



CySA+ Lab Series

Lab 09: Working with Log Data

Document Version: **2022-10-03**

Material in this Lab Aligns to the Following	
CompTIA CySA+ (CS0-002) Exam Objectives	1.6 - Explain the threats and vulnerabilities associated with operating in the cloud 3.1 - Given a scenario, analyze data as part of security monitoring activities 4.3 - Given an incident, analyze potential indicators of compromise 4.4 - Given a scenario, utilize basic digital forensics techniques
All-In-One CompTIA CySA+ Second Edition ISBN-13: 978-1260464306 Chapters	6: Threats and Vulnerabilities Associated with Operating in the Cloud 11: Data Analysis in Security Monitoring Activities 17: Analyze Potential Indicators of Compromise 18: Utilize Basic Digital Forensics Techniques

Copyright © 2022 Network Development Group, Inc.
www.netdevgroup.com

NETLAB+ is a registered trademark of Network Development Group, Inc.
KALI LINUX™ is a trademark of Offensive Security.
ALIEN VAULT OSSIM V is a trademark of AlienVault, Inc.
Microsoft®, Windows®, and Windows Server® are trademarks of the Microsoft group of companies.
Greenbone is a trademark of Greenbone Networks GmbH.
SECURITY ONION is a trademark of Security Onion Solutions LLC.
Android is a trademark of Google LLC.
pfSense® is a registered mark owned by Electric Sheep Fencing LLC ("ESF").
All trademarks, logos, and brand names are the property of their respective owners.

Contents

Introduction	3
Objective	3
Lab Topology	4
Lab Settings	5
1 Configuring Log Rotation	6
2 Setting up a Syslog Server Using the CLI	10
2.1 Configure the UbuntuSRV computer to act as a Syslog Server.....	10
2.2 Configure the MintOS Machine to Forward its Logs to the Syslog Server.....	12
3 Setting up a Syslog Server Using the GUI.....	16
4 Log Analysis Using gawk.....	27
4.1 Creating Groups and Users Remotely	27
4.2 Use gawk to Analyze Log Files.....	30
5 Exploring Windows Event Viewer	33

Introduction

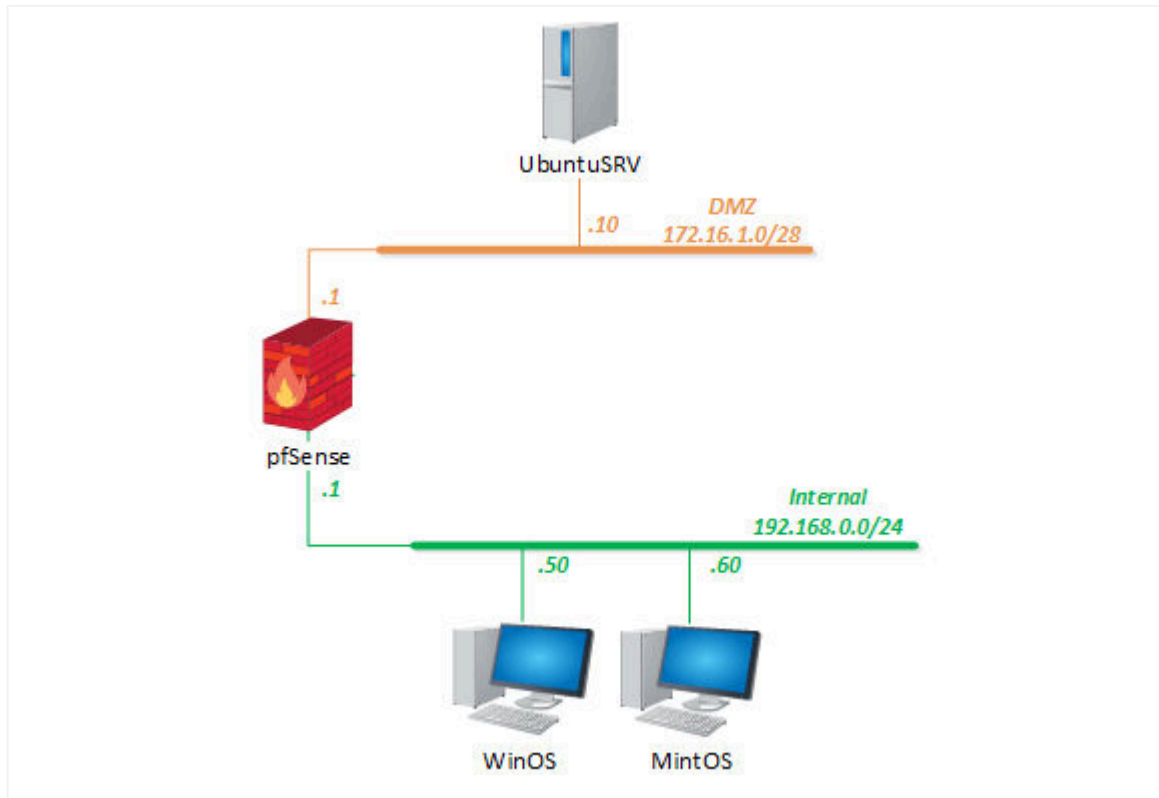
If you were to ask a security analyst what type of tasks they do most often, they would tell you they are responsible for protecting the confidentiality, integrity, and availability of their organization's information; their first task is to evaluate security risks. One of the key things they do is to look at the log files that are generated.

In this lab, you will learn how to perform several tasks pertaining to log management. You will learn how to configure log rotation, as well as set up syslog forwarders and servers. You will view these logs with the CLI and with GUI programs, as well as examine the Windows Event Viewer. Syslog is a very common format for logging to help you monitor your network and is widely used due to its compatibility with most devices and operating systems.

Objective

- Examine syslog in both CLI and GUI environments
- Forward logs to a centralized syslog server both on Linux and Windows
- Examine the Windows Event Viewer

Lab Topology



Lab Settings

The information in the table below will be needed in order to complete the lab. The task sections below provide details on the use of this information.

Virtual Machine	IP Address	Account	Password
WinOS (Server 2019)	192.168.0.50	Administrator	NDGlabpass123!
MintOS (Linux Mint)	192.168.0.60	sysadmin	NDGlabpass123!
OSSIM (Alien Vault)	172.16.1.2	root	NDGlabpass123!
UbuntuSRV (Ubuntu Server)	172.16.1.10	sysadmin	NDGlabpass123!
Kali	203.0.113.2	sysadmin	NDGlabpass123!
pfSense	203.0.113.1 172.16.1.1 192.168.0.1	admin	NDGlabpass123!

1 Configuring Log Rotation

In this task, you will view several syslog reports and examine a log rotation configuration file.

1. Set the focus to the **UbuntuSRV** computer and log in as sysadmin with the password: NDGLabpass123!

```
Ubuntu 20.04.3 LTS ubuntu:~$ ssh -o StrictHostKeyChecking=no root@192.168.1.100
ubuntu:~$ ssh root@192.168.1.100
root@192.168.1.100:~# cat /etc/passwd | grep sysadmin
sysadmin:x:1000:1000::/home/sysadmin:/bin/bash
```

2. Navigate to the `/var/log/` directory using the following command:

```
cd /var/log/
```

```
sysadmin@ubuntusrv:/$ cd /var/log
sysadmin@ubuntusrv:/var/log$
```

3. Use the **ls** command to view the contents of the directory. Notice the *syslog* files.

15

```

sysadmin@ubuntu:~$ ls /var/log$ ls
alternatives.log      dist-upgrade          kern.log.2.gz         ubuntu-advantage.log.2.gz
alternatives.log.1    dmesg                 kern.log.3.gz         unattended-upgrades
alternatives.log.2.gz dmesg.0               kern.log.4.gz         vmware-network.1.log
alternatives.log.3.gz dmesg.1.gz            landscape              vmware-network.2.log
apache2               dmesg.2.gz            lastlog                vmware-network.3.log
apt                  dmesg.3.gz            private                vmware-network.4.log
auth.log              dmesg.4.gz            syslog                 vmware-network.5.log
auth.log.1            dpkg.log               syslog.1               vmware-network.log
auth.log.2.gz          dpkg.log.1            syslog.2.gz            vmware-vmsvc-root.1.log
auth.log.3.gz          dpkg.log.2.gz         syslog.3.gz            vmware-vmsvc-root.2.log
auth.log.4.gz          dpkg.log.3.gz         syslog.4.gz            vmware-vmsvc-root.3.log
bootstrap.log          faillog                syslog.5.gz            vmware-vmsvc-root.log
btmtp                 installer              syslog.6.gz            vmware-vmttoolsd-root.log
btmtp.1               journal                syslog.7.gz            wtmp
cloud-init.log         kern.log               ubuntu-advantage.log
cloud-init-output.log  kern.log.1             ubuntu-advantage.log.1

```

4. To view the contents of the first **syslog** file, type the following command:

```
less syslog
```

```
sysadmin@ubuntusrv:/var/log$ less syslog
```

5. Scroll down to look through the messages by pressing the **Space Bar** to go to the next page. Notice the information given in each message, the date, the machine the message originated from, and

the severity of the message are given before the actual message is displayed. Press **q** when you are finished to return to the **Shell** screen.

```
Oct 27 22:14:04 ubuntu:rsrv systemd-modules-load[494]: Inserted module 'msr'
Oct 27 22:14:04 ubuntu:rsrv systemd-modules-load[494]: Inserted module 'arp_tables'
Oct 27 22:14:04 ubuntu:rsrv systemd-modules-load[494]: Inserted module 'br_netfilter'
Oct 27 22:14:04 ubuntu:rsrv systemd-modules-load[494]: Inserted module 'ip6table_filter'
Oct 27 22:14:04 ubuntu:rsrv systemd-modules-load[494]: Inserted module 'iptable_filter'
Oct 27 22:14:04 ubuntu:rsrv systemd-modules-load[494]: Inserted module 'overlay'
Oct 27 22:14:04 ubuntu:rsrv systemd-sysctl[511]: Not setting net/ipv4/conf/all/promote_secondaries (explicit setting exists).
Oct 27 22:14:04 ubuntu:rsrv lvm[486]: 1 logical volume(s) in volume group "ubuntu-vg" monitored
Oct 27 22:14:04 ubuntu:rsrv systemd-sysctl[511]: Not setting net/ipv4/conf/default/promote_secondaries (explicit setting exists).
Oct 27 22:14:04 ubuntu:rsrv systemd-networkd[514]: Enumeration completed
Oct 27 22:14:04 ubuntu:rsrv systemd-networkd[514]: ens192: IPv6 successfully enabled
Oct 27 22:14:04 ubuntu:rsrv systemd-networkd[514]: ens192: Link UP
Oct 27 22:14:04 ubuntu:rsrv systemd-networkd[514]: ens192: Gained carrier
Oct 27 22:14:04 ubuntu:rsrv systemd[1]: Starting Flush Journal to Persistent Storage...
Oct 27 22:14:04 ubuntu:rsrv systemd[1]: Finished Flush Journal to Persistent Storage.
Oct 27 22:14:04 ubuntu:rsrv systemd-udevd[531]: Using default interface naming scheme 'v245'.
Oct 27 22:14:04 ubuntu:rsrv systemd-udevd[531]: ethtool: autonegotiation is unset or enabled, the speed and duplex are not writable.
Oct 27 22:14:04 ubuntu:rsrv multipath: sda: failed to get udev uid: Invalid argument
Oct 27 22:14:04 ubuntu:rsrv multipath: sda: failed to get sysfs uid: Invalid argument
Oct 27 22:14:04 ubuntu:rsrv multipath: sda: failed to get sgio uid: No such file or directory
Oct 27 22:14:04 ubuntu:rsrv systemd[1]: Listening on Load/Save RF Kill Switch Status /dev/rfkill Watch.
Oct 27 22:14:04 ubuntu:rsrv systemd-udevd[529]: ethtool: autonegotiation is unset or enabled, the speed and duplex are not writable.
Oct 27 22:14:04 ubuntu:rsrv udevadm[536]: systemd-udev-settle.service is deprecated.
Oct 27 22:14:04 ubuntu:rsrv systemd[1]: Created slice system-lvm2\x2dvpvscan.slice.
Oct 27 22:14:04 ubuntu:rsrv systemd[1]: Starting LVM event activation on device 8:3...
Oct 27 22:14:04 ubuntu:rsrv systemd[1]: Found device Virtual_disk 2.
Oct 27 22:14:04 ubuntu:rsrv lvm[603]: pvscan[603] PV /dev/sda3 online, VG ubuntu-vg is complete.
Oct 27 22:14:04 ubuntu:rsrv lvm[603]: pvscan[603] VG ubuntu-vg skip autoactivation.
Oct 27 22:14:04 ubuntu:rsrv systemd[1]: Finished LVM event activation on device 8:3.
Oct 27 22:14:04 ubuntu:rsrv systemd[1]: Finished udev Wait for Complete Device Initialization.
Oct 27 22:14:04 ubuntu:rsrv systemd[1]: Starting Device-Mapper Multipath Device Controller...
```

- As log files fill up, log rotation will create new files to keep the files at a manageable size. When the *syslog* file fills up, a *.1* is appended to the log name, and a new *syslog* file is created. If there was an existing *syslog.1*, then it becomes *syslog.2*, *syslog.2* becomes *syslog.3*, *syslog.3* becomes *syslog.4* and so on. When the maximum number of logs in a rotation is reached, the oldest log file is deleted (the default number of rotations is 4, which is set in the *logrotate.conf* file).

