Know Abnormal, Find Evil

Beginner Guide For Security Professional

Maher Adib
maher@ofisgate.com

List of Pcaps

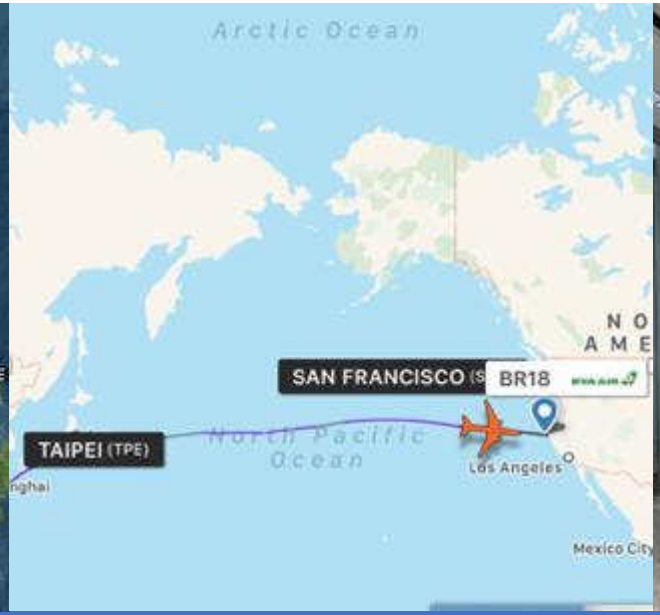
- https://www.cloudshark.org/captures/fce5f0ac3a50
- https://www.cloudshark.org/captures/d69e882c540f
- https://www.cloudshark.org/captures/77c0a7479e4e
- https://www.cloudshark.org/captures/cb3a30290372
- https://www.cloudshark.org/captures/a3882df6f4ee
- https://www.cloudshark.org/captures/263b0a95140b

8,467 mi

Distance from Kuala Lumpur Int'l Airport (KUL) to San Francisco International Airport (SFO)

18+ Hours Flight

Jetlag !@#$%^



Those Were The Days

WIRESHARK Get Acquainted Get Help Develop

What's on your network?

Ethereal-users: [Ethereal-users] monitor and analyze the users

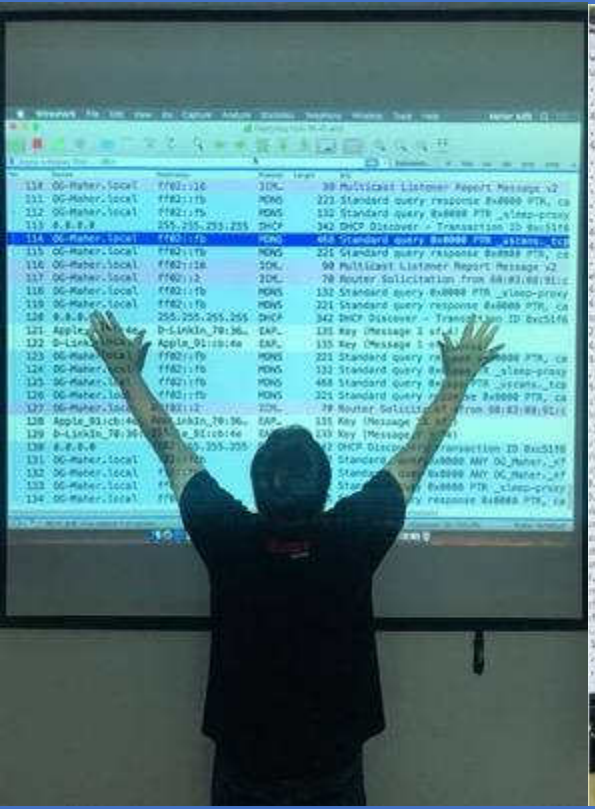
Note: This archive is from the project's previous web site, ethereal.com. This list is no longer active.

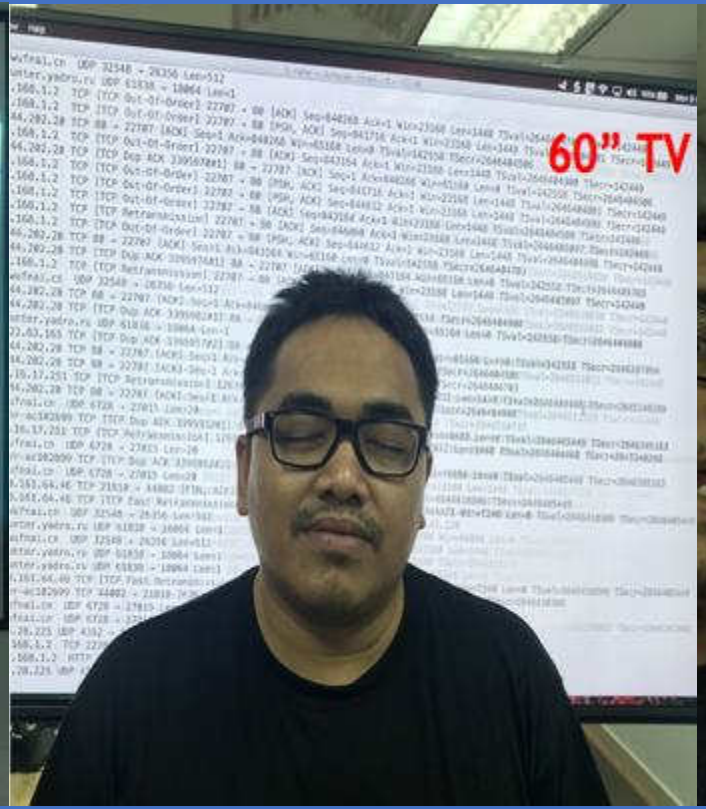
Date Index | Thread Index | Other Months | All Mailing Lists
Date Prev | Date Next | Thread Prev | Thread Next

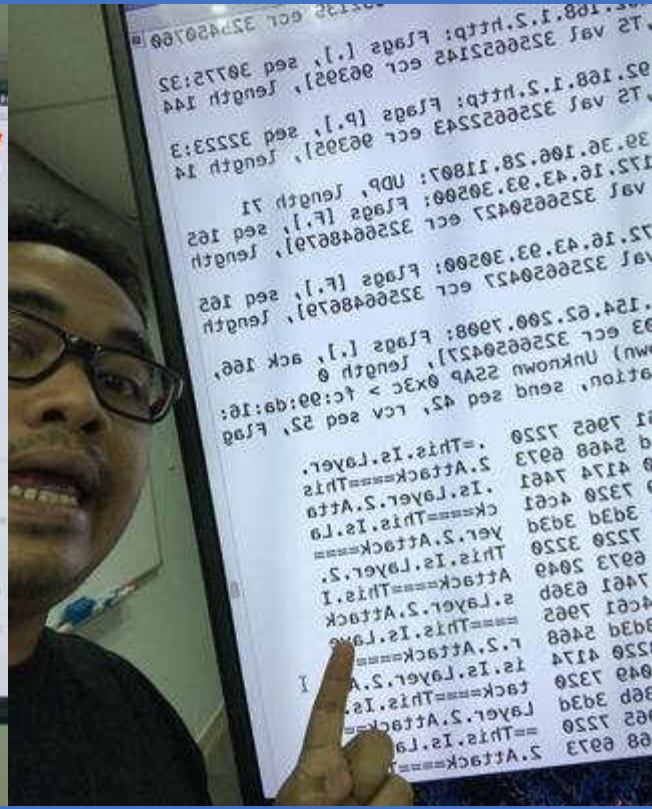
From: maher abedib <m2600@xxxxxxxxxxxx>
Date: Sun, 19 Nov 2000 07:21:36 +0800

```
Hi everyone,

I start using ethereal since Richard Sharpe give us a talk in LinuxWorld
Malaysia a few weeks ago.
When I fire up the ethereal ,wow ... I can see my users start to
logging/do some their stuff like ftp, telnet and etc.

>From there, I can monitor my users up to.But in order to monitor it, I
have to highlight and analyze some packet and use the option "follow tcp
stream" and then I can see every  keystroke/data that my users type to
my Linux server.

If possible,I would like to know, can ethereal continuously monitor the
users keystorke,for example,I targeted this user(maher) and see this
every single thing that he do.What do I know is the ethereal is a
network protocal analyzer.What is the differences between procotol
analyzer and keystroke monitoring( monitor users live some sort like
capturing the tty users).Can ethereal be functional like that?

Anyway,thank you Richard for highlight/bring up some ethereal
development in LinuxWorld Malaysia.

regards,

maher adib
```

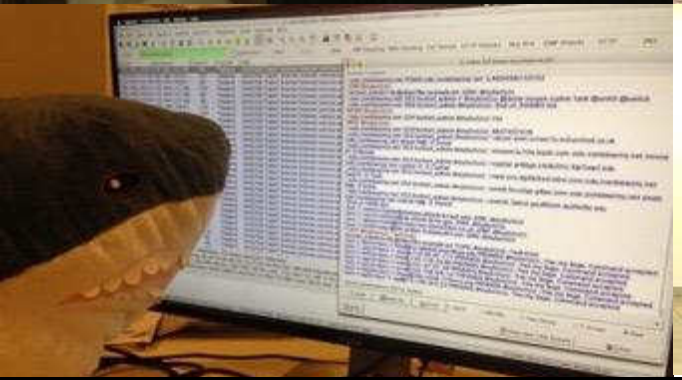
Wireshark'ing+Pcaps Everyday





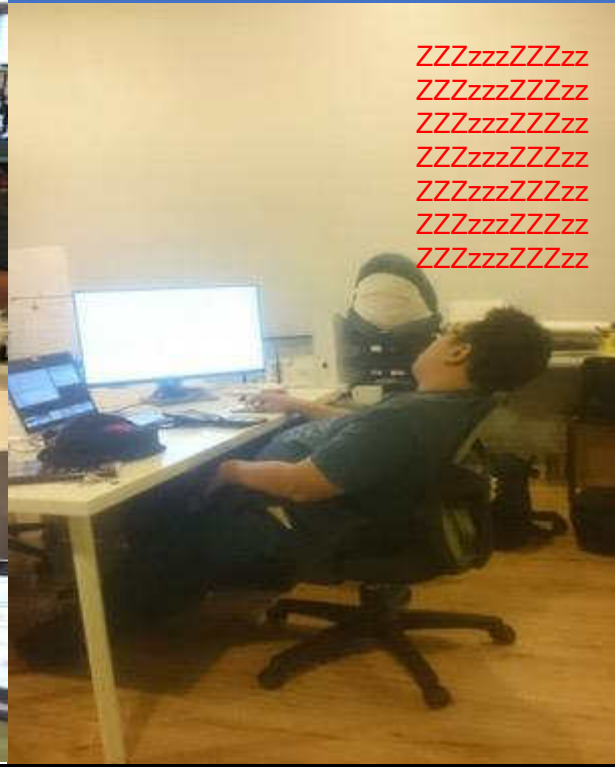


Not an easy job!



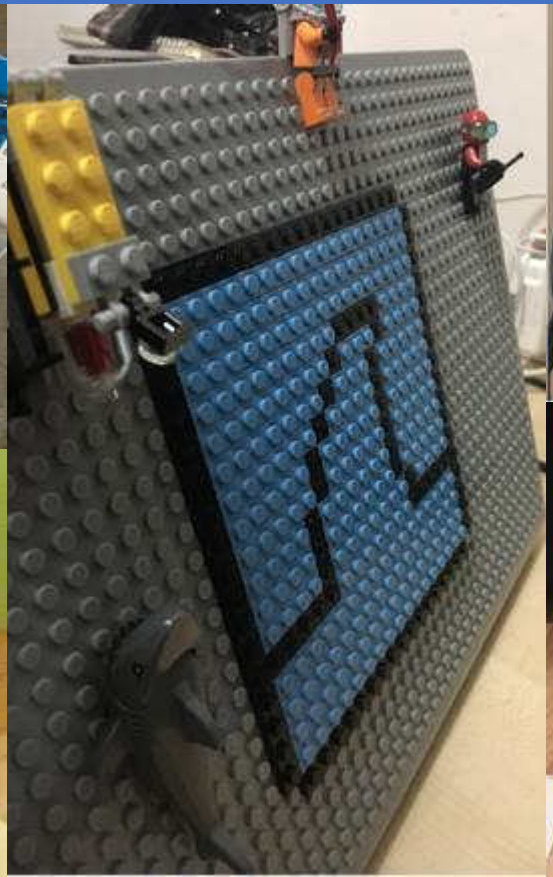
What My Company Think What I'm Doing

What My Customer see Everyday



This is what I Do Everyday!

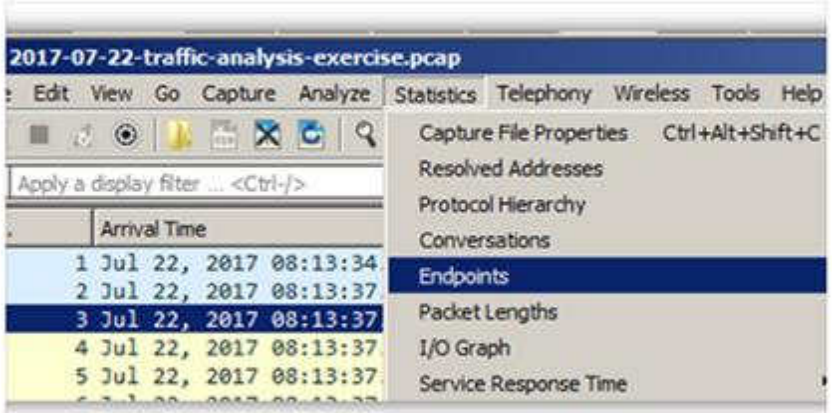
Fan of Shark Collector ,any thing!



Wireshark For Security Pro's!

InfoSec Institute
18 hrs ·

[FREE LAB] Threat Hunting: Zyklon Trojan http://ow.ly/FKdi30jZaNP
#infosec #Zyklon #Suricata #Snort



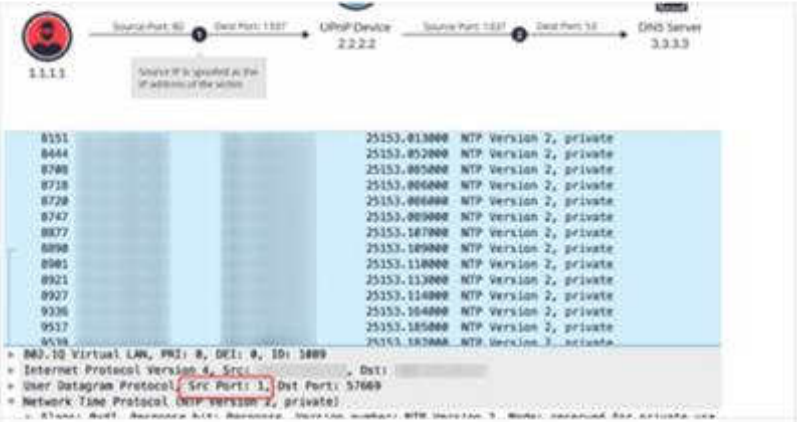
Threat Hunting – Zyklon Trojan

This is a lab that is conducted in a test bed. The resources were downloaded from malware.trafficanalysis.net. The samples provided came from a case study

RESOURCES.INFOSECINSTITUTE.COM

Security Training Share
29 mins ·

New Technology Uses UPnP Protocol to Avoid DDoS Mitigation
https://securityonline.info/new-technology-uses-upnp-proto.../



New Technology Uses UPnP Protocol to Avoid DDoS Mitigation • Penetration Testing

According to bleepingcomputer reports on the 15th, the United States well-known cybersecurity company Imperva issued a report on Monday that the...

