EC-Council

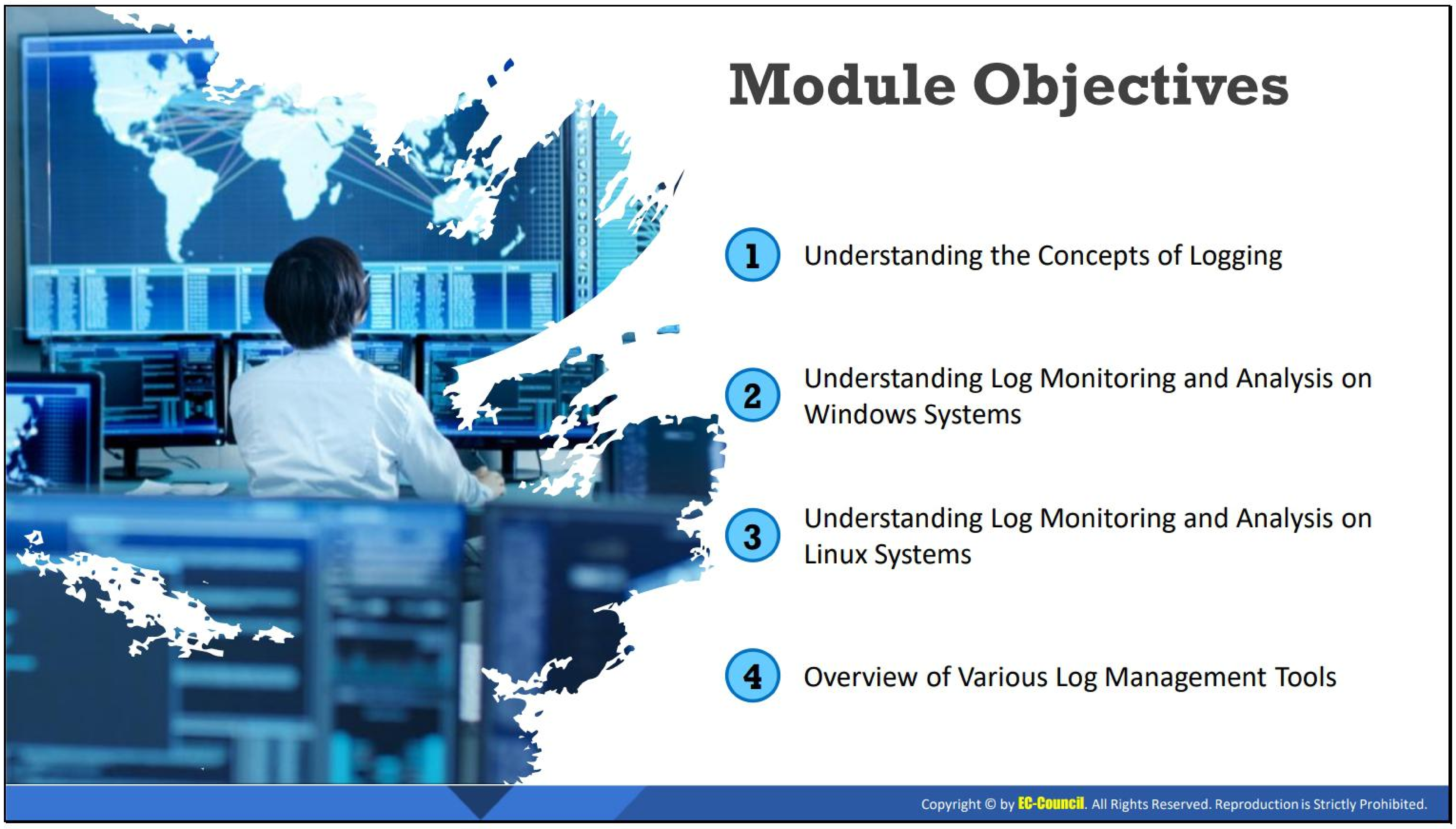
C|CT

Certified Cybersecurity Technician

Module - 18

Network Logs Monitoring and Analysis

This page is intentionally left blank.



Module Objectives

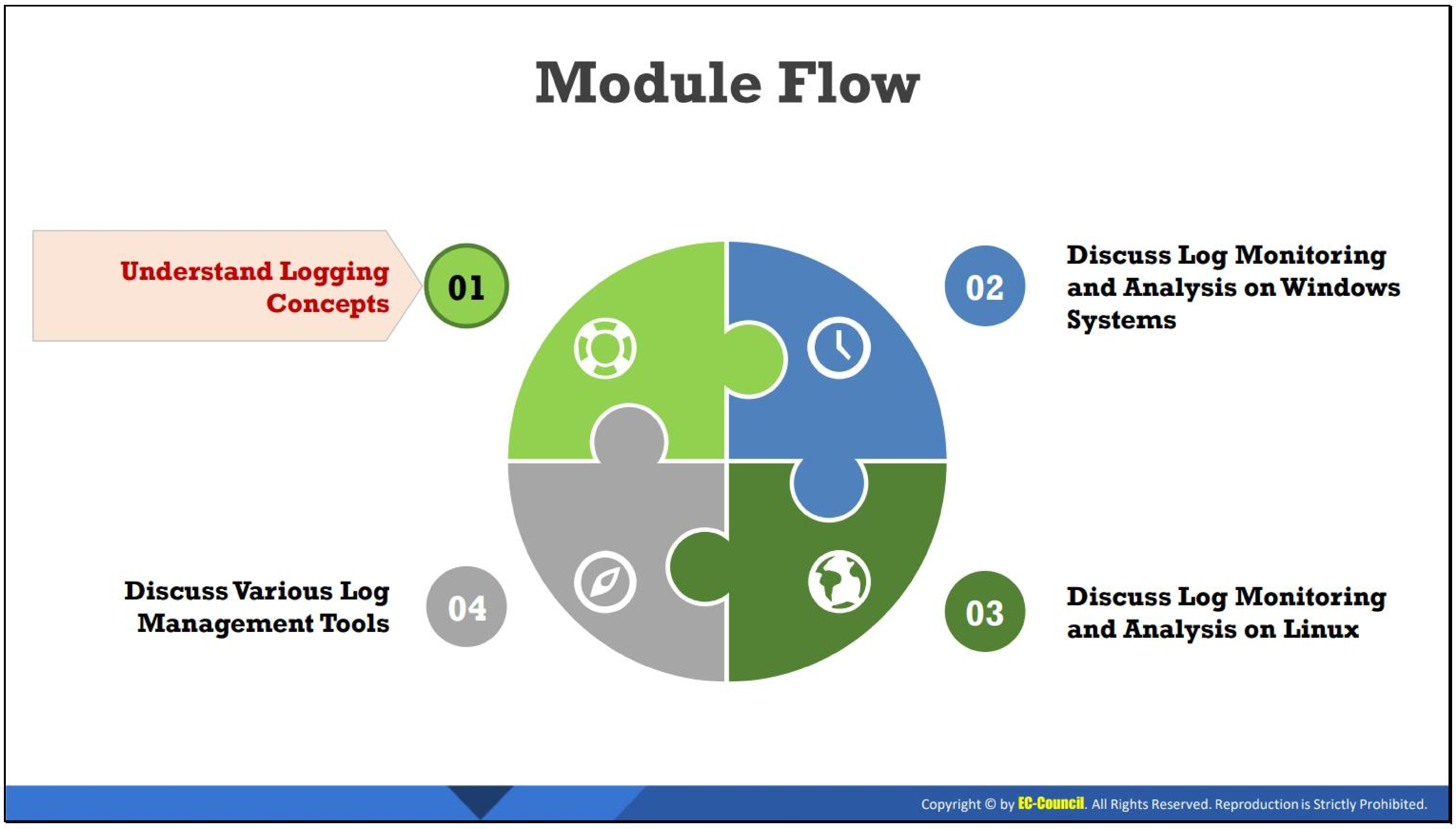
1. Understanding the Concepts of Logging
2. Understanding Log Monitoring and Analysis on Windows Systems
3. Understanding Log Monitoring and Analysis on Linux Systems
4. Overview of Various Log Management Tools

Module Objectives

To enhance the security of an organization, extensive monitoring and analysis of network logs is critical. This helps identify and respond to threats quickly and protect the network assets from various attacks. Proper network log monitoring and analysis help reduce the frequency of attacks by proactively responding to threats.

At the end of this module, you will be able to do the following:

- Understand the concepts of logging
- Understand log monitoring and analysis on Windows systems
- Understand log monitoring and analysis on Linux systems
- Use various log management tools



Module Flow

01 Understand Logging Concepts

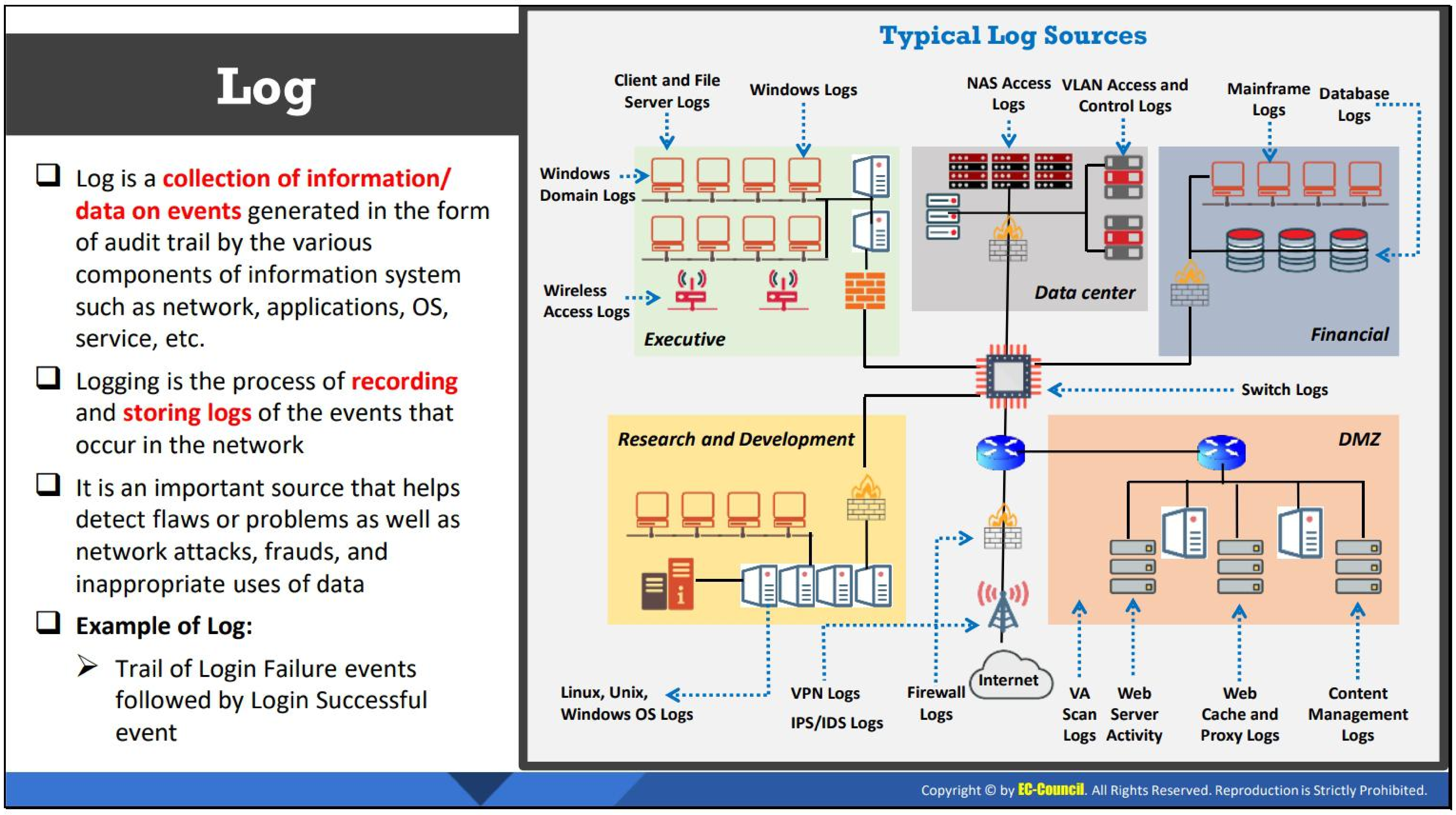
02 Discuss Log Monitoring and Analysis on Windows Systems

03 Discuss Log Monitoring and Analysis on Linux

04 Discuss Various Log Management Tools

Understand Logging Concepts

The objective of this section is to explain the basic concepts of logging. It describes the sources of logs, need of logging, log formats, and various logging approaches.



Log

- Log is a **collection of information/ data on events** generated in the form of audit trail by the various components of information system such as network, applications, OS, service, etc.
- Logging is the process of **recording** and **storing logs** of the events that occur in the network
- It is an important source that helps detect flaws or problems as well as network attacks, frauds, and inappropriate uses of data
- **Example of Log:**
  - Trail of Login Failure events followed by Login Successful event

Log

Logs are collection of information/data on events generated in the form of an audit trail by the various components of an information system such as network, applications, operating system (OS), service, etc. A log can provide an indication that something may have gone wrong and it helps security professionals in analyzing and detecting issues.

Separately, transaction logs or firewall logs or intrusion prevention system (IPS)/intrusion detection system (IDS) logs do not report faults. They simply store records of specific events; for example, deletion of record from the database. When logs from multiple devices are collected, correlated, and analyzed by security incident and event management (SIEM) systems, something meaningful is generated. For example, by combining the transaction log that represents a record entry by a user with the firewall log that represents network activity from an IP address registered by the same user who made the record entry, we can verify the authenticity of that user.

Logs are recorded and stored through the logging process. Generally, there are four types of logging: security logging, operational logging, compliance logging, and application debug logging. Security logging concentrates on identifying and responding to security-related activities such as threats, viruses, malware, data loss, etc. It records logs about user login, unauthorized access to resources, etc. Operational logging concentrates on system-processing activities. It informs the security professional regarding failures and potentially actionable conditions. It also facilitates service provisioning and financial decisions. Compliance logging is a part of security logging because regulations are developed to enhance the security of systems and data. Application debug logging is logging that is beneficial to application/system developers, not system administrators. It concentrates on recording debugging logs, which are analyzed by the application developer to detect issues. This type of logging can be disabled and enabled in a production system based on circumstantial requirements.

Typical Log Sources

A log source refers to a data source that builds an event log. Almost every device or application on the network has logging capability and can produce a log to record the information regarding an event. Every security system generates logs in some form or another. Windows logs, client and file server logs, router logs, firewall logs, and database logs are examples of the various log sources in the network.

Log sources use two mechanisms to transfer records: pull-based and push-based. In a push-based mechanism, the system or application either saves records on the local disk or sends them over the network. If the records are being sent over the network, then a log collector is needed to collect them. System Logging Protocol (syslog) and Simple Network Management Protocol (SNMP) are the two main push-based protocols through which log records are transferred. In a pull-based mechanism, a system or an application pulls the log records from a log source. It works based on the client–server model. The system or device that follows this mechanism usually stores their log data in a proprietary format. For example, Check Point provides OPSEC C library to pull logs from a Check Point device.

The required log sources need to be configured to collect important information in required formats and locations and store it for a long period of time. Log source configuration is not an easy task. Initially, the hosts and host components that are going to participate in log management infrastructure need to be identified based on the standard rules and policies. A single log file includes information from multiple sources; for example, an OS log includes information not only from OS itself but also from various other security programs. Once the log source is determined, the types of events to be logged by each log source as well as the features of data to be logged for each event need to be specified. The, the log sources need to be configured based on the features provided by that particular type of log source. Some log sources provide granular configuration options while others provide no granularity at all. In log sources with no granularity, logging is either enabled or disabled, without any control over the kind of data that can be logged.

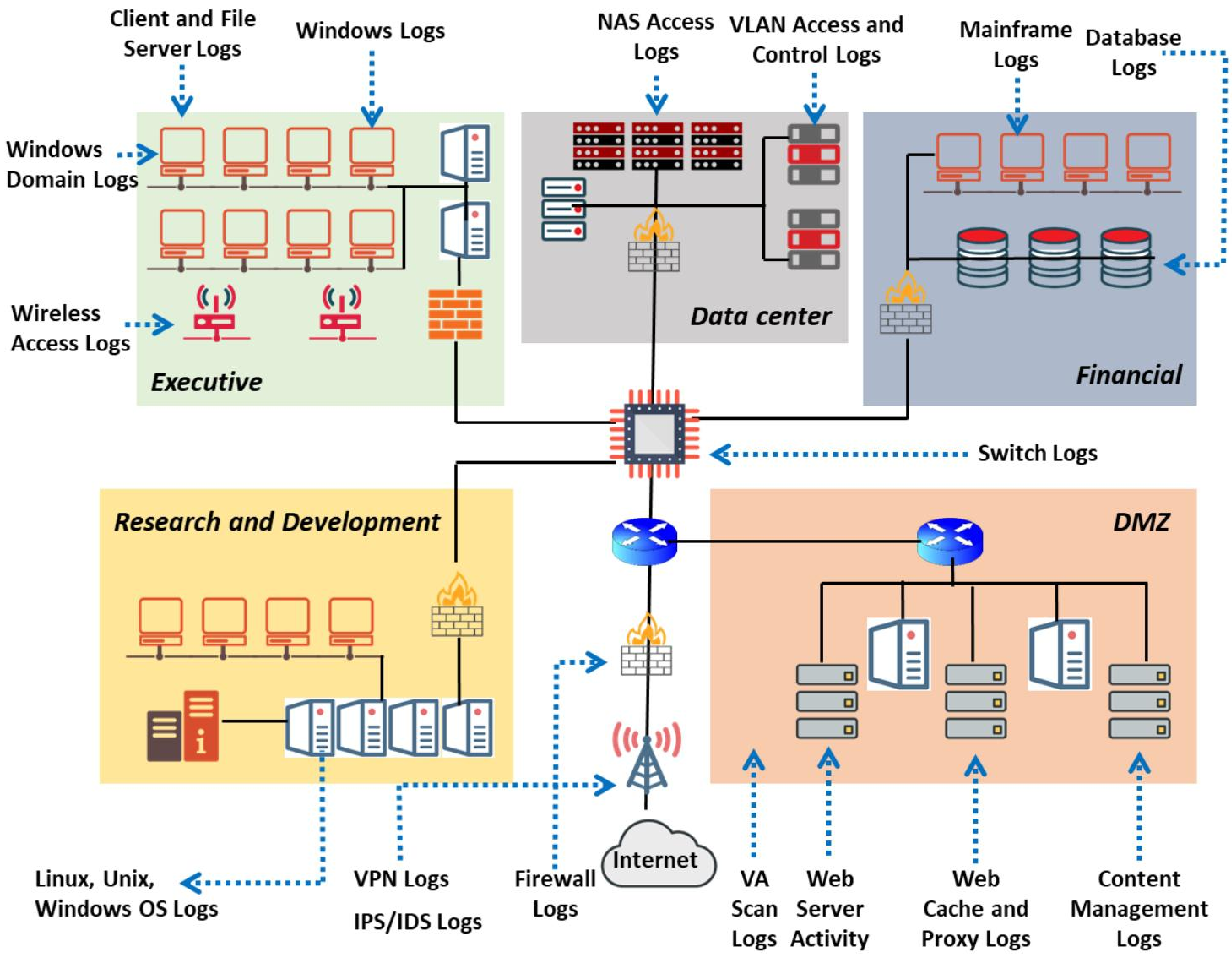
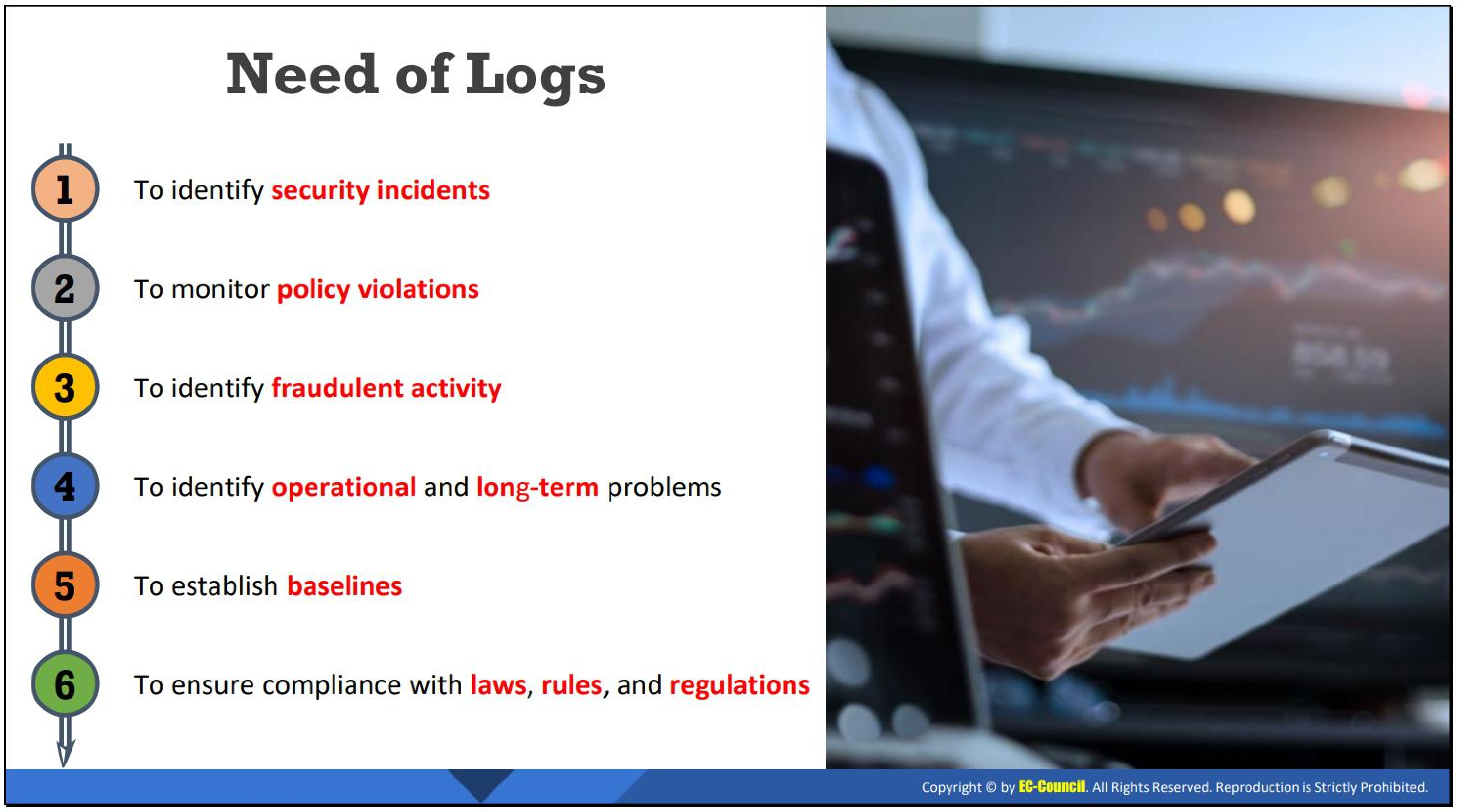


Figure 18.1: Typical log sources



Need of Logs

1. To identify **security incidents**
2. To monitor **policy violations**
3. To identify **fraudulent activity**
4. To identify **operational** and **long-term** problems
5. To establish **baselines**
6. To ensure compliance with **laws**, **rules**, and **regulations**

Need of Logs

Logs are needed to understand the various events occurring on the network and are used for various purposes—from performance to threat identification.

Typical use of logs includes:

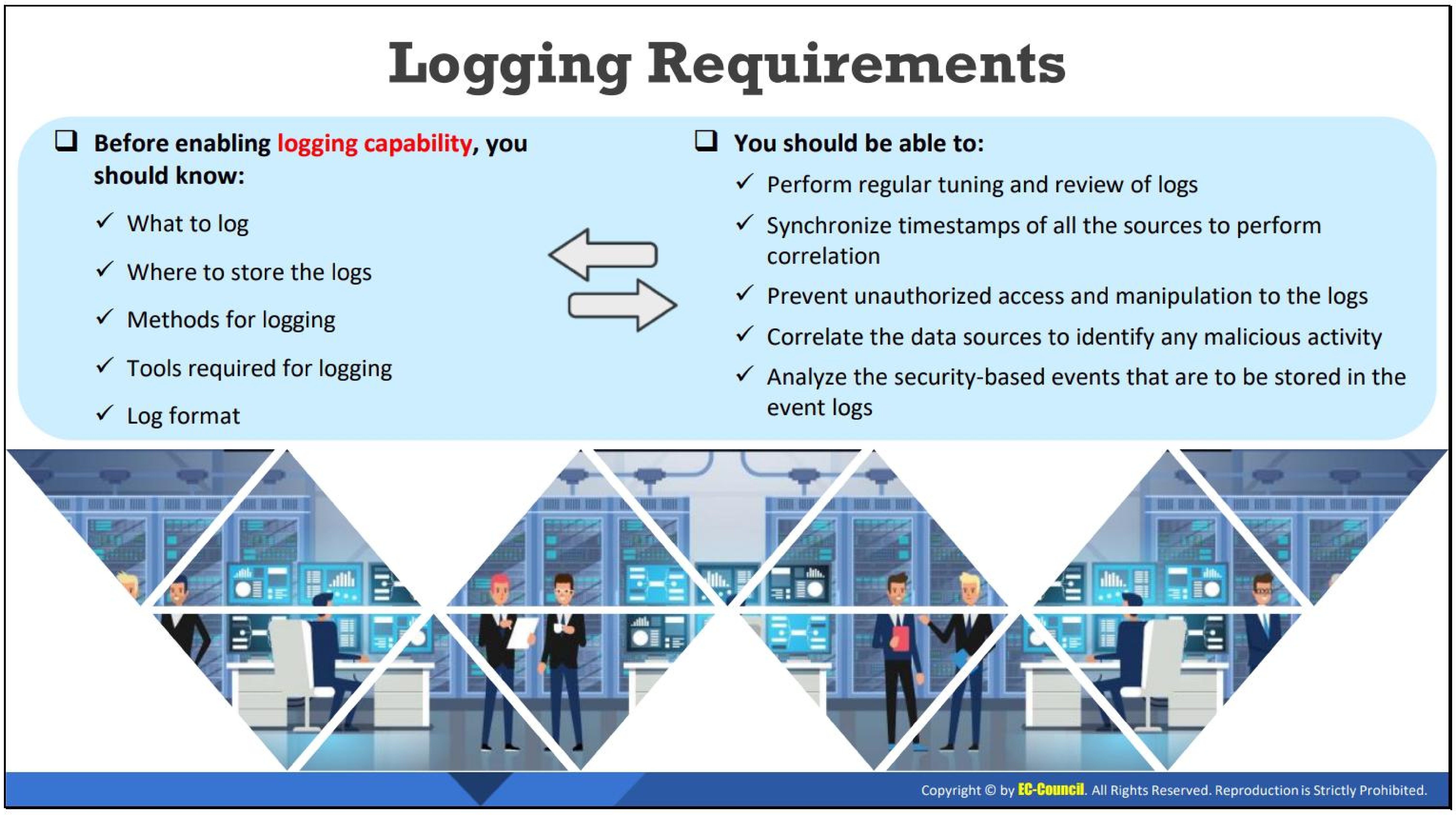
- To identify security incidents
- To monitor policy violations
- To identify fraudulent activity
- To identify operational and long-term problems
- To establish baselines
- To ensure compliance with laws, rules, and regulations

They are a good source of forensic information and help understand "what happened" after the occurrence of a security incident. Each log file comprises a variety of information, among which some may be invaluable. A proper analysis of log data enables actionable information to be identified, which helps the security professional in detecting and monitoring potential security breaches, internal misuse of information, operational issues, and long-term issues. It also helps validate whether the end-user has followed all the documented protocols to detect fraudulent activities and policy violations. It is also useful in internal investigations, security auditing and forensic analysis, determination of operational trends, and implementation of baselines. It provides information about what, how, and why a particular security intrusion occurred and, thus, helps in recovery and mitigating processes as well. It also ensures compliance with laws, rules, and regulations for storing and analyzing log data. Further, it can act as an audit trail for auditing purposes.