To view one of these older files, execute the following command:

```
less syslog.1
```

```
sysadmin@ubuntusrv:/var/log$ less syslog.1_
```

7. Examine the **syslog.1** file. Note that the dates on some of the messages in this log file are older. When you are finished, press **q** to return to the **Shell**.

```
Oct  5 19:24:50 ubuntu:rsrv systemd[1]: snap.docker.dockerd.service: Scheduled restart job, restart co
unter is at 3.
Oct  5 19:24:50 ubuntu:rsrv rsyslogd: [origin software="rsyslogd" swVersion="8.2001.0" x-pid="755" x-i
nfo="https://www.rsyslog.com"] rsyslogd was HUPed
Oct  5 19:24:50 ubuntu:rsrv systemd[1]: Stopped Service for snap application docker.dockerd.
Oct  5 19:24:50 ubuntu:rsrv systemd[1]: Started Service for snap application docker.dockerd.
Oct  5 19:24:50 ubuntu:rsrv systemd[830]: snap.docker.dockerd.service: Failed to execute command: No s
uch file or directory
Oct  5 19:24:50 ubuntu:rsrv systemd[830]: snap.docker.dockerd.service: Failed at step EXEC spawning /u
sr/bin/snap: No such file or directory
Oct  5 19:24:50 ubuntu:rsrv systemd[1]: snap.docker.dockerd.service: Main process exited, code=exited,
status=203/EXEC
Oct  5 19:24:50 ubuntu:rsrv systemd[1]: snap.docker.dockerd.service: Failed with result 'exit-code'.
Oct  5 19:24:50 ubuntu:rsrv systemd[1]: Started Disk Manager.
Oct  5 19:24:50 ubuntu:rsrv udisksd[759]: Acquired the name org.freedesktop.UDisks2 on the system mess
age bus
Oct  5 19:24:50 ubuntu:rsrv networkd-dispatcher[754]: No valid path found for iwconfig
Oct  5 19:24:50 ubuntu:rsrv systemd[1]: logrotate.service: Succeeded.
Oct  5 19:24:50 ubuntu:rsrv systemd[1]: Finished Rotate log files.
Oct  5 19:24:50 ubuntu:rsrv systemd[1]: Started Dispatcher daemon for systemd-networkd.
Oct  5 19:24:50 ubuntu:rsrv systemd[1]: snap.docker.dockerd.service: Scheduled restart job, restart co
unter is at 4.
```

8. To view the log rotation file, first navigate to **/etc/** using the following command:

```
cd /etc
```

```
sysadmin@ubuntu:rsrv:/var/log$ cd /etc
sysadmin@ubuntu:rsrv:/etc$
```

9. Use the following command to view the log rotation configuration file (**logrotate.conf**):

```
less logrotate.conf
```

```
sysadmin@ubuntu:rsrv:/etc$ less logrotate.conf_
```


10. Examine the contents of the log rotation file. Notice several details of the configuration: by default, it is set to rotate files weekly, keeping 4 weeks of files. Additionally, when a log file is rotated out, a new log file is created. Press **q** to return to the **Shell** when finished.

```
# see "man logrotate" for details
# rotate log files weekly
weekly

# use the adm group by default, since this is the owning group
# of /var/log/syslog.
su root adm

# keep 4 weeks worth of backlogs
rotate 4

# create new (empty) log files after rotating old ones
create

# use date as a suffix of the rotated file
#dateext

# uncomment this if you want your log files compressed
#compress

# packages drop log rotation information into this directory
include /etc/logrotate.d

# system-specific logs may be also be configured here.
logrotate.conf (END)
```

11. Keep the *UbuntuSRV* computer open for the next task.

2 Setting up a Syslog Server Using the CLI

In this task, you will configure the **UbuntuSRV** computer to act as a syslog server, to which the **MintOS** computer will forward its log files.

2.1 Configure the UbuntuSRV Computer to Act as a Syslog Server

1. Open the **rsyslog.conf** file by typing the following command:

```
sudo nano /etc/rsyslog.conf
```

If asked for **[sudo] password for sysadmin**, type: NDGLabpass123!

```
sysadmin@ubuntusrv:~$ sudo nano /etc/rsyslog.conf
[sudo] password for sysadmin: _
```

2. Rsyslog can use either *TCP* or *UDP* for log reception. Because the *UDP* protocol has less overhead, it will be used here. Uncomment (remove the **#s**) the two statements beneath the **Provides UDP syslog reception** module.

```
GNU nano 4.8 /etc/rsyslog.conf
# /etc/rsyslog.conf configuration file for rsyslog
#
# For more information install rsyslog-doc and see
# /usr/share/doc/rsyslog-doc/html/configuration/index.html
#
# Default logging rules can be found in /etc/rsyslog.d/50-default.conf

#####
#### MODULES ####
#####

module(load="imuxsock") # provides support for local system logging
#module(load="immark") # provides --MARK-- message capability

# provides UDP syslog reception
module(load="imudp")
input(type="imudp" port="514")

# provides TCP syslog reception
#module(load="imtcp")
#input(type="imtcp" port="514")

# provides kernel logging support and enable non-kernel klog messages
module(load="imklog" permitnonkernelfacility="on")
```

- In the *MODULES* section, beneath the *UDP* and *TCP* reception statements, designate where the *syslog* messages will be stored and how to organize them by inserting the following text:

```
$template RemoteLogs, "/var/log/%HOSTNAME%/mint.log"
*. * ?RemoteLogs
```

```
#####
#### MODULES ####
#####

module(load="imuxsock") # provides support for local system logging
#module(load="immark") # provides --MARK-- message capability

# provides UDP syslog reception
module(load="imudp")
input(type="imudp" port="514")

# provides TCP syslog reception
#module(load="imtcp")
#input(type="imtcp" port="514")

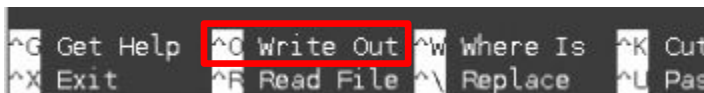
$template RemoteLogs, "/var/log/%HOSTNAME%/mint.log"
*. * ?RemoteLogs

# provides kernel logging support and enable non-kernel klog messages
module(load="imklog" permitnonkernelfacility="on")
```

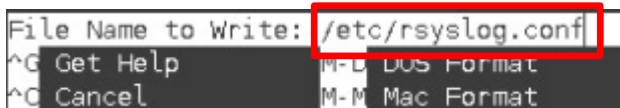


Note the path will utilize/create a directory for the remote *syslog* client's hostname. This will help to organize logs in case future review is required.

- When finished, press **Ctrl+O** to write the file.



- Press **Enter** to confirm the file name **/etc/rsyslog.conf**.



6. Press **Ctrl+X** to exit.



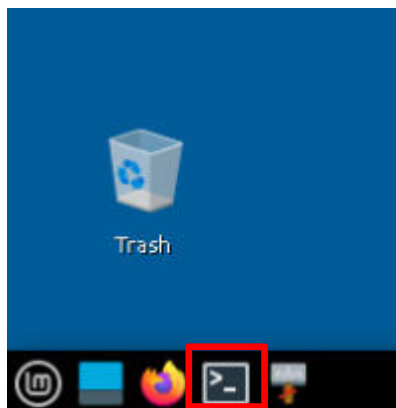
7. Restart the **syslog** service by typing the following command:

```
sudo service rsyslog restart
```

```
sysadmin@ubuntusrv:~$ sudo service rsyslog restart
```

2.2 Configure the MintOS Machine to Forward its Logs to the Syslog Server

1. Set the focus on the **MintOS** computer.
2. Log in as **sysadmin** using the password: **NDGlabpass123!**
3. Click on the **Terminal** icon in the taskbar at the bottom of the screen.



4. Open the **rsyslog.conf** file by typing the following command. If asked for **[sudo]** password for **sysadmin**, type: **NDGlabpass123!**

```
sudo nano /etc/rsyslog.conf
```

```
File Edit View Terminal Tabs Help
sysadmin@mintos:~$ sudo nano /etc/rsyslog.conf
[sudo] password for sysadmin: *****
```

- Arrow down after the initial comments and just above the *MODULES* section and add the following statement: `*.* @172.16.1.10:514`. This is the IP address of the *UbuntuSRV* machine specifying the port you allowed for UDP syslog communication.

```
File Edit View Terminal Tabs Help
GNU nano 4.8 /etc/rsyslog.conf Mo
# /etc/rsyslog.conf configuration file for rsyslog
#
# For more information install rsyslog-doc and see
# /usr/share/doc/rsyslog-doc/html/configuration/index.html
#
# Default logging rules can be found in /etc/rsyslog.d/50-default.conf
*.* @172.16.1.10:514
#####
#### MODULES ####
#####
```

- When finished, press **Ctrl+O** to write the file.

```
^G Get Help ^O Write Out ^W Where Is ^K Cut
^X Exit ^R Read File ^\ Replace ^L Pas
```

- Press **Enter** to confirm the file name `/etc/rsyslog.conf`.

```
File Name to Write: /etc/rsyslog.conf
^G Get Help M-D DOS Format
^C Cancel M-M Mac Format
```

- Press **Ctrl+X** to exit.

```
^G Get Help ^O Write Out ^W Where Is ^K Cut
^X Exit ^R Read File ^\ Replace ^L Pas
```

- Type the following command to restart the **rsyslog** service:

```
sudo service rsyslog restart
```

```
sysadmin@mintos:~$ sudo service rsyslog restart
```

- Type the following command to send a log command to the **syslog** file on the *UbuntuSRV*:

```
logger -s "TEST"
```

```
sysadmin@mintos:~$ logger -s "TEST"
<13>Oct 27 19:21:12 sysadmin: TEST
```

11. Return to the **UbuntuSRV** computer.
12. Since the logs are in a protected part of the file system, you will need to run the following commands under root. In the terminal, type the following command:

```
sudo su
```

```
sysadmin@ubuntusrv:~$ sudo su
root@ubuntusrv:/home/sysadmin# _
```

13. Go to the remote log directory by typing the following command:

```
cd /var/log/mintos
```

```
root@ubuntusrv:/home/sysadmin# cd /var/log/mintos
root@ubuntusrv:/var/log/mintos#
```

14. Type the **ls** command to see that there is a **mint.log** file in the folder:

```
ls
```

```
root@ubuntusrv:/var/log/mintos# ls
mint.log
```


15. Type the following command to view the *syslog* file:

```
cat mint.log
```

```
root@ubuntusrv:/var/log/mintos# cat mint.log
```

In the log, notice the **TEST** that was sent from the *MintOS* machine. You have successfully forwarded a syslog file.

```
2022-08-02T15:12:15+00:00 mintos systemd[1]: rsyslog.service: Succeeded.
2022-08-02T15:12:15+00:00 mintos systemd[1]: Stopped System Logging Service.
2022-08-02T15:12:15+00:00 mintos systemd[1]: Starting System Logging Service...
2022-08-02T15:12:15+00:00 mintos systemd[1]: Started System Logging Service.
2022-08-02T15:12:15+00:00 mintos sudo: pam_unix(sudo:session): session closed for
2022-08-02T15:12:15+00:00 mintos rsyslogd: imuxsock: Acquired UNIX socket '/run/s
og' (fd 3) from systemd. [v8.2001.0]
2022-08-02T15:12:15+00:00 mintos rsyslogd: rsyslogd's groupid changed to 110
2022-08-02T15:12:15+00:00 mintos rsyslogd: rsyslogd's userid changed to 104
2022-08-02T15:12:15+00:00 mintos rsyslogd: [origin software="rsyslogd" swVersion=
903" x-info="https://www.rsyslog.com"] start
2022-08-02T15:12:17+00:00 mintos sudo: pam_unix(sudo:session): session closed for
2022-08-02T15:12:20+00:00 mintos ntpd[886]: error resolving pool 2.ubuntu.pool.nt
ice not known (-2)
2022-08-02T15:12:22+00:00 mintos sysadmin: TEST
2022-08-02T15:12:31+00:00 mintos ntpd[886]: error resolving pool 0.ubuntu.pool.nt
ice not known (-2)
2022-08-02T15:12:41+00:00 mintos ntpd[886]: error resolving pool 3.ubuntu.pool.nt
ice not known (-2)
2022-08-02T15:12:51+00:00 mintos ntpd[886]: error resolving pool ntp.ubuntu.com:
known (-2)
```

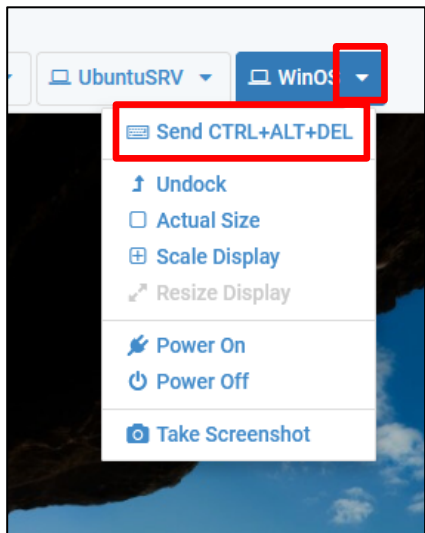
16. Type **exit** to return to the *sysadmin* account.

```
root@ubuntusrv:/var/log/mintos# exit
exit
sysadmin@ubuntusrv:~$ _
```

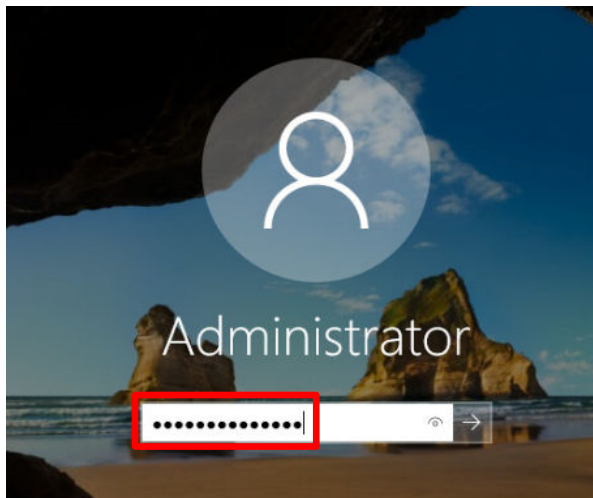
3 Setting up a Syslog Server Using the GUI

In this task, you will set up *syslog* forwarding and GUI-based *syslog* server software in order to examine system logs.