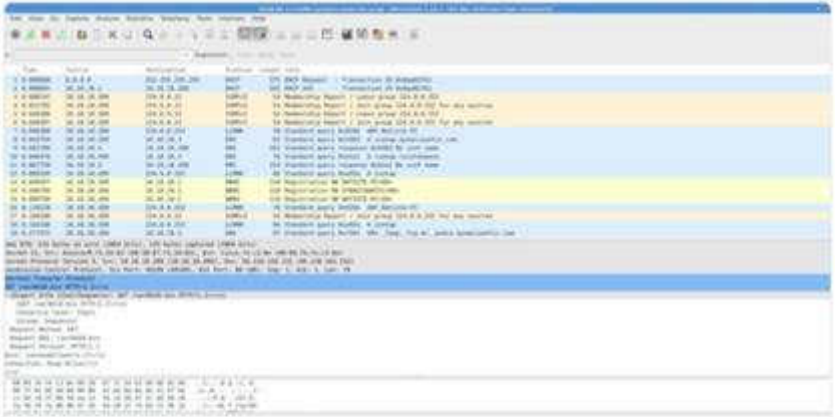
SECURITYONLINE.INFO

Wireshark For Security Pro's!

MikeR
@nahamike01

Following

I just published "Write-up of Malware Traffic Analysis Exercise: DYNACCOUNTIC"



Write-up of Malware Traffic Analysis Exercise: DYNACCOUNTIC

As always thanks to Brad at https://www.malware-traffic-analysis.net for the great exercises and constantly updating the exercise area of…

medium.com

Steve @stvemillertime · 8h

Yea so @FireEye FLAREs FakeNet-NG is my fav tool when it comes to simulating mal C2, esp for analyzing malware using legit services for C2. Here we see SSL comms to google docs, then we decrypt the stream to see the raw HTTP request #dailypcap #fakenetting github.com/fireeye/flare-…

Wireshark For Security Pro's!

Justin Warner
@sixdub

Follow

Is there a Python web server giving 401 and requesting NTLM auth... maybe from WPAD? Major lol if going to external host. content:"|53696d706c6548545450|";content:"|507974686f6e|";content:"|5757572d417574 68656e7469636174653a204e544c4d|";content:"401";http_stat_code; #dailypcap

8:21 PM - 23 Apr 2018

Nick Carr @ItsReallyNick · Apr 30

APT32 triage makes you ask broad #DFIR questions:

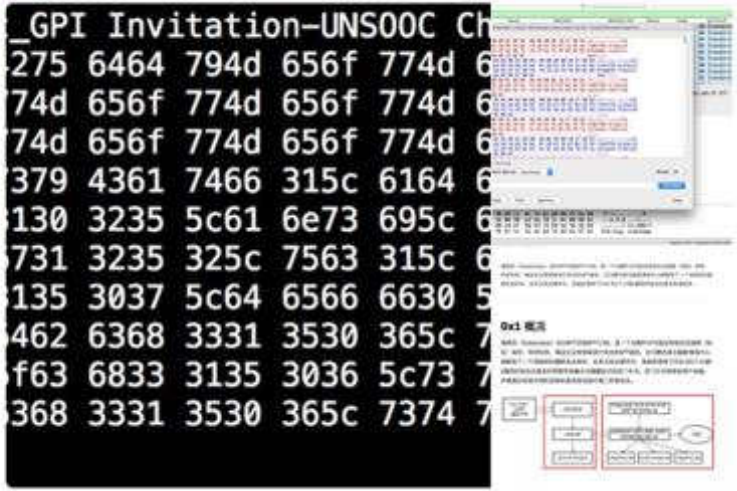
? How many meows[1] and PE files should be in RTFs?

? Should RTFs make DNS TXT queries & can they have + and / in them?

? Any good[2] Chinese language threat intel reports?

[1] twitter.com/ItsReallyNick/...

[2] s.tencent.com/research/repor...



FireEye

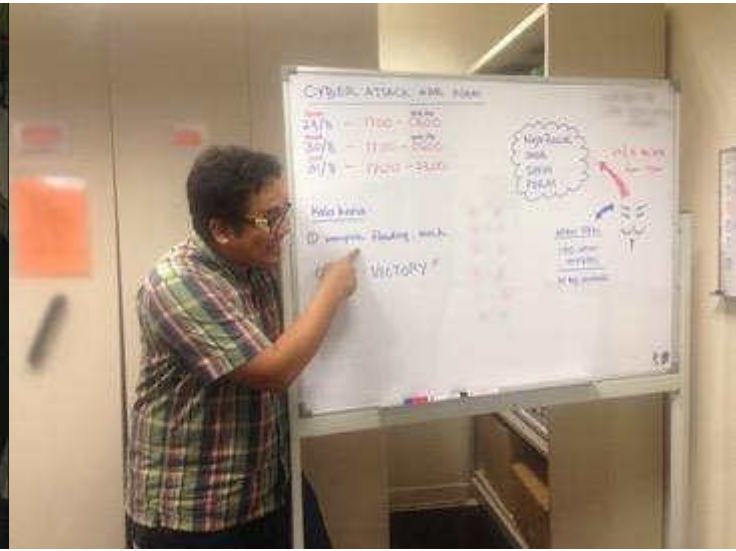
Products Services Solutions Partners

is codenamed "EternalBlue" and was leaked by ShadowBrokers. The exploited vulnerability, was patched in Microsoft MS17-010.

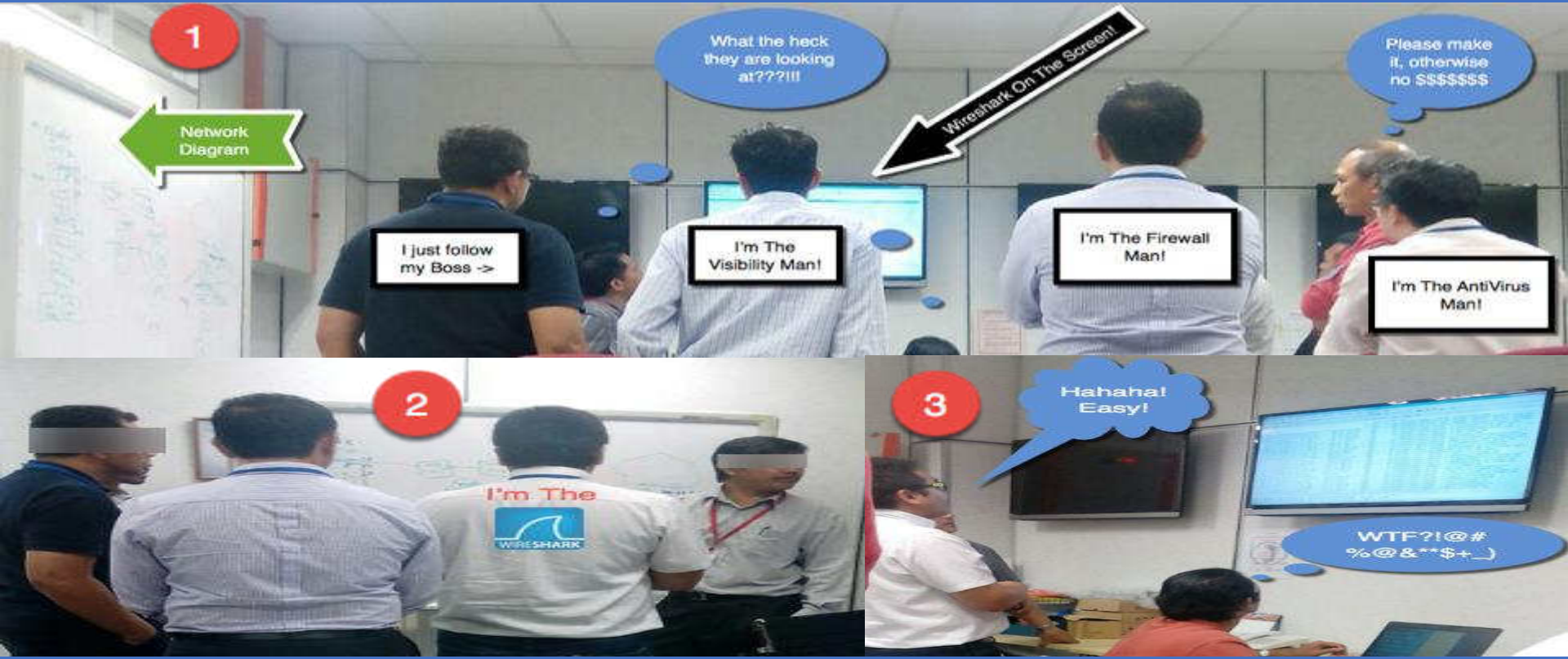
Based on our analysis, the malware spawns two threads. The first thread enumerates the network adapters and determines which subnets the system is on. The malware then generates a thread for each IP on the subnet. Each of these threads attempt to connect to the IP on TCP port 445 and, if successful, attempt exploitation of the system. An example of an attempt to exploit a remote system can be seen in Figure 1.

Protocol	Length	Info
TCP	62	1073 > 445 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1
TCP	62	445 > 1073 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 SACK_PERM=1
TCP	60	1073 > 445 [ACK] Seq=1 Ack=1 Win=64240 Len=0
TCP	60	1073 > 445 [FIN, ACK] Seq=1 Ack=1 Win=64240 Len=0
TCP	60	445 > 1073 [ACK] Seq=1 Ack=2 Win=64240 Len=0
TCP	60	445 > 1073 [RST, ACK] Seq=1 Ack=2 Win=0 Len=0
TCP	62	1074 > 445 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1
TCP	62	445 > 1074 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 SACK_PERM=1
TCP	60	1074 > 445 [ACK] Seq=1 Ack=1 Win=64240 Len=0
SMB	142	Negotiate Protocol Request
SMB	185	Negotiate Protocol Response
SMB	157	Session Setup AndX Request, User: .\
SMB	183	Session Setup AndX Response
SMB	127	Tree Connect AndX Request, Path: \\11.12.13.24\IPC$
SMB	93	Tree Connect AndX Response, Error: Non specific error code
SMB Pi	132	PeekNamedPipe Request, FID: 0x0000
SMB	93	Trans Response, Error: TID invalid
TCP	60	1074 > 445 [FIN, ACK] Seq=343 Ack=339 Win=63902 Len=0
TCP	60	445 > 1074 [ACK] Seq=339 Ack=344 Win=63986 Len=0
TCP	60	445 > 1074 [RST, ACK] Seq=339 Ack=344 Win=0 Len=0
TCP	62	1075 > 445 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1
TCP	62	445 > 1075 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 SACK_PERM=1
TCP	60	1075 > 445 [ACK] Seq=1 Ack=1 Win=64240 Len=0
SMB	191	Negotiate Protocol Request
SMB	187	Negotiate Protocol Response
SMB	194	Session Setup AndX Request, User: anonymous
SMB	267	Session Setup AndX Response
SMB	150	Tree Connect AndX Request, Path: \\192.168.56.20\IPC$
SMB	114	Tree Connect AndX Response

You have a friend! Wireshark!



You have a friend! Wireshark!



Objective

Wireshark is the de facto analysis tool across many fields. It's one of my go-to, ultimate security tools for verification and validation. When investigating possible security incidents, most of us start by firing up Wireshark and looking for packets relating to a breach or issue running inside the network/security infrastructure or devices. Sometimes it’s very hard to locate issues and we don’t know where to start.

In this hands-on lab, the presenter will share his concept of "Intercept, Listen, Discover, and Be Evil" with protocols by walking through real world exercises designed to help ascertain breach possibilities, spotting the difference between abnormal and normal traffic and demonstrating how to navigate and customize your Wireshark dashboard. This is suitable for those who want to start learning and using Wireshark from a security perspective.

The Concept!

