Network Remote Control Project

WILSON LAU WEI QIANG

Telegram: @WilXIV	CFC190324
Wilson Lau	STUDENT

INTRODUCTION

Network research and monitoring are critical components managing and securing modern digital communications infrastructures. These activities involved observing, analyzing and managing network traffic to ensure performance, detect threats and safeguard security. Where cybersecurity upholds three key principles, Confidentiality, Integrity, and Availability. Each elements addresses a crucial aspect of information security, helping organizations to protect their data from threats, ensuring it remains secure and reliable, maintain a accurate and trustworthy information and ensure that resources are accessible when needed.

METHODOLOGY

Chmod chmod o+w /var/log Chmod chmod o+w /var/log –– Command line used to change files/directories permissions.Command line used to change files/directories permissions.

•• OO — stands for other user who are not the ‘owner’ of the files or file group.stands for other user who are not the ‘owner’ of the files or file group.

•• ++W W — Add writes permissions.Add writes permissions.

•• /var/log /var/log ---- Directory of permissions it changing.Directory of permissions it is changing .

! -x ./nipe

! -x ./nipe• ! – Not, which checking not executable.

•• !-x — Not, which checking not executable. Check if

a file/commands that exists/executable. •• -

nipex – Check if a file/commands that exists/executable. –

Directory of executable.

Geoiplookup • nipe -- ¯Look up Directory of executable. information’s for a IP address such as country/region/city/latitude/longitude.

Geoiplookup -- Look up informations for a IP address such as country/region/city/latitude/longitude.

SshpassSshpass — Secure Shell facilities the nonSecure Shell facilities the non--interactive passing of a password without manual password input.interactive passing of a password without manual password input.

-
- StrictHostKeyChecking=no StrictHostKeyChecking=no — More convenient. Bypass the host key verification without checking the More convenient. Bypass the host key verification without checking the 'known_hosts' file thus no prompt and will automatically connect to the server.'known hosts' file thus no prompt and will automatically connect to the server.
- nmap $domain > /tmp/nmapscan_results.txt $domain > /tmp/nmapscan_results.txt ---- It allow commands to execute It allow commands to execute
- nmap

ForLoopForLoop

-
- check_nipe_connection() check_nipe_connection() — For loop function.For loop function.
- attempt=1; attempt<=$MAX_RETRIES; attempt++ ---- CheckCheck if 'attempt' is less or equal to 'MAX_RETRIES' if 'attempt' is less or equal to 'MAX_RETRIES'
- attempt=1; attempt<=$MAX_RETRIES; attempt++
  and increment by 1 after each loop.and increment by 1 after each loop.
- sudo perl nipe.pl start – Start the Nipe connection with superuser privileges.

Curl -S -- command line tool is widely used to interact with web services and API.

- -s -- The `-s` flag in `curl` stands for "silent mode"

Nmap – Scan host for open ports

- -Pn – (No Ping) Skip ping to host and assume host is up. Useful if host has firewall that is blocking ping requests.
- -sV – (Service Version Detection) Determine the version of the services running on the port. Analyse the responses to match with it database of known service signatures
- > /var/log/ -- " > " is to saved the scan results and log into <directory><filesname>.

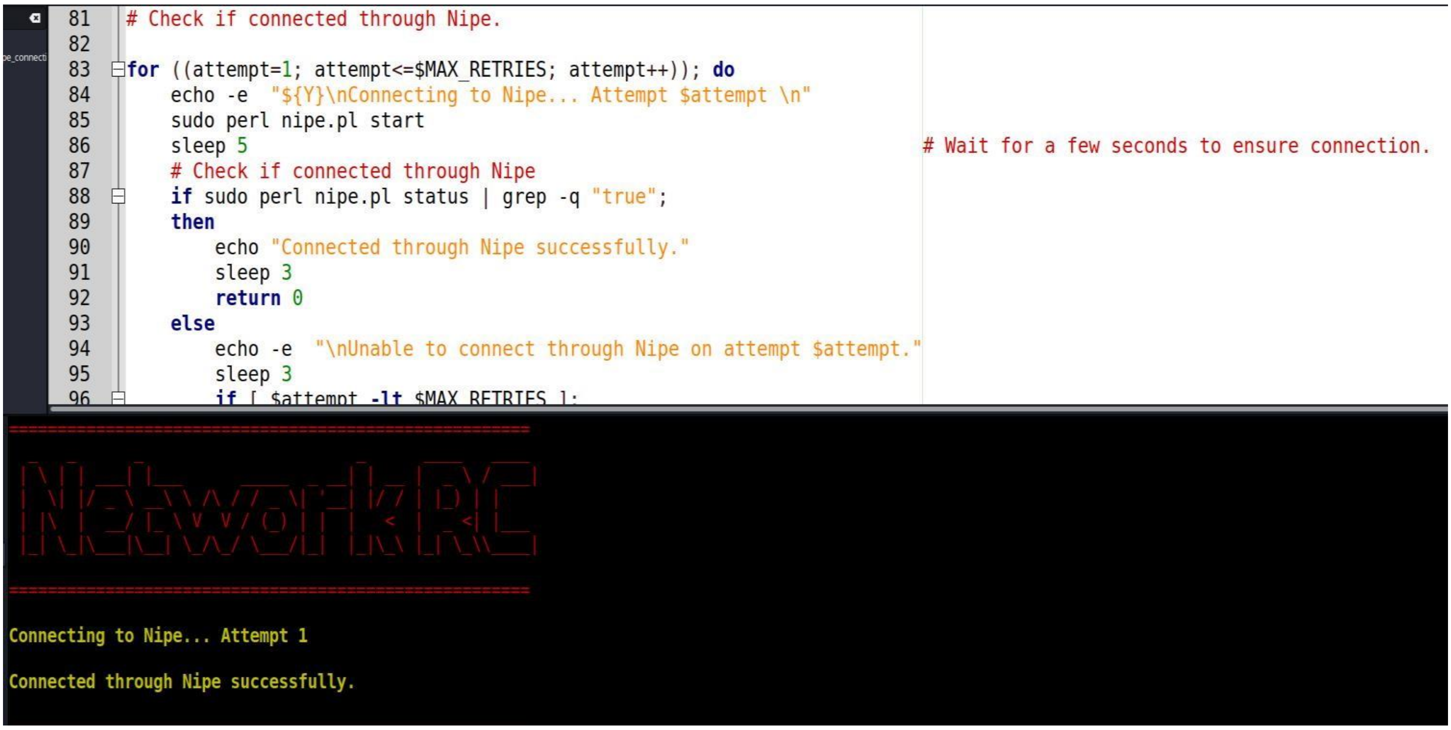
Scp – (Secure Copy) transfer files between hosts

- -o StrictHostKeyChecking -- – More convenient. Bypass the host key verification without checking the 'known hosts' file thus no prompt and will automatically connect to the server.

timestamp=$(date '+%A %Y-%m-%d %H:%M:%S') – Generate current date and time in specific format

- %A – Weekday.
- %Y – Four-digit year.
- %m – Two-digit month.
- %d – Two-digit day.
- %H – Two-digit hour.
- %M – Two-digit minute.
- %S – Two-digit second.

DISCUSSION



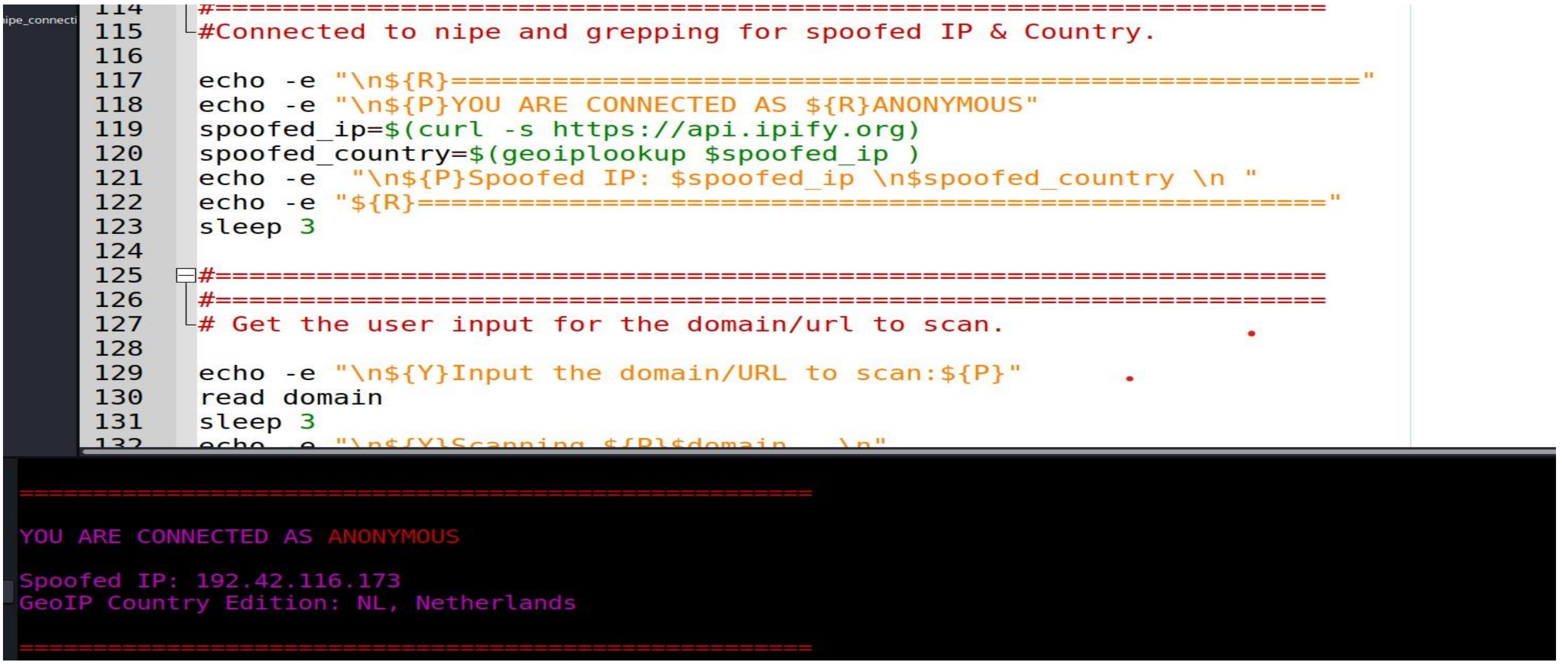
```
# Check if connected through Nipe.

for ((attempt=1; attempt<=$MAX_RETRIES; attempt++)); do
    echo -e  "${Y}\nConnecting to Nipe... Attempt $attempt \n"
    sudo perl nipe.pl start
    sleep 5                                                                         # Wait for a few seconds to ensure connection.
    # Check if connected through Nipe
    if sudo perl nipe.pl status | grep -q "true";
    then
        echo "Connected through Nipe successfully."
        sleep 3
        return 0
    else
        echo -e  "\nUnable to connect through Nipe on attempt $attempt."
        sleep 3
        if [ $attempt -lt $MAX_RETRIES ];
```

```
Connecting to Nipe... Attempt 1

Connected through Nipe successfully.
```

