

Unix User Enumeration

One of the first activities while conducting a penetration test in Unix environments is to perform a user enumeration in order to discover valid usernames. In this article we will examine how we can manually discover usernames based on the services that are running.

Lets say that we have perform a port scan with Nmap on our host and we have discover that the finder daemon is running on port 79.

```
root@bt:~# nmap -sV 192.168.1.80
Install
Starting Nmap 5.61TEST4 ( http://nmap.org ) at 2012-04-10 00:29 BST
Nmap scan report for 192.168.1.80
Host is up (0.00090s latency).
Not shown: 989 closed ports
PORT      STATE SERVICE          VERSION
7/tcp     open  echo
13/tcp    open  daytime
19/tcp    open  chargen          Linux chargen
21/tcp    open  ftp              vsftpd 2.0.4
22/tcp    open  ssh              OpenSSH 4.3 (protocol 1.99)
25/tcp    open  smtp             Sendmail 8.13.5/8.13.5
37/tcp    open  time             (32 bits)
79/tcp    open  finger           Debian fingerd
111/tcp   open  rpcbind (rpcbind V2) 2 (rpc #100000)
139/tcp   open  netbios-ssn      Samba smbd 3.X (workgroup: YORK)
445/tcp   open  netbios-ssn      Samba smbd 3.X (workgroup: YORK)
MAC Address: 00:50:56:BB:00:79 (VMware)
Service Info: Host: localhost.localdomain; OSs: Linux, Unix; CPE: cpe:/o:linux:kernel
```

Discovery of the Finger Service

We can use the finger command in order to enumerate the users on this remote machine. For example if we execute the command **finger @host** we will get the following output.

```
root@bt:~# finger @192.168.1.80
[192.168.1.80]
Login      Name      Tty      Idle   Login Time   Office   Office Phone
root       root      tty1     1:42   Apr 10 01:20
```

List the logged users on the remote host

As you can see the root user is the only account which is logged on the remote host. Now that we have a specific username we can use it in order to obtain more information about this user with the command **finger root@host**.

```

root@bt:~# finger root@192.168.1.80
[192.168.1.80]
Login: root                                Name: root
Directory: /root                          Shell: /bin/bash
On since Tue Apr 10 01:20 (BST) on tty1    2 hours 2 minutes idle
New mail received Tue Apr 10 02:20 2012 (BST)
      Unread since Sat Sep 18 10:37 2010 (BST)
No Plan.

```

Finger a specific user

As the image indicates the finger command obtained information about the name, the home directory, login name and shell. Also we can see that the root user doesn't have a **.plan** file.

Another effective use of the finger command is when you use it with the following syntax:
finger user@host

```

root@bt:~# finger user@192.168.1.80
[192.168.1.80]
Login: ftp                                Name: FTP User
Directory: /var/ftp                      Shell: /sbin/nologin
Never logged in.
No mail.
No Plan.

Login: netdump                            Name: Network Crash Dump user
Directory: /var/crash                    Shell: /bin/bash
Never logged in.
No mail.
No Plan.

Login: rpc                                Name: Portmapper RPC user
Directory: /                             Shell: /sbin/nologin
Never logged in.
No mail.
No Plan.

```

Enumerate all users with the string user

This specific command will enumerate all user accounts that have the string **user**. Alternatively you can use other words instead of user like **admin**, **account** and **project**.

Older versions of Solaris that run the finger daemon are affected by an enumeration bug. For example you can run the command **finger 0@host** and it will enumerate all users with an empty GCOS field in the password file. Additionally you can run **finger 'a b c d e f g h'@host** and it will enumerate all users on the remote target.

In SunOS there are RPC services that allow also user enumeration. For example the command **rusers** will return a list with the users that are logged into machines on the local network. Alternatively if you are looking for the list of a specific host you can combine it with **rusers -al host**.

```

root@bt:~# rusers
192.168.1.82      root
root@bt:~# rusers -al 192.168.1.82
root      192.168.1.82:console      Apr 10 14:25      1:05
root@bt:~#

```

rusers output

Another option is the **rwho** command which can be used also to enumerate network users. All the systems that are running the **rwhod** daemon will respond and an output will be produced of the users that are currently logged in to these systems. This service runs at 513 (UDP) port.

If you discover a host which is running an SMTP service (port 25) you can also use it for username enumeration. We can connect through telnet to the mail server and then we can execute the command help in order to see the available commands.

```

root@bt:~# telnet 192.168.1.80 25
Trying 192.168.1.80...
Connected to 192.168.1.80.
Escape character is '^]'.
220 localhost.localdomain ESMTP Sendmail 8.13.5/8.13.5; Tue, 10 Apr 2012 15:54:36 +0100
help
214-2.0.0 This is sendmail version 8.13.5
214-2.0.0 Topics:
214-2.0.0      HELO      EHLO      MAIL      RCPT      DATA
214-2.0.0      RSET      NOOP      QUIT      HELP      VRFY
214-2.0.0      EXPN      VERB      ETRN      DSN        AUTH
214-2.0.0      STARTTLS
214-2.0.0 For more info use "HELP <topic>".
214-2.0.0 To report bugs in the implementation send email to
214-2.0.0      sendmail-bugs@sendmail.org.
214-2.0.0 For local information send email to Postmaster at your site.
214 2.0.0 End of HELP info

```

SMTP - Commands

As you can see from the image above there are plenty of commands but the commands that we will need for the discovery of valid usernames are the **VRFY** and **EXPN**.

```

root@bt:~# telnet 192.168.1.82 25
Trying 192.168.1.82...
Connected to 192.168.1.82.
Escape character is '^]'.
220 unknown ESMTP Sendmail 8.13.4+Sun/8.13.3; Tue, 10 Apr 2012 17:15:56 +0100 (BST)
EXPN root
250 2.1.5 Super-User <root@unknown>
VRFY adm
250 2.1.5 Admin <adm@unknown>

```

Discover valid usernames through SMTP

The image above indicates that we have successfully verified the existence of two users root and admin.

Conclusion

In production systems it is almost impossible to find any of these services running due to this information leakage. However many Linux distributions include these daemons as part of their default installation.

In nowadays this process can be done automatically through the nmap script engine but it is good to know also how you can manually discover usernames in Unix systems. Also many commercial certifications are still requiring from you to know how to enumerate users with these commands.