Logs can help in the following tasks.

- **System monitoring:** Logs provide detailed information about the transactions that are occurring across the environment. This facilitates constant system monitoring and helps in determining errors, anomalies, and suspicious system activities; it also helps respond to such situations as early as possible.
- **Troubleshooting:** The various information and messages available in log files can be utilized to troubleshoot a problem. However, these log files are not enough to troubleshoot network problems. Syslog need to be utilized for this purpose. It records events and arranges them into log files, which is beneficial in monitoring the various activities of the OS and in troubleshooting issues.
- **Forensics and analysis:** Logs play an important role in forensics and analysis process. They are a permanent source of record that cannot be altered through the normal course of actions. Logs are stored in a chronological sequence, thus they describe not only what happened but also when and how it happened. When logs are sent to another host or a central log collector, they act as a backup source of evidence and are especially useful if the original copy is suspected to have been tampered. If the information on the original source is found suspicious, then the separate copy is considered for authentication.

  Logs also support the findings of other evidences and improve their authenticity if their findings corroborate. Often, identifying the complete scenario of an event is not only dependent on one source of information but multiple sources such as files and their corresponding timestamps, network data, logs, etc. Logs may also assist in rejecting other evidences if they are suspected to have been tampered by an attacker.
- **Incident response:** Incident response activity requires proper correlation of log events across all devices and assets. This helps in determining the extent and impact of a network compromise and the steps required for remediation. However, the various security devices in a network may not have the required correlation capabilities to provide a complete picture of the attacker’s activities during an attack. A common solution to this problem is to correlate the activities using log files.



Logging Requirements

❑ **Before enabling logging capability, you should know:**

- ✓ What to log
- ✓ Where to store the logs
- ✓ Methods for logging
- ✓ Tools required for logging
- ✓ Log format

❑ **You should be able to:**

- ✓ Perform regular tuning and review of logs
- ✓ Synchronize timestamps of all the sources to perform correlation
- ✓ Prevent unauthorized access and manipulation to the logs
- ✓ Correlate the data sources to identify any malicious activity
- ✓ Analyze the security-based events that are to be stored in the event logs

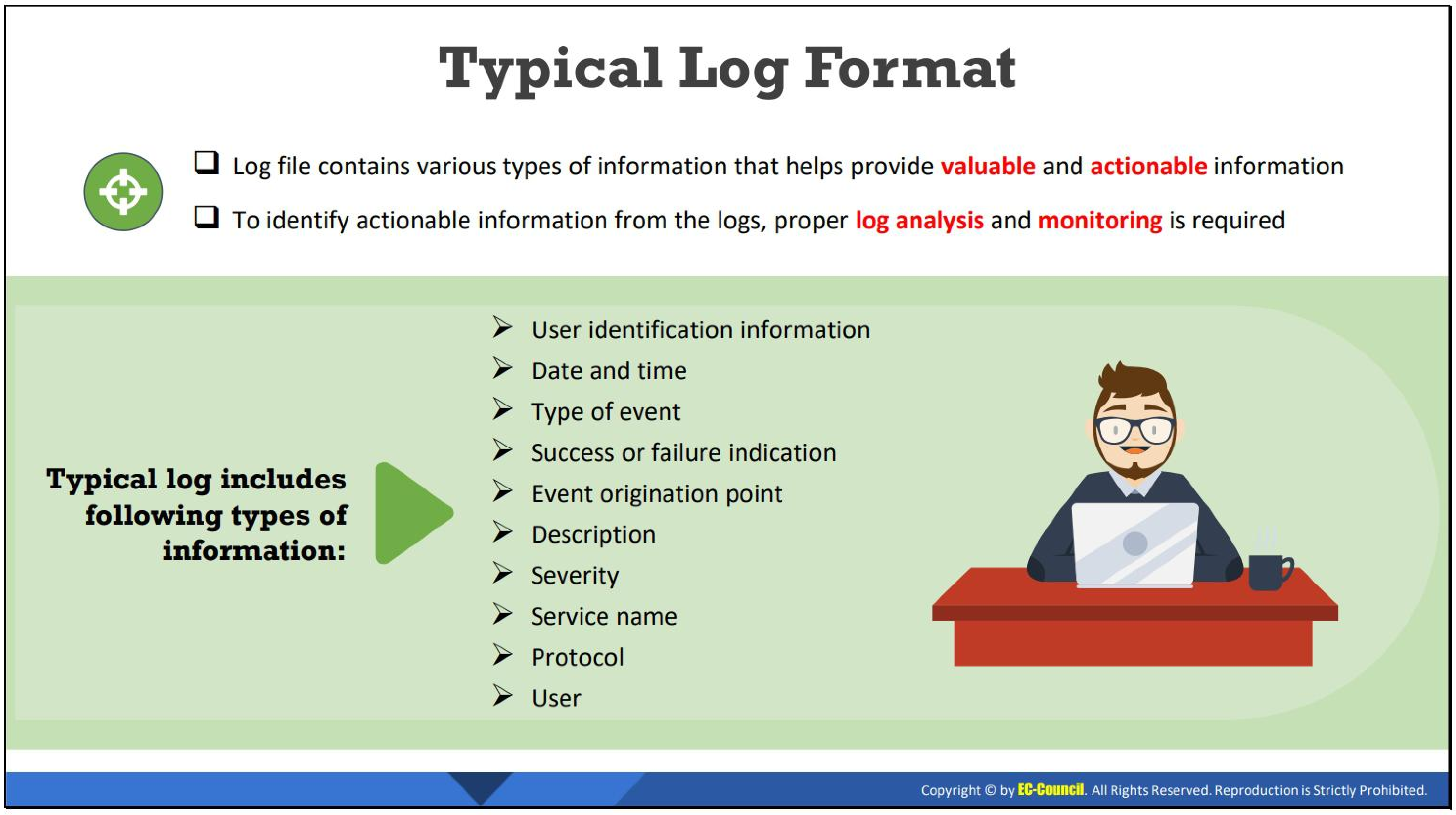
Logging Requirements

Before setting the requirements for a logging solution, details such as what to log, where to store the logs, methods for logging, tools required for logging, log format, etc. should be known.

Requirements for Logging

For effective security events logging, you should be able to do the following:

- Determine applications and systems (including those that are outsourced or are on the cloud) on which event logging is enabled
- Configure the information system for providing correct security incidents
- Perform regular tuning and review of logs to minimize the number of false positives
- Store events in event logs
- Normalize and aggregate security-related events
- Correlate the data sources to identify any malicious activity
- Synchronize timestamps of all the sources to perform correlation
- Prevent unauthorized access and manipulation of the logs
- Analyze security-based events that are to be stored in the event logs



Typical Log Format

- ❑ Log file contains various types of information that helps provide **valuable** and **actionable** information
- ❑ To identify actionable information from the logs, proper **log analysis** and **monitoring** is required

Typical log includes following types of information:

- ➢ User identification information
- ➢ Date and time
- ➢ Type of event
- ➢ Success or failure indication
- ➢ Event origination point
- ➢ Description
- ➢ Severity
- ➢ Service name
- ➢ Protocol
- ➢ User

Typical Log Format

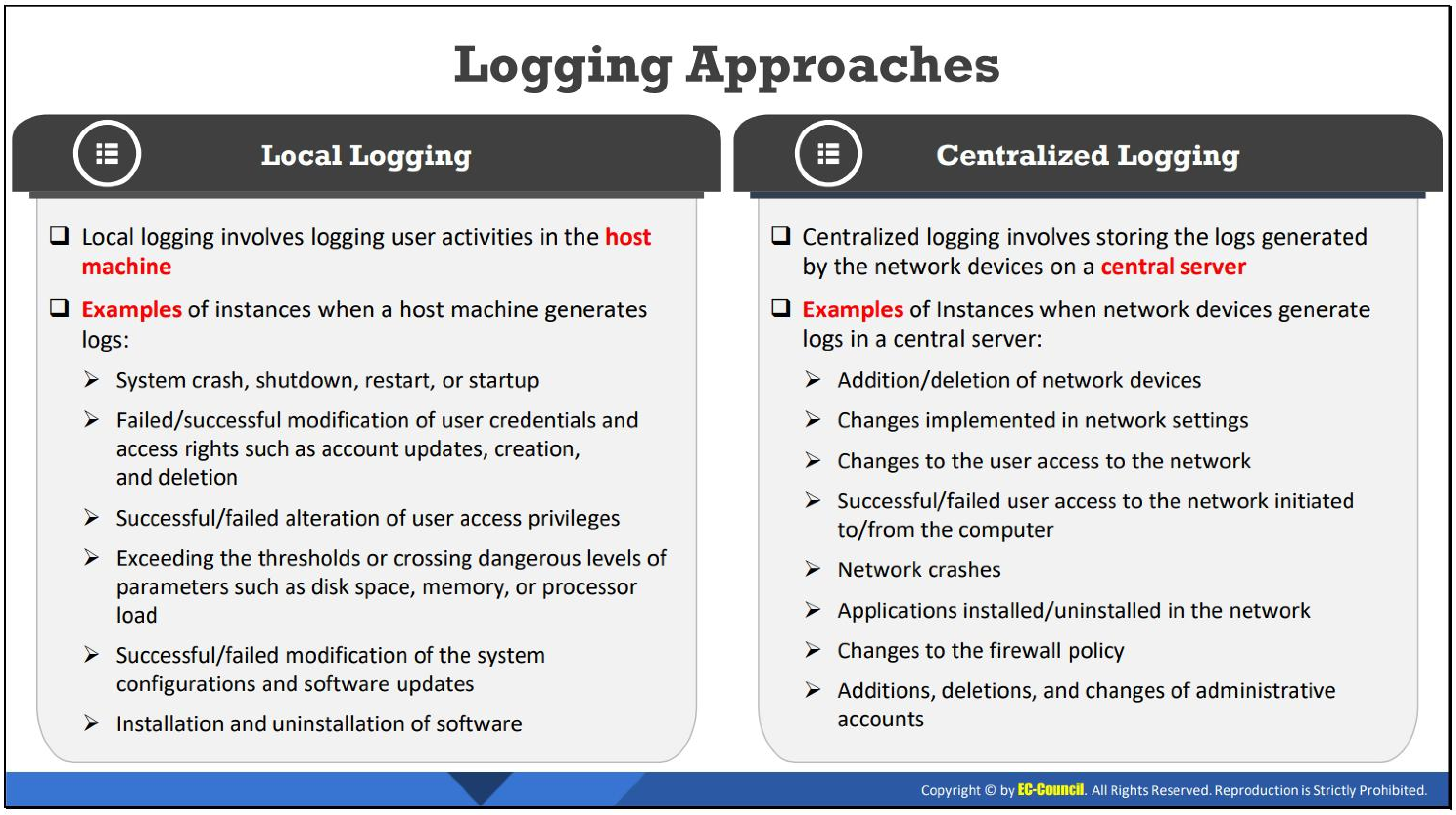
A log file contains various types of valuable and actionable information. The timestamp is an important item in any log as it tells when an event occurred. Therefore, all systems must be configured to synchronize their time from an authoritative time source. The timestamp also helps in filtering data in search results and in identifying the logs that are not matching the standards and require attention. It can help correlate events from multiple systems and present the complete picture of an event. In addition to a timestamp, a description of the event and why it occurred is also recorded. In a network-based event, the IP address information is also present in a log. Further, authenticated user information should also be available in a log. The information present in a log helps security professionals in monitoring user activities or analyzing them if something goes wrong.

The following are the types of information included in a log:

- User identification information
- Date and time
- Type of event
- Success or failure indication
- Event origination point
- Description
- Severity
- Service name
- Protocol
- User

There are also several things that need attention while setting up the configuration of a log. It should be ensured that private or protected information is not transmitted to the log files. Additionally, information such as passwords, encryption keys, bank information, credit card information, personal identifiable information, personal health information, source code, etc. should not be logged. Only the following should be sent to a log file:

- Initialization/clearing of audit logs
- Creation/deletion of system-level objects
- All access to the log
- All administrative access
- All actions taken by administrators
- All actions taken by a person having root or administrative privileges



Logging Approaches

Local Logging

- ❑ Local logging involves logging user activities in the **host machine**
- ❑ **Examples** of instances when a host machine generates logs:
  - ➢ System crash, shutdown, restart, or startup
  - ➢ Failed/successful modification of user credentials and access rights such as account updates, creation, and deletion
  - ➢ Successful/failed alteration of user access privileges
  - ➢ Exceeding the thresholds or crossing dangerous levels of parameters such as disk space, memory, or processor load
  - ➢ Successful/failed modification of the system configurations and software updates
  - ➢ Installation and uninstallation of software

Centralized Logging

- ❑ Centralized logging involves storing the logs generated by the network devices on a **central server**
- ❑ **Examples** of Instances when network devices generate logs in a central server:
  - ➢ Addition/deletion of network devices
  - ➢ Changes implemented in network settings
  - ➢ Changes to the user access to the network
  - ➢ Successful/failed user access to the network initiated to/from the computer
  - ➢ Network crashes
  - ➢ Applications installed/uninstalled in the network
  - ➢ Changes to the firewall policy
  - ➢ Additions, deletions, and changes of administrative accounts

Logging Approaches

Logs are stored either on local disks or the central server, depending on the size of the system.

- **Local logging**

  Local logging involves logging user activities in the host machine. In other words, it is the process of writing logs into files stored on the local disk. This approach is used by those systems that have a limited number of hosts. If a system has many hosts, then it becomes difficult to manage logs and analyze them. It also becomes complicated to identify security-related events across multiple log files on multiple servers. A common solution to this problem is to use centralized logging.

  Example of instances when a host machine generates a log record:

  - System crash, shutdown, restart, or startup
  - Failed and successful modification of user credentials and access rights such as account updates, creation, and deletion
  - Successful/failed alteration of user access privileges
  - Exceeding the thresholds or crossing dangerous levels of parameters such as disk space, memory, or processor load
  - Successful/failed modification of system configuration and software updates
  - Installation and uninstallation of software

- **Centralized logging**

  Centralized logging involves storing the logs generated by the network devices on a central server. In other words, it is the process of collecting and aggregating logs in one central location. It works in four parts: log collection, transport, storage, and analysis.

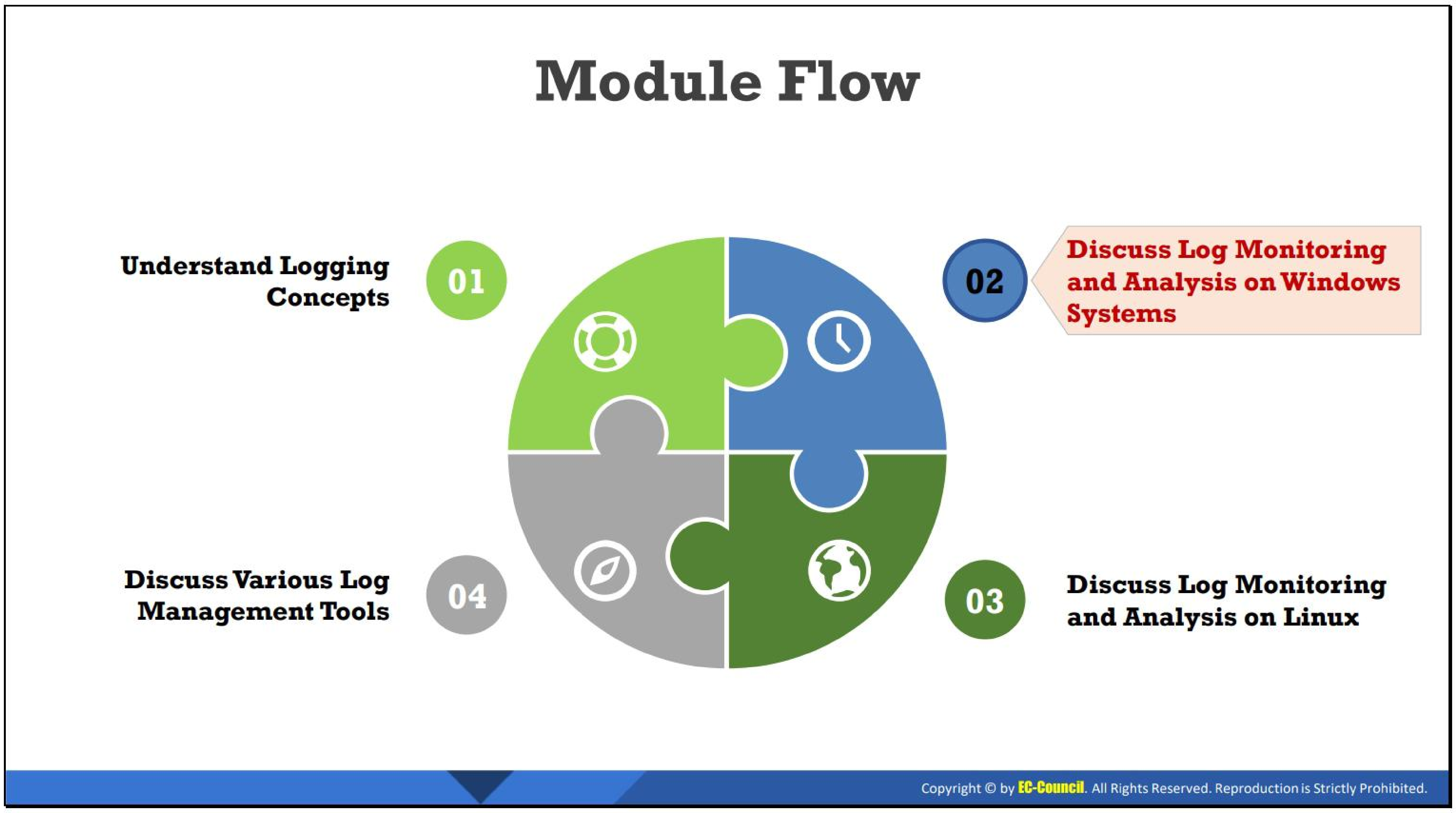
  Benefits of centralized logging:

  - Logs stored in a central location are indispensable when trying to troubleshoot security-related problems and determine why they happened.
  - It enables proactive management of the network.
  - It facilitates in-depth data analysis and delivers greater value.
  - It minimizes the risk of losing data.
  - It enhances network security.

  Examples of instances when a network device generates a log record in a central server:

  - Addition/deletion of network devices
  - Changes implemented to network settings
  - Changes in user access to the network
  - Successful/failed user access to the network initiated to/from the computer
  - Network crashes
  - Applications installed/uninstalled in the network
  - Changes to the firewall policy
  - Additions, deletions, and changes in administrative accounts

Module Flow

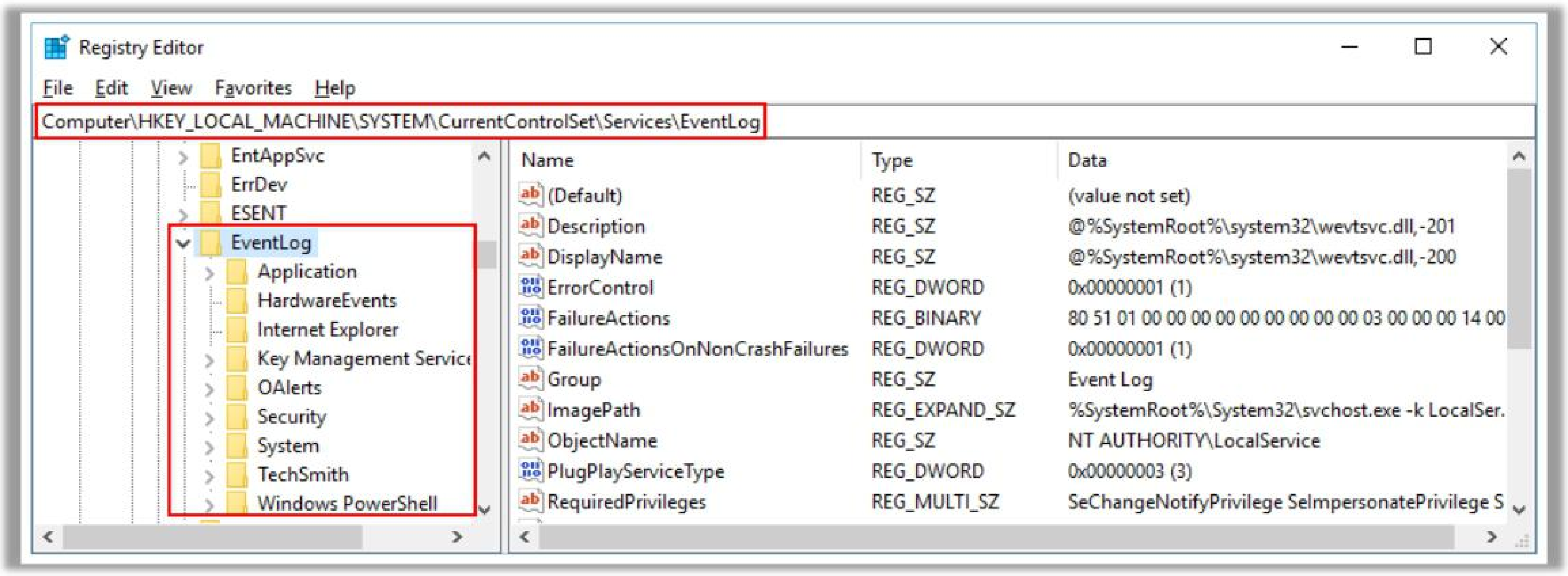

Understand Logging Concepts
01
02
Discuss Log Monitoring and Analysis on Windows Systems
Discuss Various Log Management Tools
04
03
Discuss Log Monitoring and Analysis on Linux
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Discuss Log Monitoring and Analysis on Windows Systems

The objective of this section is to explain monitoring and analysis of logs in Windows systems. It describes Windows event logs, their types, and how to monitor and analyze them.

Windows Logs

- ❑ Windows OS tracks various events, activities, and functions through logs
- ❑ Windows event logs, consisting of a header and a series of **event records**, provide a standard, centralized way for applications (and the OS) to record important software and hardware events
- ❑ Windows Event log audit configurations (i.e., log retention, log size, etc.) are recorded based on the registry key: ***`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\<Event Log>`***



Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog

- EntAppSvc
- ErrDev
- ESENT
- EventLog
  - Application
  - HardwareEvents
  - Internet Explorer
  - Key Management Servic
  - OAlerts
  - Security
  - System
  - TechSmith
  - Windows PowerShell

Name	Type	Data
(Default)	REG_SZ	(value not set)
Description	REG_SZ	@%SystemRoot%\system32\wevtsvc.dll,-201
DisplayName	REG_SZ	@%SystemRoot%\system32\wevtsvc.dll,-200
ErrorControl	REG_DWORD	0x00000001 (1)
FailureActions	REG_BINARY	80 51 01 00 00 00 00 00 00 00 00 00 03 00 00 00 14 00
FailureActionsOnNonCrashFailures	REG_DWORD	0x00000001 (1)
Group	REG_SZ	Event Log
ImagePath	REG_EXPAND_SZ	%SystemRoot%\System32\svchost.exe -k LocalSer.
ObjectName	REG_SZ	NT AUTHORITY\LocalService
PlugPlayServiceType	REG_DWORD	0x00000003 (3)
RequiredPrivileges	REG_MULTI_SZ	SeChangeNotifyPrivilege SeImpersonatePrivilege S

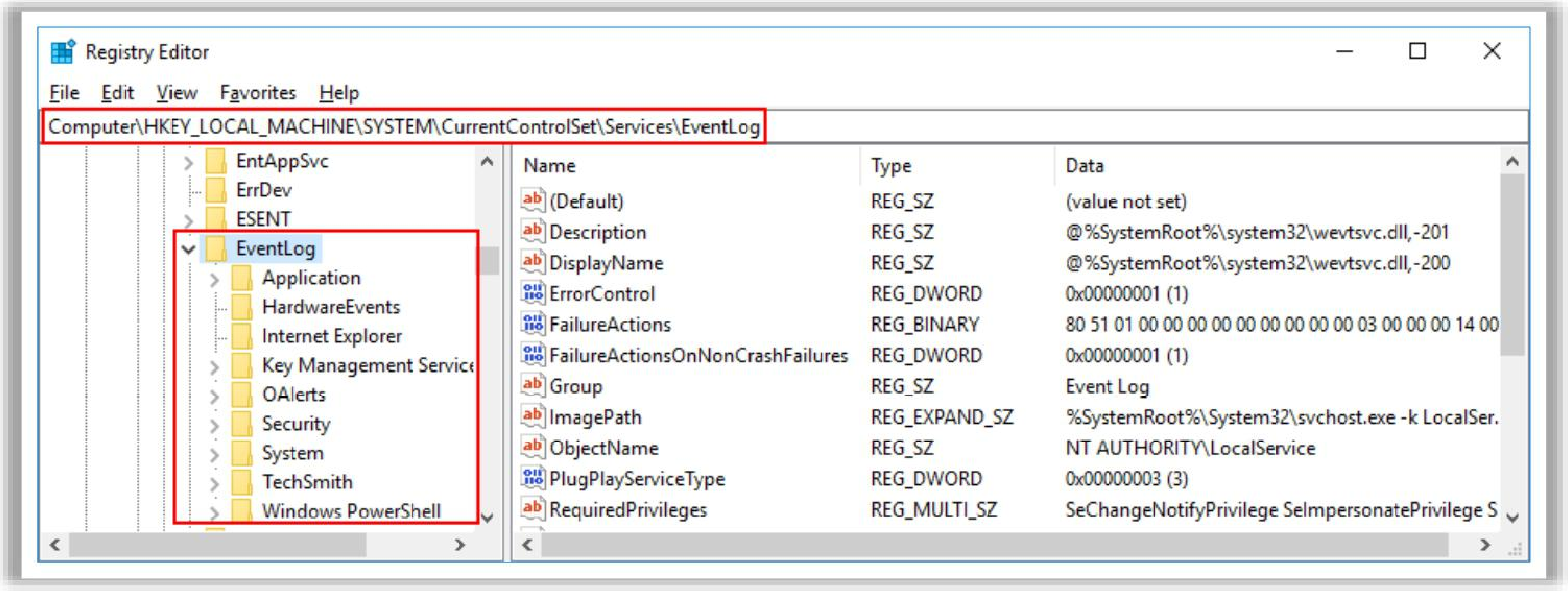
Windows Logs

Windows event logging service collects events from multiple sources and keeps them in a single location known as Windows event log. These logs act as the primary source of evidence for all important actions/activities on a Windows system. Windows event log contains logs of system, security, and application notifications that are monitored and analyzed by security professionals to detect issues in the system. It provides a standard, centralized way for applications (and the OS) to record important software and hardware events. It uses a structured data format that simplifies the process of searching and filtering for a particular type of log. Windows event log files can be viewed through Event Viewer, which is the programming interface that facilitates analysis of these logs. Each event is a log entry that includes information such as event time, event source that caused the event, event type (Information, Warning, Error, Success Audit, or Failure Audit), and event ID for the event type.

Windows event log audit configurations, that is, log retention, log size, etc. are recorded based on the below registry key.