For Loop Nipe connection

- Connecting to Nipe service required afew tried as it may sometime failed to establish a connection due to variety of reasons. Using For loop to automate the retry mechanism reducing the need for user intervention. With implementing controlled delays (sleep) between retries, it prevents overwhelming attempt requests to Tor network. By providing ($attempts) after each attempt, users can see the progression and helps debugging or monitoring the connection process and able to identify any attempt behaves differently which is very valuable for troubleshooting.

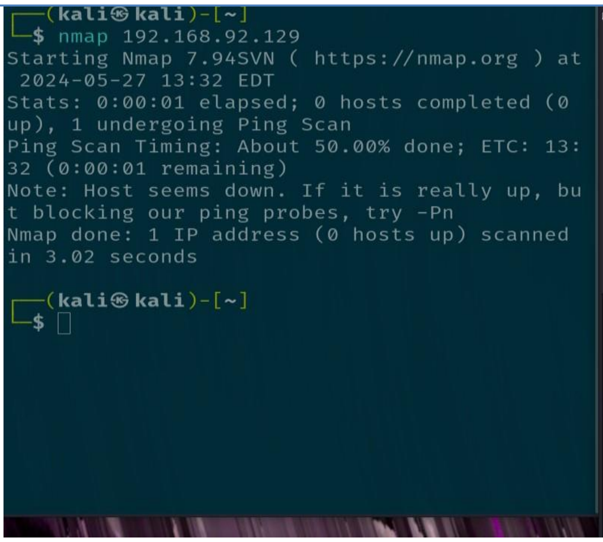
- This script segments provide user information's an automated way to check if connected to internet through Nipe successfully. By using "curl -s" and "geoiplookup" it can retrieve and display the user spoofed IP address and country as it is crucial to user who prioritize anonymity and privacy as to be sure if user IP is not exposed



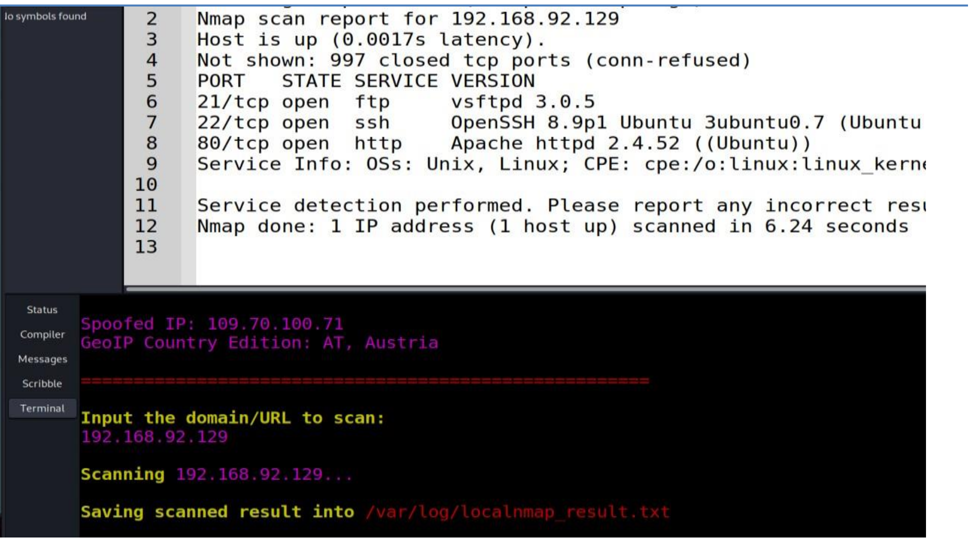
```
#=================================================================
#Connected to nipe and grepping for spoofed IP & Country.

echo -e "\n${R}=================================================================="
echo -e "\n${P}YOU ARE CONNECTED AS ${R}ANONYMOUS"
spoofed_ip=$(curl -s https://api.ipify.org)
spoofed_country=$(geoiplookup $spoofed_ip )
echo -e  "\n${P}Spoofed IP: $spoofed_ip \n$spoofed_country \n "
echo -e "${R}================================================================"
sleep 3

#=================================================================
#=================================================================
# Get the user input for the domain/url to scan.

echo -e "\n${Y}Input the domain/URL to scan:${P}"
read domain
sleep 3
echo -e "\n${Y}Scanning ${P}$domain...\n"
```

```
===============================================================

YOU ARE CONNECTED AS ANONYMOUS

Spoofed IP: 192.42.116.173
GeoIP Country Edition: NL, Netherlands

===============================================================
```

Connected as anonymous



```
┌──(kali㉿kali)-[~]
└─$ nmap 192.168.92.129
Starting Nmap 7.94SVN ( https://nmap.org ) at
 2024-05-27 13:32 EDT
Stats: 0:00:01 elapsed; 0 hosts completed (0
up), 1 undergoing Ping Scan
Ping Scan Timing: About 50.00% done; ETC: 13:
32 (0:00:01 remaining)
Note: Host seems down. If it is really up, bu
t blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned
in 3.02 seconds

┌──(kali㉿kali)-[~]
└─$ 
```



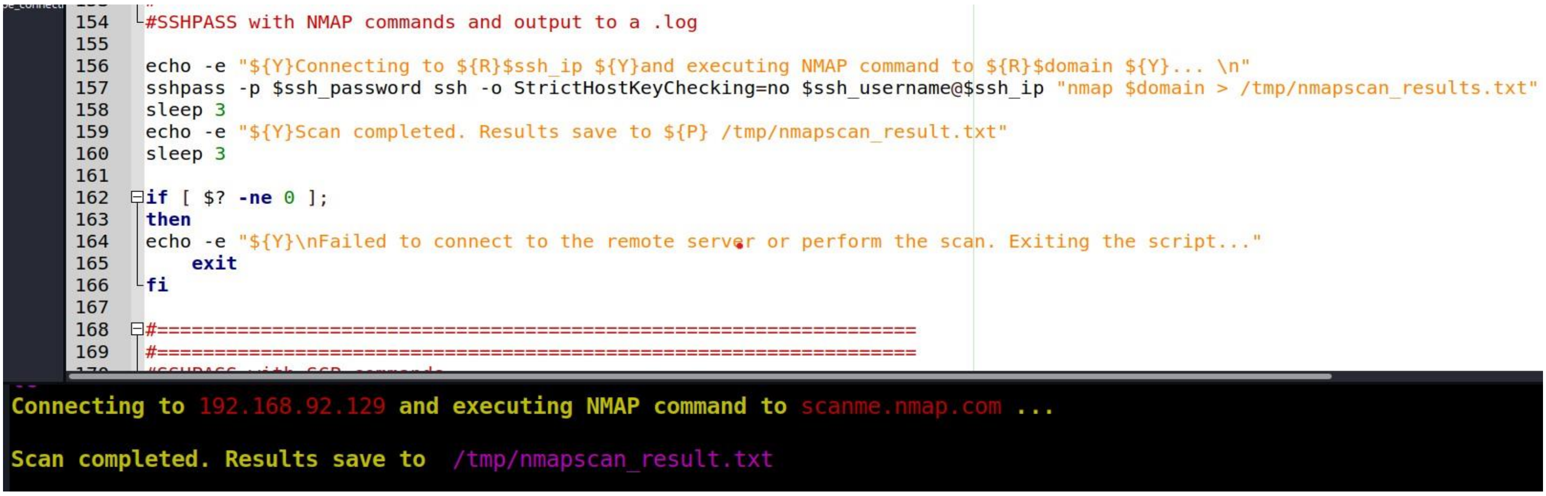
```
Nmap scan report for 192.168.92.129
Host is up (0.0017s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.5
22/tcp open  ssh     OpenSSH 8.9p1 Ubuntu 3ubuntu0.7 (Ubuntu
80/tcp open  http    Apache httpd 2.4.52 ((Ubuntu))
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kern

Service detection performed. Please report any incorrect resu
Nmap done: 1 IP address (1 host up) scanned in 6.24 seconds
```

```
Spoofed IP: 109.70.100.71
GeoIP Country Edition: AT, Austria

===============================================================

Input the domain/URL to scan:
192.168.92.129

Scanning 192.168.92.129...

Saving scanned result into /var/log/localnmap_result.txt
```

NMAP

- By adding -Pn we can skip a ICMP echo request to host as it might have firewall to block nmap ping scan so it can directly start scanning the target host for ports. (-sV) enable version detection of the services that runs on the target ports. It can be identifying vulnerabilities associated with specific services version as it crucial for patches and mitigating potential risks.

