
Lesson 8: Penetration Testing and Audits



Vulnerability Scan v Pen Test v Audit



Vulnerability Scan

Uses tools to scan for known vulnerabilities



Pen Test

Actually attempts to break into the network



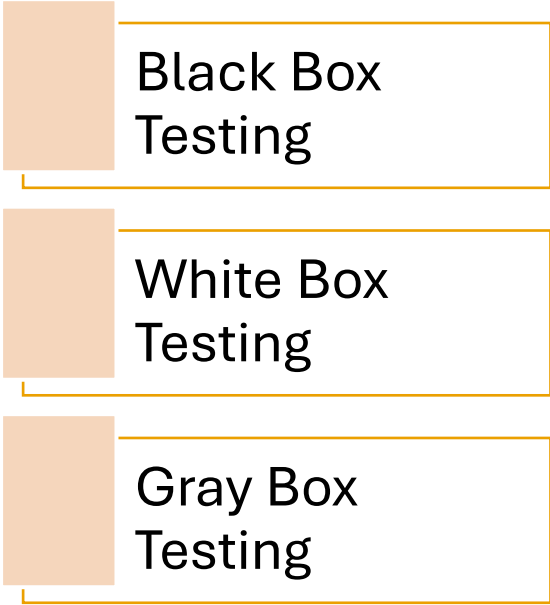
Audit

Checking logs, policies, looks for compliance to policies

Terminology

- ▶ **Ad hoc testing:** Testing carried out with no systematic approach or methodology. It is hoped that this book will steer you away from that.
- ▶ **Black hat hacker:** A hacker who does break the law. This term is synonymous with cracker, but the term *black hat hacker* is far more common. Contrary to some media portrayals, a black hat is not necessarily any more skilled. Someone can break the law and still have only minimal skill.
- ▶ **Cracker:** One who breaks into a system in order to do something malicious, illegal, or harmful. Synonymous with black hat hacker.
- ▶ **Ethical hacking:** Someone who is using hacking techniques for legal and ethical purposes.
- ▶ **Footprinting:** Scanning a target to learn about that target.
- ▶ **Gray hat hacker:** A hacker who usually obeys the law but in some instances will cross the line into black hat hacking.
- ▶ **Hacker:** One who tries to learn about a system by examining it in detail by reverse-engineering or probing the system. This is an important definition. Hackers, are not necessarily criminals. One can be a hacker and never break the law, nor do anything unethical.
- ▶ **Script kiddy:** A slang term for an unskilled person who purports to be a skilled hacker. Some people download a tool or two, learn to use those, then consider themselves great hackers, when they are not.
- ▶ **White hat hacker:** A hacker who does not break the law, often synonymous with ethical hacker. Essentially this is a person who uses hacking skills in a legal and ethical manner.

Testing Terms



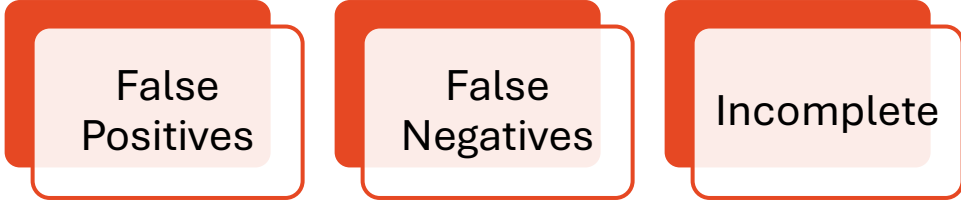
Black Box
Testing

White Box
Testing

Gray Box
Testing

Note: White box is also known as clear box testing, glass box testing, transparent box testing, and structural testing

Quality of Security Testing



False
Positives

False
Negatives

Incomplete

Reasons for false negatives



The Tester



The Objectives



Test Coverage

NIST 800-115

NIST 800-115 describes security assessments and has four phases:

Planning

Discovery

Attack

Reporting

NIST 800-115

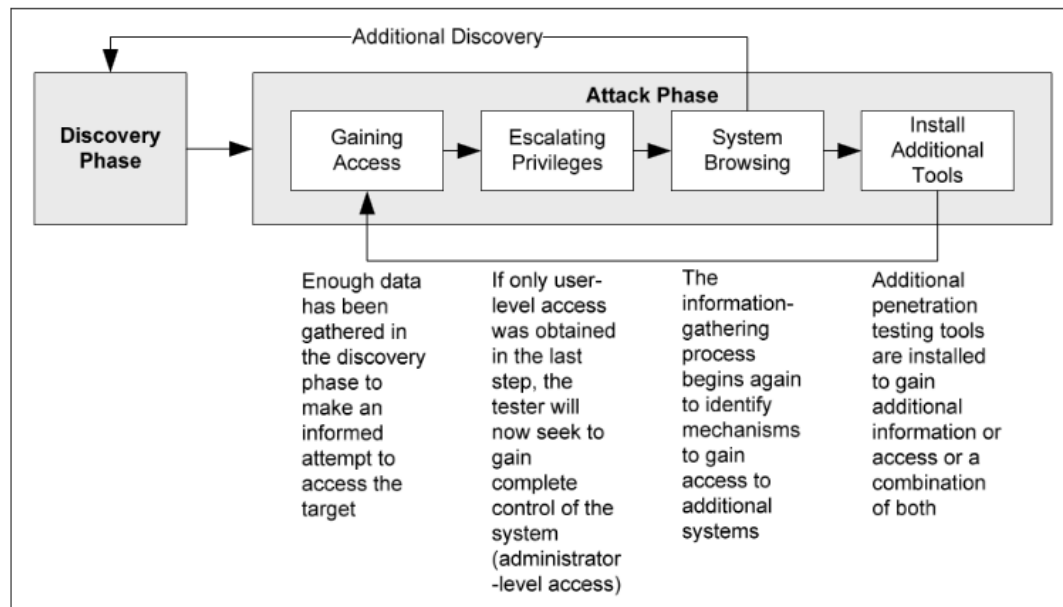


Figure 5-2. Attack Phase Steps with Loopback to Discovery Phase

Image from
<http://csrc.nist.gov/publications/nistpubs/800-115/SP800-115.pdf>

NIST 800-115



Implement a repeatable and documented assessment methodology.



Determine the objectives of each security assessment, and tailor the approach accordingly.



Analyze findings, and develop risk mitigation techniques to address weaknesses.

NIST 800-115

Image from
<http://csrc.nist.gov/publications/nistpublications/800-115/SP800-115.pdf>

Security Testing Technique	Security Testing Tool
Review	
Network Sniffing	Dsniff, Ettercap, Kismet, MailSnarf, Msgsnarf, Ntop, Phoss, SinFP, SMB Sniffer, and Wireshark
File Integrity Checking	Autopsy, Foremost, RootkitHunter, and Sleuthkit
Target Identification and Analysis	
Application Security Testing	CIRT Fuzzer, Fuzzer 1.2, NetSed, Paros Proxy, and Peach
Network Discovery	Autonomous System Scanner, Ettercap, Firewalk, Netdiscover, Netenum, Netmask, Nmap, P0f, Tctrace, and Umit
Network Port and Service Identification	Amap, AutoScan, Netdiscover, Nmap, P0f, Umit, and UnicornScan
Vulnerability Scanning	Firewalk, GFI LANguard, Hydra, Metasploit, Nmap, Paros Proxy, Snort, and SuperScan
Wireless Scanning	Airsnarf, Airtight, BdAddr, Bluesnarfer, Btscanner, FakeAP, GFI LANguard, Kismet, and WifiTAP
Target Vulnerability Validation	
Password Cracking	Hydra, John the Ripper, RainbowCrack, Rcrack, SIPcrack, SIPdump, TFTP-Brute, THC PPTP, VNCrack, and WebCrack
Remote Access Testing	IKEProbe, IKE-Scan, PSK-Crack, and VNC_bypauth
Penetration Testing	Driftnet, Dsniff, Ettercap, Kismet, Metasploit, Nmap, Ntop, SinFP, SMB Sniffer, and Wireshark

**National Security
Agency (NSA)
Information
Assessment
Methodology (IAM)**

Pre-Assessment

On-site

Post Assessment

NSA-IAM Overview



Pre-Assessment

- Determine and manage the customer's expectations
- Gain an understanding of the organization's information criticality
- Determine customer's goals and objectives
- Determine the system boundaries
- Coordinate with customer
- Request documentation



On-Site Assessment

- Conduct opening meeting
- Gather and validate system information (via interview, system demonstration, and document review)
- Analyze assessment information
- Develop initial recommendations
- Present out-brief



Post-Assessment

- Additional review of documentation
- Additional expertise (get help understanding what you learned)
- Report coordination (and writing)
- See also
<http://www.isaca.org/Journal/archives/2007/Volume-2/Documents/jopdf0702-info-security-request.pdf>

IAM

Information Criticality matrix

System Criticality matrices

Baseline INFOSEC evaluation areas

Technical Assessment Plan (TAP)

<http://www.sans.org/reading-room/whitepapers/auditing/application-nsa-infosec-assessment-methodology-1045>

<http://systemexperts.com/media/pdf/NSAIAM.pdf>

PCI Penetration Testing standard

Scope

Qualifications of a Penetration tester

Penetration Testing Components

Methodology

Pre-engagement

Pre-engagement includes scoping, documentations (network diagram, cardholder data flow diagram, etc.), rules of engagement, success criteria, review of past issues.

The actual penetration test

Post-Engagement

Remediation best practices, retest vulnerabilities, reporting and documentation standards.

https://www.pcisecuritystandards.org/documents/Penetration_Testing_Guidance_March_2015.pdf

PCI Penetration Testing standard

-  Build and Maintain a Secure Network
 -  Protect Cardholder Data
 -  Maintain a vulnerability management program
 -  Implement strong access control measures
 -  Regularly monitor and test networks
 -  Maintain a vulnerability management program
-

PCI Penetration Testing standard

	Vulnerability Scan	Penetration Test
Purpose	Identify, rank, and report vulnerabilities that, if exploited, may result in an intentional or unintentional compromise of a system.	Identify ways to exploit vulnerabilities to circumvent or defeat the security features of system components.
When	At least quarterly or after significant changes.	At least annually and upon significant changes. (Refer to Section 2.6 of this document for information on significant changes.)
How	Typically a variety of automated tools combined with manual verification of identified issues.	A manual process that may include the use of vulnerability scanning or other automated tools, resulting in a comprehensive report.