`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\<Event Log>`

This key comprises various subkeys, which are known as logs. Each log includes registry values such as CustomSD, DisplayNameFile, DisplayNameID, File, MaxSize, etc., which can be configured as per requirement.



Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog

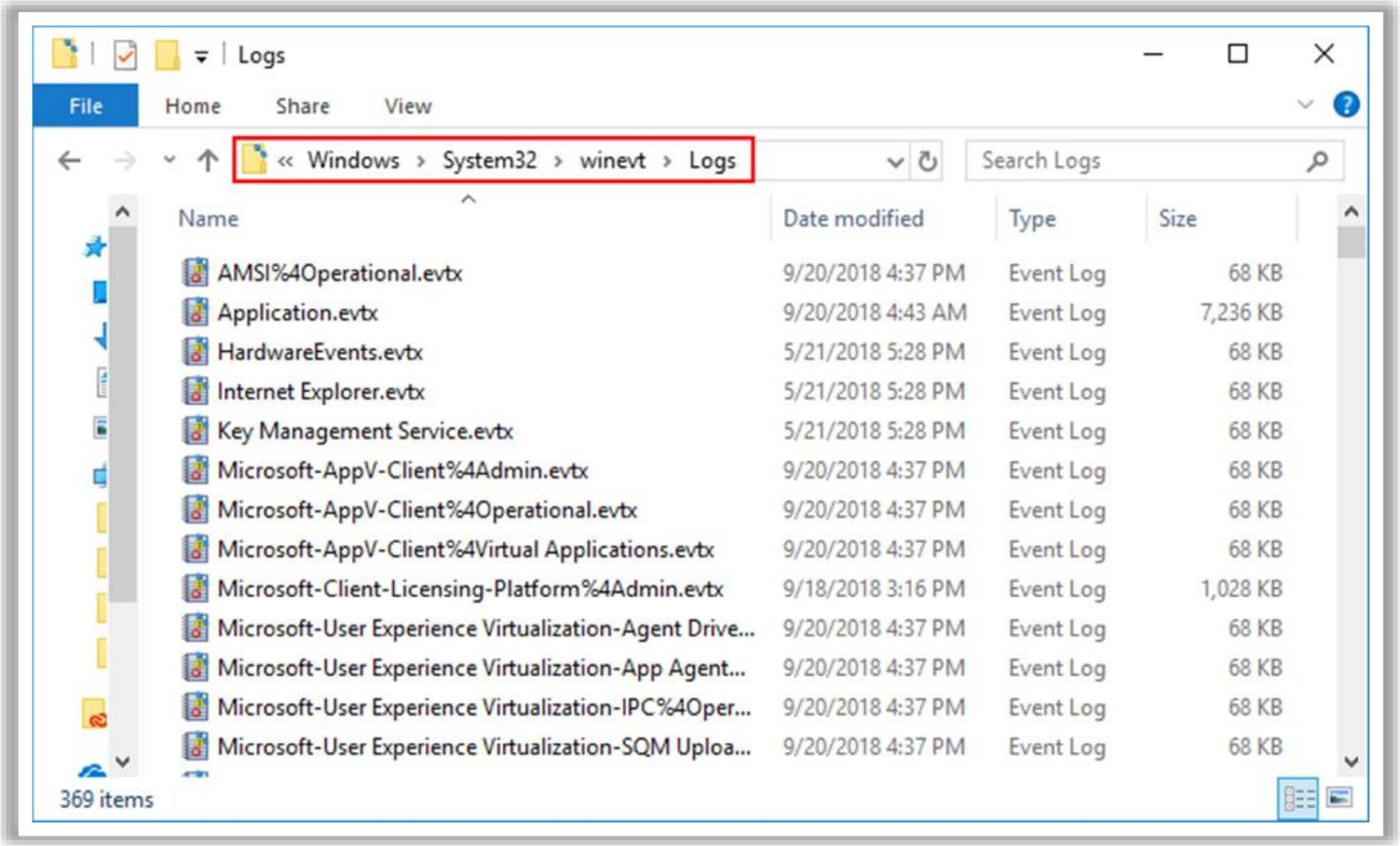
- EntAppSvc
- ErrDev
- ESENT
- EventLog
  - Application
  - HardwareEvents
  - Internet Explorer
  - Key Management Servic
  - OAlerts
  - Security
  - System
  - TechSmith
  - Windows PowerShell

Name	Type	Data
(Default)	REG_SZ	(value not set)
Description	REG_SZ	@%SystemRoot%\system32\wevtsvc.dll,-201
DisplayName	REG_SZ	@%SystemRoot%\system32\wevtsvc.dll,-200
ErrorControl	REG_DWORD	0x00000001 (1)
FailureActions	REG_BINARY	80 51 01 00 00 00 00 00 00 00 00 00 03 00 00 00 14 00
FailureActionsOnNonCrashFailures	REG_DWORD	0x00000001 (1)
Group	REG_SZ	Event Log
ImagePath	REG_EXPAND_SZ	%SystemRoot%\System32\svchost.exe -k LocalSer.
ObjectName	REG_SZ	NT AUTHORITY\LocalService
PlugPlayServiceType	REG_DWORD	0x00000003 (3)
RequiredPrivileges	REG_MULTI_SZ	SeChangeNotifyPrivilege SeImpersonatePrivilege S

Figure 18.2: Screenshot of Registry Editor

Windows Event Log File Internals

In simple terms, Windows event log files are databases with records related to the system, security, and applications. The databases related to the system are stored in a file named **`System.evtx`**, the databases related to security are stored in a file named **`Security.evtx`**, and the databases related to applications are stored in a file named **`Application.evtx`**. These Windows event log files are stored in **`C:\Windows\System32\winevt\Logs`** folder, as shown in the below figure:



« Windows › System32 › winevt › Logs

Name	Date modified	Type	Size
AMSI%4Operational.evtx	9/20/2018 4:37 PM	Event Log	68 KB
Application.evtx	9/20/2018 4:43 AM	Event Log	7,236 KB
HardwareEvents.evtx	5/21/2018 5:28 PM	Event Log	68 KB
Internet Explorer.evtx	5/21/2018 5:28 PM	Event Log	68 KB
Key Management Service.evtx	5/21/2018 5:28 PM	Event Log	68 KB
Microsoft-AppV-Client%4Admin.evtx	9/20/2018 4:37 PM	Event Log	68 KB
Microsoft-AppV-Client%4Operational.evtx	9/20/2018 4:37 PM	Event Log	68 KB
Microsoft-AppV-Client%4Virtual Applications.evtx	9/20/2018 4:37 PM	Event Log	68 KB
Microsoft-Client-Licensing-Platform%4Admin.evtx	9/18/2018 3:16 PM	Event Log	1,028 KB
Microsoft-User Experience Virtualization-Agent Drive...	9/20/2018 4:37 PM	Event Log	68 KB
Microsoft-User Experience Virtualization-App Agent...	9/20/2018 4:37 PM	Event Log	68 KB
Microsoft-User Experience Virtualization-IPC%4Oper...	9/20/2018 4:37 PM	Event Log	68 KB
Microsoft-User Experience Virtualization-SQM Uploa...	9/20/2018 4:37 PM	Event Log	68 KB

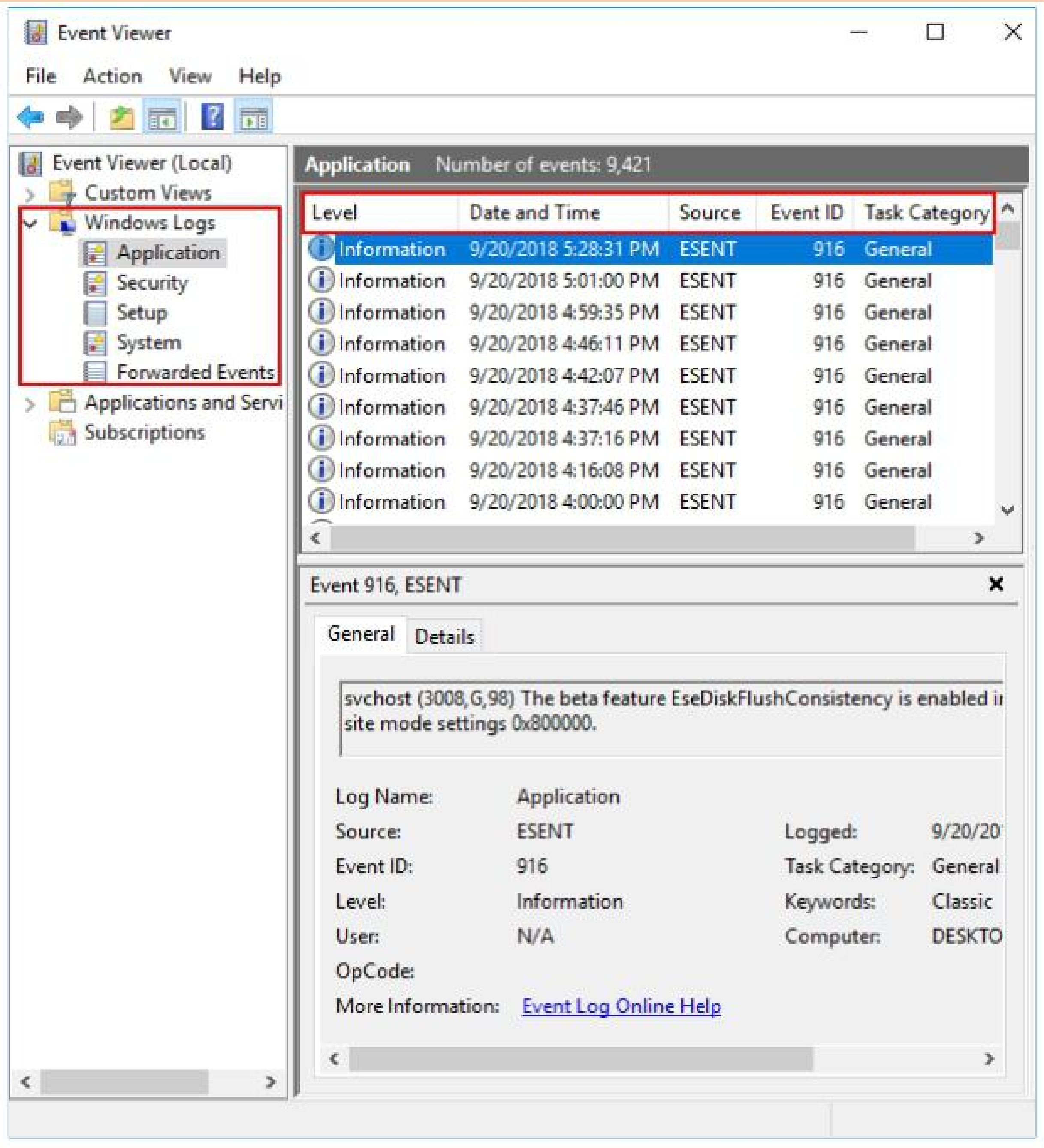
369 items

Figure 18.3: Screenshot of Windows event log files

All **`.evtx`** files can be opened and read with Event Viewer.

Windows Event Log Types and Entries

- **Event Viewer** provides a quick overview of when, where, and how an event occurred
- Check **Windows Event Log** for various types of logs:
  - **System logs**: Windows and Windows system service logs
  - **Security logs**: Audit logs based on success/failed events
  - **Setup logs**: Configuration logs
  - **Application logs**: Events based on severity categorized
  - **Forwarded event logs**: Events forwarded by another computer in a network
- Typical log entries contain the following types of information about the events:
  - **Level:** It defines the **severity of the event**; various types of severity levels are Error, Warning, Information, Success Audit, and Failure Audit
  - **Keywords:** It defines the **type of event** occurred; various types of events are AuditFailure, AuditSuccess, Classic, Correlation Hint, Response Time, SQM, WDI Context, and WDI Diag
  - **Date and Time:** It defines the **date of events** occurred
  - **Source:** It defines the **source of the event**
  - **Event ID:** A **unique** event ID is assigned for each type of event
  - **Task Category:** It defines task categories



Application Number of events: 9,421

Level	Date and Time	Source	Event ID	Task Category
Information	9/20/2018 5:28:31 PM	ESENT	916	General
Information	9/20/2018 5:01:00 PM	ESENT	916	General
Information	9/20/2018 4:59:35 PM	ESENT	916	General
Information	9/20/2018 4:46:11 PM	ESENT	916	General
Information	9/20/2018 4:42:07 PM	ESENT	916	General
Information	9/20/2018 4:37:46 PM	ESENT	916	General
Information	9/20/2018 4:37:16 PM	ESENT	916	General
Information	9/20/2018 4:16:08 PM	ESENT	916	General
Information	9/20/2018 4:00:00 PM	ESENT	916	General

Event 916, ESENT

svchost (3008,G,98) The beta feature EseDiskFlushConsistency is enabled i
site mode settings 0x800000.

Log Name:	Application		
Source:	ESENT	Logged:	9/20/20
Event ID:	916	Task Category:	General
Level:	Information	Keywords:	Classic
User:	N/A	Computer:	DESKTO
OpCode:			
More Information:	Event Log Online Help		

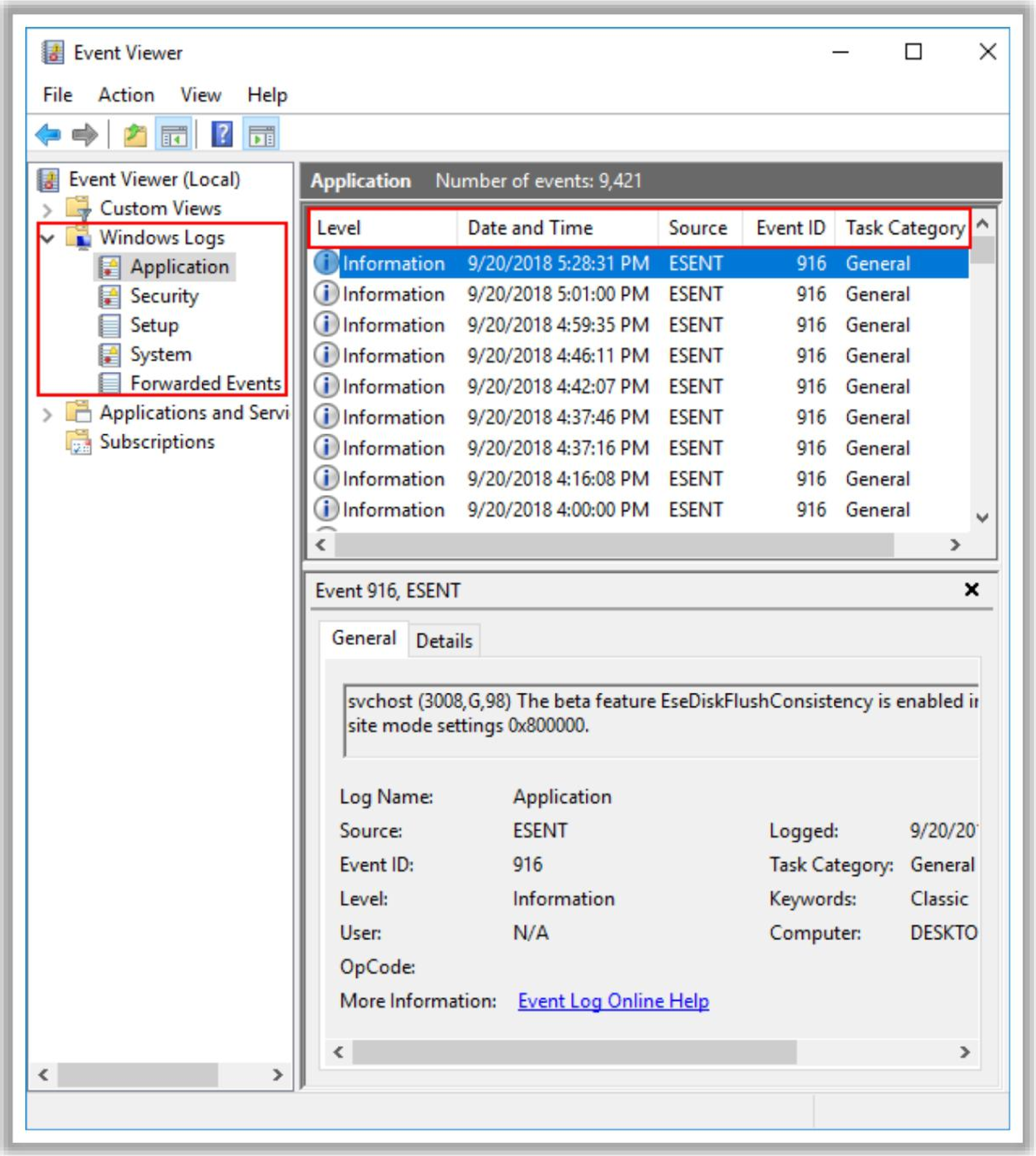
Windows Event Log Types and Entries

Windows Event Viewer is a tool that provides a quick overview of when, where, and how an event occurred. It provides detailed information about events, errors, warnings, and information messages that are generated by the OS. It is especially useful for troubleshooting issues.

In Event Viewer, events are stored in Windows logs files under different categories such as application, security, system, setup, and forwarded.

- **Application event log:** This includes events related to the applications installed on the system; specifically, informational events, warnings from the applications, and errors raised in an application. For example, if an application such as Microsoft Excel breaks down, then this event will be logged into Windows event log with the name of the application and why it is crashed.
- **Security event log:** This includes events related to Windows security; specifically, log-on/log-off activities, resource access, and also information based on Windows system's audit policies. It is analyzed by security professionals to identify attempted and/or successful unauthorized activities. For example, if the system attempts to verify account credentials when an end-user tries to log-on to a machine.
- **Setup event log:** This includes enterprise-focused events that cover all actions that occurred during installation; for example, the location of memory dump from bug checks.
- **System event log:** This includes events that are logged by the OS segments; specifically, information about hardware changes, system changes, device drivers, and all machine-related activities (for example, failure of the device driver).

- **Forwarded event log:** This includes events that are received from other systems present on the same network.
- **Custom log:** A custom log facilitates an application to change the size of the log or add access control lists (ACLs) without influencing other applications.



Level	Date and Time	Source	Event ID	Task Category
Information	9/20/2018 5:28:31 PM	ESENT	916	General
Information	9/20/2018 5:01:00 PM	ESENT	916	General
Information	9/20/2018 4:59:35 PM	ESENT	916	General
Information	9/20/2018 4:46:11 PM	ESENT	916	General
Information	9/20/2018 4:42:07 PM	ESENT	916	General
Information	9/20/2018 4:37:46 PM	ESENT	916	General
Information	9/20/2018 4:37:16 PM	ESENT	916	General
Information	9/20/2018 4:16:08 PM	ESENT	916	General
Information	9/20/2018 4:00:00 PM	ESENT	916	General

svchost (3008,G,98) The beta feature EseDiskFlushConsistency is enabled ir
site mode settings 0x800000.

Log Name:	Application		
Source:	ESENT	Logged:	9/20/20
Event ID:	916	Task Category:	General
Level:	Information	Keywords:	Classic
User:	N/A	Computer:	DESKTO
OpCode:			
More Information:	Event Log Online Help		

Figure 18.4: Screenshot of Windows Event Viewer

Each log entry in a particular log type contains the following types of information about the event:

- **Level:** It defines the severity of the event. The various types of severity levels are Error, Warning, Information, Success Audit, and Failure Audit.
- **Keywords:** It is a set of categories or tags that defines a type of event. The various types of events are AuditFailure, AuditSuccess, Classic, Correlation Hint, Response Time, SQM, WDI Context, and WDI Diag.

- **Date and time:** It defines the date on which an event occurred.
- **Source:** It defines the source of the event.
- **Event ID:** A unique event ID is assigned for each type of event.
- **Task category:** It defines the category of ask.
- **User:** It defines the username on whose behalf a particular event was generated.
- **Operational code:** It defines the activity that an event was performing when an event was raised.
- **Log:** It defines the name of the log on which the event was recorded.
- **Computer:** It defines the computer name on which the event was raised.

Additional event properties can be viewed by adding columns in Event Viewer display. To do so, click on View (menu bar) and then Add/Remove Columns. The following types of properties can be added or removed.

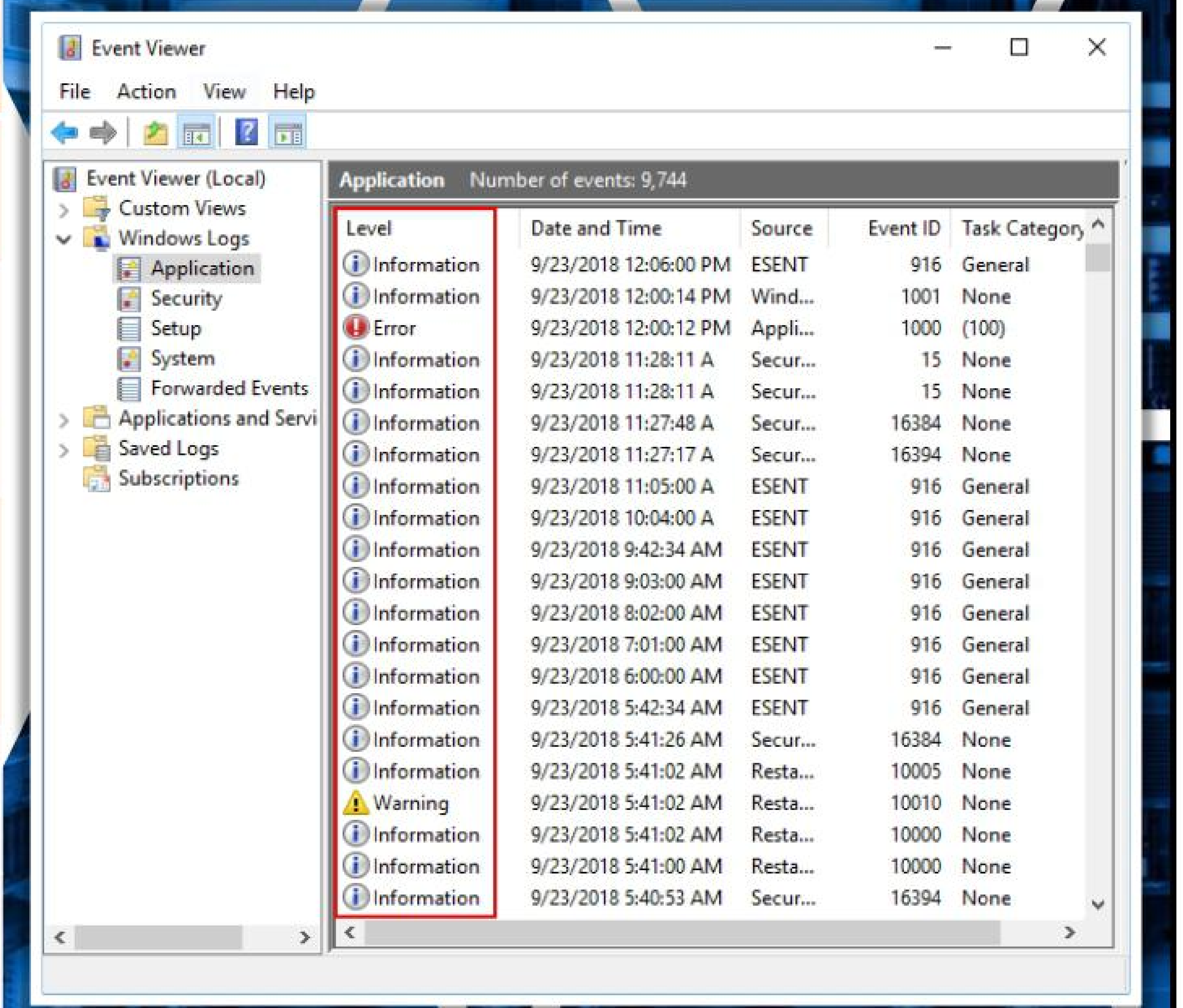
- **Process ID:** It defines the process identification number for the generated event.
- **Thread ID:** It defines the thread identification number for the generated event.
- **Processor ID:** It defines the processor identification number that processed the event.
- **Session ID:** It defines the terminal server session identification number in which the event was raised.
- **Kernel time:** It defines the time taken in executing kernel-mode instructions in CPU time units.
- **User time:** It defines the time taken in executing user-mode instructions in CPU time units.
- **Processor time:** It defines the time taken in executing kernel-mode instructions in CPU ticks.
- **Correlation ID:** It defines the activity in the process for which the event was involved.
- **Relative correlation ID:** It defines the related activity in a process for which the event was involved.

You can also view XML representation of an event by clicking the Details tab in an event's properties.