- SSHPASS (Secure Shell) is a protocol for secure remote login but its not highly recommended due to its security concerns. SSHPASS require users to store passwords in plain text in a script or command which will pose a high security risk while SSH uses key cryptography which is more secure than text based authentication but due convenience or necessity in certain scenarios which can be more simpler and straightforward for automated processes. It also enable its automation for password promopting which will result in the script get stuck at this point waiting for input and stop without completing its execution.

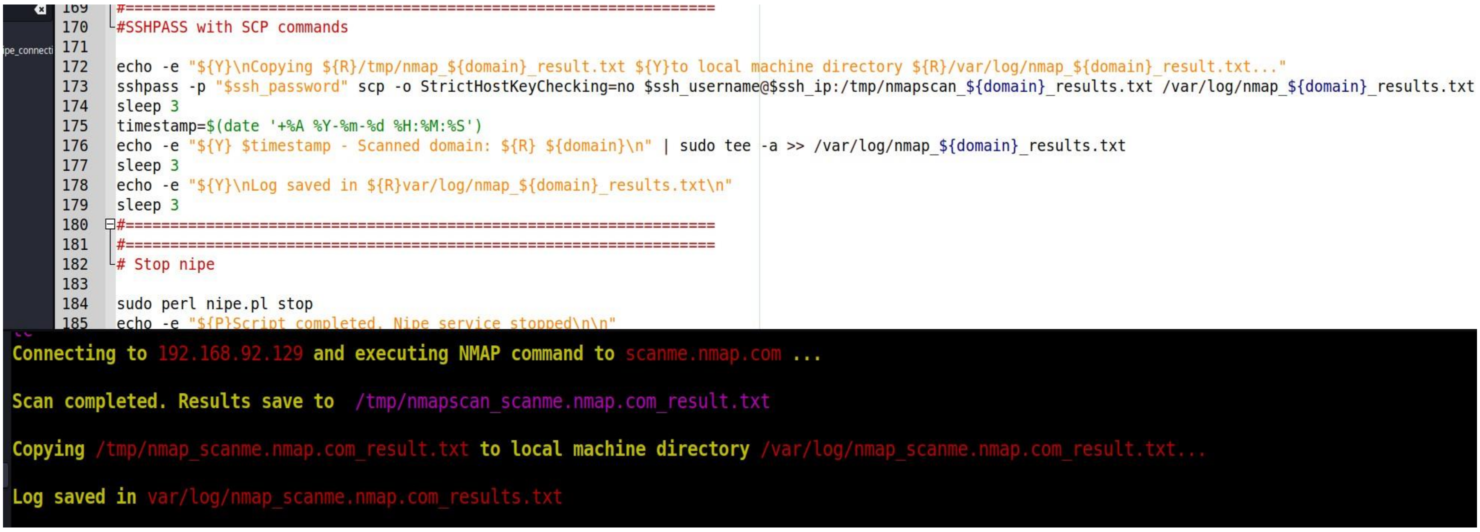


```
#SSHPASS with NMAP commands and output to a .log

echo -e "${Y}Connecting to ${R}$ssh_ip ${Y}and executing NMAP command to ${R}$domain ${Y}... \n"
sshpass -p $ssh_password ssh -o StrictHostKeyChecking=no $ssh_username@$ssh_ip "nmap $domain > /tmp/nmapscan_results.txt"
sleep 3
echo -e "${Y}Scan completed. Results save to ${P} /tmp/nmapscan_result.txt"
sleep 3

if [ $? -ne 0 ];
then
echo -e "${Y}\nFailed to connect to the remote server or perform the scan. Exiting the script..."
    exit
fi

#=================================================================
#=================================================================
```

```
Connecting to 192.168.92.129 and executing NMAP command to scanme.nmap.com ...

Scan completed. Results save to  /tmp/nmapscan_result.txt
```

SSH PASS

- As SCP command require password based authentication, it can also grab the stored user input previously for its domain, username and password to connect to the remote server and copy the file over to the local machine. The "-o StrictHostKeyChecking=no" disable host key checking making the automation smoother as it bypasses a crucial security measure.

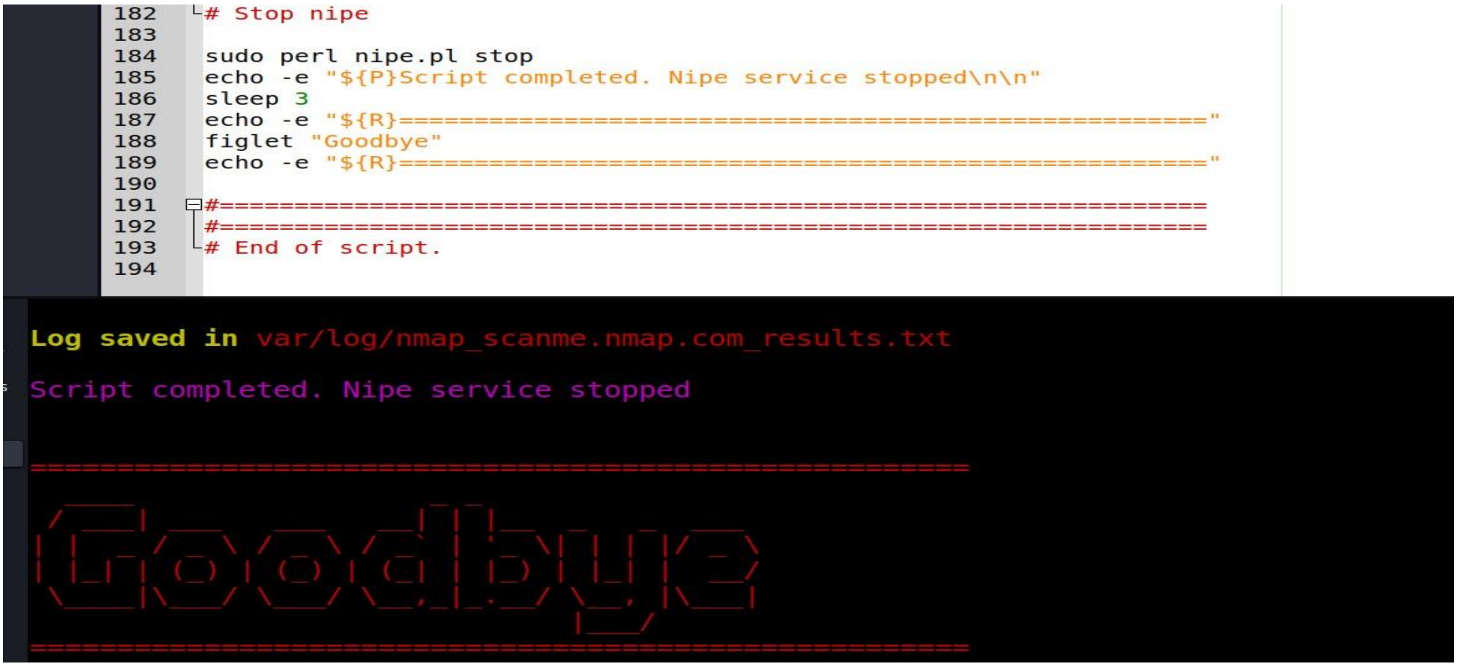


```
#=================================================================
#SSHPASS with SCP commands

echo -e "${Y}\nCopying ${R}/tmp/nmap_${domain}_result.txt ${Y}to local machine directory ${R}/var/log/nmap_${domain}_result.txt..."
sshpass -p "$ssh_password" scp -o StrictHostKeyChecking=no $ssh_username@$ssh_ip:/tmp/nmapscan_${domain}_results.txt /var/log/nmap_${domain}_results.txt
sleep 3
timestamp=$(date '+%A %Y-%m-%d %H:%M:%S')
echo -e "${Y} $timestamp - Scanned domain: ${R} ${domain}\n" | sudo tee -a >> /var/log/nmap_${domain}_results.txt
sleep 3
echo -e "${Y}\nLog saved in ${R}var/log/nmap_${domain}_results.txt\n"
sleep 3
#=================================================================
#=================================================================
# Stop nipe

sudo perl nipe.pl stop
echo -e "${P}Script completed. Nipe service stopped\n\n"
```

```
Connecting to 192.168.92.129 and executing NMAP command to scanme.nmap.com ...

Scan completed. Results save to  /tmp/nmapscan_scanme.nmap.com_result.txt

Copying /tmp/nmap_scanme.nmap.com_result.txt to local machine directory /var/log/nmap_scanme.nmap.com_result.txt...

Log saved in var/log/nmap_scanme.nmap.com_results.txt
```

SCP (SECURE COPY)

- After getting all the logs files over to user local machine, its important and best practices to do a script cleanup, in this case stopping the Nipe connection ensuring any resources or processes started by the script are properly terminated and informed user about the end of script.