Chart from Penetration Testing Guidance

https://www.pcisecuritystandards.org/documents/Penetration_Testing_Guidance_March_2015.pdf

PCI Highlights

Pre-Engagement

- Scope
- Documentation
- Rules of engagement
- Environment
- Success Criteria
- Past vulnerability scans

Actual Test

- Application Layer
- Network Layer
- Segmentation
- How to handle card holder data
- Post-exploitation

Post-Engagement

- Remediation
 - Retesting identified vulnerabilities
 - Cleaning Up
 - Reporting
-

Cyber Kill Chain

Phases of the Intrusion Kill Chain



CEH Lifecycle

Gain Access

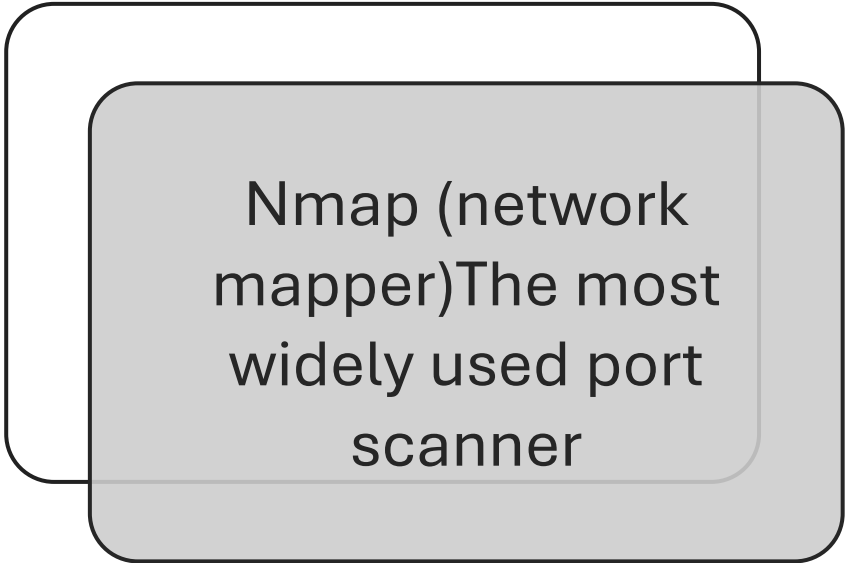
Escalate Privileges

Execute Applications

Hide Files

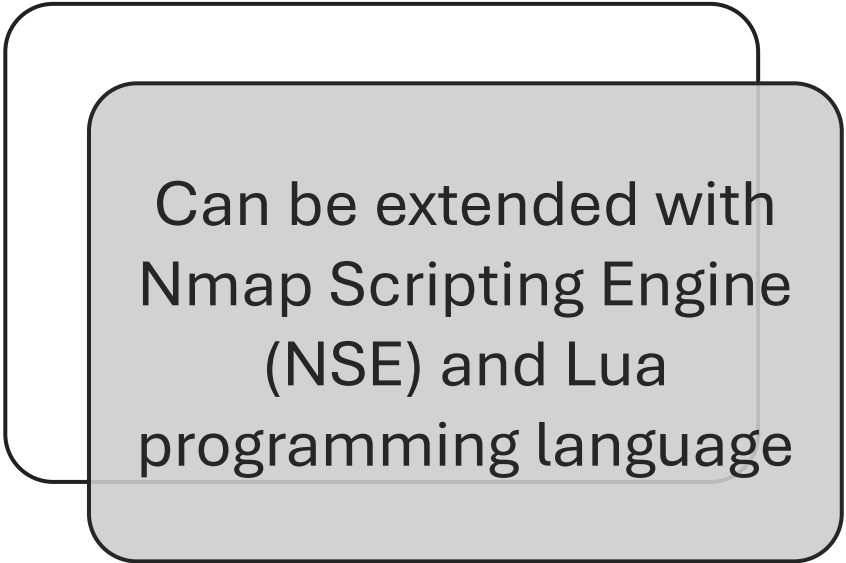
Covering Tracks

Nmap



Nmap (network mapper)The most widely used port scanner

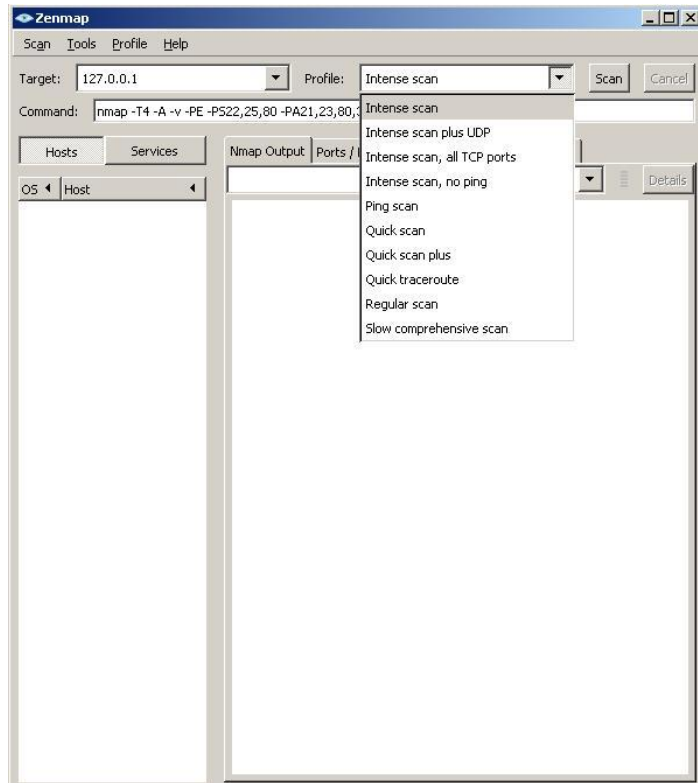
The diagram consists of two overlapping rounded rectangles. The front rectangle is light gray and contains the text 'Nmap (network mapper)The most widely used port scanner'. The back rectangle is white and is slightly offset to the top-left.



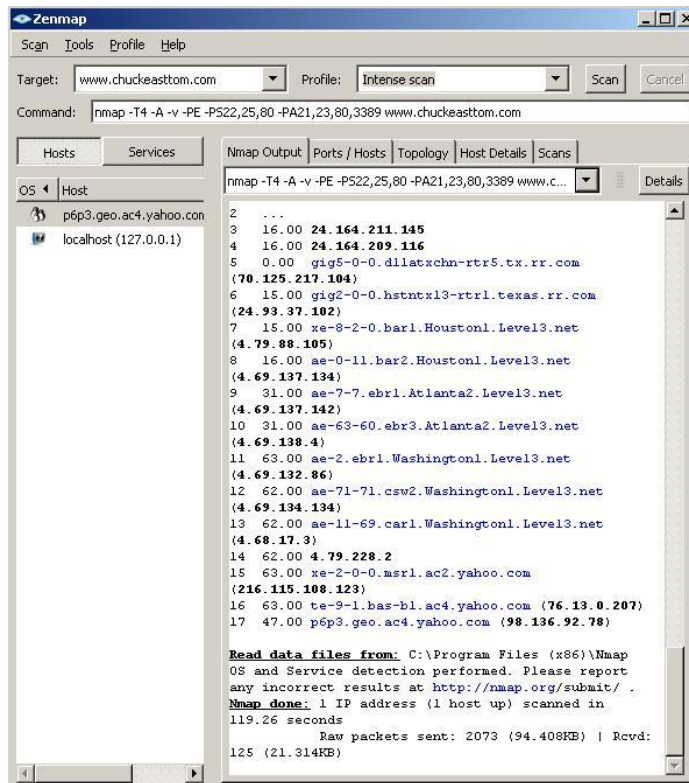
Can be extended with Nmap Scripting Engine (NSE) and Lua programming language

The diagram consists of two overlapping rounded rectangles. The front rectangle is light gray and contains the text 'Can be extended with Nmap Scripting Engine (NSE) and Lua programming language'. The back rectangle is white and is slightly offset to the top-left.

Nmap (ZenMap the GUI version)



Nmap Continued



NMAP Flags

- ▶ -O detects operating system
- ▶ -sP is a ping scan
- ▶ -sT TCP connect scan
- ▶ -sS SYN scan
- ▶ -sF FIN scan
- ▶ -sX XMAS Tree scan
- ▶ -sN NULL Scan
- ▶ -sU UDP scan
- ▶ -sO Protocol Scan
- ▶ -sA ACK Scan
- ▶ -sW Windows Scan
- ▶ -sR RPC scan
- ▶ -sL List/DNS scan
- ▶ -sI Idle scan
- -Po Don't Ping
- -PT TCP ping
- -PS SYN ping
- -PI ICMP Ping
- -PB TCP and ICMP ping
- -PM ICMP netmask
- -oN Normal Output
- -oX XML output
- -oG Greppable output
- -oA all output
- -T timing
 - -T0 paranoid
 - -T1 Sneaking
 - -T2 Polite
 - -T3 Normal
 - -T4 Aggressive
 - -T5 Insane

NMAP -continued

- Examples
 - Basic syn scan : `nmap -sS 192.168.0.1`
 - Basic Null scan : `nmap -sN 192.168.0.1`
 - Basic protocol scan: `nmap -sO 192.168.0.1`
 - Nmap stack fingerprinting: `Nmap -O -p80<hosts>`
 - Scan lower 1024 UDP Nmap `-sU -p 1-1024 <hosts>`
 - Scan a range of IP `nmap 192.168.1.*`
 - Scan IP v 6 `nmap -6 IPv6-Address-Here`
 - Show only open ports `nmap --open 192.168.1.1`

Types of Scans



Ping scan: This scan simply sends a ping to the target port. Many network administrators block incoming ICMP packets for the purpose of stopping ping scans.



Connect scan: This is the most reliable scan, but also the most likely to be detected. With this type of scan a complete connection is made with the target system.



SYN scan: This scan is very stealthy. Most systems accept SYN (Synchronize) requests. This scan is similar to the SYN flood DoS attack described in Chapter 4, “Denial of Service Attacks.” In this scan you send a SYN packet but never respond when the system sends a SYN/ACK. However, unlike the DoS SYN flood, you only send one packet per port. This is also called the *half-open scan*.