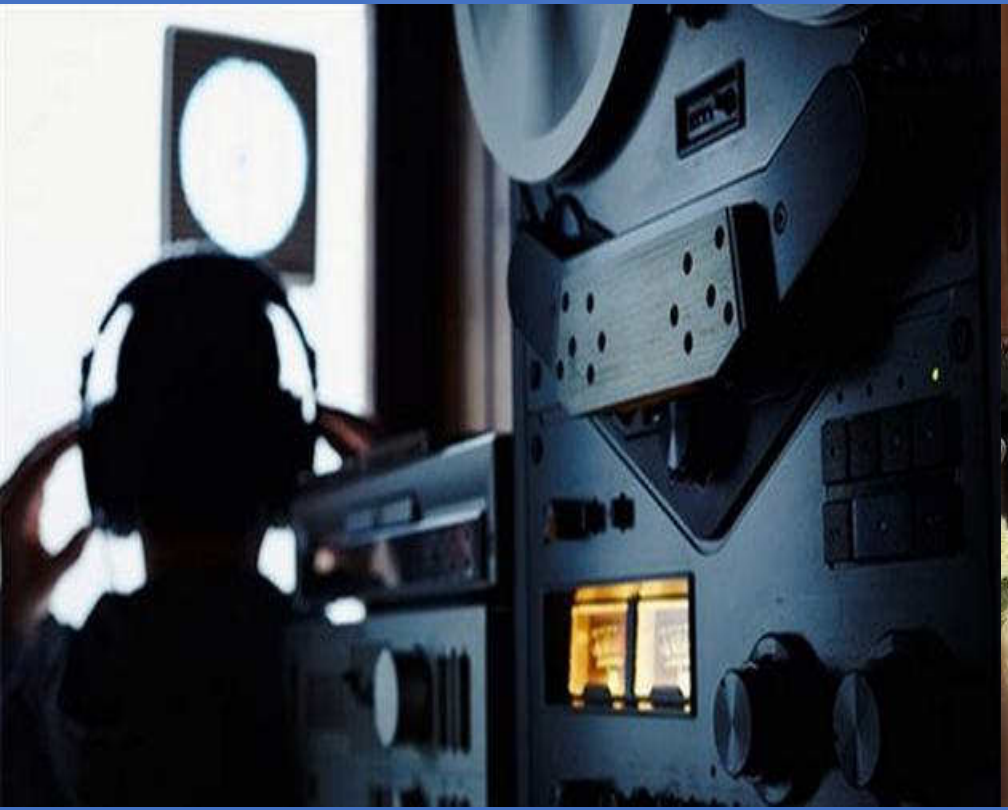
Intercept

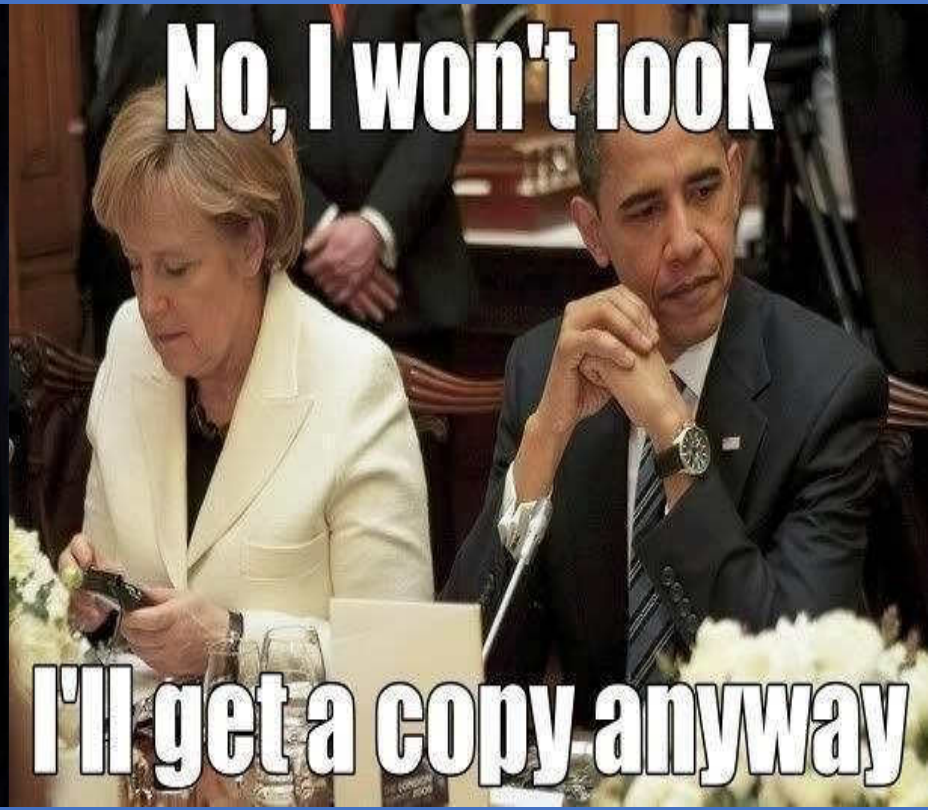
Listen

Discover

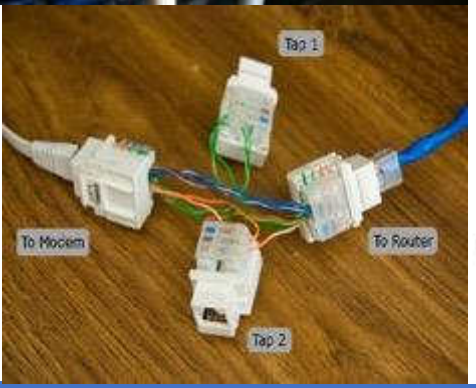
Be Evil!

Intercept The Communication

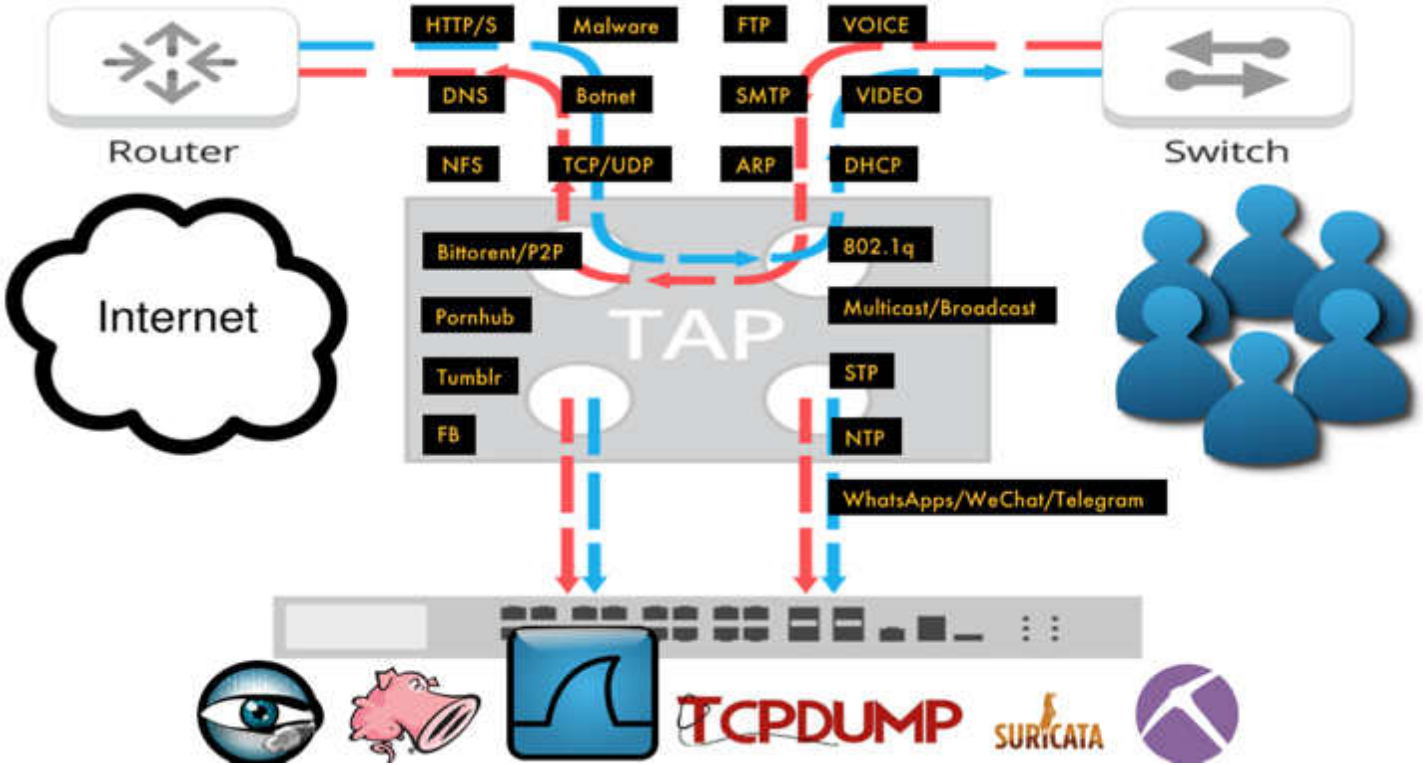




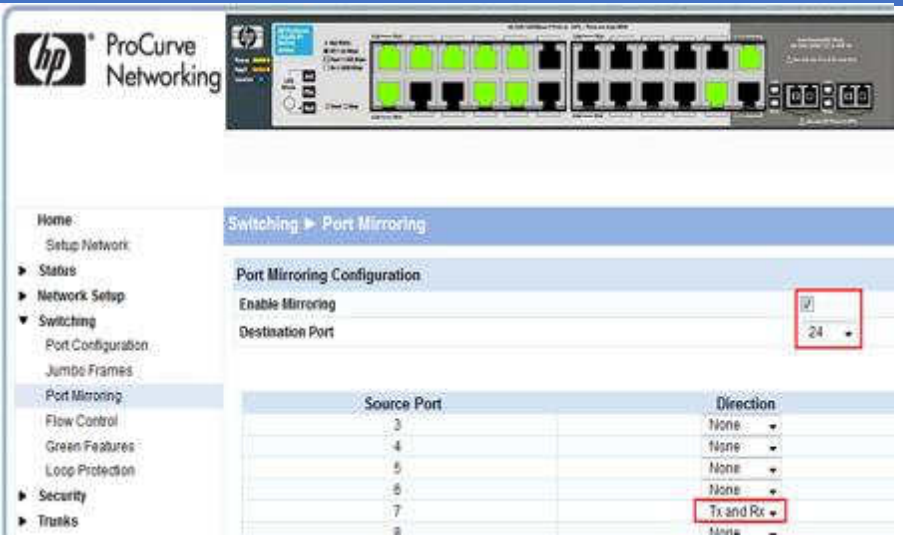
Intercept The Communication: The Tools

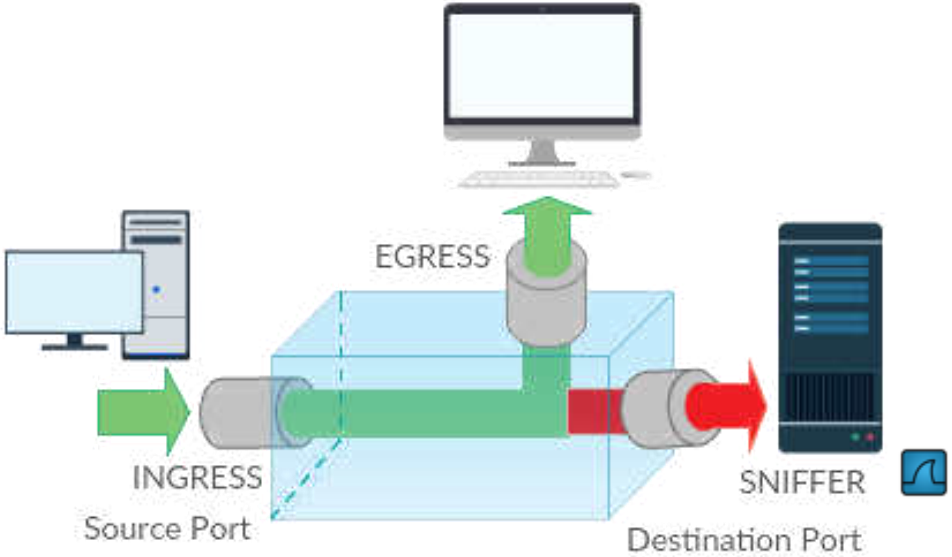


Intercept: Use Taps



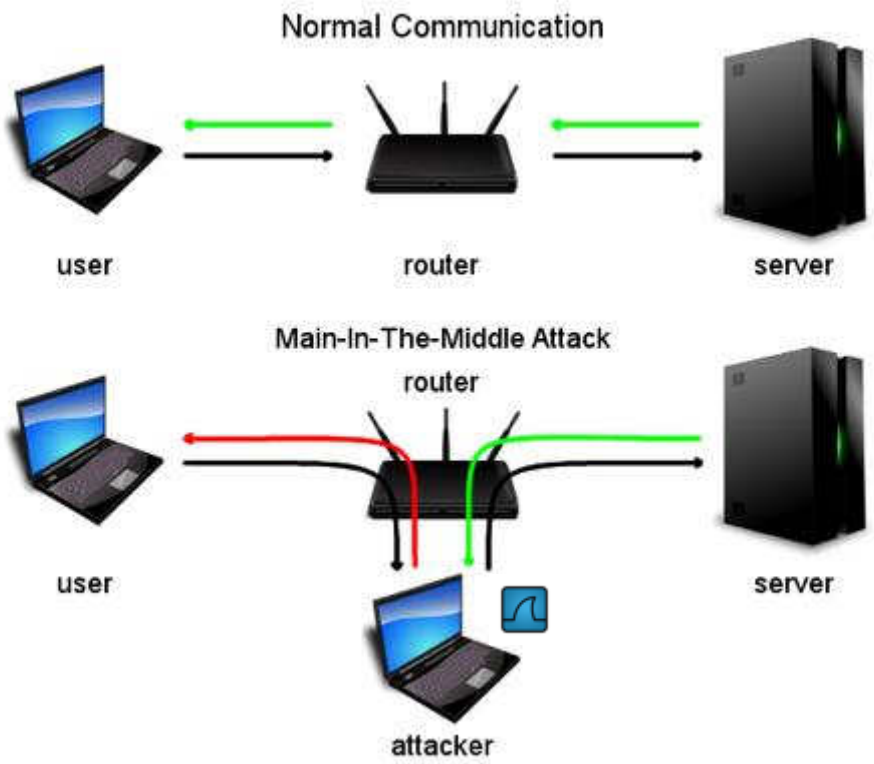
Intercept: SPAN/Mirroring

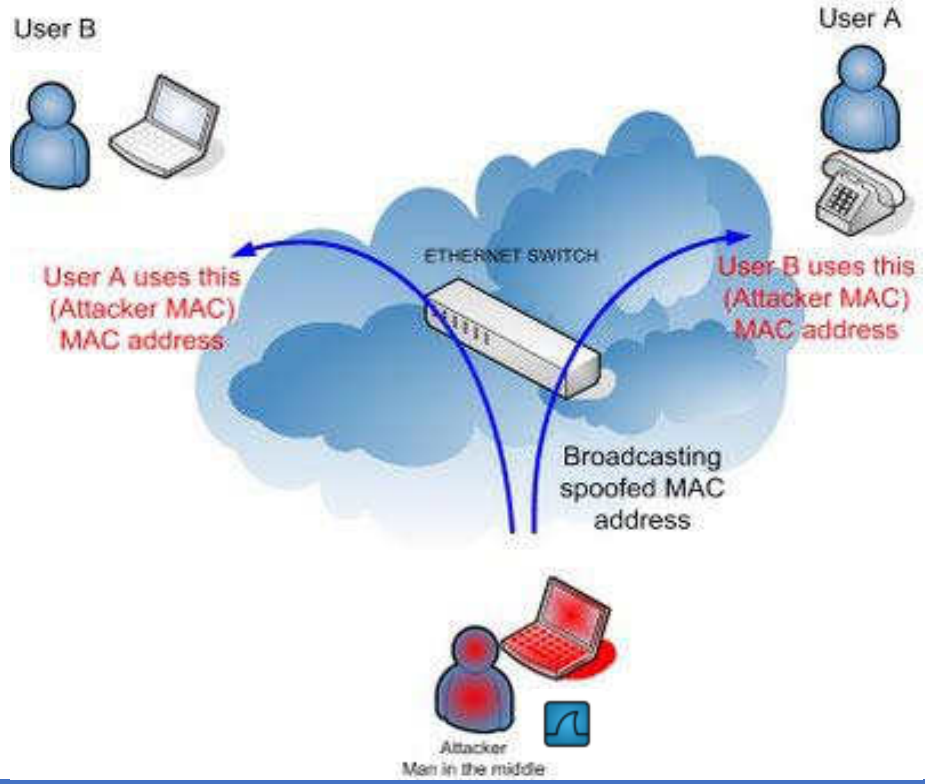




Switch(config)#monitor session 1 source interface gigabitEthernet 1/7 both
Switch(config)#monitor session 1 destination interface gigabitEthernet 1/24

Intercept: Man in The Middle (MiTM)





What Is Your Goal?

