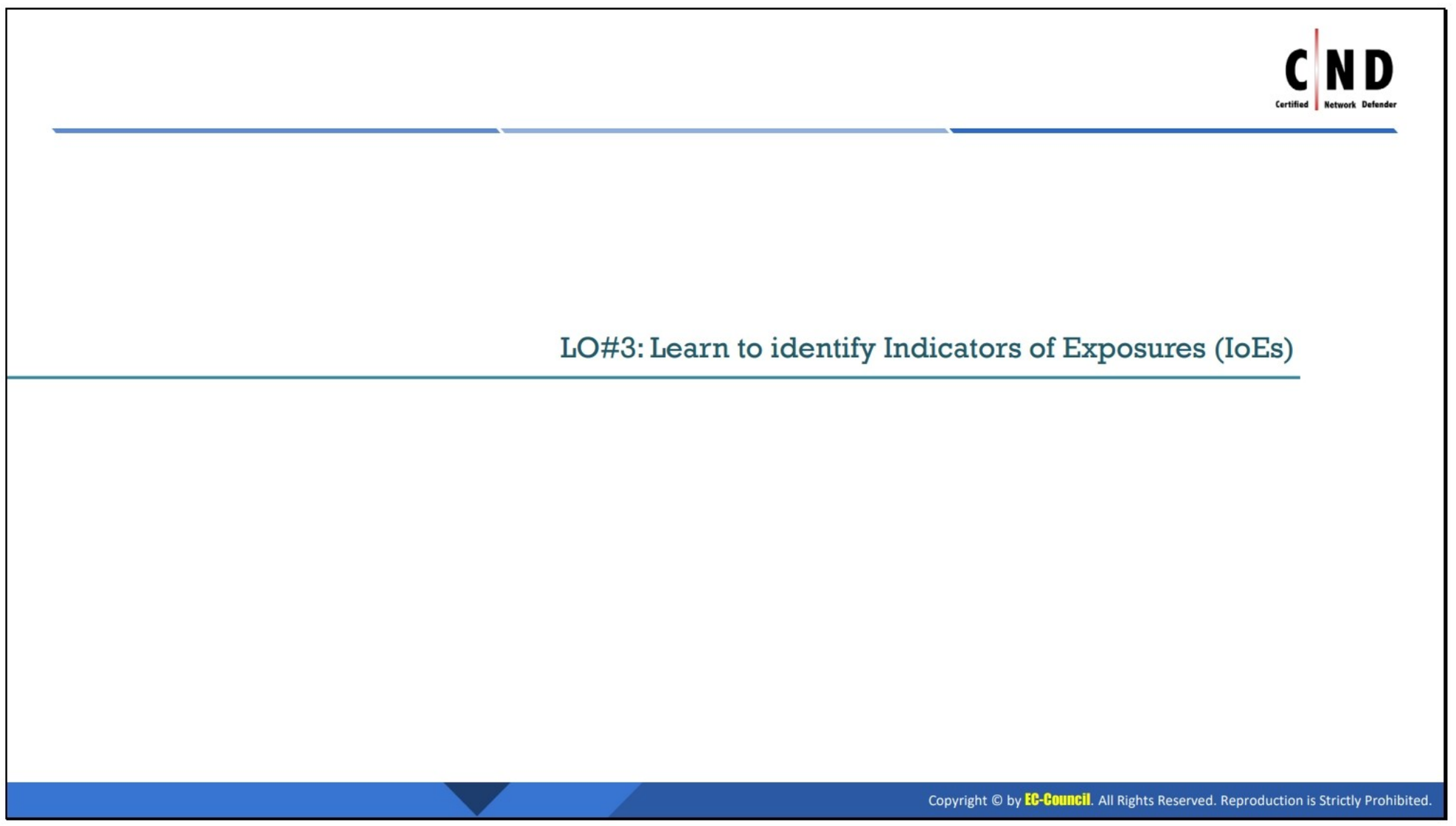
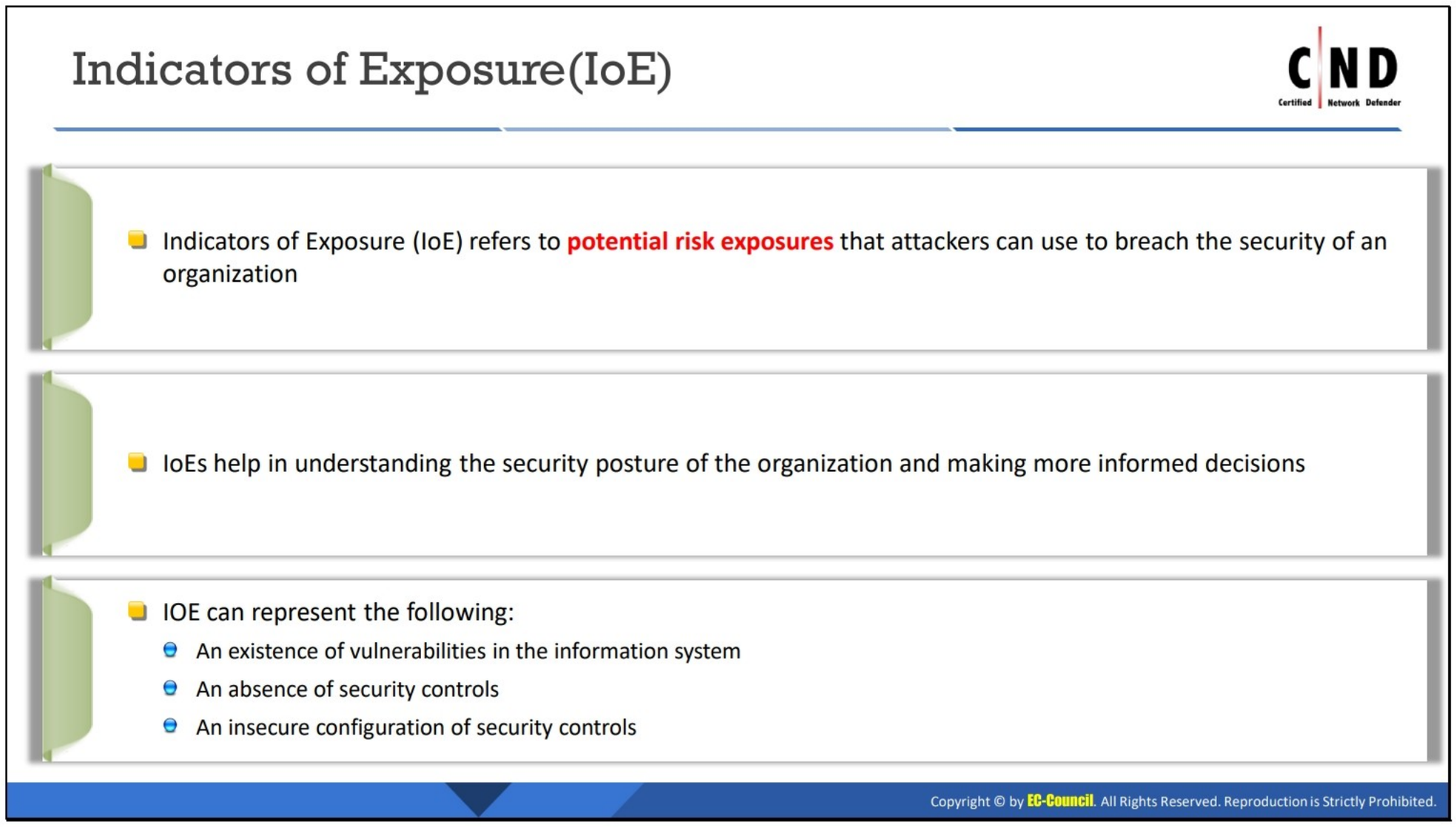


Figure 19.2: Path Visualization Using Skybox



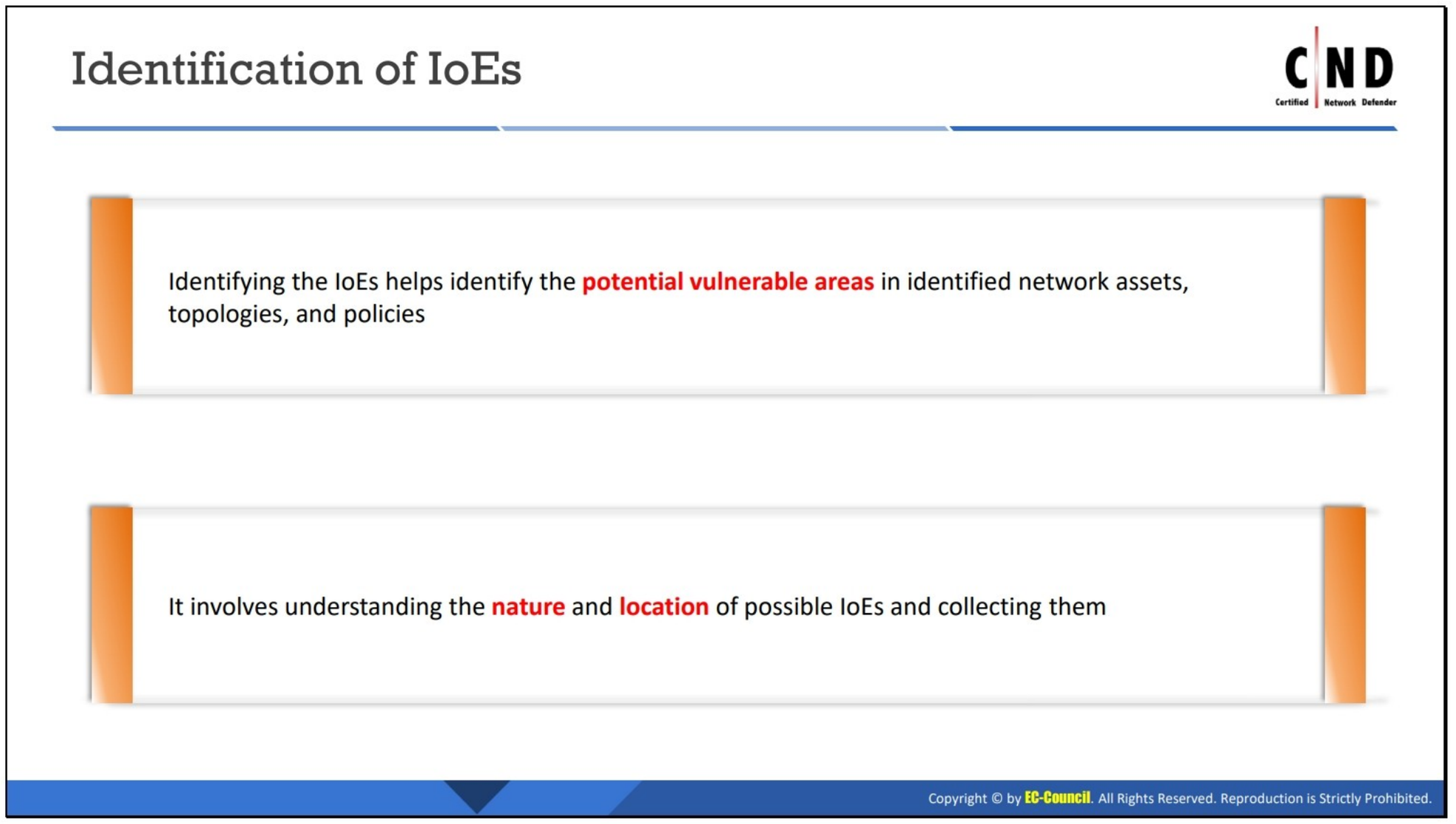
LO#03: Identify Indicators of Exposures (IoEs)

Identifying the potential risk exposure helps network defenders determine network security weaknesses of the organization that can be exploited by an attacker. Understanding the Indicators of Exposures (IoEs) helps identify potential attacks and the various assets of the organization that are at risk. This section will help you understand how to identify IoEs and the various tools to identify IoEs.



Indicators of Exposure(IoE)

IoEs refers to potential risk exposures that attackers can use to breach the security of an organization. They show potentially exploitable vectors before an incident actually occurs and help in understanding the security posture of the organization and making more informed decisions to prevent the breaches in advance. IoEs include software vulnerabilities, misconfigurations, missing security controls, overly permissive rules, and policy violations. IoEs are collected from vulnerability scanners, network and security device logs, and threat intelligence sources. IoEs can be visualized with attack surface visualization tools.

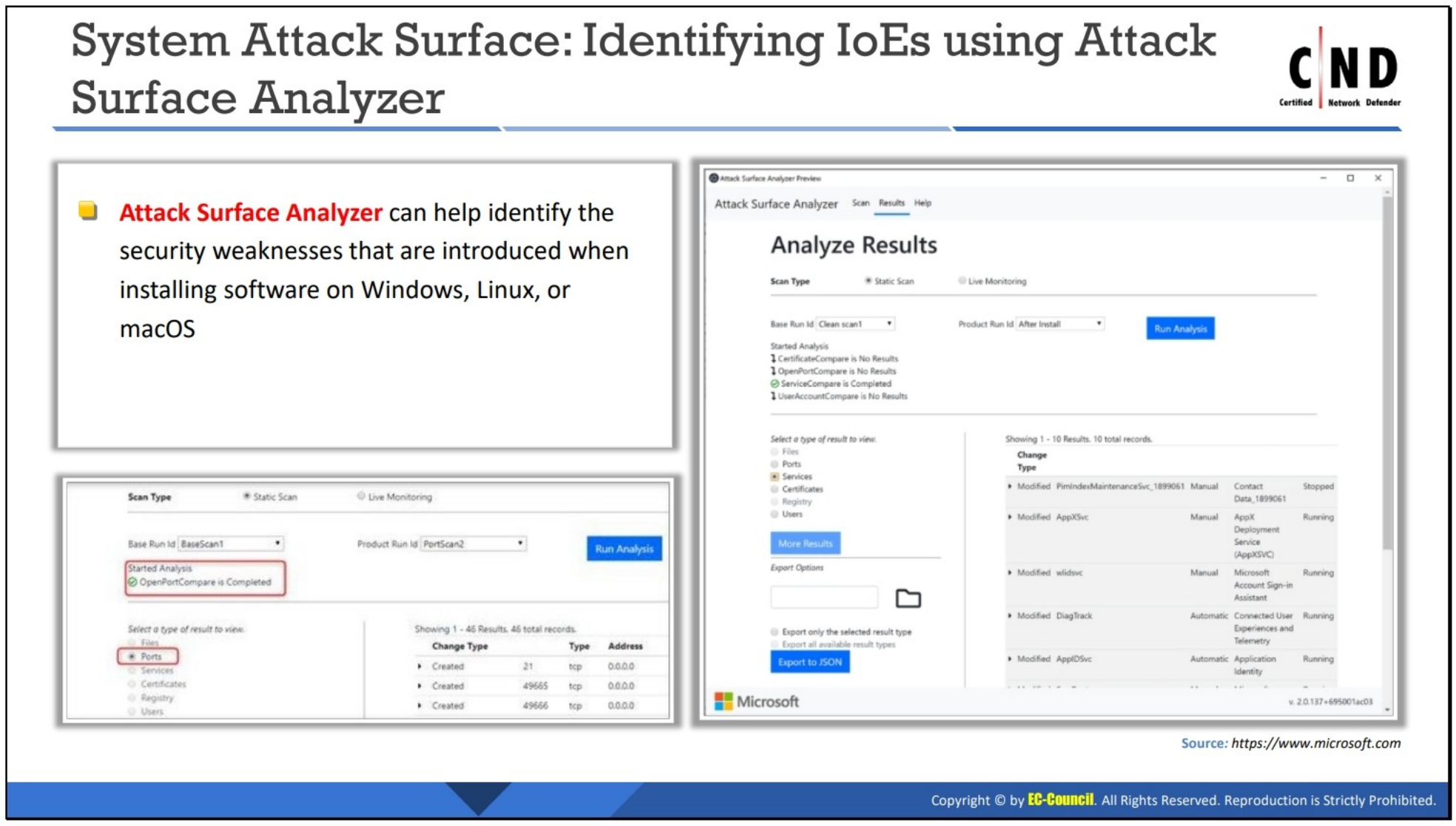


Identification of IoEs

Identifying IoEs helps identify the potential vulnerable areas in identified network assets, topologies, and policies in last step. It involves understanding the nature and location of possible IoEs and collecting them. By viewing and understanding the exploitable attack vectors in a network, the security teams focus on critical exposures and find measures to prevent data breaches.

Working with identified IoEs instead of raw vulnerabilities helps organizations use their power of contextual analysis to formulate actions that will minimize the attack surface with less effort.

Determining IoEs involves analyzing different factors, including events. For example, an unexpected firewall rule change (event) that opens up an access path to a critical asset in an IoE.



System Attack Surface: Identifying IoEs using Attack Surface Analyzer

Source: www.microsoft.com

Attack Surface Analyzer (ASA) helps identify the security weaknesses that are introduced while installing software on Windows, Linux, or macOS.

Benefits of Attack Surface Analyzer

- It shows the changes to the key elements of the system attack surface by taking a snapshot of the system before and after the installation of other software.