FIN scan: This scan has the FIN flag, or connection finished flag set. This is also not an unusual packet for systems to receive, so it is considered stealthy.

FLAGS

- In TCP communications there are 8 flags; FIN, SYN, RST, PSH, ACK, URG, ECE, CWR. These flags have decimal numbers assigned to them:
- FIN = 1
- SYN = 2
- RST = 4
- PSH = 8
- ACK = 16
- URG = 32
- ECE = 64
- CWR = 128

Packet Flags

- CWR (1 bit) – Congestion Window Reduced (CWR) flag is set by the sending host to indicate that it received a TCP segment with the ECE flag set and had responded in congestion control mechanism.
 - ECE (1 bit) – ECN-Echo indicates
 - If the SYN flag is set, that the TCP peer is ECN capable.
 - If the SYN flag is clear, that a packet with Congestion Experienced flag in IP header set is received during normal transmission.
 - URG (1 bit) – indicates that the Urgent pointer field is significant
 - ACK (1 bit) – indicates that the Acknowledgment field is significant. All packets after the initial SYN packet sent by the client should have this flag set.
 - PSH (1 bit) – Push function
 - RST (1 bit) – Reset the connection
 - SYN (1 bit) – Synchronize sequence numbers. Only the first packet sent from each end should have this flag set. Some other flags change meaning based on this flag, and some are only valid for when it is set, and others when it is clear.
 - FIN (1 bit) – No more data from sender
- Note: SYN, FIN, ACK, URG, and RST are the flags most commonly used in port scanning.

Scan Responses

FIN

- Port closed response is RST

- Port open no response

- Windows PCs do not comply with RFC 793; therefore, they do not provide accurate results with this type of scan

XMAS

- Port closed response is RST

- Port open no response

SYN (considered stealthy, also called half open scan)

- Port closed RST

- IF the port is open, the target responds with a SYN-ACK. If it is closed, it responds with an RST

NULL scan

- All flags off

- RFC 793 states that if a TCP segment arrives with no flags set, the receiving host should drop the segment and send an RST

ACK Scan

- If a port is filtered on a firewall, nothing comes back. If a port is unfiltered an RST is sent back.

Cisco has a great paper on port scanning

<http://www.ciscopress.com/articles/article.asp?p=469623&seqNum=3>

hping

- <http://www.hping.org/>
- Send TCP SYN packets to port 0 on host xyz.com
- hping xyz.com -S -V
- Send TCP SYN packets every 100,000 microseconds to port 443
- hping xyz.com -S -p 443 -i u100000

- -F -fin set FIN flag
-S -syn set SYN flag
-R -rst set RST flag
-P -push set PUSH flag
-A -ack set ACK flag
-U -urg set URG flag
-X -xmas set X unused flag (0x40)
-Y -ymas set Y unused flag (0x80)

Netdiscover

*Just netdiscover
without any flags*

```
root@kali: ~
File Edit View Search Terminal Help
Currently scanning: 192.168.20.0/16 | Screen View: Unique Hosts
32 Captured ARP Req/Rep packets, from 28 hosts. Total size: 1986
```

IP	At MAC Address	Count	Len	MAC Vendor / Hostname
192.168.1.1	60:38:e0:6f:c9:d3	2	120	Belkin International Inc.
192.168.1.19	b8:3a:9d:12:7b:52	1	60	Alarm.com
192.168.1.70	b8:3a:9d:12:7c:ea	1	60	Alarm.com
192.168.1.108	b8:3a:9d:12:79:c6	1	60	Alarm.com
192.168.1.18	60:01:94:95:aa:6b	1	60	Espressif Inc.
192.168.1.100	c8:3a:6b:14:51:44	1	60	Roku, Inc
192.168.1.163	b8:3a:9d:12:79:fd	1	60	Alarm.com
192.168.1.41	b8:3a:9d:20:3f:fa	2	120	Alarm.com
192.168.1.182	d8:97:ba:4f:e2:a8	1	60	PEGATRON CORPORATION
192.168.1.189	00:90:a9:da:7b:93	1	60	WESTERN DIGITAL
192.168.1.190	b0:83:fe:b8:83:84	2	120	Dell Inc.
192.168.1.46	60:01:94:a0:59:8e	1	60	Espressif Inc.
192.168.1.52	e4:f0:42:32:97:e9	1	60	Google, Inc.
192.168.1.206	b8:3a:9d:12:7d:15	1	60	Alarm.com
192.168.1.207	b8:3a:9d:12:7d:16	1	60	Alarm.com
192.168.1.210	b8:3a:9d:01:11:76	1	60	Alarm.com
192.168.1.212	b8:3a:9d:12:7b:34	1	60	Alarm.com

Netdiscover

Flags

```
root@kali: ~  
File Edit View Search Terminal Help  
Netdiscover 0.3-pre-beta7 [Active/passive arp reconnaissance tool]  
Written by: Jaime Penalba <jpenalbae@gmail.com>  
  
Usage: netdiscover [-i device] [-r range | -l file | -p] [-m file] [-s time] [-n  
node] [-c count] [-f] [-d] [-S] [-P] [-c]  
-i device: your network device  
-r range: scan a given range instead of auto scan. 192.168.6.0/24,/16,/8  
-l file: scan the list of ranges contained into the given file  
-p passive mode: do not send anything, only sniff  
-m file: scan the list of known MACs and host names  
-F filter: Customize pcap filter expression (default: "arp")  
-s time: time to sleep between each arp request (milliseconds)  
-n node: last ip octet used for scanning (from 2 to 253)  
-c count: number of times to send each arp request (for nets with packet loss)  
-f enable fastmode scan, saves a lot of time, recommended for auto  
-d ignore home config files for autoscan and fast mode  
-S enable sleep time suppression between each request (hardcore mode)  
-P print results in a format suitable for parsing by another program  
-N Do not print header. Only valid when -P is enabled.  
-L in parsable output mode (-P), continue listening after the active scan is completed  
  
If -r, -l or -p are not enabled, netdiscover will scan for common lan addresses.  
root@kali:~#
```

Netdiscover flags

- i device: your network device
- r range: scan a given range instead of auto scan. 192.168.6.0/24,/16,/8
- l file: scan the list of ranges contained into the given file
- p passive mode: do not send anything, only sniff
- m file: scan the list of known MACs and host names
- F filter: Customize pcap filter expression (default: "arp")
- s time: time to sleep between each arp request (milliseconds)
- n node: last ip octet used for scanning (from 2 to 253)
- c count: number of times to send each arp request (for nets with packet loss)
- f enable fastmode scan, saves a lot of time, recommended for auto
- d ignore home config files for autoscan and fast mode
- S enable sleep time suppression between each request (hardcore mode)
- P print results in a format suitable for parsing by another program
- N Do not print header. Only valid when -P is enabled.
- L in parsable output mode (-P), continue listening after the active scan is completed

Netdiscover examples



```
netdiscover -i eth0 -r  
192.168.1.1/24
```



```
Netdiscover -r  
192.168.1.1/24
```

Pre-test activities



RISK ASSESSMENT



SETTING CLEAR
GOALS



DEFINING SCOPE



KNOW WHAT IS
AND IS NOT BEING
COVERED



PLAN TOOLS AND
TECHNIQUES

Tools

- NetScan Tools Pro. Used for discovering network devices including IPv4/IPv6 address, host name, email address, etc.
- Other scanners
 - SuperScan
 - Network Inventory Explorer
 - Advanced Port Scanner
 - CurrPorts
 - MegaPing
 - Net Tools
 - IP-Tools
- Mobile Scanners
 - Ip Network Scanner
 - Fing
 - Umit Network Scanner
 - PortDroid
 - Pamn IP Scanner
 - Network Discovery

▶ Mobile Scanners

- ▶ Ip Network Scanner
- ▶ Fing
- ▶ Umit Network Scanner
- ▶ PortDroid
- ▶ Pamn IP Scanner
- ▶ Network Discovery



Port Scanning Counter Measures

Configure	Configure firewall and IDS to block probes
Block	Block ICMP
Perform	Perform your own scanning
Filter	Filter at routers
Update	Update router, IDS, and Firewall

Netcat

A popular tool that is widely used by network administrators, hackers, and others.

You can get it

<http://netcat.sourceforge.net/>

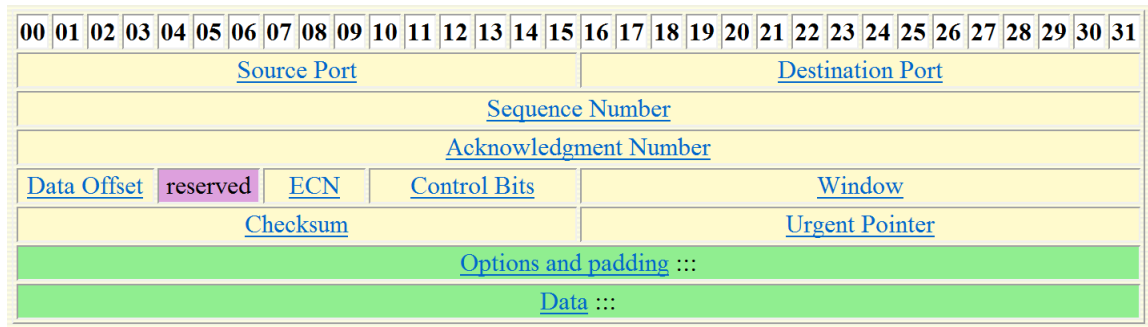
You can get netcat for Windows here

<http://joncraton.org/blog/netcat-for-windows>

Netcat examples

- Listen on a given port
 - `nc -l 3333`
- Connect to listening port
 - `nc 132.22.15.43 3333`
- Connect to a mail server
 - `nc mail.server.net 25`
- Turn Netcat into a proxy server
 - `nc -l 3333 | nc www.google.com 80`

Packet Structure - TCP Headers



00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31		
<u>Version</u>				<u>IHL</u>			<u>Differentiated Services</u>									<u>Total length</u>																	
<u>Identification</u>																<u>Flags</u>			<u>Fragment offset</u>														
<u>TTL</u>								<u>Protocol</u>								<u>Header checksum</u>																	
<u>Source IP address</u>																																	
<u>Destination IP address</u>																																	
<u>Options and padding :::</u>																																	