Peter Wu
@Lekensteyn

This happens way too often:
"help, need to learn wireshark"
"What is your goal?"
"hacking web password like gmail facebook"

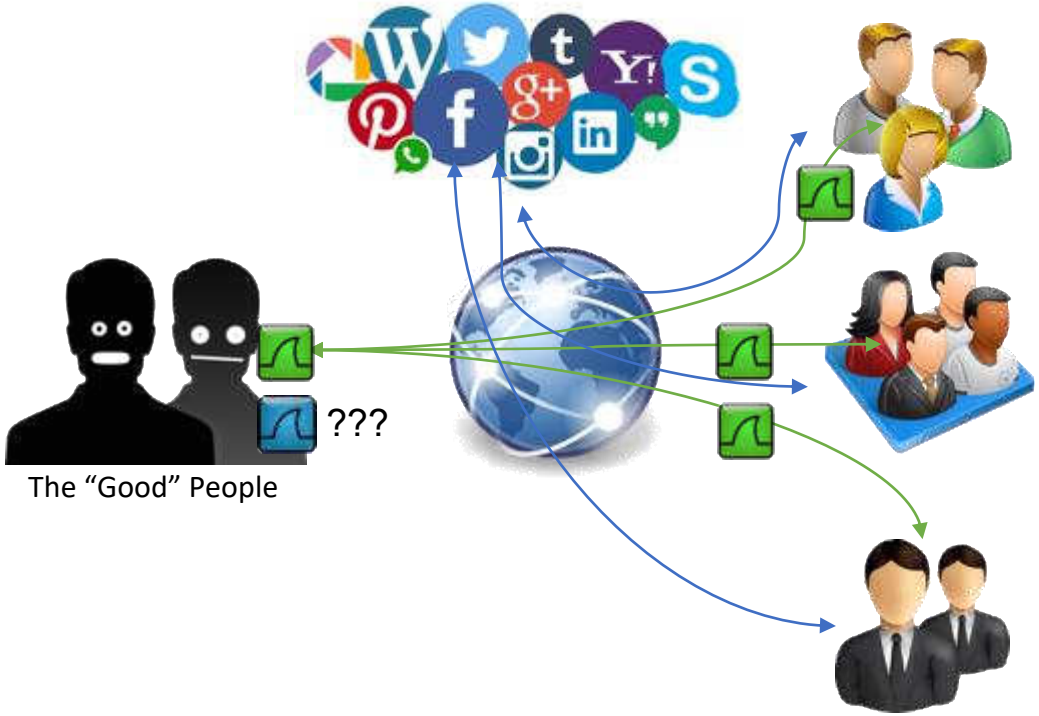
...

What Is Your Goal?

The real big question

Now, the real big question is often something like this: "how can I capture packets from someone on **another network than my own**?". So let's say you're at home, running Wireshark on your computer, and you want to capture packets of a friend sitting in his own home, at his own computer. This is like trying to read a postcard she or he is sending via snail mail – you need to be at the right spot at the right time to see it pass you by, or you can't read it obviously. It's the same with network packets on a remote network – you need to get **physical access** to it, or you can't capture any of it's packets. So if you want to capture packets with someone's Facebook password in it, you need to either be

- physically connected to their network (good luck with that)
- physically connected to the Facebook network (good luck with that)
- physically connected to any network in between those two (good luck with… you get the drift).



https://blog.packet-foo.com/2016/07/how-to-use-wireshark-to-steal-passwords/#more-1244

Don’t Just Look at Wireshark??!!!

Thunderbolt Ethernet: en14 (not ether host 68:5b:35:cb:39:4b)

Apply a display filter ... <⌘/> Expression... +

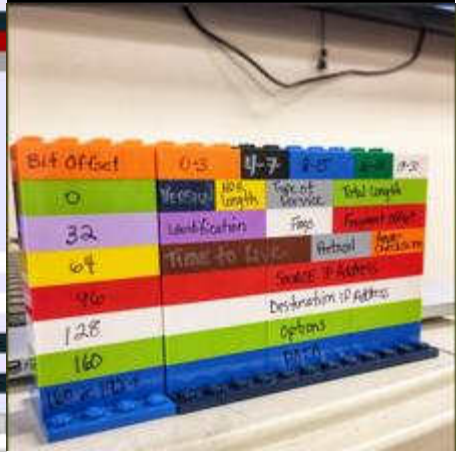
No.	Time	Source	Destination	Protocol	Length	Info
88804	0.160334	172.20.215.253	224.0.0.2	HSRP	62	Hello (state Standby)
88805	0.011761	172.20.212.179	172.20.2…	NBNS	92	Name query NB PMBIPRCIM03<00>
88806	0.029309	172.20.212.176	172.20.2…	NBNS	92	Name query NB WPAD<00>
88807	0.031753	172.20.214.226	255.255.…	UDP	67	49541 → 9273 Len=25
88808	0.085212	423-qbusjdl15.l…	Broadcast	ARP	60	Who has 172.20.215.213? Tell 172.20.214.189
88809	0.047256	NPI27DB87.local	Broadcast	ARP	60	Who has 172.20.215.254? Tell 172.20.214.4
88810	0.031008	172.20.212.179	224.0.0.…	LLM…	71	Standard query 0xe843 AAAA PMBIPRCIM03
88811	0.000002	172.20.212.179	224.0.0.…	LLM…	71	Standard query 0xd0a5 A PMBIPRCIM03
88812	0.228457	fe80::401c:47d7…	ff02::1:3	LLM…	94	Standard query 0x1bd7 A zeocybskgsipox
88813	0.000158	172.20.212.176	224.0.0.…	LLM…	74	Standard query 0x1bd7 A zeocybskgsipox
88814	0.003079	fe80::401c:47d7…	ff02::1:3	LLM…	95	Standard query 0x25b8 A tqfydkveyepackl
88815	0.000130	172.20.212.176	224.0.0.…	LLM…	75	Standard query 0x25b8 A tqfydkveyepackl
88816	0.004524	fe80::401c:47d7…	ff02::1:3	LLM…	87	Standard query 0x883c A borgghn
88817	0.000009	172.20.212.176	224.0.0.…	LLM…	67	Standard query 0x883c A borgghn

Listen To Conversation

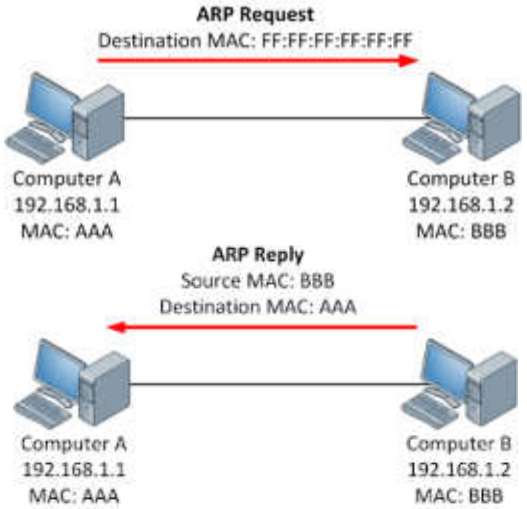
No.	Source	Destination	Protocol	Length	Info
144226	172.20.212.176	224.0.0.252	LLMNR	64	Standard query 0x1ee9 A wpad
144227	172.20.212.176	172.20.215…	NBNS	92	Name query NB WPAD<00>
144228	Cisco_db:ef:2a	Spanning-t…	STP	60	Conf. Root = 0/0/00:0c:cf:2e:dd:c2 Cost = 6008 Port = 0x802a
144229	172.20.215.252	224.0.0.5	OSPF	98	Hello Packet
144230	fe80::401c:47d7:8a…	ff02::1:3	LLMNR	84	Standard query 0x1ee9 A wpad
144231	172.20.212.176	224.0.0.252	LLMNR	64	Standard query 0x1ee9 A wpad
144232	172.20.212.176	172.20.215…	NBNS	92	Name query NB WPAD<00>
144233	Cisco_bc:fd:9c	Broadcast	ARP	60	Who has 172.20.215.230? Tell 172.20.215.252
144234	155.69.5.151	172.20.212…	TCP	60	[TCP Keep-Alive] 135 → 62813 [ACK] Seq=1 Ack=1 Win=256 Len=1
144235	Cisco_bc:fd:9c	Broadcast	ARP	60	Who has 172.20.214.176? Tell 172.20.215.252
144236	172.20.215.252	224.0.0.2	HSRP	62	Hello (state Active)
144237	172.20.214.226	255.255.25…	UDP	67	49541 → 9273 Len=25
144238	Cisco_db:ef:2a	CDP/VTP/DT…	CDP	398	Device ID: NEC-05-E04_STD2.ntu.edu.sg Port ID: FastEthernet0/42
144239	172.20.212.176	172.20.215…	NBNS	92	Name query NB WPAD<00>
144240	Cisco_bc:fd:9c	Broadcast	ARP	60	Who has 172.20.215.214? Tell 172.20.215.252
144241	Cisco_bc:fd:9c	Broadcast	ARP	60	Who has 172.20.212.5? Tell 172.20.215.252
144242	155.69.5.151	172.20.212…	TCP	60	[TCP Keep-Alive] 135 → 62813 [ACK] Seq=1 Ack=1 Win=256 Len=1
144243	172.20.214.226	255.255.25…	UDP	67	49541 → 9273 Len=25
144244	Cisco_db:ef:2a	Spanning-t…	STP	60	Conf. Root = 0/0/00:0c:cf:2e:dd:c2 Cost = 6008 Port = 0x802a
144245	155.69.5.177	172.20.214…	TCP	60	135 → 51130 [ACK] Seq=1 Ack=1 Win=256 Len=1
144246	Cisco_bc:fd:9c	Broadcast	ARP	60	Who has 172.20.212.224? Tell 172.20.215.252
144247	Cisco_bc:fd:9c	Broadcast	ARP	60	Who has 172.20.215.211? Tell 172.20.215.252
144248	155.69.5.151	172.20.212…	TCP	60	[TCP Keep-Alive] 135 → 62813 [ACK] Seq=1 Ack=1 Win=256 Len=1
144249	172.20.214.226	255.255.25…	UDP	67	49541 → 9273 Len=25

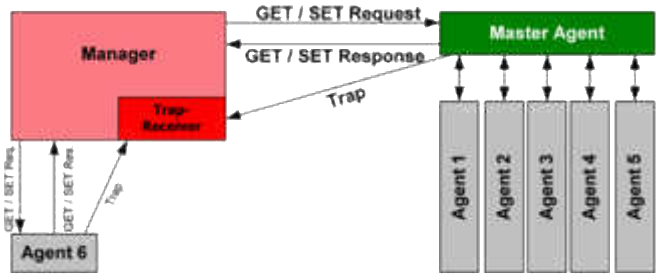
Discover: I know this! What???

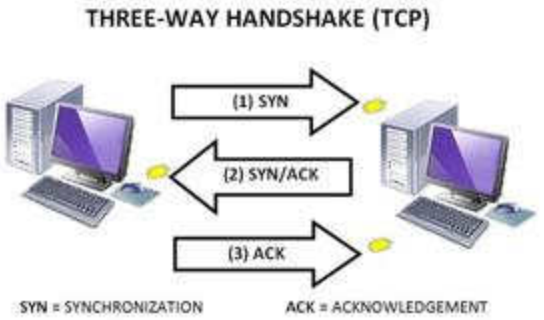
No.	Time	Source	Destination	Protocol	Info
50…	0.000s	58.193.0.208	atsweb.arvixe…	TCP	[TCP Previous segment not captured] 33998→384 [PSH, ACK] Seq=1447 Ack=1 Win=94896128 Len=582 TSval=0 TSecr=…
50…	0.000s	58.193.0.208	atsweb.arvixe…	TCP	[TCP Previous segment not captured] 33998→384 [PSH, ACK] Seq=2041 Ack=1 Win=94896128 Len=274 TSval=0 TSecr=…
50…	0.000s	atsweb.arvixecloud…	58.193.0.208	TCP	[TCP ACKed unseen segment] 384→33998 [ACK] Seq=1 Ack=8101 Win=237240320 Len=0 TSval=0 TSecr=3484048935
50…	0.000s	58.193.0.208	atsweb.arvixe…	TCP	[TCP Previous segment not captured] 33998→384 [PSH, ACK] Seq=2857 Ack=1 Win=94896128 Len=675
50…	- 0.0…	58.193.0.208	atsweb.arvixe…	IPv4	Host Monitoring (20)
50…	0.000s	58.193.0.208	atsweb.arvixe…	TCP	[TCP Previous segment not captured] 33998→384 [PSH, ACK] Seq=4093 Ack=1 Win=94896128 Len=1224 TSval=0 TSecr…
50…	0.000s	58.193.0.208	atsweb.arvixe…	TCP	33998→384 [PSH, ACK] Seq=5317 Ack=1 Win=94896128 Len=648 TSval=0 TSecr=3484049091
50…	0.000s	atsweb.arvixecloud…	58.193.0.208	TCP	[TCP ACKed unseen segment] 384→33998 [ACK] Seq=1 Ack=8101 Win=237240320 Len=4
50…	- 0.0…	58.193.0.208	atsweb.arvixe…	TCP	33998→384 [PSH, ACK] Seq=5965 Ack=1 Win=94896128 Len=263 TSval=0 TSecr=3484049091
50…	0.000s	58.193.0.208	atsweb.arvixe…	TCP	[TCP Previous segment not captured] 33998→384 [ACK] Seq=6252 Ack=1 Win=94896128 Len=572 TSval=0 TSecr=3484…
50…	0.000s	atsweb.arvixecloud…	58.193.0.208	IPv4	Fragmented IP protocol (proto=TCP 6, off=24928, ID=0eef)
50…	0.000s	atsweb.arvixecloud…	58.193.0.208	TCP	[TCP ACKed unseen segment] 384→33998 [ACK, URG] Seq=1 Ack=8101 Win=237240320 Urg=46609 Len=0 TSval=0 TSecr=…
50…	0.000s	58.193.0.208	atsweb.arvixe…	TCP	[TCP Port numbers reused] 33998→384 [SYN, RST, ACK, CWR] Seq=8101 Ack=5 Win=7240 Len=862 TSval=0 TSecr=3484…
50…	0.000s	58.193.0.208	atsweb.arvixe…	IPv4	Unassigned (162)
50…	0.000s	58.193.0.208	atsweb.arvixe…	TCP	[TCP Previous segment not captured] 33998→384 [PSH, ACK] Seq=10629 Ack=5 Win=118620160 Len=780 TSval=0 TSec…
50…	0.000s	58.193.0.208	atsweb.arvixe…	TCP	33998→384 [PSH, ACK] Seq=11409 Ack=5 Win=118620160 Len=656 TSval=0 TSecr=3484049091
50…	0.000s	58.193.0.208	atsweb.arvixe…	TCP	33998→384 [PSH, ACK] Seq=12065 Ack=5 Win=118620160 Len=316 TSval=0 TSecr=3484049091
50…	0.000s	atsweb.arvixecloud…	58.193.0.208	TCP	[TCP ACKed unseen segment] 384→33998 [RST, ACK] Seq=5 Ack=11409 Win=0 Len=0
50…	0.000s	atsweb.arvixecloud…	58.193.0.208	TCP	384→33998 [RST, ACK] Seq=5 Ack=12065 Win=0 Len=0
50…	0.000s	192.168.227.35	5.233.154.122	TCP	420→56056 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1460 TSval=3490051389 TSecr=3490044018
50…	0.000s	5.234.157.84	192.168.185.1…	TCP	37310→371 [ACK] Seq=1 Ack=1 Win=94896128 Len=0 TSval=3490051803 TSecr=3490043398
50…	0.000s	5.233.154.122	192.168.227.35	TCP	56056→420 [ACK] Seq=1 Ack=1 Win=94896128 Len=0 TSval=3490051981 TSecr=3490051389
51…	0.000s	175.59.132.32	192.169.38.39	TCP	39871→128 [PSH, ACK] Seq=1 Ack=1 Win=94896128 Len=572 TSval=0 TSecr=3488057202
51…	0.000s	175.59.132.32	192.169.38.39	TCP	39871→128 [PSH, URG, CWR] Seq=573 Win=94896128 Urg=0 Len=582 TSval=0 TSecr=3488057202
51…	0.000s	175.59.132.32	192.169.38.39	TCP	39871→128 [PSH, ACK, URG] Seq=1155 Ack=1 Win=94896128 Urg=40650 Len=292 TSval=0 TSecr=3488057202
51…	0.000s	175.59.132.32	192.169.38.39	TCP	39871→128 [PSH, ACK] Seq=1447 Ack=1 Win=94896128 Len=594 TSval=0 TSecr=3488057202
51…	0.000s	175.59.132.32	192.169.38.39	TCP	39871→128 [PSH, ACK] Seq=2041 Ack=1 Win=94896128 Len=274
51…	0.000s	175.59.132.32	192.169.38.39	TCP	39871→128 [PSH, ACK] Seq=2315 Ack=1 Win=94896128 Len=542 TSval=3490052268 TSecr=3488057202
51…	0.000s	175.59.132.32	192.169.38.39	TCP	39871→128 [PSH, ACK] Seq=2857 Ack=1 Win=94896128 Len=675
51…	0.000s	192.169.38.39	175.59.132.32	IPv4	Fragmented IP protocol (proto=TCP 6, off=43256, ID=0911)
51…	0.000s	175.59.132.32	192.169.38.39	TCP	39871→128 [PSH, ACK] Seq=3532 Ack=1 Win=94896128 Len=561 TSval=0 TSecr=3488057202
51…	0.000s	192.169.38.39	175.59.132.32	IPv4	Packet radio (21)
51…	0.000s	192.169.38.39	175.59.132.32	TCP	128→39871 [ACK, URG] Seq=1 Ack=4093 Win=189792256 Urg=64441 Len=0 TSval=0 TSecr=3488057060
51…	0.000s	192.169.38.39	175.59.132.32	TCP	[TCP Port numbers reused] 128→39871 [SYN, RST, PSH, ACK, URG, ECN] Seq=1 Ack=4093 Win=11584 Urg=0 Len=0 TSv…
51…	- 0.0…	175.59.132.32	192.169.38.39	TCP	39871→128 [PSH, ACK] Seq=4093 Ack=1 Win=94896128 Len=1228 TSval=0 TSecr=3488057202
51…	0.000s	175.59.132.32	192.169.38.39	TCP	[TCP Previous segment not captured] 39871→128 [PSH, ACK] Seq=5965 Ack=1 Win=94896128 Len=287 TSval=0 TSecr=…
51…	0.000s	175.59.132.32	192.169.38.39	TCP	39871→128 [PSH, ACK] Seq=6252 Ack=1 Win=94896128 Len=560 TSval=0 TSecr=3488057202
5…	0.000s	175.59.132.32	192.169.38.39	TCP	39871→128 [RST, ACK, CWR] Seq=6824 Ack=1 Win=94896128 Len=1277 TSval=0 TSecr=3488057202
51…	0.000s	175.59.132.32	192.169.38.39	IPv4	DCN Measurement (19)

