

Splunk Log Collection and Security Event Detection in a Virtual Lab

Cybersecurity Research Report

**An-Najah National University
Faculty of Engineering and Information Technology
Department of Network and Information Security**



**Prepared by: Yazan Azmi Balawneh
Submitted to: Dyaa Tumezeh
Submission Date: Aug 13, 2025**

Table of Contents

1. Introduction
2. Lab Environment Setup
3. Splunk Configuration
4. Attack Scenarios and Testing
5. Log Collection and Analysis
6. Detection Results
7. Summary

1. Introduction

Splunk is a powerful Security Information and Event Management (SIEM) platform designed to collect, index, and analyze machine-generated data from various sources in real-time. It enables security analysts to monitor events, detect anomalies, and investigate potential threats efficiently. One of Splunk's key components is the Splunk Universal Forwarder, a lightweight agent that runs on source machines to securely collect and forward logs to the main Splunk Enterprise server for indexing and analysis.

In addition to Splunk, this project also integrates Suricata, an open-source intrusion detection and prevention system (IDS/IPS) capable of monitoring network traffic and detecting malicious activities using predefined rules. Suricata was deployed to enhance network-level visibility and provide detailed alerts for suspicious packets and potential intrusions.

To validate the monitoring and detection capabilities of Splunk, multiple attack scenarios were simulated in a controlled lab environment. These included port scanning using Nmap, SSH brute-force attempts, web application attacks such as Cross-Site Scripting (XSS) and SQL Injection (SQLi), and a Distributed Denial-of-Service (DDoS) simulation using UDP flood via LOIC. Each attack was executed from a Parrot Security OS virtual machine targeting an Ubuntu 24 machine running Suricata and Splunk Universal Forwarder.

The purpose of this report is to demonstrate the process of collecting, forwarding, and analyzing security logs using Splunk, supported by Suricata's network detection capabilities.

The main objectives are:

- To showcase Splunk's role in centralized log management and real-time threat detection.
- To integrate Suricata for enhanced network monitoring.
- To simulate different types of cyberattacks and observe their detection in Splunk.
- To analyze and interpret security events for incident response.

The problem addressed in this project is the lack of centralized visibility and correlation between security events across different systems. By implementing Splunk with Universal Forwarder and Suricata, security logs from multiple sources were aggregated, allowing for comprehensive monitoring, faster detection, and improved incident analysis.



2. Lab Environment Setup

The lab environment was designed to simulate a small-scale network infrastructure where security monitoring and attack detection could be tested in a controlled setting. The setup consisted of three main systems:

2.1 Host Machine

- Operating System: Windows 10 Pro
- Role: Hosting the virtual lab environment using VirtualBox and running Splunk Enterprise locally.
- Purpose: Serve as the central SIEM server for collecting, storing, and analyzing logs from other machines.

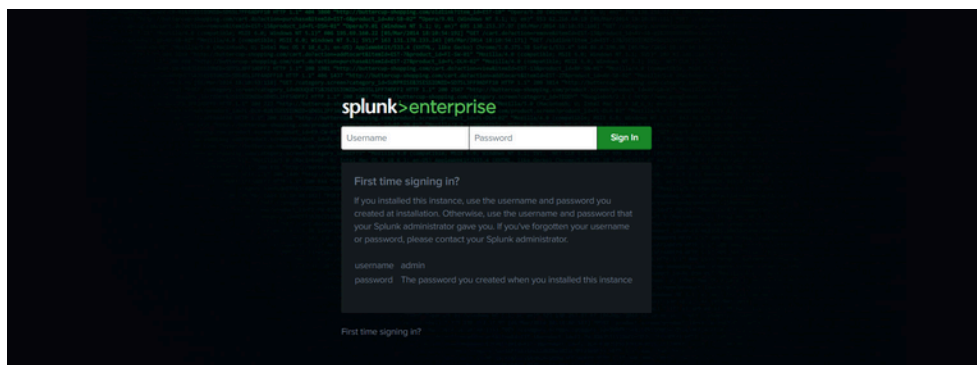


Figure 2.1: Splunk Enterprise Web Interface on Host Machine

2.2 Ubuntu 24 Virtual Machine

- Role: Monitored endpoint running Splunk Universal Forwarder and Suricata IDS.
- Purpose: Generate security logs (system logs, authentication logs, kernel logs) and forward them to the Splunk Enterprise server via port 9997. Suricata monitored network traffic for suspicious activity and forwarded alerts to Splunk.

```
yazan@yazan-VirtualBox:~/Downloads$ sudo dpkg -i splunkforwarder-*.deb
[sudo] password for yazan:
Selecting previously unselected package splunkforwarder.
(Reading database ... 149774 files and directories currently installed.)
Preparing to unpack splunkforwarder-10.0.0-e8eb0c4654f8-linux-amd64.deb ...
verify that this system has all the commands we will require to perform the preflight step
no need to run the splunk-preinstall upgrade check
Unpacking splunkforwarder (10.0.0) ...
Setting up splunkforwarder (10.0.0) ...
find: '/opt/splunkforwarder/lib/python3.7/site-packages': No such file or directory
find: '/opt/splunkforwarder/lib/python3.9/site-packages': No such file or directory
complete
yazan@yazan-VirtualBox:~/Downloads$ sudo /opt/splunkforwarder/bin/splunk start -
-accept-license
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
This appears to be your first time running this version of Splunk.
```

Figure 2.2: Splunk Universal Forwarder Running on Ubuntu 24

2.3 Parrot Security OS Virtual Machine

- Role: Attacker machine used for penetration testing and simulated cyberattacks.
- Purpose: Execute various security tests including Nmap scans, SSH brute force, web application attacks (XSS, SQLi), and DDoS simulations.

```
[yazan@parrot]~  
$sudo nmap -Pn -sS -p 1-1000 -T4 --reason 10.1.1.105  
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-08-11 12:55 EDT  
Nmap scan report for 10.1.1.105  
Host is up, received arp-response (0.00025s latency).  
All 1000 scanned ports on 10.1.1.105 are in ignored states.  
Not shown: 1000 filtered tcp ports (no-response)  
MAC Address: 08:00:27:DF:53:28 (Oracle VirtualBox virtual NIC)
```

Figure 2.3: Nmap Scan from Parrot Security OS

2.4 Network Topology

The virtual machines and the host machine were connected through VirtualBox's Host-Only Network and NAT settings, enabling both internet connectivity and internal network communication. This allowed the attacker VM to target the Ubuntu VM, and the Ubuntu VM to forward logs to the Splunk Enterprise server on the host machine.

