Lab 8 Implementing Security Monitoring and Logging

Osamudiamen Eweka

Cyb-605-Z2 Principles of Cybersecurity

Utica University

Introduction

Security event logging and monitoring are crucial for upholding a secure infrastructure and documenting every event from email interactions to access attempts and firewall changes. This detailed record-keeping is vital for tracing all activities within an organization's digital environment. Analyzing these logs, particularly those with sensitive data, helps identify unauthorized access attempts, with findings centralized for in-depth investigation and response. In an era of relentless digital threats, the insights from these logs are key to maintaining a proactive defense (ControlCase, 2023).

This lab exercise guides participants through cybersecurity's nuances in three sections, enhancing their expertise. Section 1 focuses on understanding the Windows Event Viewer, specifically on authentication and authorization events. Section 2 explores the Linux environment, emphasizing identifying failed login attempts and implementing Tripwire for file integrity. The final section, Section 3, involves analyzing Audit Failure events, notably Event ID 5061, simulating the role of a security analyst.

Objective

The purpose of this laboratory report is to furnish a comprehensive description of a multifaceted cybersecurity laboratory that includes both Windows and Linux operating systems. This document will elaborate on various critical aspects such as the monitoring of authentication and authorization activities, analysis of unsuccessful login attempts, the deployment of Tripwire for ensuring file integrity, examination of Audit Failure incidents (specifically Event ID 5061), and the implementation of Snort as an Intrusion Prevention System (IPS).

This exploration aims to elucidate the methodologies and outcomes associated with these security measures, thereby providing insights into their effectiveness in safeguarding the digital infrastructure within mixed operating system environments. Through detailed analysis and practical application, the report will offer a holistic view of the operational capabilities of these tools and techniques in the context of cybersecurity defense mechanisms.

Lab Setup

The laboratory employs a combination of virtual machines, software applications, and utilities to create a variety of cyber-attack simulations, providing participants with a practical and engaging educational experience. To facilitate these simulations, the lab utilizes a selection of hardware and software resources designed to replicate real-world cybersecurity challenges effectively. This setup enables learners to interact with and respond to these scenarios in a controlled environment, thereby enhancing their understanding and skills in cyber defense mechanisms.

- vWorkstation
- Switch01
- PuTTY
- Snort
- Event Viewer (Windows)
- Tripwire
- pfSense

Section 1

Part 1: Identify Failed Logon Attempts on Windows Systems

In the current laboratory demonstration, attention is centered on employing the Windows Event Viewer for the detection of unsuccessful logon attempts. This utility is paramount within Windows frameworks for monitoring system and application events, vital for incident responses and forensic inquiries. It serves as a critical interface for logging files, particularly with security implications, and its utility is magnified when employed alongside Security Information and Event Management (SIEM) and Security Orchestration, Automation, and Response (SOAR) systems. These integrations enable the derivation of immediate analytical insights into anomalous activities that could indicate security breaches.

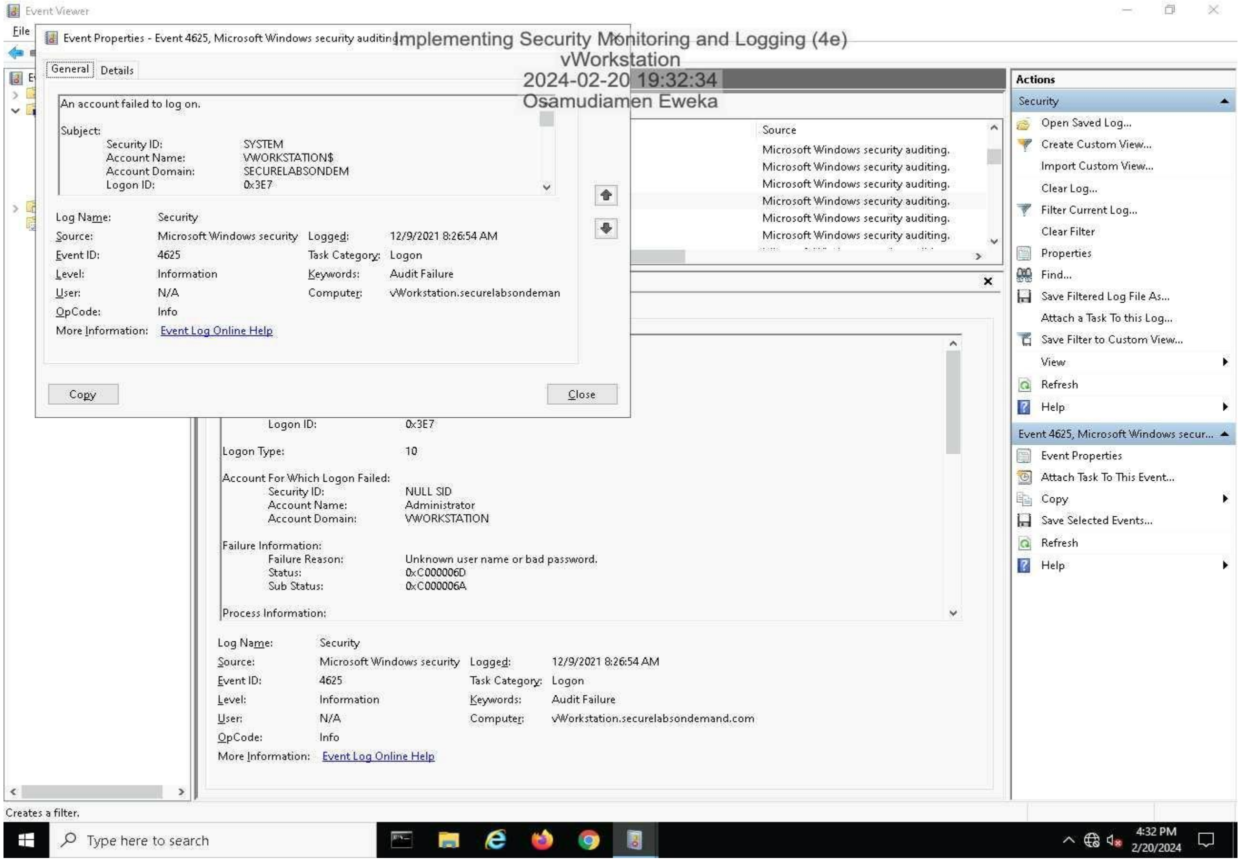
Participants will be instructed to initiate the Windows Event Viewer, proceed to the Security log section, and implement a filter for Event ID 4625, which is an identifier for logon failures. This exercise aims to accentuate the importance of analyzing such events to discern patterns indicative of security threats, such as brute force attacks. The session will stress the necessity of instituting defensive measures, for example, account lockout protocols and the deployment of SIEM systems to track and alert on repeated unsuccessful logon attempts.

The focus will then shift to an in-depth evaluation of the initial failed logon attempt, extracting details like the TargetUserName, source IP address, and the communication port, to underscore the richness of data available for scrutiny. Participants will gain insights into how such comprehensive information can be pivotal in detecting and mitigating security threats.

Concluding this part, the workshop will facilitate an examination of the Event Properties dialog box, which contains critical data about the failed logon attempt. Participants will be advised to carefully inspect the General and Details tabs to glean relevant data points. The lab intends to reinforce the practical aspect of capturing the Security Event Properties dialog box on the vWorkstation as a screenshot for documentation, which is an essential practice for maintaining records in cybersecurity investigations.

Figure 1

Make a screen capture showing the Security Event Properties dialog box on the vWorkstation.



Implementing Security Monitoring and Logging (4e)
vWorkstation
2024-02-20 19:32:34
Osamudiamen Eweka

Microsoft Windows security auditing.
Microsoft Windows security auditing.
Microsoft Windows security auditing.
Microsoft Windows security auditing.
Microsoft Windows security auditing.
Microsoft Windows security auditing.

```
An account failed to log on.

Subject:
        Security ID:            SYSTEM
        Account Name:           VWORKSTATION$
        Account Domain:         SECURELABSONDEM
        Logon ID:               0x3E7

Log Name:          Security
Source:            Microsoft Windows security   Logged:          12/9/2021 8:26:54 AM
Event ID:          4625                         Task Category:   Logon
Level:             Information                  Keywords:        Audit Failure
User:              N/A                          Computer:        vWorkstation.securelabsondeman
OpCode:            Info
More Information:  Event Log Online Help
```

```
        Logon ID:               0x3E7

Logon Type:                     10

Account For Which Logon Failed:
        Security ID:            NULL SID
        Account Name:           Administrator
        Account Domain:         VWORKSTATION

Failure Information:
        Failure Reason:         Unknown user name or bad password.
        Status:                 0xC000006D
        Sub Status:             0xC000006A

Process Information:

Log Name:          Security
Source:            Microsoft Windows security   Logged:          12/9/2021 8:26:54 AM
Event ID:          4625                         Task Category:   Logon
Level:             Information                  Keywords:        Audit Failure
User:              N/A                          Computer:        vWorkstation.securelabsondemand.com
OpCode:            Info
More Information:  Event Log Online Help
```

Note. The screen capture referenced is presumed to display the Security Event Properties dialog box on the vWorkstation, capturing indispensable data from the earliest recorded logon failure on December 9, 2021 (Jones & Bartlett, 2024). The dialog box is expected to present exhaustive details such as the event's timestamp, the rationale behind the logon failure, the target username, the originating IP address, the utilized port, and the initiating process. This information serves as a cornerstone for a detailed and informed cybersecurity analysis.