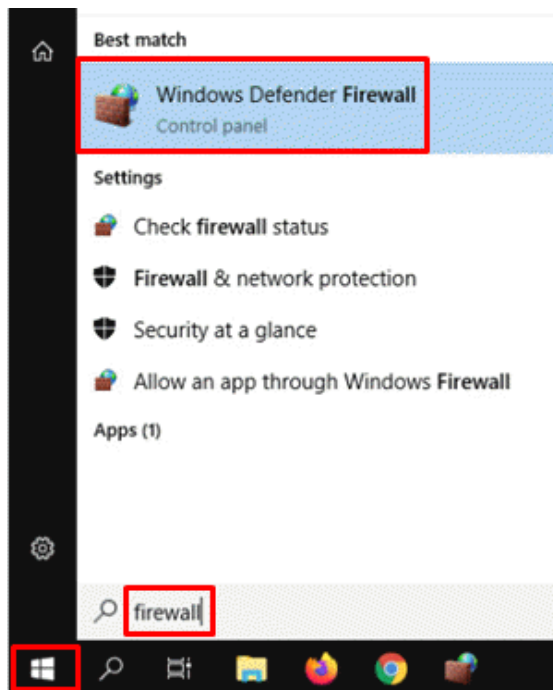
1. Set the focus on the **WinOS** computer to access the graphical login screen.
2. Bring up the login window by sending a Ctrl + Alt + Delete. To do this, click the **WinOS** dropdown menu and click **Send CTRL+ALT+DEL**.



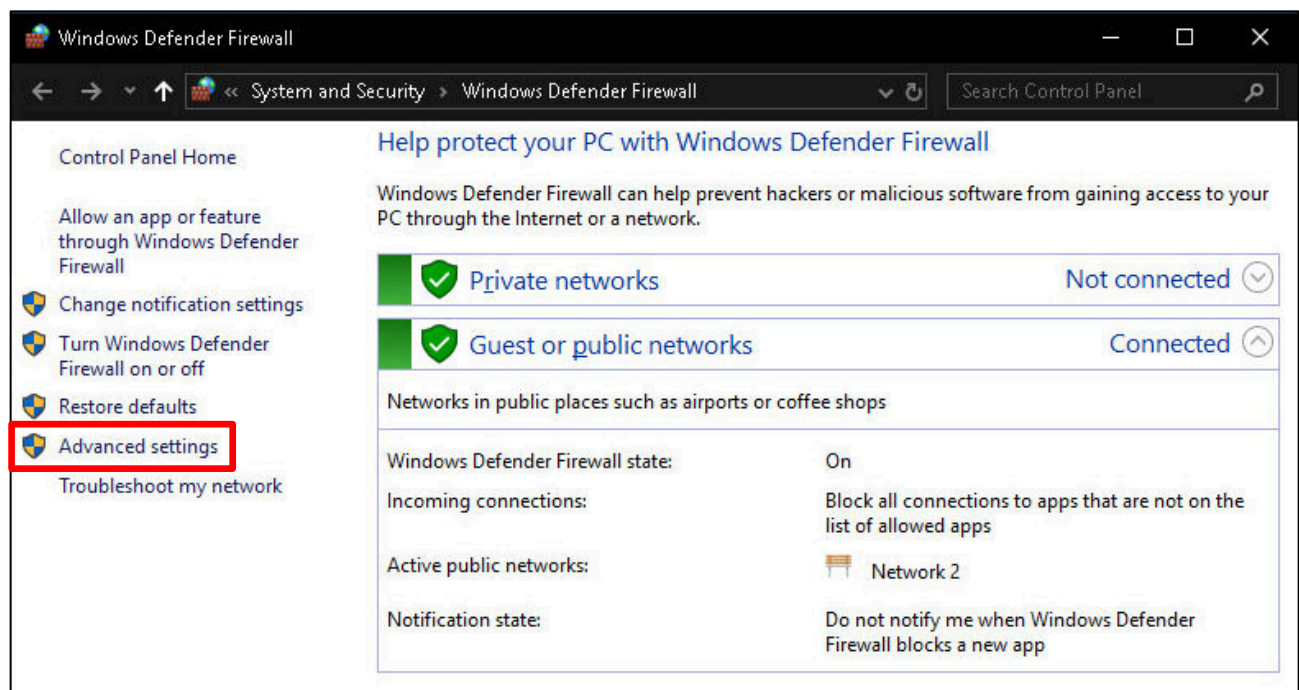
3. Log in as *Administrator* using the password: NDGLabpass123!



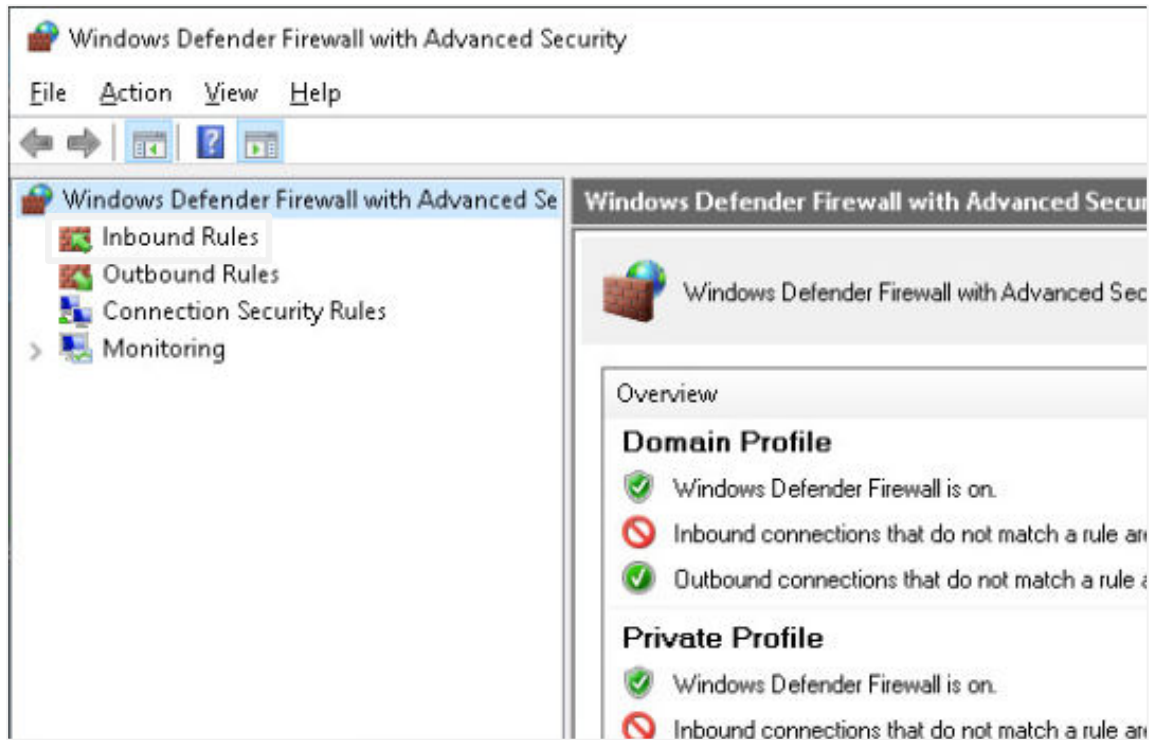
4. Since the *MintOS* computer will be reconfigured later in this task to send syslog packets to the *WinOS* computer, it will be necessary to open the Syslog UDP Port 514. Click on the **Windows Start** button, type `firewall` and click on **Windows Defender Firewall** under *Best Match*.



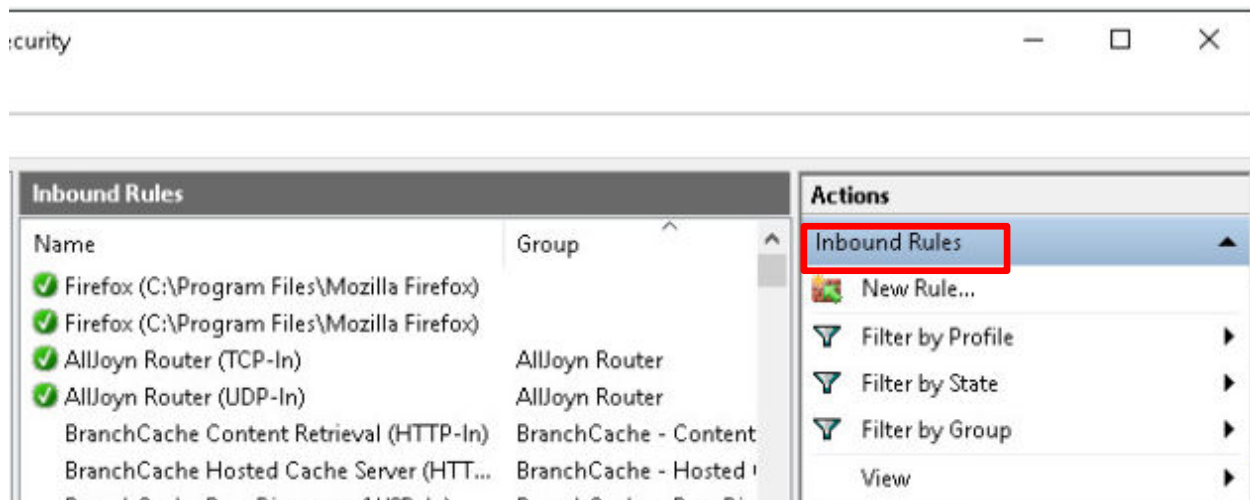
5. On the left side of the window, click on **Advanced Settings**.



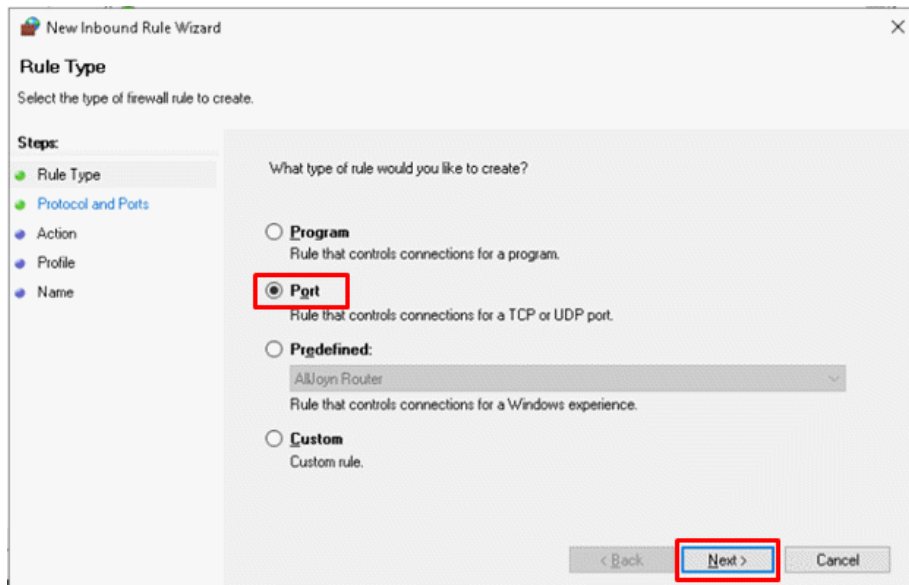
6. The *Windows Defender Firewall with Advanced Security* page will open and in the left panel, click on **Inbound Rules**.



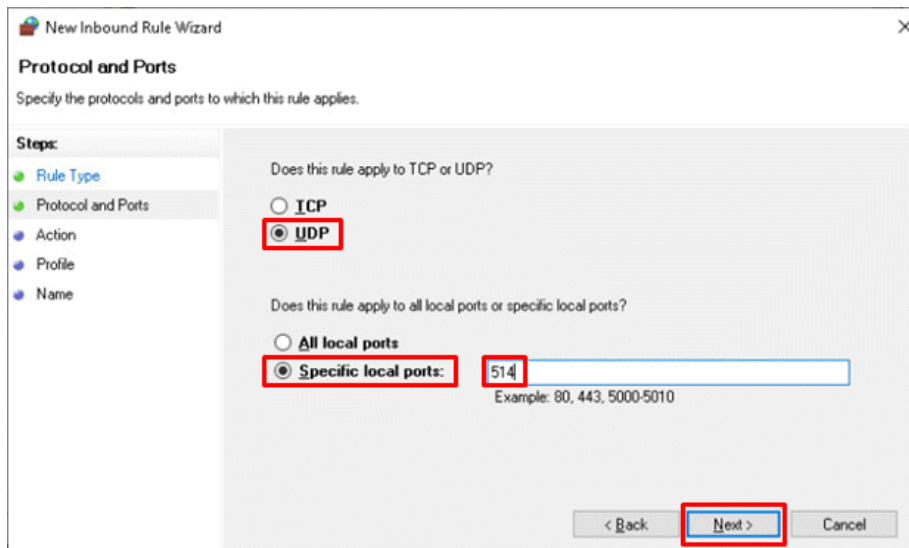
7. On the right panel, click on **New Rule**.