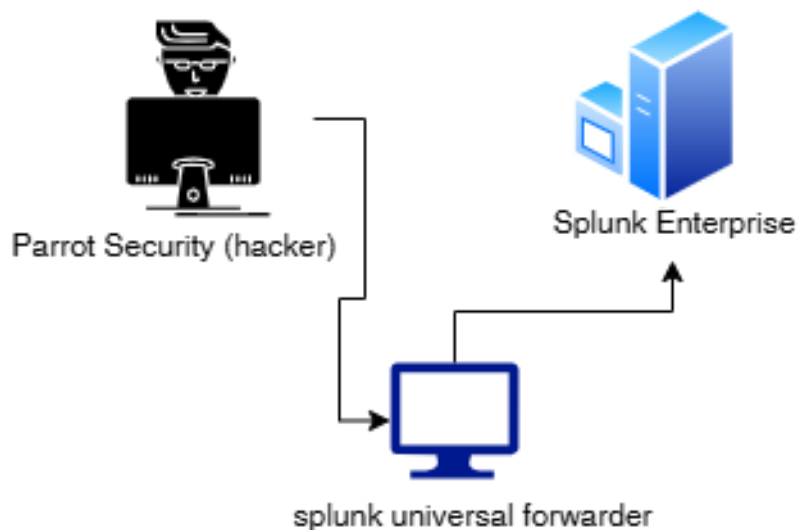


Figure 2.4: Lab Network Topology Diagram

3. Splunk Configuration

In this project, Splunk Enterprise was installed and configured on the Windows 10 Pro host machine to serve as the main SIEM platform. The configuration process ensured that logs from multiple sources—including the local host and the Ubuntu VM—were collected, indexed, and ready for analysis.

3.1 Splunk Enterprise Installation

Splunk Enterprise was downloaded and installed on the host machine. After installation, the Splunk Web interface was accessed through a web browser, providing a central dashboard for managing inputs, indexes, and searches.

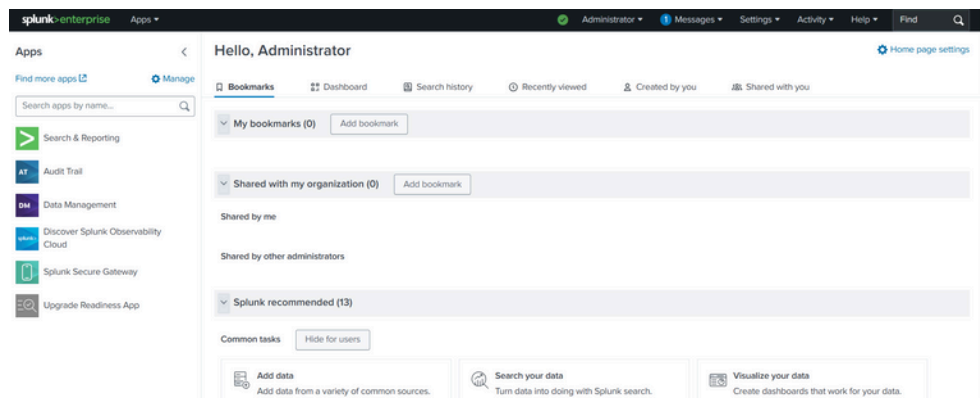


Figure 3.1: Splunk Enterprise Interface

3.2 Local Log Collection

The host machine was configured to forward its own Application, Security, and System event logs to Splunk. This allowed monitoring of security-relevant activities occurring on the Windows 10 Pro host.