Event Types

Event Type	Description
Error	An event indicates a **significant problem** such as loss of data or loss of functionality. For example, if a service fails to load during startup, an Error event is logged
Warning	An event that is not necessarily **significant** but may indicate a **possible future problem**. For example, when disk space is low, a Warning event is logged. If an application can recover from an event without loss of functionality or data, it can generally classify the event as a Warning event
Information	An event that describes **successful operation** of an application, driver, or service. For example, when a network driver loads successfully, it may be appropriate to log an Information event. Note that it is generally inappropriate for a desktop application to log an event each time it starts
Success Audit	An event that records an **audited security access attempt** that is **successful**. For example, a user's successful attempt to log on to the system is logged as a Success Audit event
Failure Audit	An event that records an **audited security access attempt** that **fails**. For example, if a user tries to access a network drive and fails, the attempt is logged as a Failure Audit event

https://docs.microsoft.com

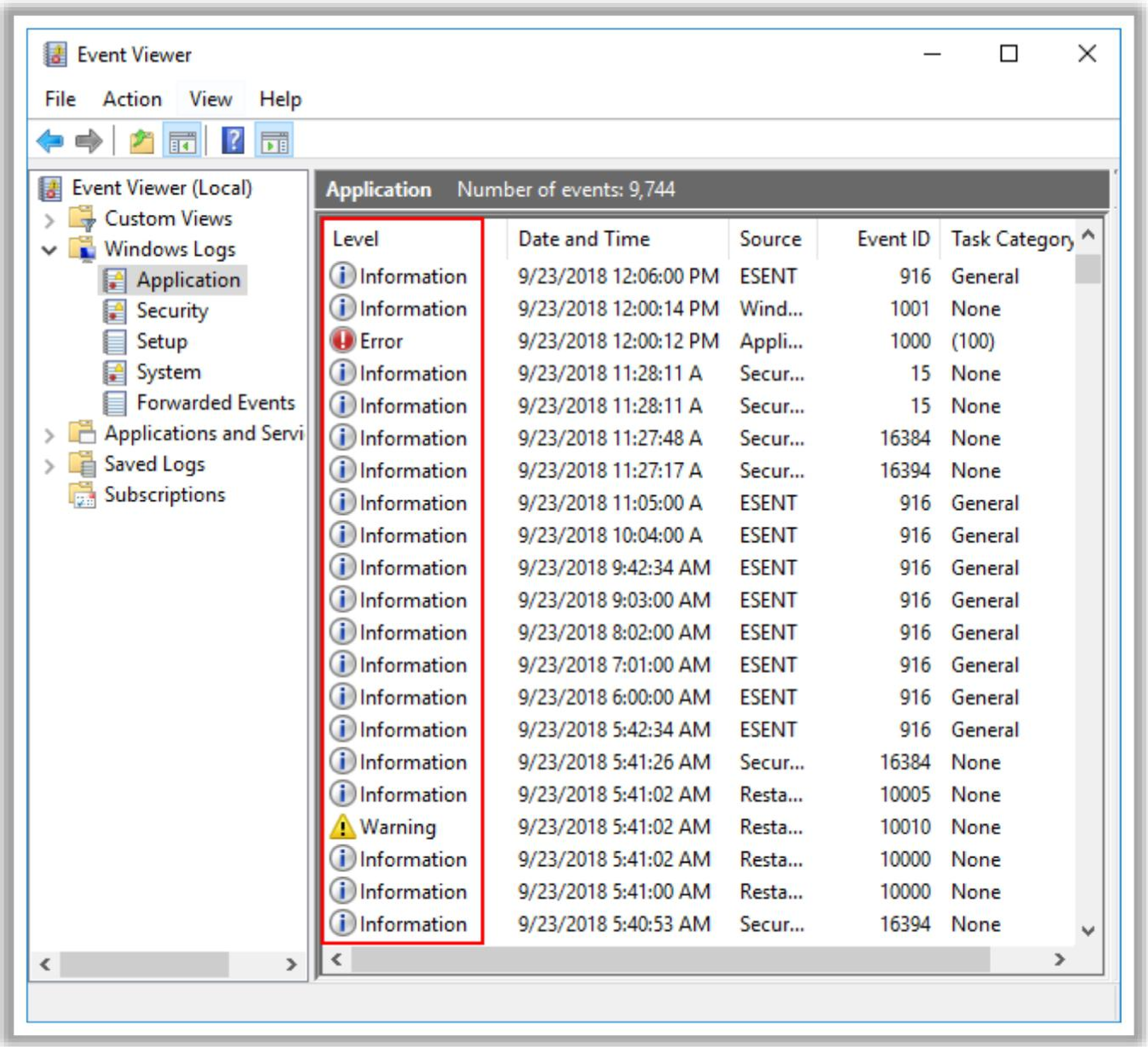


Level	Date and Time	Source	Event ID	Task Category
Information	9/23/2018 12:06:00 PM	ESENT	916	General
Information	9/23/2018 12:00:14 PM	Wind...	1001	None
Error	9/23/2018 12:00:12 PM	Appli...	1000	(100)
Information	9/23/2018 11:28:11 A	Secur...	15	None
Information	9/23/2018 11:28:11 A	Secur...	15	None
Information	9/23/2018 11:27:48 A	Secur...	16384	None
Information	9/23/2018 11:27:17 A	Secur...	16394	None
Information	9/23/2018 11:05:00 A	ESENT	916	General
Information	9/23/2018 10:04:00 A	ESENT	916	General
Information	9/23/2018 9:42:34 AM	ESENT	916	General
Information	9/23/2018 9:03:00 AM	ESENT	916	General
Information	9/23/2018 8:02:00 AM	ESENT	916	General
Information	9/23/2018 7:01:00 AM	ESENT	916	General
Information	9/23/2018 6:00:00 AM	ESENT	916	General
Information	9/23/2018 5:42:34 AM	ESENT	916	General
Information	9/23/2018 5:41:26 AM	Secur...	16384	None
Information	9/23/2018 5:41:02 AM	Resta...	10005	None
Warning	9/23/2018 5:41:02 AM	Resta...	10010	None
Information	9/23/2018 5:41:02 AM	Resta...	10000	None
Information	9/23/2018 5:41:00 AM	Resta...	10000	None
Information	9/23/2018 5:40:53 AM	Secur...	16394	None

Event Types

Events are categorized into five types, based on their severity levels.

- **Error:** This type of event describes a significant problem such as loss of data or functionality. For example, an error event is recorded when a service is unable to load at startup.
- **Warning:** This type of event is of less importance but may describe a possible future problem. For example, a warning event is recorded when there is low space on the disk. Events are also classified as a warning event when an application can recover from an event without any loss.
- **Information:** This type of event indicates the successful operation of an application, driver, or service. For example, an information event is recorded when an application driver loads successfully.
- **Success audit:** This type of event is recorded when any successfully audited security access attempt is detected. For example, a success audit event is recorded when a user successfully logs on to the system.
- **Failure Audit:** This type of event is recorded when any unsuccessful audited security access attempt is detected. For example, a Failure Audit event is recorded when a user fails in accessing a network drive.



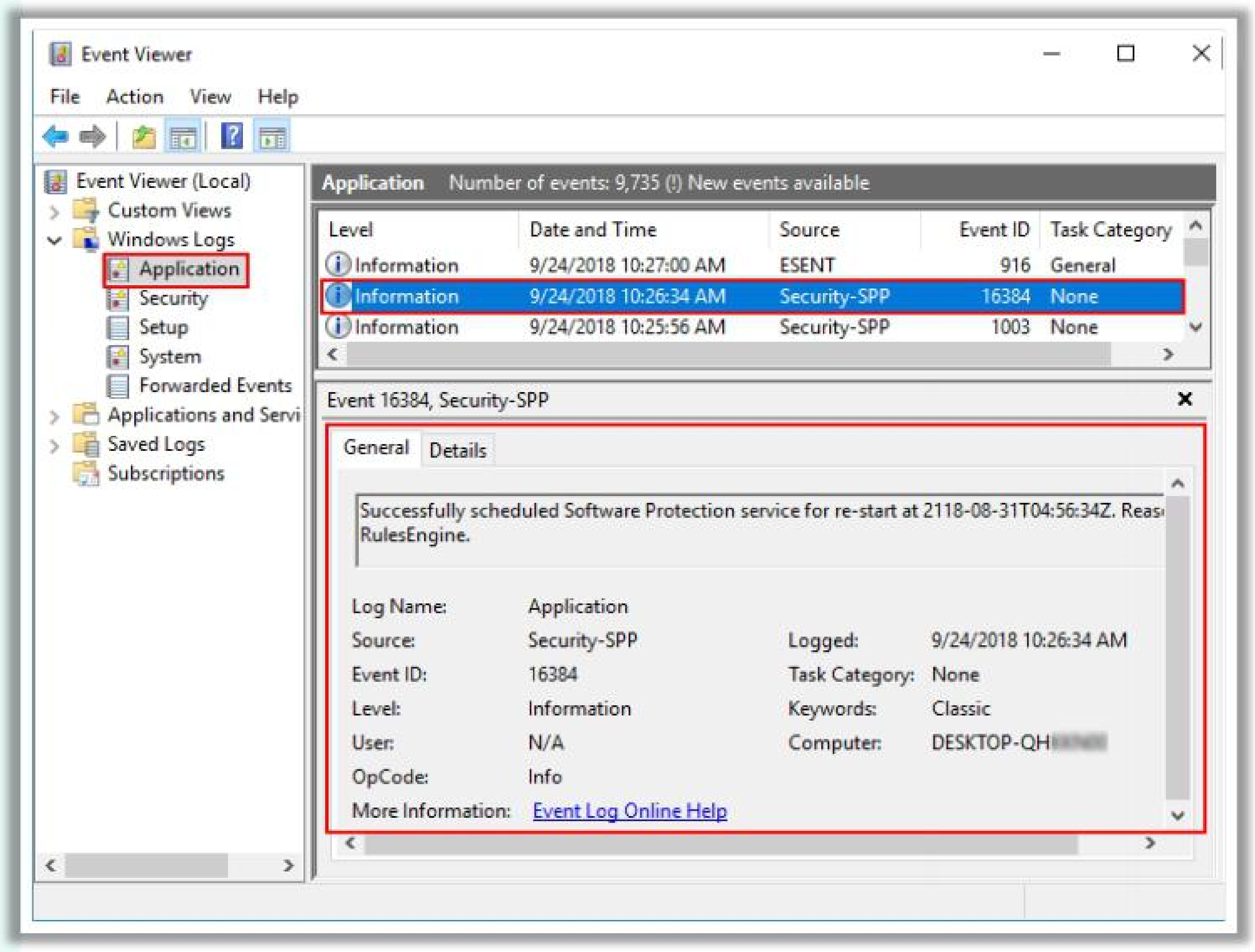
Application Number of events: 9,744

Level	Date and Time	Source	Event ID	Task Category
Information	9/23/2018 12:06:00 PM	ESENT	916	General
Information	9/23/2018 12:00:14 PM	Wind...	1001	None
Error	9/23/2018 12:00:12 PM	Appli...	1000	(100)
Information	9/23/2018 11:28:11 A	Secur...	15	None
Information	9/23/2018 11:28:11 A	Secur...	15	None
Information	9/23/2018 11:27:48 A	Secur...	16384	None
Information	9/23/2018 11:27:17 A	Secur...	16394	None
Information	9/23/2018 11:05:00 A	ESENT	916	General
Information	9/23/2018 10:04:00 A	ESENT	916	General
Information	9/23/2018 9:42:34 AM	ESENT	916	General
Information	9/23/2018 9:03:00 AM	ESENT	916	General
Information	9/23/2018 8:02:00 AM	ESENT	916	General
Information	9/23/2018 7:01:00 AM	ESENT	916	General
Information	9/23/2018 6:00:00 AM	ESENT	916	General
Information	9/23/2018 5:42:34 AM	ESENT	916	General
Information	9/23/2018 5:41:26 AM	Secur...	16384	None
Information	9/23/2018 5:41:02 AM	Resta...	10005	None
Warning	9/23/2018 5:41:02 AM	Resta...	10010	None
Information	9/23/2018 5:41:02 AM	Resta...	10000	None
Information	9/23/2018 5:41:00 AM	Resta...	10000	None
Information	9/23/2018 5:40:53 AM	Secur...	16394	None

Figure 18.5: Various Windows Event Types

Monitoring and Analysis of Windows Logs

- ❑ Open **Event Viewer**, click the required log you want to view
- ❑ In the details pane, click the event that you want to view. Description and header information is displayed in the **Preview Pane**
- ❑ The information displayed in the **Preview Pane** about the event is as follows:
  - **Log Name**: The type of Windows log
  - **Source**: Source is the cause that is responsible for the event raised by either an individual, or a system, or a program
  - **Event ID**: The type of event that occurred
  - **Level**: Event level type is divided into five types: Error, Warning, Information, Success Audit, and Failure Audit
  - **User**: User responsible and who logged on the computer at the instance of the event
  - **Logged**: The timestamp of the event
  - **Task Category**: Primarily used in case of security log, which classifies an event based on the event source
  - **Computer**: The name assigned to the computer where the event occurred



Application Number of events: 9,735 (!) New events available

Level	Date and Time	Source	Event ID	Task Category
Information	9/24/2018 10:27:00 AM	ESENT	916	General
Information	9/24/2018 10:26:34 AM	Security-SPP	16384	None
Information	9/24/2018 10:25:56 AM	Security-SPP	1003	None

Event 16384, Security-SPP

General Details

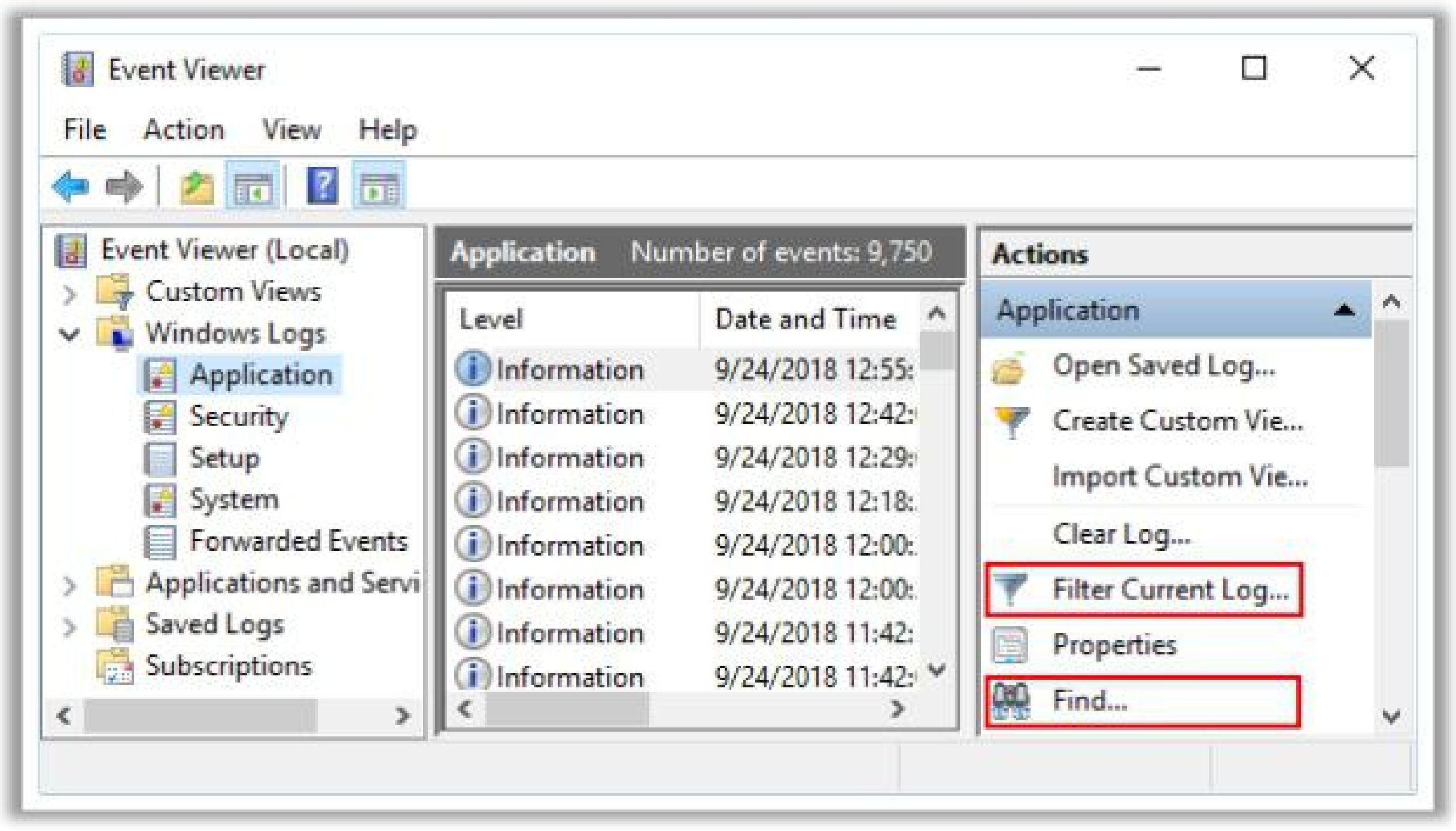
Successfully scheduled Software Protection service for re-start at 2118-08-31T04:56:34Z. Reas
RulesEngine.

Log Name:	Application		
Source:	Security-SPP	Logged:	9/24/2018 10:26:34 AM
Event ID:	16384	Task Category:	None
Level:	Information	Keywords:	Classic
User:	N/A	Computer:	DESKTOP-QH[redacted]
OpCode:	Info		
More Information:	Event Log Online Help		

Monitoring and Analysis of Windows Logs (Cont'd)

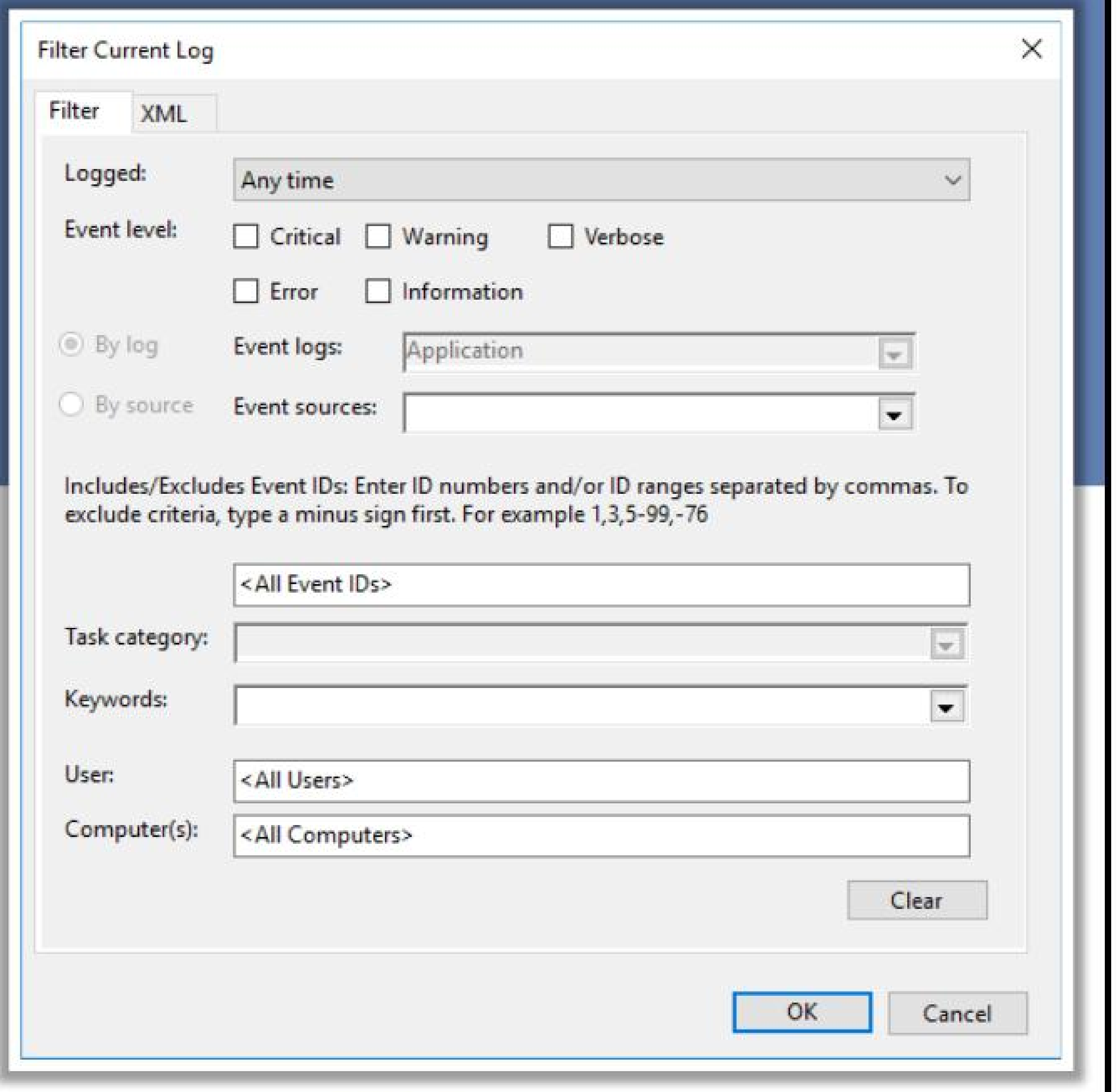
Filtering/Finding Events in Event Viewer

- ➢ The **Filter** feature in Event Viewer allows the removal of clutter from the event log display
- ➢ Each log can be independently configured with different filter properties
- ➢ Use **Filter** and **Find** features in Event Viewer, under the **Actions** pane
- ➢ After applying the filter, the Event Viewer shows the log with matching properties



Application Number of events: 9,750

Level	Date and Time
Information	9/24/2018 12:55:
Information	9/24/2018 12:42:
Information	9/24/2018 12:29:
Information	9/24/2018 12:18:
Information	9/24/2018 12:00:
Information	9/24/2018 12:00:
Information	9/24/2018 11:42:
Information	9/24/2018 11:42:



Logged: Any time

Event level: □ Critical □ Warning □ Verbose □ Error □ Information

○ By log Event logs: Application

○ By source Event sources:

Includes/Excludes Event IDs: Enter ID numbers and/or ID ranges separated by commas. To exclude criteria, type a minus sign first. For example 1,3,5-99,-76

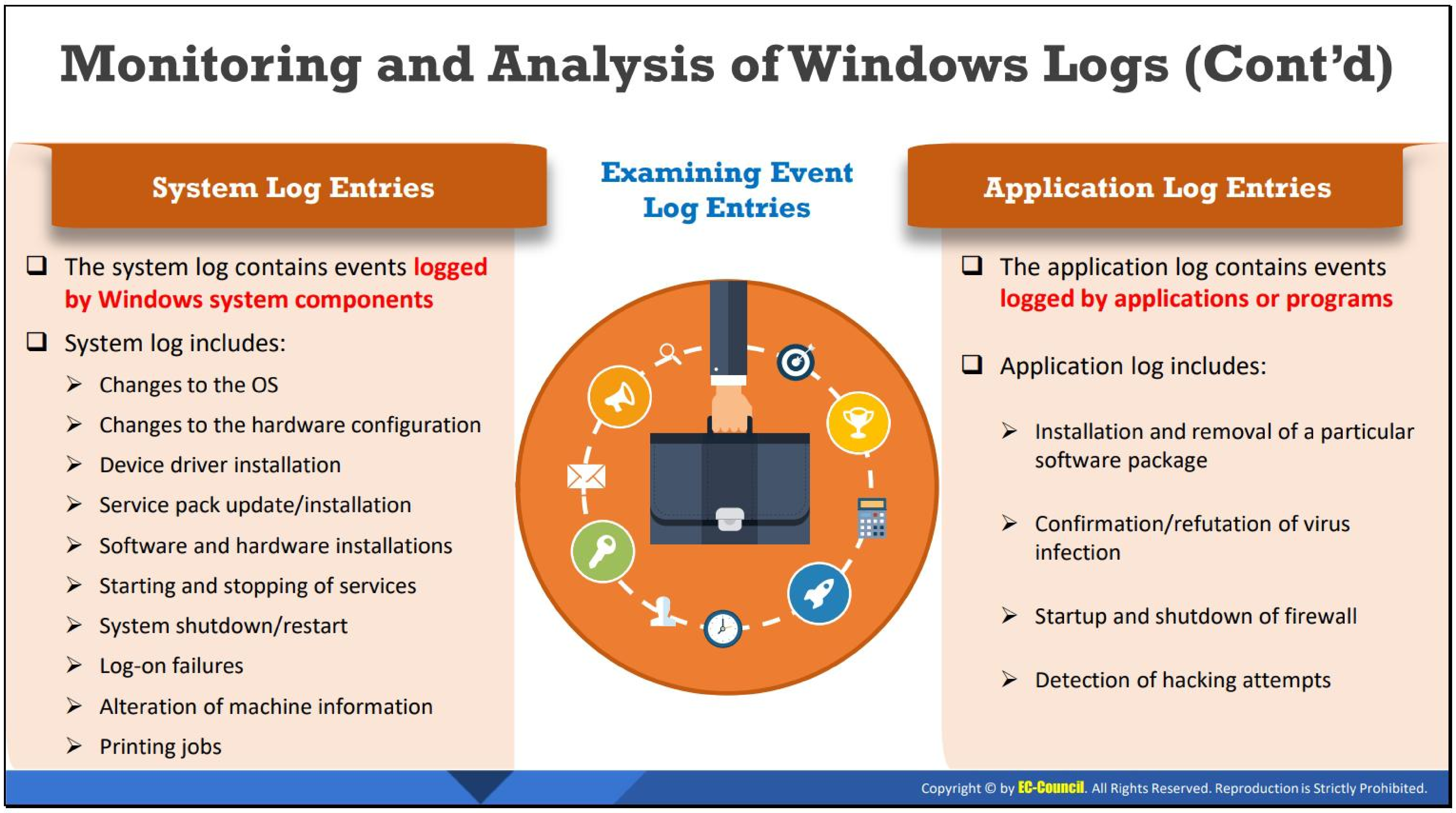
<All Event IDs>

Task category:

Keywords:

User: <All Users>

Computer(s): <All Computers>



Monitoring and Analysis of Windows Logs (Cont'd)

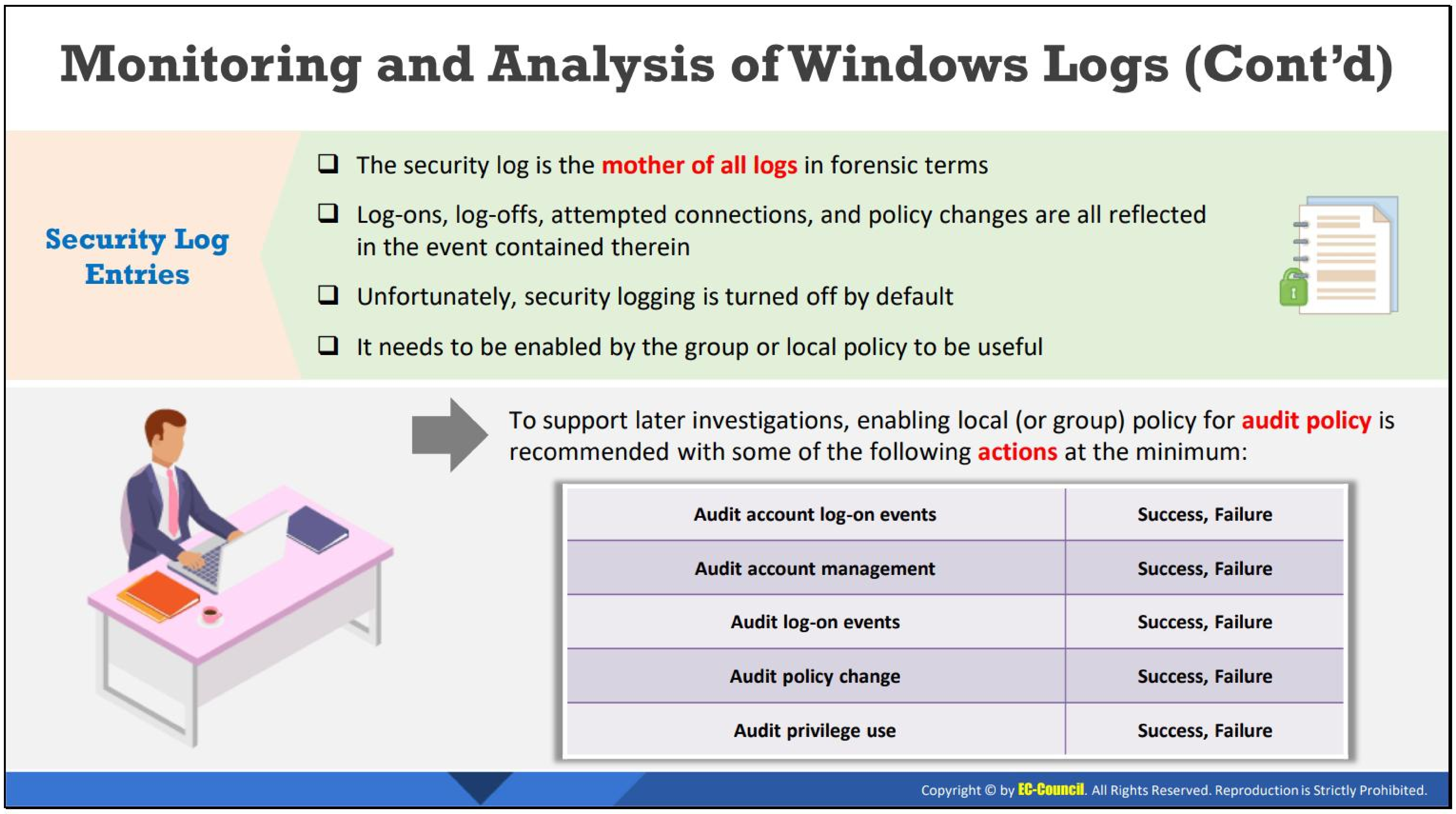
Examining Event Log Entries

System Log Entries

- ❑ The system log contains events **logged by Windows system components**
- ❑ System log includes:
  - ➢ Changes to the OS
  - ➢ Changes to the hardware configuration
  - ➢ Device driver installation
  - ➢ Service pack update/installation
  - ➢ Software and hardware installations
  - ➢ Starting and stopping of services
  - ➢ System shutdown/restart
  - ➢ Log-on failures
  - ➢ Alteration of machine information
  - ➢ Printing jobs

Application Log Entries

- ❑ The application log contains events **logged by applications or programs**
- ❑ Application log includes:
  - ➢ Installation and removal of a particular software package
  - ➢ Confirmation/refutation of virus infection
  - ➢ Startup and shutdown of firewall
  - ➢ Detection of hacking attempts



Monitoring and Analysis of Windows Logs (Cont'd)

Security Log Entries

- ❑ The security log is the **mother of all logs** in forensic terms
- ❑ Log-ons, log-offs, attempted connections, and policy changes are all reflected in the event contained therein
- ❑ Unfortunately, security logging is turned off by default
- ❑ It needs to be enabled by the group or local policy to be useful

To support later investigations, enabling local (or group) policy for **audit policy** is recommended with some of the following **actions** at the minimum:

Audit account log-on events	Success, Failure
Audit account management	Success, Failure
Audit log-on events	Success, Failure
Audit policy change	Success, Failure
Audit privilege use	Success, Failure

Monitoring and Analysis of Windows Logs

Windows event logs include critical information such as log-on failures, log tampering, failed attempts to access files, etc. They also warn regarding upcoming system issues and protect the system from unexpected disasters. In addition to this, these event logs may also describe an attempt made by a user to compromise the system or an unsanctioned configuration change. Thus, these event logs need to be monitored and analyzed to identify network vulnerabilities, security breaches, and threat intruders. These event logs enable security professionals to protect the network against internal threats and vulnerabilities. The most common way to monitor and analyze Windows event logs is to use the Windows Event Viewer.