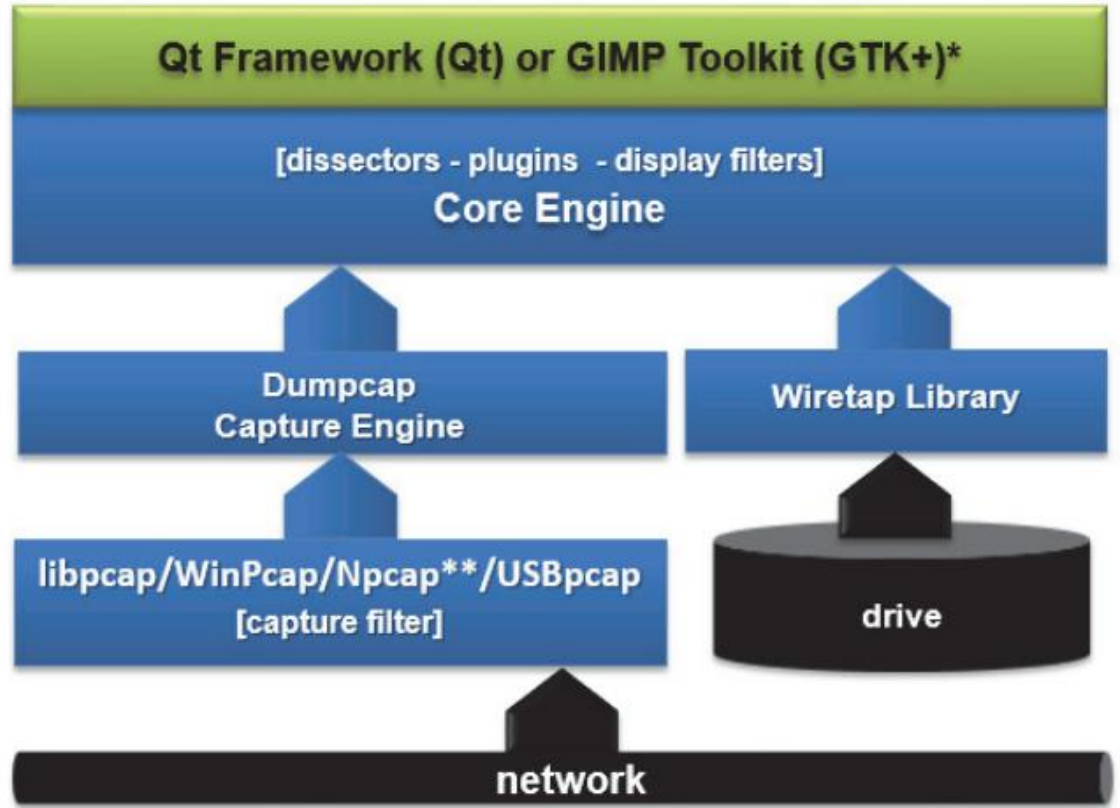
Preamble	Start of frame delimiter	MAC destination	MAC source	802.1Q tag (optional)	Ethertype or length	Payload	Frame check sequence (32-bit CRC)	Interframe gap	
7 octets of 10101010	1 octet of 10101011	6 octets	6 octets	(4 octets)	2 octets	46–1500 octets	4 octets	12 octets	
		64–1522 octets							
		72–1530 octets							
		84–1542 octets							

Wireshark

- Wireshark is one of the most widely known network packet sniffers. Often a penetration tester can learn a great deal from simply sniffing the network traffic on a target network.

Wireshark provides a convenient graphical user interface (GUI) for examining network traffic. It is a free download, which you can get at <https://www.wireshark.org/>. tool can be downloaded for Windows or Macintosh

Wireshark



Selecting something to capture

If you don't see traffic, it is not live.

Capture

...using this filter:

	VirtualBox Host-Only Network #4	
	Ethernet 4	
	Wi-Fi	
	Local Area Connection* 3	
	Ethernet 2	
	Ethernet	
	USBPcap1	
	USBPcap2	

Analyzing network traffic

*Wi-Fi [Wireshark 1.12.4 (v1.12.4-0-gb4861da from master-1.12)]

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
2	0.307611000	192.168.1.100	239.255.255.250	SSDP	380	NOTIFY * HTTP/1.1
3	0.307736000	192.168.1.100	239.255.255.250	SSDP	389	NOTIFY * HTTP/1.1
4	0.307768000	192.168.1.100	239.255.255.250	SSDP	432	NOTIFY * HTTP/1.1
5	0.409543000	ArpIsro_f8:24:ca	Broadcast	ARP	60	who has 192.168.1.1? Tell 192.168.1.101
6	0.409572000	192.168.1.100	239.255.255.250	SSDP	444	NOTIFY * HTTP/1.1
7	0.409622000	192.168.1.100	239.255.255.250	SSDP	446	NOTIFY * HTTP/1.1
8	0.884307000	192.168.1.153	239.255.255.250	SSDP	175	M-SEARCH * HTTP/1.1
9	0.888416000	192.168.1.1	192.168.1.153	SSDP	487	HTTP/1.1 200 OK
10	1.896522000	192.168.1.153	192.168.1.255	NBNS	92	Name query NB wPAD<00>
11	2.355141000	192.168.1.100	239.255.255.250	SSDP	380	NOTIFY * HTTP/1.1
12	2.355247000	192.168.1.100	239.255.255.250	SSDP	389	NOTIFY * HTTP/1.1
13	2.355271000	192.168.1.100	239.255.255.250	SSDP	432	NOTIFY * HTTP/1.1
14	2.458012000	192.168.1.100	239.255.255.250	SSDP	444	NOTIFY * HTTP/1.1
15	2.458124000	192.168.1.100	239.255.255.250	SSDP	446	NOTIFY * HTTP/1.1
16	2.580735000	192.168.1.153	98.138.227.170	TCP	66	21231-995 [SYN] Seq=0 win=8192 Len=0 MSS=14
17	2.622895000	98.138.227.170	192.168.1.153	TCP	66	995-21231 [SYN, ACK] Seq=0 Ack=1 win=14600
18	2.623067000	192.168.1.153	98.138.227.170	TCP	54	21231-995 [ACK] Seq=1 Ack=1 win=65536 Len=0
19	2.623330000	192.168.1.153	98.138.227.170	TLSv1.2	427	Client Hello
20	2.650931000	192.168.1.153	192.168.1.255	NBNS	92	Name query NB wPAD<00>
21	2.669520000	98.138.227.170	192.168.1.153	TCP	54	995-21231 [ACK] Seq=1 Ack=374 win=15872 Len=0
22	2.674261000	98.138.227.170	192.168.1.153	TLSv1.2	1514	Server Hello
23	2.674317000	98.138.227.170	192.168.1.153	TCP	1514	[TCP segment of a reassembled PDU]
24	2.674357000	192.168.1.153	98.138.227.170	TCP	54	21231-995 [ACK] Seq=374 Ack=2921 win=65536
25	2.674437000	98.138.227.170	192.168.1.153	TCP	1514	[TCP segment of a reassembled PDU]
26	2.674458000	98.138.227.170	192.168.1.153	TLSv1.2	737	Certificate
27	2.674483000	192.168.1.153	98.138.227.170	TCP	54	21231-995 [ACK] Seq=374 Ack=5064 win=65536
28	2.681258000	192.168.1.153	98.138.227.170	TLSv1.2	220	client key exchange, change cipher spec, enc
29	2.725567000	98.138.227.170	192.168.1.153	TLSv1.2	352	New Session Ticket, change cipher spec, enc
30	2.746904000	192.168.1.153	98.138.227.170	TCP	54	21231-995 [ACK] Seq=540 Ack=5362 win=65280
31	2.789218000	98.138.227.170	192.168.1.153	TLSv1.2	187	Application Data
32	2.789520000	192.168.1.153	98.138.227.170	TLSv1.2	123	Application Data
33	2.831771000	98.138.227.170	192.168.1.153	TLSv1.2	267	Application Data
34	2.832151000	192.168.1.153	98.138.227.170	TLSv1.2	139	Application Data
35	2.873963000	98.138.227.170	192.168.1.153	TLSv1.2	139	Application Data
36	2.874168000	192.168.1.153	98.138.227.170	TLSv1.2	139	Application Data

< Frame 1: 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface 0

Ethernet II, Src: Seikoeps_bb:27:b8 (00:26:ab:bb:27:b8), Dst: Broadcast (ff:ff:ff:ff:ff:ff)

```
0000 ff ff ff ff ff 00 26 ab bb 27 b8 08 06 00 01 .....& ..|....
0010 08 00 06 04 00 01 00 26 ab bb 27 b8 c0 a8 01 9e .....& ..|....
0020 00 00 00 00 00 00 c0 a8 01 9e f8 3d 1c 40 f8 3d .....& ..|....
0030 1c 40 65 74 68 31 01 00 00 00 dc 52 37 00 b7 d0 .@ethl...R7...
```

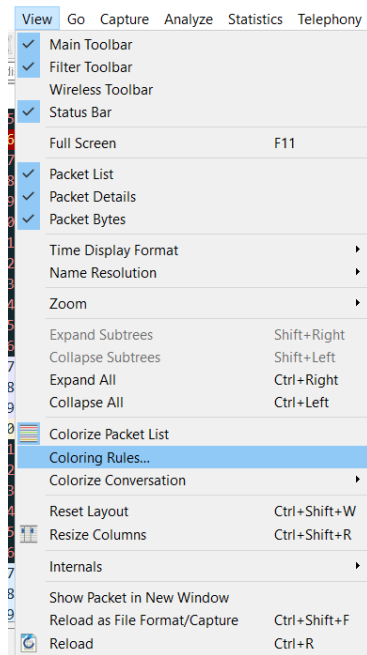
What do the colors mean

- Wireshark uses colors to help you identify the types of traffic at a glance. By default, green is TCP traffic, dark blue is DNS traffic, light blue is UDP traffic, and black identifies TCP packets with problems