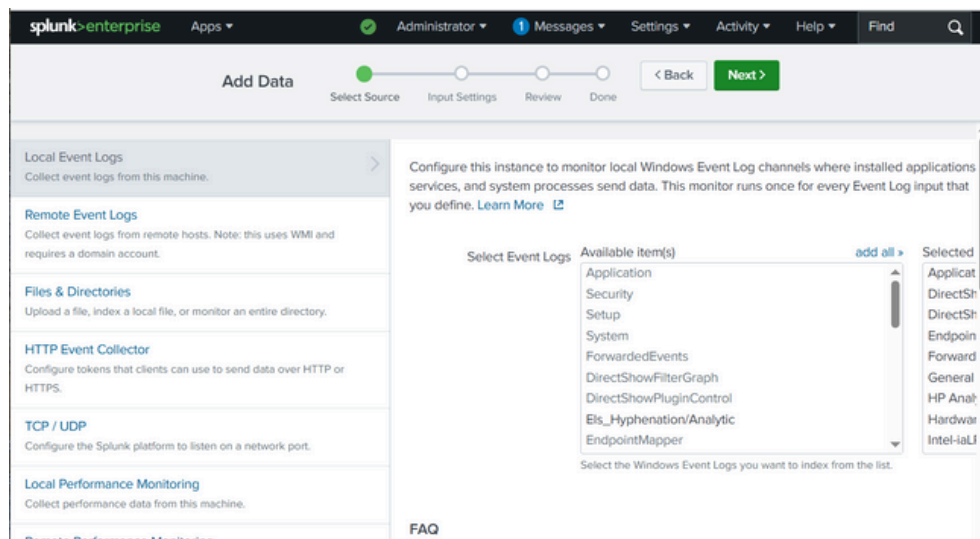
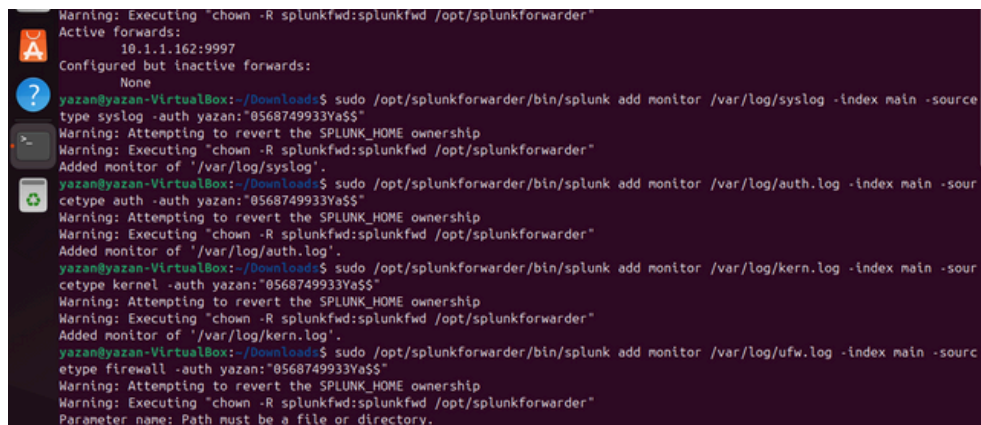


Figure 3.2: Indexed Windows Event Logs in Splunk

3.3 Configuring Splunk Universal Forwarder on Ubuntu

The Splunk Universal Forwarder was installed on the Ubuntu VM to send specific logs to the Splunk Enterprise server on port 9997. The forwarded logs included:

- Authentication logs (/var/log/auth.log)
- Kernel logs (/var/log/kern.log)
- Syslog entries (/var/log/syslog)



```
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Active forwards:
  10.1.1.162:9997
Configured but inactive forwards:
  None
yazan@yazan-VirtualBox:~/Downloads$ sudo /opt/splunkforwarder/bin/splunk add monitor /var/log/syslog -index main -source
type syslog -auth yazan:"0568749933Ya$$"
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Added monitor of '/var/log/syslog'.
yazan@yazan-VirtualBox:~/Downloads$ sudo /opt/splunkforwarder/bin/splunk add monitor /var/log/auth.log -index main -sour
cetype auth -auth yazan:"0568749933Ya$$"
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Added monitor of '/var/log/auth.log'.
yazan@yazan-VirtualBox:~/Downloads$ sudo /opt/splunkforwarder/bin/splunk add monitor /var/log/kern.log -index main -sour
cetype kernel -auth yazan:"0568749933Ya$$"
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Added monitor of '/var/log/kern.log'.
yazan@yazan-VirtualBox:~/Downloads$ sudo /opt/splunkforwarder/bin/splunk add monitor /var/log/ufw.log -index main -sourc
etype firewall -auth yazan:"0568749933Ya$$"
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Parameter name: Path must be a file or directory.
```

Figure 3.3: Universal Forwarder Sending Logs to Splunk

3.4 Receiving Data on Splunk

On the Splunk Enterprise server, a TCP input was created on port 9997 to receive logs from the Ubuntu Universal Forwarder. Once received, the logs were stored in their designated indexes for search and analysis.

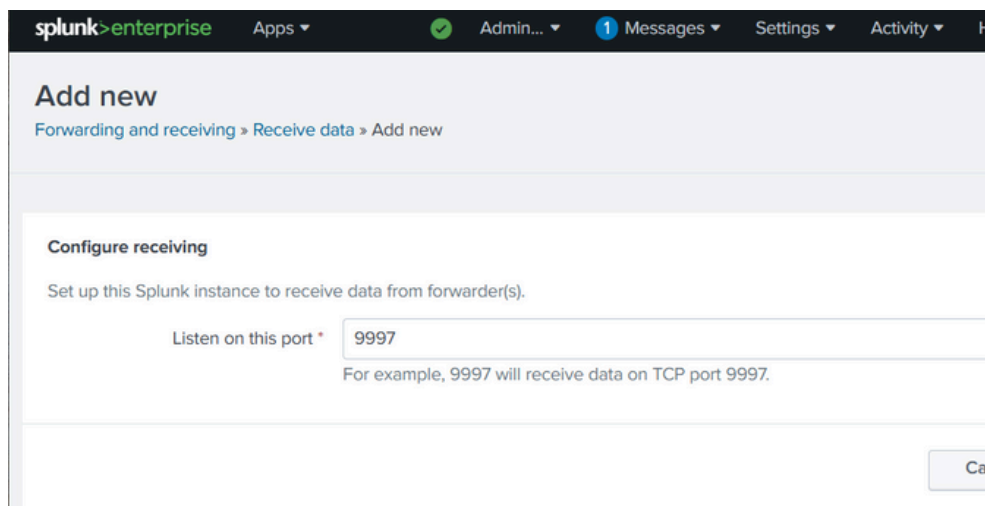


Figure 3.4: TCP Data Input Configuration in Splunk

4.3 Web Application Attacks

Although the Ubuntu VM was not configured as a full web server, test payloads for common web application vulnerabilities were executed against specific URLs to generate Suricata alerts.