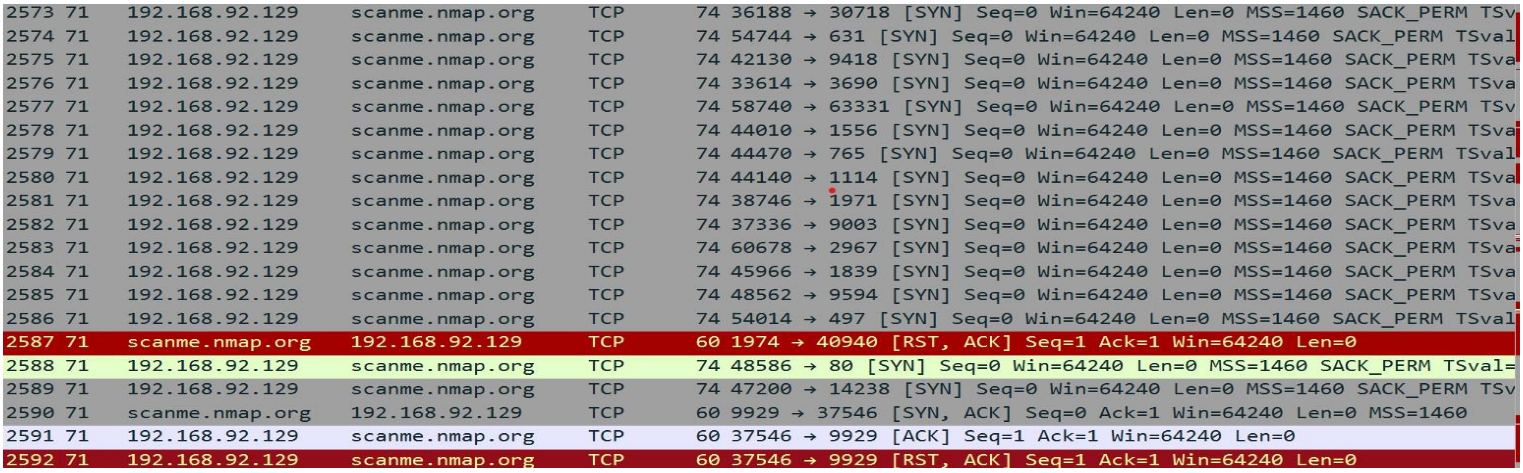


```
# Stop nipe

sudo perl nipe.pl stop
echo -e "${P}Script completed. Nipe service stopped\n\n"
sleep 3
echo -e "${R}============================================================"
figlet "Goodbye"
echo -e "${R}============================================================"

#===================================================================
#===================================================================
# End of script.
```

```
Log saved in var/log/nmap_scanme.nmap.com_results.txt

Script completed. Nipe service stopped


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

SCRIPT STOP

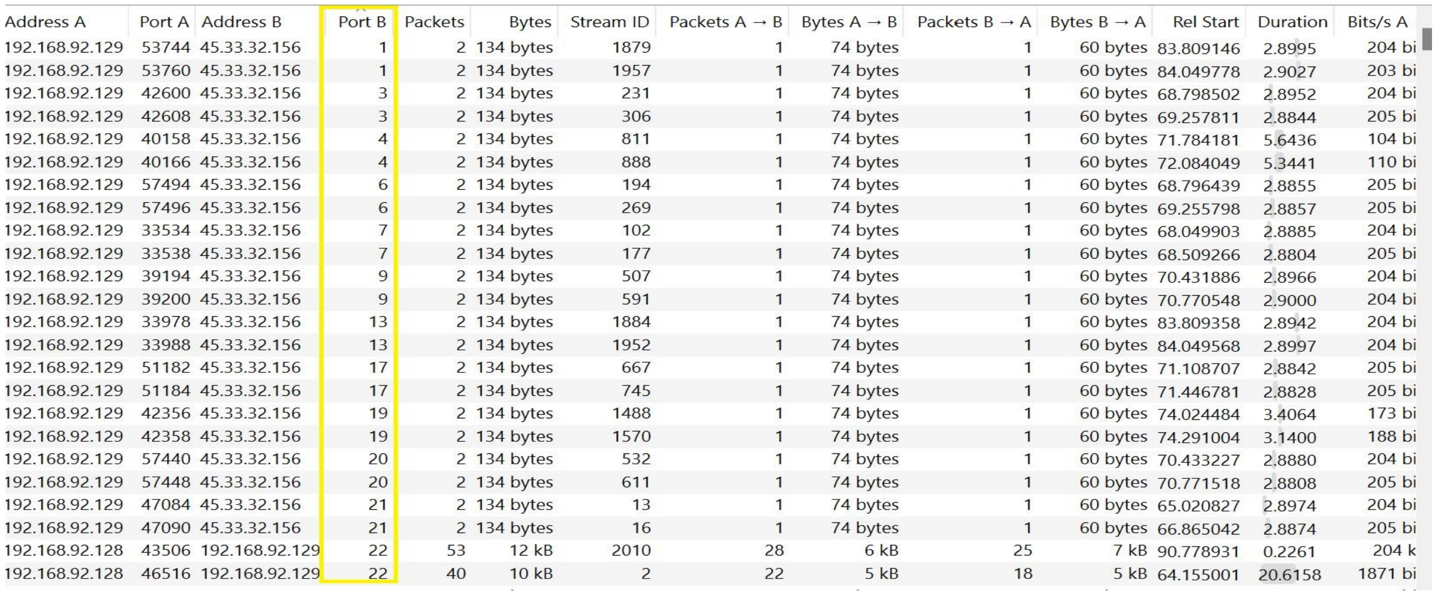
NETWORK TRAFFIC (WIRESHARK)