Section 1

Part 2: Monitor Network Activity with Snort

In the current laboratory exercise, attendees are introduced to the sophisticated domain of network security through the lens of Snort, a potent network intrusion detection system (IDS). As a preeminent tool for scrutinizing network traffic, Snort empowers users to detect, categorize, and potentially thwart cyber threats in real-time. The session underscores the indispensability of advanced IDS solutions like Snort to acquire an exhaustive perspective of network security dynamics.

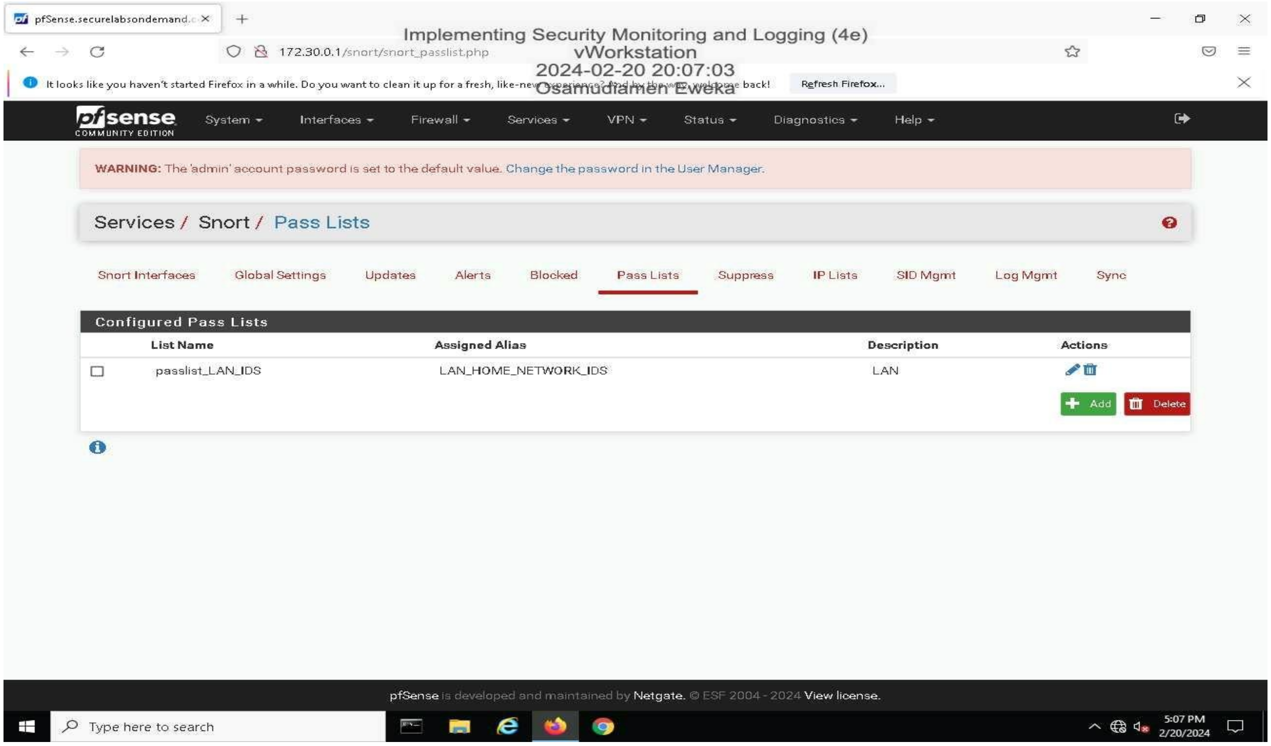
The activity encourages participants to activate Snort's surveillance mechanisms across internal subnetworks, particularly concentrating on the LAN and DMZ. These areas are pivotal within the lab's network infrastructure. The session's guidance encompasses the meticulous configuration of Snort's operational parameters, the formulation of pass lists that differentiate between internal and external network transmissions, and the refinement of detection rules aimed at bolstering network defenses.

The journey continues as participants navigate the pfSense's Web GUI, fine-tuning Snort's settings, authenticating rule sets, and crafting a pass list specifically for the LAN subnetwork. Participants then engage in enhancing Snort with a new interface tailored for the LAN, which authorizes Snort to channel alerts to the system's log. The pedagogic initiative acquaints participants with the concept of pass lists, instrumental for discerning internal network traffic from external ingress. Through hands-on experience, participants calibrate Snort to meticulously monitor the LAN interface, utilizing tailored rulesets adept at managing ICMP traffic. The culmination of this segment is a diagnostic test where participants deploy pfSense's ping function to instigate ICMP alerts, which Snort is configured to log. This real-world test validates Snort's capabilities in identifying and logging network intrusions.

The final segment of the lab is dedicated to capturing and documenting the active status of Snort on the LAN interface, as well as the successful ICMP ping attempts. This step cements the utility of Snort as a practical solution for real-time network monitoring and intrusion detection. The lab exercise thus equips participants with not only a theoretical framework for understanding network intrusions but also the practical skills necessary for deploying and leveraging Snort for effective network security management. The hands-on approach ensures that participants leave with a thorough grasp of network security monitoring, prepared to implement Snort in various network scenarios.

Figure 2

Make a screen capture showing the updated Pass Lists page.



Implementing Security Monitoring and Logging (4e)
vWorkstation
2024-02-20 20:07:03
Osamudiamen Eweka

172.30.0.1/snort/snort_passlist.php

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Services / Snort / Pass Lists

Configured Pass Lists

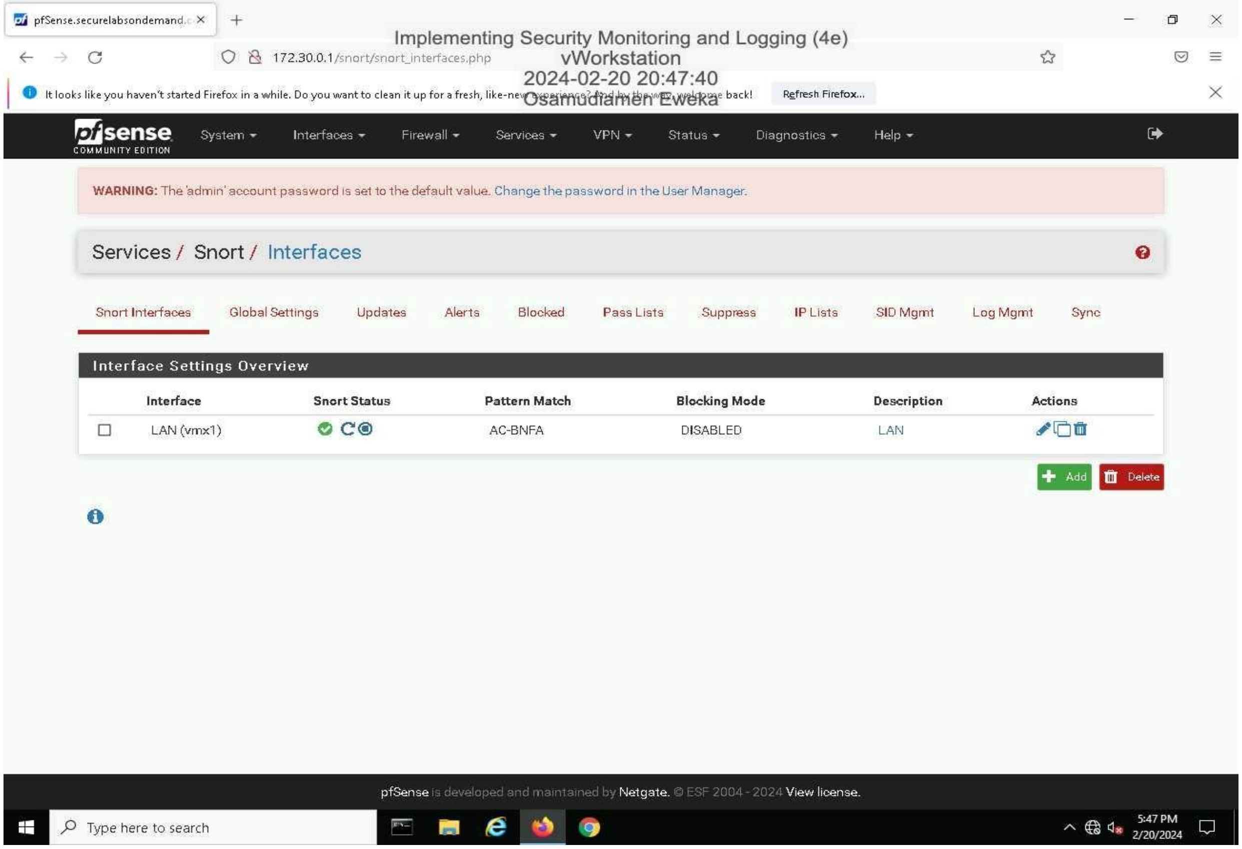
	List Name	Assigned Alias	Description	Actions
□	passlist_LAN_IDS	LAN_HOME_NETWORK_IDS	LAN	

pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

Note. The screenshot above displays the refreshed Pass Lists page within the vWorkstation (Jones & Bartlett, 2024). After investigating unsuccessful login attempts via the Windows Event Viewer, users are instructed to take this screenshot as a record of any modifications or new entries in the Pass Lists. The Pass Lists page plays a pivotal role in monitoring and controlling user access, making this image a crucial depiction of the updates applied throughout the laboratory exercise.

Figure 3

Make a screen capture showing the active Snort status on the LAN interface.



Implementing Security Monitoring and Logging (4e)
vWorkstation
2024-02-20 20:47:40
Osamudiamen Eweka

172.30.0.1/snort/snort_interfaces.php

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Services / Snort / Interfaces

Interface Settings Overview

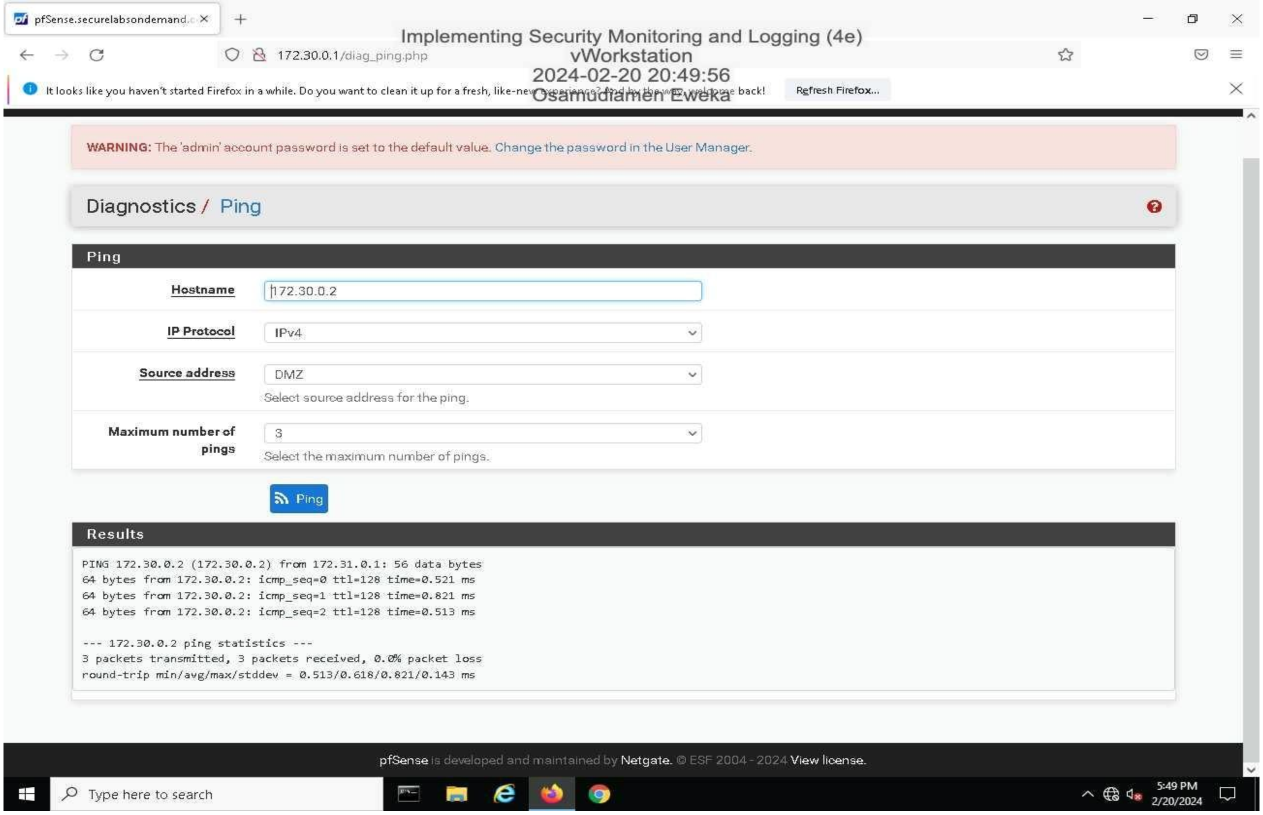
	Interface	Snort Status	Pattern Match	Blocking Mode	Description	Actions
□	LAN (vmx1)	✓	AC-BNFA	DISABLED	LAN	

pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

Note. The image above captures the present condition of Snort's operations on the Local Area Network (LAN) interface (Jones & Bartlett, 2024). After starting Snort, participants are encouraged to take this screenshot, providing a visual account of the live monitoring and intrusion detection system's operational status. This illustration demonstrates the practical aspect of setting up and overseeing Snort, serving as a concrete documentation of its operational state on the LAN interface.

Figure 4

Make a screen capture showing the successful ping results.



Implementing Security Monitoring and Logging (4e)
vWorkstation
2024-02-20 20:49:56
Osamudiamen Eweka

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Diagnostics / Ping

Ping

Hostname: 172.30.0.2

IP Protocol: IPv4

Source address: DMZ
Select source address for the ping.

Maximum number of pings: 3
Select the maximum number of pings.

Ping

Results

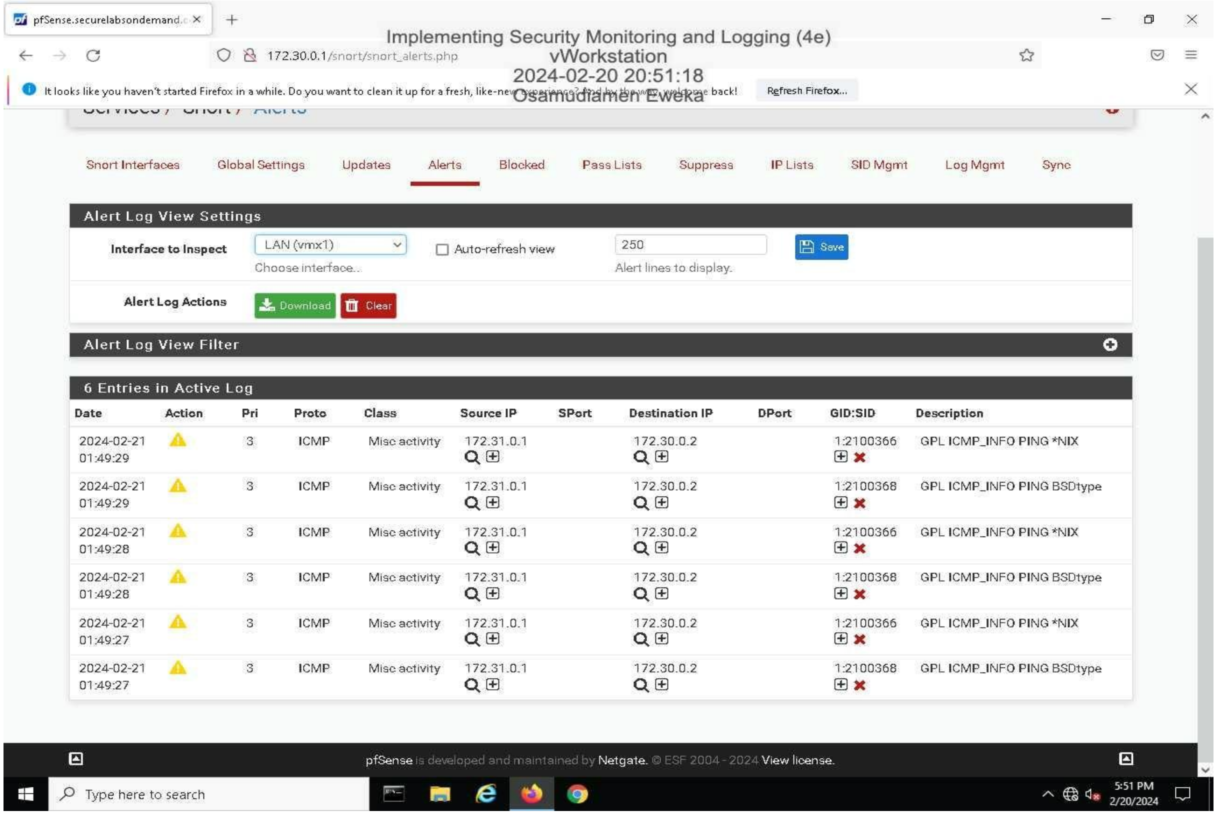
```
PING 172.30.0.2 (172.30.0.2) from 172.31.0.1: 56 data bytes
64 bytes from 172.30.0.2: icmp_seq=0 ttl=128 time=0.521 ms
64 bytes from 172.30.0.2: icmp_seq=1 ttl=128 time=0.821 ms
64 bytes from 172.30.0.2: icmp_seq=2 ttl=128 time=0.513 ms

--- 172.30.0.2 ping statistics ---
3 packets transmitted, 3 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.513/0.618/0.821/0.143 ms
```

pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

Note. The screenshot provided above illustrates the successful results of executing the ping command (Jones & Bartlett, 2024). After examining unsuccessful login attempts through the Windows Event Viewer, participants are advised to document this instance with a screenshot, thereby visually recording the effective completion of the ping operation. This image acts as a definitive visual proof of network connectivity, highlighting the real-world application of laboratory concepts about system communication.

Figure 5

Make a screen capture showing the ICMP alerts in the Snort Active Log.



Implementing Security Monitoring and Logging (4e)
vWorkstation
2024-02-20 20:51:18
Osamudiamen Eweka

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Alert Log View Settings

Interface to Inspect: LAN (vmx1) — Choose interface...

□ Auto-refresh view

250 — Alert lines to display.

Alert Log Actions

Alert Log View Filter

6 Entries in Active Log

Date	Action	Pri	Proto	Class	Source IP	SPort	Destination IP	DPort	GID:SID	Description
2024-02-21 01:49:29	⚠	3	ICMP	Misc activity	172.31.0.1		172.30.0.2		1:2100366	GPL ICMP_INFO PING *NIX
2024-02-21 01:49:29	⚠	3	ICMP	Misc activity	172.31.0.1		172.30.0.2		1:2100368	GPL ICMP_INFO PING BSDtype
2024-02-21 01:49:28	⚠	3	ICMP	Misc activity	172.31.0.1		172.30.0.2		1:2100366	GPL ICMP_INFO PING *NIX
2024-02-21 01:49:28	⚠	3	ICMP	Misc activity	172.31.0.1		172.30.0.2		1:2100368	GPL ICMP_INFO PING BSDtype
2024-02-21 01:49:27	⚠	3	ICMP	Misc activity	172.31.0.1		172.30.0.2		1:2100366	GPL ICMP_INFO PING *NIX
2024-02-21 01:49:27	⚠	3	ICMP	Misc activity	172.31.0.1		172.30.0.2		1:2100368	GPL ICMP_INFO PING BSDtype

pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

Note. The screenshot above showcases the detection of ICMP alerts in the Snort Active Log. After setting up and initiating Snort, participants are guided to take this screenshot as a visual record of the intrusion detection system's reaction to ICMP traffic (Jones & Bartlett, 2024). This image serves as solid proof of Snort's capability to monitor and generate alerts for potential security risks, demonstrating the system's proficiency in identifying and documenting network occurrences.

Section 2

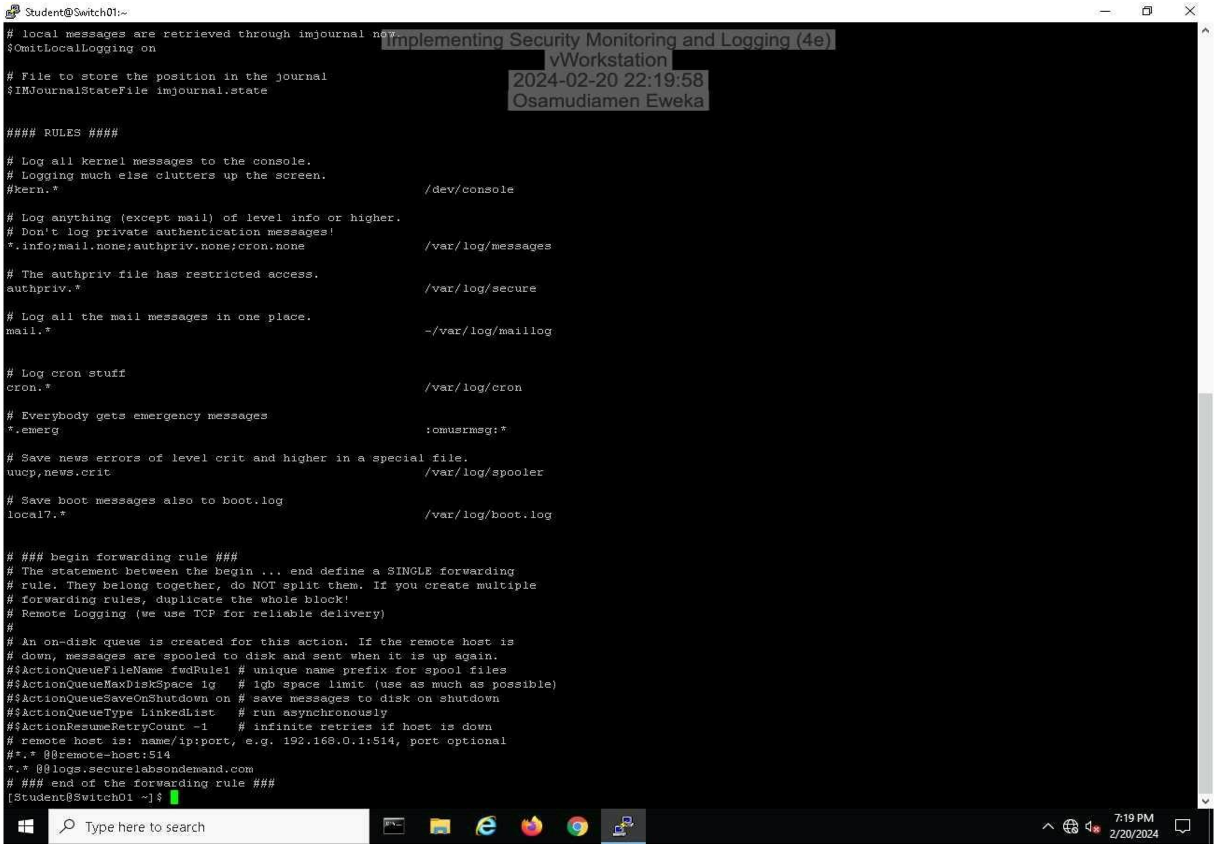
Part 1: Identify Failed Logon Attempts on Linux Systems

Section 2 begins with a shift to a Linux setting, focusing on identifying failed login attempts and setting up Tripwire for monitoring file integrity. The Linux logging system is highlighted for its modular design and syslog standard compliance. Linux logs, often housed in /var/log, differ in structure based on the system's role, such as a web server or network device. The lab introduces Switch01, a Linux-based network switch, accessed via PuTTY. Instructions are given for configuring remote logging to an external syslog server to enhance security and aid forensic analysis.

In Part 1, participants use PuTTY to connect to Switch01 and configure it to send logs to a remote server, editing the rsyslog.conf file via vi Editor. This step is vital for log security and forensic processes. After editing, they document the new rsyslog.conf configuration and restart the syslog service to apply the changes. A simulated brute force attack is then conducted to generate log entries, which are searched using grep. Commands like sudo lastb and sudo tail are used to review failed login attempts and recent log activity.

Figure 6

Make a screen capture showing the edited rsyslog.conf file.