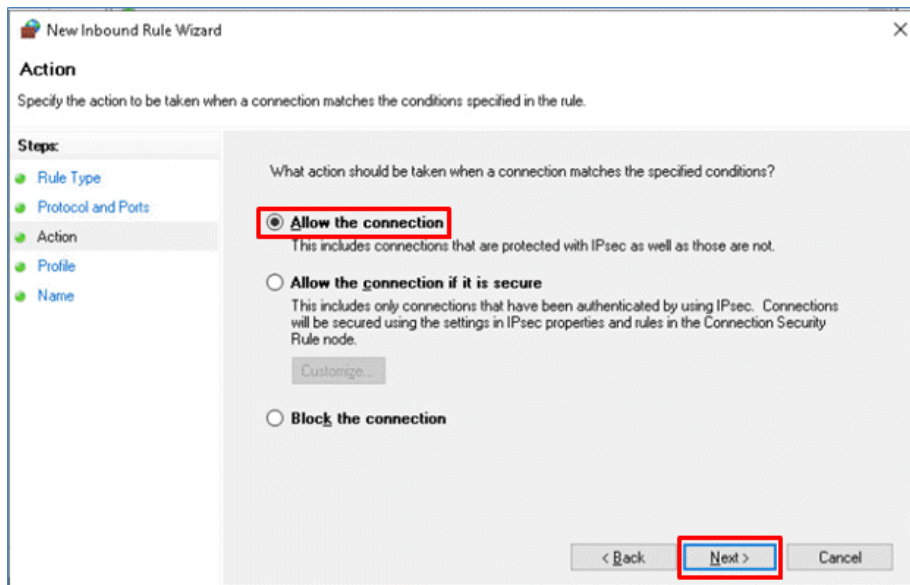
8. In the *New Inbound Rule Wizard*, in the *Rule Type* window, click on the **Port** radio button and click the **Next** button.



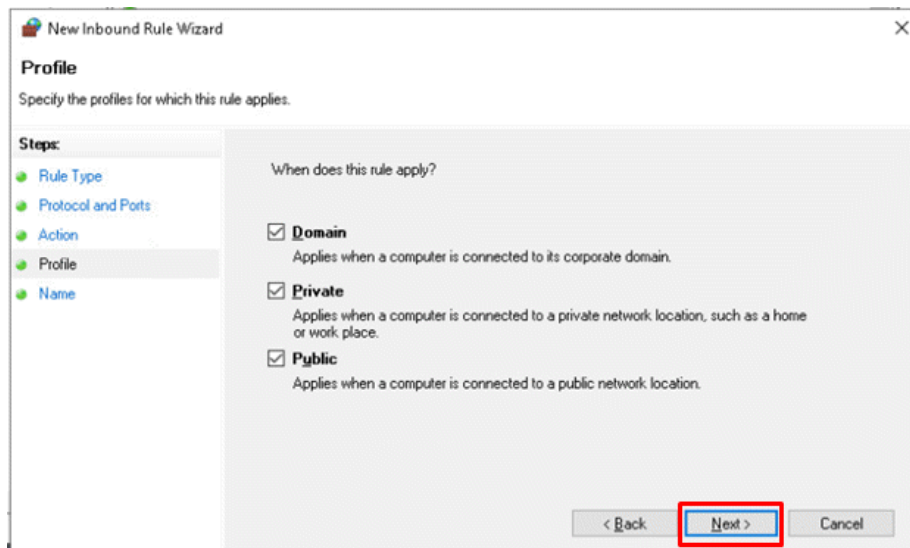
9. In the *Protocol and Ports* window, click on the **UDP** radio button, the **Specific Local Ports** radio button, and type 514, which represents *Syslog Port 514*. Then, click the **Next** button.



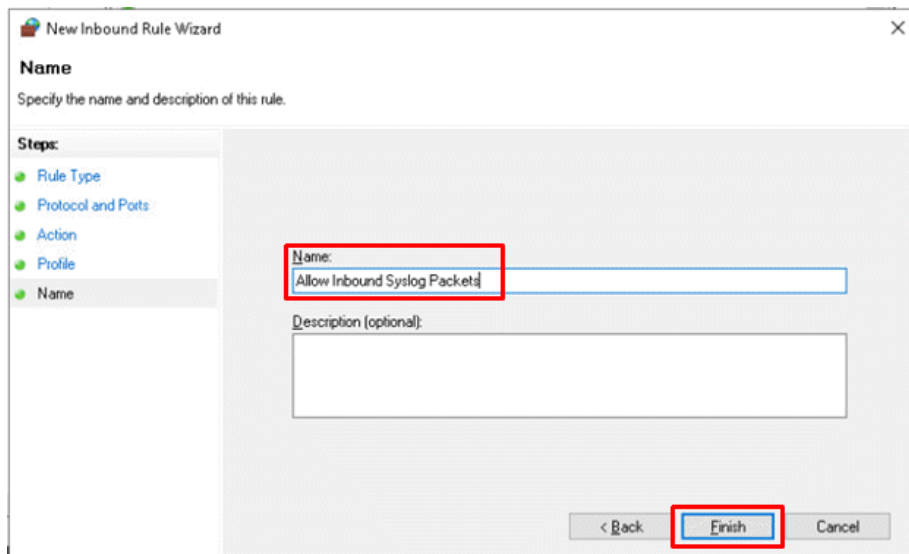
10. In the *Action* window, make sure the **Allow the Connection** radio button is selected and click **Next**.



11. In the *Profile* window, leave the three network types checkboxes set, then click on the **Next** button.

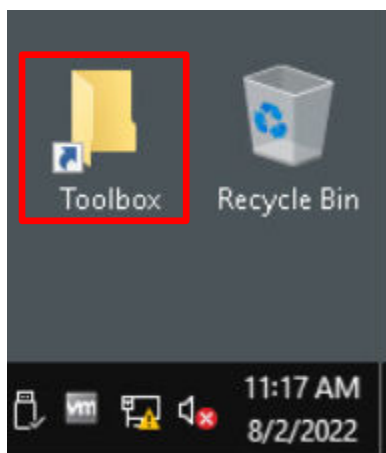


12. In the *Name* window, in the *Name* box, type `Allow Inbound Syslog Packets` and click on **Finish**.

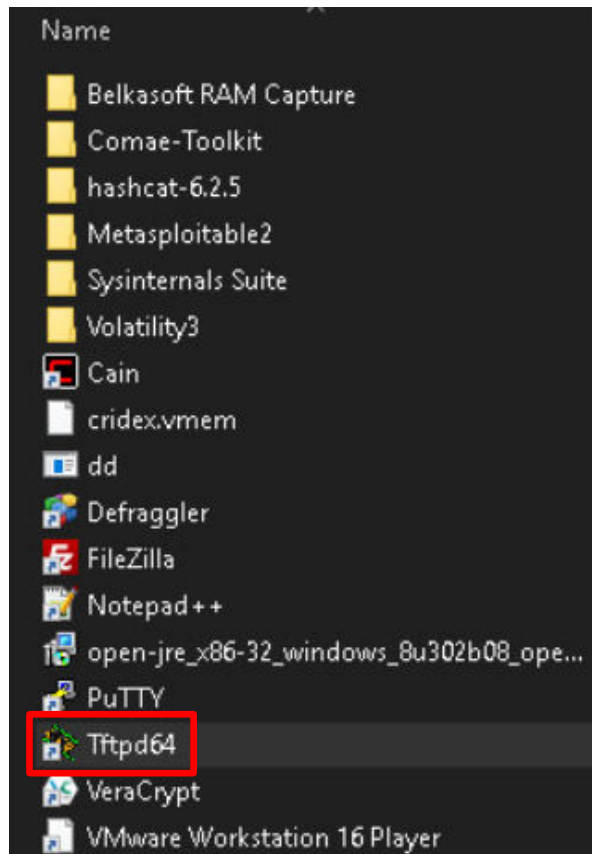


13. Close all open windows.

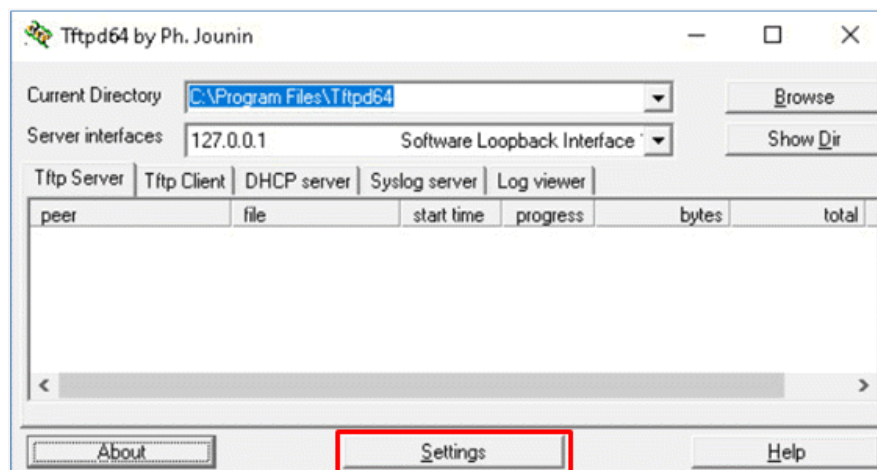
14. On the *WinOS* desktop, double-click on the **Toolbox** folder.



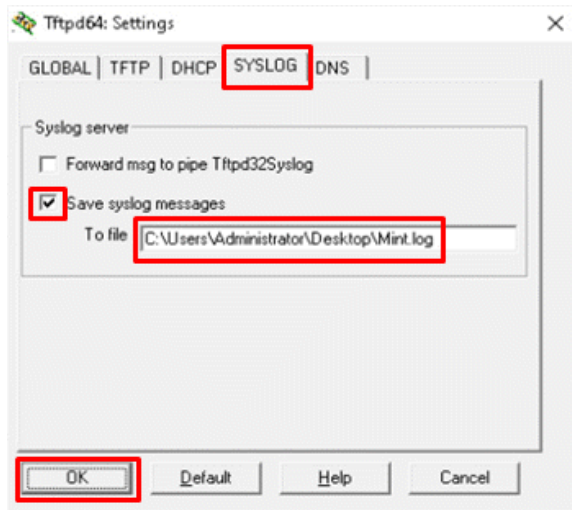
15. In the *Toolbox* folder, you should see the **Tftpd64** program. Double-click to start the *Syslog* server.



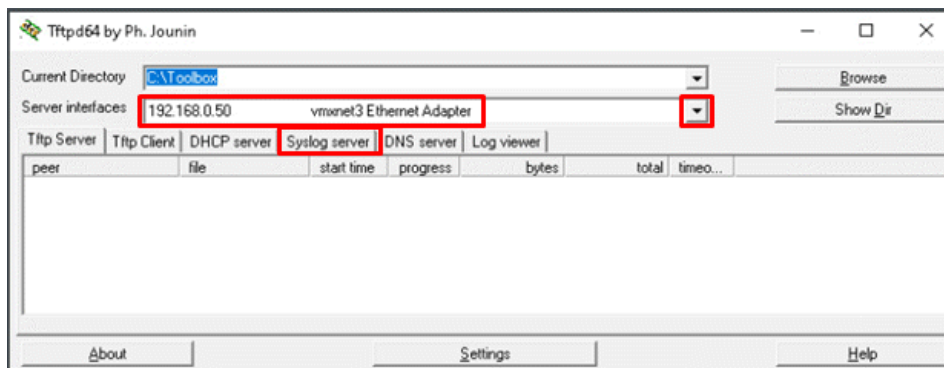
16. By default, *TFTPD64* does not save any logs. To save the information in a file, click on the **Settings** button at the bottom of the window.



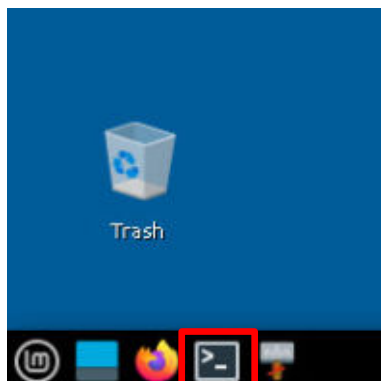
17. In the Tftpd64 Settings window, click the **SYSLOG** tab. Then, click the **Save Syslog Messages** checkbox and finally change the *To File:* to C:\Users\Administrator\Desktop\Mint.log. Finally, click the **OK** button.



18. Click the **Syslog Server** tab. Then change the *Server Interfaces* to **192.168.0.50** by clicking on the list arrow and changing the interface.



19. Return to the **MintOS** computer.
20. If you do not have a terminal window open, click on the **Terminal** icon in the taskbar at the bottom of the screen.