```
2573 71   192.168.92.129   scanme.nmap.org   TCP   74 36188 → 30718 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSv
2574 71   192.168.92.129   scanme.nmap.org   TCP   74 54744 → 631 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval
2575 71   192.168.92.129   scanme.nmap.org   TCP   74 42130 → 9418 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSva
2576 71   192.168.92.129   scanme.nmap.org   TCP   74 33614 → 3690 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSva
2577 71   192.168.92.129   scanme.nmap.org   TCP   74 58740 → 63331 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSv
2578 71   192.168.92.129   scanme.nmap.org   TCP   74 44010 → 1556 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSva
2579 71   192.168.92.129   scanme.nmap.org   TCP   74 44470 → 765 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval
2580 71   192.168.92.129   scanme.nmap.org   TCP   74 44140 → 1114 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSva
2581 71   192.168.92.129   scanme.nmap.org   TCP   74 38746 → 1971 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSva
2582 71   192.168.92.129   scanme.nmap.org   TCP   74 37336 → 9003 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSva
2583 71   192.168.92.129   scanme.nmap.org   TCP   74 60678 → 2967 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSva
2584 71   192.168.92.129   scanme.nmap.org   TCP   74 45966 → 1839 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSva
2585 71   192.168.92.129   scanme.nmap.org   TCP   74 48562 → 9594 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSva
2586 71   192.168.92.129   scanme.nmap.org   TCP   74 54014 → 497 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval
2587 71   scanme.nmap.org  192.168.92.129    TCP   60 1974 → 40940 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0
2588 71   192.168.92.129   scanme.nmap.org   TCP   74 48586 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=
2589 71   192.168.92.129   scanme.nmap.org   TCP   74 47200 → 14238 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSv
2590 71   scanme.nmap.org  192.168.92.129    TCP   60 9929 → 37546 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
2591 71   192.168.92.129   scanme.nmap.org   TCP   60 37546 → 9929 [ACK] Seq=1 Ack=1 Win=64240 Len=0
2592 71   192.168.92.129   scanme.nmap.org   TCP   60 37546 → 9929 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0
```

NMAP

- NMAP uses TCP protocol to scan for open ports. In order to see suspicious traffic, we can use WIRESHAK to see any 3 way handshake scan packet to see if ports are scanned for open. If ports are close, its normally send a SYN without any SYN,ACK back but when it does, it mean the ports are opened and listening for connection and the attacker machine will send a RST/ACK (Reset / Acknowledged) back to host thus 3 way handshake will be formed. Attacker can also use Stealth scan <Nmap -sS>, it will not do a 3 way handshake as it will not be sending <ACK> back to host, this way its less detectable by intrusion detection systems.

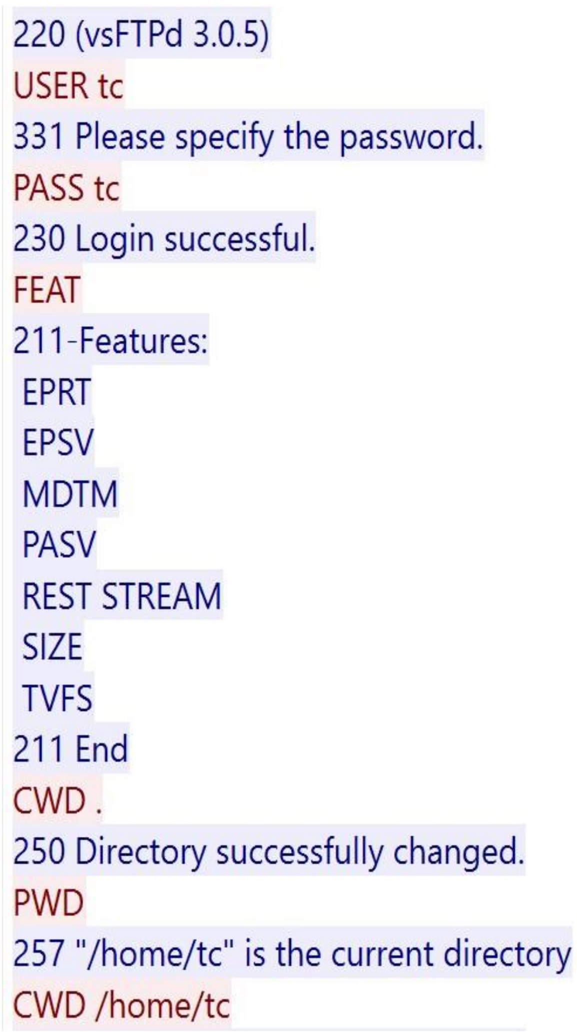


Address A	Port A	Address B	Port B	Packets	Bytes	Stream ID	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A
192.168.92.129	53744	45.33.32.156	1	2	134 bytes	1879	1	74 bytes	1	60 bytes	83.809146	2.8995	204 bi
192.168.92.129	53760	45.33.32.156	1	2	134 bytes	1957	1	74 bytes	1	60 bytes	84.049778	2.9027	203 bi
192.168.92.129	42600	45.33.32.156	3	2	134 bytes	231	1	74 bytes	1	60 bytes	68.798502	2.8952	204 bi
192.168.92.129	42608	45.33.32.156	3	2	134 bytes	306	1	74 bytes	1	60 bytes	69.257811	2.8844	205 bi
192.168.92.129	40158	45.33.32.156	4	2	134 bytes	811	1	74 bytes	1	60 bytes	71.784181	5.6436	104 bi
192.168.92.129	40166	45.33.32.156	4	2	134 bytes	888	1	74 bytes	1	60 bytes	72.084049	5.3441	110 bi
192.168.92.129	57494	45.33.32.156	6	2	134 bytes	194	1	74 bytes	1	60 bytes	68.796439	2.8855	205 bi
192.168.92.129	57496	45.33.32.156	6	2	134 bytes	269	1	74 bytes	1	60 bytes	69.255798	2.8857	205 bi
192.168.92.129	33534	45.33.32.156	7	2	134 bytes	102	1	74 bytes	1	60 bytes	68.049903	2.8885	204 bi
192.168.92.129	33538	45.33.32.156	7	2	134 bytes	177	1	74 bytes	1	60 bytes	68.509266	2.8804	205 bi
192.168.92.129	39194	45.33.32.156	9	2	134 bytes	507	1	74 bytes	1	60 bytes	70.431886	2.8966	204 bi
192.168.92.129	39200	45.33.32.156	9	2	134 bytes	591	1	74 bytes	1	60 bytes	70.770548	2.9000	204 bi
192.168.92.129	33978	45.33.32.156	13	2	134 bytes	1884	1	74 bytes	1	60 bytes	83.809358	2.8942	204 bi
192.168.92.129	33988	45.33.32.156	13	2	134 bytes	1952	1	74 bytes	1	60 bytes	84.049568	2.8997	204 bi
192.168.92.129	51182	45.33.32.156	17	2	134 bytes	667	1	74 bytes	1	60 bytes	71.108707	2.8842	205 bi
192.168.92.129	51184	45.33.32.156	17	2	134 bytes	745	1	74 bytes	1	60 bytes	71.446781	2.8828	205 bi
192.168.92.129	42356	45.33.32.156	19	2	134 bytes	1488	1	74 bytes	1	60 bytes	74.024484	3.4064	173 bi
192.168.92.129	42358	45.33.32.156	19	2	134 bytes	1570	1	74 bytes	1	60 bytes	74.291004	3.1400	188 bi
192.168.92.129	57440	45.33.32.156	20	2	134 bytes	532	1	74 bytes	1	60 bytes	70.433227	2.8880	204 bi
192.168.92.129	57448	45.33.32.156	20	2	134 bytes	611	1	74 bytes	1	60 bytes	70.771518	2.8808	205 bi
192.168.92.129	47084	45.33.32.156	21	2	134 bytes	13	1	74 bytes	1	60 bytes	65.020827	2.8974	204 bi
192.168.92.129	47090	45.33.32.156	21	2	134 bytes	16	1	74 bytes	1	60 bytes	66.865042	2.8874	205 bi
192.168.92.128	43506	192.168.92.129	22	53	12 kB	2010	28	6 kB	25	7 kB	90.778931	0.2261	204 k
192.168.92.128	46516	192.168.92.129	22	40	10 kB	2	22	5 kB	18	5 kB	64.155001	20.6158	1871 bi

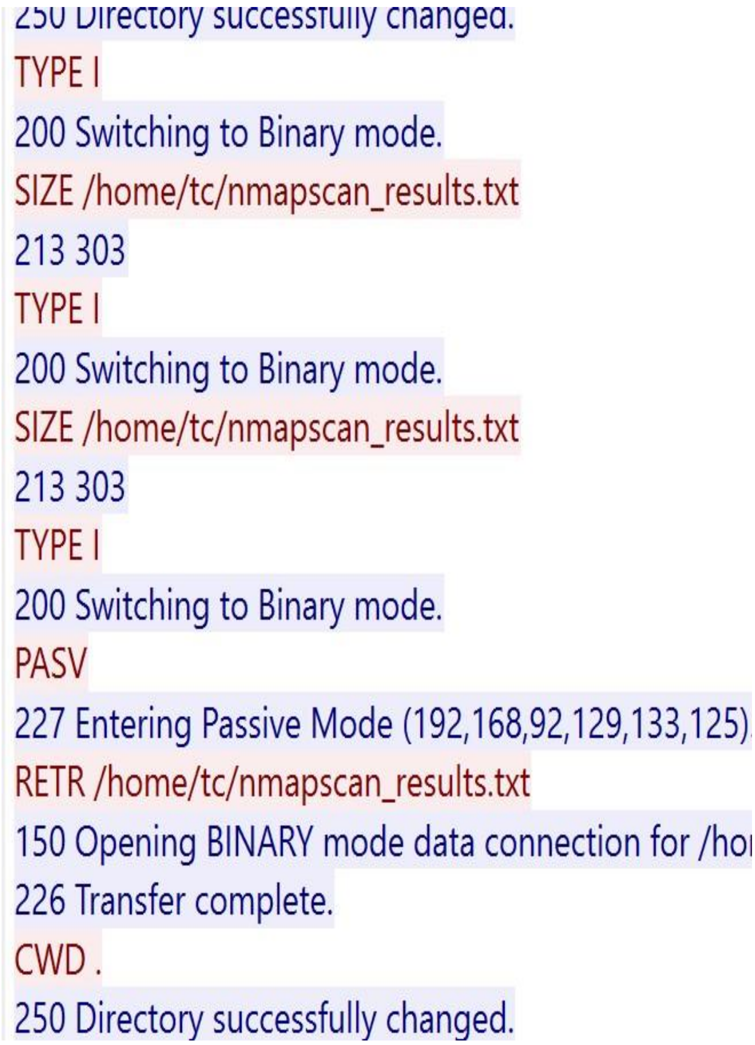
NMAP

- If we go to Wireshark>Statistics>Conversations under ports A/B tab, we can see multiple ports by increasement of every 1-3 number. If you sees an IP attempting to connect too many different ports in a short period, it’s a strong indicator of a port scan.

FTP VS



```
220 (vsFTPd 3.0.5)
USER tc
331 Please specify the password.
PASS tc
230 Login successful.
FEAT
211-Features:
 EPRT
 EPSV
 MDTM
 PASV
 REST STREAM
 SIZE
 TVFS
211 End
CWD .
250 Directory successfully changed.
PWD
257 "/home/tc" is the current directory
CWD /home/tc
```

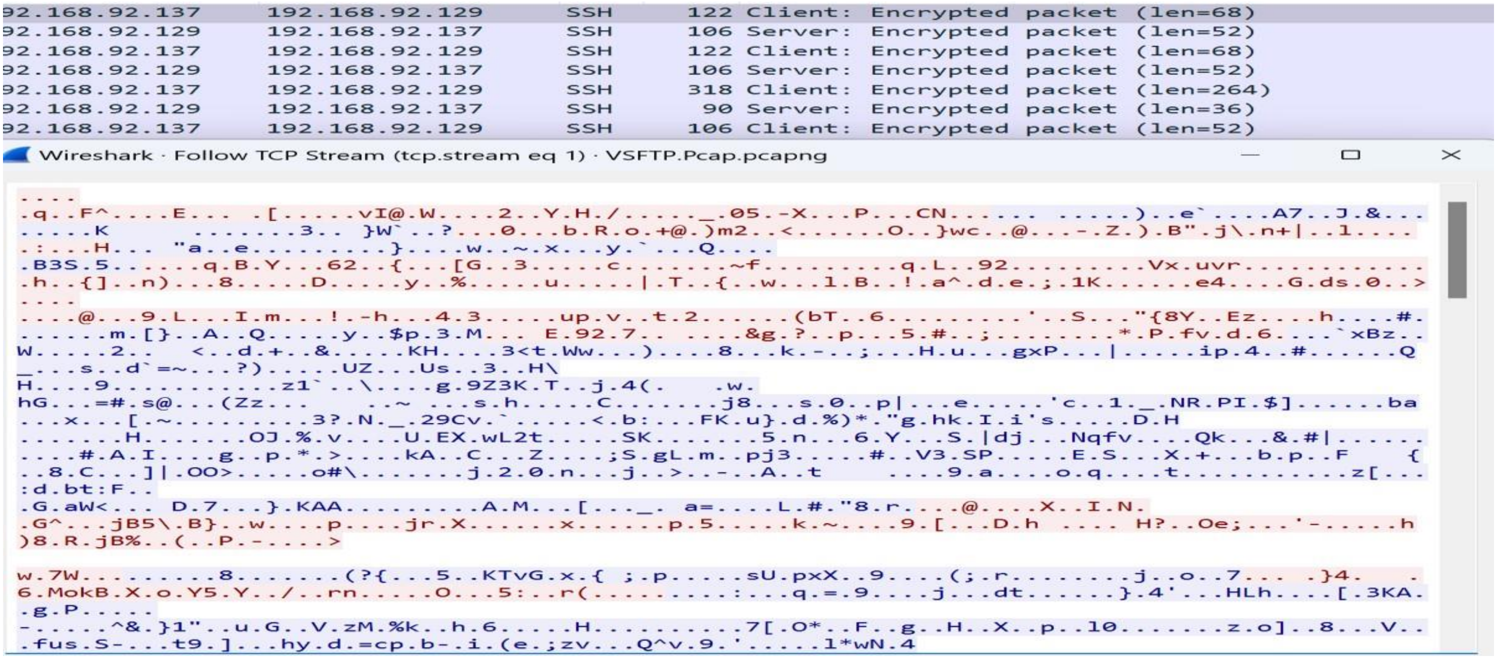