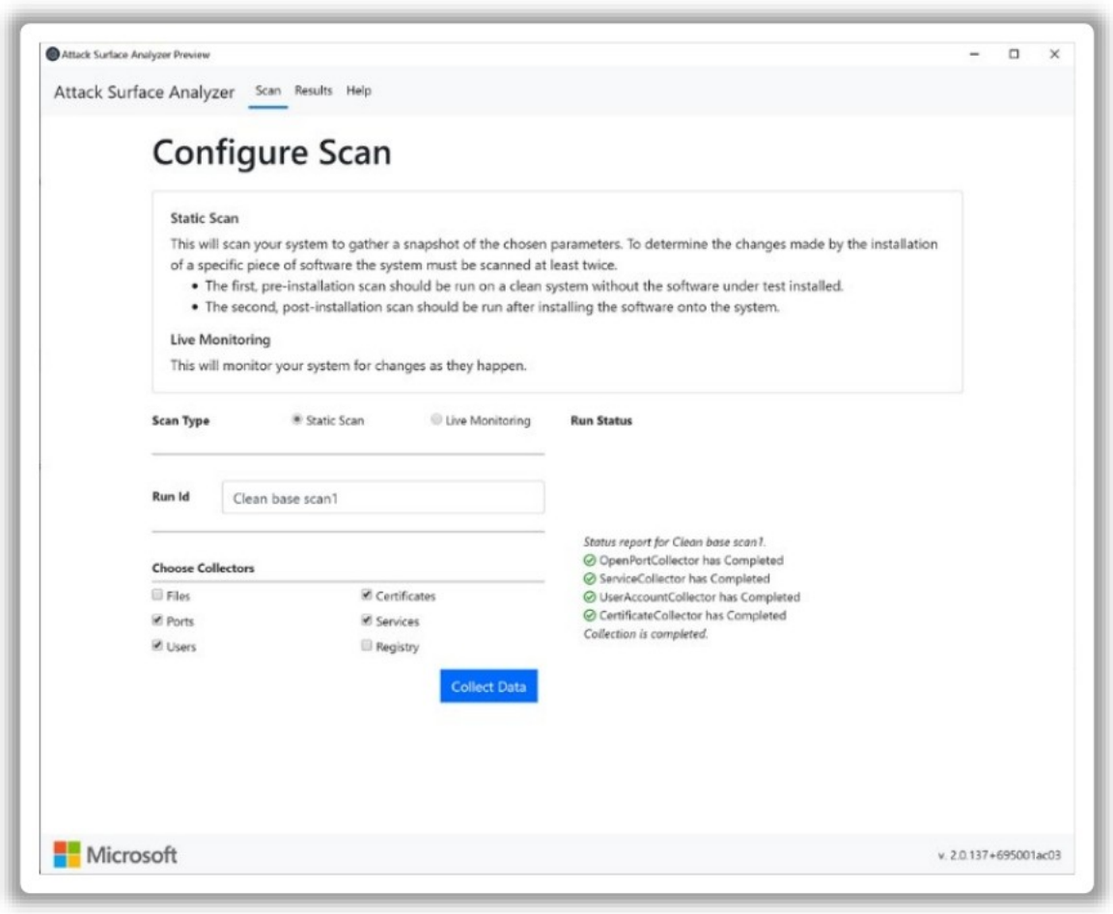


Figure 19.3: Analyze Results with Attack Surface Analyzer

- It allows developers to see the changes in the attack surface, which result from adding their code to assess the aggregation of the attack surface.
- It displays particular changes in the configuration, which may result in potential threats.
- It determines threat severity and shows the severity of the threat by its category.
- The tool comprises an Electron-based Graphical User Interface (GUI) and command line interface options. The results for the command line can be written to a local HTML or JSON file. The snapshots are stored in a local SQLite database that are used to generate reports on system changes.

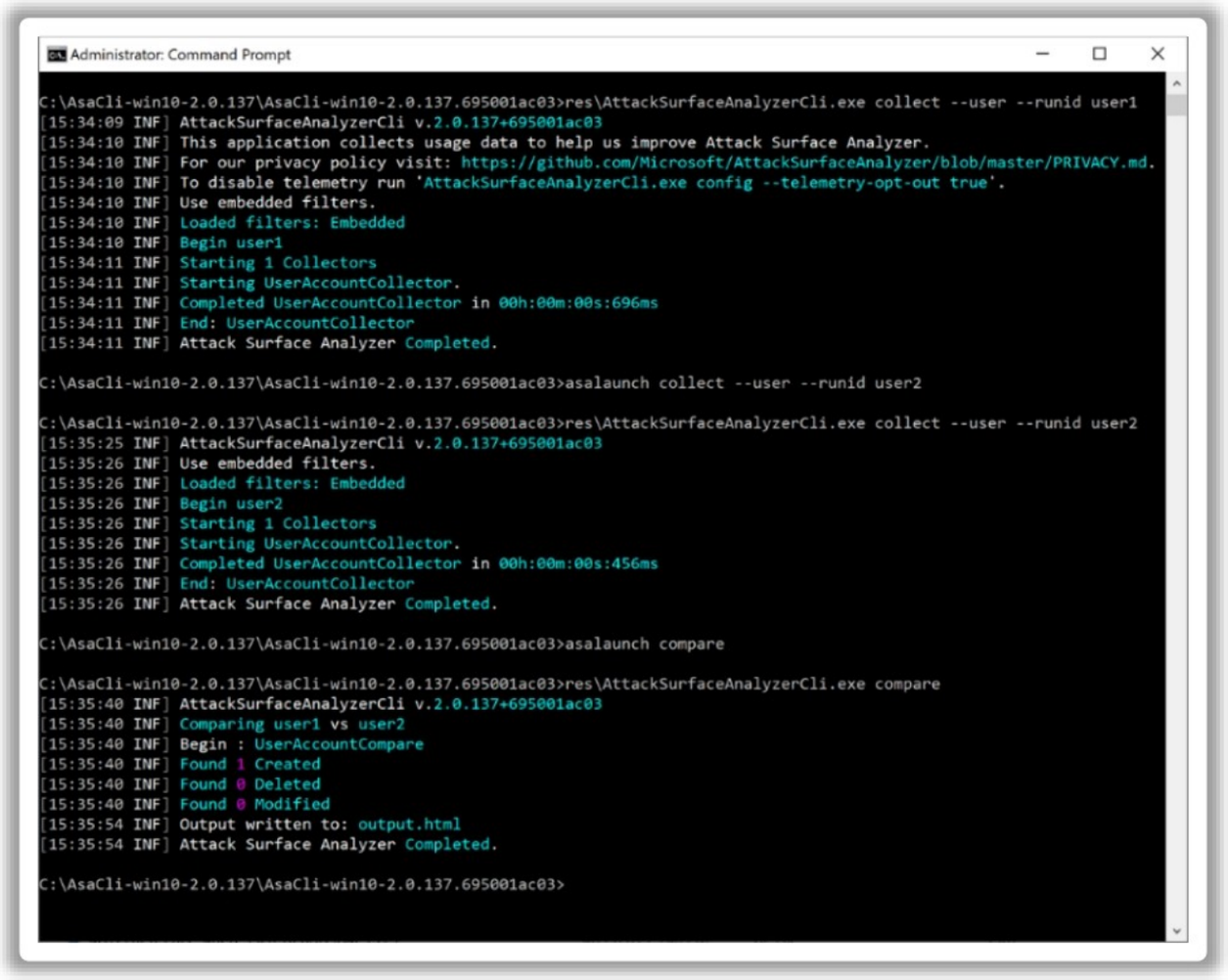
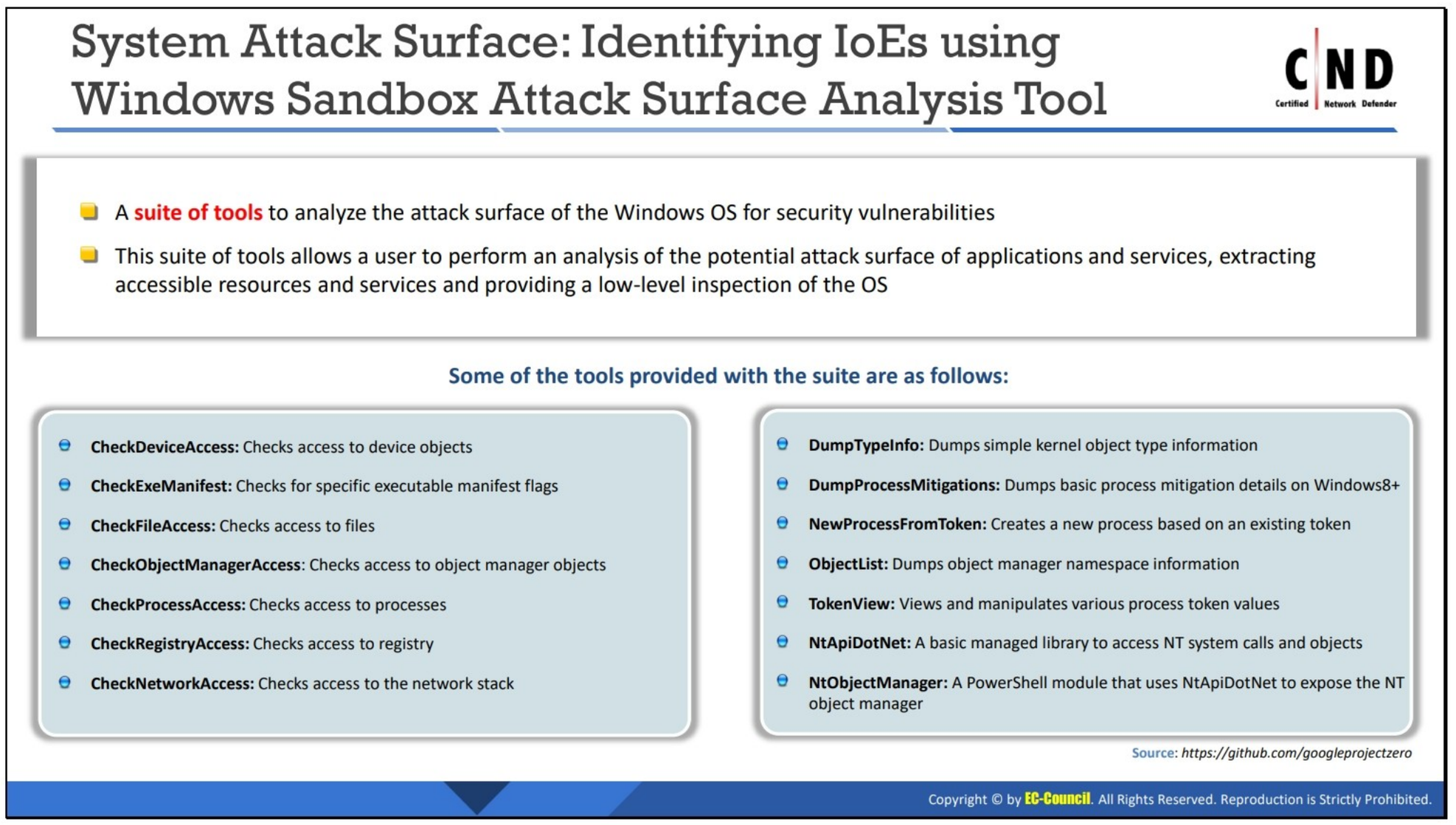


Figure 19.4: Attack Surface Analyzer Command Line Interface

Key Features

The Attack Surface Analyzer reports on changes to the following OS components:

- File system
- Network ports
- Certificates
- Event logs
- Registry
- Services
- Component Object Model (COM) objects
- User accounts
- Firewall settings



System Attack Surface: Identifying IoEs using Windows Sandbox Attack Surface Analysis Tool

Source: https://github.com/googleprojectzero

Windows Sandbox Attack Surface Analysis Tool is a tools suite that analyzes the attack surface of the Windows OS for security vulnerabilities. Researchers and developers find it to be really useful to verify the security of products on Windows OS.

A few of the objects inspected by this tool include NT system calls and objects, files, ports, network stack, running processes, and registry.

With tools suite, you can perform the following:

- Analysis of the potential attack surface of applications and services
- Extraction of accessible resources and services
- Low-level inspection of the OS

Some of the tools provided with the suite and their functions are as follows:

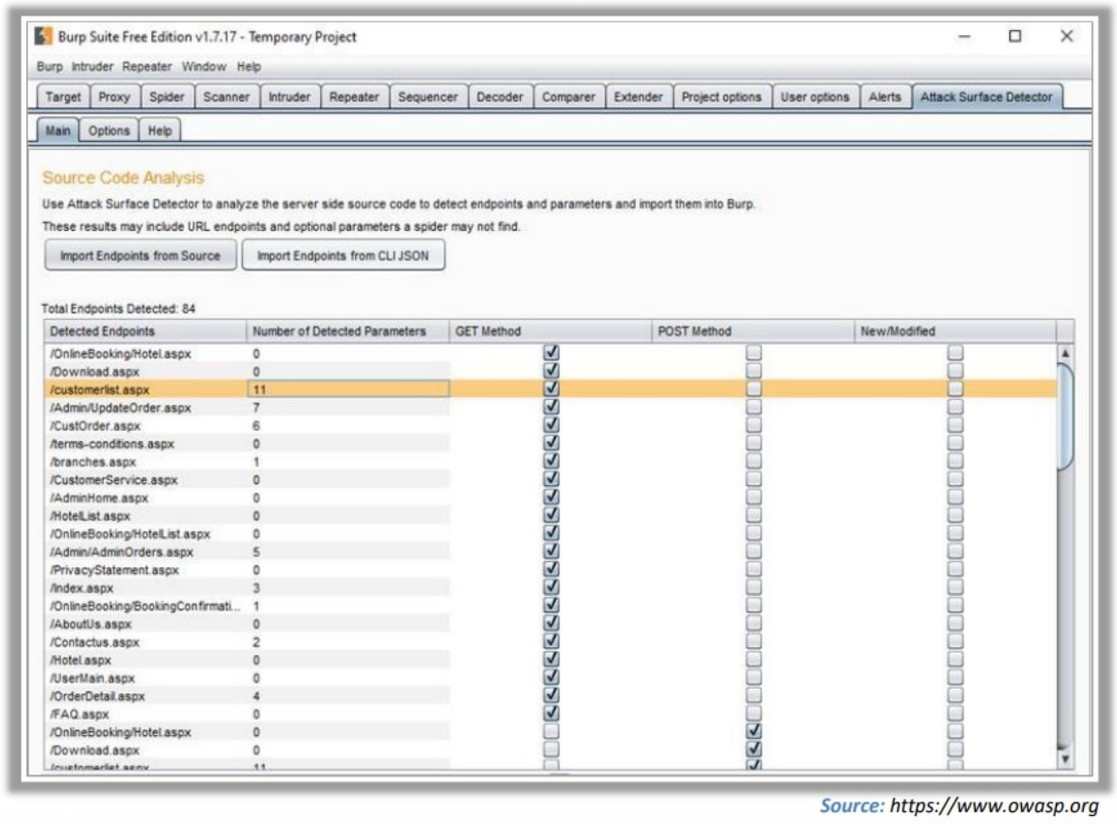
- **CheckDeviceAccess** checks access to device objects.
- **CheckExeManifest** checks for specific executable manifest flags.
- **CheckFileAccess** checks access to files.
- **CheckObjectManagerAccess** checks access to object manager objects.
- **CheckProcessAccess** checks access to processes.
- **CheckRegistryAccess** checks access to the registry.

- **CheckNetworkAccess** checks access to the network stack.
- **DumpTypeInfo** dumps simple kernel object type information.
- **DumpProcessMitigations** dumps basic process mitigation details on Windows8+.
- **NewProcessFromToken** creates a new process based on an existing token.
- **ObjectList** dumps object manager namespace information.
- **TokenView** views and manipulates various process token values.
- **NtApiDotNet** provides a basic managed library to access NT system calls and objects.
- **NtObjectManager** provides a PowerShell module that uses NtApiDotNet to expose the NT object manager.