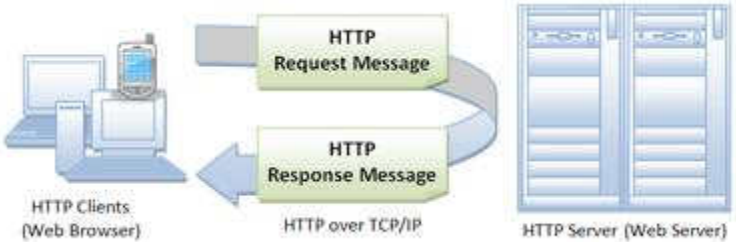


Know Abnormal, Find Evil









Profile and Short-Cut Button!

NETWORK | APPLICATION | SECURITY | TROUBLESHOOTING -> Role Based

MyProfile | MyWiFe | MyDad | MyMom | MyBOSS | IHateThisGuy | My-eX -> Relation Based

TCP | UDP | ARP | DHCP | OSPF | HTTP | DNS -> Protocol Based

So Many TCP? | UDP Flood? | Who's IP is this | DHCP rogue | OSPF Authentication | Clear Text HTTP with Password | DNS Weird | I Don't use this Apps | Brute Force Password? | Any weird things pass to Security team!

The Power Of The Right Click!

No.	Source	Destination	Protocol	Length	Info
25	172.20.214.226	255.255.255.2…	UDP	67	49541 → 9273 Len=25
26	Toshiba_88:c2:76	Broadcast	ARP	60	Who has 172.20.215.230? Tell 172.20.213.220
27	172.20.215.252	224.0.0.2	HSRP	62	Hello (state Active)
28	Cisco_bc:fd:9c	Broadcast	ARP	60	Who has 172.20.214.176? Tell 172.20.215.252
29	Cisco_db:ef:2a	Spanning-tree…	STP	60	Conf. Root = 0/0/00:0c:cf:2e:dd:c2 Cost = 6008 Port = 0x002a
30	hbsu-PC.local	ff02::c	SSDP	208	M-SEARCH * HTTP/1.1
31	172.20.214.226	255.255.255.2…	UDP	67	49541 → 9273 Len=25
32	Toshiba_88:c2:76	Broadcast	ARP	60	Who has 172.20.215.230? Tell 172.20.213.220
33	172.20.215.253	224.0.0.2	HSRP	62	Hello (state Standby)
34	172.20.214.226	255.255.255.2…	UDP	67	49541 → 9273 Len=25

```
▶ Frame 33: 62 bytes on wire (496 bits), 62 bytes capt
▶ Ethernet II, Src: Cisco_42:dd:7c (00:0c:cf:42:dd:7c)                    :00:02)
▶ Internet Protocol Version 4, Src: 172.20.215.253 (17                    ecast.net (224.0.0.2)
▶ User Datagram Protocol, Src Port: 1985, Dst Port: 19
▼ Cisco Hot Standby Router Protocol
    Version: 0
    Op Code: Hello (0)
    State: Standby (8)
    Hellotime: Non-Default (5)
    Holdtime: Non-Default (15)
    Priority: 100
    Group: 3
    Reserved: 0
    Authentication Data: Default (cisco)
    Virtual IP Address: 172.20.215.254 (172.20.215.254
```

- Expand Subtrees ⇧→
- Expand All ⌘→
- Collapse All ⌘←
- Apply as Column
- Apply as Filter ▶
- Prepare a Filter ▶
- Conversation Filter ▶
- Colorize with Filter ▶
- Follow ▶
- Copy ▶
- Show Packet Bytes…
- Export Packet Bytes… ⌘H
- Wiki Protocol Page
- Filter Field Reference
- Protocol Preferences ▶
- Decode As…
- Go to Linked Packet
- Show Linked Packet in New Window

```
0000  01 00 5e 00 00 02 00 0c  cf 42 dd 7c 08 00 45 c0   ..^..... .B.|..E.
0010  00 30 00 00 00 00 02 11  53 e9 ac 14 d7 fd e0 00   .0......  S.......
0020  00 02 07 c1 07 c1 00 1c  a7 d5 00 00 08 05 0f 64   ........  .......d
0030  03 00 63 69 73 63 6f 00  00 00 ac 14 d7 fe          ..cisco. ......
```

Where To Look At?

- Packet...
  - Field values
  - Patterns
  - Types
  - Payload
  - Timing

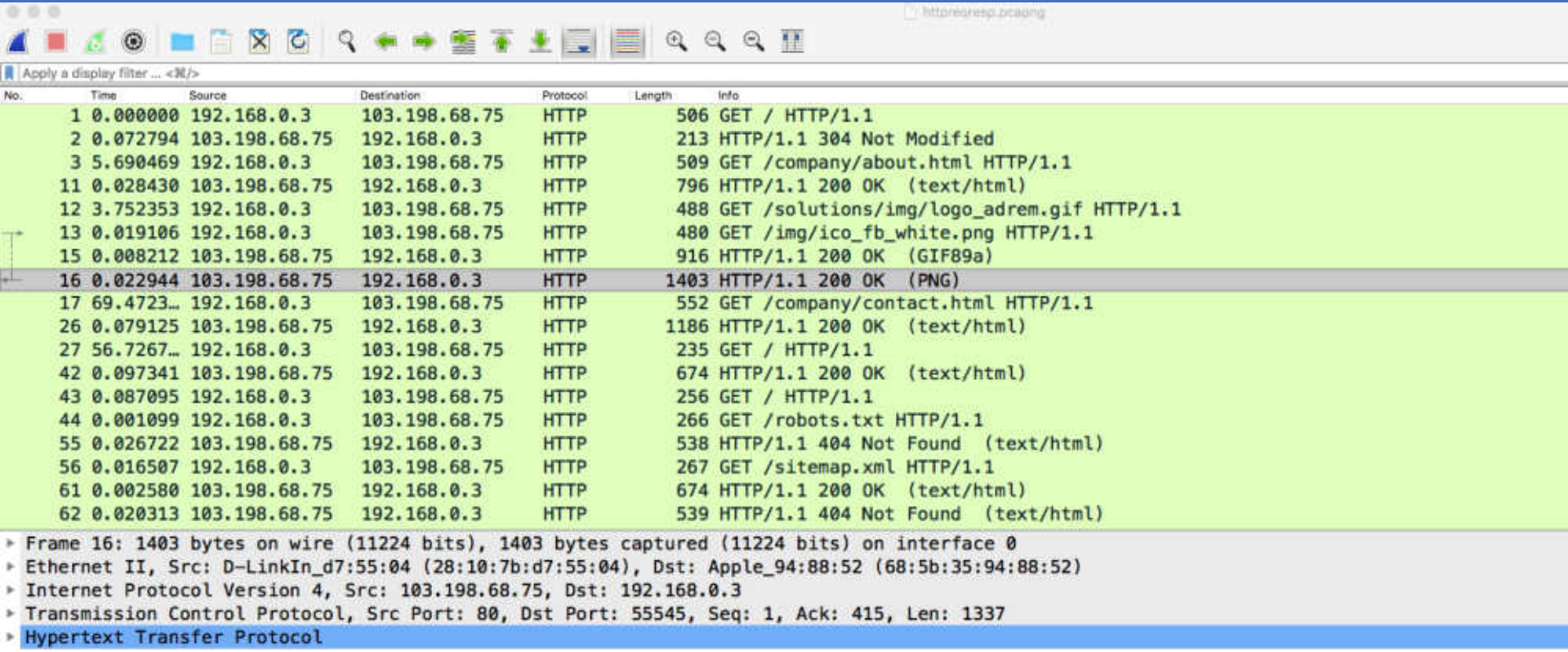
Lab: office_laptop.pcapng

- Let’s do this together!
- Tell me something about this pcap
- What do you see?
- Can you describe what’s going on?
- It’s normal? It’s abnormal? Any evilness?

Lab: maple-tree-inn.pcapng

- Try on your own!
- Tell me something about this pcapng
- What do you see?
- Can you describe what's going on?
- It's normal? It's abnormal? Any evilness?

Be Evil: Know Normal First! [httpreqresp.pcapng]



Apply a display filter ... <⌘/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.3	103.198.68.75	HTTP	506	GET / HTTP/1.1
2	0.072794	103.198.68.75	192.168.0.3	HTTP	213	HTTP/1.1 304 Not Modified
3	5.690469	192.168.0.3	103.198.68.75	HTTP	509	GET /company/about.html HTTP/1.1
11	0.028430	103.198.68.75	192.168.0.3	HTTP	796	HTTP/1.1 200 OK (text/html)
12	3.752353	192.168.0.3	103.198.68.75	HTTP	488	GET /solutions/img/logo_adrem.gif HTTP/1.1
13	0.019106	192.168.0.3	103.198.68.75	HTTP	480	GET /img/ico_fb_white.png HTTP/1.1
15	0.008212	103.198.68.75	192.168.0.3	HTTP	916	HTTP/1.1 200 OK (GIF89a)
16	0.022944	103.198.68.75	192.168.0.3	HTTP	1403	HTTP/1.1 200 OK (PNG)
17	69.4723…	192.168.0.3	103.198.68.75	HTTP	552	GET /company/contact.html HTTP/1.1
26	0.079125	103.198.68.75	192.168.0.3	HTTP	1186	HTTP/1.1 200 OK (text/html)
27	56.7267…	192.168.0.3	103.198.68.75	HTTP	235	GET / HTTP/1.1
42	0.097341	103.198.68.75	192.168.0.3	HTTP	674	HTTP/1.1 200 OK (text/html)
43	0.087095	192.168.0.3	103.198.68.75	HTTP	256	GET / HTTP/1.1
44	0.001099	192.168.0.3	103.198.68.75	HTTP	266	GET /robots.txt HTTP/1.1
55	0.026722	103.198.68.75	192.168.0.3	HTTP	538	HTTP/1.1 404 Not Found (text/html)
56	0.016507	192.168.0.3	103.198.68.75	HTTP	267	GET /sitemap.xml HTTP/1.1
61	0.002580	103.198.68.75	192.168.0.3	HTTP	674	HTTP/1.1 200 OK (text/html)
62	0.020313	103.198.68.75	192.168.0.3	HTTP	539	HTTP/1.1 404 Not Found (text/html)

```
▸ Frame 16: 1403 bytes on wire (11224 bits), 1403 bytes captured (11224 bits) on interface 0
▸ Ethernet II, Src: D-LinkIn_d7:55:04 (28:10:7b:d7:55:04), Dst: Apple_94:88:52 (68:5b:35:94:88:52)
▸ Internet Protocol Version 4, Src: 103.198.68.75, Dst: 192.168.0.3
▸ Transmission Control Protocol, Src Port: 80, Dst Port: 55545, Seq: 1, Ack: 415, Len: 1337
▸ Hypertext Transfer Protocol
```

Don’t try this at hotel!