- Cross-Site Scripting (XSS) Test – Injected a basic JavaScript payload to trigger detection.
- SQL Injection (SQLi) Test – Sent crafted SQL payloads to simulate a database attack attempt.

```

yazan@yazan-VirtualBox:~$ # SQLi (URL-encoded)
for i in {1..8}; do
  curl -s "http://testphp.vulnweb.com/listproducts.php?cat=%27%20OR%20%271%27%3D%271" -o /dev/null
done

# XSS
for i in {1..8}; do
  curl -s "http://neverssl.com/?q=%3Cscript%3Ealert(1)%3C/script%3E" -o /dev/null
done
^C
yazan@yazan-VirtualBox:~$ sudo tail -f /var/log/syslog | grep -E 'TEST SQLi|TEST XSS|event type=="alert'

```

Figure 4.3: XSS and SQL Injection Test Execution

4.4 DDoS Simulation (UDP Flood via LOIC)

A Distributed Denial-of-Service (DDoS) simulation was conducted using the Low Orbit Ion Cannon (LOIC) tool to flood the target Ubuntu VM with UDP packets. Suricata detected a high number of incoming packets, and logs were sent to Splunk for analysis.

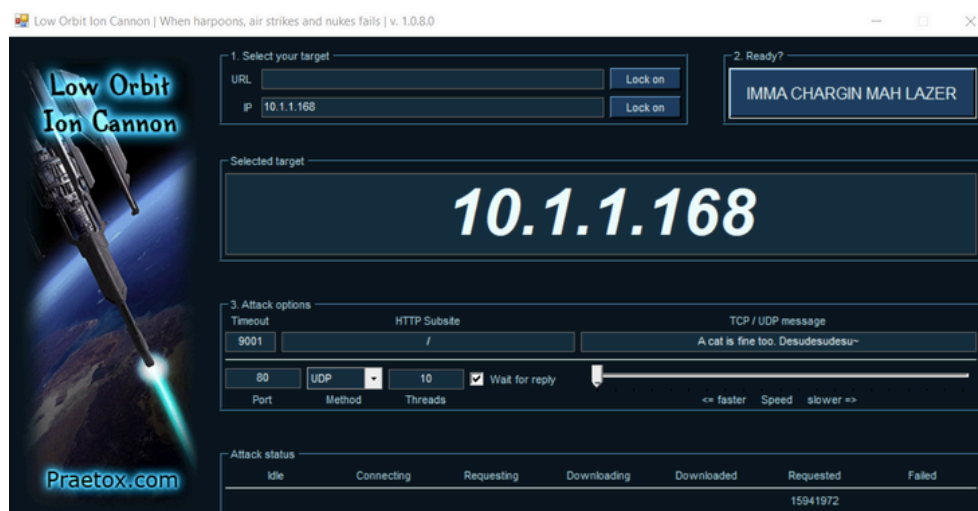


Figure 4.4: UDP Flood DDoS Simulation with LOIC

5. Log Collection and Analysis

Logs from both the host machine (Windows 10 Pro) and the Ubuntu 24 VM were successfully collected and centralized in Splunk Enterprise. This section describes how the logs were aggregated, indexed, and analyzed for each attack scenario.

5.1 Local Event Logs from Windows

The host machine's Application, Security, and System event logs were indexed directly into Splunk. This allowed for monitoring of local administrative actions, system events, and security incidents on the Splunk server itself.