Application Attack Surface: Identifying IoEs using OWASP Attack Surface Detector

- The Attack Surface Detector tool uncovers the **endpoints** of a web application, the parameters that these endpoints accept, and the data type of these parameters. This includes the unlinked endpoints that a spider will not find in the client-side code or optional parameters that are completely unused in the client-side code
- It also has the capability to calculate the changes in the attack surface between two **versions** of an application
- The Attack Surface Detector is available as a plugin to both ZAP and **Burp Suite** and as a Command Line Interface (CLI) tool

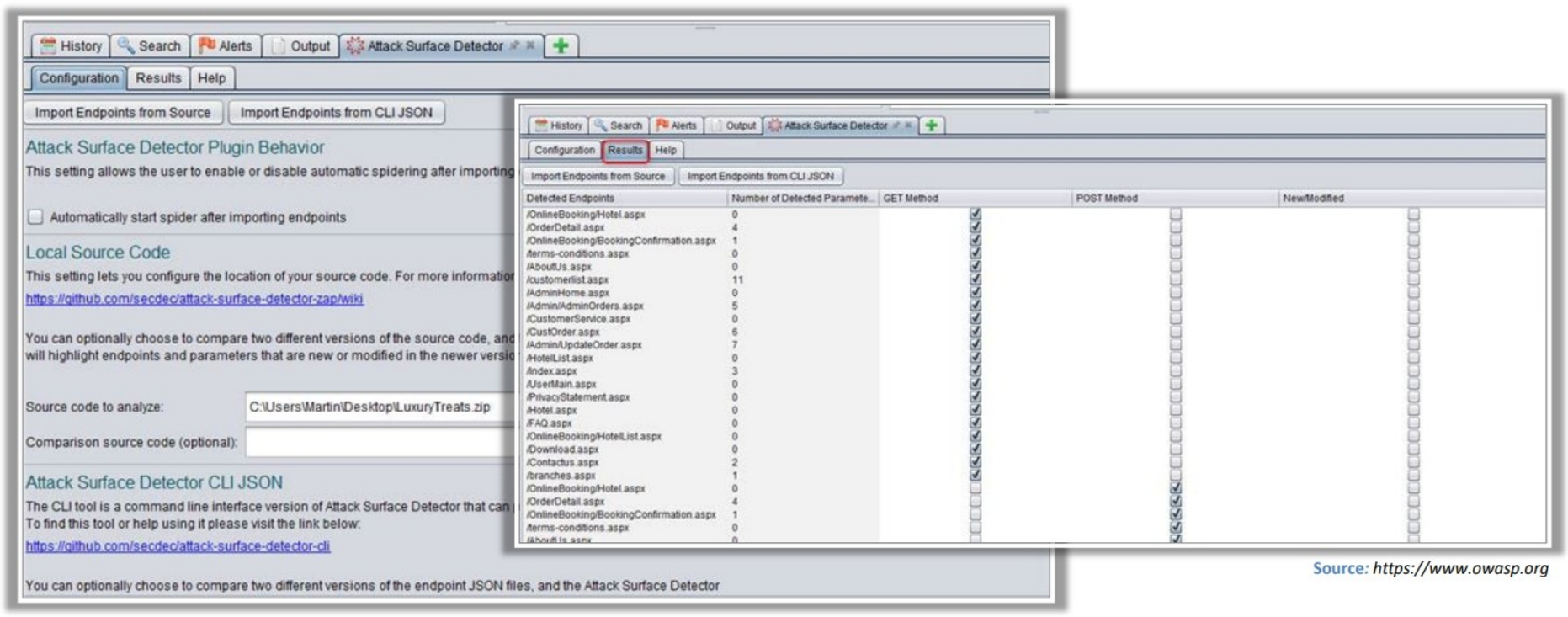
Attack Surface Detector plugin for Burp Suite



Source: https://www.owasp.org

Application Attack Surface: Identifying IoEs using OWASP Attack Surface Detector (Cont'd)

Attack Surface Detector plugin for OWASP ZAP



Source: https://www.owasp.org

Application Attack Surface: Identifying IoEs using OWASP Attack Surface Detector

Source: https://owasp.org

The Attack Surface Detector (ASD) tool can do the following:

- Find the endpoints of a web application. Perform static code analyses to identify web application endpoints by parsing routes and identifying parameters. The data are made available in OWASP ZAP and Burp Suite to help improve testing coverage.
- Find the allowed parameters by the endpoints and the data type of parameters. This includes the unlinked endpoints that a spider will not find in the client-side code or optional parameters that are completely unused in the client-side code.
- Calculate the changes in the attack surface between two versions of an application.

Key Features

- The Attack Surface Detector is available as a ZAP plugin and PortSwigger BApp Store and can be installed directly from within those tools
- The Burp Suite Attack Surface Detector plugin that shows a list of endpoints, endpoint details, and their corresponding requests

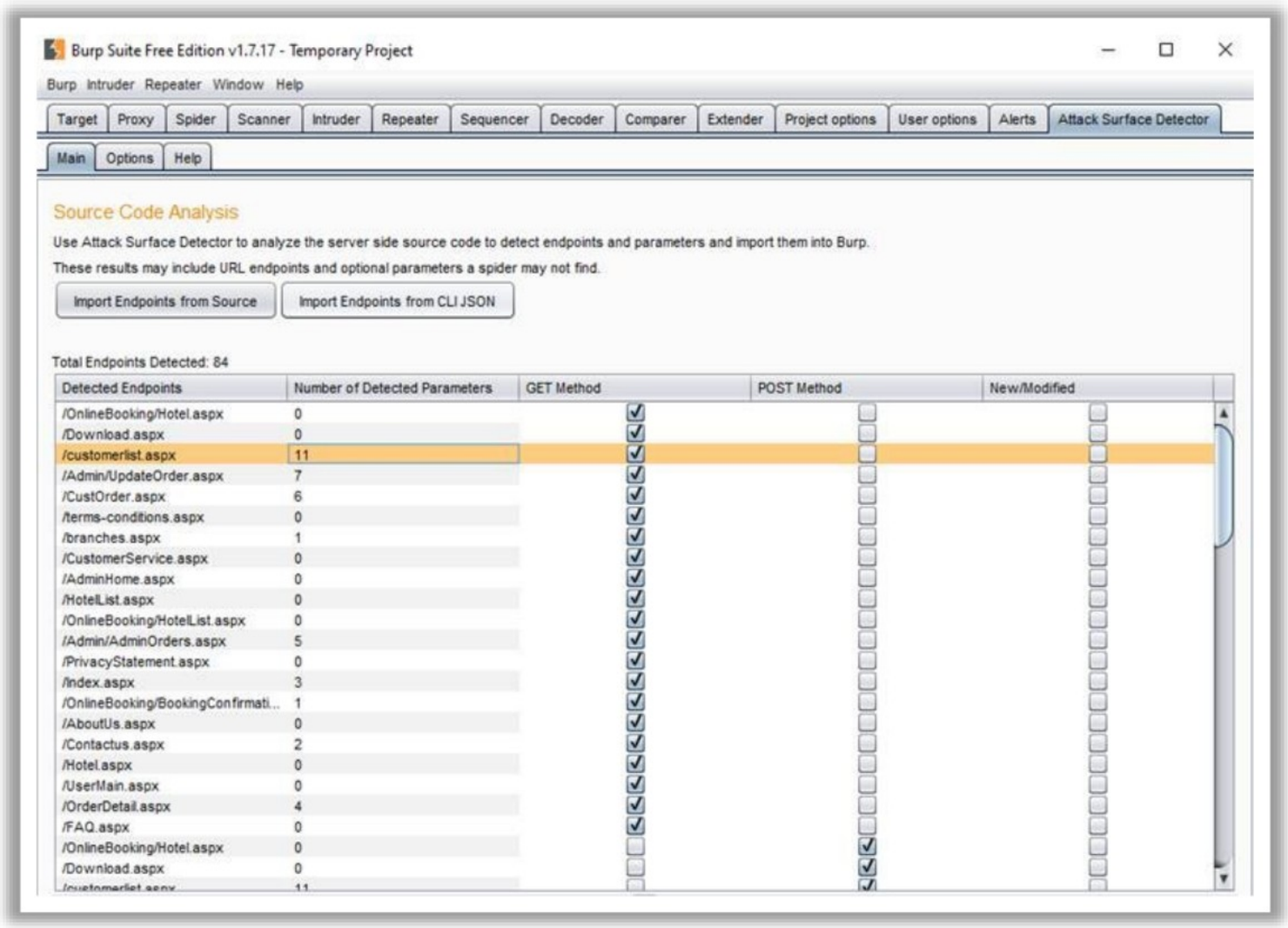
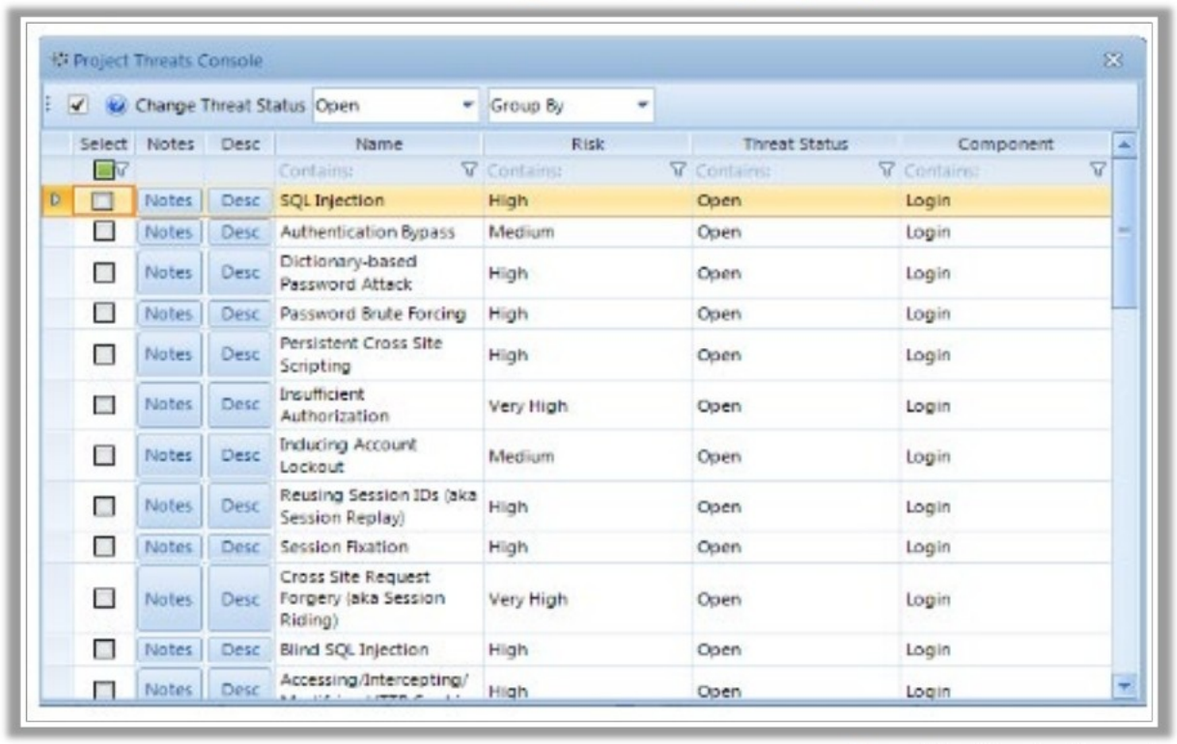


Figure 19.5: Burp Suite

Application Attack Surface: Identifying IoEs using ThreatModeler

- **ThreatModeler** is a threat modeling software that assists organizations in **managing** their **attack surface** and avoiding threats.

ThreatModeler: Threat Management Console Grouped by Components



Source: *https://threatmodeler.com/*

Application Attack Surface: Identifying IoEs using ThreatModeler

Source: https://threatmodeler.com

ThreatModeler is an automated threat modeling software that assists organizations in managing their attack surface and avoiding threats. It allows defining a communication channel (protocols) between components and allocates data elements and widgets (Cookie, Session, Form, or URL) to these components. If the user finishes the component diagram, ThreatModeler's intelligent threat engine automatically recognizes threats and automatically prioritizes the threats according to the risk level.

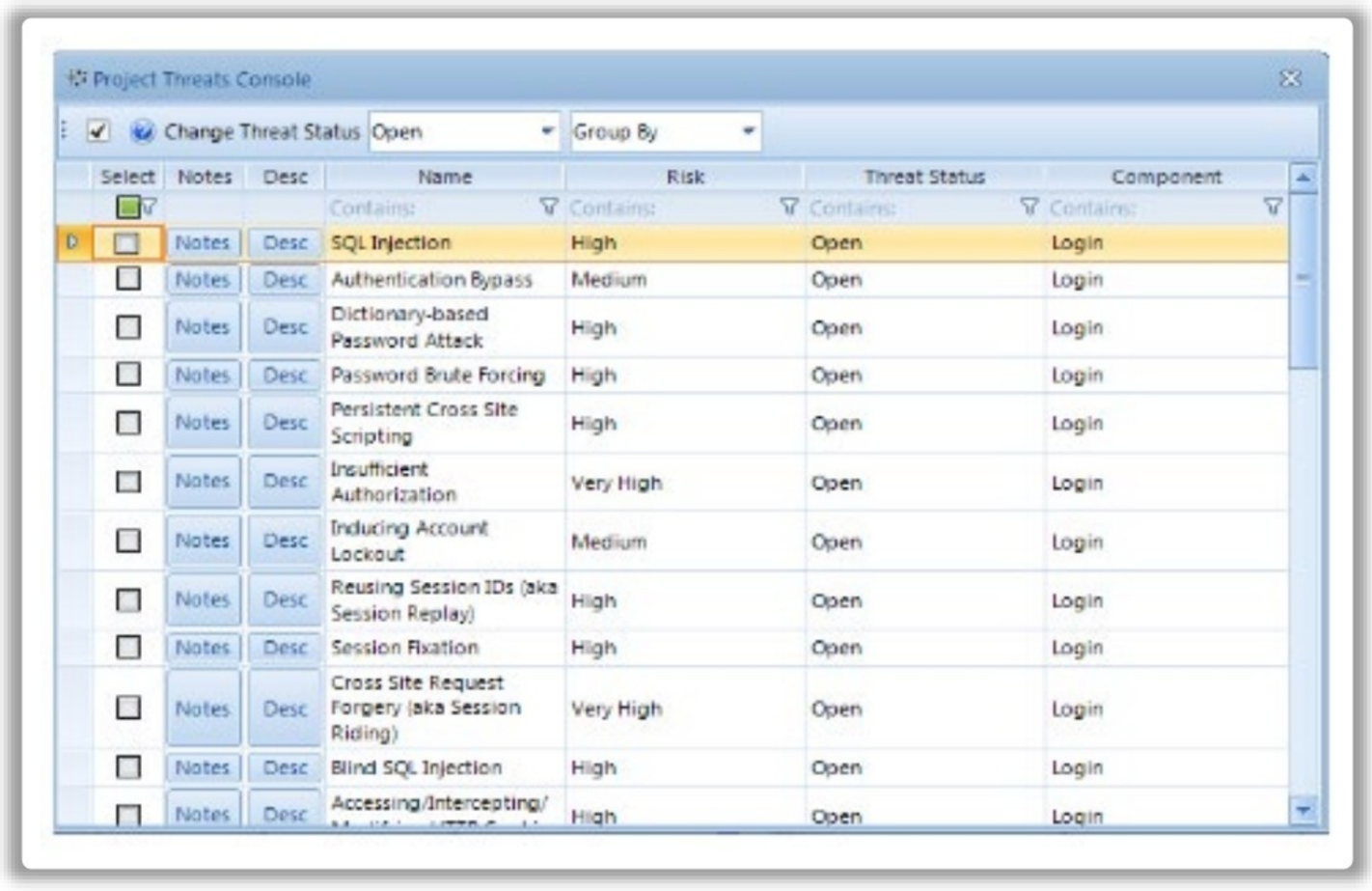
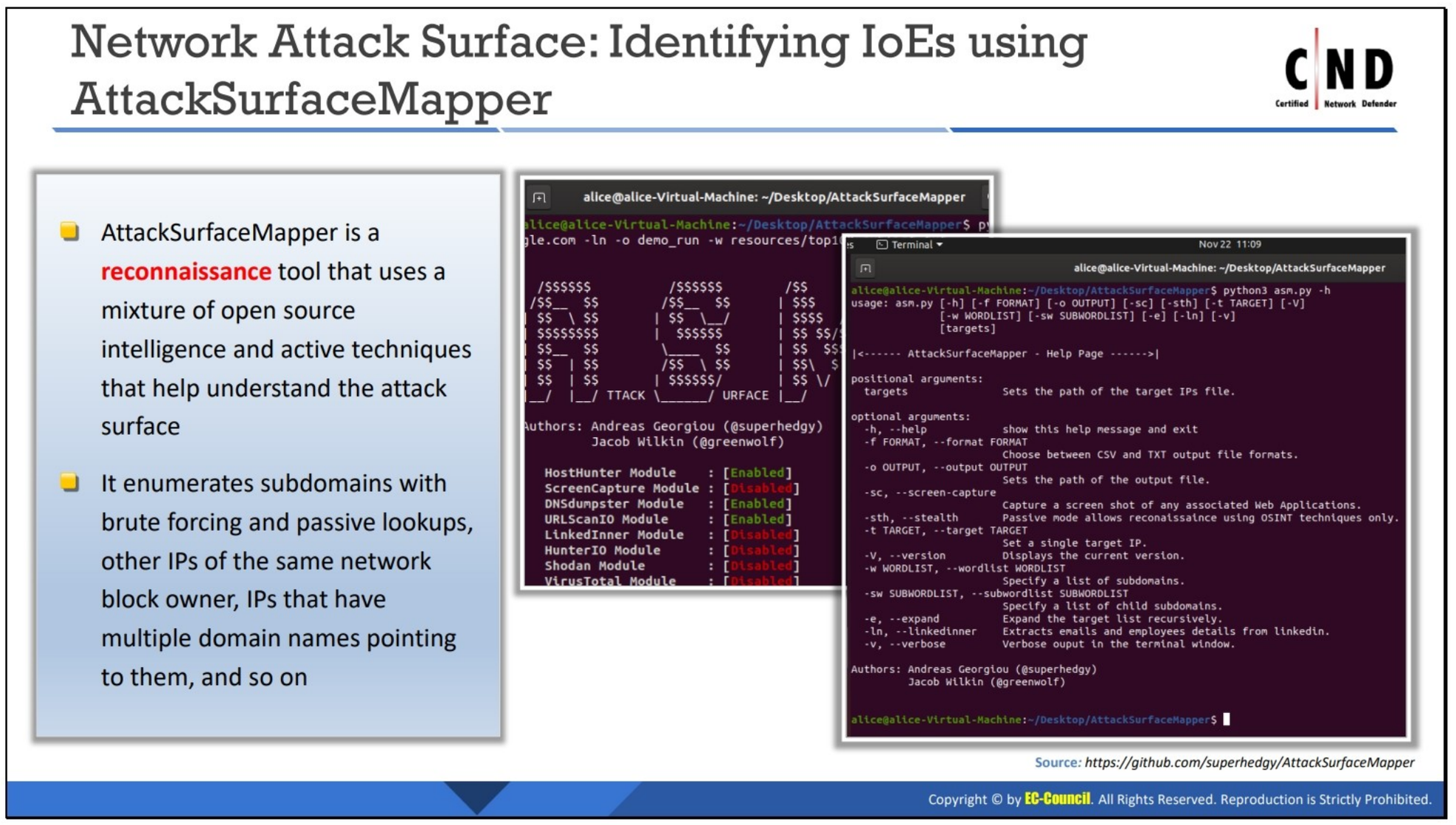