StarHub

STB INFORMATION

DECODER INFO >

STB Model	Samsung GX-SH530CF
Firmware Version	OTV5.1.2_BLD18.42e_080317_V2_0
CM Firmware	S-SH530CFDEAG-0122
UI Version	1.8.49Pr1
Bootloader Version	8.0.9.5.4
MAC Address	20:D5:BF:D7:1F:B7
IP Address	10.27.30.167
Gateway	10.27.0.1

TOSHIBA

Lab: [sharkfest18asia.pcapng]

sharkfest18asia.pcapng

Apply a display filter ... <⌘/> Expression... +

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	fe80::65:6ee2:2367:16…	ff02::16	ICMPv6	90	Multicast Listener Report Message v2
2	0.000004	172.16.45.72	224.0.0.251	IGMPv2	60	Membership Report group 224.0.0.251
3	0.000005	192.168.2.41	224.0.0.1	IGMPv3	50	Membership Query, general
4	0.000006	192.168.2.41	224.0.0.1	IGMPv3	50	Membership Query, general
5	0.001587	Shenzhen_05:2d:ff	Broadcast	ARP	60	Who has 172.16.43.162? Tell 0.0.0.0
6	0.103045	172.16.43.105	224.0.0.22	IGMPv3	54	Membership Report / Join group 224.0.0.251 for any sources
7	0.612952	172.16.43.28	255.255.255.255	UDP	82	57621 → 57621 Len=40
8	0.613414	Blackber_31:dd:50	Broadcast	ARP	60	Who has 172.16.43.28? Tell 172.16.43.156
9	0.924948	172.16.45.211	224.0.0.22	IGMPv3	60	Membership Report / Join group 224.0.0.251 for any sources

```
▸ Frame 1: 90 bytes on wire (720 bits), 90 bytes captured (720 bits) on interface 0
▸ Ethernet II, Src: Apple_40:8a:a4 (90:60:f1:40:8a:a4), Dst: IPv6mcast_16 (33:33:00:00:00:16)
▸ Internet Protocol Version 6, Src: fe80::65:6ee2:2367:168e, Dst: ff02::16
▸ Internet Control Message Protocol v6

0000  33 33 00 00 00 16 90 60  f1 40 8a a4 86 dd 60 00   33·····` ·@····`·
0010  00 00 00 24 00 01 fe 80  00 00 00 00 00 00 00 65   ···$···· ·******e
0020  6e e2 23 67 16 8e ff 02  00 00 00 00 00 00 00 00   n·#g···· ********
0030  00 00 00 00 00 16 3a 00  01 00 05 02 00 00 8f 00   ******:· ********
0040  c5 d3 00 00 00 01 04 00  00 00 ff 02 00 00 00 00   ******** ********
0050  00 00 00 00 00 00 00 00  00 fb                     ********  **
```

sharkfest18asia.pcapng Packets: 26608 · Displayed: 26608 (100.0%) Profile: Sharkfest18US

Lab: [sharkfest18asia.pcapng]

https://stackoverflow.com/questions/21899933/network-broadcast-from-bluestacks-beacon-v1

network broadcast from bluestacks - Beacon-v1

3

The latest update of Bluestacks is sending a network broadcast every 2 seconds from port 10505.

Beacon-v1|pcName|WindowsPC OpenSensor-v2|54321

to IP 255.255.255.255

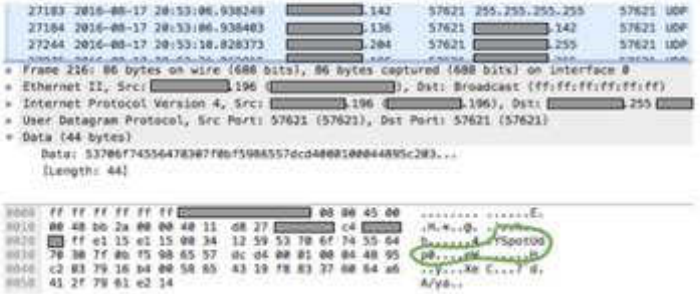
this wasnt happening with the previous version. is this some autosync announcement waiting to talk with another device? i dont want bluestacks talking to other networked devices unless i tell it to do so. i havent checked off or agreed to anything yet that says it requires a network broadcast like this.

even if it's only 53 bytes it's still network pollution to me. how can i turn this off until i actually want it ? thanks

https://awakesecurity.com/10-minutes-life-network/

Hmm, Spotify had a P2P Network?

While searching across traffic communicating with Spotify's music streaming service, I discovered a high-numbered UDP port sending several packets each minute to the same UDP port on two broadcast addresses, as well as responses from local devices. Looking more closely, it turns out that each packet had a "SpotUdp" plaintext string in its payload, which piqued my interest. After a brief search, I discovered that until mid-2014, Spotify had a **P2P network** that a lot of people didn't seem to know about. While the network was phased out over two years ago, this particular connection still exists, and still clearly has some local subnet P2P communication. If legacy things like this exist but often go undetected, imagine how hard it is for junior analysts trying to hunt and discern what is or isn't legitimate!



Don’t try this at home!

Owh my ISP!

No.	Time	Source	Destination	Protocol	Length	ID	Info
372	0.000000	D-LinkIn_d7:55:0d	IETF-VRRP-VRID_0d	PPP PAP	64	500	Authenticate-Request (Peer-ID='maher910@unifi', Password= [redacted])
638	45.3987…	D-LinkIn_d7:55:0d	IETF-VRRP-VRID_0d	PPP PAP	64	500	Authenticate-Request (Peer-ID='maher910@unifi', Password= [redacted])
640	0.041034	IETF-VRRP-VRID_0d	D-LinkIn_d7:55:0d	PPP PAP	62	500	Authenticate-Ack (Message='Authentication success,Welcome!')

```
▸ Frame 372: 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface 0
▸ Ethernet II, Src: D-LinkIn_d7:55:0d (28:10:7b:d7:55:0d), Dst: IETF-VRRP-VRID_0d (00:00:5e:00:01:0d)
▸ 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 500
▸ PPP-over-Ethernet Session
▸ Point-to-Point Protocol
▾ PPP Password Authentication Protocol
    Code: Authenticate-Request (1)
    Identifier: 1
    Length: 33
  ▾ Data
      Peer-ID-Length: 14
      Peer-ID: maher910@unifi
      Password-Length: 13
      Password: [redacted]
```

I know VLANs! Let's do this!

vlan.id

No.	Time	Source	Destination	Protocol	Length	ID	Info
270	28.229898	Cisco_64:8a:90	PVST+	STP	68	34	Conf. TC + Root =
272	28.588894	Cisco_64:8a:90	PVST+	STP	68	1010	Conf. TC + Root =
275	29.617946	Cisco_64:8a:90	PVST+	STP	68	2020	Conf. Root = 3276
278	30.218976	Cisco_64:8a:90	PVST+	STP	68	3030	Conf. Root = 3276
281	30.220688	Cisco_64:8a:90	PVST+	STP	68	10	Conf. Root = 3276
282	30.221071	Cisco_64:8a:90	PVST+	STP	68	12	Conf. Root = 3276
283	30.221811	Cisco_64:8a:90	PVST+	STP	68	13	Conf. Root = 3276
284	30.222184	Cisco_64:8a:90	PVST+	STP	68	14	Conf. Root = 3276
285	30.227006	Cisco_64:8a:90	PVST+	STP	68	23	Conf. Root = 3276
286	30.227449	Cisco_64:8a:90	PVST+	STP	68	24	Conf. Root = 3276
287	30.229830	Cisco_64:8a:90	PVST+	STP	68	34	Conf. Root = 3276

```
▸ Frame 285: 68 bytes on wire (544 bits), 68 bytes captured (544 bits) on interface 0
▸ Ethernet II, Src: Cisco_64:8a:90 (7c:95:f3:64:8a:90), Dst: PVST+ (01:00:0c:cc:cc:cd)
▾ 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 23
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0001 0111 = ID: 23
    Length: 50
▸ Logical-Link Control
▸ Spanning Tree Protocol

0000  01 00 0c cc cc cd 7c 95  f3 64 8a 90 81 00 00 17   ......|· ·d····
0010  00 32 aa aa 03 00 00 0c  01 0b 00 00 00 00 00 80   ·2····· ........
0020  17 7c 95 f3 64 8a 80 00  00 00 00 80 17 7c 95 f3   ·|··d··· ·····|··
0030  64 8a 80 80 10 00 00 14  00 02 00 0f 00 00 00 00   d····· ........
0040  00 02 00 17                                         ····
```

Configure IPv4: Using DHCP

IP Address: 192.168.30.4

Subnet Mask: 255.255.255.0

Router: 192.168.30.254

DNS Server: 8.8.4.4

vlan1010
Connected

vlan2020
Connected

vlan3030
Connected

Got it! Be Evil!

No.	Source	Destination	Protocol	ID	Info
13893	10.254.253.10	10.63.4.57	TCP	209	8082→4152 [SYN, ACK…
13897	HuaweiTe_4d:dd…		ARP	209	Who has 10.63.2.170…
13898	HuaweiTe_4d:dd…		ARP	209	Who has 10.63.9.64?…
13899	HuaweiTe_4d:dd…		ARP	209	Who has 10.63.7.165…
13900	HuaweiTe_4d:dd…		ARP		

maher — -bash — 80×24

```
Starting Nmap 7.12 ( https://nmap.org ) at 2016-12-16 11:43 MYT
Nmap scan report for 10.63.63.254
Host is up (0.051s latency).
Not shown: 999 closed ports
PORT    STATE    SERVICE VERSION
22/tcp filtered ssh
MAC Address: 28:6E:D4:4D:DD:B6 (Huawei Technologies)
Warning: OSScan results may be unreliable because we could not find at least 1 o
pen and 1 closed port
Device type: switch
Running: Huawei VRP 3.X
OS CPE: cpe:/h:huawei:s2326 cpe:/o:huawei:vrp:3
OS details: Huawei S2326 switch, Huawei S9300 switch, Huawei VRP 3 switch
Network Distance: 1 hop

TRACEROUTE
HOP RTT      ADDRESS
1   51.20 ms 10.63.63.254

OS and Service detection performed. Please report any incorrect results at https
://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 21.76 seconds
kittyhawk:~ maher$
```

Network

TCP/IP | DNS | WINS | Proxies | Hardware

IPv4: Using DHCP
Address: 10.63.29.198
Mask: 255.255.192.0
Router: 10.63.63.254
IPv6: Automatically
Router:
Address:
Length:

Renew DHCP Lease
DHCP Client ID:
(If required)

Cancel | OK

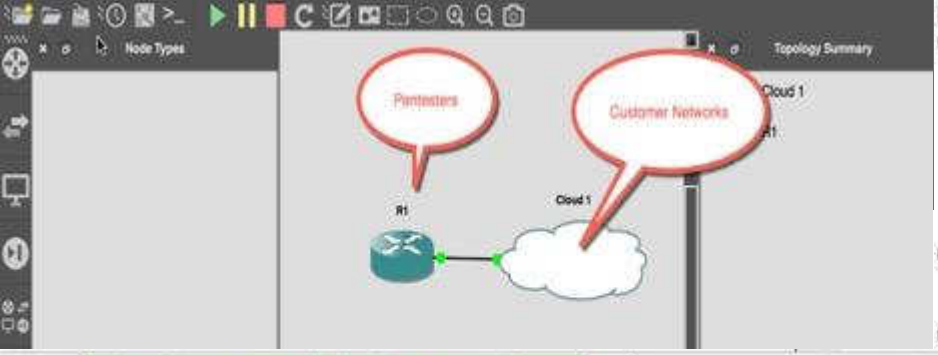
Re-route my traffic: ospf.pcapng

ospf.pcapng

Apply a display filter ... <⌘/>

No.	Time	Source	Destination	Protocol	Source OSPF Router	Area ID	Auth Type	Info
1	0.000000	192.168.0.216	224.0.0.5	OSPF	1.1.1.1	0.0.0.0	Null	Hello Packet
2	9.741205	192.168.0.216	224.0.0.5	OSPF	1.1.1.1	0.0.0.0	Null	Hello Packet
3	9.985707	192.168.0.216	224.0.0.5	OSPF	1.1.1.1	0.0.0.0	Null	Hello Packet
4	9.277099	192.168.0.216	224.0.0.5	OSPF	1.1.1.1	0.0.0.0	Null	Hello Packet
5	9.657609	192.168.0.216	224.0.0.5	OSPF	1.1.1.1	0.0.0.0	Null	Hello Packet
6	9.982329	192.168.0.216	224.0.0.5	OSPF	1.1.1.1	0.0.0.0	Null	Hello Packet
7	9.138048	192.168.0.216	224.0.0.5	OSPF	1.1.1.1	0.0.0.0	Null	Hello Packet
8	9.618687	192.168.0.216	224.0.0.5	OSPF	1.1.1.1	0.0.0.0	Null	Hello Packet

```
▸ Frame 1: 90 bytes on wire (720 bits), 90 bytes captured (720 bits) on interface 0
▸ Ethernet II, Src: Cisco_e9:d2:a0 (30:e4:db:e9:d2:a0), Dst: IPv4mcast_05 (01:00:5e:00:00:05)
▸ Internet Protocol Version 4, Src: 192.168.0.216, Dst: 224.0.0.5
▾ Open Shortest Path First
  ▸ OSPF Header
  ▸ OSPF Hello Packet
```




```
PENTESTER-R1#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS leve
       ia - IS-IS inter area, * - candidate default, U - per-user static
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.168.0.1 to network 0.0.0.0

     1.0.0.0/32 is subnetted, 1 subnets
O       1.1.1.1 [110/11] via 192.168.0.216, 00:00:51, FastEthernet0/0
```

Filter: eth.addr==98:5a:eb:db:4f:3e && ospf Expression... Clear Apply Save

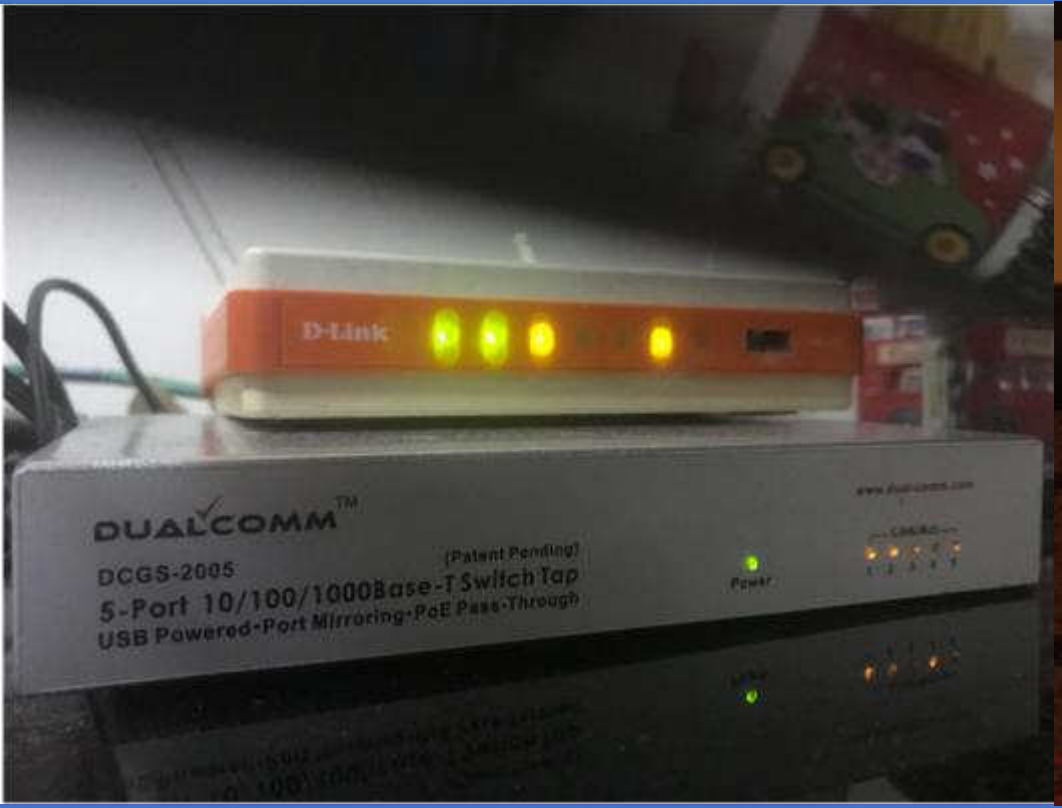
No.	Time	Source	Destination	Time to live	Protocol	Info
3429	8.462s	192.168.0.216	ospf-all.mcast.net	1	OSPF	Hello Packet
3430	0.013s	192.168.0.192	192.168.0.216	1	OSPF	DB Description
3431	0.008s	192.168.0.192	192.168.0.216	1	OSPF	DB Description
3432	0.000s	192.168.0.216	192.168.0.192	1	OSPF	DB Description
3433	0.000s	192.168.0.216	192.168.0.192	1	OSPF	DB Description
3434	0.003s	192.168.0.192	192.168.0.216	1	OSPF	Hello Packet
3435	0.006s	192.168.0.192	192.168.0.216	1	OSPF	Hello Packet
3436	0.005s	192.168.0.192	192.168.0.216	1	OSPF	DB Description
3437	0.004s	192.168.0.192	192.168.0.216	1	OSPF	DB Description
3438	0.000s	192.168.0.216	192.168.0.192	1	OSPF	DB Description
3439	0.006s	192.168.0.192	192.168.0.216	1	OSPF	DB Description