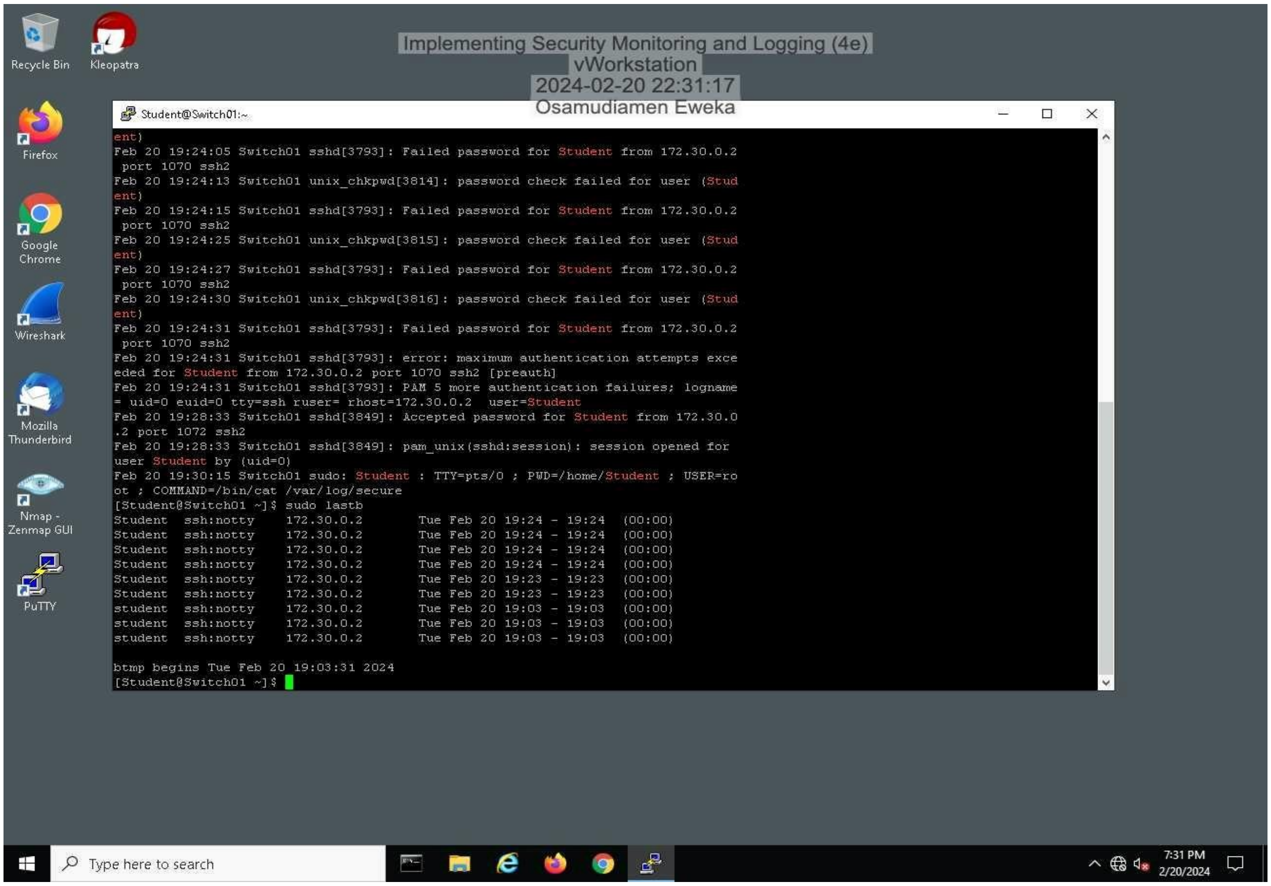


```
# local messages are retrieved through imjournal now.
$OmitLocalLogging on

# File to store the position in the journal
$IMJournalStateFile imjournal.state


#### RULES ####

# Log all kernel messages to the console.
# Logging much else clutters up the screen.
#kern.*                                                 /dev/console

# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;authpriv.none;cron.none                /var/log/messages

# The authpriv file has restricted access.
authpriv.*                                              /var/log/secure

# Log all the mail messages in one place.
mail.*                                                  -/var/log/maillog


# Log cron stuff
cron.*                                                  /var/log/cron

# Everybody gets emergency messages
*.emerg                                                 :omusrmsg:*

# Save news errors of level crit and higher in a special file.
uucp,news.crit                                          /var/log/spooler

# Save boot messages also to boot.log
local7.*                                                /var/log/boot.log


# ### begin forwarding rule ###
# The statement between the begin ... end define a SINGLE forwarding
# rule. They belong together, do NOT split them. If you create multiple
# forwarding rules, duplicate the whole block!
# Remote Logging (we use TCP for reliable delivery)
#
# An on-disk queue is created for this action. If the remote host is
# down, messages are spooled to disk and sent when it is up again.
#$ActionQueueFileName fwdRule1 # unique name prefix for spool files
#$ActionQueueMaxDiskSpace 1g   # 1gb space limit (use as much as possible)
#$ActionQueueSaveOnShutdown on # save messages to disk on shutdown
#$ActionQueueType LinkedList   # run asynchronously
#$ActionResumeRetryCount -1    # infinite retries if host is down
# remote host is: name/ip:port, e.g. 192.168.0.1:514, port optional
#*.* @@remote-host:514
*.* @@logs.securelabsondemand.com
# ### end of the forwarding rule ###
[Student@Switch01 ~]$
```

Note. Figure 6 should display the rsyslog.conf file edit on Switch01, where all messages are directed to @@logs.securelabsondemand.com, ensuring secure log transfer to a remote server for improved security and forensic review (Jones & Bartlett, 2024).

Figure 7

Make a screen capture showing the failed login attempts.



Implementing Security Monitoring and Logging (4e)
vWorkstation
2024-02-20 22:31:17
Osamudiamen Eweka

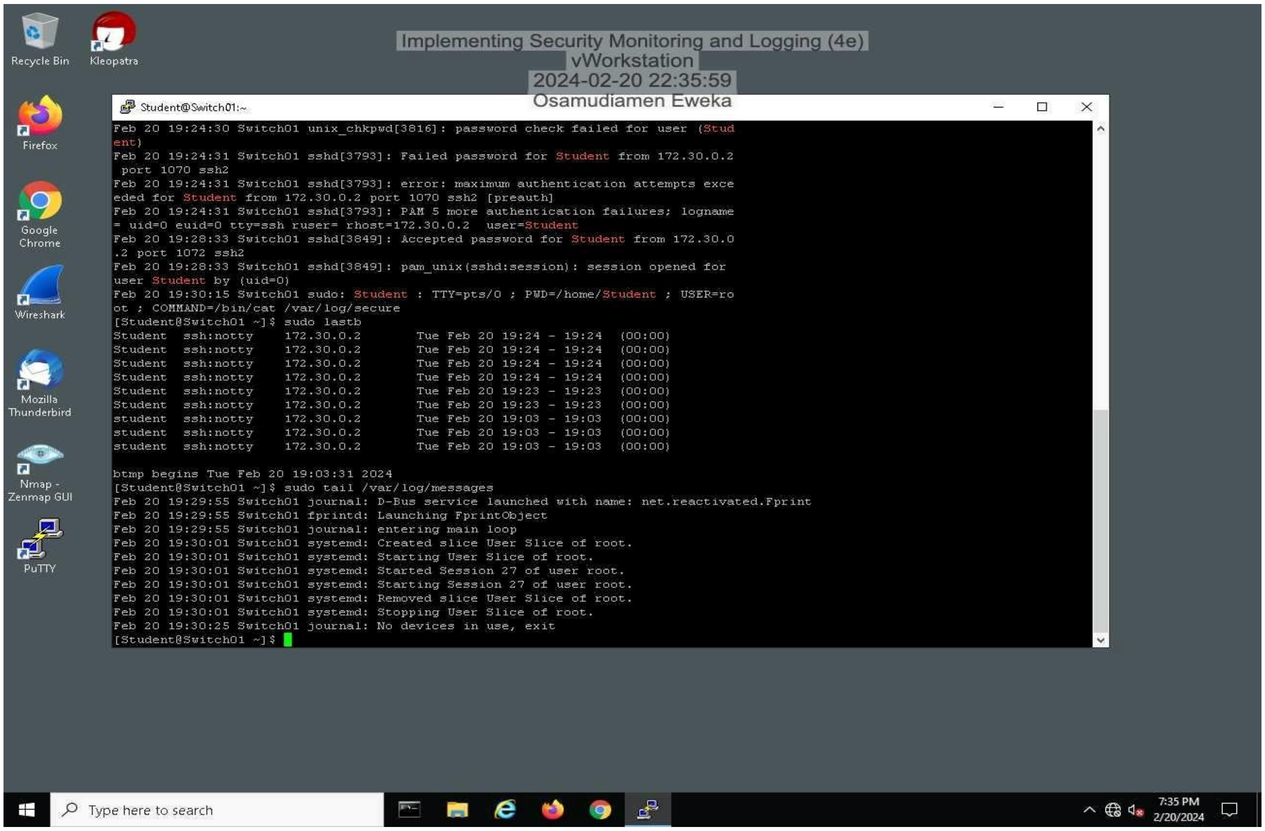
Student@Switch01:~

```
ent)
Feb 20 19:24:05 Switch01 sshd[3793]: Failed password for Student from 172.30.0.2
 port 1070 ssh2
Feb 20 19:24:13 Switch01 unix_chkpwd[3814]: password check failed for user (Stud
ent)
Feb 20 19:24:15 Switch01 sshd[3793]: Failed password for Student from 172.30.0.2
 port 1070 ssh2
Feb 20 19:24:25 Switch01 unix_chkpwd[3815]: password check failed for user (Stud
ent)
Feb 20 19:24:27 Switch01 sshd[3793]: Failed password for Student from 172.30.0.2
 port 1070 ssh2
Feb 20 19:24:30 Switch01 unix_chkpwd[3816]: password check failed for user (Stud
ent)
Feb 20 19:24:31 Switch01 sshd[3793]: Failed password for Student from 172.30.0.2
 port 1070 ssh2
Feb 20 19:24:31 Switch01 sshd[3793]: error: maximum authentication attempts exce
eded for Student from 172.30.0.2 port 1070 ssh2 [preauth]
Feb 20 19:24:31 Switch01 sshd[3793]: PAM 5 more authentication failures; logname
= uid=0 euid=0 tty=ssh ruser= rhost=172.30.0.2  user=Student
Feb 20 19:28:33 Switch01 sshd[3849]: Accepted password for Student from 172.30.0
.2 port 1072 ssh2
Feb 20 19:28:33 Switch01 sshd[3849]: pam_unix(sshd:session): session opened for
user Student by (uid=0)
Feb 20 19:30:15 Switch01 sudo: Student : TTY=pts/0 ; PWD=/home/Student ; USER=ro
ot ; COMMAND=/bin/cat /var/log/secure
[Student@Switch01 ~]$ sudo lastb
Student  ssh:notty    172.30.0.2       Tue Feb 20 19:24 - 19:24  (00:00)
Student  ssh:notty    172.30.0.2       Tue Feb 20 19:24 - 19:24  (00:00)
Student  ssh:notty    172.30.0.2       Tue Feb 20 19:24 - 19:24  (00:00)
Student  ssh:notty    172.30.0.2       Tue Feb 20 19:24 - 19:24  (00:00)
Student  ssh:notty    172.30.0.2       Tue Feb 20 19:23 - 19:23  (00:00)
Student  ssh:notty    172.30.0.2       Tue Feb 20 19:23 - 19:23  (00:00)
student  ssh:notty    172.30.0.2       Tue Feb 20 19:03 - 19:03  (00:00)
student  ssh:notty    172.30.0.2       Tue Feb 20 19:03 - 19:03  (00:00)
student  ssh:notty    172.30.0.2       Tue Feb 20 19:03 - 19:03  (00:00)

btmp begins Tue Feb 20 19:03:31 2024
[Student@Switch01 ~]$
```