12041	55.443042	52.9.36.43	172.20.0.49	TCP	66 [TCP Keep-Alive ACK] 443 → 1074 [ACK] Seq=6263 Ack=28982 Win=108032 Len=0
12042	55.450029	52.70.175.132	172.20.0.49	TCP	66 [TCP Keep-Alive ACK] 443 → 1068 [ACK] Seq=7349 Ack=5709 Win=39424 Len=0
12043	55.452855	54.186.208.153	172.20.0.49	TCP	60 [TCP Keep-Alive ACK] 443 → 1037 [ACK] Seq=3276 Ack=1579 Win=31790 Len=0
12044	55.507108	52.70.175.132	172.20.0.49	TCP	66 [TCP Keep-Alive ACK] 443 → 1072 [ACK] Seq=1322 Ack=2641 Win=32256 Len=0
12045	55.643196	172.20.0.49	54.186.208.153	TCP	55 [TCP Keep-Alive] 1038 → 443 [ACK] Seq=1571 Ack=3276 Win=65280 Len=1
12046	55.678902	54.186.208.153	172.20.0.49	TCP	66 [TCP Keep-Alive ACK] 443 → 1038 [ACK] Seq=3276 Ack=1572 Win=29952 Len=0
12047	56.321019	172.20.0.49	52.165.171.165	TLSv1.2	127 Application Data
12048	56.373976	52.165.171.165	172.20.0.49	TLSv1.2	179 Application Data
12049	56.414701	172.20.0.49	52.165.171.165	TCP	54 24299 → 443 [ACK] Seq=74 Ack=126 Win=257 Len=0
12050	56.853742	Apple_be:1c:c5	Broadcast	ARP	56 Gratuitous ARP for 172.20.0.56 (Request)
12051	58.011802	172.20.0.49	184.51.252.117	TCP	55 [TCP Keep-Alive] 1055 → 443 [ACK] Seq=13406 Ack=3533 Win=65280 Len=1[Reas]
12052	58.075770	184.51.252.117	172.20.0.49	TCP	66 [TCP Keep-Alive ACK] 443 → 1055 [ACK] Seq=3533 Ack=13407 Win=72448 Len=0
12053	58.303198	172.20.0.49	34.196.201.187	TCP	55 [TCP Keep-Alive] 1049 → 443 [ACK] Seq=2152 Ack=5951 Win=64240 Len=1[Reas]
12054	58.382502	34.196.201.187	172.20.0.49	TCP	60 [TCP Keep-Alive ACK] 443 → 1049 [ACK] Seq=5951 Ack=2153 Win=32120 Len=0
12055	58.517667	172.20.0.49	209.73.190.75	TCP	55 [TCP Keep-Alive] 1085 → 443 [ACK] Seq=9593 Ack=3796666 Win=391168 Len=1
12056	58.527582	209.73.190.75	172.20.0.49	TCP	66 [TCP Keep-Alive ACK] 443 → 1085 [ACK] Seq=3796666 Ack=9594 Win=34048 Len=0
12057	58.736214	172.20.0.49	64.134.255.2	DNS	87 Standard query 0x63b5 A shavar.services.mozilla.com
12058	58.768553	172.20.0.49	64.134.255.10	DNS	87 Standard query 0x63b5 A shavar.services.mozilla.com
12059	58.777273	64.134.255.2	172.20.0.49	DNS	529 Standard query response 0x63b5 A shavar.services.mozilla.com CNAME shava

What do the colors mean

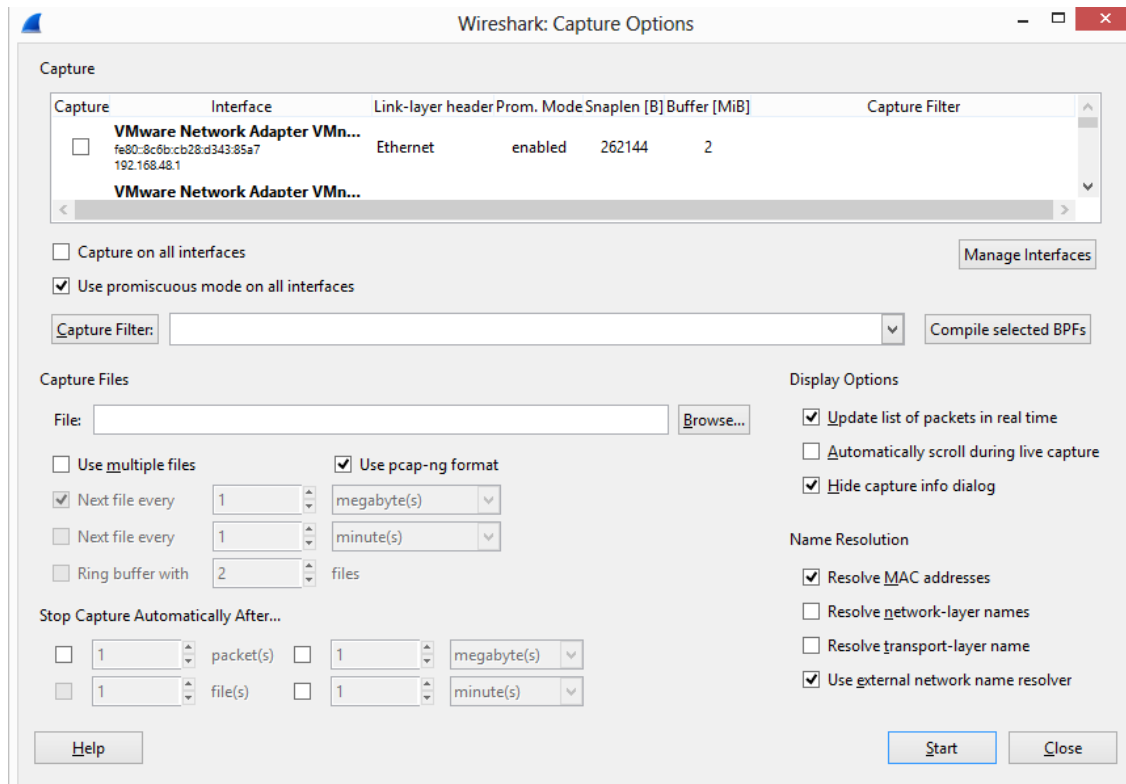
Go to View > Coloring Rules



What do the colors mean

Name	Filter
☒ Bad TCP	tcp.analysis.flags && !tcp.analysis.window_update
☒ HSRP State Change	hsrp.state != 8 && hsrp.state != 16
☒ Spanning Tree Topology Change	stp.type == 0x80
☒ OSPF State Change	ospf.msg != 1
☒ ICMP errors	icmp.type eq 3 icmp.type eq 4 icmp.type eq 5 icmp.type eq 11 icmpv6.type eq 1 icmpv6.type eq 2 icmpv6.type eq 3 icmpv6.type eq 4
☒ ARP	arp
☒ ICMP	icmp icmpv6
☒ TCP RST	tcp.flags.reset eq 1
☒ SCTP ABORT	sctp.chunk_type eq ABORT
☒ TTL low or unexpected	(! ip.dst == 224.0.0.0/4 && ip.ttl < 5 && !pim && !ospf) (ip.dst == 224.0.0.0/24 && ip.dst != 224.0.0.251 && ip.ttl != 1 && !(vrrp carp))
☒ Checksum Errors	eth.fcs.status == "Bad" ip.checksum.status == "Bad" tcp.checksum.status == "Bad" udp.checksum.status == "Bad" sctp.checksum.status == "Bad" mstp.checksum.status == "Bad"
☒ SMB	smb nbss nbns nbipx ipxsap netbios
☒ HTTP	http tcp.port == 80 http2
☒ IPX	ipx spx
☒ DCE/RPC	dcerpc
☒ Routing	hsrp eigrp ospf bgp cdp vrrp carp gvrp igmp ismp
☒ TCP SYN/FIN	tcp.flags & 0x02 tcp.flags.fin == 1
☒ TCP	tcp
☒ UDP	udp
☒ Broadcast	eth[0] & 1

Wireshark Configuration



Wireshark Interface

Highlighted Packet

The image shows the Wireshark network protocol analyzer interface. The top pane displays a list of captured packets. Packet 29 is highlighted in blue and circled in red. A red arrow points from the text 'Highlighted Packet' to this packet. The packet list shows the following details for packet 29:

No.	Time	Source	Destination	Protocol	Length	Info
29	2.725567000	98.138.227.170	192.168.1.153	TLSv1.2	352	New Session Ticket, Change Cipher Spec, Enc

The bottom pane shows the detailed view of packet 29, which is a TLSv1.2 packet. The details are as follows:

- Frame 29: 352 bytes on wire (2816 bits), 352 bytes captured (2816 bits) on interface 0
- Ethernet II, Src: Verizon8_87:7c:8d (c8:a7:0a:87:7c:8d), Dst: IntelCor_1d:ac:d1 (c8:f7:33:1d:ac:d1)
- Internet Protocol Version 4, Src: 98.138.227.170 (98.138.227.170), Dst: 192.168.1.153 (192.168.1.153)
- Transmission Control Protocol, Src Port: 995 (995), Dst Port: 21231 (21231), Seq: 5064, Ack: 540, Len: 298
- Secure Sockets Layer

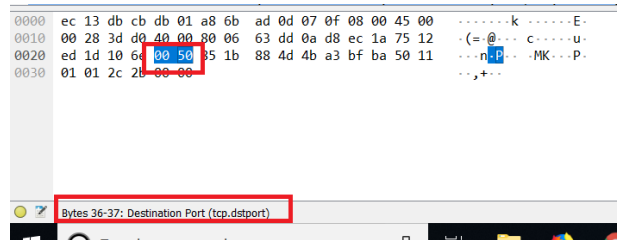
The packet bytes pane shows the raw data in hexadecimal and ASCII format:

```
0000 c8 f7 33 1d ac d1 c8 a7 0a 87 7c 8d 08 00 45 00  .3....E...
0010 01 52 34 fa 40 00 35 06 07 36 62 8a e3 aa c0 a8  .R4.@.5..6b...
0020 01 99 03 e3 52 ef 58 83 73 0c 26 8b e0 98 50 18  ...R.X.s.&...P.
0030 00 21 4d 54 00 00 16 03 03 00 ca 04 00 00 c6 00  .!MT.....
0040 00 01 2c 00 c0 62 d5 d5 df e7 b0 2d a7 fe b8 ef  ....b.....
0050 c8 15 31 75 c6 57 05 12 7b 43 cd 64 cd 48 48 02  7..u.v...l.d
```