Figure 19.6: ThreatModeler



Network Attack Surface: Identifying IoEs using AttackSurfaceMapper

Source: https://github.com/superhedgy/AttackSurfaceMapper

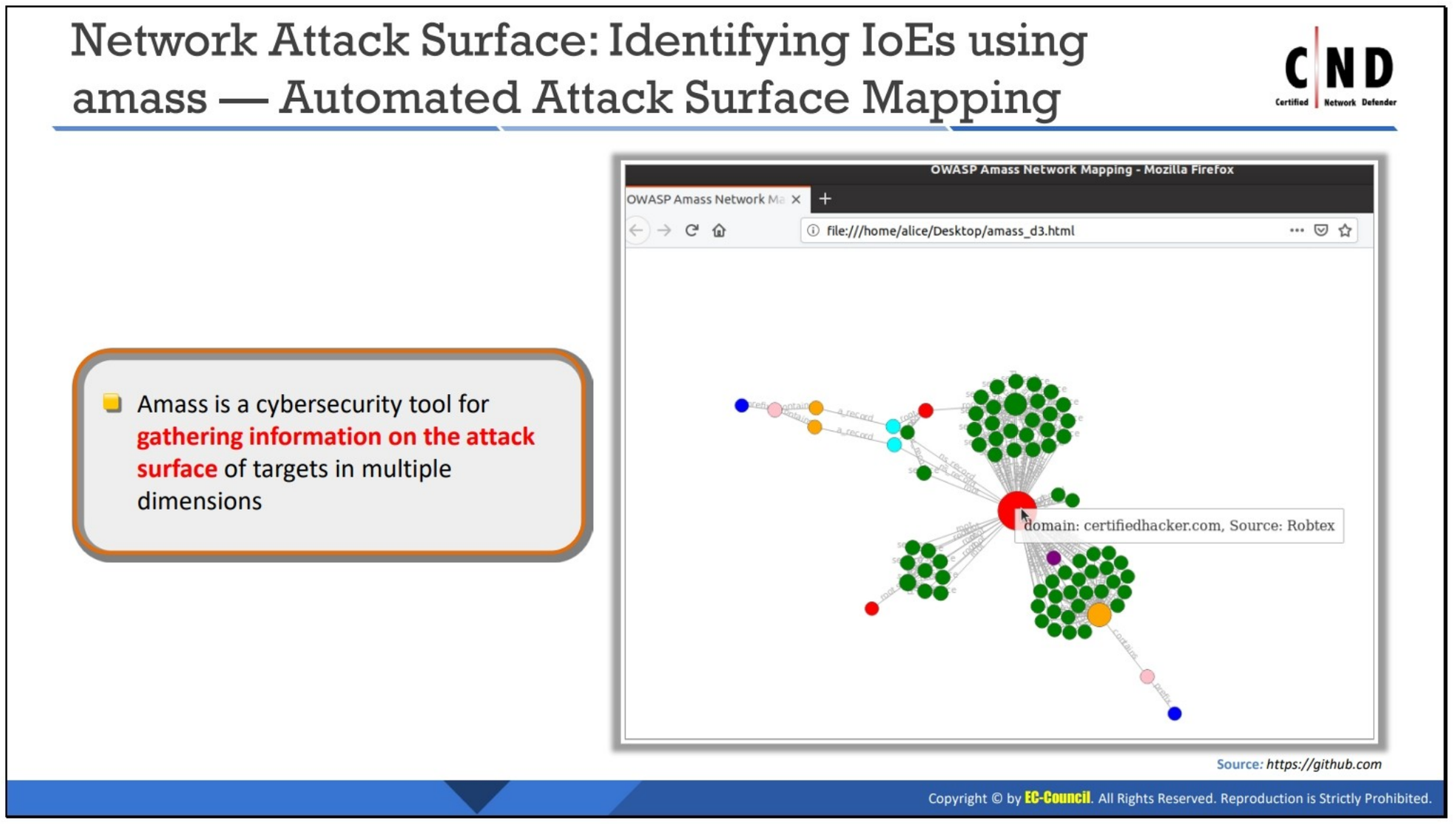
The reconnaissance AttackSurfaceMapper (ASM) tool uses open source intelligence and active techniques to expand the attack surface of the target. It uses a number of techniques to find more targets by consuming a mix of one or more domains, subdomains, and IP addresses.

It can enumerate the following:

- Subdomains with brute forcing and passive lookups
- Other IPs of the same network block owner
- IPs that have multiple domain names pointing to them

If the target list is fully expanded, ASM performs the following:

- Passive reconnaissance on multiple domain names
- Takes screenshots of websites
- Generates visual maps
- Looks up credentials in public breaches
- Passive port scanning with Shodan
- Scraps employees from LinkedIn



Network Attack Surface: Identifying IoEs using amass — Automated Attack Surface Mapping

Source: https://github.com/OWASP/Amass

The OWASP Amass Project is a cybersecurity tool for gathering information on the attack surface of targets in multiple dimensions. It allows performing network mapping of the attack surface and also performs external asset discovery by collecting open-source information and using reconnaissance techniques (OSINT Reconnaissance).

The information gathering techniques used include the following.

Domain Name System (DNS)	Basic enumeration, Brute forcing (optional), Reverse DNS sweeping, Subdomain name alterations or permutations, and Zone transfers (optional)
Scraping	Ask, Baidu, Bing, DNSDumpster, DNSTable, Dogpile, Exalead, Google, HackerOne, IPv4Info, Netcraft, PTRArchive, Riddler, SiteDossier, ViewDNS, and Yahoo
Certificates	Active pulls (optional), Censys, CertSpotter, Crtsh, Entrust, and GoogleCT
Application Program Interfaces (APIs)	AlienVault, BinaryEdge, BufferOver, CIRCL, CommonCrawl, DNSDB, GitHub, HackerTarget, Mnemonic, NetworksDB, PassiveTotal, Pastebin, RADb, Robtex, SecurityTrails, ShadowServer, Shodan, Spyse (CertDB & FindSubdomains), Sublist3rAPI, TeamCymru, ThreatCrowd, Twitter, Umbrella, URLScan, VirusTotal, and WhoisXML
Web Archives	ArchiveIt, ArchiveToday, Arquivo, LoCArchive, OpenUKArchive, UKGovArchive, and Wayback

Table 19.1: Information Gathering Techniques

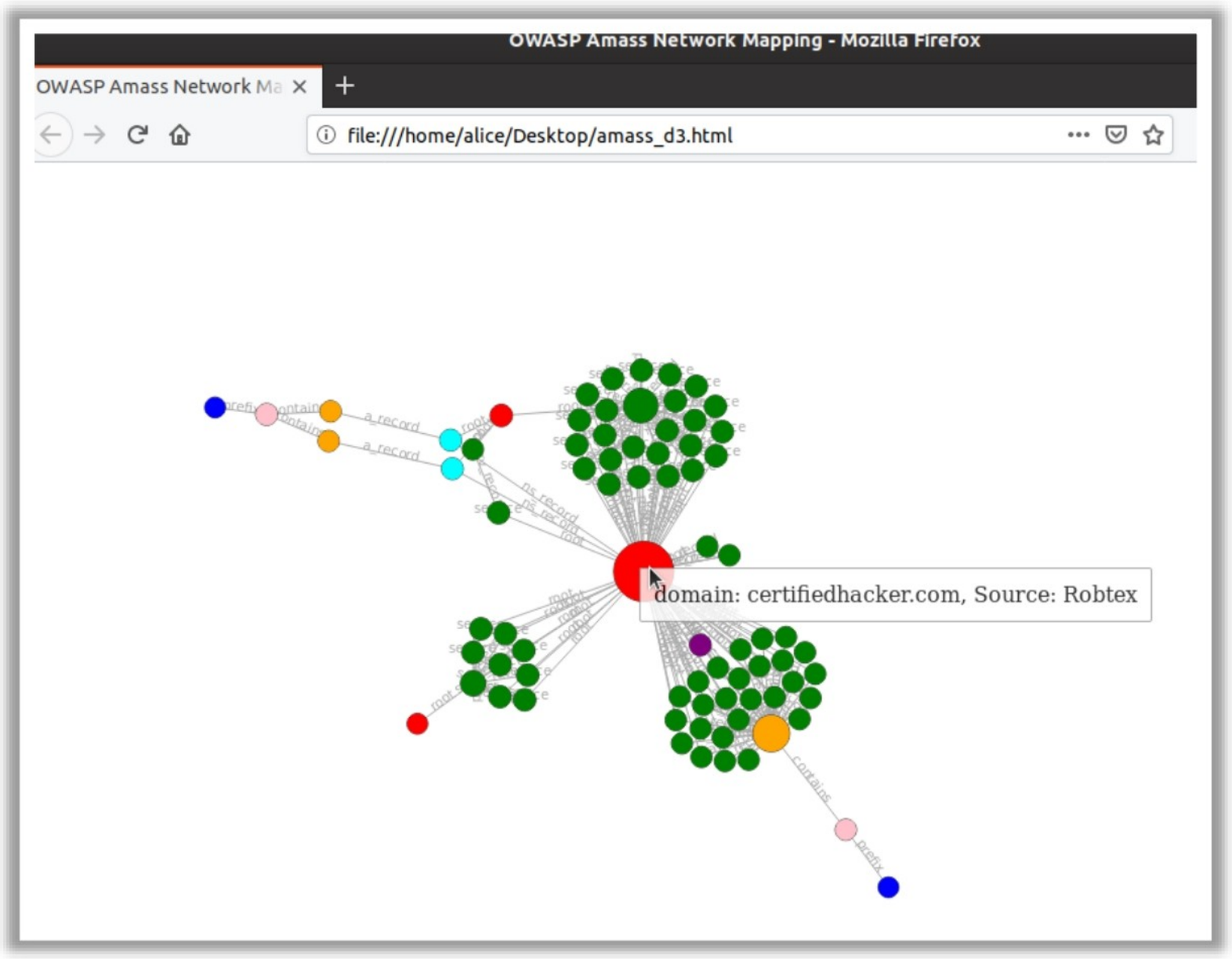
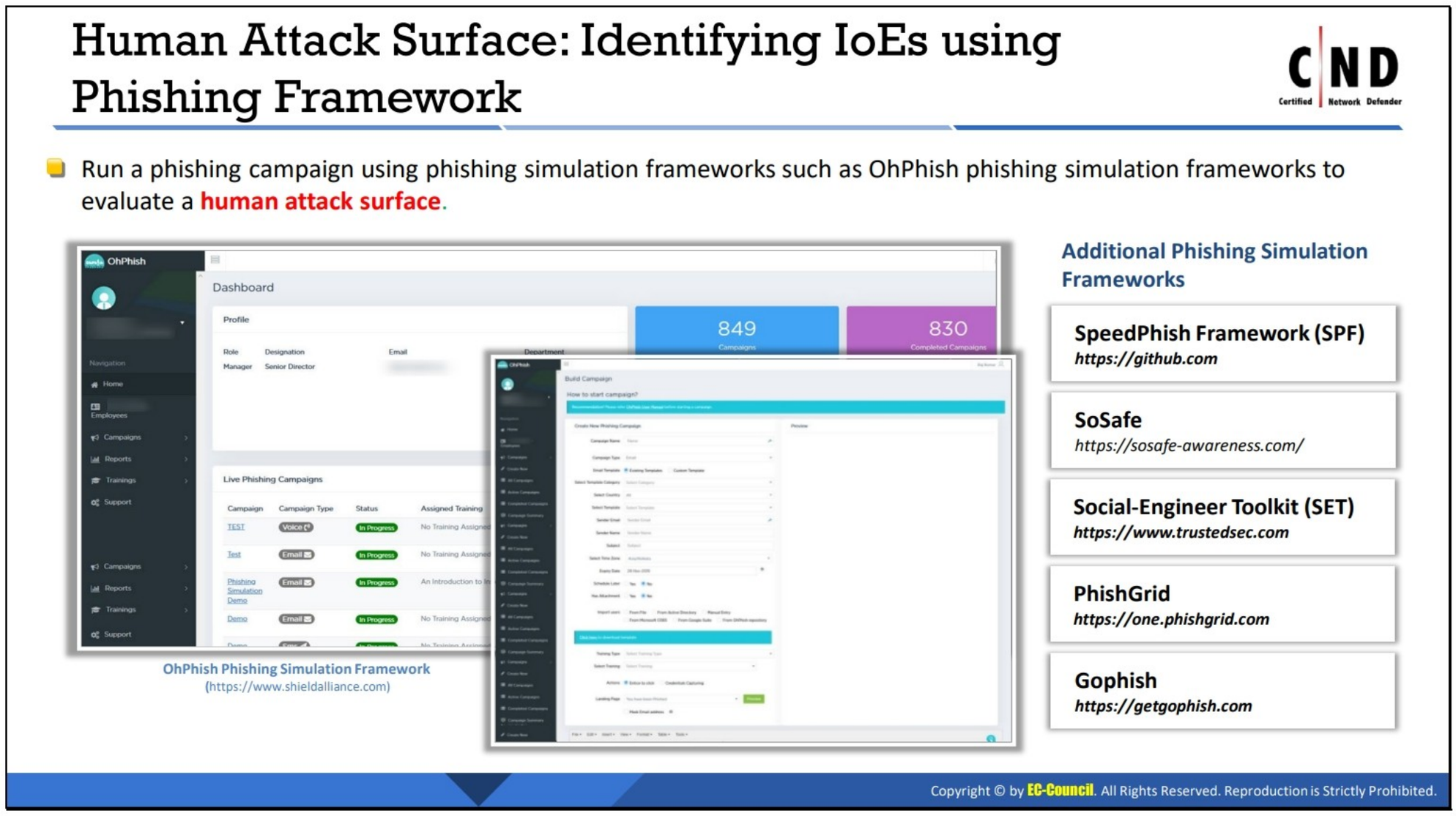


Figure 19.7: Automated Attack Surface Mapping



Human Attack Surface: Identifying IoEs using Phishing Framework

To identify and evaluate the human attack surface, use phishing frameworks to identify the IoEs related to human behavior. Phishing frameworks help fight phishing and social-engineering attacks by enabling users to do the following:

- Continuous simulation and training to get the latest attack techniques
- Recognizing subtle clues
- Stopping email fraud
- Stopping data loss and brand damage

OhPhish

Source: www.shieldalliance.com

OhPhish is a phishing simulation framework that mitigates cybersecurity risks, particularly those involving human error. It combines simulated phishing attacks with set-and-go training modules and enhances awareness, changes user behavior, and mitigates risk with social engineering attacks.

Its key features include a simple and user-friendly solution, extensive reports, predefined templates, theme-based campaigns, trend monitoring, and analytics.

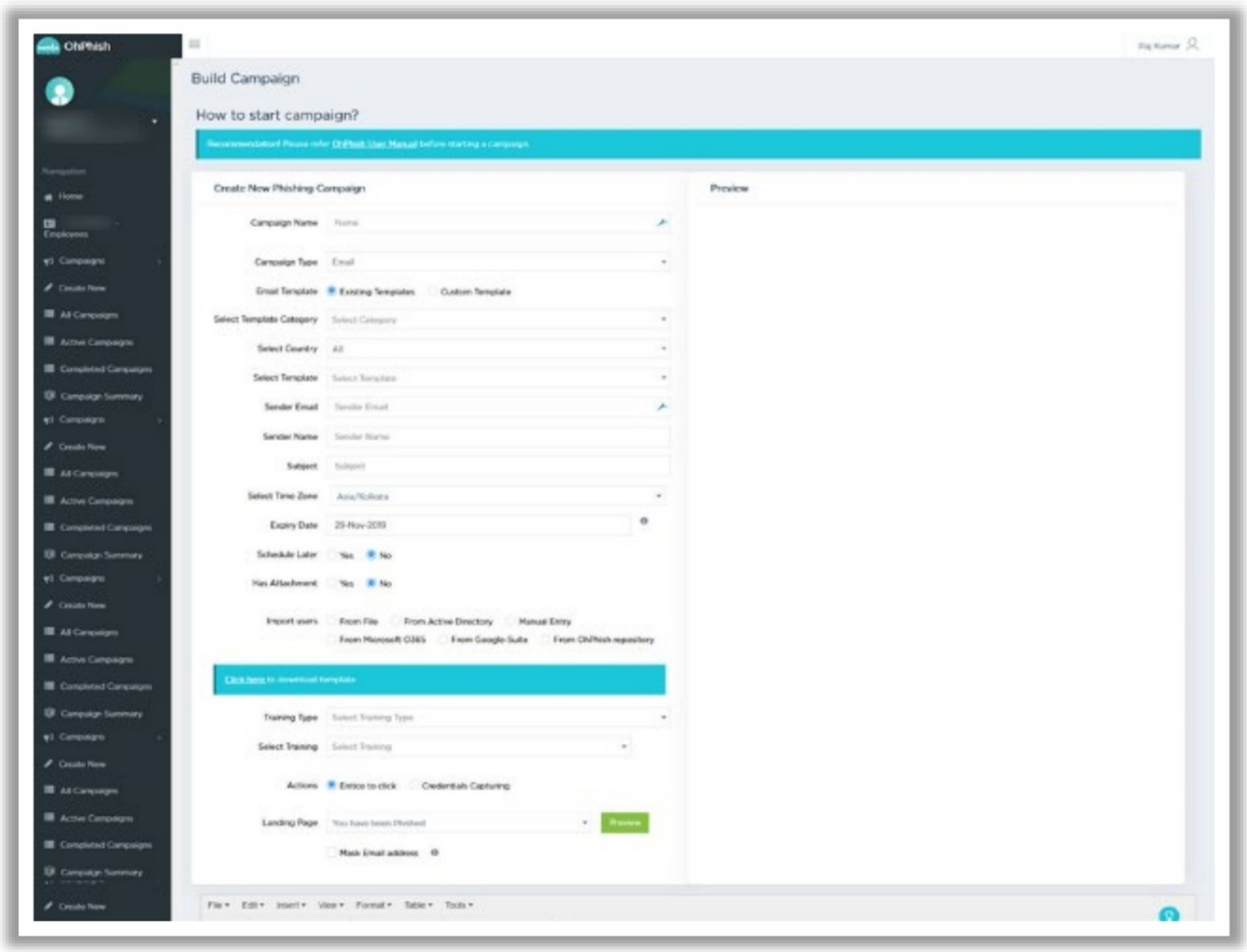


Figure 19.8: OhPhish

Additional Phishing Simulation Frameworks

- **SpeedPhish Framework (SPF)**

  Source: https://github.com

  The Python tool SpeedPhish Framework (SPF) allows for quick reconnaissance and deployment of simple social engineering phishing exercises.