Viewing Events in Event Viewer

- Open Windows Event Viewer by clicking the Start icon and then typing "Event Viewer" in the search box.

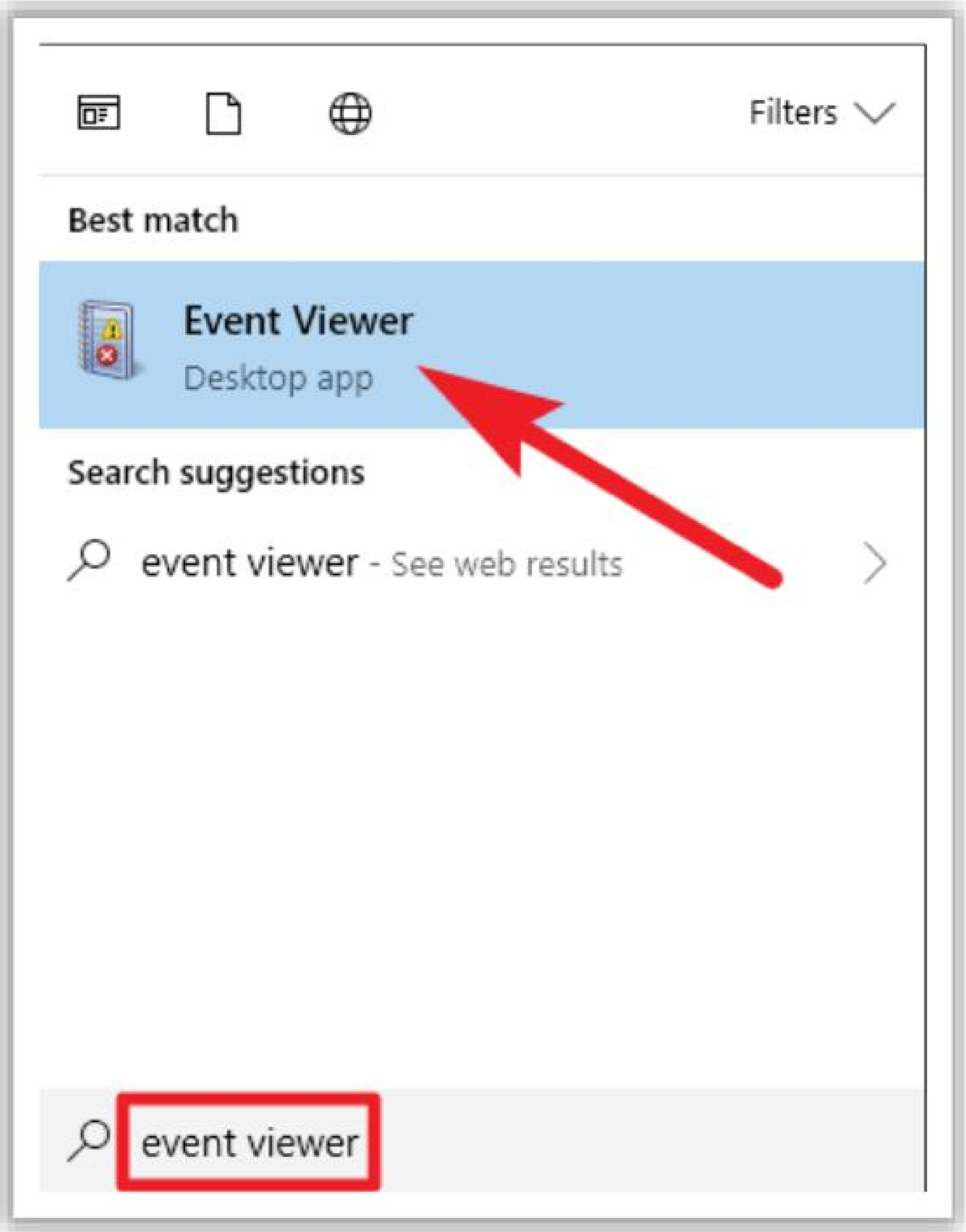
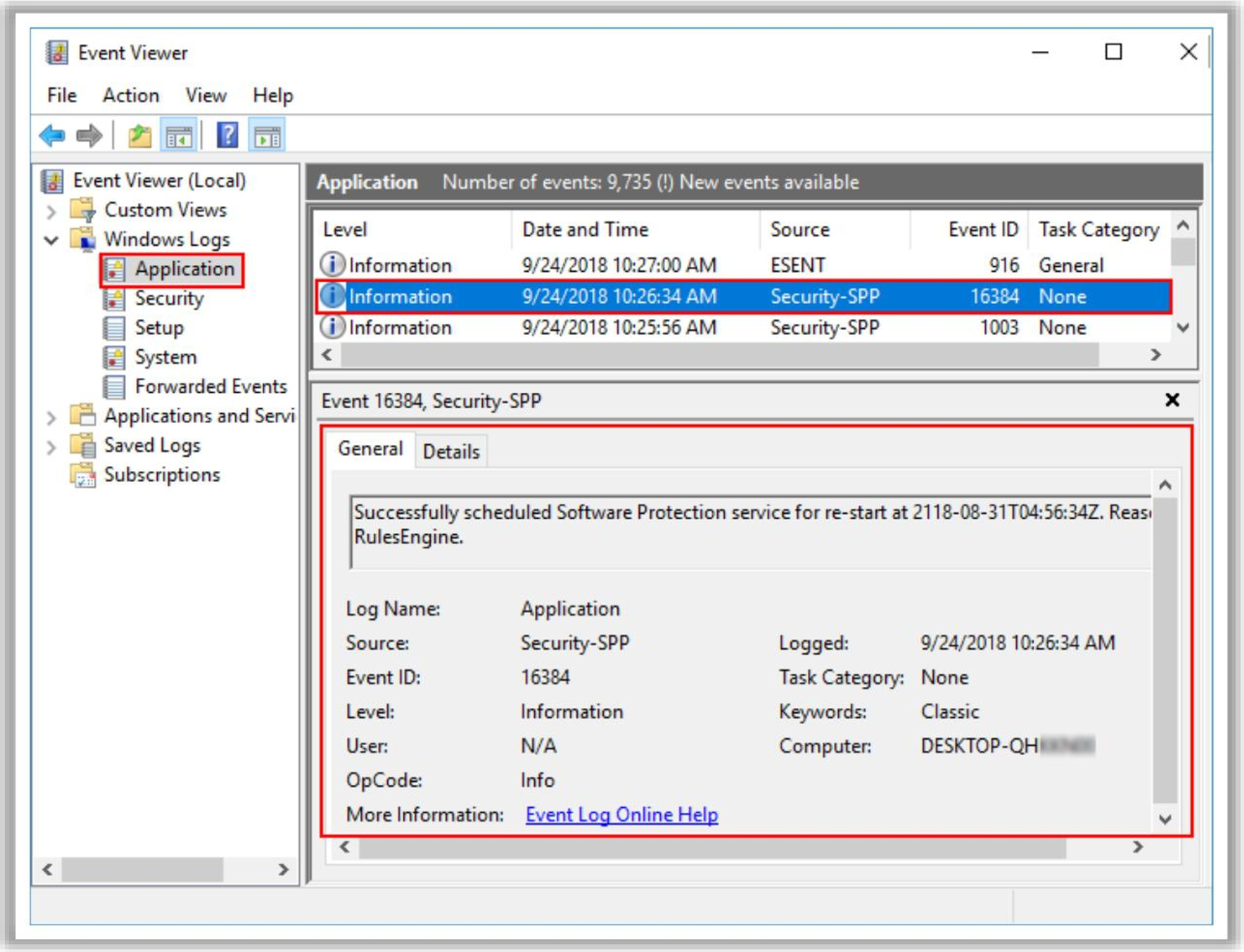


Figure 18.6: Screenshot of the Search Box

- Once Event Viewer opens, click on the required log file from the console tree. A list of events can be seen in the details pane.
- In the details pane, clicking on any specific event will reveal its description and header information in the Preview pane.

  The information displayed in the Preview pane about the event is described below.

  - **Log name:** The type of Windows log
  - **Source:** Source is the cause that is responsible for the event raised by either an individual or a system or a program
  - **Event ID:** The type of event that occurred
  - **Level:** Event level type is divided into five types: Error, Warning, Information, Success Audit, and Failure Audit
  - **User:** User responsible and who logged on the computer at the instance of the event
  - **Logged:** The timestamp of the event
  - **Task category:** Primarily used in case of a security log that classifies an event based on the event source
  - **Computer:** The name assigned to the computer where the event occurred



Application Number of events: 9,735 (!) New events available

Level	Date and Time	Source	Event ID	Task Category
Information	9/24/2018 10:27:00 AM	ESENT	916	General
Information	9/24/2018 10:26:34 AM	Security-SPP	16384	None
Information	9/24/2018 10:25:56 AM	Security-SPP	1003	None

Successfully scheduled Software Protection service for re-start at 2118-08-31T04:56:34Z. Reas
RulesEngine.

Log Name:	Application		
Source:	Security-SPP	Logged:	9/24/2018 10:26:34 AM
Event ID:	16384	Task Category:	None
Level:	Information	Keywords:	Classic
User:	N/A	Computer:	DESKTOP-QH[redacted]
OpCode:	Info		
More Information:	Event Log Online Help		

Figure 18.7: Screenshot of Event Viewer

Filtering/Finding Events in Event Viewer

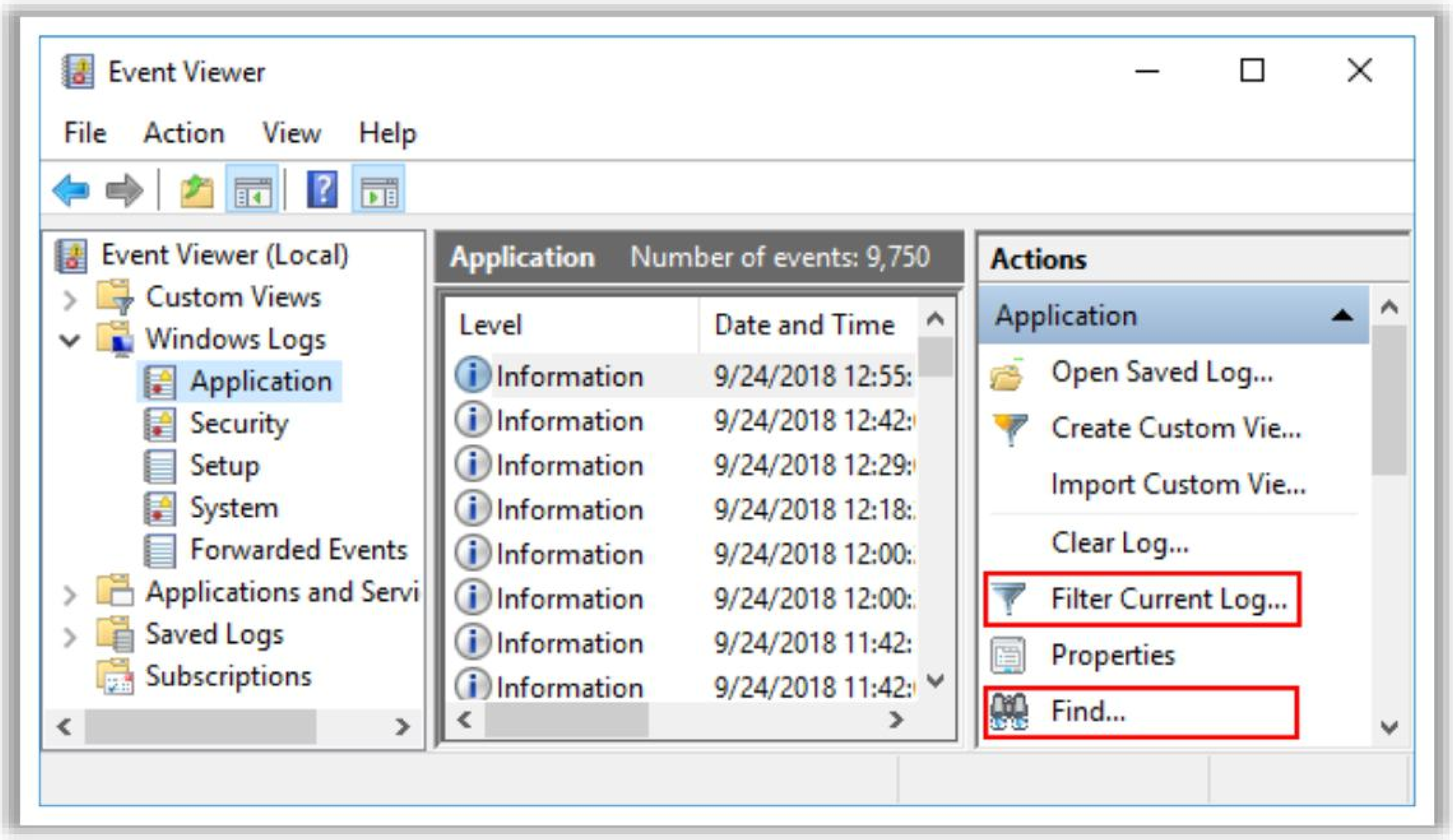
The Filter feature in Event Viewer helps in targeting the information that may be required for investigation. To save time and effort, Event Viewer provides the option to save specific filters for future use through the Create Custom View feature.

Filter feature can allow the removal of clutter from the event log display and limit the data displayed in a single log. Each log can be independently configured with different filter properties.

The following steps are used to create a filter:

- Select the log that needs to be filtered.
- After that, click on "**Filter Current Log**" option available under the **Action** pane.
- The "**Filter Current Log**” dialog box will appear.
- Specify a time period, if the approximate time when the events occurred is known.
- The event levels can be specified from the available options (Critical, Warning, Verbose, Error, and Information). If no option is specified, all event levels will be returned.
- Specific event IDs can be mentioned in the defined format.
- Specific event sources can be selected; similarly, specific keywords, users, or computers can be searched.

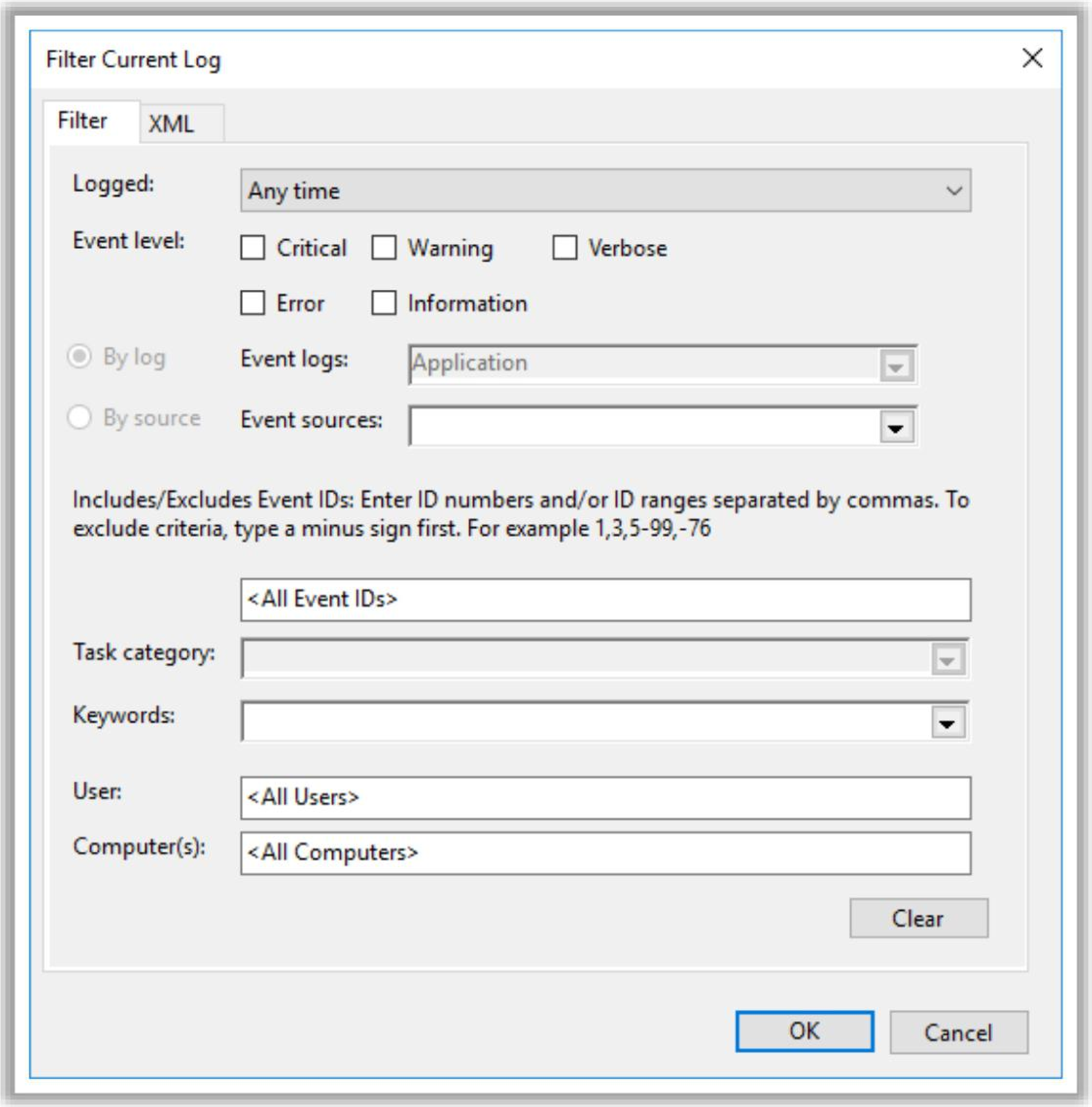
- Click **OK** to close the "**Filter Current log**" dialog box.
- After applying the filter, the Event Viewer will show the log with matching properties.



Application Number of events: 9,750

Level	Date and Time
Information	9/24/2018 12:55:
Information	9/24/2018 12:42:
Information	9/24/2018 12:29:
Information	9/24/2018 12:18:
Information	9/24/2018 12:00:
Information	9/24/2018 12:00:
Information	9/24/2018 11:42:
Information	9/24/2018 11:42:

Figure 18.8: Screenshot of Event Viewer



Logged: Any time

Event level: □ Critical □ Warning □ Verbose □ Error □ Information

● By log Event logs: Application

○ By source Event sources:

Includes/Excludes Event IDs: Enter ID numbers and/or ID ranges separated by commas. To exclude criteria, type a minus sign first. For example 1,3,5-99,-76

<All Event IDs>

Task category:

Keywords:

User: <All Users>

Computer(s): <All Computers>

Figure 18.9: Screenshot of “Filter Current Log” Dialog Box

To find an event, follow the steps below:

- Click on "Find" option available under the Action pane
- Type the information that needs to be found and then click Find Next
- Click Close, when search is complete

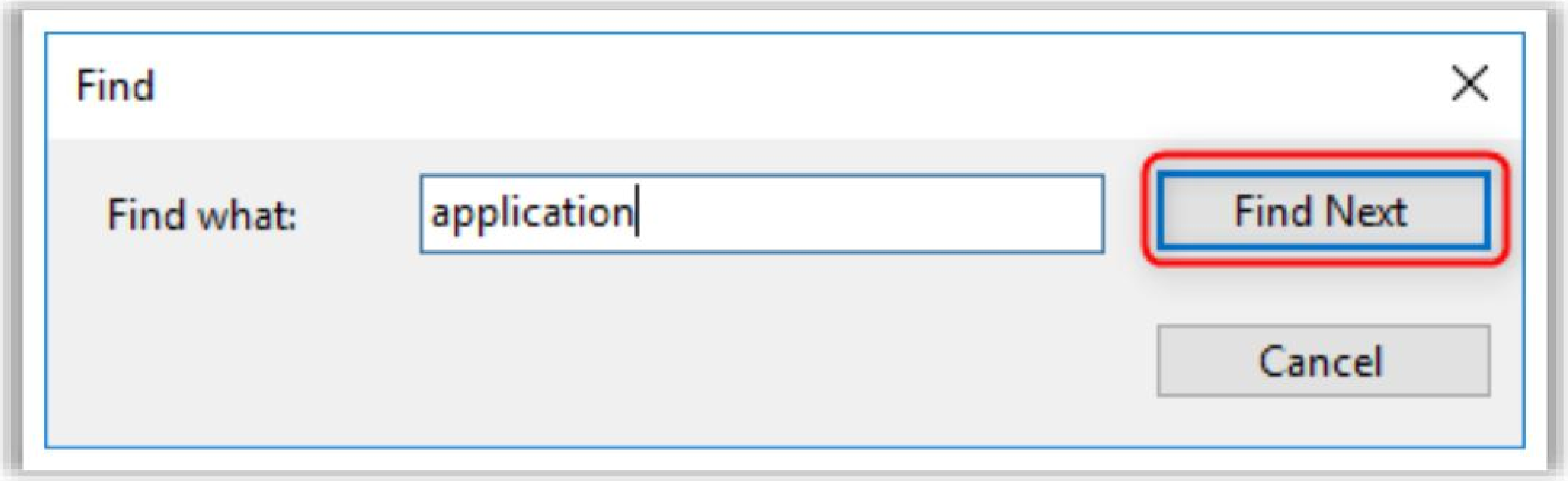


Figure 18.10: Screenshot of Find dialog box

Examining Event Log Entries

Event Viewer displays three types of event log entries, as described below.

- **System log entries**

  The system log contains events logged by Windows system components. It contains information about system changes such as device driver installations, etc. To view system log entries in Event Viewer:

  - Open Event Viewer and then select System log from Windows logs section in the console tree
  - A list of system events appears in the details pane
  - Select the specific event whose details needs to be viewed

  Examples of system log records:

  - Changes to the OS
  - Changes to the hardware configuration
  - Device driver installation
  - Service pack update/installation
  - Software and hardware installations
  - Start/stop of services
  - System shutdown/restart
  - Log-on failures
  - Alteration of machine information
  - Printing jobs

- **Application log entries**

  The application log contains events logged by applications or programs. To view application log entries in Event Viewer:

  - Open Event Viewer and then select Application log from Windows logs section in the console tree
  - A list of application events will appear in the details pane
  - Select the specific event whose details need to be viewed

  Examples of application log records:

  - Installation and removal of a particular software package
  - Confirmation/refutation of virus infection
  - Startup and shutdown of firewall
  - Detection of hacking attempts

- **Security log entries**

  The security log is the mother of all logs in forensic terms. Unfortunately, security logging is turned off by default. To view security log entries in Event Viewer,

  - Open Event Viewer and then select Security log from Windows logs section in the console tree
  - A list of security events appear in the details pane
  - Select the specific event whose details need to be viewed

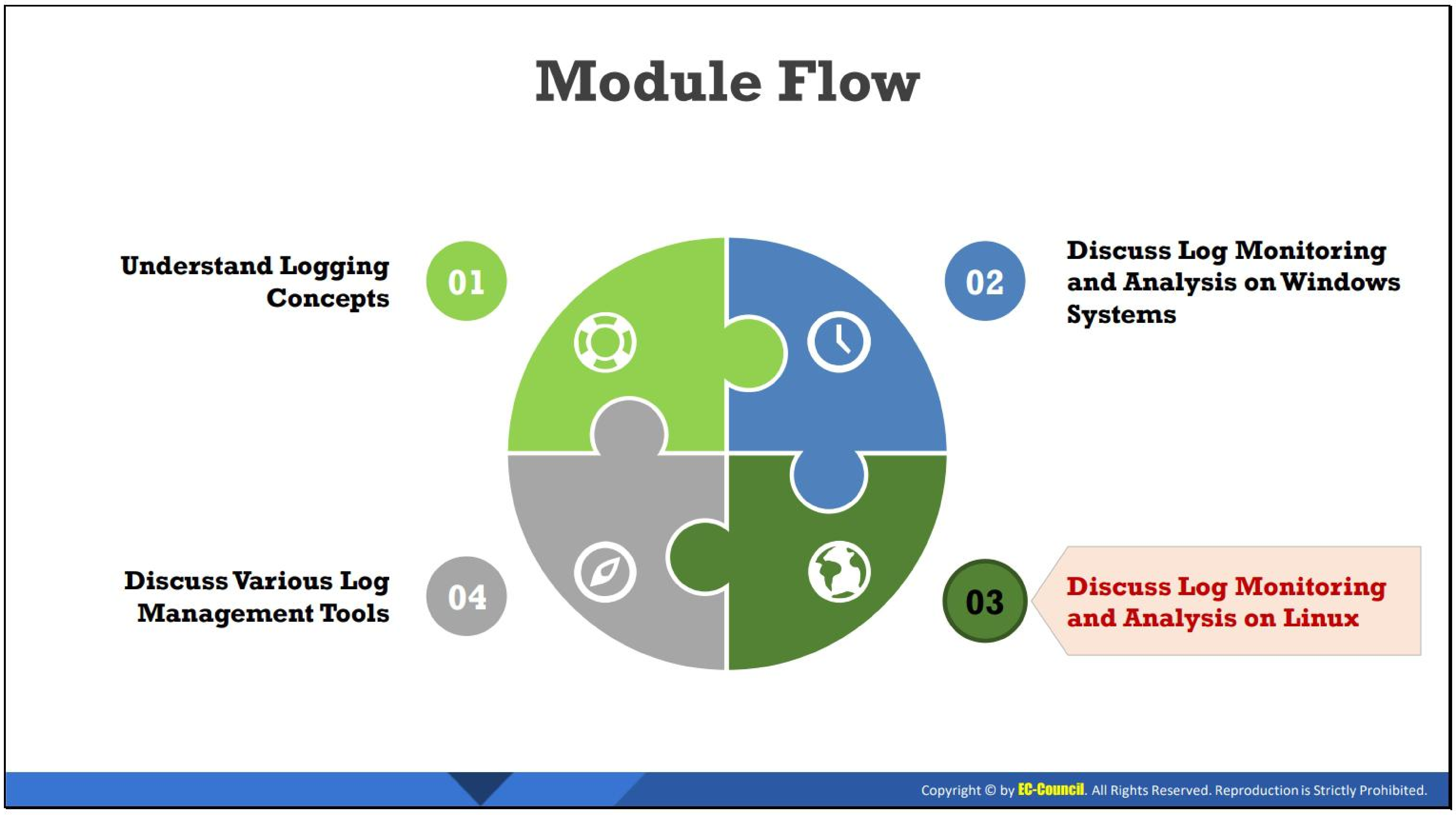
  Examples of security log records:

  - Log-ons
  - Log-offs
  - Attempted connections
  - Policy changes

To support later investigations, enabling local (or group) policy for audit policy is recommended with some of the following actions at the minimum:

Audit account log-on events	Success, Failure
Audit account management	Success, Failure
Audit log-on events	Success, Failure
Audit policy change	Success, Failure
Audit privilege use	Success, Failure

Table 18.1: Actions to Enable Local (or Group) Policy for an Audit Policy



Module Flow

01 Understand Logging Concepts

02 Discuss Log Monitoring and Analysis on Windows Systems

03 Discuss Log Monitoring and Analysis on Linux

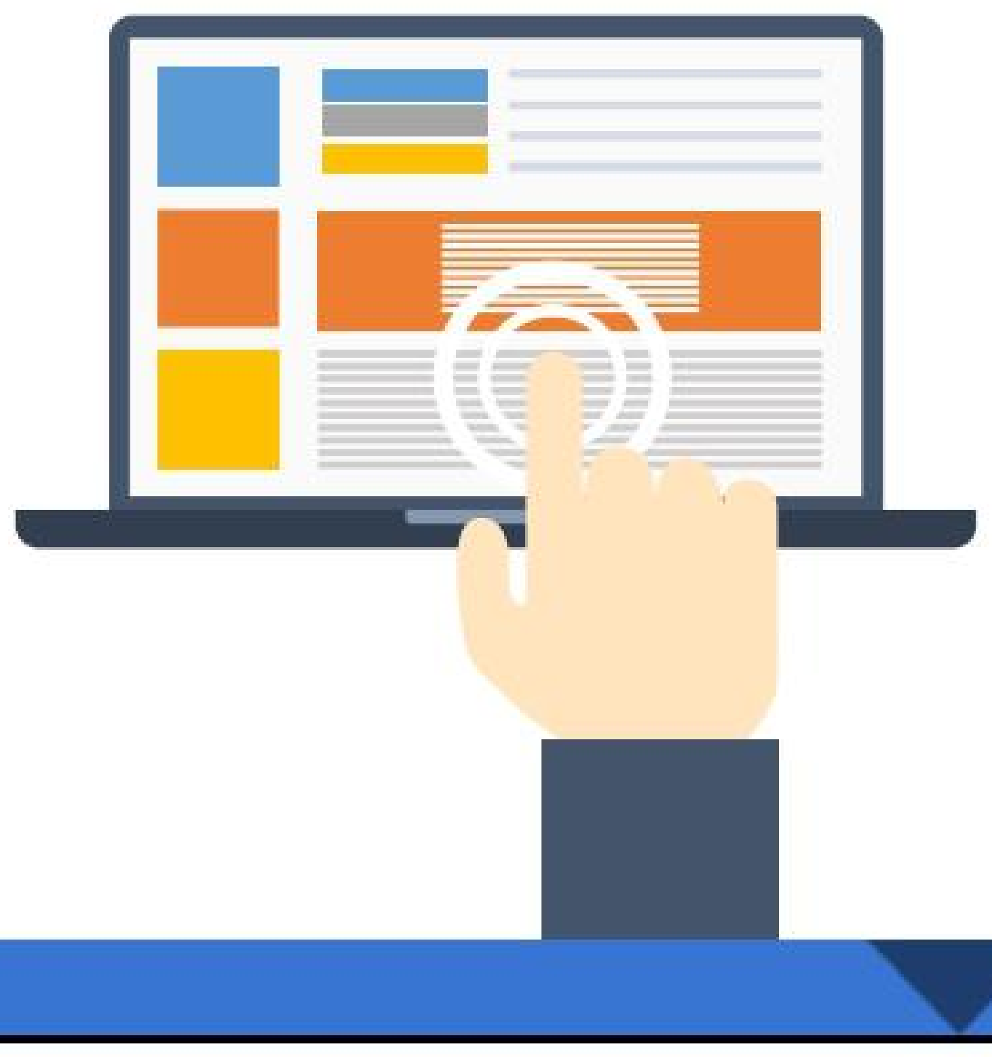
04 Discuss Various Log Management Tools

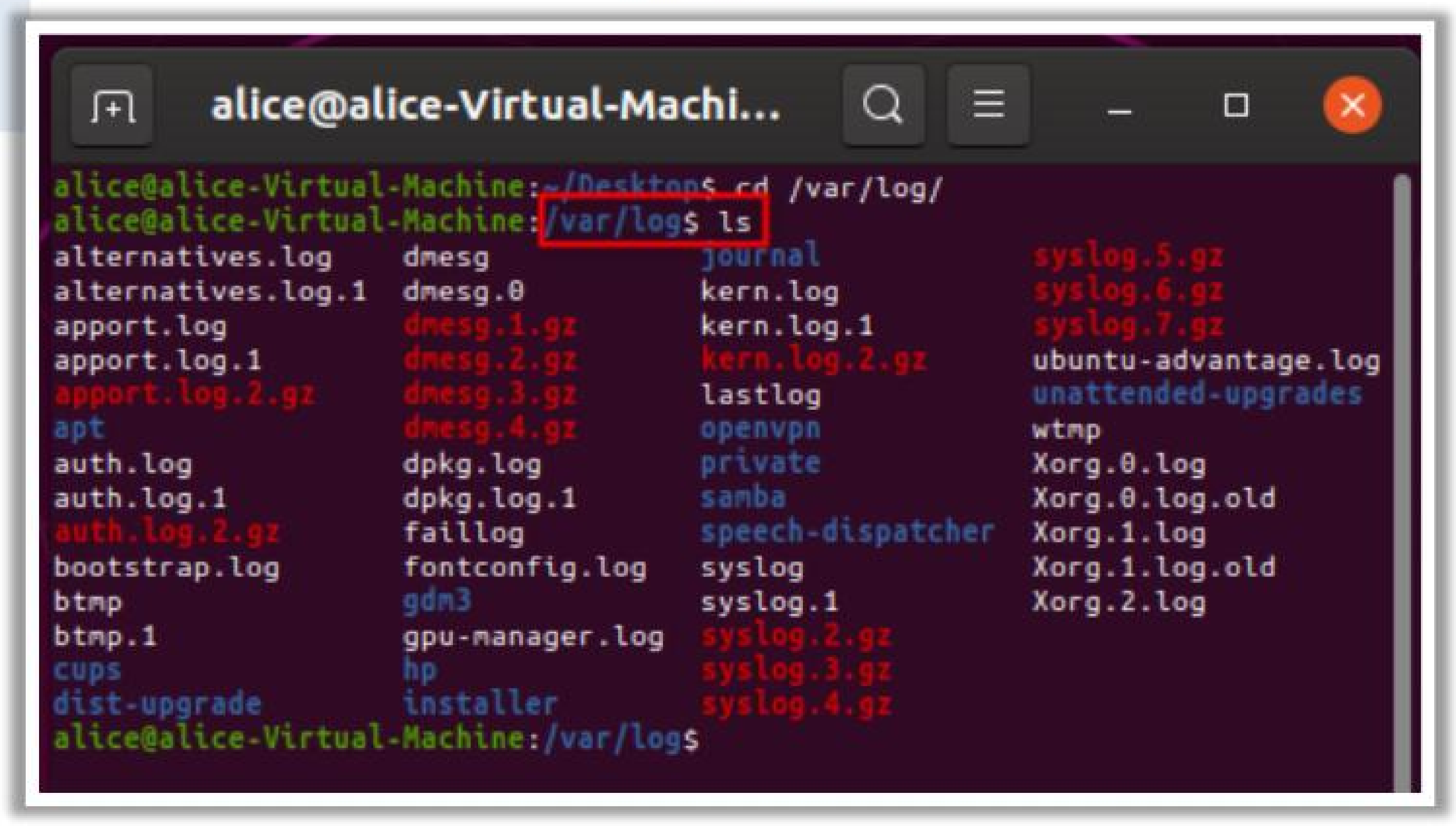
Discuss Log Monitoring and Analysis on Linux

The objective of this section is to explain monitoring and analysis of logs in Linux-based systems. It describes Linux logs, the various Linux log files, and commands to monitor and analyze Linux logs.

Linux Logs

- Linux logs are a **record of any activity** or **event** in Linux OS
- Most Linux logs are located at **/var/log directory** in plain ASCII text format
- System log daemon (syslogd) produces logs for the system and different programs in Linux OS



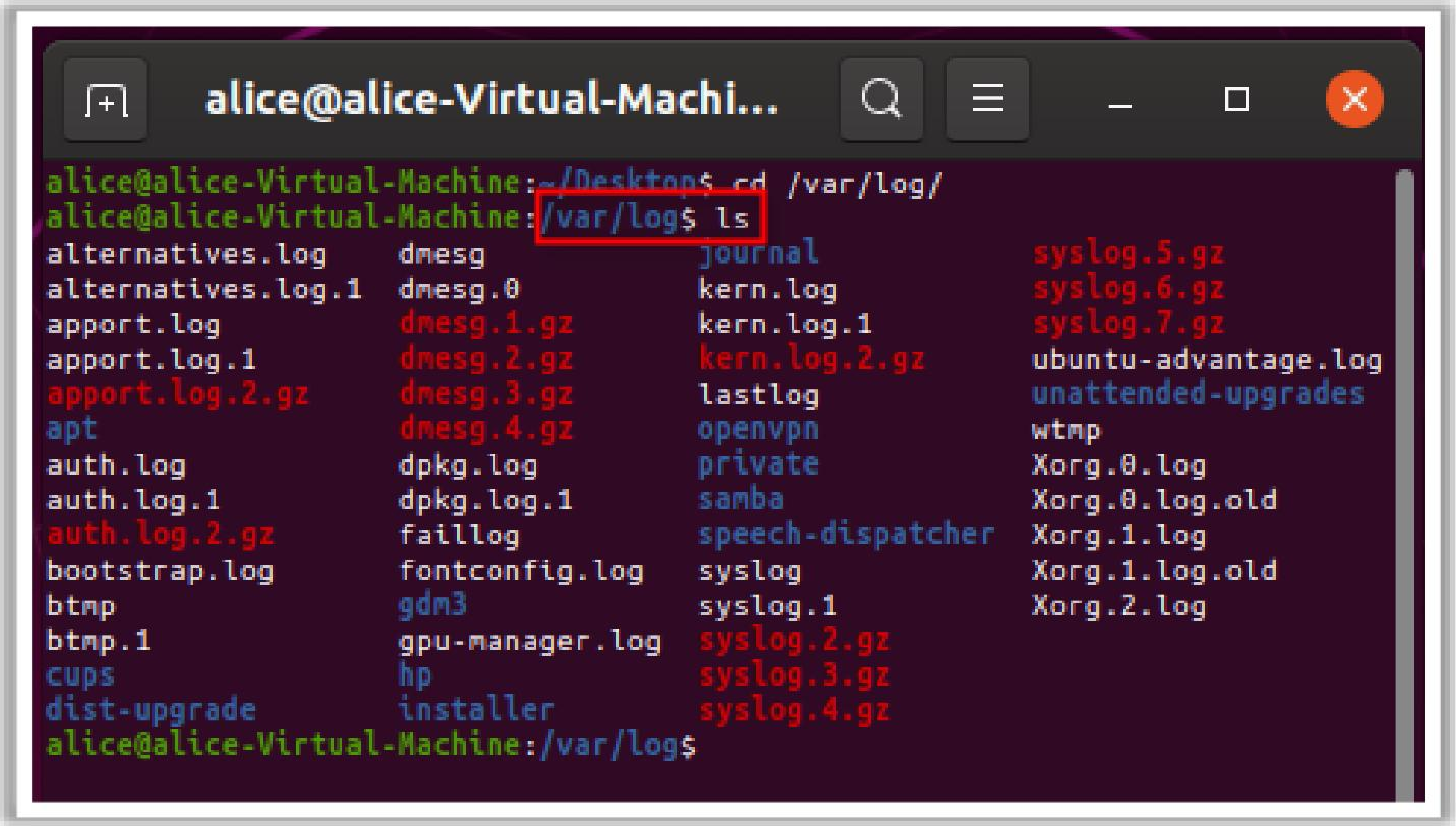


```
alice@alice-Virtual-Machine:~/Desktop$ cd /var/log/
alice@alice-Virtual-Machine:/var/log$ ls
alternatives.log    dmesg            journal             syslog.5.gz
alternatives.log.1  dmesg.0          kern.log            syslog.6.gz
apport.log          dmesg.1.gz       kern.log.1          syslog.7.gz
apport.log.1        dmesg.2.gz       kern.log.2.gz       ubuntu-advantage.log
apport.log.2.gz     dmesg.3.gz       lastlog             unattended-upgrades
apt                 dmesg.4.gz       openvpn             wtmp
auth.log            dpkg.log         private             Xorg.0.log
auth.log.1          dpkg.log.1       samba               Xorg.0.log.old
auth.log.2.gz       faillog          speech-dispatcher   Xorg.1.log
bootstrap.log       fontconfig.log   syslog              Xorg.1.log.old
btmp                gdm3             syslog.1            Xorg.2.log
btmp.1              gpu-manager.log  syslog.2.gz
cups                hp               syslog.3.gz
dist-upgrade        installer        syslog.4.gz
alice@alice-Virtual-Machine:/var/log$
```

Linux Logs

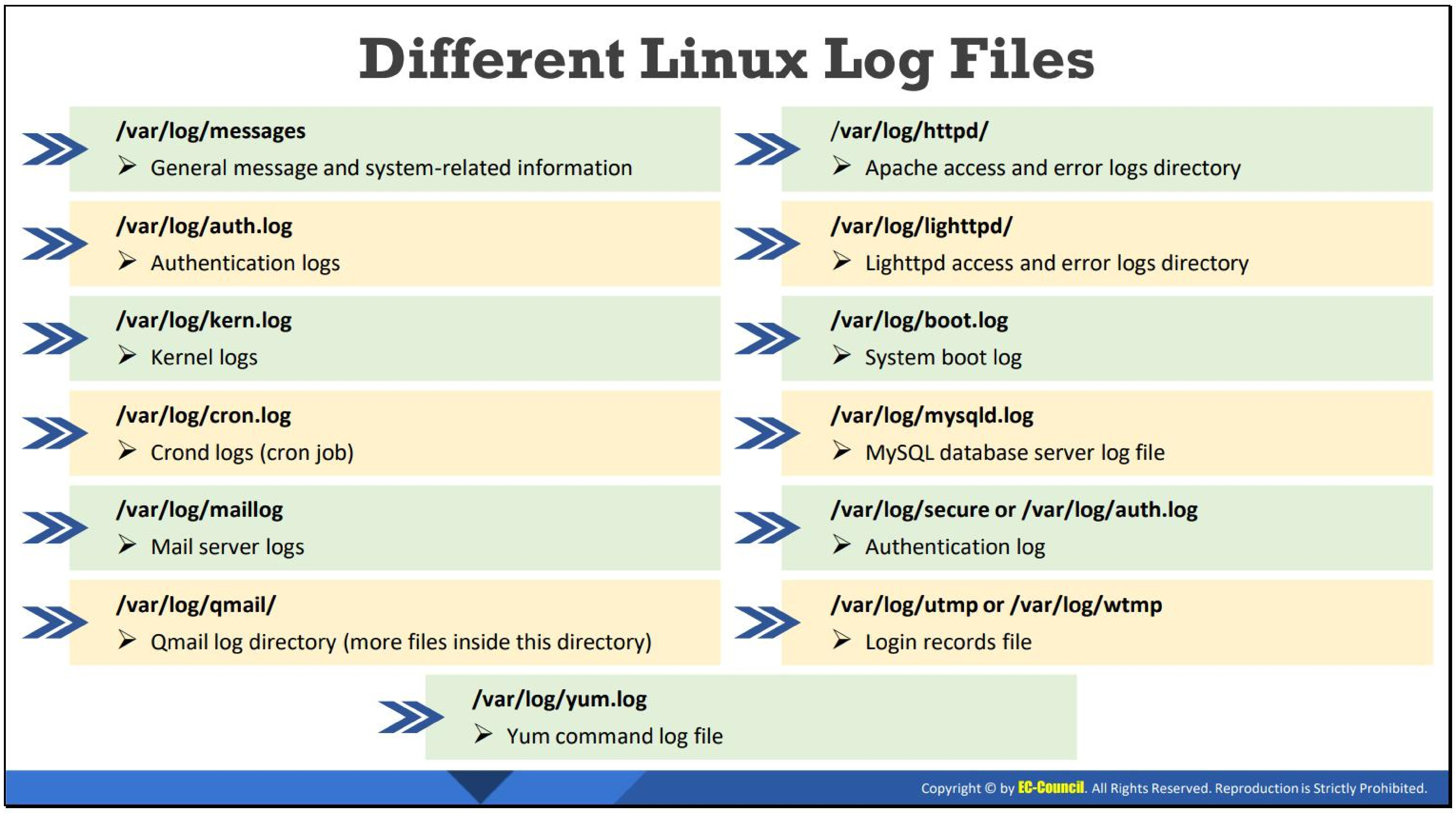
Linux logs are a record of any activity or event in a Linux-based OS (hereinafter "Linux OS"); they include messages on just about everything, including system, kernel, package managers, boot processes, Xorg, Apache, and MySQL. These log files are a useful troubleshooting tool when any security issue occurs. These files help in monitoring and analyzing security threats and vulnerabilities as well as remediate them as soon as possible. They also help in tracking the communication between one system with another system and networks.

Linux OS captures a wide range of information using multiple log files. Most logs are located in the **/var/log** directory and subdirectory in plain ASCII text format. These are system and service log files that provide information about OS-specific issues or service-specific issues. Many of them are produced by the system log daemon (**syslogd**) on behalf of the system and application whereas some applications produce logs directly into **/var/log** directory. To change the directory, the **cd** command is used. However, only the root user can view or access Linux log files.



```
alice@alice-Virtual-Machine:~/Desktop$ cd /var/log/
alice@alice-Virtual-Machine:/var/log$ ls
alternatives.log    dmesg            journal            syslog.5.gz
alternatives.log.1  dmesg.0          kern.log           syslog.6.gz
apport.log          dmesg.1.gz       kern.log.1         syslog.7.gz
apport.log.1        dmesg.2.gz       kern.log.2.gz      ubuntu-advantage.log
apport.log.2.gz     dmesg.3.gz       lastlog            unattended-upgrades
apt                 dmesg.4.gz       openvpn            wtmp
auth.log            dpkg.log         private            Xorg.0.log
auth.log.1          dpkg.log.1       samba              Xorg.0.log.old
auth.log.2.gz       faillog          speech-dispatcher  Xorg.1.log
bootstrap.log       fontconfig.log   syslog             Xorg.1.log.old
btmp                gdm3             syslog.1           Xorg.2.log
btmp.1              gpu-manager.log  syslog.2.gz
cups                hp               syslog.3.gz
dist-upgrade        installer        syslog.4.gz
alice@alice-Virtual-Machine:/var/log$
```

Figure 18.11: Linux log



Different Linux Log Files

- **/var/log/messages**
  - ➢ General message and system-related information
- **/var/log/auth.log**
  - ➢ Authentication logs
- **/var/log/kern.log**
  - ➢ Kernel logs
- **/var/log/cron.log**
  - ➢ Crond logs (cron job)
- **/var/log/maillog**
  - ➢ Mail server logs
- **/var/log/qmail/**
  - ➢ Qmail log directory (more files inside this directory)
- **/var/log/httpd/**
  - ➢ Apache access and error logs directory
- **/var/log/lighttpd/**
  - ➢ Lighttpd access and error logs directory
- **/var/log/boot.log**
  - ➢ System boot log
- **/var/log/mysqld.log**
  - ➢ MySQL database server log file
- **/var/log/secure or /var/log/auth.log**
  - ➢ Authentication log
- **/var/log/utmp or /var/log/wtmp**
  - ➢ Login records file
- **/var/log/yum.log**
  - ➢ Yum command log file

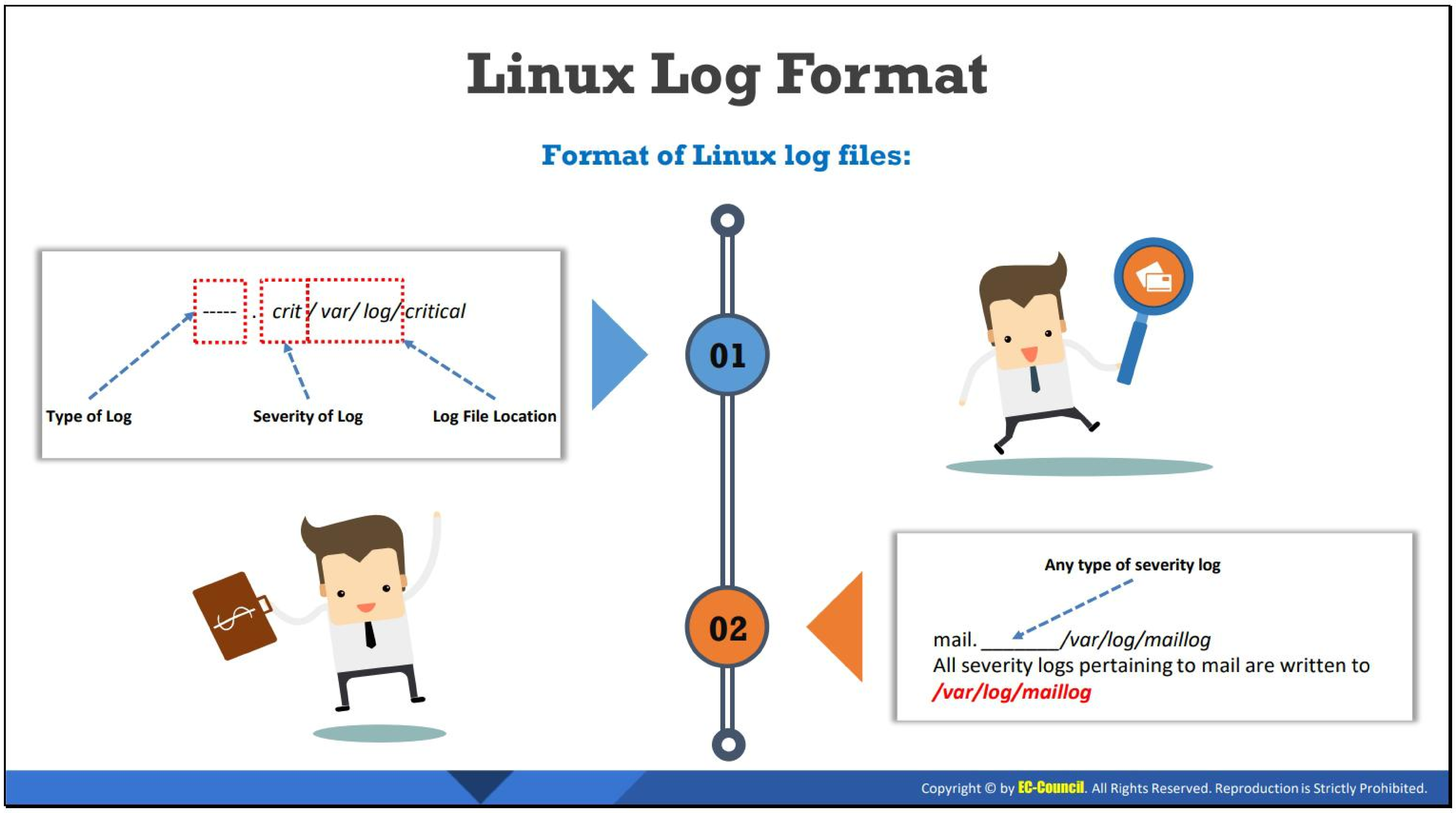
Different Linux Log Files

Linux OS generates four different categories of log files: application logs, event logs, service logs, and system logs. These log files should be monitored to predict upcoming issues before they actually occur. However, it can get cumbersome to monitor and analyze all log files or to determine which file contains the required information. Therefore, to make it the process a little simpler, a few critical Linux log files are introduced here that should be monitored effectively to gather all essential information.

- **`/var/log/messages`** or **`/var/log/syslog`**: This log file contains general messages and system-related information. It stores all informational and noncritical messages across the global system such as system error messages, system startups, and shutdowns, change in the network configuration, etc. It can also log several things such as mail, cron, daemon, kern, auth, etc. This is the first place to look if things go wrong in the network/OS. For example, if there is any issue with the sound card, then you have to check the messages logged in this file. This file stores data in plain-text format that can be checked by any tool that can examine text files.
- **`/var/log/auth.log`** or **`/var/log/secure`**: This log file contains authentication logs, including both successful and unsuccessful user login attempts as well as authentication techniques. This file is beneficial if you want to examine brute-force attacks and other vulnerabilities related to user authorization mechanism.
- **`/var/log/kern.log`**: This file stores information that is logged by the kernel. It is helpful in solving kernel-related errors and warnings as well as hardware and connectivity problems. It is also useful in troubleshooting a custom-built kernel.
- **`/var/log/cron.log`**: This file contains information about all Crond-related messages (cron jobs). For example, when the cron daemon begins the cron job, all related

information about successful or failed execution is logged on to this file. This file is helpful for solving issues with scheduled cron.

- **`/var/log/maillog`** or **`/var/log/mail.log:`** This file stores information related to mail servers. This file is useful when checking information regarding postfix, smtpd, MailScanner, and other email-related services. It keeps records of all emails that are sent or received within a time zone. In addition, it helps to examine failed delivery problems and detecting spamming attempts blocked by the mail server.
- **`/var/log/qmail/:`** It is a directory that stores information related to qmail logs. This directory is helpful when trying to track all emails sent through a qmail system, if the list of every message transmitted by the server is needed, or the number of messages processed needs to be determined.
- **`/var/log/httpd/:`** It is a directory that stores information related to the Apache web server. Apache web server stores information in two log files: access_log and error_log. This directory provides detailed information about events and errors raised during processing httpd requests. It keeps records of every page or file that is provided or loaded by Apache and also stores the IP address and user ID of every client that made a connection to the server. It also logs the status of access requests and whether a response was given or not.
- **`/var/log/lighttpd/:`** It is a directory that stores information related to light HTTPD access_log and error_log.
- **`/var/log/boot.log:`** This file stores all information related to system booting. The booting messages are sent by system initialization script, **`/etc/init.d/bootmisc.sh`**, to this log file. This file is helpful when trying to troubleshoot problems related to improper shutdowns, booting failures, or unplanned reboots. By checking this file, the time span of system downtime that occurred due to an unexpected shutdown can be determined.
- **`/var/log/mysqld.log:`** This file stores all debug, failure, and success messages about [mysqld] and [mysqld_safe] daemon. It is helpful when trying to detect issues related to starting, running, and stopping of mysqld.
- **`/var/log/utmp`** or **`/var/log/wtmp:`** This file stores information related to user login/logout, and it is helpful when trying to determine the current login state.
- **`/var/log/yum.log:`** All information related to installation of a package using `yum` command is stored in this file, which proves useful when trying to check whether a package is installed correctly or not; it also helps in identifying and solving software installation issues.