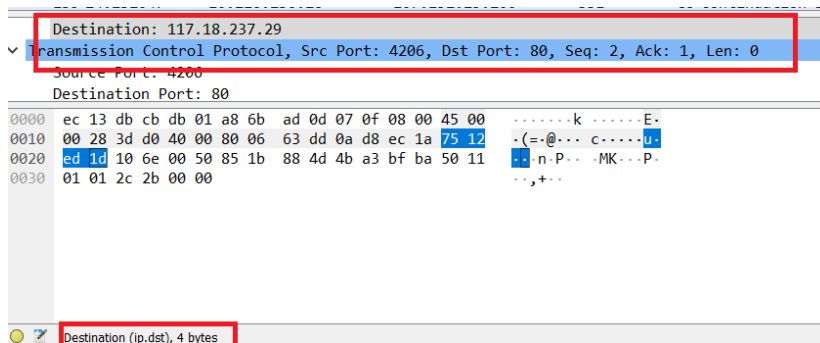
The status bar at the bottom indicates: File: "C:\Users\CHUCKE~1\AppData\Local\T... Packets: 55 - Displayed: 55 (100.0%) - Dropped: 0 (0.0%) Profile: Default

What are you looking at

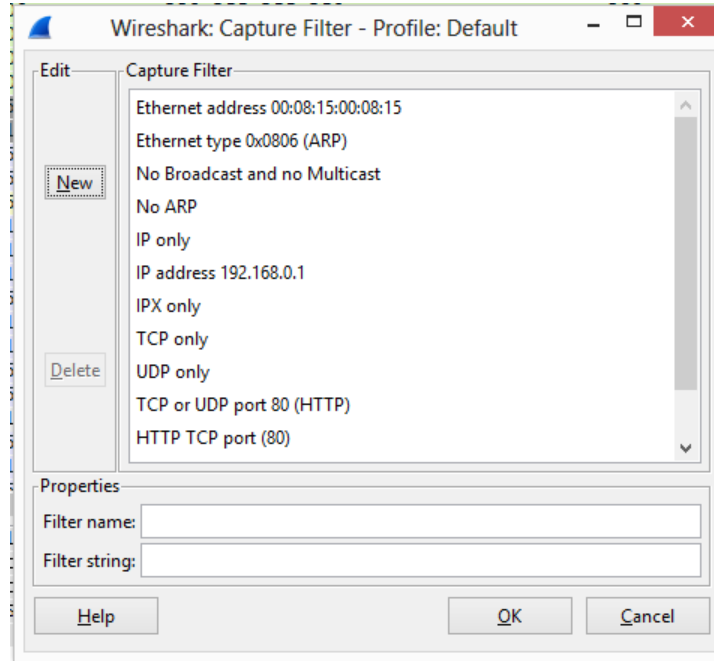
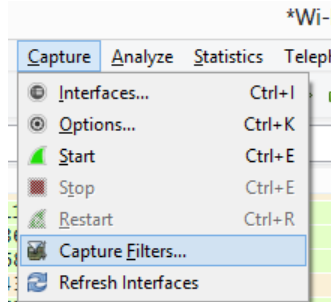
As you highlight items in the data pane, the content is described below



And if you click on it, then Details are shown above.



Capture Filter



Capture Filter examples

host 10.1.20.55

host 192.168.0.1 and host 10.1.20.55

tcp port http

ip

not broadcast not multicast

ether host 00:04:13:00:09:a3

Display Filters (Post-Filters)

Display filters (also called post-filters) only filter the view of what you are seeing. All packets in the capture still exist in the trace

Display filters use their own format and are much more powerful than capture filters

View Filter



Display Filter Examples

ip.src==10.2.21.00/24

ip.addr==192.168.1.20 && ip.addr==192.168.1.30

tcp.port==80 || tcp.port==443

!(ip.addr==192.168.1.20 && ip.addr==192.168.1.30)

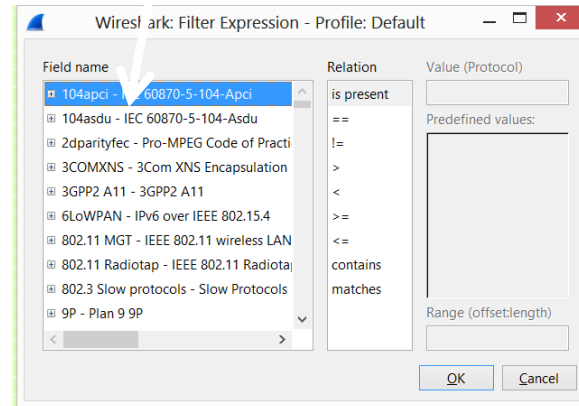
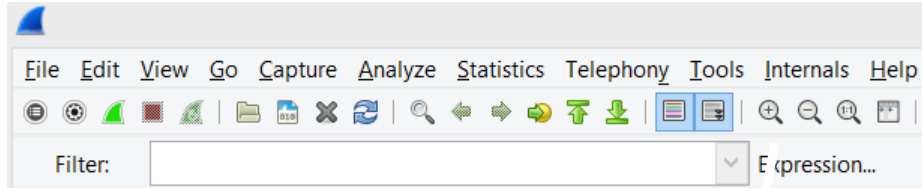
(ip.addr==192.168.1.20 && ip.addr==192.168.1.30) && (tcp.port==465|| tcp.port==139)

(ip.addr==192.168.1.20 && ip.addr==192.168.1.30) && (udp.port==80|| udp.port==443)

Display Filter

Syntax: **Protocol** . **String 1** . **String 2** **Comparison operator** **Value** **Logical Operations** **Other expression**

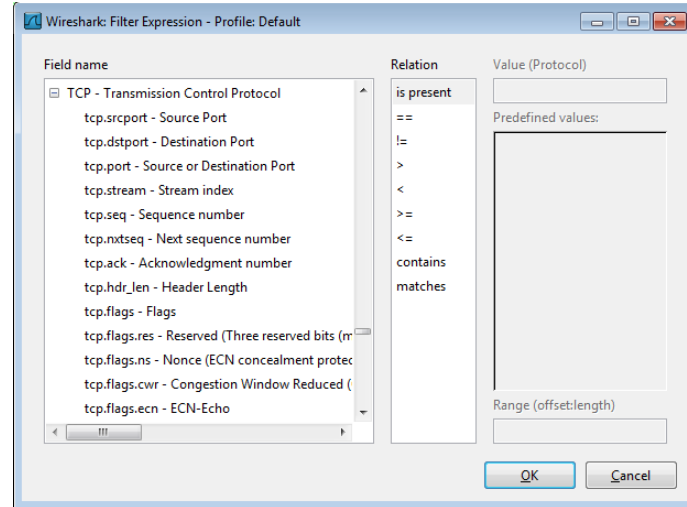
Example: ftp passive ip == 10.2.3.4 xor icmp.type



Display Filter

- String1, String2 (Optional settings):
 - Sub protocol categories inside the protocol.
 - Look for a protocol and then click on the "+" character.
 - Example:
 - **tcp.srcport == 80**
 - **tcp.flags == 2**
 - **SYN packet**
 - **Tcp.flags.syn==1**
 - **tcp.flags == 18**
 - **SYN/ACK**
- **Note of TCP**

U	A	P	R	S	F
R	C	S	S	Y	I
G	K	H	T	N	N



Display Filter Expressions

- `dns || icmp`
 - Display the DNS or ICMP traffic.
- `tcp.port == 443`
 - Display packets with TCP source or destination port 443.
- `tcp.flags`
 - Display packets having a TCP flags
- `tcp.flags.syn == 0x02`
 - Display packets with a TCP SYN flag.

Six **comparison** operators are available:

English format:	C like format:	Meaning:
eq	==	Equal
ne	!=	Not equal
gt	>	Greater than
lt	<	Less than
ge	>=	Greater or equal
le	<=	Less or equal

→ **Logical expressions:**

English format:	C like format:	Meaning:
and	&&	Logical AND
or		Logical OR
xor	^^	Logical XOR
not	!	Logical NOT

Filter: `tcp.port == 100`

Filter: `tcp.port = 100`

Correct syntax

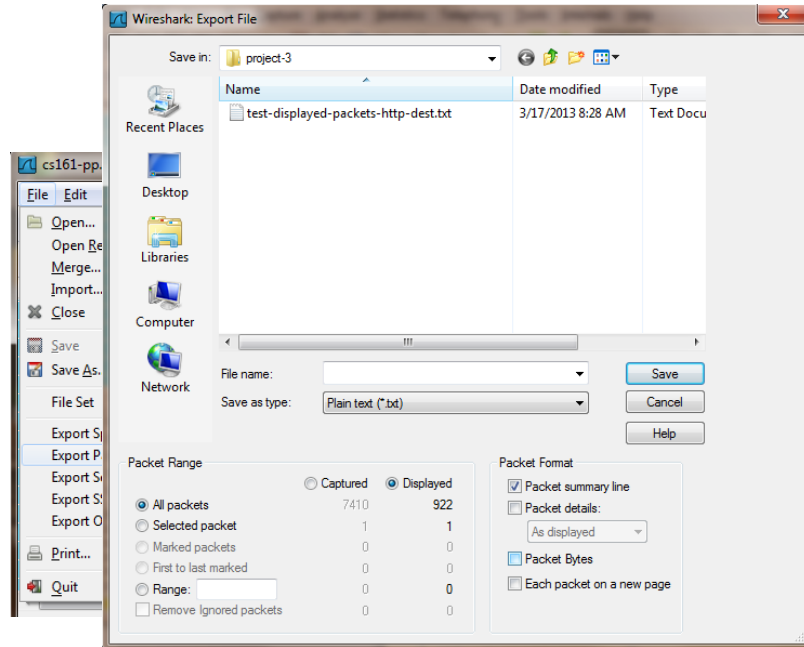
Wrong syntax

Save Filtered Packets After Using Display Filter

- We can also save all filtered packets in text file for further analysis
- Operation:

File→Export packet dissections
→as “plain text” file

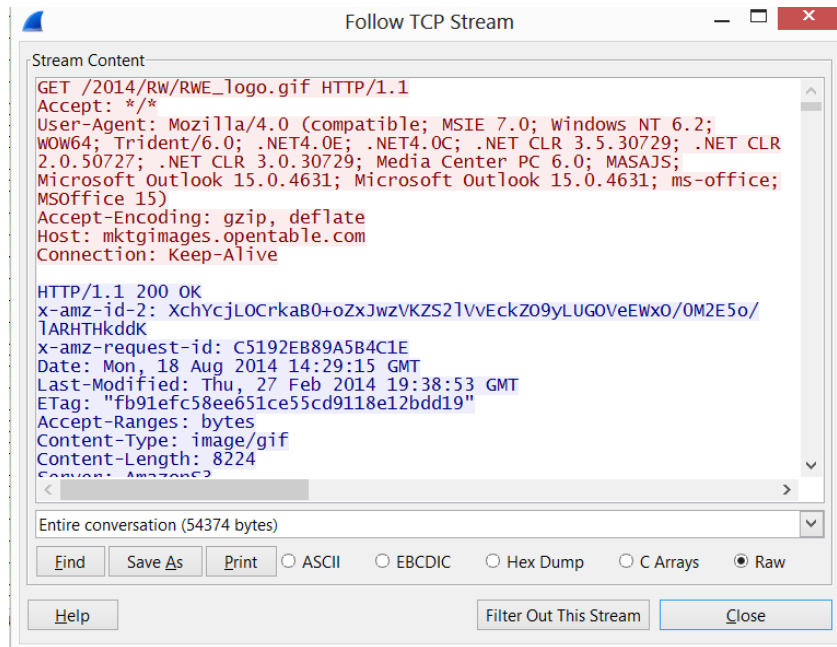
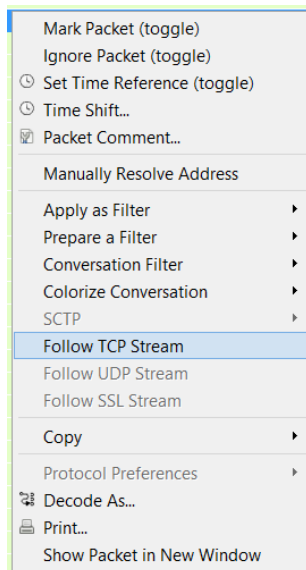
- 1). In “packet range” option, select “Displayed”
- 2). In choose “summary line” or “detail”



Follow TCP Stream

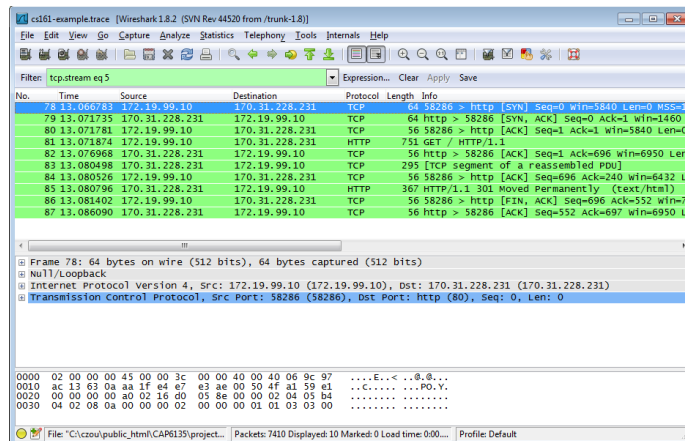
red - stuff you sent

blue - stuff you get

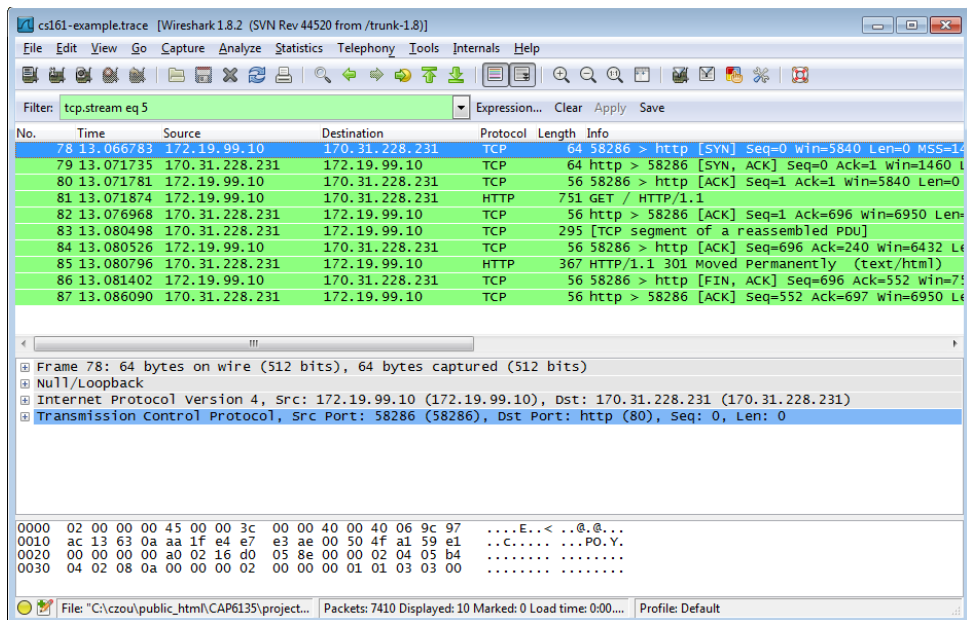


Filter out/in Single TCP Stream

- When click “filter out this TCP stream” in previous page’s box, new filter string will contain like:
 - `http and !(tcp.stream eq 5)`
- So, if you use “`tcp.stream eq 5`” as filter string, you keep this HTTP session



Filter out/in Single TCP Stream



- When click “filter out this TCP stream” in previous page’s box, new filter string will contain like:
 - http and !(tcp.stream eq 5)
- So, if you use “tcp.stream eq 5” as filter string, you keep this HTTP session

Expert Info

*Wi-Fi [Wireshark 1.12.4 (v1.12.4-0-gb4861da from master-1.12)]

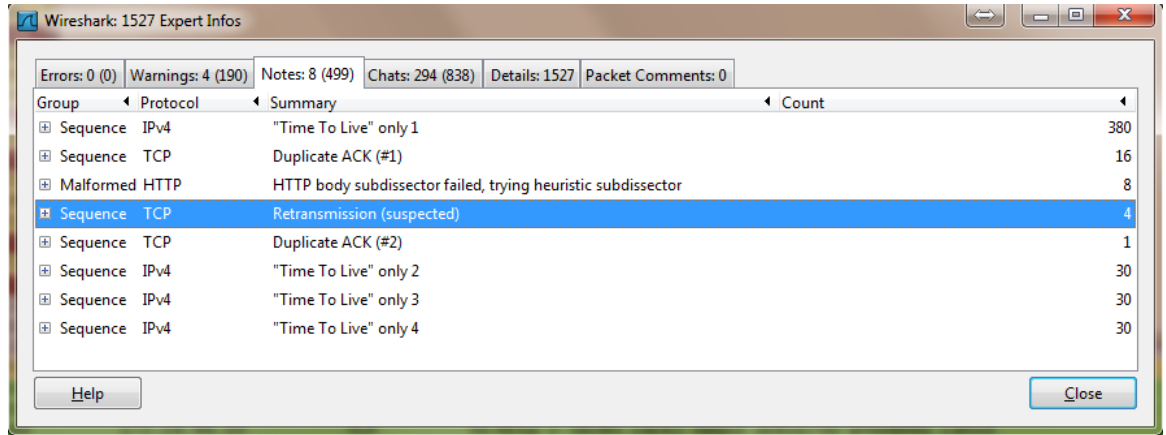
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter:

Display Filters...
Display Filter Macros...
Apply as Column
Apply as Filter
Prepare a Filter
Enabled Protocols... Shift+Ctrl+E
Decode As...
User Specified Decodes...
Follow TCP Stream
Follow UDP Stream
Follow SSL Stream
Expert Info
Conversation Filter

No.	Time	Source	Destination	Protocol	Length	Info
2	0.307611000	192.168.1.100	192.168.1.153	SSDP	380	NOTIFY * H
3	0.307736000	192.168.1.100	192.168.1.153	SSDP	389	NOTIFY * H
4	0.307768000	192.168.1.100	192.168.1.153	SSDP	432	NOTIFY * H
5	0.409543000	192.168.1.100	192.168.1.153	ARP	60	who has 19
6	0.409572000	192.168.1.100	192.168.1.153	SSDP	444	NOTIFY * H
7	0.409622000	192.168.1.100	192.168.1.153	SSDP	446	NOTIFY * H
8	0.884307000	192.168.1.100	192.168.1.153	SSDP	175	M-SEARCH *
9	0.888416000	192.168.1.100	192.168.1.153	SSDP	487	HTTP/1.1 2
10	1.896522000	192.168.1.100	192.168.1.153	NBNS	92	Name query
11	2.355141000	192.168.1.100	192.168.1.153	SSDP	380	NOTIFY * H
12	2.355247000	192.168.1.100	192.168.1.153	SSDP	389	NOTIFY * H
13	2.355271000	192.168.1.100	192.168.1.153	SSDP	432	NOTIFY * H
14	2.458012000	192.168.1.100	192.168.1.153	SSDP	444	NOTIFY * H
15	2.458124000	192.168.1.100	192.168.1.153	SSDP	446	NOTIFY * H
16	2.580735000	192.168.1.100	192.168.1.153	TCP	66	21231-995
17	2.622895000	192.168.1.100	192.168.1.153	TCP	66	995-21231
18	2.623067000	192.168.1.100	192.168.1.153	TCP	54	21231-995
19	2.623330000	192.168.1.100	192.168.1.153	TLSv1.2	427	Client Hel
20	2.650931000	192.168.1.100	192.168.1.153	NBNS	92	Name query
21	2.669520000	192.168.1.100	192.168.1.153	TCP	54	995-21231

Expert Info



Conversations

The image shows the Wireshark network protocol analyzer interface. The top menu bar includes FILE, HOME, INSERT, DESIGN, TRANSITIONS, ANIMATIONS, and SI. Below this is a toolbar with icons for file operations and analysis. The main menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Tools, and Internals. The 'Statistics' menu is open, showing options like Summary, Comments Summary, Show address resolution, Protocol Hierarchy, Conversations (highlighted), Endpoints, Packet Lengths..., IO Graph, Conversation List, Endpoint List, Service Response Time, and ANCP. The packet list pane shows a table of captured packets with columns No., Time, and Source. The filter 'tcp.stream eq 8' is applied. The packet list shows several packets, with packet 554 highlighted in black.