21. Edit the **rsyslog.conf** file by typing the following command. If asked for **[sudo]** password for **sysadmin**, type: **NDGLabpass123!**

```
sudo nano /etc/rsyslog.conf
```

```
File Edit View Terminal Tabs Help
sysadmin@mintos:~$ sudo nano /etc/rsyslog.conf
[sudo] password for sysadmin: *****
```

22. In the previous task, you added a statement to enable sending of log entries to the *UbuntuSRV* computer. You need to change the destination to the *WinOS* computer.

Arrow down and change the address to ***.* @192.168.0.50:514**. This is the IP address of the **WinOS** machine running the **TFTPD64 Syslog Server** and using the default **Syslog UDP** port, **514**.

```
File Edit View Terminal Tabs Help
GNU nano 4.8 /etc/rsyslog.conf
# /etc/rsyslog.conf configuration file for rsyslog
#
# For more information install rsyslog-doc and see
# /usr/share/doc/rsyslog-doc/html/configuration/index.html
#
# Default logging rules can be found in /etc/rsyslog.d/50-default.conf
*.* @192.168.0.50:514
```

23. When finished, press **Ctrl+O** to write the file.

```
^G Get Help ^O Write Out ^W Where Is ^K Cut
^X Exit ^R Read File ^\ Replace ^L Pas
```

24. Press **Enter** to confirm the file name **/etc/rsyslog.conf**.

```
File Name to Write: /etc/rsyslog.conf
^G Get Help ^M-D DOS Format
^C Cancel ^M-M Mac Format
```

25. Press **Ctrl+X** to exit.

```
^G Get Help ^O Write Out ^W Where Is ^K Cut
^X Exit ^R Read File ^\ Replace ^L Pas
```


26. Type the following command to restart the **rsyslog** service:

```
sudo service rsyslog restart
```

```
sysadmin@mintos:~$ sudo service rsyslog restart
```

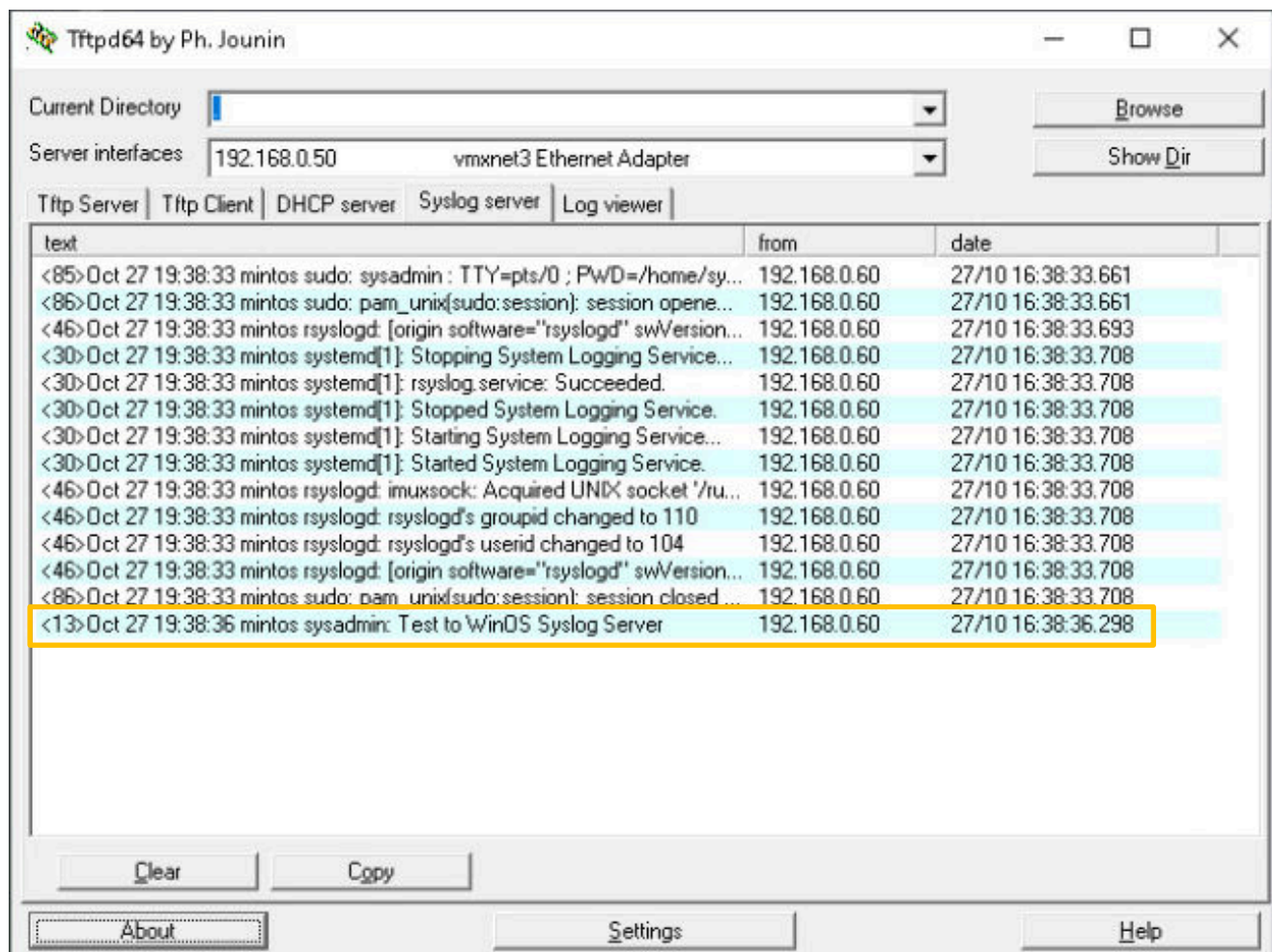
27. Type the following command to send a log command to the **syslog** file on the *WinOS* computer.

```
logger -s "Test to WinOS Syslog Server"
```

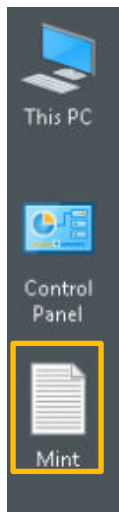
```
sysadmin@mintos:~$ logger -s "Test to WinOS Syslog Server"
<13>Sep 23 15:15:46 sysadmin: Test to WinOS Syslog Server
```

28. Return to the **WinOS** computer.

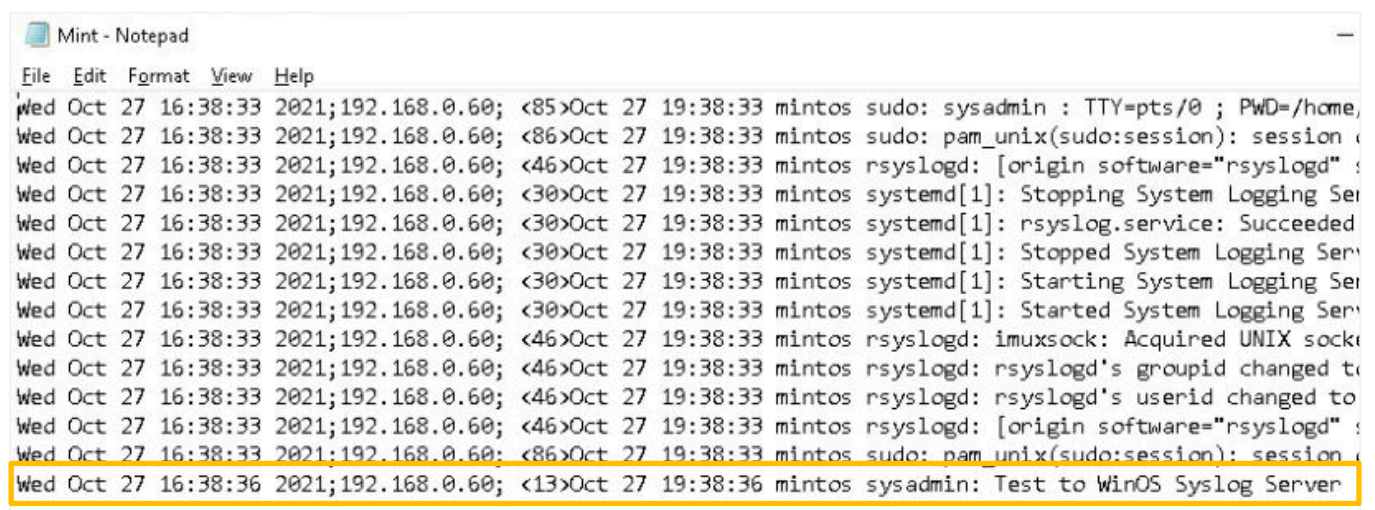
29. Inside of the log screen, notice that **Test to WinOS Syslog Server** was sent from the *MintOS* machine. You have successfully forwarded a syslog file.



30. On the desktop, you will see the **Mint.log** file.



31. Double-click on the **file** to open the log using *Notepad*, and you will see the Syslog entry from the *MintOS* computer.



```
File Edit Format View Help
Wed Oct 27 16:38:33 2021;192.168.0.60; <85>Oct 27 19:38:33 mintos sudo: sysadmin : TTY=pts/0 ; PWD=/home,
Wed Oct 27 16:38:33 2021;192.168.0.60; <86>Oct 27 19:38:33 mintos sudo: pam_unix(sudo:session): session c
Wed Oct 27 16:38:33 2021;192.168.0.60; <46>Oct 27 19:38:33 mintos rsyslogd: [origin software="rsyslogd" s
Wed Oct 27 16:38:33 2021;192.168.0.60; <30>Oct 27 19:38:33 mintos systemd[1]: Stopping System Logging Ser
Wed Oct 27 16:38:33 2021;192.168.0.60; <30>Oct 27 19:38:33 mintos systemd[1]: rsyslog.service: Succeeded
Wed Oct 27 16:38:33 2021;192.168.0.60; <30>Oct 27 19:38:33 mintos systemd[1]: Stopped System Logging Ser
Wed Oct 27 16:38:33 2021;192.168.0.60; <30>Oct 27 19:38:33 mintos systemd[1]: Starting System Logging Ser
Wed Oct 27 16:38:33 2021;192.168.0.60; <30>Oct 27 19:38:33 mintos systemd[1]: Started System Logging Ser
Wed Oct 27 16:38:33 2021;192.168.0.60; <46>Oct 27 19:38:33 mintos rsyslogd: imuxsock: Acquired UNIX socke
Wed Oct 27 16:38:33 2021;192.168.0.60; <46>Oct 27 19:38:33 mintos rsyslogd: rsyslogd's groupid changed to
Wed Oct 27 16:38:33 2021;192.168.0.60; <46>Oct 27 19:38:33 mintos rsyslogd: rsyslogd's userid changed to
Wed Oct 27 16:38:33 2021;192.168.0.60; <46>Oct 27 19:38:33 mintos rsyslogd: [origin software="rsyslogd" s
Wed Oct 27 16:38:33 2021;192.168.0.60; <86>Oct 27 19:38:33 mintos sudo: pam_unix(sudo:session): session c
Wed Oct 27 16:38:36 2021;192.168.0.60; <13>Oct 27 19:38:36 mintos sysadmin: Test to WinOS Syslog Server
```

32. Close all open windows.

4 Log Analysis Using gawk

The *gawk* command in Linux is used to process text files and is used for pattern scanning. The *gawk* command can be used to:

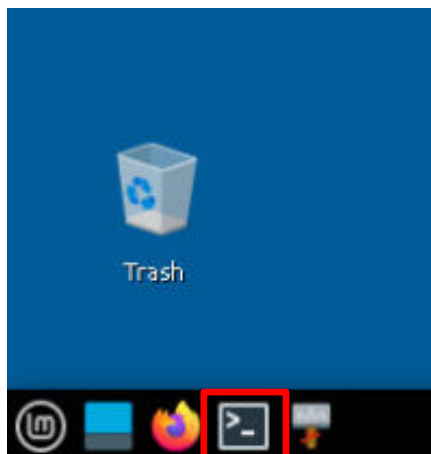
- Scans a file line by line.
- Splits each input line into fields.
- Compares input line/fields to pattern.
- Performs action(s) on matched lines.
- Transform data files.
- Produce formatted reports.
- Format output lines.
- Arithmetic and string operations.
- Conditionals and loops.

<https://www.geeksforgeeks.org/gawk-command-in-linux-with-examples/>

By using *gawk*, a security analyst can more easily analyze log files for interesting information. One way that hackers can cause havoc is by gaining access to a computer and adding users who can then access resources on the network. In this task, you will use **SSH** to add users to the **UbuntuSRV** computer. And then, you will capture the addition of new users in the Syslog files and then use *gawk* to filter the new user entries.

4.1 Creating Groups and Users Remotely

1. Set the focus on the **MintOS** computer.
2. If the terminal window is not open, click on the **Terminal** icon in the taskbar at the bottom of the screen.



3. Type the following command to log in remotely to the **UbuntuSRV**. If prompted with a message asking, *Are you sure you want to continue connecting?* type yes, and when asked for a password, type: NDGLabpass123!

```
ssh 172.16.1.10
```

```
sysadmin@mintos:~$ ssh 172.16.1.10
The authenticity of host '172.16.1.10 (172.16.1.10)' can't be established.
ECDSA key fingerprint is SHA256:7Sg+SzFPN9w2kxkp7hPut1tBh4YkXKA8rK/F6lp+5oY.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.16.1.10' (ECDSA) to the list of known hosts.
sysadmin@172.16.1.10's password:
Last login: Fri Sep 24 20:10:26 2021
```

4. You will see the prompt changed to **sysadmin@ubuntusrv:~\$**.

```
sysadmin@ubuntusrv:~$
```

5. Create the group **anongroup** on the **UbuntuSRV** by typing the following command. When asked for **[sudo] password for sysadmin** type: NDGLabpass123!

```
sudo groupadd anongroup
```

```
sysadmin@ubuntusrv:~$ sudo groupadd anongroup
[sudo] password for sysadmin:
```

6. View the list of groups on the **UbuntuSRV** by typing the following command:

```
getent group
```

```
uidd:x:112:
tcpdump:x:113:
ssh:x:114:
landscape:x:115:
lxd:x:116:sysadmin
systemd-coredump:x:999:
sysadmin:x:1000:
ssl-cert:x:117:
anongroup:x:1001:
```

You should see the group **anongroup** you created at the bottom of the list.