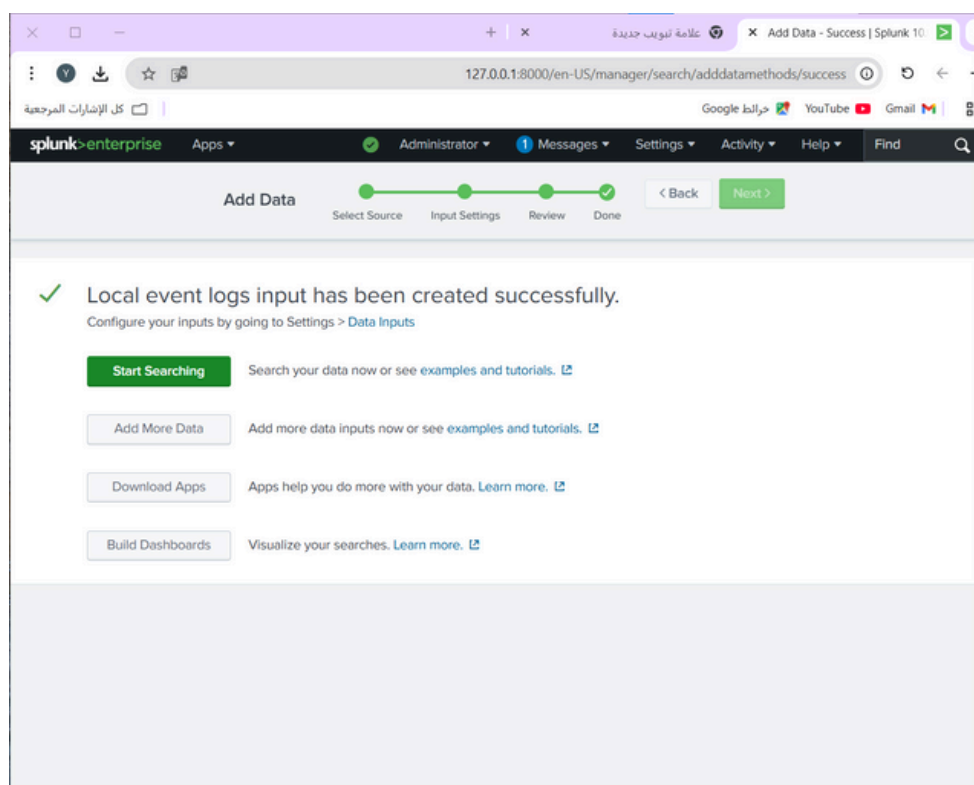


Figure 5.1: Windows Security Event Logs in Splunk

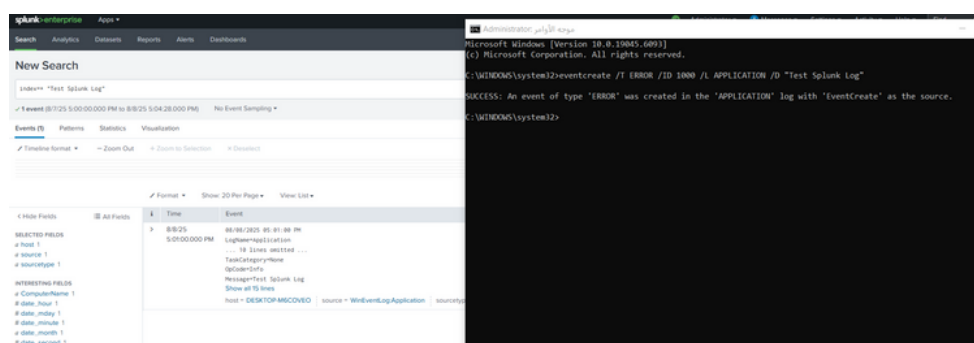
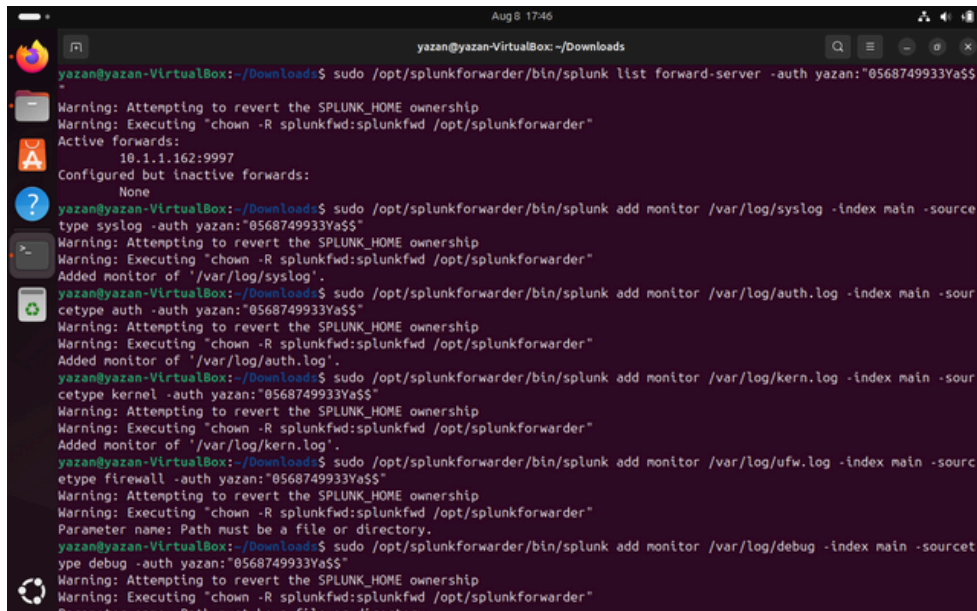


Figure 5.1: Windows Security Event Logs in Splunk

5.2 Remote Logs from Ubuntu via Splunk Universal Forwarder

The Splunk Universal Forwarder on Ubuntu sent key log files—auth.log, kern.log, and syslog—to Splunk Enterprise on port 9997. These logs included authentication attempts, kernel messages, and general system logs, enabling detailed visibility into potential attacks.

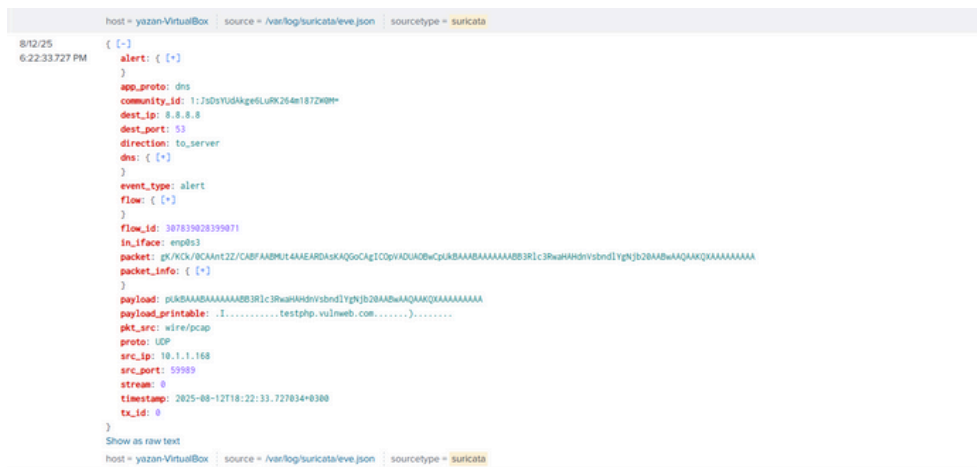


```
Aug 8 17:46
yazan@yazan-VirtualBox: ~/Downloads
yazan@yazan-VirtualBox:~/Downloads$ sudo /opt/splunkforwarder/bin/splunk list forward-server -auth yazan:"0568749933Ya$$"
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Active forwards:
10.1.1.162:9997
Configured but inactive forwards:
None
yazan@yazan-VirtualBox:~/Downloads$ sudo /opt/splunkforwarder/bin/splunk add monitor /var/log/syslog -index main -sourcetype syslog -auth yazan:"0568749933Ya$$"
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Added monitor of '/var/log/syslog'.
yazan@yazan-VirtualBox:~/Downloads$ sudo /opt/splunkforwarder/bin/splunk add monitor /var/log/auth.log -index main -sourcetype auth -auth yazan:"0568749933Ya$$"
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Added monitor of '/var/log/auth.log'.
yazan@yazan-VirtualBox:~/Downloads$ sudo /opt/splunkforwarder/bin/splunk add monitor /var/log/kern.log -index main -sourcetype kernel -auth yazan:"0568749933Ya$$"
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Added monitor of '/var/log/kern.log'.
yazan@yazan-VirtualBox:~/Downloads$ sudo /opt/splunkforwarder/bin/splunk add monitor /var/log/ufw.log -index main -sourcetype firewall -auth yazan:"0568749933Ya$$"
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Parameter name: Path must be a file or directory.
yazan@yazan-VirtualBox:~/Downloads$ sudo /opt/splunkforwarder/bin/splunk add monitor /var/log/debug -index main -sourcetype debug -auth yazan:"0568749933Ya$$"
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Parameter name: Path must be a file or directory
```