- **SoSafe**

  https://sosafe-awareness.com/
  Empower employees to identify and report phishing attempts with this comprehensive training and simulation platform. Helps organizations build a stronger security culture and mitigate the risk of phishing attacks.

- **PhishGrid**

  Source: https://one.phishgrid.com/

  Phishing simulations that help workforce understand, detect and neutralize threats. Lower threat response time by transforming employees into active defenders with safer behaviors. Empower employees to improve threat reporting skills and lower detection time with greater security and spear phishing awareness spurred by smart reporting tools.

- **Social-Engineer Toolkit (SET)**

  Source: www.trustedsec.com

The open-source Python tool Social-Engineer Toolkit (SET) targets penetration testing around social engineering. It has a number of custom attack vectors that allow to make a legitimate attack quickly. The built-in attacks of the toolkit can be targeted against a person or organization used during a penetration testing.

- **Phishing Frenzy**

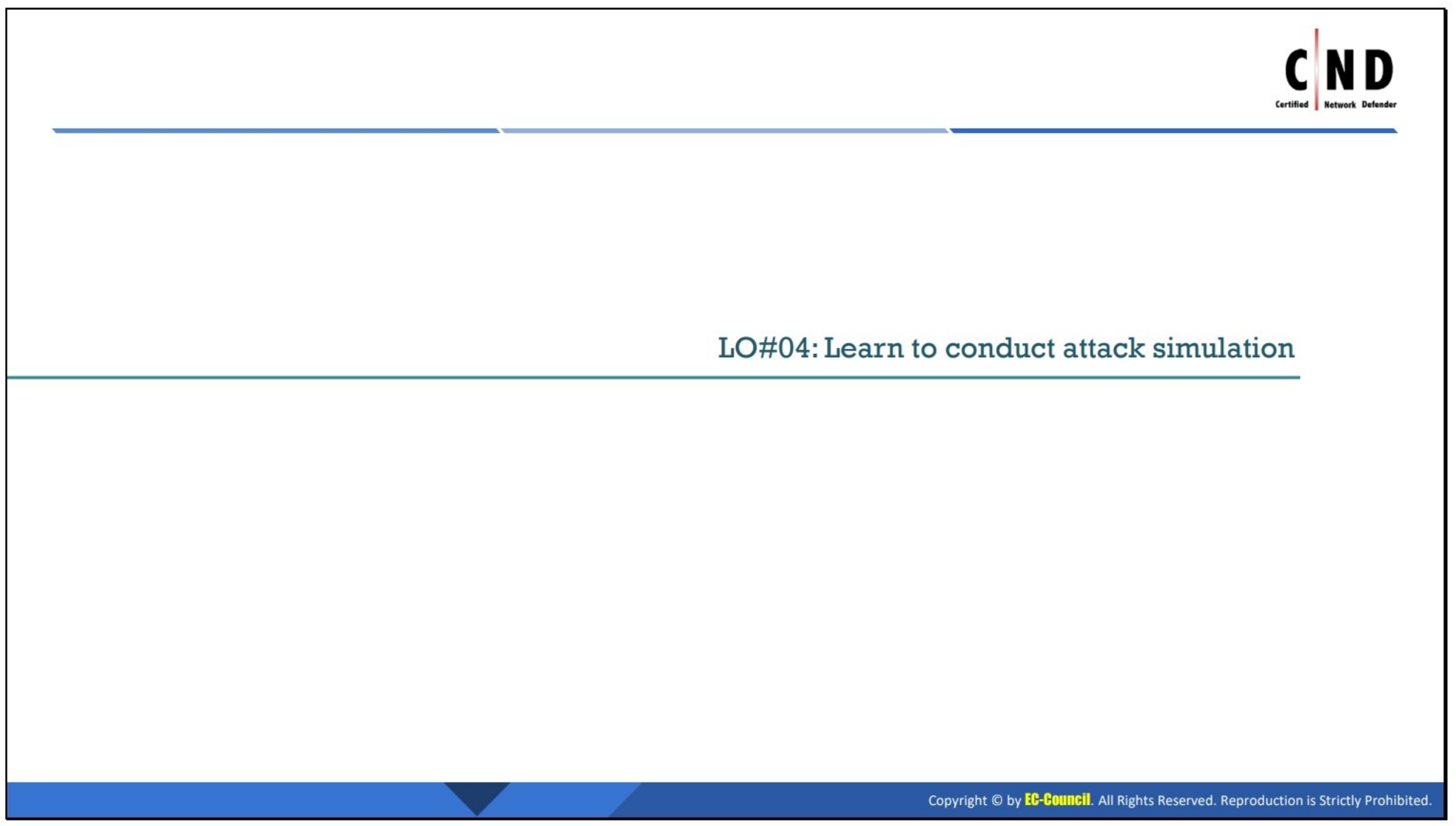
  Source: www.github.com

  Phishing Frenzy is an Open-Source Ruby on Rails e-mail phishing framework. It allows pen testers to manage multiple and complex phishing campaigns. It offers campaign management, template reuse, and statistical generation to streamline the phishing process while providing clients realistic phishing campaigns. It works by sending emails, hosing websites, and tracking analytics.

- **GoPhish**

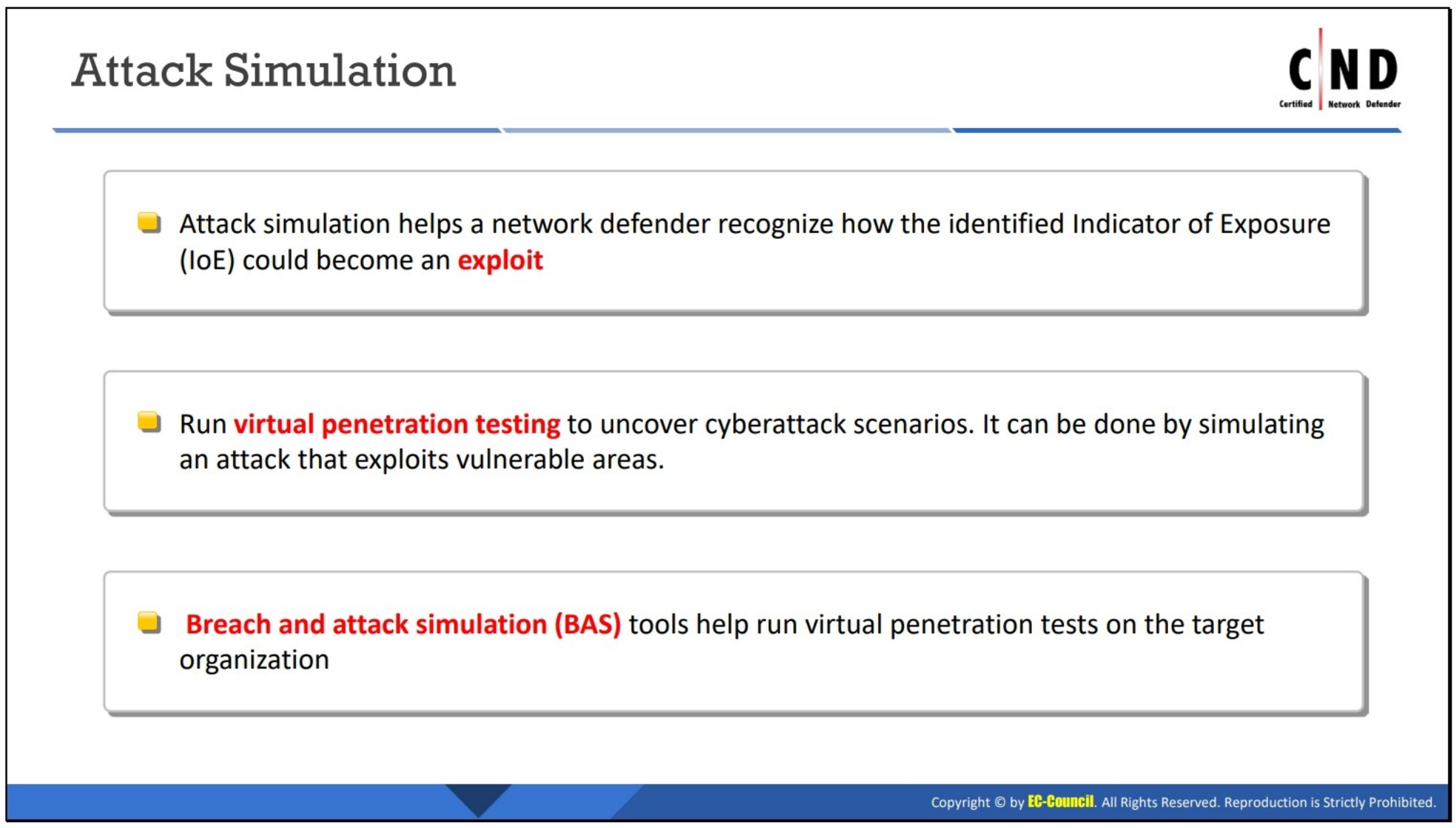
  Source: www.getgophish.com

  GoPhish is an open-source phishing framework to test an organization's exposure to phishing. It enables organizations to set templates and targets, launch campaigns, and track results.



LO#04: Conduct Attack Simulation

Conducting an attack simulation helps a network defender validate and manage the security controls across the organization. It enables a network defender to assess the security flaws before any attack takes place. This section will help you understand the benefits of attack simulation and how to conduct an attack simulation using breach and attack simulation (BAS) tools.



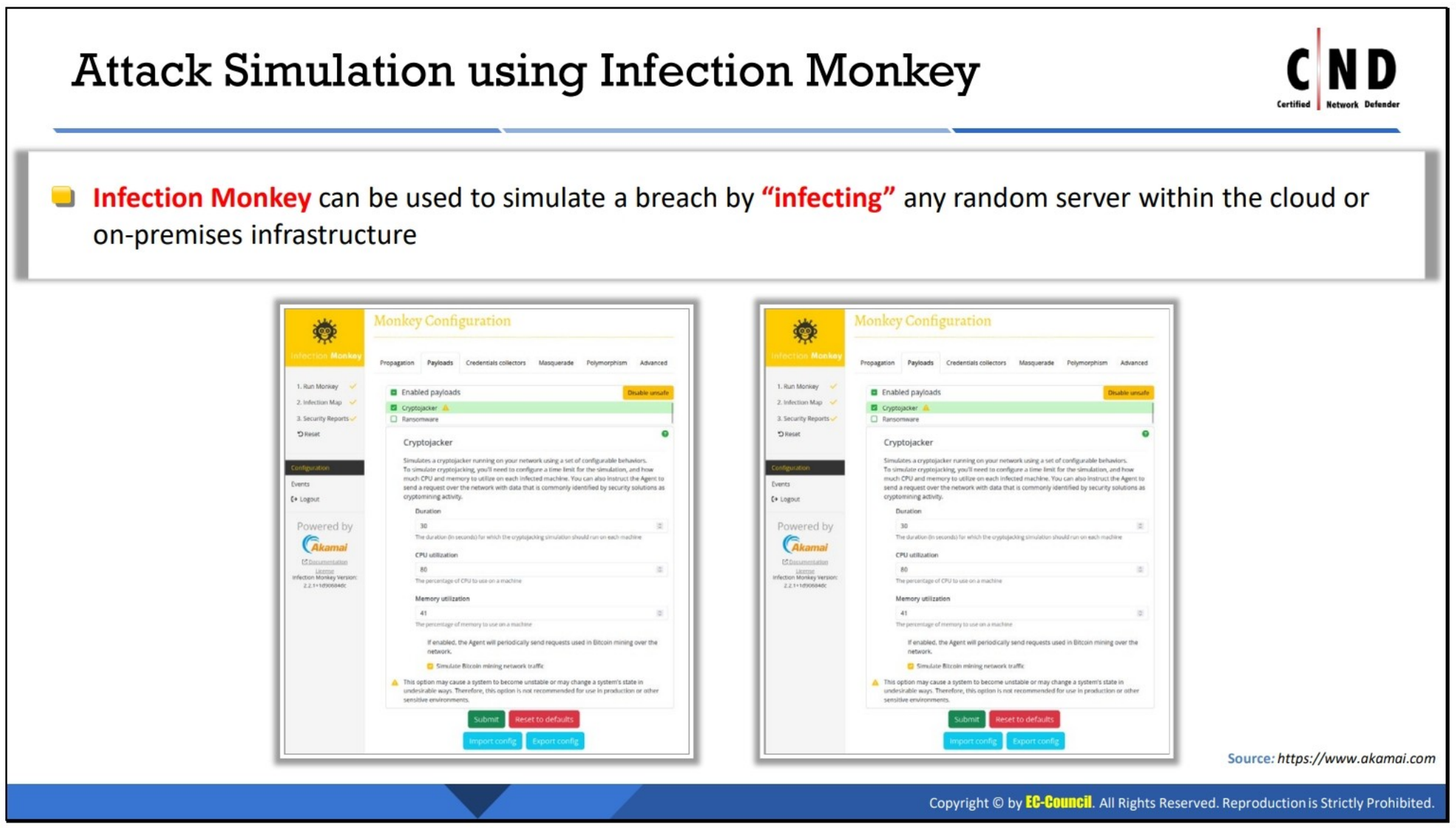
Attack Simulation

An attack simulation helps a network defender recognize how identified Indicators of Exposures (IoEs) could become an exploit or how the organization looks from the attacker's perspective. It involves running virtual penetration testing to uncover cyberattack scenarios. It can be done by simulating an attack that aims to exploit vulnerable areas. BAS tools help run a virtual penetration test on the target organization. The attack simulation looks at an organization as a single unit but focuses on a specific application/system while performing penetration testing.

During an attack simulation, the target (Network, Software, Application, or Human) is attacked to meet the assessment goals. The simulation starts with goal setting, reconnaissance, and attacking servers and services to find the spots in the network. Through social engineering, phishing simulation, and data exfiltration testing, the network is attacked to identify the breaches. The organization obtains a broad overview of the attack surface and security posture.

Simulate the attack by taking a small input or change to the network to know the following:

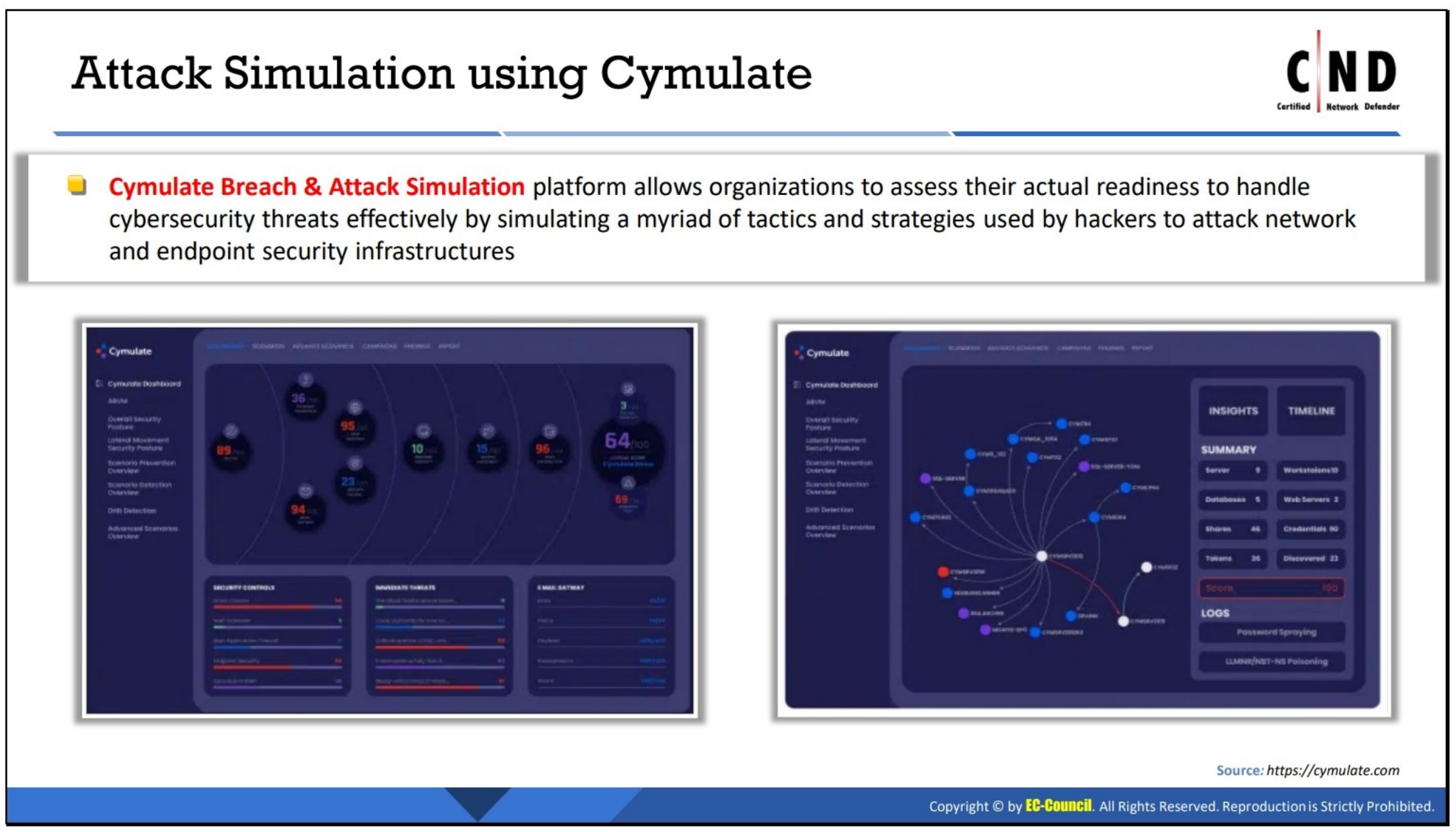
- How can any of the vulnerable exposures become exploits?
- What happens if the asset it moved?
- What happens if the topology and routing rules are changed?
- What happens if a policy is added or removed?
- What would be the result if an attack came in particular way?



