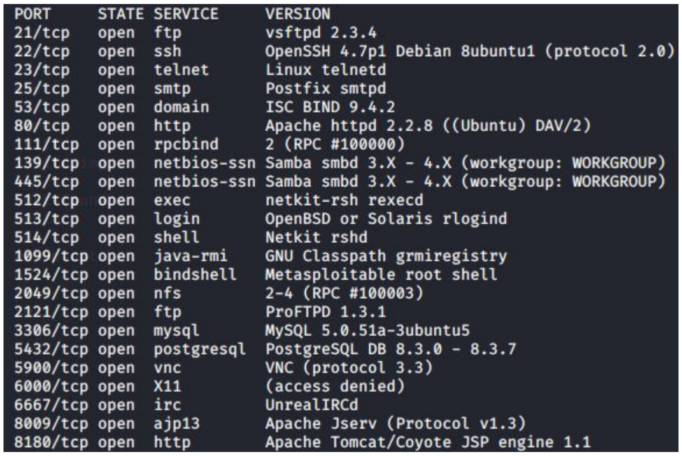
Penetration Test

Jack Kraemer

04.28.2020
INET 4165

SUMMARY OF FINDINGS

To begin, I would be using Kali Linux as the attacking machine and Metasploitable 2 as the target machine. I already know the IP address which is 10.10.10.2, but if I did not know the IP and I was on the same network, I would be using a network scanner to find the devices on the network. One can also use other information found to attack the machine such as finding any personal information on the user or software that the company would be using. Searching for domain addresses, specific software used, and physical security such as wifi passwords or biometric protection can help attackers find a way into the system. Below is the passive scanning to show the open ports that will be vulnerable to an attack.



```
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       OpenBSD or Solaris rlogind
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         2-4 (RPC #100003)
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  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
```

After using NMAP to scan the target machine for open ports, it can be seen that there are a lot of ports vulnerable for attack. The main ones that are commonly exploited are ftp, ssh, mysql, postgresql, and telnet. There are many reasons to not have these ports open and vulnerable to others without protection such as a firewall or antivirus to catch anything trying to exploit the open port. A service openly listening to a port will expose the service to exploits if the software is not secure, which allows attackers to obtain access to that service and can lead to access to the system.

Someone being able to access information regarding the open ports can find which service is listening on that port and then the attacker can research vulnerabilities related to that service. For example, on port 21, the VSFTPD v2.3.4 is a service that can be researched online and one can easily find that there is a backdoor in this service allowing attackers to obtain access to the root account. There is also a well known exploit with the Telnet service that if the username ends with a :) then it will open a listening port for anyone to be able to login. This shows that attackers can easily find vulnerabilities just by scanning the target machine, finding open ports and the service that is listening, and researching vulnerabilities known in these services.

FINDINGS

Exploit	Impact	Likelihood	Total Risk
FTP port 21	**HIGH**	**HIGH**	**Critical**
POSTGRESQL Port 5432	**HIGH**	**MEDIUM**	**HIGH**
MYSQL port 3306	**HIGH**	**MEDIUM**	**HIGH**
SAMBA Port 139	HIGH	**MEDIUM**	**HIGH**
TELNET Port 23	**HIGH**	**HIGH**	**Critical**

Exploit 1: VSFTPD v2.3.4 Backdoor

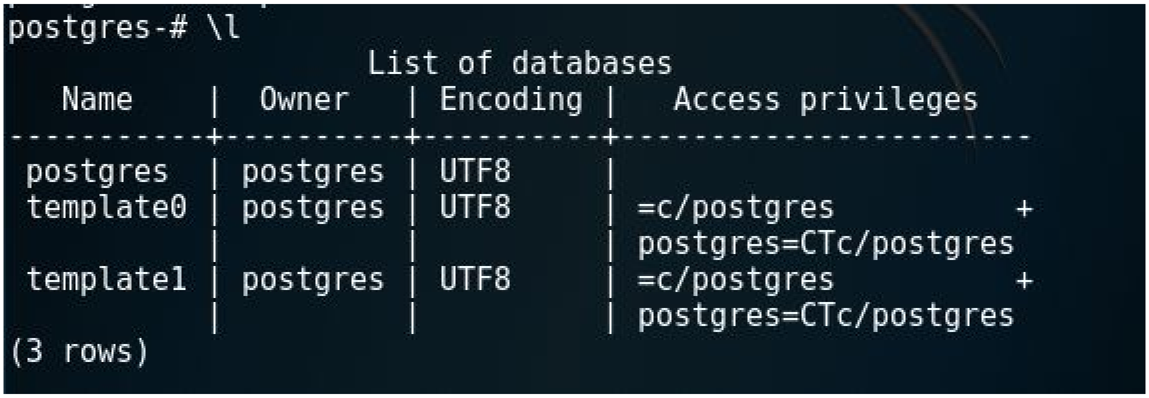
This is a very easy backdoor that attackers can use if the target machine has the 2.3.4 version of VSFTPD on port 21. As this target machine does have this version, all one would have to do is use the metasploit framework which has a built in exploit for VSFTPD. The attacker will only need to know the IP address of your machine and the port number, which is 21. Using this exploit, it will open a remote terminal right into the root account. Very easy exploit that has a high likelihood of being used and can have a large impact as the attacker will have access to the root account.