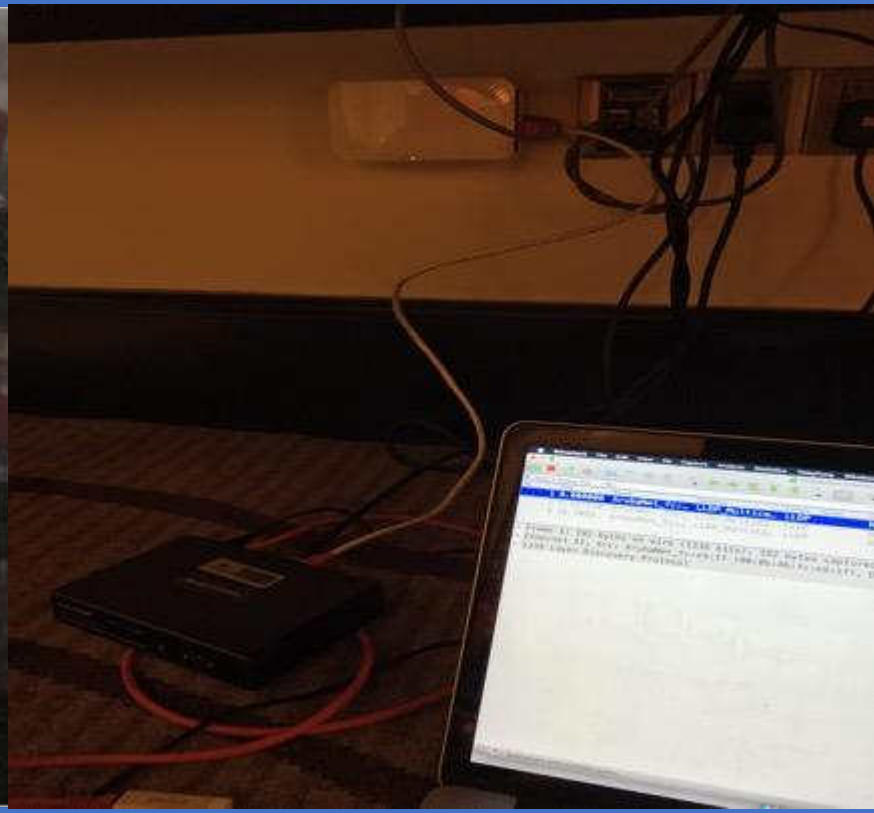
maher — R1 — telnet 127.0.0.1 2001 — 114×17

(Good: Fa
(Bad: Fal
Source: 192
Destination:
(Source Gec
(Destination
Open Shortest
OSPF Heade
Version: 2
Message
Packet Le
Source OS
Area ID: 0
Checksum
Auth Type
Auth Data

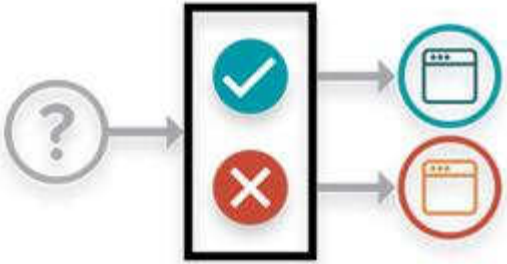
```
PENTESTER-R1(config-router)#network 0.0.0.0 255.255.255.255 a 0
PENTESTER-R1(config-router)#int fa0/0
PENTESTER-R1(config-if)#ip add
PENTESTER-R1(config-if)#ip address d
PENTESTER-R1(config-if)#ip address dhcp
PENTESTER-R1(config-if)#^Z
PENTESTER-R1#
*Mar  1 00:04:57.811: %SYS-5-CONFIG_I: Configured from console by console
PENTESTER-R1#
*Mar  1 00:05:05.195: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet0/0 assigned DHCP address 192.168.0.192, mask
255.255.255.0, hostname PENTESTER-R1

PENTESTER-R1#
*Mar  1 00:05:12.683: %OSPF-5-ADJCHG: Process 99, Nbr 1.1.1.1 on FastEthernet0/0 from LOADING to FULL, Loading Do
e
```

Tips: Don’t bring any taps!





Please authenticate me!



Avoid Default at ALL COST!

No.	Time	Source	Destination	Protocol	Length	Info
99	0.000328	192.168.0.3	192.168.0.1	TCP	568	62989 → 80 [PSH, ACK] Seq=1 Ack=1 Win=131744 Len=502 TSval=2148815…
100	0.000197	192.168.0.3	192.168.0.1	HTTP	127	POST /login.php HTTP/1.1 (application/x-www-form-urlencoded)
101	0.000210	192.168.0.1	192.168.0.3	TCP	66	80 → 62989 [ACK] Seq=1 Ack=503 Win=6864 Len=0 TSval=3463064 TSecr=…
102	0.000323	192.168.0.1	192.168.0.3	TCP	66	80 → 62989 [ACK] Seq=1 Ack=564 Win=6864 Len=0 TSval=3463064 TSecr=…
103	0.031749	192.168.0.1	192.168.0.3	TCP	219	80 → 62989 [PSH, ACK] Seq=1 Ack=564 Win=6864 Len=153 TSval=3463072…
104	0.000049	192.168.0.3	192.168.0.1	TCP	66	62989 → 80 [ACK] Seq=564 Ack=154 Win=131584 Len=0 TSval=214881620 …
105	0.000598	192.168.0.1	192.168.0.3	TCP	149	80 → 62989 [PSH, ACK] Seq=154 Ack=564 Win=6864 Len=83 TSval=346307…
106	0.000024	192.168.0.3	192.168.0.1	TCP	66	62989 → 80 [ACK] Seq=564 Ack=237 Win=131520 Len=0 TSval=214881620 …
107	0.000938	192.168.0.1	192.168.0.3	HTTP	71	HTTP/1.1 200 OK (text/html)
108	0.000024	192.168.0.3	192.168.0.1	TCP	66	62989 → 80 [ACK] Seq=564 Ack=242 Win=131520 Len=0 TSval=214881621 …
109	0.049144	192.168.0.3	192.168.0.1	HTTP	465	GET /index.php HTTP/1.1

```
▸ Frame 100: 127 bytes on wire (1016 bits), 127 bytes captured (1016 bits) on interface 0
▸ Ethernet II, Src: Apple_94:88:52 (68:5b:35:94:88:52), Dst: D-LinkIn_d7:55:04 (28:10:7b:d7:55:04)
▸ Internet Protocol Version 4, Src: 192.168.0.3, Dst: 192.168.0.1
▸ Transmission Control Protocol, Src Port: 62989, Dst Port: 80, Seq: 503, Ack: 1, Len: 61
▸ [2 Reassembled TCP Segments (563 bytes): #99(502), #100(61)]
▸ Hypertext Transfer Protocol
▾ HTML Form URL Encoded: application/x-www-form-urlencoded
  ▸ Form item: "ACTION_POST" = "LOGIN"
  ▸ Form item: "LOGIN_USER" = "admin"
  ▸ Form item: "LOGIN_PASSWD" = ""
  ▸ Form item: "login" = "Login "
```

I’m a Pentester! Watch Out!

Ed has a great quote on this: "If a penetration tester promises they will not crash a system, it means they are lying to you, or they are not planning on sending any packets to your network.” – Ed Skoudis

Lab: [pentester.pcapng]

pentester.pcapng

Apply a display filter ... <⌘/> Expression...

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.0.1	192.168.0.57	DHCP	DHCP Offer - Transaction ID 0x2a7c5f36
2	0.004800	192.168.0.1	192.168.0.57	DHCP	DHCP ACK - Transaction ID 0x2a7c5f36
3	1.185361	192.168.0.1	192.168.0.57	ICMP	Echo (ping) request id=0x2c10, seq=0/0, ttl=64 (reply in 5)
4	0.000017	192.168.0.57	8.8.8.8	DNS	Standard query 0x2e25 SOA local
5	0.000003	192.168.0.57	192.168.0.1	ICMP	Echo (ping) reply id=0x2c10, seq=0/0, ttl=64 (request in 3)
6	0.023522	8.8.8.8	192.168.0.57	DNS	Standard query response 0x2e25 No such name SOA local SOA a.root-servers.net
7	0.221721	192.168.0.57	8.8.8.8	DNS	Standard query 0xca93 SOA local
8	0.021747	8.8.8.8	192.168.0.57	DNS	Standard query response 0xca93 No such name SOA local SOA a.root-servers.net
9	11.5780…	192.168.0.57	192.168.0.255	BJNP	Scanner Command: Discover
10	0.000010	192.168.0.57	192.168.0.255	BJNP	Scanner Command: Discover
11	0.010145	192.168.0.57	192.168.0.255	BJNP	Scanner Command: Discover
12	0.000008	192.168.0.57	192.168.0.255	BJNP	Scanner Command: Discover
13	0.825021	192.168.0.57	255.255.255.255	UDP	43704 → 3289 Len=15
14	1.085145	192.168.0.57	255.255.255.255	UDP	39531 → 1124 Len=37
15	6.795231	192.168.0.57	8.8.8.8	DNS	Standard query 0x33c5 A 2.debian.pool.ntp.org
16	0.000002	192.168.0.57	8.8.8.8	DNS	Standard query 0xabcf AAAA 2.debian.pool.ntp.org
17	0.074097	8.8.8.8	192.168.0.57	DNS	Standard query response 0x33c5 A 2.debian.pool.ntp.org A 103.16.182.23 A 202.45.138.123
18	0.000003	8.8.8.8	192.168.0.57	DNS	Standard query response 0xabcf AAAA 2.debian.pool.ntp.org AAAA 2402:1f00:8000:800::8d2
19	0.000928	192.168.0.57	103.16.182.23	NTP	NTP Version 4, client
20	0.042235	103.16.182.23	192.168.0.57	NTP	NTP Version 4, server

Lab: [pentester.pcapng]

- Try to do on your own!
- Tell me something about this pcap?
- What do you see?
- Can you describe what's going on?
- It's normal? It's abnormal? Any evilness?

Cookies anyone?

Cookie Insert Information Leakage

While cookie insert is a great persistence method, the default settings create some security issues with information leakage. The default F5 cookie has the following format –

> BIGipServertest_pool=335653056.20480.0000
> BIGipServer<pool name> =<coded server IP>.<coded server port>.0000

The cookie tells us the following information –

- BIGipServer – We now know that the server is behind an F5 BigIP device.
- <pool name> – The name of the pool as configured on the F5.
- <coded server IP> – The real IP of the server with a simple encoding method.
- <coded server port> – The real port of the server with a simple encoding method.

Decode The Cookies

```
▾ Hypertext Transfer Protocol
  ▾ HTTP/1.1 200 OK\r\n
    ▸ [Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]
      Response Version: HTTP/1.1
      Status Code: 200
      [Status Code Description: OK]
      Response Phrase: OK
    Date: Mon, 18 Jun 2018 05:53:17 GMT\r\n
    Server: Apache\r\n
    X-Powered-By: PHP/5.6.13\r\n
    Expires: Thu, 19 Nov 1981 08:52:00 GMT\r\n
    Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0\r\n
    Pragma: no-cache\r\n
    X-FRAME-OPTIONS: SAMEORIGIN\r\n
    Content-Type: text/html; charset=UTF-8\r\n
    Set-Cookie: PHPSESSID=id0d07a8iu6icl9s4b17qf5p84; path=/\r\n
    Set-Cookie: BIGipServerNEW_EPMS_VS=1695918272.20480.0000; path=/\r\n
```

```
./BIG-IP_cookie_decoder.py 1695918272.20480.0000

[*] String to decode: 1695918272.20480.0000
[*] Decoded IP:   192.168.21.101
[*] Decoded port: 80
```

More PCAPs To Enhance Your Skills

Experts in network security monitoring and network forensics

NETRESEC

NETRESEC | Products | Training | Resources | Blog | About Netresec

NETRESEC > Resources > PCAP Files

Publicly available PCAP files

This is a list of public packet capture repositories, which are freely available on the Internet. Most of the sites listed below share Full Packet Capture (FPC) files, but some do unfortunately only have truncated frames.

Cyber Defence Exercises (CDX)

This category includes network traffic from exercises and competitions, such as Cyber Defense Exercises (CDX) and red-team/blue-team competitions.