Note. Figure 7 should display the result of a simulated brute force attack on the Linux Switch01 system (Jones & Bartlett, 2024). The image typically feature the output from a ‘sudo lastb’ command, which retrieves the details of failed login attempts. These details would include user IDs, methods used to attempt logins, originating IP addresses, and the timestamps of each attempt. This visual is essential for understanding the system’s defenses against unauthorized attempts and for shaping strategies to mitigate such security risks.

Figure 8

Make a screen capture showing the last 10 log messages.



Implementing Security Monitoring and Logging (4e)
vWorkstation
2024-02-20 22:35:59
Osamudiamen Eweka

```
Feb 20 19:24:30 Switch01 unix_chkpwd[3816]: password check failed for user (Stud
ent)
Feb 20 19:24:31 Switch01 sshd[3793]: Failed password for Student from 172.30.0.2
 port 1070 ssh2
Feb 20 19:24:31 Switch01 sshd[3793]: error: maximum authentication attempts exce
eded for Student from 172.30.0.2 port 1070 ssh2 [preauth]
Feb 20 19:24:31 Switch01 sshd[3793]: PAM 5 more authentication failures; logname
= uid=0 euid=0 tty=ssh ruser= rhost=172.30.0.2  user=Student
Feb 20 19:28:33 Switch01 sshd[3849]: Accepted password for Student from 172.30.0
.2 port 1072 ssh2
Feb 20 19:28:33 Switch01 sshd[3849]: pam_unix(sshd:session): session opened for
user Student by (uid=0)
Feb 20 19:30:15 Switch01 sudo: Student : TTY=pts/0 ; PWD=/home/Student ; USER=ro
ot ; COMMAND=/bin/cat /var/log/secure
[Student@Switch01 ~]$ sudo lastb
Student  ssh:notty    172.30.0.2       Tue Feb 20 19:24 - 19:24  (00:00)
Student  ssh:notty    172.30.0.2       Tue Feb 20 19:24 - 19:24  (00:00)
Student  ssh:notty    172.30.0.2       Tue Feb 20 19:24 - 19:24  (00:00)
Student  ssh:notty    172.30.0.2       Tue Feb 20 19:24 - 19:24  (00:00)
Student  ssh:notty    172.30.0.2       Tue Feb 20 19:23 - 19:23  (00:00)
Student  ssh:notty    172.30.0.2       Tue Feb 20 19:23 - 19:23  (00:00)
student  ssh:notty    172.30.0.2       Tue Feb 20 19:03 - 19:03  (00:00)
student  ssh:notty    172.30.0.2       Tue Feb 20 19:03 - 19:03  (00:00)
student  ssh:notty    172.30.0.2       Tue Feb 20 19:03 - 19:03  (00:00)

btmp begins Tue Feb 20 19:03:31 2024
[Student@Switch01 ~]$ sudo tail /var/log/messages
Feb 20 19:29:55 Switch01 journal: D-Bus service launched with name: net.reactivated.Fprint
Feb 20 19:29:55 Switch01 fprintd: Launching FprintObject
Feb 20 19:29:55 Switch01 journal: entering main loop
Feb 20 19:30:01 Switch01 systemd: Created slice User Slice of root.
Feb 20 19:30:01 Switch01 systemd: Starting User Slice of root.
Feb 20 19:30:01 Switch01 systemd: Started Session 27 of user root.
Feb 20 19:30:01 Switch01 systemd: Starting Session 27 of user root.
Feb 20 19:30:01 Switch01 systemd: Removed slice User Slice of root.
Feb 20 19:30:01 Switch01 systemd: Stopping User Slice of root.
Feb 20 19:30:25 Switch01 journal: No devices in use, exit
[Student@Switch01 ~]$ 
```

Note. Figure 8 is intended to show a snapshot from the Linux Switch01 system, capturing the most recent 10 log messages (Jones & Bartlett, 2024). This would be achieved using the command ‘sudo tail /var/log/messages’. In the ideal scenario, this screen capture includes a variety of system logs, encompassing critical and non-critical messages, informational notes, and warnings. Such insights are instrumental for learners to comprehend the system's recent activities, aiding in the monitoring and diagnostic processes.

Section 2

Part 2: Monitor File Integrity with Tripwire

In this segment of the laboratory exercise, we will engage in configuring Tripwire, a tool dedicated to monitoring the integrity of files on a Linux operating system. The process initiates with the command ‘sudo /usr/sbin/tripwire-setup-keyfiles’, which is pivotal for importing the Tripwire configuration files into the system. During this phase, we will establish unique passphrases for both the site and local keys. These passphrases are fundamental to the encryption of the configuration and policy files, as well as for safeguarding the Tripwire database. The significance of these passphrases cannot be overstated, as they are instrumental in preserving the security framework of the system's monitoring configurations and in the identification of any modifications.