7. Create three new users, **Sheldon**, **Leonard**, and **Penny**, and put them in the **anongroup** by typing the following commands:

```
sudo useradd Sheldon -g anongroup
sudo useradd Leonard -g anongroup
sudo useradd Penny -g anongroup
```

```
sysadmin@ubuntu:~$ sudo useradd Sheldon -g anongroup
sysadmin@ubuntu:~$ sudo useradd Leonard -g anongroup
sysadmin@ubuntu:~$ sudo useradd Penny -g anongroup
```

8. Assign the user **Sheldon** a new password by typing the command. When prompted for a password, use **Password1** and confirm the password.

```
sudo passwd Sheldon
```

```
sysadmin@ubuntu:~$ sudo passwd Sheldon
New password:
Retype new password:
passwd: password updated successfully
```

9. Now assign the new password **Password1** to the user **Leonard**:

```
sudo passwd Leonard
```

```
sysadmin@ubuntu:~$ sudo passwd Leonard
New password:
Retype new password:
passwd: password updated successfully
```

10. And finally, assign the password **Password1** to **Penny**:

```
sudo passwd Penny
```

```
sysadmin@ubuntu:~$ sudo passwd Penny
New password:
Retype new password:
passwd: password updated successfully
```

11. Remain on the **MintOS** computer connected to the UbuntuSRV using **SSH** for the next task.

4.2 Use gawk to Analyze Log Files

1. Change to the directory **/var/log** by typing the following:

```
cd /var/log
```

```
sysadmin@ubuntusrv:~$ cd /var/log
```

2. Type the following command to view the authentication log file:

```
less auth.log
```

```
sysadmin@mintos:/var/log$ less auth.log
```

Just looking at the raw output, the security analyst will quickly realize finding the entries for the addition of the group and user accounts and the password changes would take quite a bit of time. In addition, some items might be missed in the rows and rows of log entries. But, by using *gawk*, you can isolate specific fields in a line of text. *gawk* parses each line of text, looking for a space or a tab between the text, and tags them as **fields**. These fields are numbered **\$1**, **\$2**, **\$3**, and so on.

Looking at a line of output that shows a group was added you see the following:

```
Oct 28 00:53:43 ubuntusrv groupadd[7580]: new group: name=anongroup, GID=1001
```

where:

\$1 = Oct
\$2 = 28
\$3 = 00:53:43
\$4 = ubuntusrv
\$5 = groupadd[7580]:
\$6 = new
\$7 = group:
\$8 = name=anongroup,
\$9 = GID=1001

... a user was added

```
Oct 28 00:57:13 ubuntusrv useradd[7675]: new user: name=Sheldon, UID=1001, GID=1001,
```

where:

\$1 = Oct
\$2 = 28
\$3 = 00:57:13
\$4 = ubuntusrv
\$5 = useradd[7675]:
\$6 = new
\$7 = user:

\$8 = name=Sheldon,
\$9 = UID=1001
\$10 = GID = 1001
 (additional fields are not shown)

... and the user's password changed

```
Oct 28 00:58:02 ubuntu:svr passwd[7714]: pam_unix(passwd:chauthtok): password changed for Sheldon
```

where:

\$1 = Oct
\$2 = 28
\$3 = 00:58:02
\$4 = ubuntu:svr
\$5 = passwd[7714]:
\$6 = pam_unix(passwd:chauthtok):
\$7 = password
\$8 = changed
\$9 = for
\$10 = Sheldon

- To identify groups that have been added, type the following *gawk* command with the **field identifiers** to extract those specific fields for the *auth.log* file. In this case, list the **date** (**\$1**) and (**\$2**), the **time** (**\$3**), the label for the **new group** (**\$6**) and (**\$7**) and the **group name** (**\$8**), and then pipe (**|**) the output through *grep* (Global Regular Expression Print) to limit the output to just new groups by searching for “**new group**”.

```
gawk '{print $1,$2,$3,$6,$7,$8}' auth.log | grep "new group"
```

```
sysadmin@ubuntu:~$ gawk '{print $1,$2,$3,$6,$7,$8}' auth.log | grep "new group"
Oct 28 00:53:43 new group: name=anongroup,
sysadmin@ubuntu:~$
```

- To identify the new users who have been added and when they were added, type the same *gawk* command, only this time *grep* on “**new user**” instead.

```
gawk '{print $1,$2,$3,$6,$7,$8}' auth.log | grep "new user"
```

```
sysadmin@ubuntu:~$ gawk '{print $1,$2,$3,$6,$7,$8}' auth.log | grep "new user"
Oct 28 00:57:13 new user: name=Sheldon,
Oct 28 00:57:21 new user: name=Leonard,
Oct 28 00:57:28 new user: name=Penny,
sysadmin@ubuntu:~$
```

- Finally, to identify users who have had their passwords changed, and more importantly, when their passwords were changed, type the following command:

```
gawk '{print $1,$2,$3,$7,$8,$9,$10}' auth.log | grep "password changed"
```

```
sysadmin@ubuntusrv:/var/log$ gawk '{print $1,$2,$3,$7,$8,$9,$10}' auth.log | grep "password changed"
Oct 28 00:58:02 password changed for Sheldon
Oct 28 00:58:32 password changed for Leonard
Oct 28 00:58:54 password changed for Penny
sysadmin@ubuntusrv:/var/log$
```

Now, the security analyst can quickly discover if rogue accounts have been created or if the user's passwords have been changed without either the user's or administrator's knowledge.

6. Logout from the **SSH** session and return control back to the **MintOS** computer by typing the following command:

```
logout
```

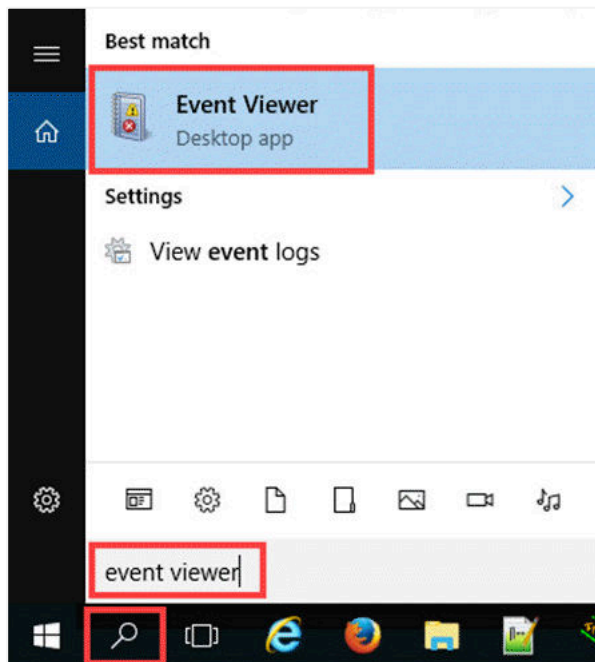
```
sysadmin@ubuntusrv:/var/log$ logout
Connection to 172.16.1.10 closed.
sysadmin@mintos:~$
```

7. Close the terminal session.

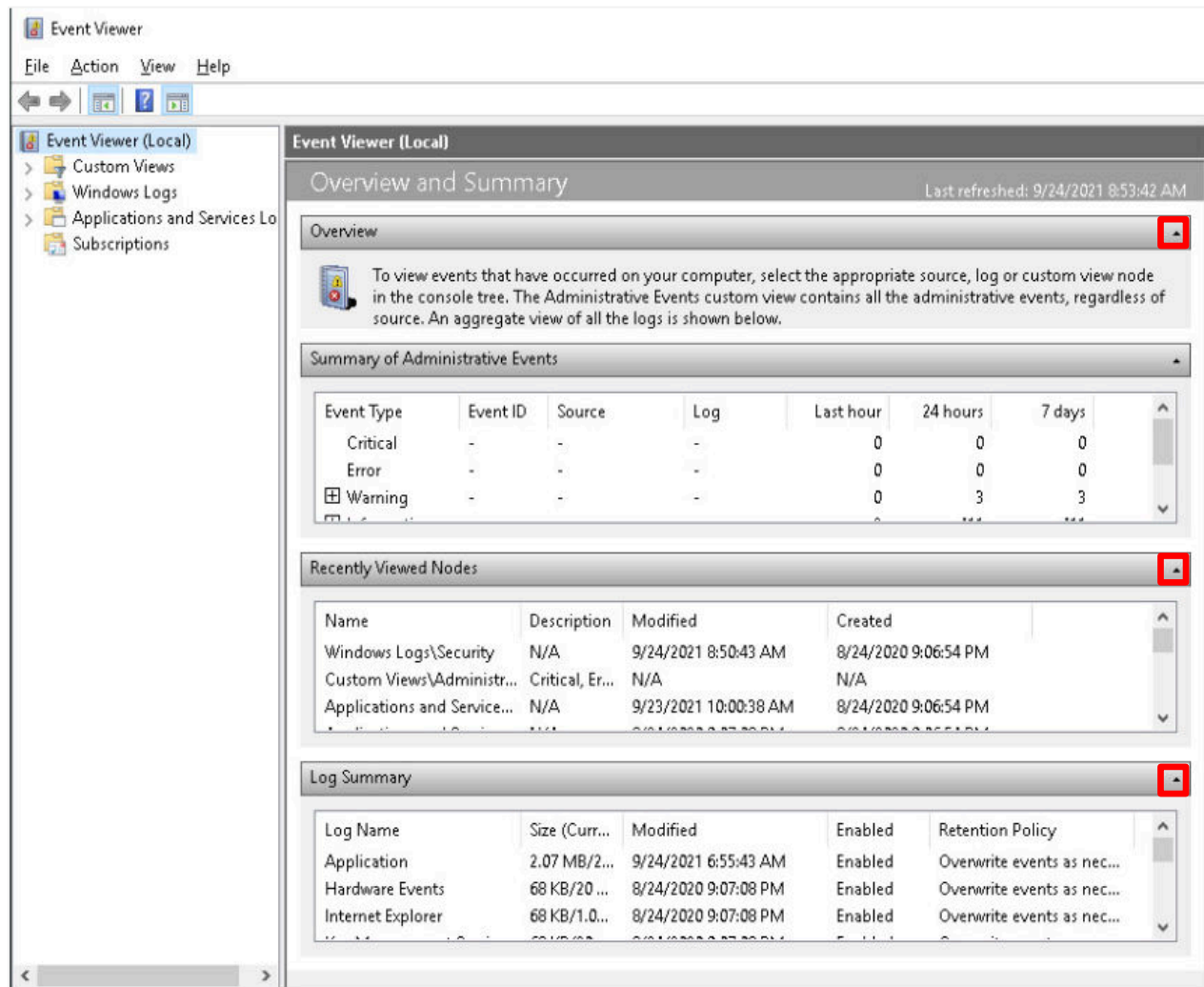
5 Exploring Windows Event Viewer

The *Windows Event Viewer* is a built-in tool included in all modern versions of Windows. It is a useful tool for examining event logs that are organized in a similar manner to syslog. In this task, you will examine the Windows Event Viewer log entries, and explore how to create a custom view to filter the results for easy viewing.




1. Set the focus to the **WinOS** computer.
2. Click on the **Windows Search** icon and type event viewer. Click on **Event Viewer** under *Best Match*.






- Notice the *Event Viewer* appears, and by default, it shows the *Overview and Summary* page. This page displays all events from all Windows logs on the system. The total number of events for each type that has occurred is displayed, along with the number of events of each type that have occurred over the last seven days, the last 24 hours, and the last hour. In the middle pane, click the **small up arrows** at the end of the section headers to minimize everything but the *Summary of Administrative Events* section.



4. Viewing the *Summary of Administrative Events* section allows you to view the diverse types of events that may have occurred on this system. Click the **+** next to the **Warning** events to view the corresponding events that have occurred on this system.