```
250 Directory successfully changed.
TYPE I
200 Switching to Binary mode.
SIZE /home/tc/nmapscan_results.txt
213 303
TYPE I
200 Switching to Binary mode.
SIZE /home/tc/nmapscan_results.txt
213 303
TYPE I
200 Switching to Binary mode.
PASV
227 Entering Passive Mode (192,168,92,129,133,125).
RETR /home/tc/nmapscan_results.txt
150 Opening BINARY mode data connection for /ho
226 Transfer complete.
CWD .
250 Directory successfully changed.
```

VSFTP

- One of the main disadvantages of FTP over SFTP is the lack of built-in encryptions for data transmission. Over WIRESHARK > ftp filter > Follow stream, we can see it transmit data including username and password in plain text. This make it highly susceptible to man in the middle attack {MITM} where attacker can easily capture and read transmitted data.

- By using VSFTPD, everything is encrypted using SSL/TLS and it also limit the ability to access to specific directories. File transferring is also secured and protect sensitive data from unauthorized access.



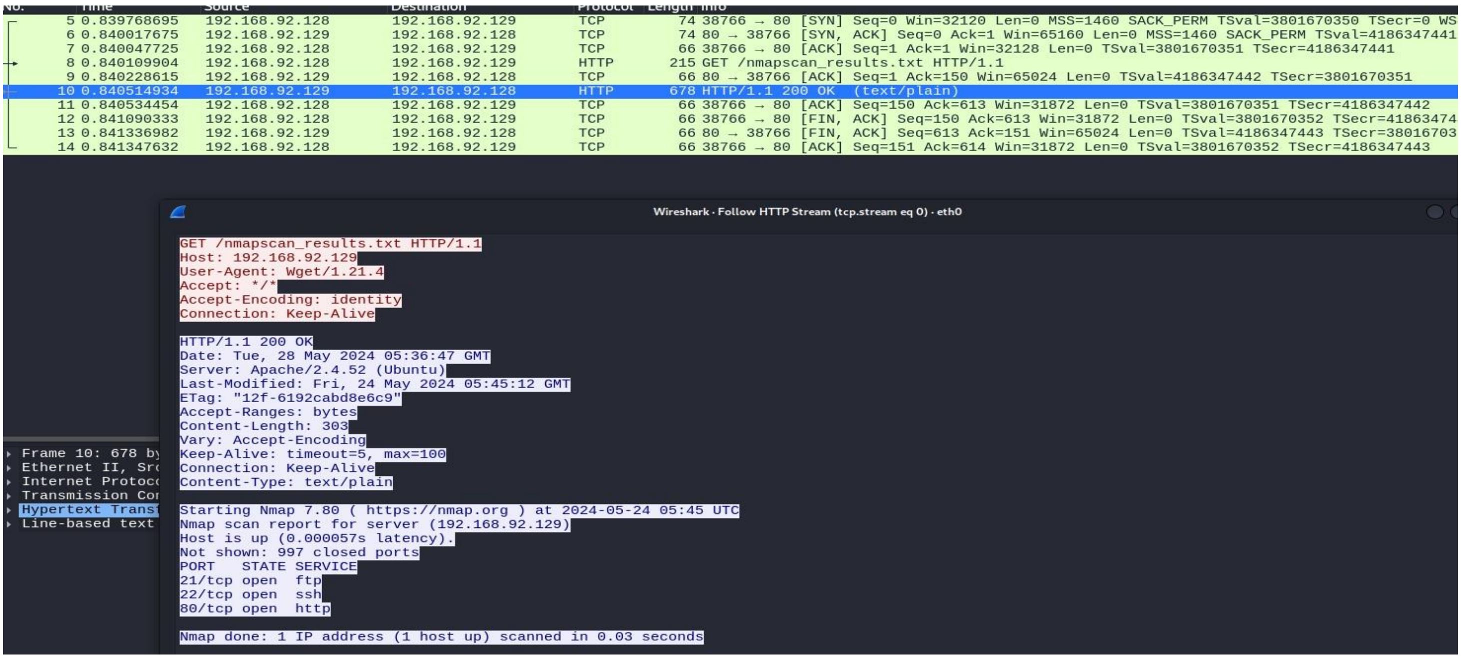
```
92.168.92.137     192.168.92.129     SSH     122 Client: Encrypted packet (len=68)
92.168.92.129     192.168.92.137     SSH     106 Server: Encrypted packet (len=52)
92.168.92.137     192.168.92.129     SSH     122 Client: Encrypted packet (len=68)
92.168.92.129     192.168.92.137     SSH     106 Server: Encrypted packet (len=52)
92.168.92.137     192.168.92.129     SSH     318 Client: Encrypted packet (len=264)
92.168.92.129     192.168.92.137     SSH      90 Server: Encrypted packet (len=36)
92.168.92.137     192.168.92.129     SSH     106 Client: Encrypted packet (len=52)
```

Wireshark · Follow TCP Stream (tcp.stream eq 1) · VSFTP.Pcap.pcapng

```
....
.q..F^....E... .[.....vI@.W....2..Y.H./......_.05.-X...P...CN...... .....)..e`....A7..J.&...
.....K      ........3.. }W`..?...0...b.R.o.+@.)m2..<.......O..}wc..@...-.Z.).B".j\.n+|..1....
.:...H... "a..e..........}....w..~.x...y.`...Q....
.B3S.5........q.B.Y...62..{...[G..3.....c........~f...........q.L..92..........Vx.uvr............
.h..{]..n)...8......D.....y..%......u......|.T..{..w...1.B..!.a^.d.e.;.1K......e4....G.ds.0..>
....
....@...9.L...I.m...!.-h...4.3......up.v..t.2.......(bT..6...........'..S..."{8Y..Ez.....h.....#.
.......m.[}..A..Q.....y..$p.3.M... E.92.7.. ....&g.?..p...5.#..;.........*.P.fv.d.6....`xBz..
W.....2..  <..d.+..&.....KH....3<t.Ww...)....8...k.-..;...H.u...gxP...|......ip.4..#.......Q
_...s..d`=~...?).....UZ...Us..3..H\
H....9...............z1`..\....g.9Z3K.T..j.4(.    .w.
hG...=#.s@...(Zz...     ..~ ...s.h.....C........j8...s.0..p|...e......'c..1._.NR.PI.$]......ba
...x...[.~...........3?.N._.29Cv.`......<.b:...FK.u}.d.%)*."g.hk.I.i's.....D.H
........H.........OJ.%.v.....U.EX.wL2t.....SK.........5.n...6.Y...S.|dj...Nqfv.....Qk...&.#|......
....#.A.I....g..p.*.>....kA..C...Z.....;S.gL.m..pj3.....#..V3.SP.....E.S...X.+...b.p..F     {
..8.C...]|.OO>......o#\.........j.2.0.n...j..>..-..A..t     ....9.a....o.q....t.............z[...
:d.bt:F..
.G.aW<...  D.7...}.KAA..........A.M...[..._.  a=....L.#."8.r....@....X..I.N.
.G^...jB5\.B}..w....p....jr.X.......x.......p.5......k.~.....9.[...D.h  .....  H?..Oe;....'-......h
)8.R.jB%..(..P.-.....>

w.7W..........8........(?{...5..KTvG.x.{ ;.p.....sU.pxX..9....(;.r..........j..o..7...  .}4.   .
6.MokB.X.o.Y5.Y../..rn.....O...5:..r(............:...q.=.9....j...dt.......}.4'...HLh....[.3KA.
.g.P.....
-.....^&.}1"..u.G..V.zM.%k..h.6.....H..........7[.O*..F..g..H..X..p..10.........z.o]..8...V..
.fus.S-...t9.]...hy.d.=cp.b-.i.(e.;zv...Q^v.9.'......1*wN.4
```

FTP VS VSFTP

- When comparing HTTP (Hypertext Transfer Protocol) over HTTPS(Hypertext Transfer Protocol Secure) , HTTP transmit data in plain text and can be intercepted and read by anyone who have access to the network. Even files that downloaded can be seen in plain text that can be easily downloaded by attacker.