Attack Simulation using Infection Monkey

Source: www.guardicore.com

Infection Monkey is an open-source BAS tool that tests and evaluates the strength of a network security configuration. Infection Monkey simulates a breach by infecting any random server within Cloud or on-premises infrastructure. Next, it runs around the network through different methods to enter the propagation paths and to attack every identified vulnerability point.



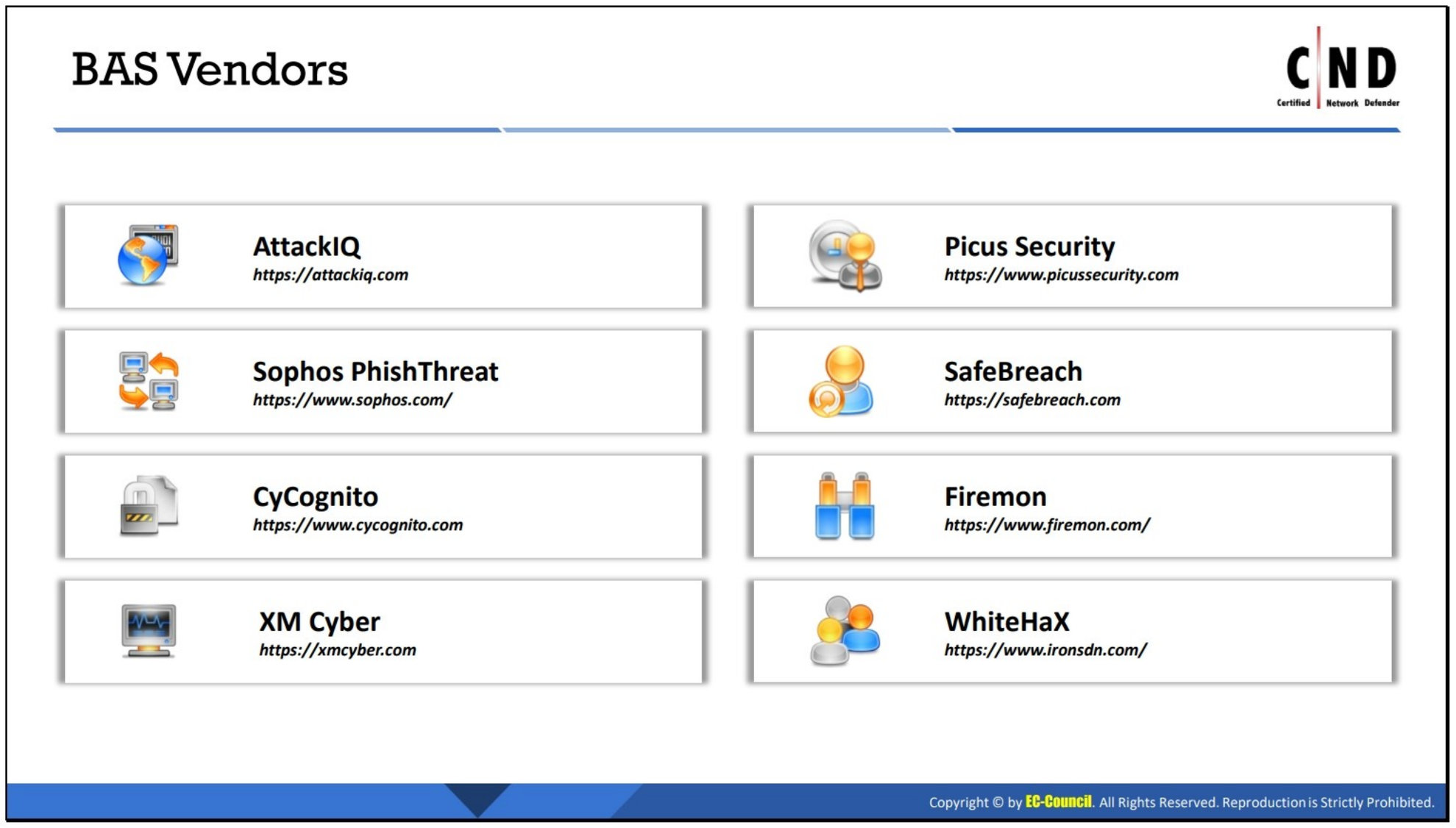
Attack Simulation using Cymulate

Source: https://cymulate.com

The Cymulate Breach & Attack Simulation tool enables organizations to handle cybersecurity threats by simulating the different strategies that hackers use to attack network and endpoint security infrastructures. It identifies security gaps automatically in one click and describes how to fix them exactly.

With Cymulate, you can perform the following:

- Exercise the organization's defenses against a wide range of attack vectors.
- Provide an Advanced Persistent Threat (APT) simulation of security posture at all times.
- Test the network's ability to cope with pre-exploitation-stage threats in email, web-gateway, and web applications.
- Analyze the ability to respond to real incidents with post-exploitation modules such as Lateral movement, Endpoint, and Data Exfiltration.
- Assess and improve the awareness of employees about phishing, ransomware, and other attacks.



BAS Vendors

BAS (Breach and Attack Simulation) technology automatically identifies vulnerabilities in an organization's cyber defence and offers continuous automated pen testing.

The following are some of the additional BAS tools that can be helpful in an attack simulation.

Attack IQ

Source: https://www.attackiq.com/

Attack IQ tool tests the cyber security programs against real-world threats, optimizes for efficiency and effectiveness and ensures the environment is prepared for future attacks. It ensures cybersecurity readiness and researches, develops, and emulates advanced cybersecurity concepts in disastrous situations to achieve cybersecurity readiness. It also validates the user's program's effectiveness at every stage of the attacker campaign and gives users the frequency, scale and scope needed to validate the entire security program.

Key Features:

- Endpoint detection and response
- Next Generation Firewalls
- Micro-segmentation
- Web application firewall
- Data Loss Prevention

Sophos PhishThreat

Source: *https://www.sophos.com/en-us*

Sophos PhishThreat educates and tests the end users through automated attack simulations and quality security training awareness. This provides flexibility and customization to the organizations to have a positive security environment and combines training and testing to make easy-to-use campaigns which provide automated on-the-spot training. It gives numerous realistic and challenging phishing attacks in less processes and helps users to understand the health of an organization.

Key Features:

- Reduces larger attack surfaces.
- Comprehensive reports
- Quality security awareness training
- Monitoring data points
- Available in 9 languages

CyCognito:

Source: *https://www.cycognito.com/*

Cycognito is an external attack surface management platform created to assist businesses in assessing their entire attack surface and drastically reducing risk. It automates the risk prioritization for external assets to emphasize the most critical exploited assets which are vital. It instantly spots the exploits to keep the attacker away from the environment and enables visibility into the attack to protect assets and surfaces. It can define risks, automate penetration testing operations, and keep the environment immune for future attacks.

Key Features:

- Graph business and asset relationship
- Risk prioritization
- Integration for communication
- Uses most common security frameworks.

XM Cyber:

Source: *https://xmcyber.com*

XM Cyber tool shows how attacker leverages the misconfigurations, and vulnerabilities across the cloud to compromise users' critical assets. This is designed to address the hybrid cloud exposures to proactively uncover the attacker paths and security flaws on the cloud. This helps to pinpoint the attacker's position and cut off the risk caused by attack paths to reduce the attack surface area. This emphasizes the backtracking of exploitation.

Key Features:

- Cloud security
- Attack path management

- Automated red teaming
- SOC optimization
- Continuous control monitoring
- Vulnerability Prioritization
- Cyber Risk reporting

Picus Security:

Source: *https://www.picussecurity.com/*

Picus Security tool helps to measure and strengthen cyber solutions and resilience by automatically testing the effectiveness of cyber detection tools. This finds and fixes the unwanted exposures that put critical assets at risk. This is a complete security control validation platform which helps to identify the logging and alert gaps which need additional action to optimize the SIEM in the organization's environments.

Key Features:

- Security control validation
- Mitigation Library
- Attack path validation
- Continuous assessment
- Compliance enablement
- Detection Rule validation

SafeBreach:

Source: *https://www.safebreach.com*

Safe Breach is a tool that continuously monitors and validates all layers of security by simulating real-world attacks. It unlocks visibility into security controls. Administrators can take immediate action because of the visibility they have against potential vulnerabilities. It increases the organisation's security control effectiveness, and real threat emulation and helps to improve cloud security. It helps you to visualize reports to have a better understanding of the status of the environment.

Key Features

- Threat assessment
- Security control validation
- Cloud security assessment
- Risk-based vulnerability management
- Flexible to IT/OT Environments

Firemon

Source: *https://www.firemon.com/*

FireMon addresses change detection and reporting, compliance, and behavioral analysis. Find it's vulnerability management technology under Security Manager and Cloud Defense, offering real-time risk assessment, mitigation and validation. It's attack path graphics and analysis enable administrators greater visibility.

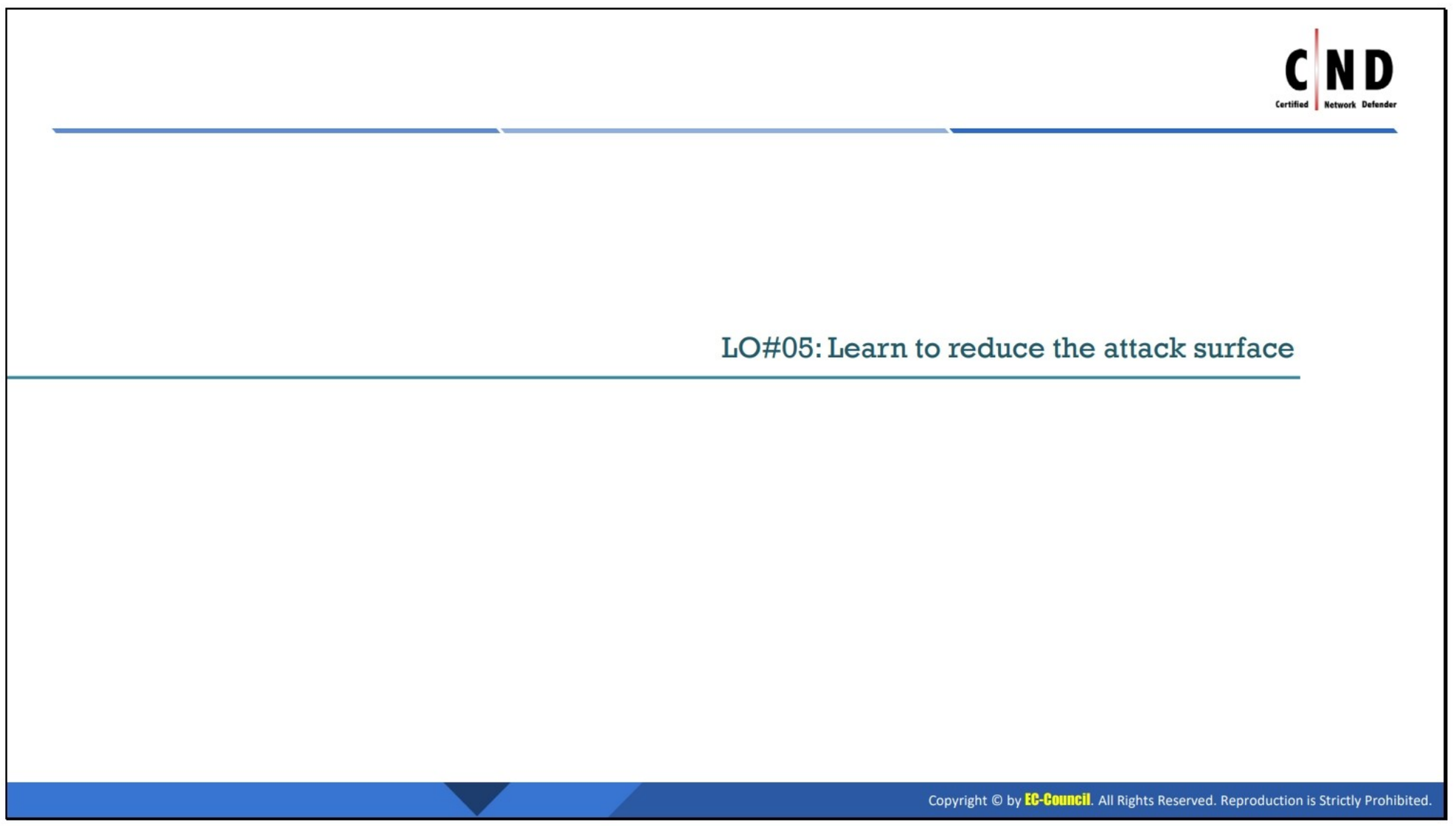
WhiteHaX

Source: *https://www.ironsdn.com/*

WhiteHaX is a cloud-hosted automated, cyber security readiness verification pen-testing platform. This provides a cyber insurance version of business readiness by enabling numerous attack simulations against security architecture that is deployed in business environments, perimeter firewalls and security and controls. It helps to protect and safeguard compliance and policyholders from cyber threats. This protects from phishing, ransomware attacks and malware.

Key Features

- Deep analysis of security and data privacy
- Wi-fi security scan
- AES-256 secured
- Cloud-based VPN
- Cross-platform password vault



LO#05: Reduce the Attack Surface

This section will help you understand the best practices to reduce the system attack surface, application attack surface, network attack surface, human attack surface, and physical attack surface.

Reducing Attack Surface

- Apply **vulnerability patch** to the identified risk exposures
- **Retest** the **vulnerabilities** to analyze the effects of a given fix

Reducing System Attack Surface

- **Disable** or **uninstall** any unnecessary and unused services, applications, or components running on the system

Reducing Application Attack Surface

- **Eliminate** redundant and unnecessary functionalities, entry points, Application Program Interfaces (APIs), code, and unnecessary complexity within an application's architecture

Reducing Attack Surface (Cont'd)

Reducing Network Attack Surface

Analyze the SSL certificates

- An **invalid** or **expired** SSL certificate can become an attack surface for the organization
- Analyze all valid, active, and expired SSL certificates
- Use tools such as **SurfaceBrowser** to analyze SSL certificates

Segment the network

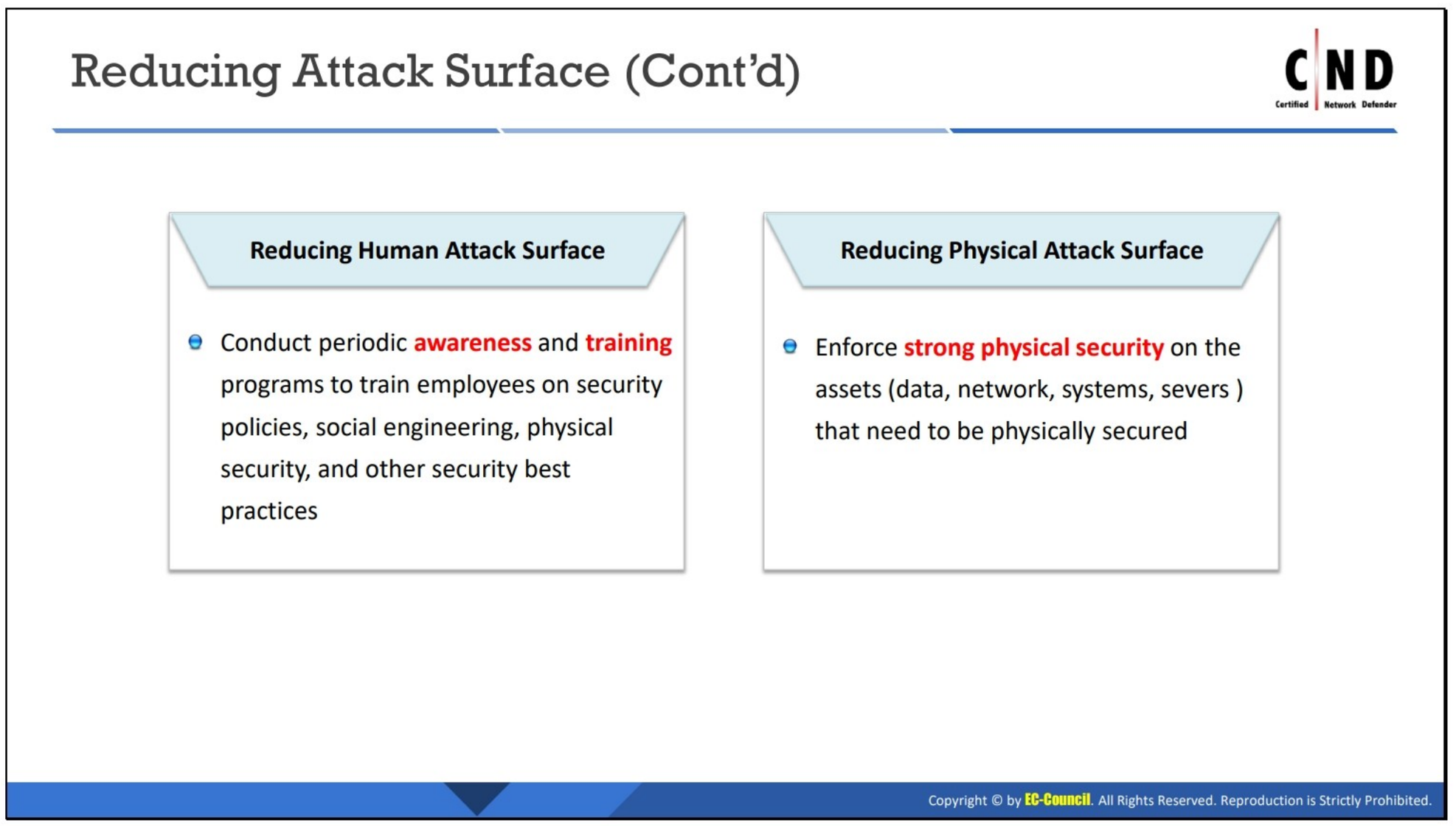
- A single **flat network** can increase the attack surface
- Segment the network into different segments
- A segmented network helps attain better network control by putting network barriers and access controls

Scan the network ports

- Keeping all ports open increases the network attack surface
- **Close** all unnecessary and unused ports on the public IP addresses
- Use tools such as nmap to identify the open ports on the public IP

Inspect the domain, IPs, and DNS zones

- Attackers can use publicly available information to craft an attack
- Publishing the IP address space and DNS information can increase the attack surface of the network
- Use a **DNS utility** such as DNS toolkit to review and check how much DNS information is being exposed to the Internet



Reducing Attack Surface

Reducing the attack surface reduces the likelihood of compromising the organization's assets. It involves activities that minimize the number of vulnerabilities in a system. This activity is called **Attack Surface Reduction (ASR)**. This process closes all but the needed doors that lead to system assets and restricts others with access rights and managing.