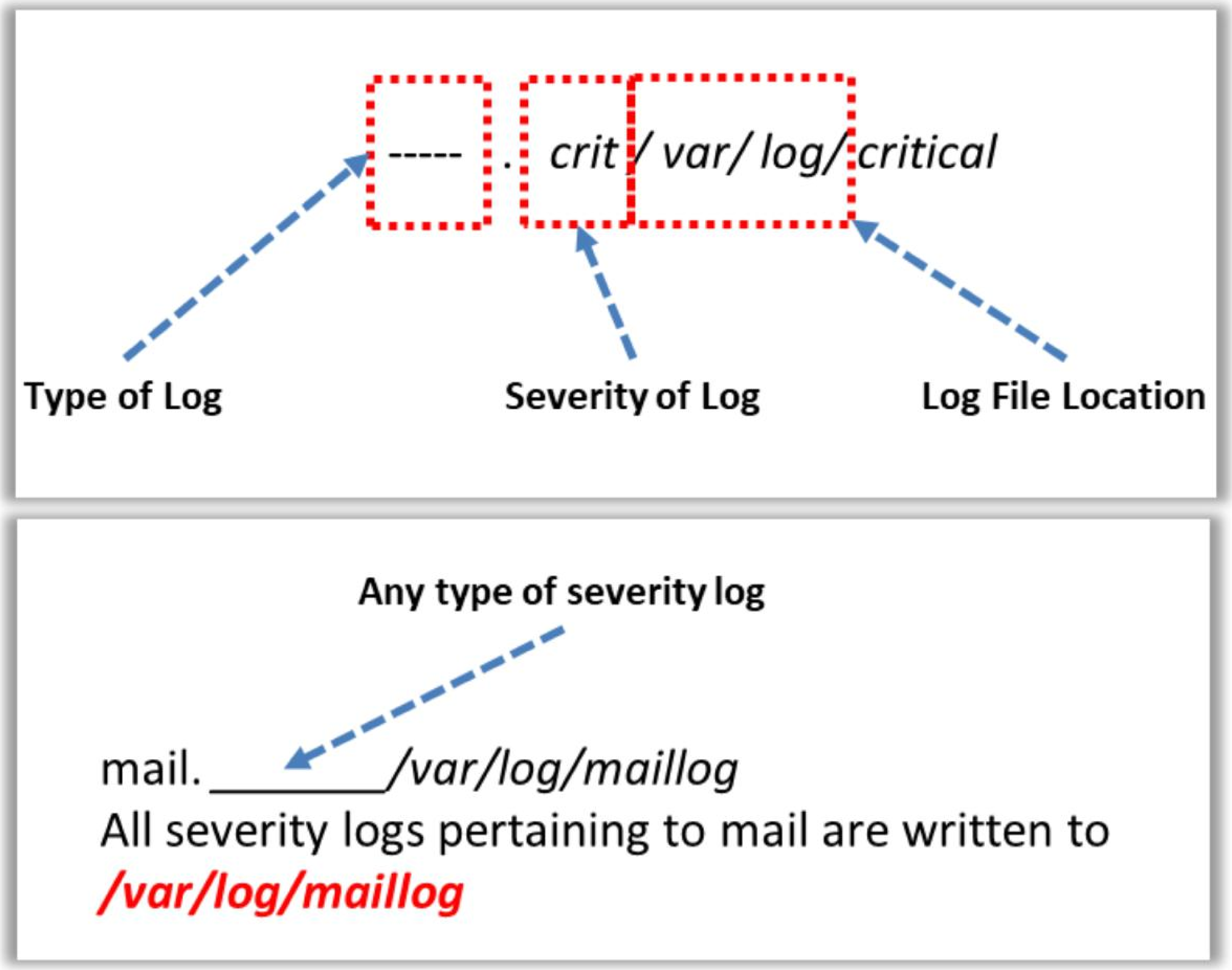
Linux Log Format

Format of Linux log files:

All severity logs pertaining to mail are written to ***/var/log/maillog***

Linux Log Format

The system log file provides information about where messages are logged. It is in the following format:



All severity logs pertaining to mail are written to ***/var/log/maillog***

Figure 18.12: Linux log format

Consider some examples below to understand the log file format.

- **Example 1:** Log all kernel messages to the console

  ```
  kern.* /dev/console
  ```

- **Example 2:** Log anything (except mail) of level info or higher. Do not log private authentication messages

  ```
  *.info;mail.none;news.none;authpriv.none;cron.none
  /var/log/messages
  ```

- **Example 3:** The authpriv file has restricted access

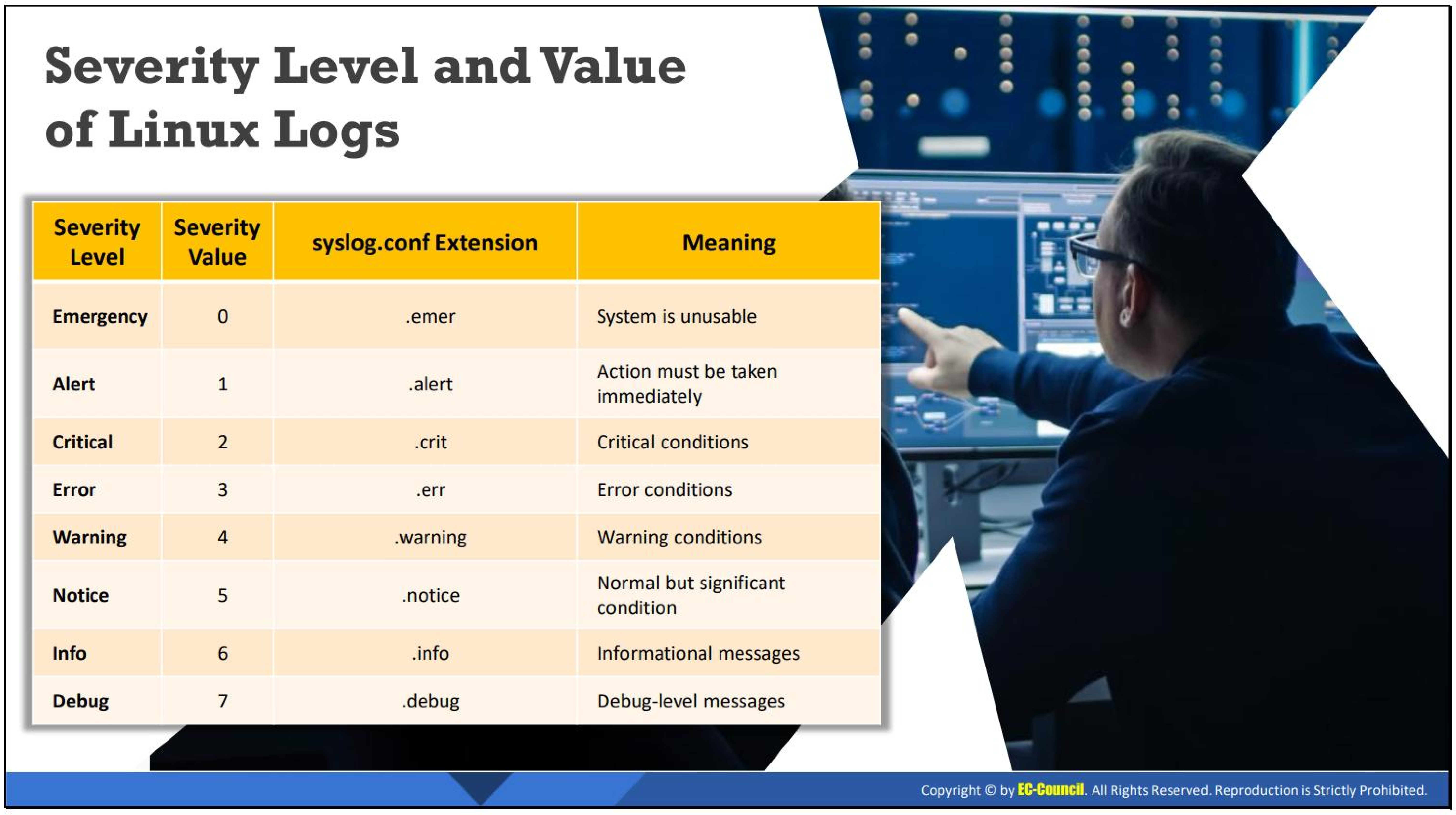
  ```
  authpriv.* /var/log/secure
  ```

- **Example 4**: Log all the mail messages in one place

  ```
  mail.* /var/log/maillog
  ```

Each line of the log file is divided into two portions: message selector and action field.

- Message selector represents the type of message to log. It is a combination of log type and severity level. In the above example, **`kern.*`**, **`*.info;mail.none;news.none;authpriv.none;cron.none`**, **`authpriv.*`**, and **`mail.*`** are the various selectors. Here, * indicates "all" such as **`kern.*`** all messages generated by the kernel.
- An action field describes the type of action to be applied to the message. It indicates a log file location. In the above examples, **`;/dev/console`**, **`/var/log/messages`**, **`/var/log/secure`**, **`/var/log/maillog`** represent the actions.



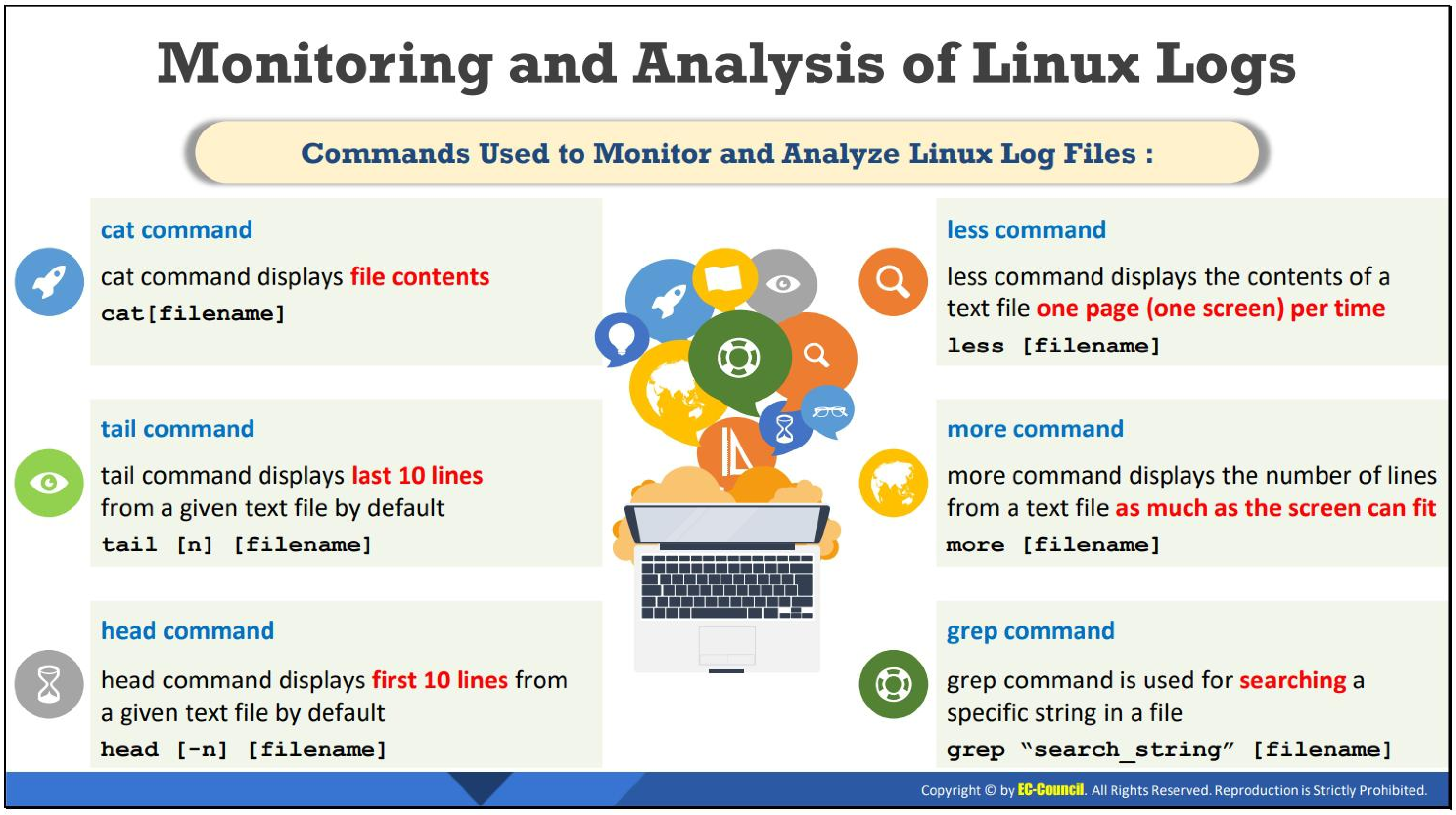
Severity Level and Value of Linux Logs

Severity Level	Severity Value	syslog.conf Extension	Meaning
Emergency	0	.emer	System is unusable
Alert	1	.alert	Action must be taken immediately
Critical	2	.crit	Critical conditions
Error	3	.err	Error conditions
Warning	4	.warning	Warning conditions
Notice	5	.notice	Normal but significant condition
Info	6	.info	Informational messages
Debug	7	.debug	Debug-level messages

Severity Level and Value of Linux Logs

The combination of Linux log file(s) and severity levels facilitates determination of what is logged and where that information is stored. When a system logger receives messages from multiple programs, it will make decisions as to what to keep and what to discard on the basis of severity levels defined by the selector. There are eight severity levels for sending a message in Linux, starting from level 0 to level 7. The highest severe message is at level 0, and the lowest severe message is at level 7.

- **Level 0—Emergency:** This level represents emergency conditions where the system comes unusable; for example, imminent system crash.
- **Level 1—Alert:** This level represents those conditions that require immediate actions; for example, a corrupted system database.
- **Level 2—Critical:** This level represents critical conditions such as a hardware error.
- **Level 3—Error:** This level represents error messages.
- **Level 4—Warning:** This level represents warning messages.
- **Level 5—Notice:** This level represents those messages that are not an error but require special attention.
- **Level 6—Information:** This level represents informational messages.
- **Level 7—Debug:** This level represents those messages that are required during debugging programs.



Monitoring and Analysis of Linux Logs

Commands Used to Monitor and Analyze Linux Log Files :

cat command

cat command displays **file contents**

`cat[filename]`

tail command

tail command displays **last 10 lines** from a given text file by default

`tail [n] [filename]`

head command

head command displays **first 10 lines** from a given text file by default

`head [-n] [filename]`

less command

less command displays the contents of a text file **one page (one screen) per time**

`less [filename]`

more command

more command displays the number of lines from a text file **as much as the screen can fit**

`more [filename]`

grep command

grep command is used for **searching** a specific string in a file

`grep "search_string" [filename]`

Monitoring and Analysis of Linux Logs

Monitoring and analysis of Linux logs helps determine security issues before they can significantly harm the system. Various types of commands are provided by Linux to monitor and analyze log files. Some of them are described below.

- **`cat` command:** cat stands for concatenate. It is one of the most important commands used in Linux OS. It reads data from the file and displays its content. It can combine the contents of two files by appending the content of the second file to the end of the first file. It can also copy the content of one file to another file. Its syntax is as follows:

  ```
  cat [option] [filename]
  ```

  Described below are the different types of `cat` commands.

  - **`cat[filename]`:** This command will display the content of a given filename.
  - **`cat[filename1] [filename2]`**: This command will display the content of filename1 and filename2.
  - **`cat>newfilename`**: This command will create a new file with the name "newfilename."
  - **`cat -n [filename]`**: This command displays the content of a given file with line number.
  - **`cat [filename1]>[filename2]`**: This command copies the content of filename1 to filename2.
  - **`cat -s [filename]`**: This command suppresses repeated empty lines.

- **`cat [filename1]>>[filename2]`**: This command appends the content of filename1 to the end of filename2.
- **`tac [filename]`**: This command displays the file in reverse order.
- **`cat -E [filename]`**: This command highlights the end of the line.

- **`tail` command:** This command displays last 10 lines from a given text file by default. It also allows options ***n*** number of lines and *c* number of characters. Its syntax is as follows:

  `tail [options] [filename(s)]`

  Described below are the different types of **`tail`** commands.

  - **`tail [filename]`**: This command displays the last 10 lines from a given file.
  - **`tail [filename1] [filename2]`**: This command displays the last 10 lines of both the files.
  - **`tail [-n] [filename]`**: This command displays last *n* number of lines from a given file. For example, if 5 is used in place *n*, then only the last five lines will be displayed from a given file.
  - **`tail [-c] [n][filename]`**: This command displays last *n* number of characters from a given file.

- **`head` command:** This command displays the first 10 lines from a given text file by default. It also allows options *n* number of lines and *c* number of characters. Its syntax is as follows:

  `head [options] [filename(s)]`

  Described below are the different types of **`head`** commands.

  - **`head [filename]`**: This command displays the first 10 lines from a given file.
  - **`head [filename1] [filename2]`**: This command displays the first 10 lines of both the files.
  - **`head [-n] [filename]`**: This command displays the first *n* number of lines from a given file. For example, if 5 is used in place *n*, then only the first five lines will be displayed from a given file.
  - **`head [-c] [n][filename]`**: This command displays the first *n* number of characters from a given file.

- **`less` command:** This command displays the contents of a text file, one page (one screen) per time. In case of a large size file, it will not access the complete file; instead, it will access page by page. For example, when using any text editor for reading a large size file, it will get loaded completely to main memory. However, by using **`less`** command, it will not load complete file; instead, it loads part by part, thus making it faster. Its syntax is as follows:

  `less filename`

- **`more` command:** This command displays a number of lines from a text file—as much as the screen can fit. It helps view files in a scrollable manner and search the text, strings, and regular expressions. Its syntax is as follows:

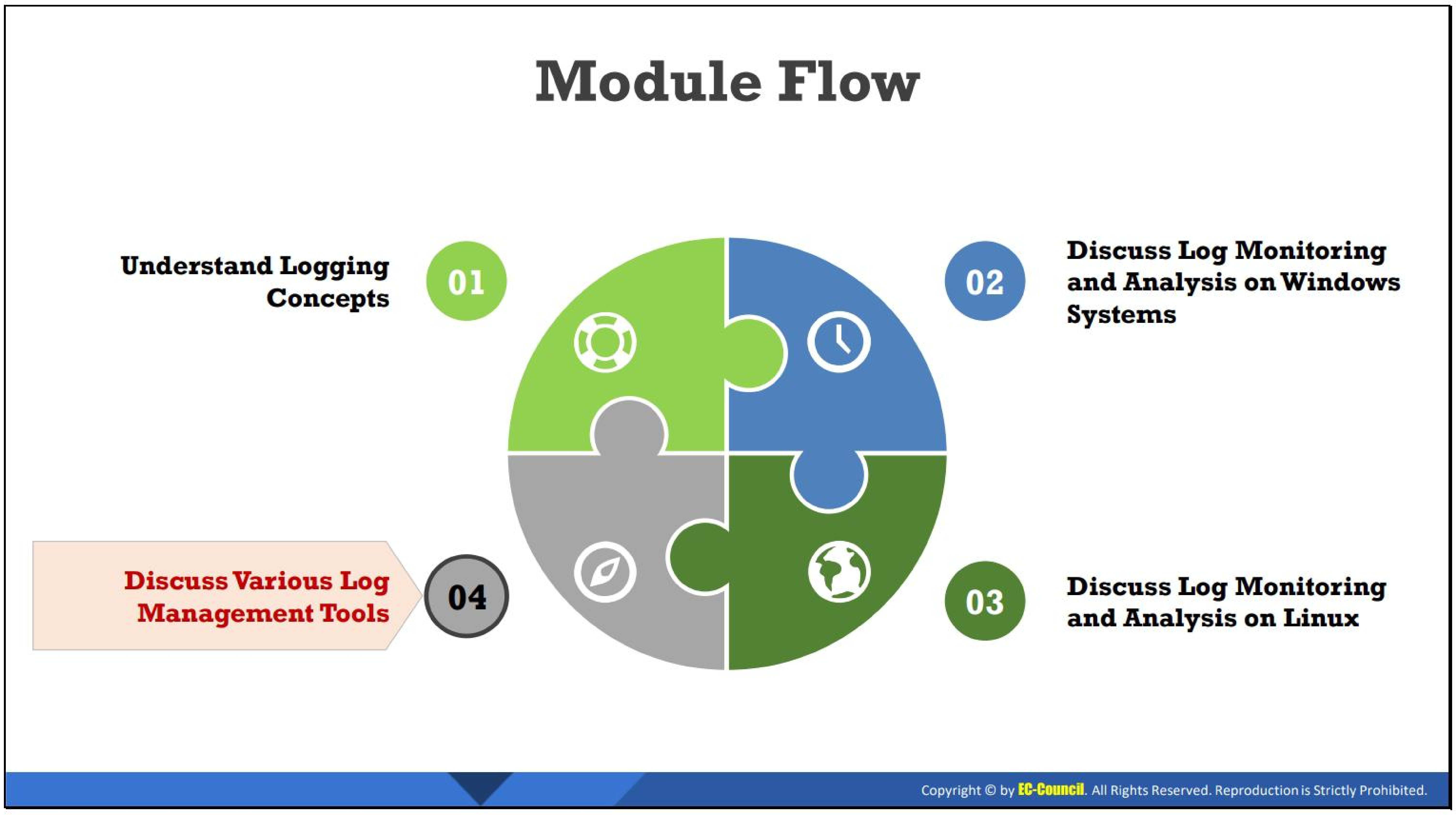
  ```
  more filename
  ```

- **`grep` command:** This command is used for searching a specific string in a file.

  ```
  grep "search_string" [filename]
  ```

  The following available options can be used to search the string:

  - **`-c`**: It displays a count of number of lines that match a pattern.
  - **`-h`**: It displays the matched lines but not the filenames.
  - **`-i`**: It ignores the case for matching.
  - **`-l`**: It displays file names' list.
  - **`-n`**: It displays the line numbers as well as the matched line.
  - **`-v`**: It displays all the lines without a matching pattern.
  - **`-w`**: It matches the whole word.



Module Flow

01 Understand Logging Concepts

02 Discuss Log Monitoring and Analysis on Windows Systems

03 Discuss Log Monitoring and Analysis on Linux

04 Discuss Various Log Management Tools

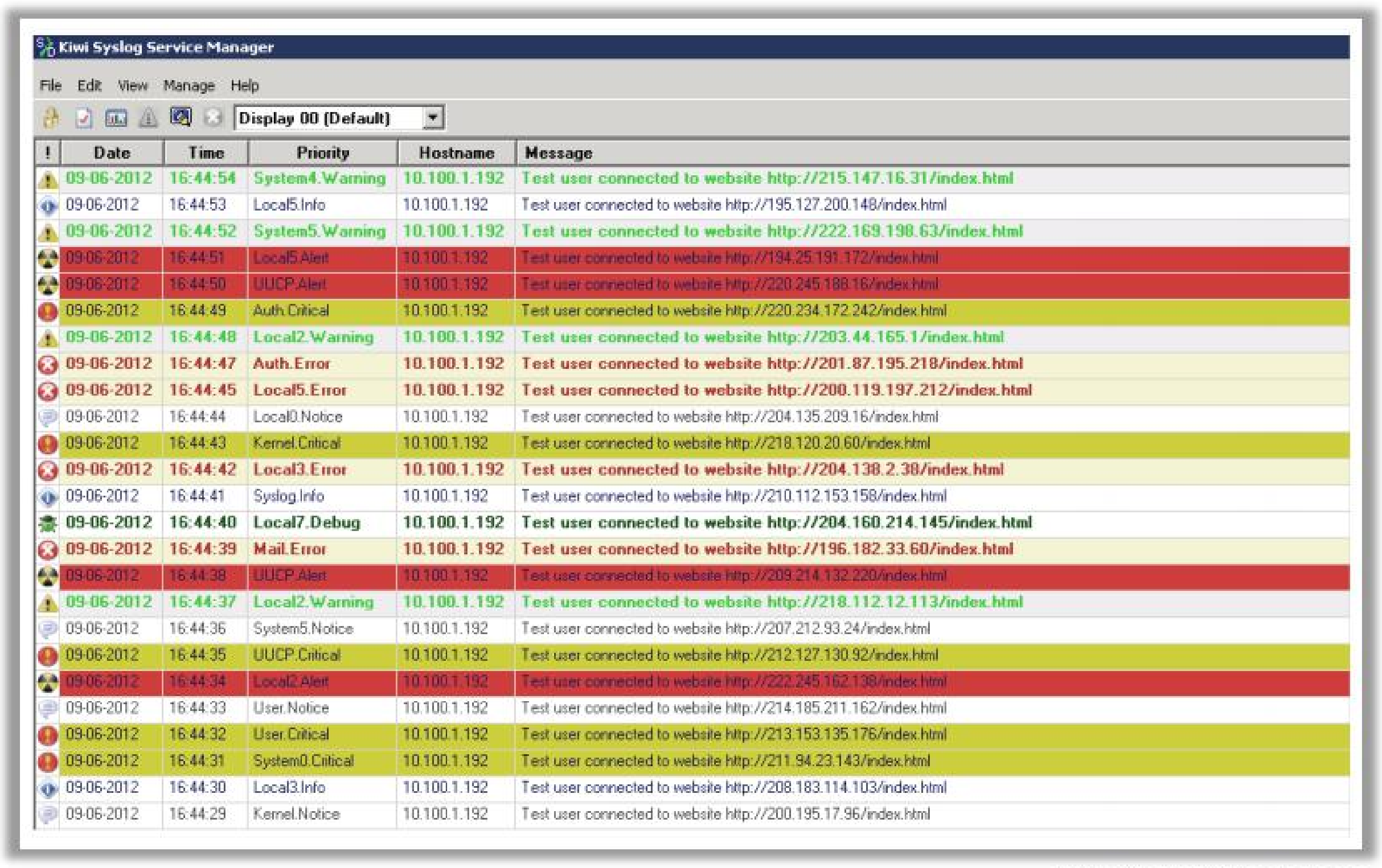
Discuss Various Log Management Tools

In this section, various syslog and log management tools are discussed that help security professionals in monitoring systems, applications, and network events in real-time.