```
No.  Time         Source           Destination      Protocol Length Info
  5 0.839768695  192.168.92.128   192.168.92.129   TCP       74 38766 → 80 [SYN] Seq=0 Win=32120 Len=0 MSS=1460 SACK_PERM TSval=3801670350 TSecr=0 WS
  6 0.840017675  192.168.92.129   192.168.92.128   TCP       74 80 → 38766 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM TSval=4186347441
  7 0.840047725  192.168.92.128   192.168.92.129   TCP       66 38766 → 80 [ACK] Seq=1 Ack=1 Win=32128 Len=0 TSval=3801670351 TSecr=4186347441
  8 0.840109904  192.168.92.128   192.168.92.129   HTTP     215 GET /nmapscan_results.txt HTTP/1.1
  9 0.840228615  192.168.92.129   192.168.92.128   TCP       66 80 → 38766 [ACK] Seq=1 Ack=150 Win=65024 Len=0 TSval=4186347442 TSecr=3801670351
 10 0.840514934  192.168.92.129   192.168.92.128   HTTP     678 HTTP/1.1 200 OK  (text/plain)
 11 0.840534454  192.168.92.128   192.168.92.129   TCP       66 38766 → 80 [ACK] Seq=150 Ack=613 Win=31872 Len=0 TSval=3801670351 TSecr=4186347442
 12 0.841090333  192.168.92.128   192.168.92.129   TCP       66 38766 → 80 [FIN, ACK] Seq=150 Ack=613 Win=31872 Len=0 TSval=3801670352 TSecr=41863474
 13 0.841336982  192.168.92.129   192.168.92.128   TCP       66 80 → 38766 [FIN, ACK] Seq=613 Ack=151 Win=65024 Len=0 TSval=4186347443 TSecr=38016703
 14 0.841347632  192.168.92.128   192.168.92.129   TCP       66 38766 → 80 [ACK] Seq=151 Ack=614 Win=31872 Len=0 TSval=3801670352 TSecr=4186347443
```

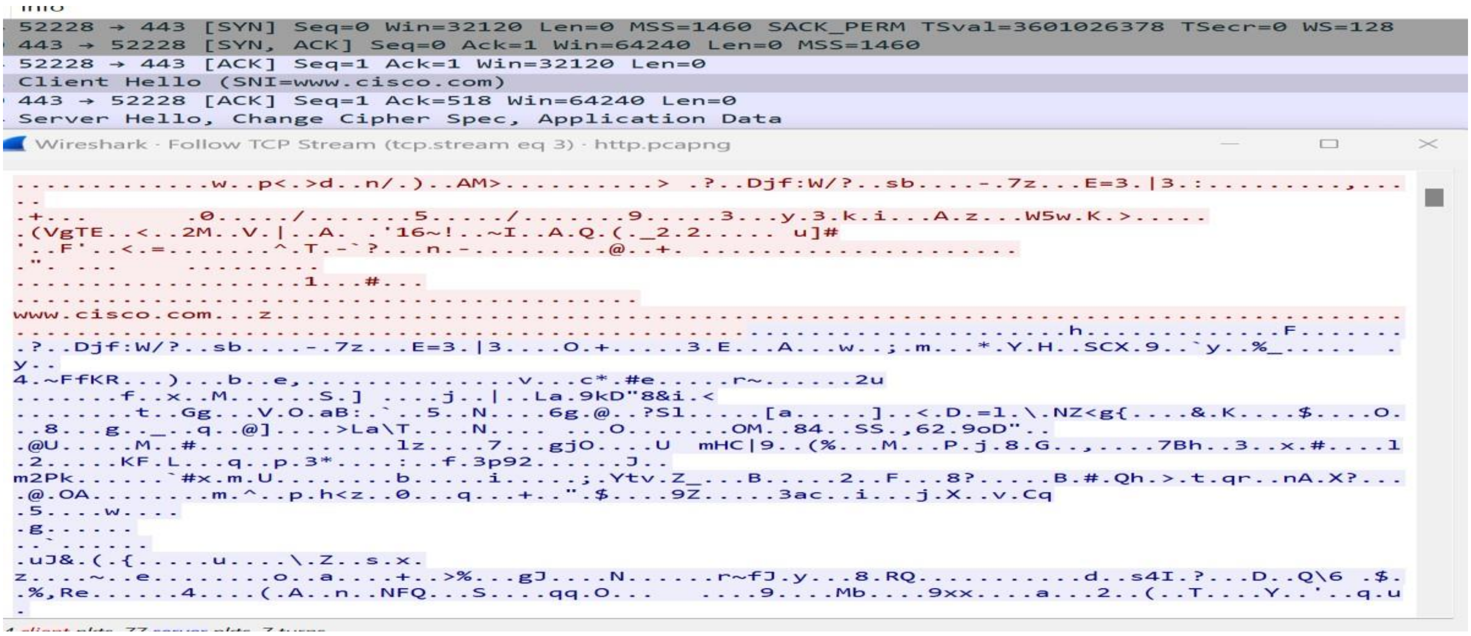
```
▸ Frame 10: 678 by
▸ Ethernet II, Src
▸ Internet Protoco
▸ Transmission Con
▸ Hypertext Transf
▸ Line-based text
```

Wireshark · Follow HTTP Stream (tcp.stream eq 0) · eth0

```
GET /nmapscan_results.txt HTTP/1.1
Host: 192.168.92.129
User-Agent: Wget/1.21.4
Accept: */*
Accept-Encoding: identity
Connection: Keep-Alive

HTTP/1.1 200 OK
Date: Tue, 28 May 2024 05:36:47 GMT
Server: Apache/2.4.52 (Ubuntu)
Last-Modified: Fri, 24 May 2024 05:45:12 GMT
ETag: "12f-6192cabd8e6c9"
Accept-Ranges: bytes
Content-Length: 303
Vary: Accept-Encoding
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/plain

Starting Nmap 7.80 ( https://nmap.org ) at 2024-05-24 05:45 UTC
Nmap scan report for server (192.168.92.129)
Host is up (0.000057s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE
21/tcp open  ftp
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 0.03 seconds
```

HTTP vs HTTPS

While over HTTPS, we only able to see the 3 way handshake (SSL/TLS) over port 443 and encryption parameters, key exchange and verification of the digital certificates are establish a secure connection



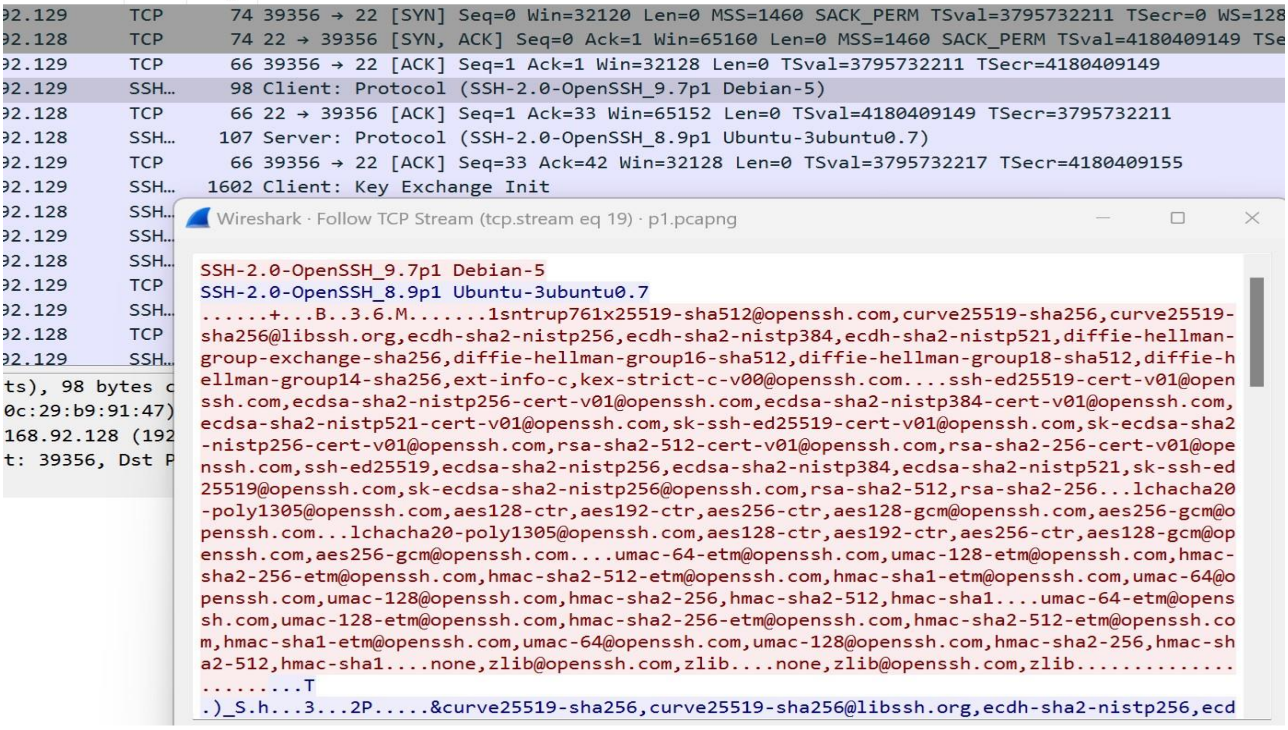
```
52228 → 443 [SYN] Seq=0 Win=32120 Len=0 MSS=1460 SACK_PERM TSval=3601026378 TSecr=0 WS=128
443 → 52228 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
52228 → 443 [ACK] Seq=1 Ack=1 Win=32120 Len=0
Client Hello (SNI=www.cisco.com)
443 → 52228 [ACK] Seq=1 Ack=518 Win=64240 Len=0
Server Hello, Change Cipher Spec, Application Data
```

Wireshark · Follow TCP Stream (tcp.stream eq 3) · http.pcapng