Figure 5.2: Ubuntu Authentication Logs in Splunk

5.3 Suricata Alerts

Suricata generated IDS alerts whenever suspicious network activity was detected, such as port scanning, SQL injection, or DDoS attempts. These alerts were forwarded to Splunk, allowing correlation between system logs and network-level detection.



```
host = yazan-VirtualBox | source = /var/log/suricata/eve.json | sourcetype = suricata
8/12/25 6:22:33.727 PM
{
  alert: {
    app_proto: dns
    community_id: 1:7d5YUdAqetLUK264e187ZWMH
    dest_ip: 8.8.8.8
    dest_port: 53
    direction: to_server
    dns: {
    }
    event_type: alert
    flow: {
    }
    flow_id: 387839028398071
    in_iface: enp0s3
    packet: pK/KC/XCAntZZ/CABF AABM/Ut4AAEARDAuKAGoCagICOpvXDUADwCpKBAABAAAAAABBB3Rlc3RwAHdndvbnndIYnJhZDABwAAQAAQAAAAAAAA
    packet_info: {
    }
    payload: pKBAABAAAAAABBB3Rlc3RwAHdndvbnndIYnJhZDABwAAQAAQAAAAAAAA
    payload_printable: .I.....testphp.vulnweb.com.....
    pkt_src: wire/pcap
    proto: UDP
    src_ip: 10.1.1.168
    src_port: 55989
    stream: 8
    timestamp: 2025-08-12T18:22:33.727834+0300
    tx_id: 0
  }
}
Show as raw text
host = yazan-VirtualBox | source = /var/log/suricata/eve.json | sourcetype = suricata
```

Figure 5.3: Suricata logs in Splunk

6. Detection Results

The detection phase focused on verifying whether the simulated attacks were successfully identified and recorded in Splunk. By correlating logs from the Ubuntu VM, Suricata alerts, and the Windows host event logs, each attack scenario was confirmed as detected.

6.1 Port Scanning Detection

Splunk displayed multiple connection attempts to various ports in a short time frame.

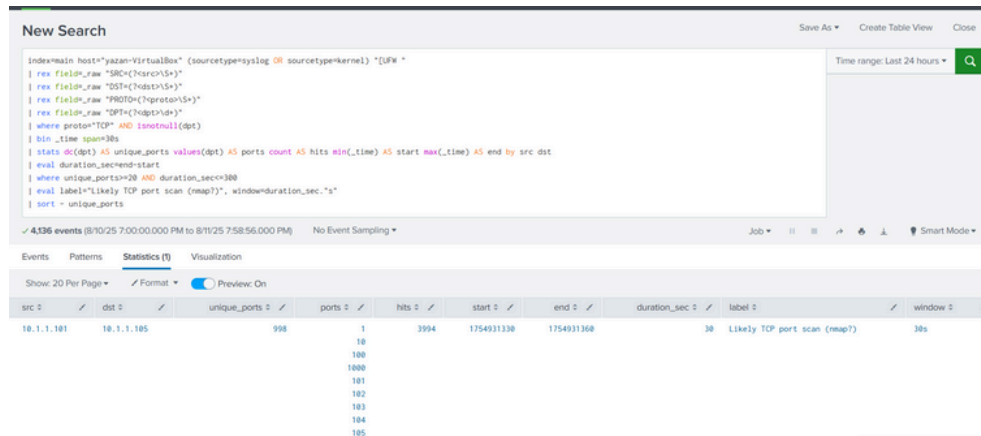


Figure 6.1: Detected Nmap Port Scanning Activity in Splunk

6.2 SSH Brute Force Detection

The authentication logs revealed numerous failed SSH login attempts from the attacker IP address. Splunk search queries highlighted the excessive failed logins, confirming a brute force attack.

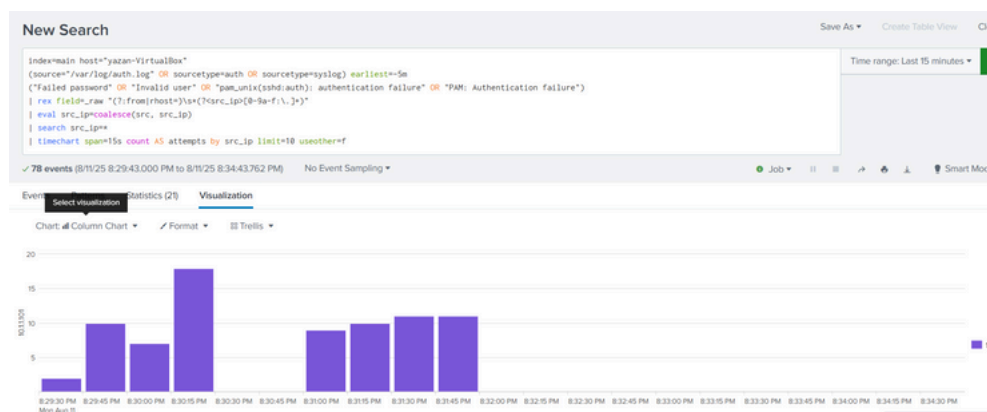


Figure 6.2: SSH Brute Force Detection in Splunk

6.3 Web Application Attack Detection

Suricata rules triggered alerts for suspicious HTTP requests matching XSS and SQLi attack patterns. These alerts were visible in Splunk's event listings, showing request details and source IP addresses.