Syslog Tools

Kiwi Syslog Server

Kiwi Syslog Server provides **centralized** and **simplified log message management** across network devices and servers



!	Date	Time	Priority	Hostname	Message
⚠	09-06-2012	16:44:54	System4.Warning	10.100.1.192	Test user connected to website http://215.147.16.31/index.html
ⓘ	09-06-2012	16:44:53	Local5.Info	10.100.1.192	Test user connected to website http://195.127.200.148/index.html
⚠	09-06-2012	16:44:52	System5.Warning	10.100.1.192	Test user connected to website http://222.169.198.63/index.html
☢	09-06-2012	16:44:51	Local5.Alert	10.100.1.192	Test user connected to website http://194.25.191.172/index.html
☢	09-06-2012	16:44:50	UUCP.Alert	10.100.1.192	Test user connected to website http://220.245.188.16/index.html
❗	09-06-2012	16:44:49	Auth.Critical	10.100.1.192	Test user connected to website http://220.234.172.242/index.html
⚠	09-06-2012	16:44:48	Local2.Warning	10.100.1.192	Test user connected to website http://203.44.165.1/index.html
⊗	09-06-2012	16:44:47	Auth.Error	10.100.1.192	Test user connected to website http://201.87.195.218/index.html
⊗	09-06-2012	16:44:45	Local5.Error	10.100.1.192	Test user connected to website http://200.119.197.212/index.html
💬	09-06-2012	16:44:44	Local0.Notice	10.100.1.192	Test user connected to website http://204.135.209.16/index.html
❗	09-06-2012	16:44:43	Kernel.Critical	10.100.1.192	Test user connected to website http://218.120.20.60/index.html
⊗	09-06-2012	16:44:42	Local3.Error	10.100.1.192	Test user connected to website http://204.138.2.38/index.html
ⓘ	09-06-2012	16:44:41	Syslog.Info	10.100.1.192	Test user connected to website http://210.112.153.158/index.html
🐞	09-06-2012	16:44:40	Local7.Debug	10.100.1.192	Test user connected to website http://204.160.214.145/index.html
⊗	09-06-2012	16:44:39	Mail.Error	10.100.1.192	Test user connected to website http://196.182.33.60/index.html
☢	09-06-2012	16:44:38	UUCP.Alert	10.100.1.192	Test user connected to website http://209.214.132.220/index.html
⚠	09-06-2012	16:44:37	Local2.Warning	10.100.1.192	Test user connected to website http://218.112.12.113/index.html
💬	09-06-2012	16:44:36	System5.Notice	10.100.1.192	Test user connected to website http://207.212.93.24/index.html
❗	09-06-2012	16:44:35	UUCP.Critical	10.100.1.192	Test user connected to website http://212.127.130.92/index.html
☢	09-06-2012	16:44:34	Local2.Alert	10.100.1.192	Test user connected to website http://222.245.162.138/index.html
💬	09-06-2012	16:44:33	User.Notice	10.100.1.192	Test user connected to website http://214.185.211.162/index.html
❗	09-06-2012	16:44:32	User.Critical	10.100.1.192	Test user connected to website http://213.153.135.176/index.html
❗	09-06-2012	16:44:31	System0.Critical	10.100.1.192	Test user connected to website http://211.94.23.143/index.html
ⓘ	09-06-2012	16:44:30	Local3.Info	10.100.1.192	Test user connected to website http://208.183.114.103/index.html
💬	09-06-2012	16:44:29	Kernel.Notice	10.100.1.192	Test user connected to website http://200.195.17.96/index.html

https://www.kiwisyslog.com

Splunk Light
https://www.splunk.com

WhatsUp Gold
https://www.whatsupgold.com

syslog-ng
https://www.syslog-ng.com

Fastvue Syslog
https://www.fastvue.co

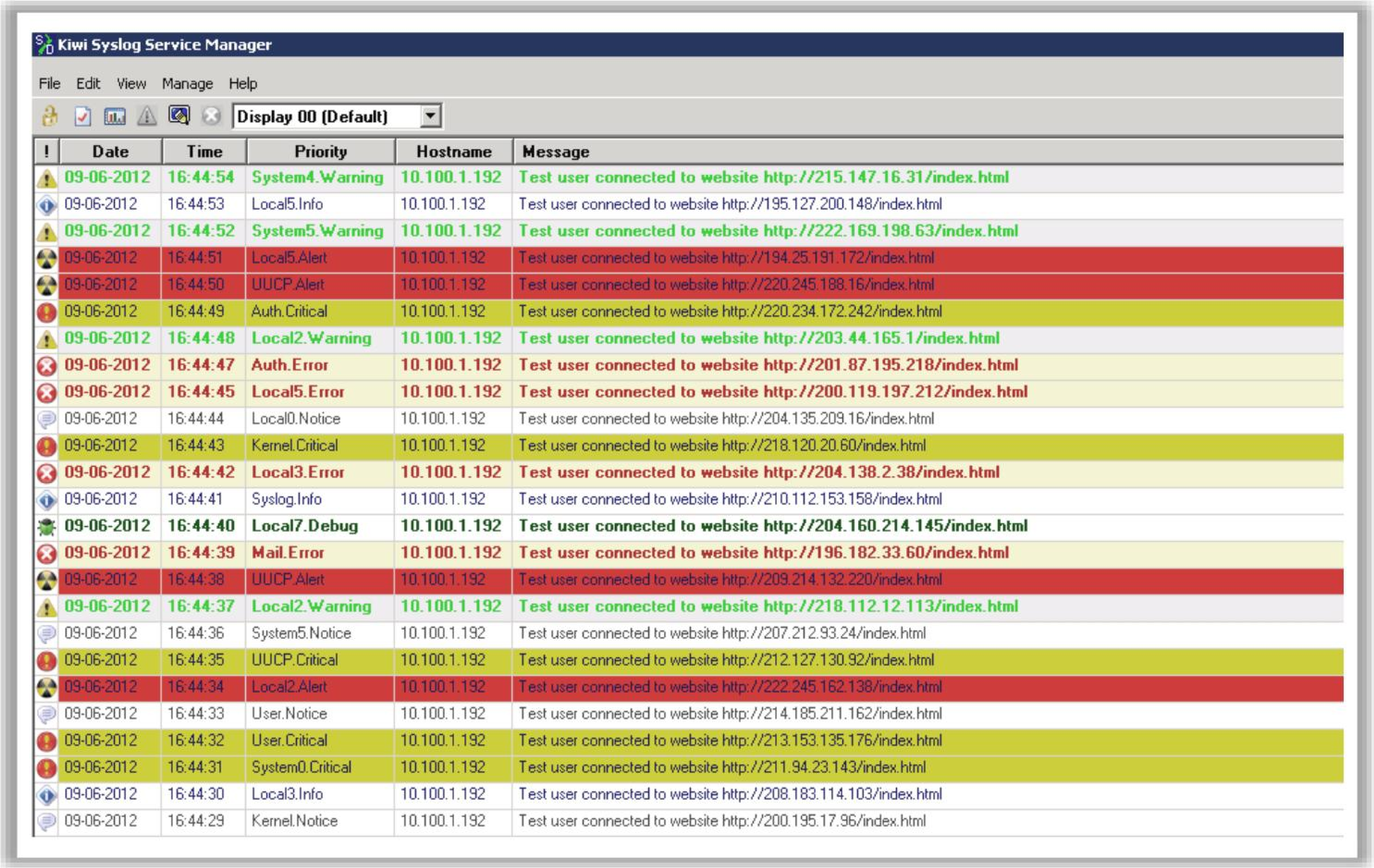
NxLog
https://nxlog.co

Syslog Tools

- **Kiwi Syslog Server**

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

  Kiwi Syslog Server provides centralized and simplified log message management across network devices and servers. It helps in managing syslog messages, SNMP traps, and Windows event logs. It can also be used to monitor real-time logs on a secure and intuitive web interface, thereby further enabling the centralization of logs, to quickly identify issues.



!	Date	Time	Priority	Hostname	Message
⚠	09-06-2012	16:44:54	System4.Warning	10.100.1.192	Test user connected to website http://215.147.16.31/index.html
ℹ	09-06-2012	16:44:53	Local5.Info	10.100.1.192	Test user connected to website http://195.127.200.148/index.html
⚠	09-06-2012	16:44:52	System5.Warning	10.100.1.192	Test user connected to website http://222.169.198.63/index.html
☢	09-06-2012	16:44:51	Local5.Alert	10.100.1.192	Test user connected to website http://194.25.191.172/index.html
☢	09-06-2012	16:44:50	UUCP.Alert	10.100.1.192	Test user connected to website http://220.245.188.16/index.html
❗	09-06-2012	16:44:49	Auth.Critical	10.100.1.192	Test user connected to website http://220.234.172.242/index.html
⚠	09-06-2012	16:44:48	Local2.Warning	10.100.1.192	Test user connected to website http://203.44.165.1/index.html
⊗	09-06-2012	16:44:47	Auth.Error	10.100.1.192	Test user connected to website http://201.87.195.218/index.html
⊗	09-06-2012	16:44:45	Local5.Error	10.100.1.192	Test user connected to website http://200.119.197.212/index.html
💬	09-06-2012	16:44:44	Local0.Notice	10.100.1.192	Test user connected to website http://204.135.209.16/index.html
❗	09-06-2012	16:44:43	Kernel.Critical	10.100.1.192	Test user connected to website http://218.120.20.60/index.html
⊗	09-06-2012	16:44:42	Local3.Error	10.100.1.192	Test user connected to website http://204.138.2.38/index.html
ℹ	09-06-2012	16:44:41	Syslog.Info	10.100.1.192	Test user connected to website http://210.112.153.158/index.html
🐞	09-06-2012	16:44:40	Local7.Debug	10.100.1.192	Test user connected to website http://204.160.214.145/index.html
⊗	09-06-2012	16:44:39	Mail.Error	10.100.1.192	Test user connected to website http://196.182.33.60/index.html
☢	09-06-2012	16:44:38	UUCP.Alert	10.100.1.192	Test user connected to website http://209.214.132.220/index.html
⚠	09-06-2012	16:44:37	Local2.Warning	10.100.1.192	Test user connected to website http://218.112.12.113/index.html
💬	09-06-2012	16:44:36	System5.Notice	10.100.1.192	Test user connected to website http://207.212.93.24/index.html
❗	09-06-2012	16:44:35	UUCP.Critical	10.100.1.192	Test user connected to website http://212.127.130.92/index.html
☢	09-06-2012	16:44:34	Local2.Alert	10.100.1.192	Test user connected to website http://222.245.162.138/index.html
💬	09-06-2012	16:44:33	User.Notice	10.100.1.192	Test user connected to website http://214.185.211.162/index.html
❗	09-06-2012	16:44:32	User.Critical	10.100.1.192	Test user connected to website http://213.153.135.176/index.html
❗	09-06-2012	16:44:31	System0.Critical	10.100.1.192	Test user connected to website http://211.94.23.143/index.html
ℹ	09-06-2012	16:44:30	Local3.Info	10.100.1.192	Test user connected to website http://208.183.114.103/index.html
💬	09-06-2012	16:44:29	Kernel.Notice	10.100.1.192	Test user connected to website http://200.195.17.96/index.html

Figure 18.13: Screenshot of Kiwi Syslog

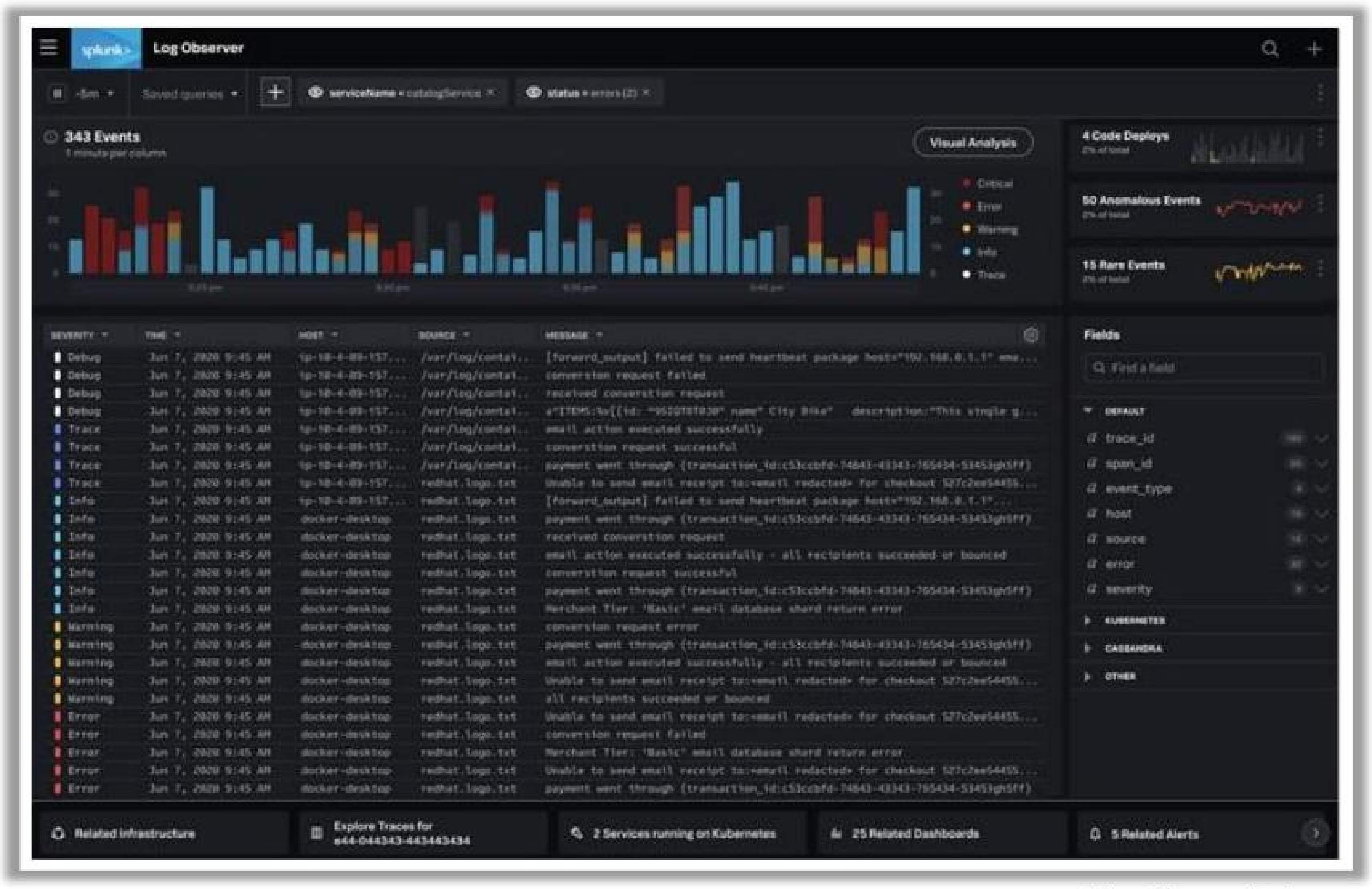
Some additional syslog tools are as follows:

- Splunk Light (*https://www.splunk.com*)
- WhatsUp Gold (*https://www.whatsupgold.com*)
- syslog-ng (*https://www.syslog-ng.com*)
- Fastvue Syslog (*https://www.fastvue.co*)
- NxLog (*https://nxlog.co*)

Log Management Tools

Splunk

Splunk aggregates and analyzes **log data**. It provides insights to quickly detect and respond to internal and **external attacks** and simplify threat management



```
SEVERITY  TIME                  HOST              SOURCE               MESSAGE
Debug     Jun 7, 2020 9:45 AM   ip-10-4-89-157... /var/log/contai...   [forward_output] failed to send heartbeat package host="192.168.0.1.1" ama...
Debug     Jun 7, 2020 9:45 AM   ip-10-4-89-157... /var/log/contai...   conversion request failed
Debug     Jun 7, 2020 9:45 AM   ip-10-4-89-157... /var/log/contai...   received conversion request
Debug     Jun 7, 2020 9:45 AM   ip-10-4-89-157... /var/log/contai...   a"ITEMS:%v[{id: "9520T8T030" name" City Bike"  description:"This single g...
Trace     Jun 7, 2020 9:45 AM   ip-10-4-89-157... /var/log/contai...   email action executed successfully
Trace     Jun 7, 2020 9:45 AM   ip-10-4-89-157... /var/log/contai...   conversion request successful
Trace     Jun 7, 2020 9:45 AM   ip-10-4-89-157... /var/log/contai...   payment went through (transaction_id:c53ccbfd-74843-43343-765434-53453gh5ff)
Trace     Jun 7, 2020 9:45 AM   ip-10-4-89-157... redhat.logo.txt      Unable to send email receipt to:<email redacted> for checkout 527c2ee54455...
Info      Jun 7, 2020 9:45 AM   ip-10-4-89-157... redhat.logo.txt      [forward_output] failed to send heartbeat package host="192.168.0.1.1"...
Info      Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      payment went through (transaction_id:c53ccbfd-74843-43343-765434-53453gh5ff)
Info      Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      received conversion request
Info      Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      email action executed successfully - all recipients succeeded or bounced
Info      Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      conversion request successful
Info      Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      payment went through (transaction_id:c53ccbfd-74843-43343-765434-53453gh5ff)
Info      Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      Merchant Tier: 'Basic' email database shard return error
Warning   Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      conversion request error
Warning   Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      payment went through (transaction_id:c53ccbfd-74843-43343-765434-53453gh5ff)
Warning   Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      email action executed successfully - all recipients succeeded or bounced
Warning   Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      Unable to send email receipt to:<email redacted> for checkout 527c2ee54455...
Warning   Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      all recipients succeeded or bounced
Error     Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      Unable to send email receipt to:<email redacted> for checkout 527c2ee54455...
Error     Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      conversion request failed
Error     Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      Merchant Tier: 'Basic' email database shard return error
Error     Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      Unable to send email receipt to:<email redacted> for checkout 527c2ee54455...
Error     Jun 7, 2020 9:45 AM   docker-desktop    redhat.logo.txt      payment went through (transaction_id:c53ccbfd-74843-43343-765434-53453gh5ff)
```

https://www.splunk.com

Logstash
https://www.elastic.co

Sumo Logic
https://www.sumologic.com

Papertrail
https://papertrailapp.com

LogRhythm
https://logrhythm.com

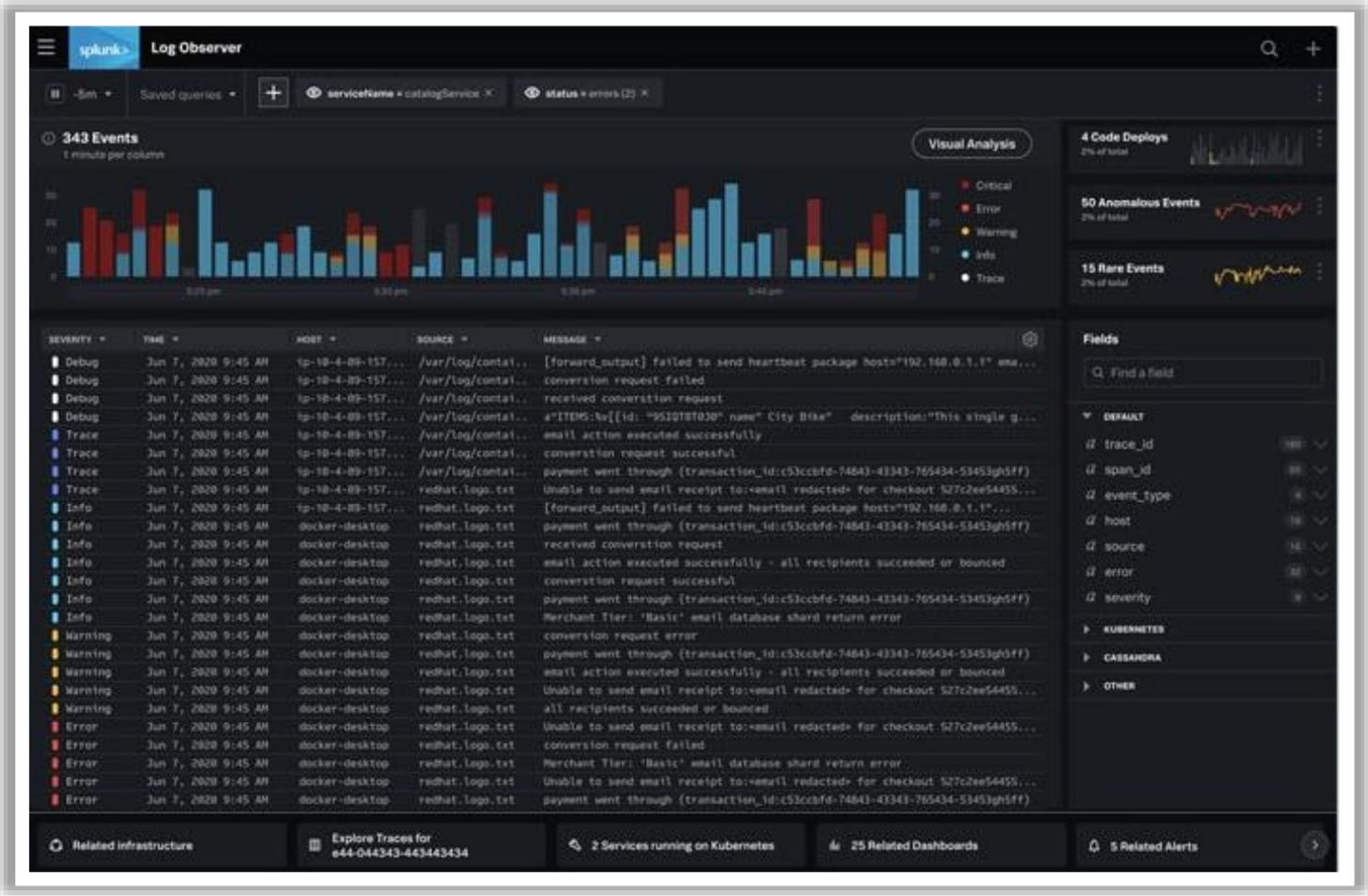
Logentries
https://logentries.com

Log Management Tools

- **Splunk**

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

  Splunk aggregates and analyzes log data. It provides insights to quickly detect and respond to internal and external attacks and simplify threat management. It helps teams gain organization-wide visibility and security intelligence for continuous monitoring, incident response, security operations, and provides executives a window into business risk.

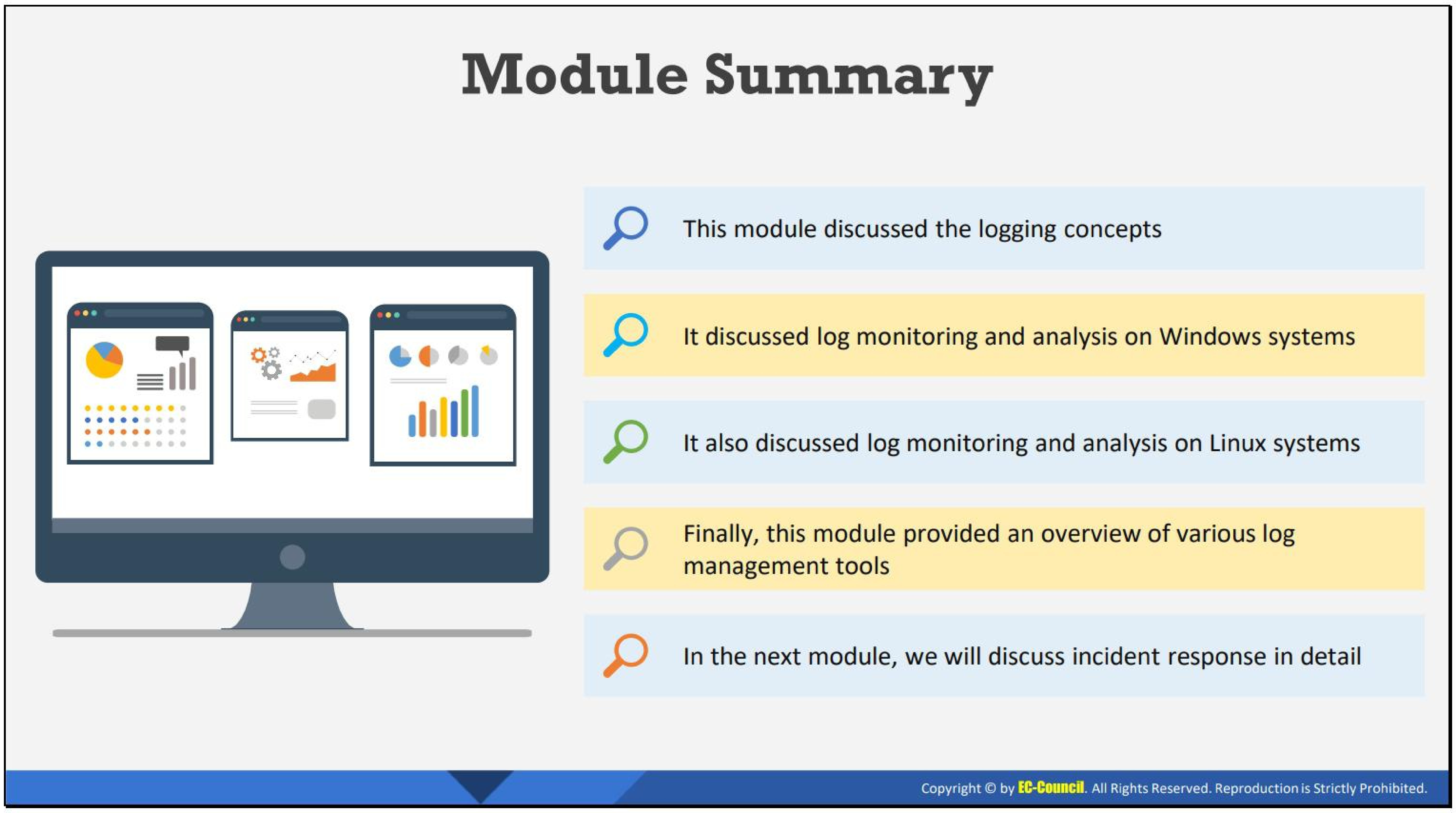


```
SEVERITY   TIME                  HOST                 SOURCE               MESSAGE
Debug      Jun 7, 2020 9:45 AM   ip-10-4-89-157...    /var/log/contai..    [forward_output] failed to send heartbeat package host="192.168.0.1.1" ema...
Debug      Jun 7, 2020 9:45 AM   ip-10-4-89-157...    /var/log/contai..    conversion request failed
Debug      Jun 7, 2020 9:45 AM   ip-10-4-89-157...    /var/log/contai..    received converstion request
Debug      Jun 7, 2020 9:45 AM   ip-10-4-89-157...    /var/log/contai..    a"ITEMS:%v[[id: "9SIDT8T030" name" City Bike"   description:"This single g...
Trace      Jun 7, 2020 9:45 AM   ip-10-4-89-157...    /var/log/contai..    email action executed successfully
Trace      Jun 7, 2020 9:45 AM   ip-10-4-89-157...    /var/log/contai..    converstion request successful
Trace      Jun 7, 2020 9:45 AM   ip-10-4-89-157...    /var/log/contai..    payment went through (transaction_id:c53ccbfd-74843-43343-765434-53453gh5ff)
Trace      Jun 7, 2020 9:45 AM   ip-10-4-89-157...    redhat.logo.txt      Unable to send email receipt to:<email redacted> for checkout 527c2ee54455...
Info       Jun 7, 2020 9:45 AM   ip-10-4-89-157...    redhat.logo.txt      [forward_output] failed to send heartbeat package host="192.168.0.1.1"...
Info       Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      payment went through (transaction_id:c53ccbfd-74843-43343-765434-53453gh5ff)
Info       Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      received converstion request
Info       Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      email action executed successfully - all recipients succeeded or bounced
Info       Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      converstion request successful
Info       Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      payment went through (transaction_id:c53ccbfd-74843-43343-765434-53453gh5ff)
Info       Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      Merchant Tier: 'Basic' email database shard return error
Warning    Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      conversion request error
Warning    Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      payment went through (transaction_id:c53ccbfd-74843-43343-765434-53453gh5ff)
Warning    Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      email action executed successfully - all recipients succeeded or bounced
Warning    Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      Unable to send email receipt to:<email redacted> for checkout 527c2ee54455...
Warning    Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      all recipients succeeded or bounced
Error      Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      Unable to send email receipt to:<email redacted> for checkout 527c2ee54455...
Error      Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      conversion request failed
Error      Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      Merchant Tier: 'Basic' email database shard return error
Error      Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      Unable to send email receipt to:<email redacted> for checkout 527c2ee54455...
Error      Jun 7, 2020 9:45 AM   docker-desktop       redhat.logo.txt      payment went through (transaction_id:c53ccbfd-74843-43343-765434-53453gh5ff)
```

Figure 18.14: Screenshot of Splunk

Some of the additional centralized log management tools include:

- Logstash (*https://www.elastic.co*)
- Sumo Logic (*https://www.sumologic.com*)
- Papertrail (*https://papertrailapp.com*)
- LogRhythm (*https://logrhythm.com*)
- Logentries (*https://logentries.com*)



Module Summary

- This module discussed the logging concepts
- It discussed log monitoring and analysis on Windows systems
- It also discussed log monitoring and analysis on Linux systems
- Finally, this module provided an overview of various log management tools
- In the next module, we will discuss incident response in detail

Module Summary

This module discussed the logging concepts. It discussed log monitoring and analysis on Windows systems. It also discussed log monitoring and analysis on Linux systems. Finally, this module provided an overview of various log management tools.

In the next module, we will discuss incident response in detail.