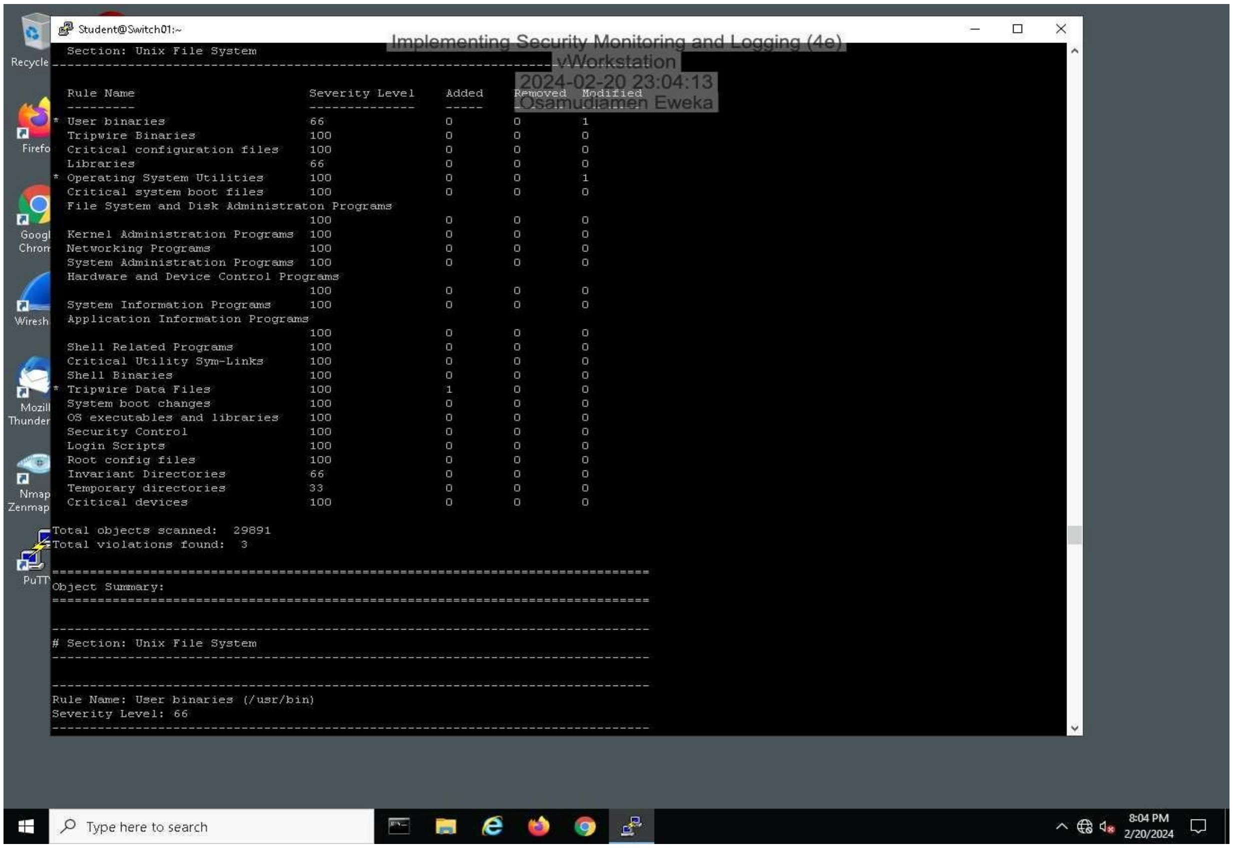
Post the establishment of passphrases, the next step involves the activation of Tripwire’s database through the command ‘sudo /usr/sbin/tripwire --init’. This crucial step indexes all the files designated by the policy file, employing the local passphrase to secure the database. The essence of this procedure lies in the analysis of the policy file and the subsequent creation of a database reflective of its directives. Following this, we will simulate an alteration in a file by executing ‘sudo touch /bin/ls’, which modifies the timestamp of the ‘/bin/ls’ file. This act serves as a practical demonstration of how changes are detected.

The final phase encompasses executing a file integrity assessment via ‘sudo /usr/sbin/tripwire --check’. This action entails a meticulous comparison of the indexed files against the baseline established by the policy file. The Integrity Check Report generated will detail any anomalies found, inclusive of the intentionally induced modification to the ‘/bin/ls’ file. This report offers an exhaustive insight into the state of file integrity within the system.

Engaging in this hands-on activity with Tripwire not only elucidates the operational mechanics of file integrity monitoring but also underscores its critical role in fortifying the security posture of a Linux system. Through this exercise, the participants are afforded a practical understanding of the mechanisms by which Tripwire operates, enhancing their comprehension of the significance of monitoring file integrity as a measure of cybersecurity.

Figure 9

Make a screen capture showing the Object Summary section for the Tripwire report.



```
  Section: Unix File System
-------------------------------------------------------------------------------

  Rule Name                       Severity Level    Added    Removed  Modified
  ---------                       --------------    -----    -------  --------
* User binaries                   66                0        0        1
  Tripwire Binaries               100               0        0        0
  Critical configuration files    100               0        0        0
  Libraries                       66                0        0        0
* Operating System Utilities      100               0        0        1
  Critical system boot files      100               0        0        0
  File System and Disk Administraton Programs
                                  100               0        0        0
  Kernel Administration Programs  100               0        0        0
  Networking Programs             100               0        0        0
  System Administration Programs  100               0        0        0
  Hardware and Device Control Programs
                                  100               0        0        0
  System Information Programs     100               0        0        0
  Application Information Programs
                                  100               0        0        0
  Shell Related Programs          100               0        0        0
  Critical Utility Sym-Links      100               0        0        0
  Shell Binaries                  100               0        0        0
* Tripwire Data Files             100               1        0        0
  System boot changes             100               0        0        0
  OS executables and libraries    100               0        0        0
  Security Control                100               0        0        0
  Login Scripts                   100               0        0        0
  Root config files               100               0        0        0
  Invariant Directories           66                0        0        0
  Temporary directories           33                0        0        0
  Critical devices                100               0        0        0

Total objects scanned:  29891
Total violations found:  3

===============================================================================
Object Summary:
===============================================================================

-------------------------------------------------------------------------------
# Section: Unix File System
-------------------------------------------------------------------------------

-------------------------------------------------------------------------------
Rule Name: User binaries (/usr/bin)
Severity Level: 66
-------------------------------------------------------------------------------
```

Note. The screenshot provided delineates the Object Summary Section from a Tripwire report, offering a streamlined visual summary of the results from the file integrity assessment detailed previously (Jones & Bartlett, 2024). This section meticulously categorizes each monitored object by its name, type, and the comparison between its expected and observed states. This organization is invaluable for system administrators, enabling immediate recognition of discrepancies or unauthorized changes within the system's files and directories. The layout is designed for swift analysis, assisting in the rapid identification and rectification of potential security concerns, thereby reinforcing the integrity and security of the Linux environment.

Section 3: Challenge and Analysis

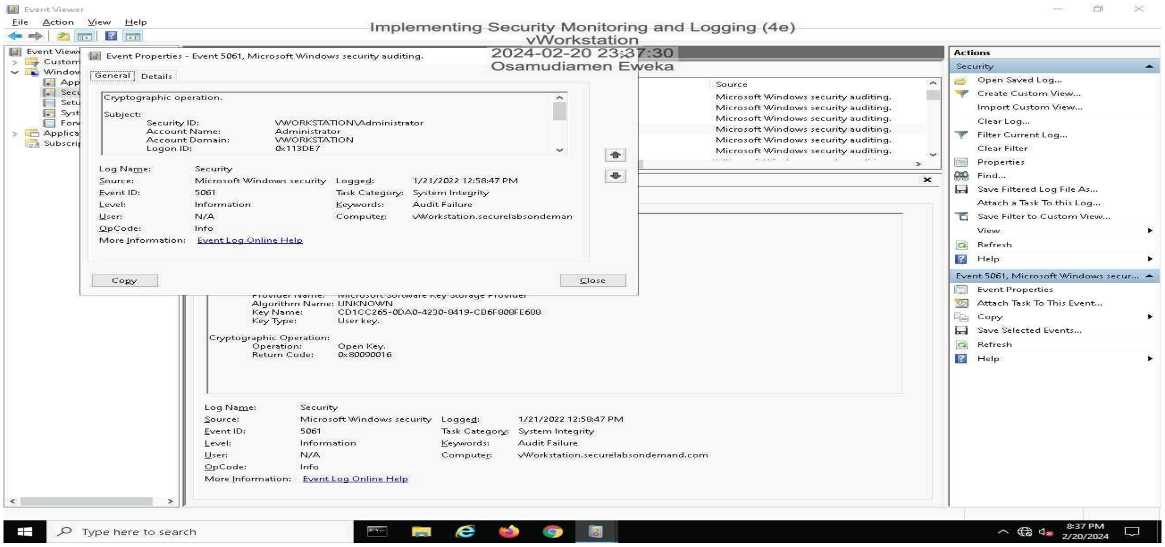
Part 1: Identify Additional Event Types in the Event Viewer

In this portion of the laboratory exercise, following a directive from management at Secure Labs on Demand, the participant initiates a session on the virtual Workstation to delve into Audit Failure incidents tagged with Event ID 5061 in the Windows Event Viewer. Upon navigating to the Security log, the participant singles out and documents the Security Event Properties dialog box for a particular Audit Failure event bearing Event ID 5061. The image captured below is earmarked for subsequent examination.

This event, Event ID 5061, is commonly indicative of a failure within the security audit mechanisms of Microsoft Windows. Such a failure usually points to difficulties in the negotiation or formation of a security association, hinting at potential discrepancies in network security protocols or authentication mechanisms. An in-depth grasp of these incidents is essential for the ongoing security management and improvement of the organization's security stance.