New Search

Index=main (sourcetype=suricata OR sourcetype=suricata-evt OR source="/var/log/suricata-evt.log") earliest=1m

! alert

search event_type=alert.alert.signature OR ("TEST SQLi (decoded)","TEST SQLi (encoded)")

table _time src_ip dest_ip http.hostname http.url alert.signature

sort = _time

✓ 6 events 6/12/25 6:48:31:000 PM to 6/12/25 6:58:31:548 PM

No Event Sampling

Job

Smart Mode

Events Patterns Statistics (6) Visualization

Show: 20 Per Page

Format

Preview On

_time	src_ip	dest_ip	http.hostname	http.url	alert.signature
2025-06-12 18:58:22.329	10.1.1.168 10.1.1.168	34.223.124.45 34.223.124.45	newrs1.com newrs1.com	/19-6-12/763095292/132/132/132/1 /19-6-12/763095292/132/132/132/1	TEST SQLi (decoded) TEST SQLi (decoded)
2025-06-12 18:58:22.329	10.1.1.168 10.1.1.168	34.223.124.45 34.223.124.45	newrs1.com newrs1.com	/19-6-12/763095292/132/132/132/1 /19-6-12/763095292/132/132/132/1	TEST SQLi (decoded) TEST SQLi (decoded)
2025-06-12 18:58:16.688	10.1.1.168 10.1.1.168	34.223.124.45 34.223.124.45	newrs1.com newrs1.com	/19-6-12/763095292/132/132/132/1 /19-6-12/763095292/132/132/132/1	TEST SQLi (decoded) TEST SQLi (decoded)
2025-06-12 18:58:16.688	10.1.1.168 10.1.1.168	34.223.124.45 34.223.124.45	newrs1.com newrs1.com	/19-6-12/763095292/132/132/132/1 /19-6-12/763095292/132/132/132/1	TEST SQLi (decoded) TEST SQLi (decoded)
2025-06-12 18:57:56.451	10.1.1.168 10.1.1.168	34.223.124.45 34.223.124.45	newrs1.com newrs1.com	/19-6-12/763095292/132/132/132/1 /19-6-12/763095292/132/132/132/1	TEST SQLi (decoded) TEST SQLi (decoded)
2025-06-12 18:57:56.451	10.1.1.168 10.1.1.168	34.223.124.45 34.223.124.45	newrs1.com newrs1.com	/19-6-12/763095292/132/132/132/1 /19-6-12/763095292/132/132/132/1	TEST SQLi (decoded) TEST SQLi (decoded)

Figure 6.3: SQL Injection Detection in Splunk

New Search

Save As

Create Table View

Undermin (sourcetype=uriscata OR sourcetype=uriscata:evn OR source=*r/rig/uriscata:evn.json) earliest=1m

1 search

search event_type=alert alert.signature IN ("TEST XSS (decoded)", "TEST XSS (encoded)")

table _time src_ip dest_ip http.hostname http.uri alert.signature

sort - _time

Time range: Date time range >

14 events 8/12/25 6:40:57000 PM to 8/12/25 6:50:57005 PM

No Event Sampling

Events

Patterns

Statistics (14)

Visualization

Show: 20 Per Page

Format

Preview On

_time	src_ip	dest_ip	http.hostname	http.uri	alert.signature
2025-08-12 18:58:35.154	10.1.1.168	34.223.124.45	neverurl.com	/fwk0ccoriptkMalert(1)XGcoriptkth	TEST XSS (decoded)
	10.1.1.168	34.223.124.45	neverurl.com	/fwk0ccoriptkMalert(1)XGcoriptkth	TEST XSS (encoded)
2025-08-12 18:58:35.154	10.1.1.168	34.223.124.45	neverurl.com	/fwk0ccoriptkMalert(1)XGcoriptkth	TEST XSS (decoded)
	10.1.1.168	34.223.124.45	neverurl.com	/fwk0ccoriptkMalert(1)XGcoriptkth	TEST XSS (encoded)
2025-08-12 18:43:10.868	10.1.1.168	34.223.124.45	neverurl.com	/fwk0ccoriptkMalert(1)XGcoriptkth	TEST XSS (decoded)
	10.1.1.168	34.223.124.45	neverurl.com	/fwk0ccoriptkMalert(1)XGcoriptkth	TEST XSS (encoded)
2025-08-12 18:43:10.868	10.1.1.168	34.223.124.45	neverurl.com	/fwk0ccoriptkMalert(1)XGcoriptkth	TEST XSS (decoded)
	10.1.1.168	34.223.124.45	neverurl.com	/fwk0ccoriptkMalert(1)XGcoriptkth	TEST XSS (encoded)
2025-08-12 18:43:37.961	10.1.1.168	34.223.124.45	neverurl.com	/fwk0ccoriptkMalert(1)XGcoriptkth	TEST XSS (decoded)
	10.1.1.168	34.223.124.45	neverurl.com	/fwk0ccoriptkMalert(1)XGcoriptkth	TEST XSS (encoded)
2025-08-12 18:43:37.961	10.1.1.168	34.223.124.45	neverurl.com	/fwk0ccoriptkMalert(1)XGcoriptkth	TEST XSS (decoded)
	10.1.1.168	34.223.124.45	neverurl.com	/fwk0ccoriptkMalert(1)XGcoriptkth	TEST XSS (encoded)

Figure 6.3: XSS Detection in Splunk

6.4 DDoS (UDP Flood) Detection

Suricata detected an abnormally high rate of UDP packets targeting the Ubuntu machine. Splunk recorded a spike in network traffic events, confirming the DDoS simulation.