MACCDC - Pcaps from National CyberWatch Mid-Atlantic Collegiate Cyber Defense Competition
https://www.netresec.com/?page=MACCDC

ISTS - Pcaps from the Information Security Talent Search
https://www.netresec.com/?page=ISTS

WRCCDC - Pcaps from the Western Regional Collegiate Cyber Defense Competition (over 1TB of PCAPs)
https://archive.wrccdc.org/pcaps/

Captures from the "2009 Inter-Service Academy Cyber Defense Competition" served by Information Technology Operations Center (ITOC), United States Military Academy
http://www.westpoint.edu/crc/SitePages/DataSets.aspx

PCAP PCAP P

MACCDC 2012

maccdc2012_00000.pcap.gz	316M
maccdc2012_00001.pcap.gz	279M
maccdc2012_00002.pcap.gz	393M
maccdc2012_00003.pcap.gz	481M
maccdc2012_00004.pcap.gz	428M
maccdc2012_00005.pcap.gz	227M
maccdc2012_00006.pcap.gz	412M
maccdc2012_00007.pcap.gz	344M
maccdc2012_00008.pcap.gz	194M
maccdc2012_00009.pcap.gz	218M
maccdc2012_00010.pcap.gz	223M
maccdc2012_00011.pcap.gz	276M
maccdc2012_00012.pcap.gz	313M
maccdc2012_00013.pcap.gz	532M
maccdc2012_00014.pcap.gz	274M
maccdc2012_00015.pcap.gz	301M
maccdc2012_00016.pcap.gz	195M

Show Data As Text

No.	Time	Source	Destination	Protocol	Info
1	0.000	192.168.1.70	209.200.39.18	TCP	59609 → 800 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=1 SACK_PERM=1
2	0.147	209.200.39.18	192.168.1.70	TCP	800 → 59609 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1360 WS=256…
3	0.000	192.168.1.70	209.200.39.18	TCP	59609 → 8 192 Len=0
4	0.147	209.200.39.18	192.168.1.70	TCP	800 → 59 Win=17664 Len=143
5	0.036	192.168.1.70	209.200.39.18	TCP	59609 → 8 4 Win=8049 Len=14
6	0.151	209.200.39.18	192.168.1.70	TCP	800 → 59 15 Win=17664 Len=64
7	0.061	192.168.1.70	209.200.39.18	TCP	59609 → 8 n=7985 Len=0
8	9.471	192.168.1.70	209.200.39.18	TCP	59609 → 8 08 Win=7985 Len=13
9	0.148	209.200.39.18	192.168.1.70	TCP	800 → 59 28 Win=17408 Len=34
10	0.053	192.168.1.70	209.200.39.18	TCP	59609 → 8 n=7951 Len=0
11	6.671	192.168.1.70	209.200.39.18	TCP	59609 → 8 42 Win=7951 Len=13
12	0.144	209.200.39.18	192.168.1.70	TCP	800 → 59 41 Win=17408 Len=15
13	0.051	192.168.1.70	209.200.39.18	TCP	59609 → 8 n=7936 Len=0
14	2.099	192.168.1.70	209.200.39.18	TCP	59609 → 8 57 Win=7936 Len=27

```
▸ Frame 4: 197 bytes on wire (1576 bits), 197 bytes captured (1
▸ Ethernet II, Src: 2wire_2c:0b:15 (dc:7f:a4:2c:0b:15), Dst: Mi
▸ Internet Protocol Version 4, Src: 209.200.39.18, Dst: 192.168
▸ Transmission Control Protocol, Src Port: 800, Dst Port: 59609
▾ Data (143 bytes)
    Data: 3232302d46696c655a696c6c6120536572766572220302e39...
    [Length: 143]
```

- Expand Subtrees ⇧→
- Collapse Subtrees ⇧←
- Expand All ⌘→
- Collapse All ⌘←
- Apply as Column ⇧⌘I
- Apply as Filter ▸
- Prepare a Filter ▸
- Conversation Filter ▸
- Colorize with Filter ▸
- Follow ▸
- Copy ▸
- Show Packet Bytes… ⇧⌘O
- Export Packet Bytes… ⇧⌘X
- Wiki Protocol Page
- Filter Field Reference
- Protocol Preferences ▸
- Decode As…
- Go to Linked Packet
- Show Linked Packet in New Window

- Open Data preferences…
- Show not dissected data on new Packet Bytes pane
- Try to uncompress zlib compressed data
- Show data as text
- Generate MD5 hash
- Disable Data…

Show Data As Text

No.	Time	Source	Destination	Protocol	Info
1	0.000	192.168.1.70	209.200.39.18	TCP	59609 → 800 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=1 SACK_PERM=1
2	0.147	209.200.39.18	192.168.1.70	TCP	800 → 59609 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1360 WS=256…
3	0.000	192.168.1.70	209.200.39.18	TCP	59609 → 800 [ACK] Seq=1 Ack=1 Win=8192 Len=0
4	0.147	209.200.39.18	192.168.1.70	TCP	800 → 59609 [PSH, ACK] Seq=1 Ack=1 Win=17664 Len=143
5	0.036	192.168.1.70	209.200.39.18	TCP	59609 → 800 [PSH, ACK] Seq=1 Ack=144 Win=8049 Len=14
6	0.151	209.200.39.18	192.168.1.70	TCP	800 → 59609 [PSH, ACK] Seq=144 Ack=15 Win=17664 Len=64
7	0.061	192.168.1.70	209.200.39.18	TCP	59609 → 800 [ACK] Seq=15 Ack=208 Win=7985 Len=0
8	9.471	192.168.1.70	209.200.39.18	TCP	59609 → 800 [PSH, ACK] Seq=15 Ack=208 Win=7985 Len=13
9	0.148	209.200.39.18	192.168.1.70	TCP	800 → 59609 [PSH, ACK] Seq=208 Ack=28 Win=17408 Len=34
10	0.053	192.168.1.70	209.200.39.18	TCP	59609 → 800 [ACK] Seq=28 Ack=242 Win=7951 Len=0
11	6.671	192.168.1.70	209.200.39.18	TCP	59609 → 800 [PSH, ACK] Seq=28 Ack=242 Win=7951 Len=13
12	0.144	209.200.39.18	192.168.1.70	TCP	800 → 59609 [PSH, ACK] Seq=242 Ack=41 Win=17408 Len=15
13	0.051	192.168.1.70	209.200.39.18	TCP	59609 → 800 [ACK] Seq=41 Ack=257 Win=7936 Len=0
14	2.099	192.168.1.70	209.200.39.18	TCP	59609 → 800 [PSH, ACK] Seq=41 Ack=257 Win=7936 Len=27

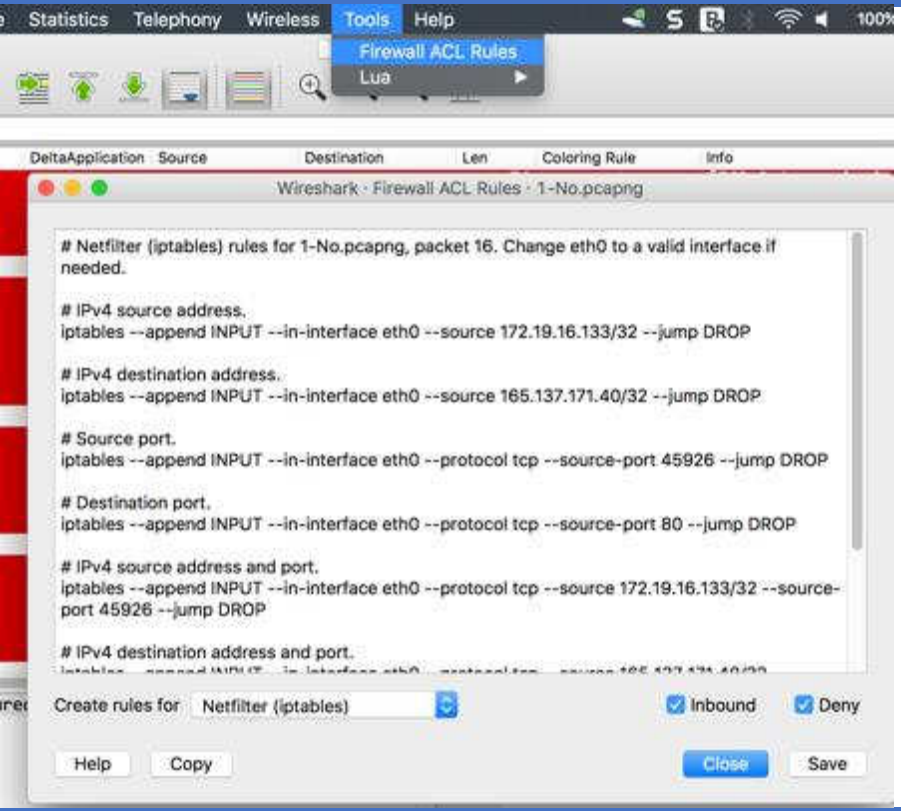
```
▸ Frame 4: 197 bytes on wire (1576 bits), 197 bytes captured (1576 bits) on interface 0
▸ Ethernet II, Src: 2wire_2c:0b:15 (dc:7f:a4:2c:0b:15), Dst: Micro-St_a6:41:fd (d4:3d:7e:a6:41:fd)
▸ Internet Protocol Version 4, Src: 209.200.39.18, Dst: 192.168.1.70
▸ Transmission Control Protocol, Src Port: 800, Dst Port: 59609, Seq: 1, Ack: 1, Len: 143
▾ Data (143 bytes)
    Data: 3232302d46696c655a696c6c6120536572766572202302e39...
    [Length: 143]
```

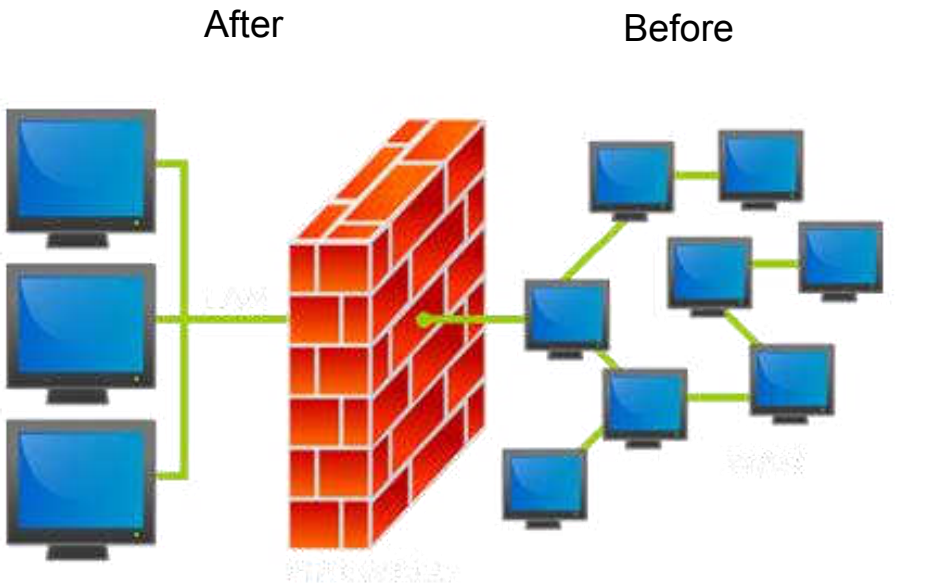
Ahaaa! FTP using Port 800?

No.	Time	Source	Destination	Protocol	Info
1	0.000	192.168.1.70	209.200.39.18	TCP	59609 → 800 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=1 SACK_PERM=1
2	0.147	209.200.39.18	192.168.1.70	TCP	800 → 59609 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1360 WS=256…
3	0.000	192.168.1.70	209.200.39.18	TCP	59609 → 800 [ACK] Seq=1 Ack=1 Win=8192 Len=0
4	0.147	209.200.39.18	192.168.1.70	TCP	800 → 59609 [PSH, ACK] Seq=1 Ack=1 Win=17664 Len=143
5	0.036	192.168.1.70	209.200.39.18	TCP	59609 → 800 [PSH, ACK] Seq=1 Ack=144 Win=8049 Len=14
6	0.151	209.200.39.18	192.168.1.70	TCP	800 → 59609 [PSH, ACK] Seq=144 Ack=15 Win=17664 Len=64
7	0.061	192.168.1.70	209.200.39.18	TCP	59609 → 800 [ACK] Seq=15 Ack=208 Win=7985 Len=0
8	9.471	192.168.1.70	209.200.39.18	TCP	59609 → 800 [PSH, ACK] Seq=15 Ack=208 Win=7985 Len=13
9	0.148	209.200.39.18	192.168.1.70	TCP	800 → 59609 [PSH, ACK] Seq=208 Ack=28 Win=17408 Len=34
10	0.053	192.168.1.70	209.200.39.18	TCP	59609 → 800 [ACK] Seq=28 Ack=242 Win=7951 Len=0
11	6.671	192.168.1.70	209.200.39.18	TCP	59609 → 800 [PSH, ACK] Seq=28 Ack=242 Win=7951 Len=13
12	0.144	209.200.39.18	192.168.1.70	TCP	800 → 59609 [PSH, ACK] Seq=242 Ack=41 Win=17408 Len=15
13	0.051	192.168.1.70	209.200.39.18	TCP	59609 → 800 [ACK] Seq=41 Ack=257 Win=7936 Len=0
14	2.099	192.168.1.70	209.200.39.18	TCP	59609 → 800 [PSH, ACK] Seq=41 Ack=257 Win=7936 Len=27

```
▸ Frame 4: 197 bytes on wire (1576 bits), 197 bytes captured (1576 bits) on interface 0
▸ Ethernet II, Src: 2wire_2c:0b:15 (dc:7f:a4:2c:0b:15), Dst: Micro-St_a6:41:fd (d4:3d:7e:a6:41:fd)
▸ Internet Protocol Version 4, Src: 209.200.39.18, Dst: 192.168.1.70
▸ Transmission Control Protocol, Src Port: 800, Dst Port: 59609, Seq: 1, Ack: 1, Len: 143
▾ Data (143 bytes)
    Data: 3232302d46696c655a696c6c6120536572766572220302e39...
    Text: 220-FileZilla Server 0.9.60 beta\r\n220-written by Tim Kosse (tim.kosse@filezilla-project.org)\r\n220 Please visit h
    [Length: 143]
```

Firewall ACL Rules





Why Curiosity is Important

1. Keep an open mind
2. Don’t take things as granted
3. Ask questions relentlessly
4. Don’t label something as boring
5. See learning as something fun
6. Read diverse kinds of reading

*lifehacks.org

Usually life is very simple...we complicate it by imagining a non existent problem..😇

12:36 PM

Key Takeaway

You might not follow the World Cup, but I'm sure you know what an own goal is.

Painful, shameful and totally avoidable, it may look something like this:

Now, what does it have to do with cybersecurity? Everything:

My friend, DON'T score an own goal just by having a sloppy defence against malware or other threat!

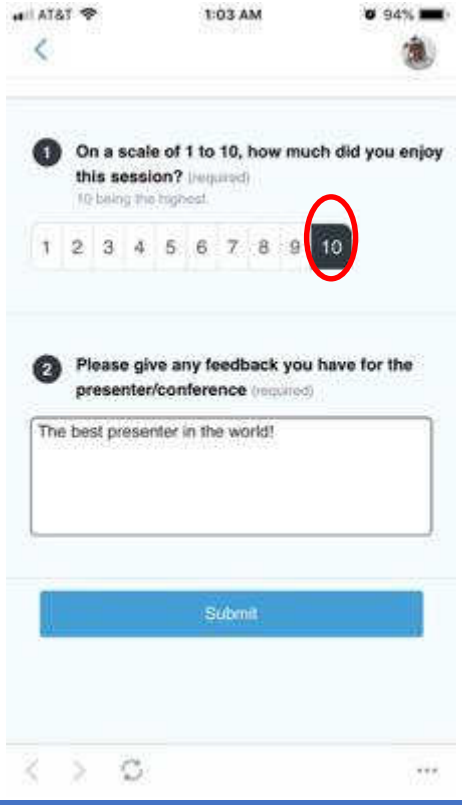
May Packet be the force with you…

Next SharkFest?

Thank You!

- Maher Adib.
- Based in Kuala Lumpur, Malaysia.
- Email: maher@ofisgate.com
- FB: OfisgateAcademy

Don’t forget to submit feedback!



- Maher Adib.
- Based in Kuala Lumpur, Malaysia.
- Email: maher@ofisgate.com
- FB: OfisgateAcademy