Figure 10

Make a screen capture showing the Security Event Properties dialog box for an Audit Failure associated with Event ID 5061.



```
Cryptographic operation.

Subject:
        Security ID:            VWORKSTATION\Administrator
        Account Name:           Administrator
        Account Domain:         VWORKSTATION
        Logon ID:               0x113DE7

Log Name:          Security
Source:            Microsoft Windows security   Logged:          1/21/2022 12:58:47 PM
Event ID:          5061                         Task Category:   System Integrity
Level:             Information                  Keywords:        Audit Failure
User:              N/A                          Computer:        vWorkstation.securelabsondeman
OpCode:            Info
More Information:  Event Log Online Help
```

```
        Provider Name:      Microsoft Software Key Storage Provider
        Algorithm Name:     UNKNOWN
        Key Name:           CD1CC265-0DA0-4230-8419-CB6F808FE688
        Key Type:           User key.

Cryptographic Operation:
        Operation:          Open Key.
        Return Code:        0x80090016

Log Name:          Security
Source:            Microsoft Windows security   Logged:          1/21/2022 12:58:47 PM
Event ID:          5061                         Task Category:   System Integrity
Level:             Information                  Keywords:        Audit Failure
User:              N/A                          Computer:        vWorkstation.securelabsondemand.com
OpCode:            Info
More Information:  Event Log Online Help
```

Note. Figure 10 above encapsulates a screenshot of the Security Event properties dialog box from the vWorkstation, elucidating the specifics of an Audit Failure event linked to Event ID 5061 (Jones & Bartlett, 2024). The visual includes pivotal details like the event's timestamp, the implicated user account, and particularities of the event itself. This imagery is a product of the security analysis conducted at Secure Labs on Demand, in response to a managerial mandate to investigate Audit Failure events. The graphical depiction of the Security Event Properties furnishes a detailed overview of the incident for further scrutiny and forms a vital component in the comprehensive understanding and mitigation of security vulnerabilities within the Windows framework.

Question: Provide a brief explanation of the operation that would generate a security event with Event ID 5061.

Answer*:* A security event with Event ID 5061 is generated when a cryptographic operation occurs, such as using, creating, or opening cryptographic keys with a Key Storage Provider. This event is part of Windows Security Auditing, designed to track the use and management of cryptographic keys to ensure security and compliance within an IT environment. For specific details, please refer to the Microsoft documentation (Vinaypamnani-Msft, 2022).

Section 3: Challenge and Analysis

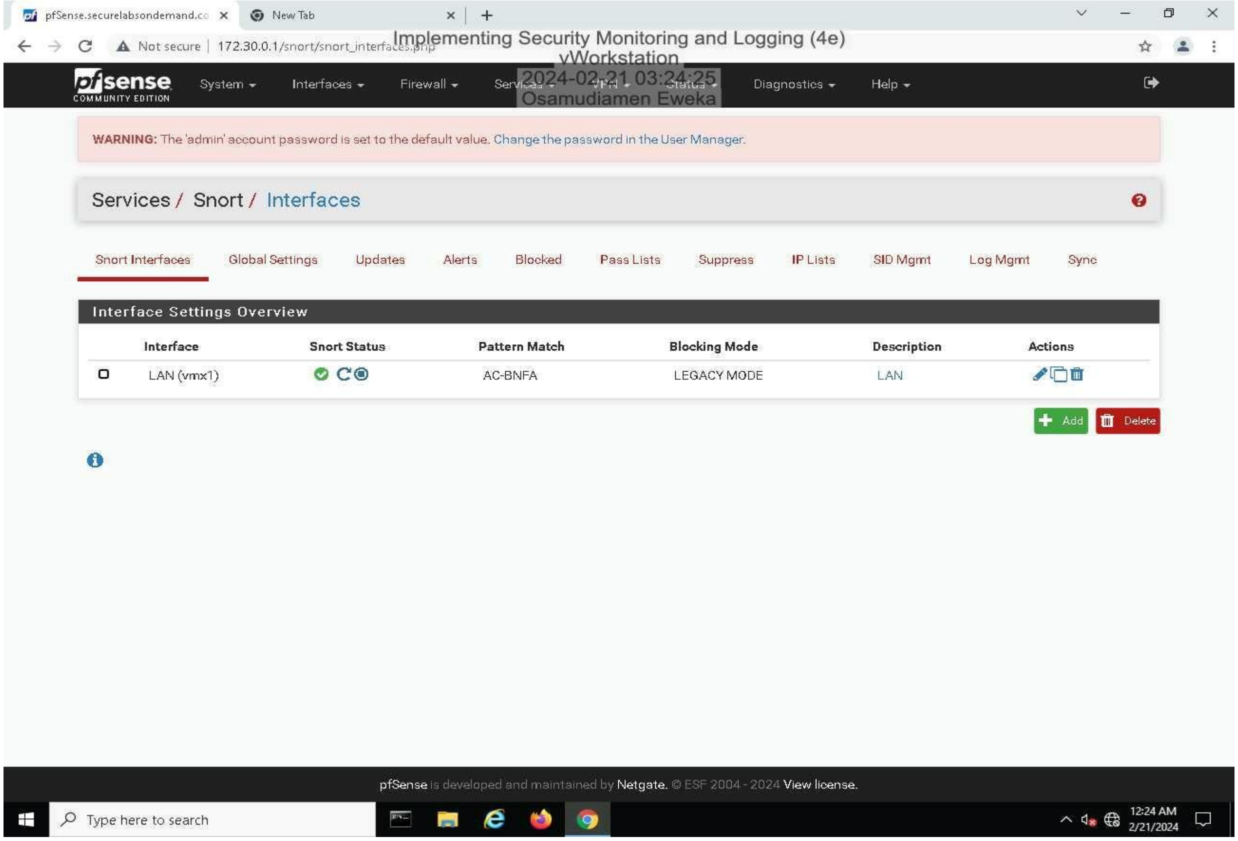
Part 2: Configure Snort as an Intrusion Prevention System

Within this laboratory exercise, the security analyst at Secure Labs on Demand undertakes a pivotal assignment following directives from their manager, post a meeting with the Chief Technology Officer (CTO). The task at hand involves upgrading the functionality of the company's Snort Intrusion Detection System (IDS) to operate as an Intrusion Prevention System (IPS). To achieve this, the analyst is to access the pfSense WebGUI from the virtual Workstation and proceed to the Snort service page. Drawing upon the Snort Package documentation for pfSense, the analyst's objective is to activate Snort's IPS feature, specifically referred to as "Block Offenders," on the Local Area Network (LAN) interface delineated in the initial section of this exercise.

Guided by the instructions, the analyst's mission includes capturing a screenshot demonstrating the activation of the Legacy Blocking Mode on the LAN interface. This adjustment is critical for enabling Snort to not just detect but actively prevent and block potential security breaches, marking a significant evolution from its traditional role as an IDS to that of an IPS. This transition is paramount in bolstering the company's defensive mechanisms and is in direct response to the CTO's strategic vision for enhancing the organization's capabilities in thwarting cyber threats.

Figure 11

Make a screen capture showing the Legacy Blocking Mode enabled on the LAN interface.



WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Services / Snort / Interfaces

Interface Settings Overview

	Interface	Snort Status	Pattern Match	Blocking Mode	Description	Actions
□	LAN (vmx1)	✓	AC-BNFA	LEGACY MODE	LAN	

pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

Note. Figure 11 embodies a screenshot captured from the pfSense WebGUI, illustrating the analyst's proficient configuration of Snort to function as an IPS (Jones & Bartlett, 2024). The image distinctly highlights the Legacy Blocking Mode as "Enabled" on the LAN interface, as outlined in the initial part of this task. This visual documentation succinctly evidences the analyst's adept execution of Snort's IPS features, thereby contributing to the realization of the company's security goals and the CTO's directive to strengthen the network's resilience against potential intrusions.

Conclusion

This lab has provided an in-depth exploration of key cybersecurity concepts, focusing on the analysis of logs, intrusion detection, and enhancing network security. Participants engaged in practical exercises, including identifying failed login attempts in Linux, setting up remote logging, analyzing brute force attacks, and utilizing Tripwire for file integrity monitoring. These activities underscored the importance of detecting unauthorized changes and understanding Windows security through Audit Failure events, particularly Event ID 5061. Additionally, the lab illustrated the transformation of Snort from an Intrusion Detection System (IDS) to an Intrusion Prevention System (IPS), highlighting the adaptation of security tools for improved defense mechanisms. Through these varied tasks, participants gained valuable skills applicable to real-world cybersecurity scenarios, promoting a comprehensive approach to system and network protection.

Reference

ControlCase. (2023, June 6). *Security event logging and monitoring services*. https://www.controlcase.com/services/log-monitoring/

Jones & Bartlett (2024). Implementing Security Monitoring and Logging (Figures). *Jones and BartlettLearning Virtual Lab*. URL: https://jbl-lti.hatsize.com/startlab

Vinaypamnani-Msft. (2021, September 8). *5061(S, F) Cryptographic operation. - Windows 10*. Microsoft Learn. https://learn.microsoft.com/en-us/windows/security/threat-protection/auditing/event-5061