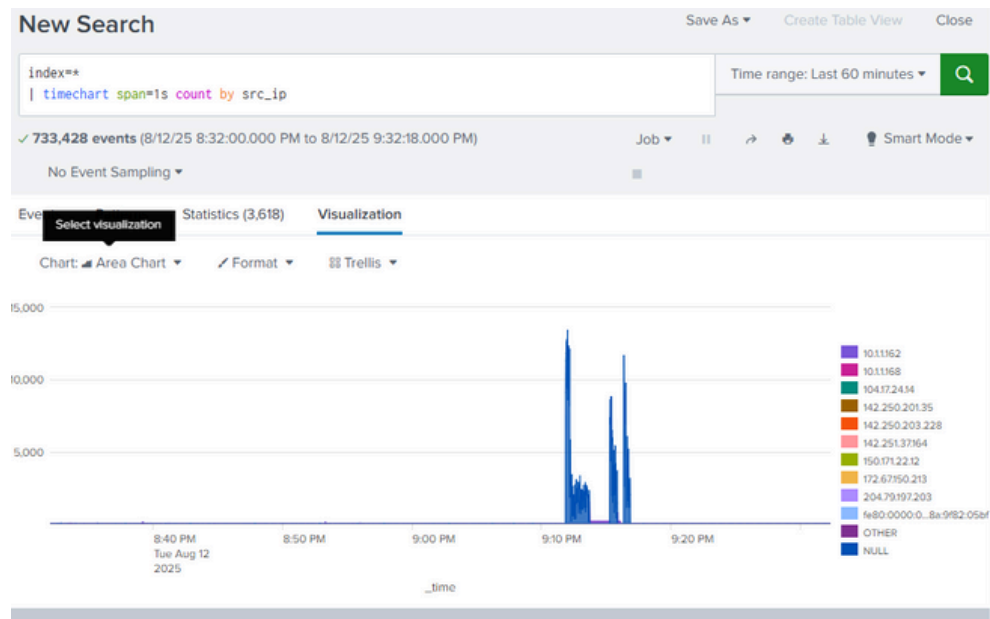


Figure 6.4: DDoS UDP Flood Detection in Splunk

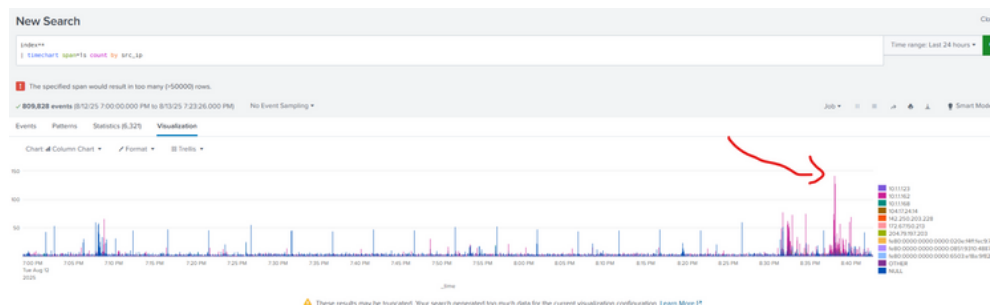


Figure 6.4: DDoS UDP Flood Detection in Splunk

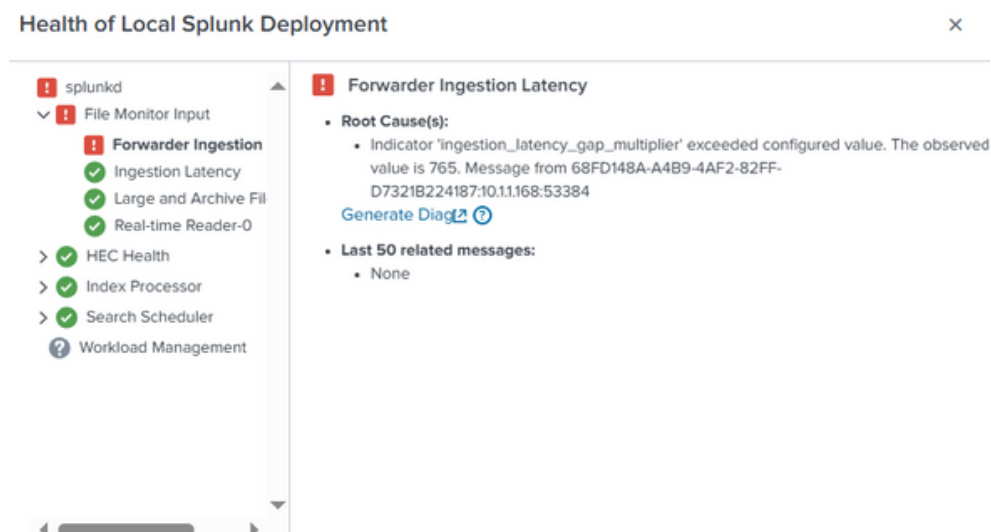


Figure 6.4: DDoS UDP Flood Detection stopped Splunk

7. Summary

This project aimed to demonstrate the use of Splunk Enterprise for centralized log management and security event monitoring. A virtual lab environment was created using Windows 10 Pro as the host (running Splunk Enterprise), an Ubuntu 24 virtual machine (with Splunk Universal Forwarder and Suricata IDS), and a Parrot Security OS virtual machine (as the attacker).

Multiple attack simulations were performed, including Nmap port scanning, SSH brute force attempts, XSS and SQL injection tests, and a UDP flood DDoS attack. Logs from Ubuntu were forwarded to Splunk via the Universal Forwarder, while Suricata provided network-level intrusion alerts.

Splunk successfully aggregated logs from both local and remote sources, enabling correlation between system events and network alerts. Each attack was detected and documented, proving the effectiveness of combining Splunk's SIEM capabilities with Suricata's IDS features.

The project highlights the importance of centralized log collection, real-time monitoring, and multi-layered detection to enhance cybersecurity visibility and incident response.