Overview and Summary						
Last refreshed: 8/18/2022 9						
Overview						
Summary of Administrative Events						
Event Type	Event ID	Source	Log	Last hour	24 hours	7 days
Critical	-	-	-	0	0	0
Error	-	-	-	0	0	0
 Warning	-	-	-	1	3	6
 Information	-	-	-	29	355	748
 Audit Success	-	-	-	55	249	579

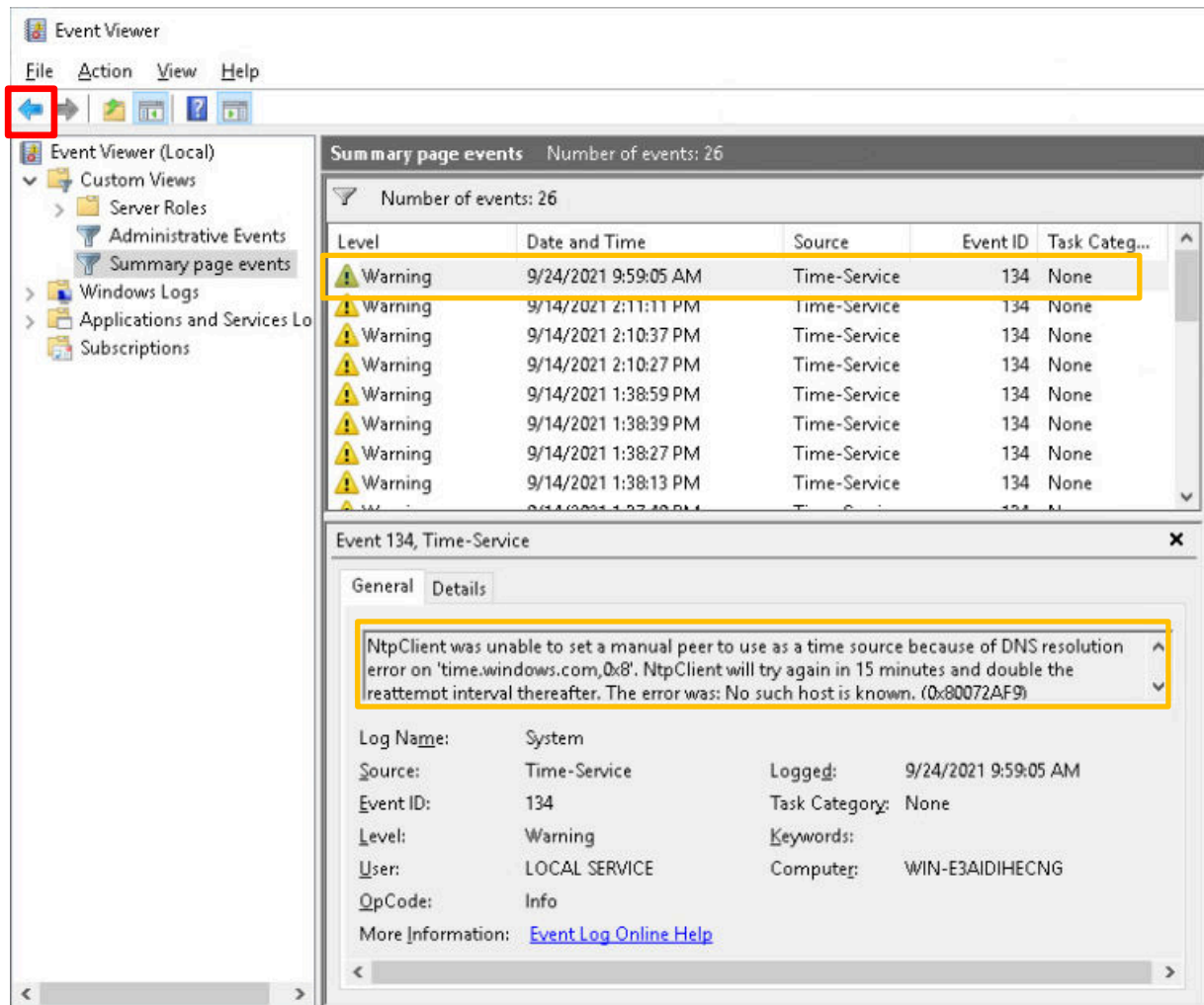
5. Double-click on **Event ID 134, Time-Service** to display detailed information on the event.

Summary of Administrative Events						
Event Type	Event ID	Source	Log	Last hour	24 hours	
Critical	-	-	-	0	0	
Error	-	-	-	0	0	
 Warning	-	-	-	3	3	
	134	Time-Service	System	1	1	
	360	User Device Re...	Microsoft...	1	1	
	1014	DNS Client Eve...	System	1	1	
 Information	-	-	-	220	220	
 Audit Success	-	-	-	220	220	

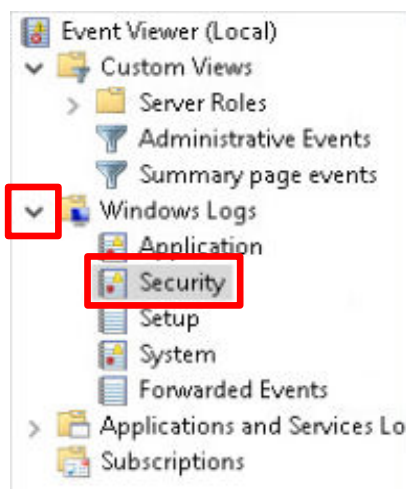


Due to the date change, your events may be different from the example used.

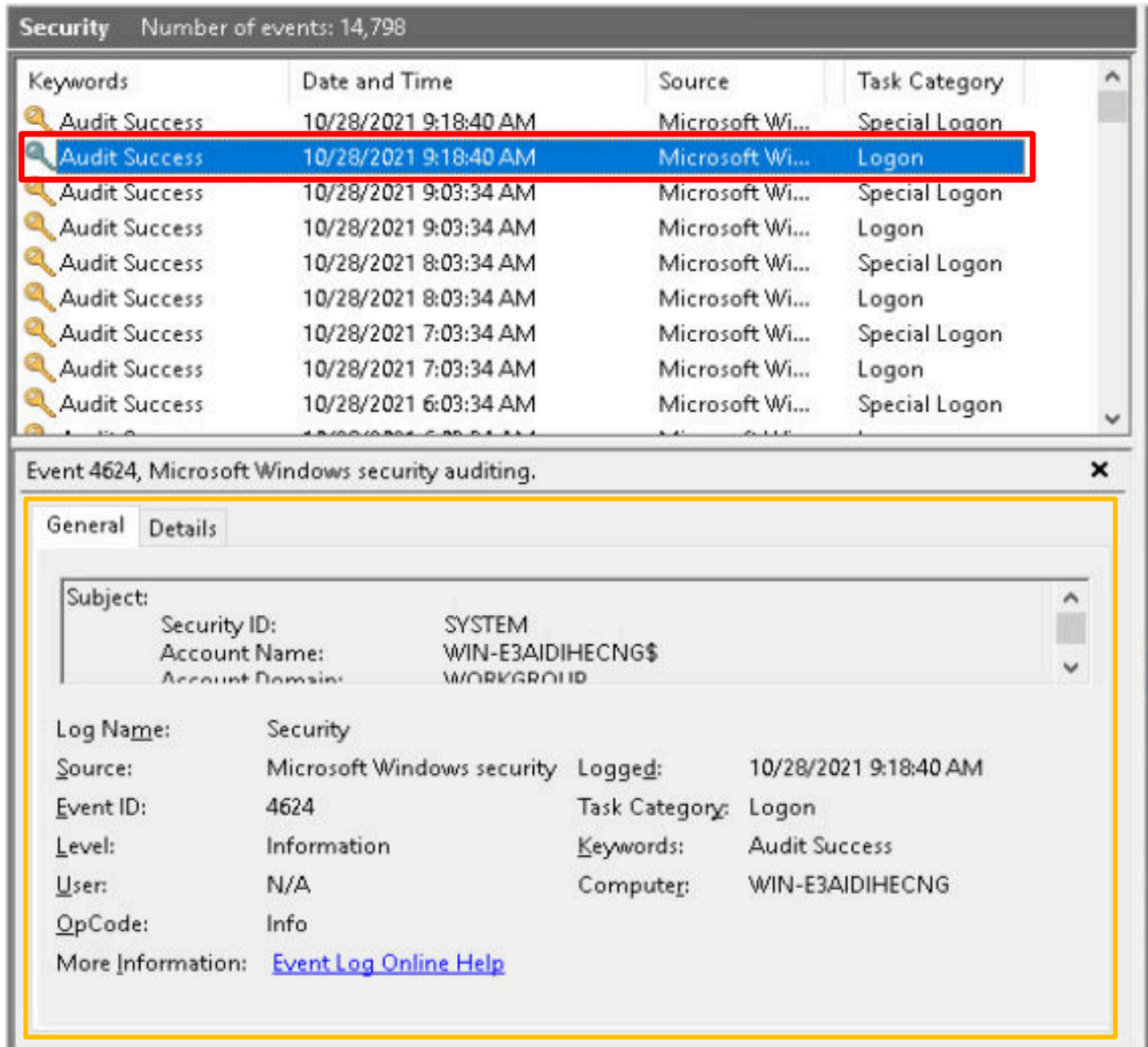
6. More details emerge for the specific event. In this example, the event in which the **Time-Service** is viewed is issuing a **Warning**. Looking at the message in the lower panel shows what causes the warning, in this case, a DNS resolution issue. Click the **back arrow** when finished analyzing the specific event.



7. In the left pane, expand *Windows Logs* by clicking on the **arrow** and then click on **Security**.



8. The *Security Log* shows when either a user or a process successfully logged into Windows. Double-click the **Audit Success** security event that has a *Task Category* of **Logon** to display detailed information on the corresponding event in the lower half of the middle pane.



The screenshot displays the Windows Security Event Viewer. The top pane shows a list of events with columns: Keywords, Date and Time, Source, and Task Category. The second row is highlighted in blue and enclosed in a red rectangle. Below this, the 'Event 4624, Microsoft Windows security auditing.' pane is shown, with the 'Details' tab selected and enclosed in a yellow rectangle. This pane provides detailed information about the selected event.

Keywords	Date and Time	Source	Task Category
Audit Success	10/28/2021 9:18:40 AM	Microsoft Wi...	Special Logon
Audit Success	10/28/2021 9:18:40 AM	Microsoft Wi...	Logon
Audit Success	10/28/2021 9:03:34 AM	Microsoft Wi...	Special Logon
Audit Success	10/28/2021 9:03:34 AM	Microsoft Wi...	Logon
Audit Success	10/28/2021 8:03:34 AM	Microsoft Wi...	Special Logon
Audit Success	10/28/2021 8:03:34 AM	Microsoft Wi...	Logon
Audit Success	10/28/2021 7:03:34 AM	Microsoft Wi...	Special Logon
Audit Success	10/28/2021 7:03:34 AM	Microsoft Wi...	Logon
Audit Success	10/28/2021 6:03:34 AM	Microsoft Wi...	Special Logon

Event 4624, Microsoft Windows security auditing.

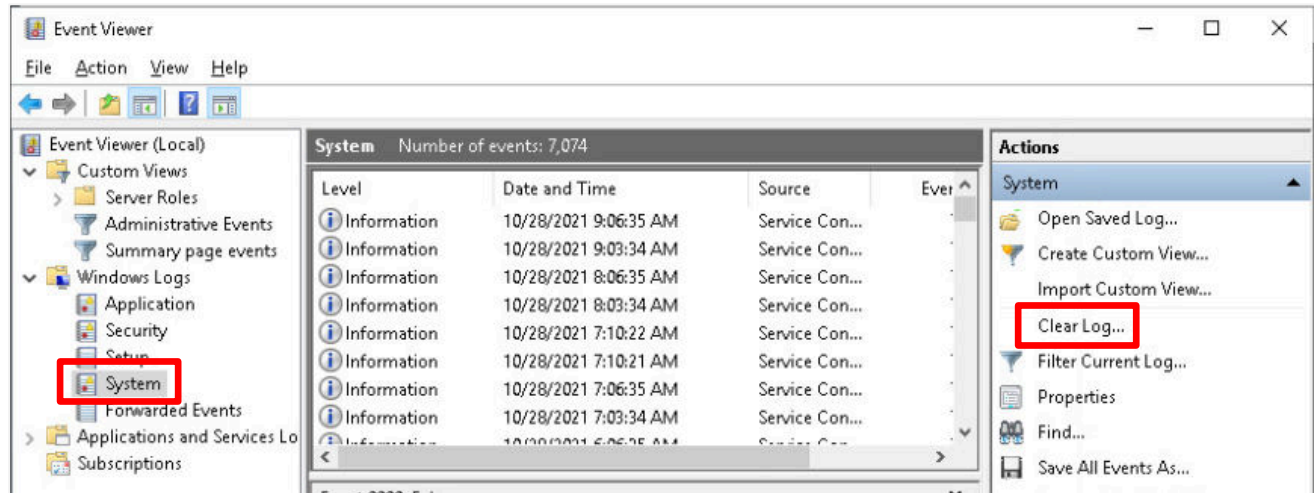
General | **Details**

Subject:
Security ID: SYSTEM
Account Name: WIN-E3AIDIHECNG\$
Account Domain: WORKGROUP

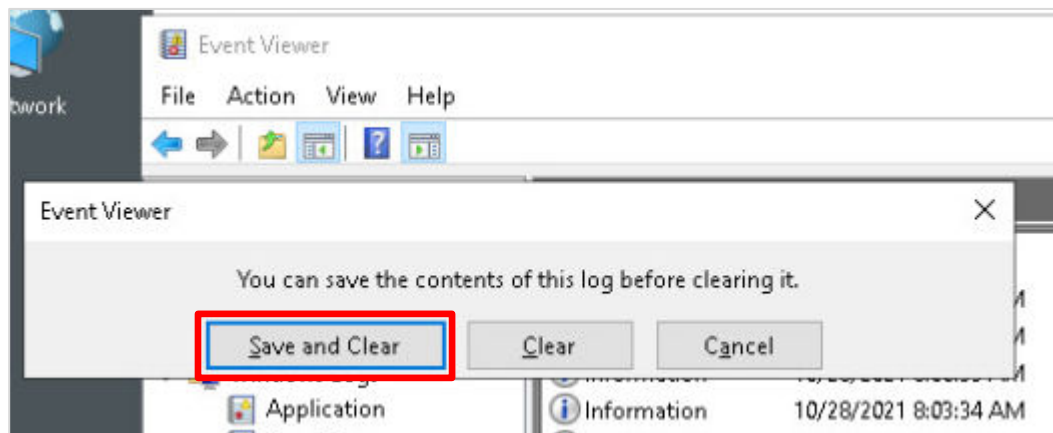
Log Name: Security
Source: Microsoft Windows security
Event ID: 4624
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online Help](#)

Logged: 10/28/2021 9:18:40 AM
Task Category: Logon
Keywords: Audit Success
Computer: WIN-E3AIDIHECNG

9. You are going to generate a simulated **Error Event** in the System Log and view the results. First, clear the System Log to make it easier to find the new **Error Events** by clicking on **System** under **Windows Logs** in the left pane and then clicking on **Clear Log...** in the right pane.