- **Reducing System Attack Surface/System ASR**

  System ASR includes preventing access to the systems in unintended ways. Systems include servers, desktops, laptops, and network devices.

  To ensure system ASR, do the following:

  - Ensure that **no processes or data resources are exposed**.

    Examples

    - **Processes** include browsers, mailers, and database servers.
    - **Data resources** include audit logs, files, directories, configuration files, registries, access rights, shares, virtual directories, and databases.
  - **Apply all available security patches** to the system.
  - **Shut down all channels and protocols** (FTP, TCP, UDP, HTTP, streaming, and RPC connections, among others).
  - **Secure system trust boundaries** (an extension of the attack surface and keep threat actions from reaching vulnerabilities).

- **Reducing Application Attack Surface/Application ASR**

  Application ASR involves the following:

  - Eliminating code redundancies and unnecessary complexity within an application's architecture, as the simplest code with least assumptions can avoid bigger attack surfaces.
  - Auditing and eliminating unnecessary functionality, entry points, and Application Program Interfaces (APIs).

  Example: Attack surface reduction of the web browser

  - Disable Firewall Traversal
  - Disable Network Prediction
  - Disable Sharing with Cloud Peripherals
  - Disable Google Data Synchronization
  - Disable Pop-ups
  - Disable 3D Graphic APIs
  - Disable JavaScript in all Available Locations
  - Disable Autocomplete on Forms Update Browser and Plugins Regularly
  - Disable Session Only Cookies
  - Disable Background Processing
  - Disable Search Suggestions
  - Disable Metrics Reporting
  - Disable Incognito Mode
  - Disable Cleartext Passwords
  - Disable Password Manager
  - Disable Import of Saved Passwords
  - Disable Outdated Plugins User Permission to Run Plugins
  - Disable Automatic Plugin Search
  - Disable Automatic Plugin Installation
  - Disable Automatic Plugin Execution
  - Enable Revocation Checks for Certificates
  - Enable Safe Browsing
  - Block Desktop Notifications
  - Block Third-Party Cookies

- Set the Default Search Provider Name
- Set Home Page
- Set Highest HTTP Authentication Scheme
- Blacklist/Whitelist Plugins and Extensions
- Limit Plugins to a Specific URL
- Use Encrypted Searching
- Disallow Location Tracking
- Save Browser History
- Chose the web browser from one of the most used browsers: Chrome, Opera, IE and Firefox.
- Research the attack methodologies: Define Attack Categories
  - Attacks against users
  - Attacks against browser
  - Attacks against extensions
  - Attacks against web applications
  - Attacks against plugins
  - Attacks against network

Plugins	Attacking ActiveX Controls, ActiveX Web Application Sending Cross-origin Requests, Enumerating Cross-origin Quirks, Preflight Requests, Implications, Cross-origin Web Application Detection, Discovering Intranet Device IP Addresses, Enumerating.	ActiveX
Web Application	Internal Domain Names, Requesting Known Resources, Cross-origin Authentication Detection, Cross-site Request Forgery, Attacking Password Reset with XSRF, Using CSRF Tokens for Protection, Cross-origin Resource Detection, Cross-origin Web Application Vulnerability Detection	Bypass Same Origin Policy
User	Signed Java Applet, Bypass Anonymization	Java
Plugins	Attacking Java	
Network	Ping Sweeping using Java, Getting Shells	
User	Change Page Content, Capture User Input, Log Where User Clicks, Log Mouse Events, Log Form Events, Log	

	Keyboard Shortcuts, Tabnabbing, Phishing, Fake Software Update, Bypass Anonymization, Hack Password Managers	
Browser	Bypassing Path Attribute Restrictions, Sidejacking Attacks, Attack JavaScript, JavaScript Encryption, Java Heap, Abusing Schemes	
Extensions	Exploring Privileges, Attacking Extensions, Impersona.ng Extensions, Cross-context Scrip.ng, Achieving OS Command Execution, Achieving OS Command Injection	
Plugins	Attacking Plugins, Bypassing Click to Play	
Network	Identifying the Hooked Browser's Internal IP, Identifying the Hooked Browser's Subnet, Ping Sweeping, Port Scanning, Bypassing Port Banning, Distributed Port Scanning, Fingerprin.ng Non-HTTP Services, Attacking Non-HTTP Services, NAT Pinning, Achieving Inter-protocol Communication, Achieving Inter-protocol Exploitation	JavaScript

Table 19.2: Attack Categories

- **Create a refined list of attack vectors** that are used by the above attack methodologies.

- **Reducing Network Attack Surface/Network ASR**

  Perform Network ASR through the following:

  - **Analyze Secure Sockets Layer (SSL) certificates**

    Ensure whether the SSL certificates are hardened, and SSL chains are secured and use strong cipher suites. Check the SSL certificate expiration and validity. Use one of the top SSL testing websites Qualys SSL Test to know how secure the SSL is. Use tools such as SurfaceBrowser to analyze SSL certificates.

  - **Segment network**

    Segment the organization's network into different segments because keeping all assets within a single network can increase the attack surface. This method helps attain better network control by putting network barriers and effective server or desktop access controls over the machines connected to the network. It also reduces the dwell time (the time cyberattacks spend undetected on networks) by placing "quick sand" in an attacker's path to stop them in their tracks.

  - **Scan network ports**

    Close unnecessary and unused ports on the public IP addresses since keeping all ports opened increases network surface. To identify the open ports in a public IP, perform an

audit of network ports before the attacker scans network ports by using the tool Nmap. Other port scanners include Unicornscan, Angry IP Scanner, and Netcat.

- **Inspect domain, IPs, and Domain Name System (DNS) zones**

  Publishing IP address space and DNS information can increase the attack surface of the network. Use DNS utility such as DNS toolkit to review and check how much DNS information is exposed to the Internet.

- **Reducing Human Attack Surface/Human ASR**

  Security awareness training plays a crucial role in reducing the human attack surface. Employee training helps employees comply with the organization's security policies. Social engineering awareness enhances data confidentiality, data integrity, and data availability by preventing various phishing attacks and other social engineering attacks. The awareness programs developed by the organizations should cover the following:

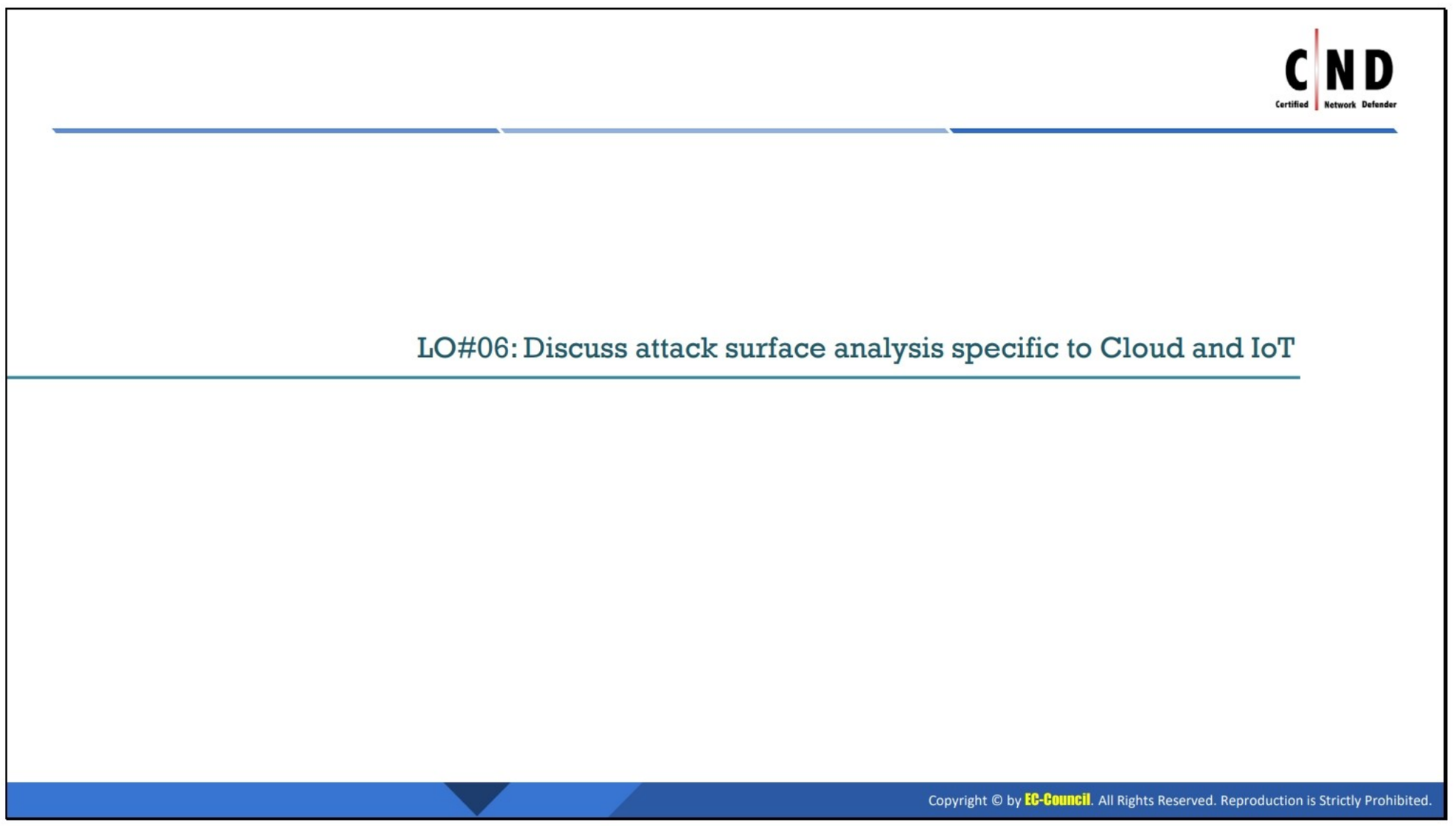
  - What is information security?
  - Why is information security necessary?
  - Where can the organization's security policies be found?
  - How can the security of sensitive information assets be ensured?
  - What are the regulations that apply to the business of the organization?
  - What are the potential effects of a security incident on the organization?

- **Reducing Physical Attack Surface/Physical ASR**

  The physical safety of device is of utmost importance, as unauthorized physical access to devices can defeat the purpose of implementing various technical security controls.

  Ensure the following for physical ASR as the human factor is considered the weakest link in cybersecurity.

  - Enforce strong physical security for the assets (data, network, systems, severs).
  - Implement appropriate physical security measures.

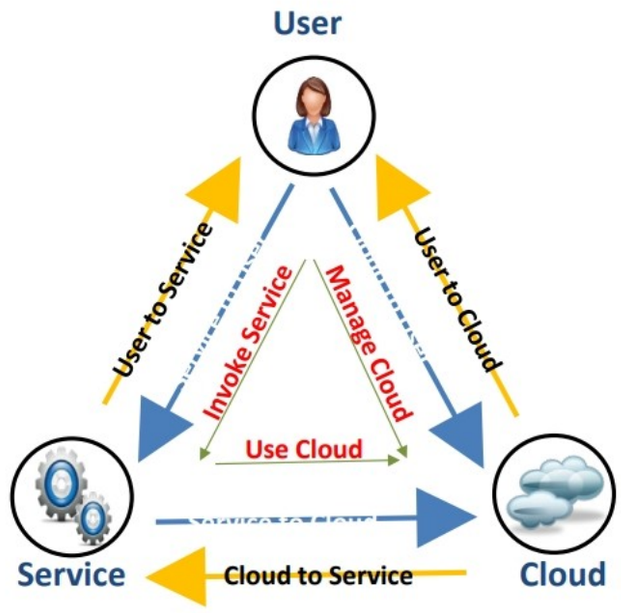


LO#06: Attack Surface Analysis Specific to Cloud and IoT

Increased use of Cloud and Internet of Things (IoT) results in expanding the organization's attack surface. The more the Cloud is used and the more IoT is connected to the organization's network, the greater the number of endpoints that need to be protected. Understanding the attack surface of Cloud and IoT will help a network defender secure the network supported with Cloud and IoT. This section will help you understand the Cloud and IoT attack surfaces and their vulnerabilities.

Cloud Attack Surface

Attack Surface	Description	Examples of Attacks
Service to User Service interface exposed to User	Server to client interface. It covers all attacks possible in a client-server architecture.	Buffer overflow, SQL Injection, and Privilege Escalation
User to Service User exposed to Service	A common interface provided by the client application to the server that covers browser-based application attacks.	SSL Certificate spoofing, Phishing
Cloud to Service Cloud resources or interfaces exposed to Service	Cloud resources interacting with the services. Covers attacks run by services on Cloud infrastructure.	Resource exhaustion, DoS
Service to Cloud Service interface exposed to Cloud	Service interface exposed to the Cloud. Covers attacks performed by the service provider against a service running on their infrastructure.	Privacy attacks, Data integrity attacks, Data confidentiality attacks
Cloud to User Cloud interface exposed to User	Cloud interface exposed to the user to manage Cloud services. Covers attacks to the services from the users.	Attacks on Cloud control
User to Cloud User exposed to Cloud	User exposed to the Cloud. Covers attacks targeting the user and originating at the Cloud such as manipulating Cloud-provided services by phishing attempts.	Presenting the user a faked usage bill of the Cloud provider



Cloud Attack Surface

Three participant classes are used in modelling a Cloud attack surface, namely, Service users, Service instances or Services, and Cloud provider. The interactions in the Cloud solution are addressed by at least two entities of the participant classes. For example, a service requested by a user or more CPU power inquired from the infrastructure system by a service instance. Therefore, the attack attempts in the Cloud solution are also treated as interactions within the three-participant class-model of the Cloud solution. For example, same attack vectors (Structured Query Language (SQL) injection, Denial of Service (DoS), Cross-site scripting (XSS), and flooding attacks) exist between a user and service instance and outside the Cloud solution. The security of the Cloud solution here involves discussing the attacks with the Cloud provider among the participant classes. The Cloud provider itself need not be malicious. It may merely play an intermediate role in an ongoing combined attack. Every participant class provides an interface to the other participant class. With three participant classes, six interfaces are possible, and these are treated as the attack surfaces in a Cloud computing scenario.