```
.............w..p<.>d..n/.)..AM>..........> .?..Djf:W/?..sb....-.7z...E=3.|3.:...........,...
..
.+...       .0...../.......5..../.......9....3...y.3.k.i...A.z...W5w.K.>.....
.(VgTE..<..2M..V.|..A.  .'16~!..~I..A.Q.(._2.2.....  u]#
'..F'..<.=.........^.T.-`?...n.-............@..+. ......................
.". ...    ..........
....................1...#...
..........................................
www.cisco.com...z.............................................................................
.................................................................h...............F.......
.?..Djf:W/?..sb....-.7z...E=3.|3....O.+.....3.E...A...w..;.m...*.Y.H..SCX.9..`y..%_.....  .
y..
4.~FfKR...)...b..e,................v...c*.#e.....r~......2u
........f..x..M......S.] ....j..|..La.9kD"8&i.<
..........t..Gg...V.O.aB:.`..5..N....6g.@..?S1.....[a.....]..<.D.=1.\.NZ<g{....&.K....$....O.
..8...g.._..q..@]....>La\T....N....  ...O........OM..84..SS.,62.9oD"..
.@U.....M..#................1z....7...gjO....U  mHC|9..(%...M...P.j.8.G..,....7Bh..3..x.#....1
.2.....KF.L...q..p.3*.....:..f.3p92......J..
m2Pk......`#x.m.U.........b.....i.....;.Ytv.Z_...B.....2..F...8?.....B.#.Qh.>.t.qr..nA.X?...
.@.OA.........m.^..p.h<z..0...q...+..".$....9Z.....3ac..i...j.X..v.Cq
.5....w....
.g......
..`.......
.uJ&.(.{.....u....\.Z..s.x.
z....~..e.........o..a....+..>%...gJ....N......r~fJ.y...8.RQ............d..s4I.?...D..Q\6  .$.
.%,Re......4....(.A..n..NFQ...S....qq.O...    ....9....Mb....9xx....a...2..(..T....Y..'..q.u
.
```

HTTP vs HTTPS

- SFTP (Secure File Transfer Protocol) is build on a Secure Shell (SSH) protocol thus providing a secure channel over encryptions and authentication such as public key and multifactor authentication which enhance security. It also included data integrity checks which ensure files not corrupted during transfer and support secure hashing algorithms.



```
92.129    TCP      74 39356 → 22 [SYN] Seq=0 Win=32120 Len=0 MSS=1460 SACK_PERM TSval=3795732211 TSecr=0 WS=128
92.128    TCP      74 22 → 39356 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM TSval=4180409149 TSe
92.129    TCP      66 39356 → 22 [ACK] Seq=1 Ack=1 Win=32128 Len=0 TSval=3795732211 TSecr=4180409149
92.129    SSH...   98 Client: Protocol (SSH-2.0-OpenSSH_9.7p1 Debian-5)
92.128    TCP      66 22 → 39356 [ACK] Seq=1 Ack=33 Win=65152 Len=0 TSval=4180409149 TSecr=3795732211
92.128    SSH...  107 Server: Protocol (SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.7)
92.129    TCP      66 39356 → 22 [ACK] Seq=33 Ack=42 Win=32128 Len=0 TSval=3795732217 TSecr=4180409155
92.129    SSH... 1602 Client: Key Exchange Init
92.128    SSH...
92.129    SSH...
92.128    SSH...
92.129    TCP
92.129    SSH...
92.128    TCP
92.129    SSH...
ts), 98 bytes c
0c:29:b9:91:47)
168.92.128 (192
t: 39356, Dst P
```

Wireshark · Follow TCP Stream (tcp.stream eq 19) · p1.pcapng

```
SSH-2.0-OpenSSH_9.7p1 Debian-5
SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.7
......+...B..3.6.M.......1sntrup761x25519-sha512@openssh.com,curve25519-sha256,curve25519-
sha256@libssh.org,ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-
group-exchange-sha256,diffie-hellman-group16-sha512,diffie-hellman-group18-sha512,diffie-h
ellman-group14-sha256,ext-info-c,kex-strict-c-v00@openssh.com....ssh-ed25519-cert-v01@open
ssh.com,ecdsa-sha2-nistp256-cert-v01@openssh.com,ecdsa-sha2-nistp384-cert-v01@openssh.com,
ecdsa-sha2-nistp521-cert-v01@openssh.com,sk-ssh-ed25519-cert-v01@openssh.com,sk-ecdsa-sha2
-nistp256-cert-v01@openssh.com,rsa-sha2-512-cert-v01@openssh.com,rsa-sha2-256-cert-v01@ope
nssh.com,ssh-ed25519,ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,ecdsa-sha2-nistp521,sk-ssh-ed
25519@openssh.com,sk-ecdsa-sha2-nistp256@openssh.com,rsa-sha2-512,rsa-sha2-256...lchacha20
-poly1305@openssh.com,aes128-ctr,aes192-ctr,aes256-ctr,aes128-gcm@openssh.com,aes256-gcm@o
penssh.com...lchacha20-poly1305@openssh.com,aes128-ctr,aes192-ctr,aes256-ctr,aes128-gcm@op
enssh.com,aes256-gcm@openssh.com....umac-64-etm@openssh.com,umac-128-etm@openssh.com,hmac-
sha2-256-etm@openssh.com,hmac-sha2-512-etm@openssh.com,hmac-sha1-etm@openssh.com,umac-64@o
penssh.com,umac-128@openssh.com,hmac-sha2-256,hmac-sha2-512,hmac-sha1....umac-64-etm@opens
sh.com,umac-128-etm@openssh.com,hmac-sha2-256-etm@openssh.com,hmac-sha2-512-etm@openssh.co
m,hmac-sha1-etm@openssh.com,umac-64@openssh.com,umac-128@openssh.com,hmac-sha2-256,hmac-sh
a2-512,hmac-sha1....none,zlib@openssh.com,zlib....none,zlib@openssh.com,zlib.............
.........T
.)_S.h...3...2P.....&curve25519-sha256,curve25519-sha256@libssh.org,ecdh-sha2-nistp256,ecd
```

SFTP (SECURE FILE TRANSFER PROTOC OL)

VSFTPD vs HTTPS vs SFTP

- VSFTPD
  - Advantage
    - Support SSL/TLS.
    - Handle large number connection.
    - Easy configurations.
    - Compatibility with numerous systems.
  - Disadvantage
    - Outdated protocol comparing to modern protocol.
    - Firewall problematic due to multiple ports used.
    - Complex configuration.

- HTTPS
  - Advantage
    - Support SSL/TLS.
    - Data integrity.
    - Simplicity.
    - Versatile.
  - Disadvantage
    - Limited to HTTP.
    - Performance overhead.
    - Limited Connection.

- VSFTP
  - Advantage
    - Robust security encrypting.
    - Resumable transfer..
    - Reliable operation.
    - Widely supported..
  - Disadvantage
    - Complexity.
    - SSH dependency.

VSFTPD

Confidentiality

- It supports strong encryptions and protecting it from interception by unauthorized attackers but if not properly configured, there is a risk of data leakage or exposure. Appropriate encryption settings needed to be implemented and configured to prevent unauthorized access to data.

Integrity

- It includes checksum verification and digital singatures which ensure file integrity which helps to detect any unauthorized modification or corruptions of data during transit. Due to the complexity of maintaining data integrity, if their server is compromised, attackers able to tamper the files during transfer thus compromising data integrity.

Availability

- Stability and efficiency and has the option for throttling bandwidth connections but not immune to vulnerabilities or attack such as DDOS.

HTTPS

Confidentiality

- Provide strong support for encryption using SSL/TLS encryption as it prevents from interception and reading sensitive information's such as login credentials, personal data and information but misconfigurations that could lead to data breach or unauthorized access. Poor certificate management process can also lead to compromised.

Integrity

- Data integrity by using Cryptographic algorithms ensure data not tampered with during transit. This prevents attackers from modifying the data without detection. However, outdated encryptions algorithms or vulnerabilities in SSL/TLS can be exploited. MITM {Man in the middle} attack can intercept and modify the data without detection.

Availability

- It can mitigate DDOS attack that aims to disrupt access or services by ensuring secure and reliable communication between host. SSL/TLS certificate expire may occur downtime, Security patches not applied correctly or promptly or if there server not properly configured to handle HTTPS traffic efficiently.

SFTP

Confidentiality

- Encrypt data in transit between host using SSH encryption as it protects sensitive information's such as login credentials, files or other data from unauthorized access. However, data can be compromised if the SSH key used for encryptions are weak or improperly managed.

Integrity

- Uses cryptographic hashes to verity integrity of files to ensure data has not be altered or corrupted during transfer. Vulnerabilities in SSH protocol can be exploited to manipulate or tamper the data without detection.

Availability

 Secured and reliable over SSH connections. SFTP can be susceptible to DDOS attack due to inadequate server maintenance or misconfigurations.

- When considering the protocol over VSFTPD, HTTPS and SFTP, SFTP emerges the best protocol due to its robust security, wide support across different operating system and simplicity over firewall configurations. While HTTPS and SFTP provides encryptions, HTTPS suited more for web communications while SFTP also handle larger file transfer well over HTTPS, as it suited more on file transfer.

Conclusion

- As Cyber security practitioners, our primary goal is to ensure security and integrity of the systems and data. Anonymity plays a crucial role especially when it comes to remote server access. By allowing users to connect to servers without revealing their true identities is a big security risk. It adds an extra layer of security by making it harder for malicious attackers to trace back the connection. This can be extremely important when accessing server with valuable data or systems that require safeguarded from unauthorized access.

  It's essential to balance out the proper authentication and access control, by implementing strong authentication system(multi-layer) that ensure only authorized individuals can access the server even when their identities are concealed/anonymous.

References

- Google.com
- Reddit
- https://www.linuxquestions.org/questions/linux-software-2/sftp-versus-vsftpd-590454/
- https://www.reddit.com/r/sysadmin/comments/m2upai/sftp_vs_https_for_file_transfers/
- https://www.jscape.com/blog/implementing-the-cia-triad-when-transferring-files-through-the-internet
- https://www.techtarget.com/whatis/definition/Confidentiality-integrity-and-availability-CIA