Exploit 2: MYSQL

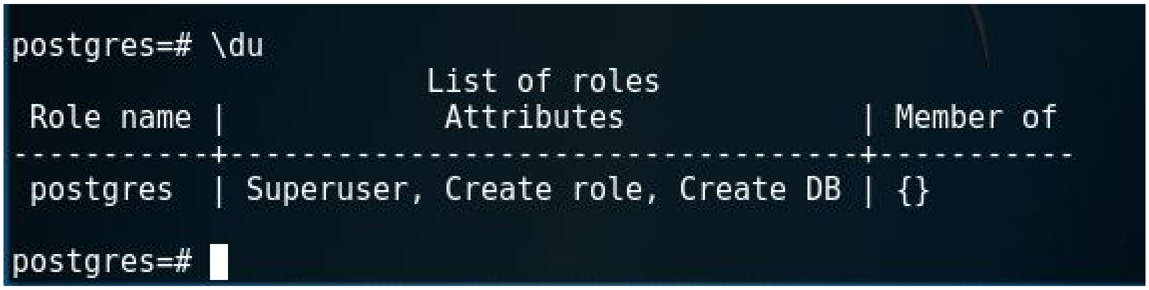
MySQL is a very popular and common database that a lot of applications use. If the MySQL version is more out of date or it has weak passwords, it can easily be attacked. To begin, an attacker would create two text files that contain possible passwords and usernames to test as the login for the victim's MySQL server. One would have to use CRUNCH or another word generator to create a randomized list. Again, using a built-in exploit in the Metasploit framework, an attacker can use the mysql_login exploit along with the generated username and password files to brute force the different combinations of logins. Once found, the attacker can use the successful credentials to login and do multiple things such as creating their own account for future logins and also extracting sensitive information that could be stored in the database.

Exploit 3: Postgresql Login

The Postgresql service is open on port 5432 and it is an open source database that, once compromised, can allow attackers to read and write system files and also execute code. Using NMAP, the attacker can find the version of the Postgresql, which is 8.3.0, allowing attackers to find vulnerabilities and exploits in that specific version. Using a Metasploit module that comes with premade credentials to test, an attacker can quickly run an exploit to brute force account information and obtain a login. Once successfully logged on, an attacker can obtain a list of databases and users shown below. An attacker can also execute code and obtain data stored in the databases as well as escalate their privileges to obtain more information as an admin.



```
postgres-# \l
                      List of databases
   Name    |  Owner   | Encoding |   Access privileges
-----------+----------+----------+-----------------------
 postgres  | postgres | UTF8     |
 template0 | postgres | UTF8     | =c/postgres          +
           |          |          | postgres=CTc/postgres
 template1 | postgres | UTF8     | =c/postgres          +
           |          |          | postgres=CTc/postgres
(3 rows)
```

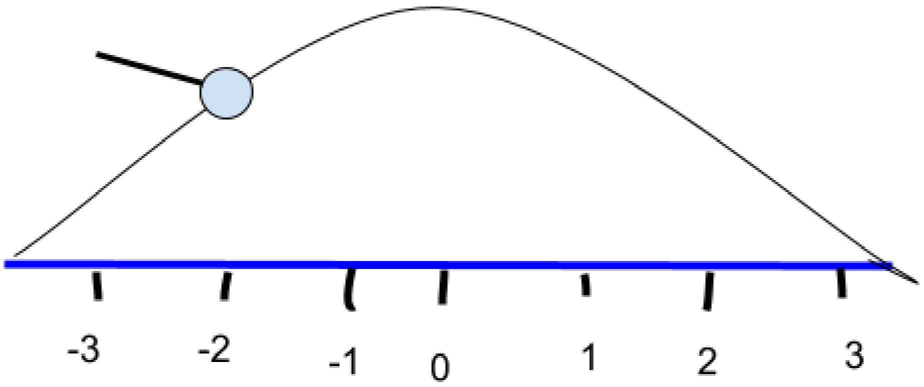


```
postgres=# \du
                       List of roles
 Role name |            Attributes             | Member of
-----------+-----------------------------------+-----------
 postgres  | Superuser, Create role, Create DB | {}

postgres=# 
```

COMPARATIVE ANALYSIS

After analyzing and exploiting the many vulnerabilities found within the victim machine, there are plenty of ways that an attacker could break in and obtain information from this machine. This environment would be given a weak rating as there are many vulnerabilities that make the machine open to attack. As this is a machine made to be vulnerable, it is pretty obvious that the machine would have a weak environment, but there are plenty of machines out there that are similar to this where the services are outdated or have simple credentials that can be easily exploited. Below is the overall rating of the environment, where -3 is far below average, 0 is average, and 3 is above average for current security posture. This environment is given that rating because of the multiple services that are vulnerable and easily exploitable that can be used to release critical information. Once an attacker has gotten into the services, there is no knowing what else they could do.



With all of the vulnerabilities shown through the penetration testing, there are a few things that can be done to prevent these exploits from damaging your machine. The biggest one is to keep your services updated, such as the VSFTPD service. Updating or patching these services can close backdoors or other issues that are open to attack. Other options that one can do to further secure their system is to have stronger passwords that are not easily guessable with brute force. Keeping your firewall up to date and running to make sure nothing unknown is able to get through. Monitoring user activity is also a big way to secure the machine as it allows the user and admin to make sure that there isn’t something going on without their knowledge such as creating new users or elevating the permissions of a certain user that should not have it. There are many improvements to be made to this machine with the ease of exploitation and the severity of the vulnerabilities.