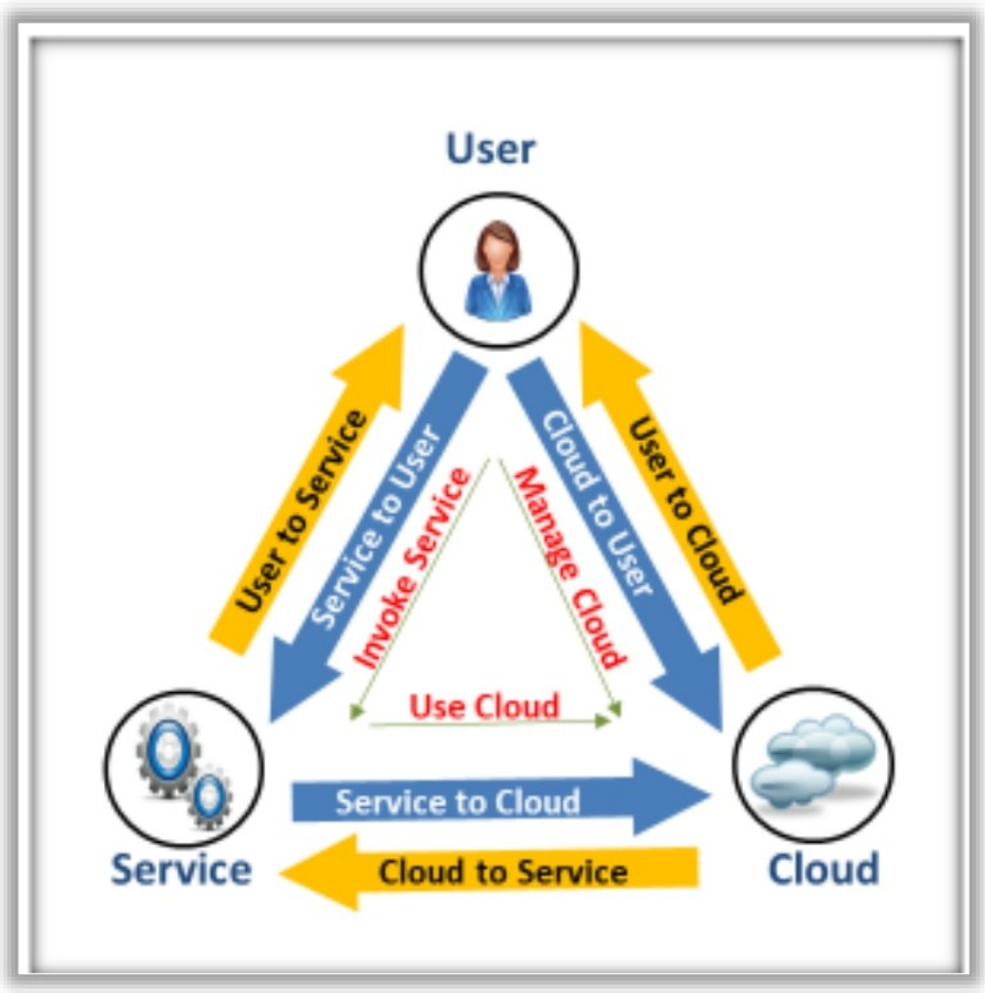


Figure 19.9: Cloud Computing Attack Surfaces

Service to User

It enables all types of attacks that affect the client-server architecture or the service interface exposed towards clients. This is the most important attack surface of a Cloud solution. Common attacks in client-server architecture are buffer overflow attacks, SQL injection, and privilege escalation, among others.

User to Service

It is the attack surface that the client program (User service) provides towards the service (server). Common attacks to this surface affect are browser-based applications, attacks on browser caches, and phishing attacks on email client, among others.

Cloud to Service

It is related to exposing Cloud resources/interfaces to service instances. The interface between a service and the Cloud is complex, and separating the service and Cloud is slightly tricky. This is because the Cloud's attack surface to the service covers the service instance's attacks against its Cloud host solution.

For example, resource exhaustion, triggering the Cloud provider to provide more resources or end up in a Denial-of-Service (DoS), and attacks on the Cloud system hypervisor.

Service to Cloud

It is related to exposing the service instance to the Cloud provider. The Cloud provider performs all types of attacks on a service running on it. This is the most critical attack surface as it is easy to exploit and has a high attack impact.

Examples include the following:

- Availability reductions (shutdown service instances)

- Privacy attacks (scanning a service's data in process)
- Malicious interference (tampering data in process, injecting addition operations to service execution)
- Data integrity attack
- Data confidentiality attack

Cloud to User

A service exists between Cloud provider and the user that pertains to Cloud control (adding new services or requiring more service instances that are in use and deleting service instances, among others). This makes it difficult to define this attack surface. This attack surface refers to the attacks that a Cloud service faces from a user's point of view.

User to Cloud

This pertains to the different types of attack vectors that target a user. It has its origins in the Cloud system. For example, phishing-like attempts that present users a fake usage bill of the Cloud provider.

Recommendations for Reducing Cloud Attack Surface

- Identify and map all assets across the Cloud.
- Map both internal and external network infrastructures to obtain a single view of the organization's network.
- Identify and understand the vulnerabilities, misconfigurations, and threats to all assets.
- Implement security controls to address the security of the local network and Cloud infrastructure.
- Protect every endpoint device.
- Ensure the security of all data repositories.
- Understand the internal access control and security contract of the service provider before signing Service Level Agreement.

Attack Surface of IoT

- The attack surface of IoT is the combination of potential security **vulnerabilities** or **threats** associated with the IoT and its applications and devices on which attacks can be initiated

Attack Surface Areas of IoT

- Ecosystem Access Control
- Device Memory
- Device Physical Interfaces
- Device Web Interface
- Device Firmware
- Device Network Services
- Administrative Interface
- Local Data Storage
- Cloud Web Interface
- Ecosystem Communication
- Vendor Backend (APIs)
- Third-Party Backend APIs
- Update Mechanism
- Mobile Application
- Vendor Backend APIs
- Network Traffic

Source: *https://www.owasp.org*

Attack Surface of IoT (Cont'd)

Attack Surface	Vulnerabilities
Ecosystem Access Control	Authentication Session management Implicit trust between components Enrollment security Decommissioning system Lost access procedures
Device Memory	Cleartext username Cleartext passwords Third-party credentials Encryption keys
Device Physical Interfaces	Firmware extraction User Command-Line Interface (CLI) Admin CLI Privilege escalation Reset to insecure state

Attack Surface	Vulnerabilities
Device Web Interface	SQL injection Cross-site scripting Username enumeration Weak passwords Account lockout Known credentials
Device Network Services	Information disclosure User CLI Admin CLI Injection DoS
Administrative Interface	SQL injection Cross-site scripting Username enumeration Weak passwords Account lockout Known credentials

Source: *https://www.owasp.org*

Attack Surface of IoT (Cont'd)

Attack Surface	Vulnerabilities
Local Data Storage	Unencrypted data Data encrypted with discovered keys Lack of data integrity checks
Cloud Web Interface	SQL injection Cross-site scripting Username enumeration Weak passwords Account lockout Known credentials
Third-party Backend APIs	Unencrypted Personally Identifiable Information (PII) sent Encrypted PII sent Device information leaked Location leaked
Update Mechanism	Update sent without encryption Updates not signed Update location writable

Attack Surface	Vulnerabilities
Mobile Application	Implicitly trusted by device or cloud Known credentials Insecure data storage Lack of transport encryption
Vendor Backend APIs	Inherent trust of Cloud or mobile application Weak authentication Weak access control Injection attacks
Ecosystem Communication	Health checks Heartbeats Ecosystem commands Deprovisioning Update pushes
Network Traffic	LAN LAN to Internet Short range Non-standard

Source: *https://www.owasp.org*

Attack Surface of IoT

The attack surface of IoT is the combination of potential security vulnerabilities or threats associated with the IoT and its applications and devices on which the attacks can be initiated.

An IoT system usually comprises four components, and each of these have different attack surfaces.

- **Devices:** Attacks can be triggered through devices. They can also be the primary targets. The parts of a device from which vulnerabilities emerge are physical interfaces (USB ports), failures in memory, firmware, web interface, admin interfaces, and network services. The other aspects of devices include their unsecured settings and outdated devices/components.
- **Communication Channels:** Attacks can occur due to the ways in which IoT components connect with each other. These attacks can be network attacks (DoS), or spoofing, among others. For example, protocols used in IoT systems, Bluetooth, and Wi-Fi can be exposed to vulnerabilities.
- **Applications and Software:** Attacks can come from vulnerabilities in application interfaces and the software of IoT devices. These can compromise systems, steal sensitive or personal data (credentials), or insert malicious firmware updates.
- **Cloud Interfaces**: Attacks are triggered by inadequate passwords, default credentials, and insecure transport encryption in using Cloud interfaces for IoT.

Key IoT Attack Surface Areas and Their Vulnerabilities

- **Ecosystem Access Control** refers to access control mechanisms, enrollment, and decommissioning procedures. Example vulnerabilities for this attack surface include the following:
  - Authentication
  - Session management
  - Implicit trust between components
  - Enrolment security
  - Decommissioning system
  - Lost access procedures
- **Device Memory** is the possibility of possessing clear-text credentials stored in memory and the monitoring of cipher keys. Example vulnerabilities associated with this attack surface are as follows:
  - **Clear-text credentials** and **third-party credentials** may lead to leakage of credentials information and compromising platform, resulting in device compromise.
  - Access to **Encryption Keys** may lead to information decryption.
- Unsecured elements in **Device Physical Interfaces** are used to compromise IoT devices. Example attack vectors for this attack surface are as follows:
  - **Firmware extraction** may lead to exposing hidden vulnerabilities in the firmware.
  - Access to **Console Access (User** Command-Line Interface **(CLI)/Admin CLI)** to access a device to obtain its data or to access the device with the administrative rights may expose to exploit data leak or compromise the device itself.
  - **Privilege escalation** in which physical access to the device unless it is configured with granular level access may lead to exposing the function of the device.
  - **Reset to insecure state** may reset the data of the device (memory or storage) to an unsecured state in the case of physical access.
  - **Removal of storage media** may lead to access to the storage media, exposing firmware, control keys, and local data information.
- **Device Web Interface** comprises the web application vulnerabilities of the IoT device web interface and credential management. Example vectors for device web interface attack surface are as follows:
  - SQL injection
  - Cross-site scripting
  - Cross-site request forgery
  - Username enumeration

- Weak passwords
- Account lockout
- Known default credentials

- **Device Firmware**

  The device firmware provides the features and functions for a solution. It handles domain and sector-specific data and logic processing. The vulnerabilities that are common to the attack surface of the device firmware are described below:

  - Many IoT devices comprise the **Hardcoded credentials/Default credentials** provided by the manufacturer. They are never reset by the consumer. Using Botnets can exploit the default credentials, compromising the device.
  - **Sensitive information disclosure** (data/control keys) may lead to devices being compromised.
  - Access to firmware leads to additional vulnerabilities such as **firmware version display and/or last update date**.

- **Device Network Services**

  Device Network Services involve physical devices, OS/Firmware, and data stored on the device side. The possible attack vectors for the device network services attack surface area are as follows:

  - Injection
  - DoS
  - Man-in-the-Middle attacks
  - Buffer overflow

- **Administrative Interface** comprises all attack vectors associated with the system's administrative web interface. Example vectors for the administrative interface attack surface are as follows:

  - SQL injection
  - Cross-site scripting
  - Username enumeration
  - Weak passwords
  - Account lockout
  - Known credentials

- **Local Data Storage**

  Unsecured local data storage includes vulnerabilities such as unencrypted sensitive data in the local storage, encrypted local storage data with discovered keys, or data stored in local storage without integrity checks.

- **Cloud Web Interface** comprises the standard web vulnerabilities, credential management, transport encryption, and lack of two-factor authentication. These are related to IoT Cloud components (applications and web services). Example attack vectors for the Cloud web interface are as follows:
  - SQL injection
  - Cross-site scripting
  - Username enumeration
  - Weak passwords
  - Account lockout
  - Known credentials
- **Unsecured Third-party Backend APIs** of IoT applications expose the user's data and places the applications and IoT devices at risk. Example attack vectors for this attack surface are as follows:
  - Unencrypted Personally Identifiable Information (PII) sent
  - Encrypted PII sent
  - Device information leaked
  - Location leaked
- **Update Mechanism** comprises vulnerabilities such as lack of encryption or signature, writable locations, and missing update mechanisms. Example attack vectors for the Update Mechanism attack surface are as follows:
  - Update sent without encryption
  - Updates not signed
  - Update location writable
- Attackers use vulnerabilities (user enumeration, weak passwords, lack of encryption, or account lockout) in a **Mobile Application** that is connected to an IoT device as attack vectors. Example attack vectors for the mobile application attack surface are as follows:
  - Implicitly trusted by device or Cloud
  - **Known credentials** may compromise the disk.
  - **Unsecure data storage** on a mobile device may lead to data leak.
  - Lack of transport encryption
- **Vendor Backend Application Program Interfaces (APIs)** are the attacks and vulnerabilities that may affect the vendor-provided APIs. Example vulnerabilities for this attack surface are as follows:
  - Inherent trust of Cloud or mobile application

- Weak authentication
- Weak access control
- Injection attacks

- **Ecosystem Communication**

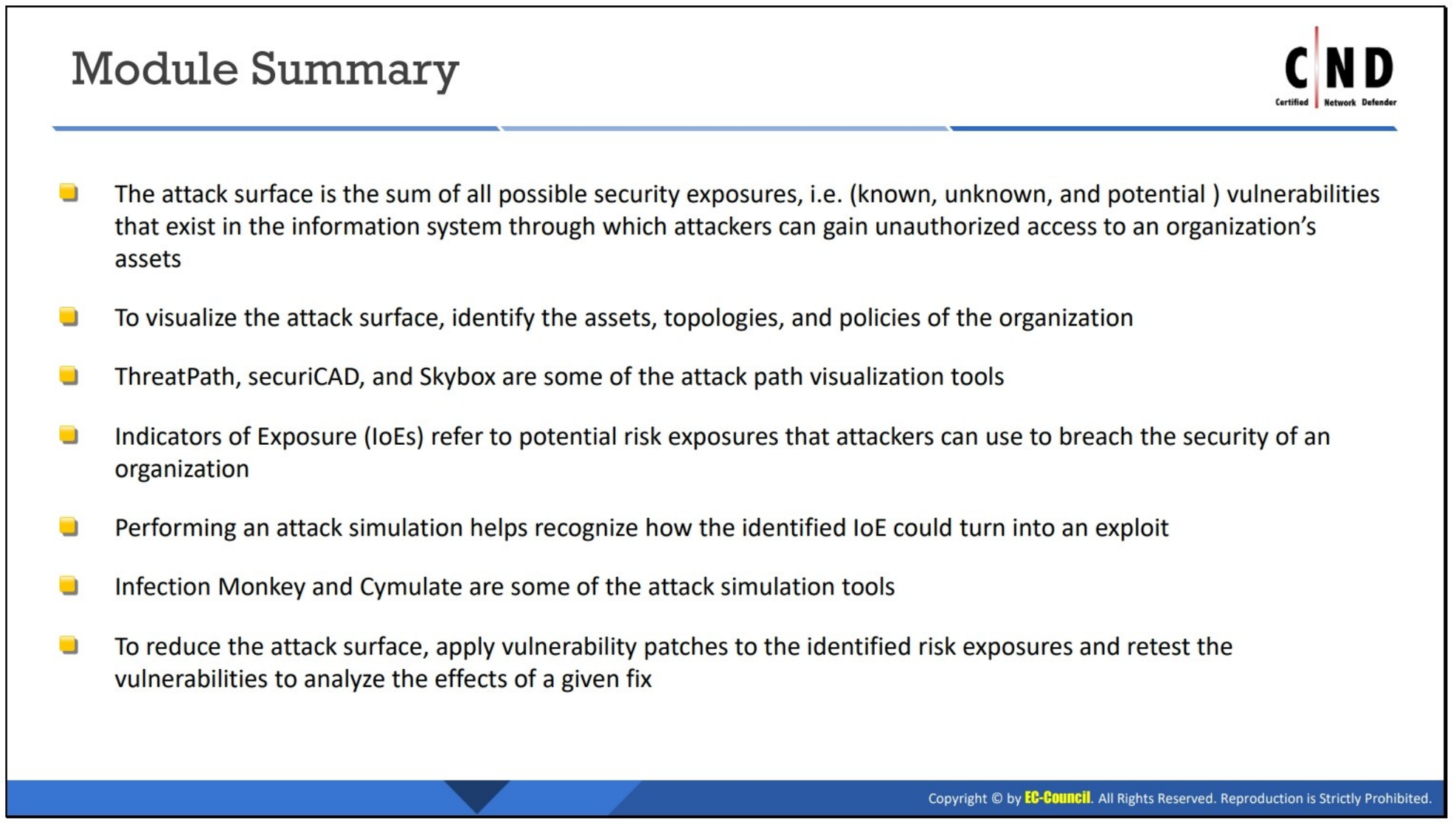
  Lack of communication between IoT components may leave the whole system compromised if one of the IoT components fails.

  - Health checks
  - Heartbeats
  - Ecosystem commands
  - Deprovisioning
  - Pushing updates

- **Network Traffic** comprises the security issues that are associated with the network and communication choices that are made during its design. Example vulnerabilities related to this attack surface are as follows:
  - Local Area Network (LAN)
  - LAN to Internet
  - Short range
  - Non-standard

Recommendations for reducing the IoT attack surface

- Perform detailed research on the product before purchasing and ensure that the manufacture has followed a Secure-by-Design approach and integrated security while manufacturing the product.
- Evaluate and understand the risks involved in using an IoT device before connecting it on to the network.
- Configure the device securely before connecting it to the network.
- Disable unnecessary features of the IoT device.
- Implement network segmentation, secure access and identity management, and secure remote access.
- Implement security controls to physically protect all IoT devices.
- Regularly monitor IoT devices for suspicious activities.



Module Summary

The following summarizes the key points discussed in this module:

- The attack surface is the sum of all possible security exposures, i.e. (known, unknown, and potential) vulnerabilities that exist in the information system through which attackers can gain unauthorized access to an organization's assets.
- To visualize the attack surface, identify the assets, topologies, and policies of the organization.
- ThreatPath, securiCAD, and Skybox are some of the attack path visualization tools.
- Indicators of Exposure (IoEs) refer to potential risk exposures that attackers can use to breach the security of an organization.
- Performing an attack simulation helps recognize how the identified IoE could turn into an exploit.
- Infection Monkey and Cymulate are some of the attack simulation tools.
- To reduce the attack surface, apply vulnerability patches to the identified risk exposures and retest the vulnerabilities to analyze the effects of a given fix.

This page is intentionally left blank.