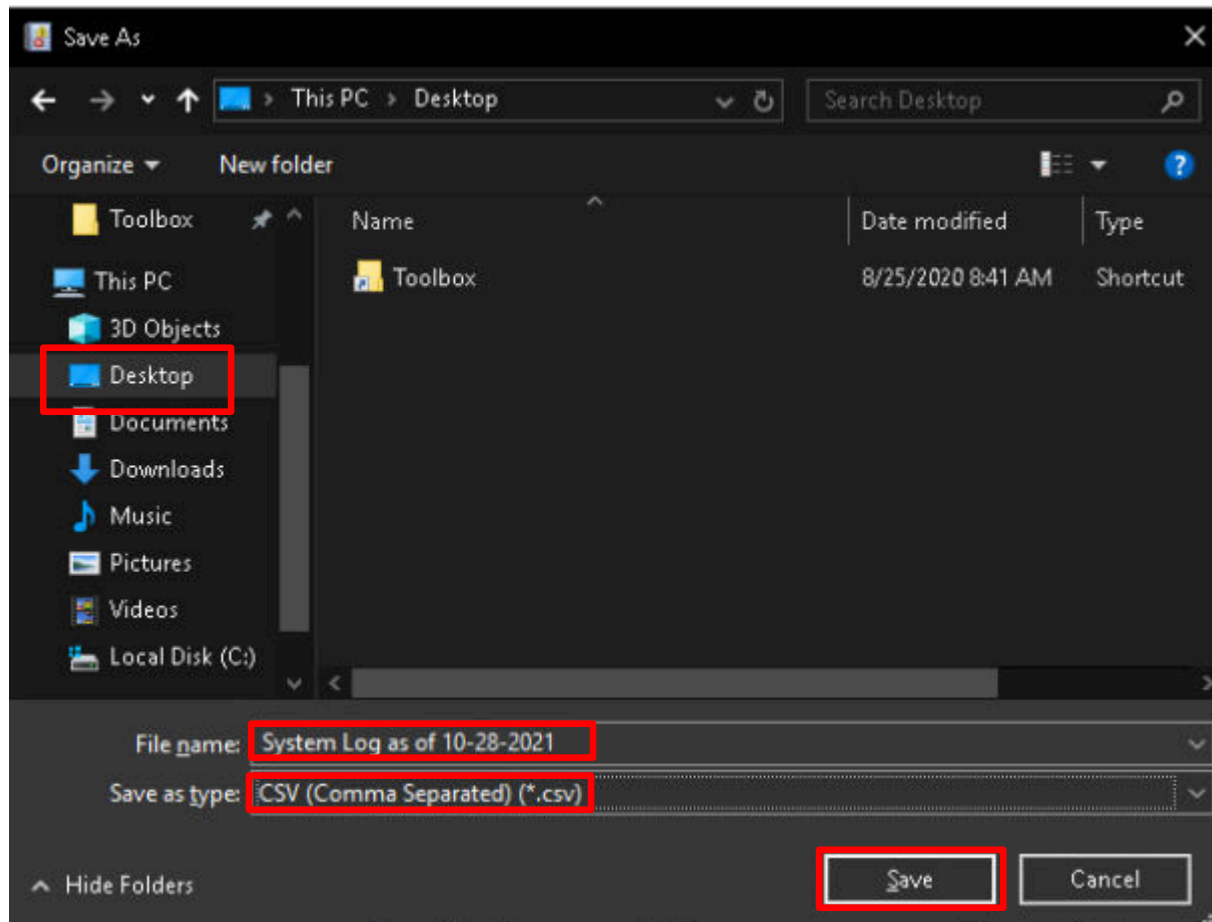


10. It's important for the security analyst to keep the log files for further analysis, documentation, and reference; you will want to save the System log. When you clear the log, Windows informs you that the log can be saved. In the popup window that tells you, *You can save the contents of this log before clearing it*, click the **Save and Clear** button.



11. In the *Save As* window, select **Desktop** as the location, type in the file name: *System Log as of <current date>* (where **<current date>** is the date the log was saved, for example, **10-28-2021**).

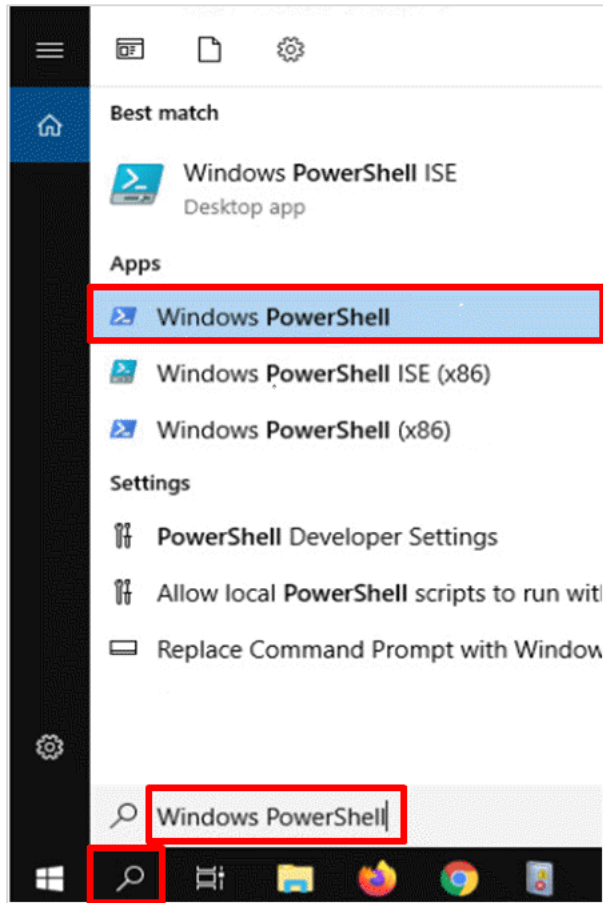
Then, click on the **list arrow** for *Save as Type* and select **CSV**, which will save the log in a format that can be opened and analyzed using a variety of tools. Finally, click the **Save** button. You should see the file on the desktop.



Another good format to save the logs are in **XML** format. **XML** can easily be converted into an **HTML** page which makes viewing more convenient.

12. Minimize the **Event Viewer** window.

13. Click on the **Search** icon and then type **Windows PowerShell**. When the list is displayed, click on **Windows PowerShell**.



14. First, create a fake source. At the *PowerShell* prompt, type the following command:

```
New-EventLog -LogName System -Source "Fake"
```

```
PS C:\Users\Administrator> New-EventLog -LogName System -Source "Fake"  
PS C:\Users\Administrator>
```

15. Next, create a fake event error log with an **Event ID** of **9999** by typing the following command:

```
Write-EventLog -LogName System -Source "Fake" -EntryType Error -EventID 9999
```

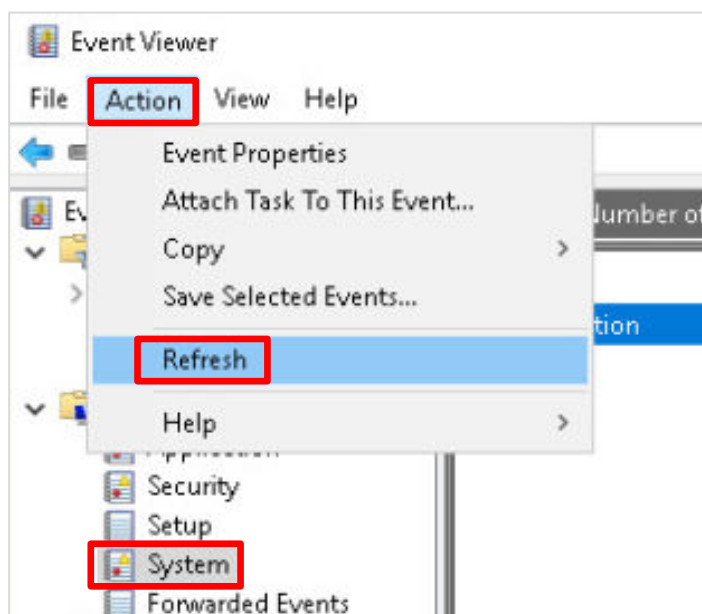
```
PS C:\Users\Administrator> Write-EventLog -LogName System -Source "Fake" -EntryType Error -EventID 9999
```


16. When the prompt for a message appears, type the following message:

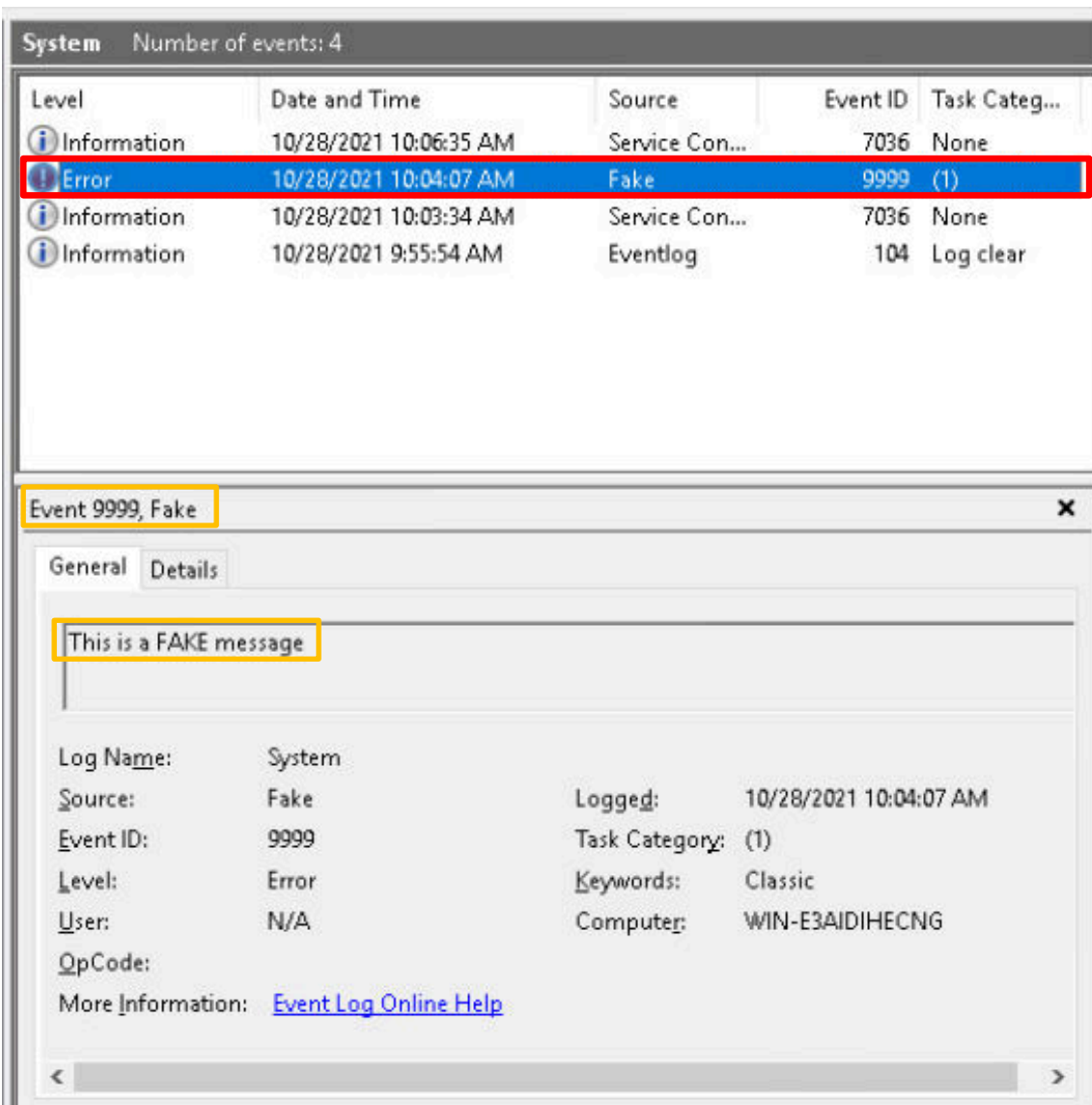
This is a FAKE message

```
PS C:\Users\Administrator> Write-EventLog -LogName System  
  
cmdlet Write-EventLog at command pipeline position 1  
Supply values for the following parameters:  
Message: This is a FAKE message
```

17. Restore the *Event Viewer* window and click on **System** under *Windows Logs*. On the menu bar, click **Action** and click **Refresh** to refresh the logs with any new events and see that the *Error* message is there.



18. Click on the **Error** event, and you will see the Event ID is **9999**, **Fake** and **This is a FAKE message**.



19. This concludes the lab. You may now end the reservation.