No.	Time	Source
540	8.23373000	192.168.0.4
554	8.54207400	192.168.0.4
555	8.59456900	54.231.8.80
561	8.74173800	54.231.8.80
563	8.74201500	192.168.0.4
564	8.79616300	54.231.8.80
565	8.79639000	192.168.0.4
570	8.88777500	54.231.8.80
573	9.01210300	54.231.8.80
574	9.01215600	54.231.8.80
575	9.01218700	192.168.0.4
576	9.01551000	192.168.0.4

HTTP Analysis

*Wi-Fi [Wireshark 1

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: tcp.stream eq 8

Summary

- Comments Summary
- Show address resolution
- Protocol Hierarchy
- Conversations
- Endpoints
- Packet Lengths...
- IO Graph
- Conversation List
- Endpoint List
- Service Response Time
- ANCP
- BACnet
- BOOTP-DHCP...
- Collectd...
- Compare...
- Flow Graph...
- HART-IP
- HTTP
- ONC-RPC Programs
- Sametime
- TCP StreamGraph
- UDP Multicast Streams
- WLAN Traffic
- IP Destinations
- IP Addresses
- IP Protocol Types

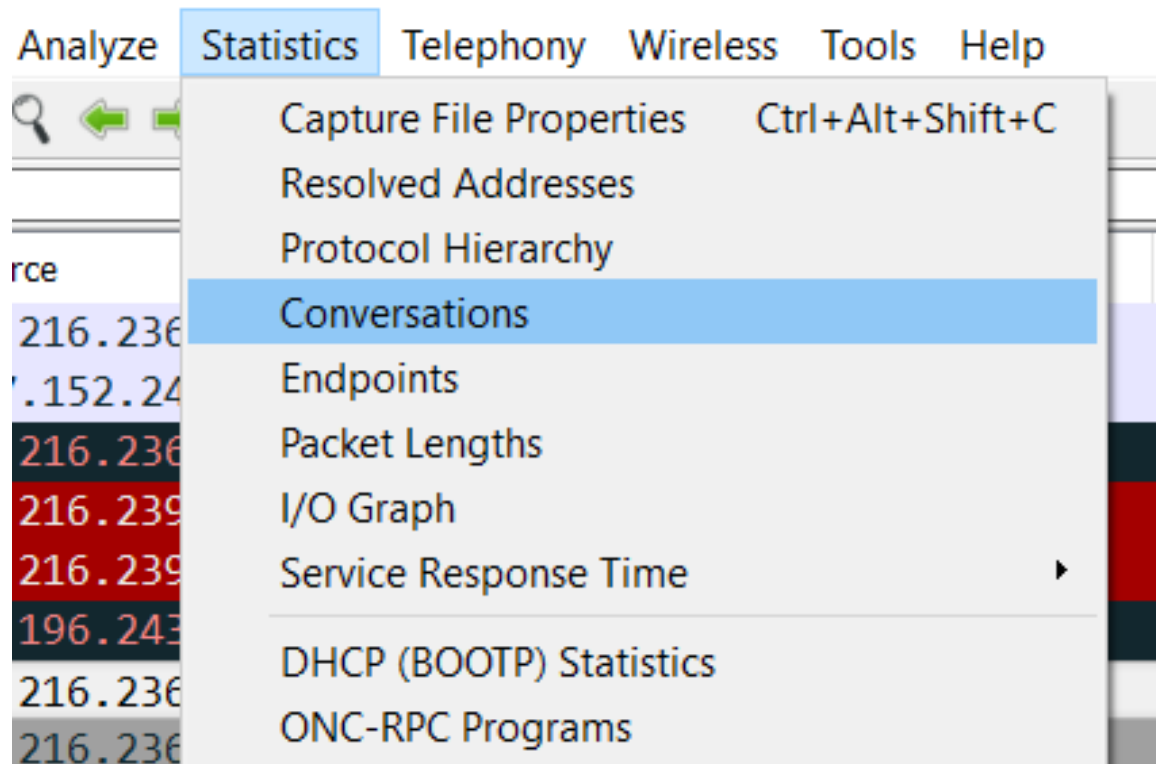
Packet Counter

Requests

Load Distribution

No.	Time	Source	Destination	Protocol	Length	Info
540	8.23373000	192.168.0.4	192.168.0.4	HTTP	403	GET /2014/RW/RWE...
554	8.54207400	192.168.0.4	192.168.0.4	HTTP	465	[TCP Retransmission]
555	8.59456900	54.231.8.80	192.168.0.4	TCP	60	http > 52528 [ACK]
561	8.74173800	54.231.8.80	192.168.0.4	HTTP	831	HTTP/1.1 200 OK
563	8.74201500	192.168.0.4	192.168.0.4	HTTP	466	GET /2014/RW/spac...
564	8.79616300	54.231.8.80	192.168.0.4	HTTP	440	HTTP/1.1 200 OK
565	8.79639000	192.168.0.4	192.168.0.4	HTTP	466	GET /2014/RW/RWE...
570	8.88777500	54.231.8.80	192.168.0.4	TCP	60	http > 52528 [ACK]
573	9.01210300	54.231.8.80	192.168.0.4	TCP	1514	[TCP segment of a
574	9.01215600	54.231.8.80	192.168.0.4	HTTP	1095	HTTP/1.1 200 OK
575	9.01218700	192.168.0.4	192.168.0.4	TCP	54	52528 > http [ACK]
576	9.01551000	192.168.0.4	192.168.0.4	HTTP	466	GET /2014/RW/spac...
578	9.07820900	54.231.8.80	192.168.0.4	HTTP	440	HTTP/1.1 200 OK
579	9.07845700	192.168.0.4	192.168.0.4	HTTP	466	GET /2014/RW/spac...
580	9.13417500	54.231.8.80	192.168.0.4	HTTP	440	HTTP/1.1 200 OK
581	9.13443000	192.168.0.4	192.168.0.4	HTTP	466	GET /2014/RW/spac...
582	9.19160600	54.231.8.80	192.168.0.4	HTTP	440	HTTP/1.1 200 OK
583	9.19185900	192.168.0.4	192.168.0.4	HTTP	466	GET /2014/RW/RWE...
584	9.24381300	54.231.8.80	192.168.0.4	TCP	60	http > 52528 [ACK]
588	9.40053800	54.231.8.80	192.168.0.4	HTTP	440	HTTP/1.1 200 OK
590	9.40679700	192.168.0.4	192.168.0.4	HTTP	466	GET /2014/RW/RWE...
592	9.50205100	54.231.8.80	192.168.0.4	TCP	1514	[TCP segment of a
597	9.60945400	54.231.8.80	192.168.0.4	HTTP	86	HTTP/1.1 200 OK
598	9.60949300	54.231.8.80	192.168.0.4	TCP	54	52528 > http [ACK]
599	9.60952300	192.168.0.4	192.168.0.4	HTTP	466	GET /2014/RW/spac...
601	9.66880000	192.168.0.4	192.168.0.4	HTTP	440	HTTP/1.1 200 OK
602	9.72303100	54.231.8.80	192.168.0.4	HTTP	466	GET /2014/RW/spac...
603	9.72330400	192.168.0.4	192.168.0.4	HTTP	440	HTTP/1.1 200 OK
607	9.77766600	54.231.8.80	192.168.0.4	HTTP	440	HTTP/1.1 200 OK
608	9.79692400	192.168.0.4	192.168.0.4	TCP	54	52528 > http [ACK]

View conversations



View conversations

Wireshark · Conversations · Wi-Fi											
Ethernet · 3			IPv4 · 13		IPv6	TCP · 16		UDP · 3			
Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
10.7.128.252	10.216.236.26	2	402	1	307	1	95	4.211881	0.0997	24 k	
10.7.128.253	10.216.236.26	2	402	1	307	1	95	4.237403	0.1005	24 k	
10.216.236.26	107.152.25.200	11	664	6	358	5	306	0.000000	24.3673	117	
10.216.236.26	69.196.243.137	24	1308	12	660	12	648	0.388188	21.7136	243	
10.216.236.26	54.172.240.22	14	870	7	413	7	457	0.388237	20.5787	160	
10.216.236.26	54.68.87.90	4	247	2	108	2	139	2.787637	0.1236	6989	
10.216.236.26	119.161.4.32	11	6429	4	476	7	5953	4.208883	0.4394	8666	
10.216.236.26	52.42.67.208	6	368	3	163	3	205	4.259463	0.4193	3110	
10.216.236.26	117.18.237.29	14	808	8	436	6	372	6.848380	16.4133	212	
10.216.236.26	107.152.24.211	10	597	5	303	5	294	6.986907	14.3364	169	
10.216.236.26	162.247.242.21	4	218	2	110	2	108	7.870307	10.3360	85	
10.216.236.26	107.152.24.197	21	1249	11	661	10	588	7.976273	14.7092	359	
10.216.239.254	255.255.255.255	12	4104	12	4104	0	0	1.353393	20.6871	1587	

Ngrep

- Ngrep
 - Download from <http://ngrep.sourceforge.net/download.html>
 - Examples
 - Monitor any traffic using port 443
 - `ngrep -d any port 443`
 - Capture traffic on eth0 with HTTP GET or POST
 - `ngrep -l -q -d eth0 -i "^GET|^POST" tcp and port 80`

Ngrep man page <http://linux.die.net/man/8/ngrep>

ISACA CISA Definition for Audit

- “Systematic process by which a qualified, competent, independent team or person objectively obtains and evaluates evidence regarding assertions about a process for the purpose of forming an opinion about and reporting on the degree to which the assertion is implemented.”



Auditor Qualifications

- **Independent:**
 - **Professional Independence:** Auditor acts independent of group being audited
 - No friendships, dating, suggestive language, parties, lunches
 - **Organizational Independence:** Auditor and his/her organization has no special interest in the audited organization
- **Qualified, Competent:**
 - Adhere to Professional Ethics Standard
 - ISACA standard and professional care
 - Professional Competence
 - Has skills/knowledge to complete task
 - Continued professional training/education



Terms used in Audits

Control: The policies, procedures, practices and organizational structures designed to provide reasonable assurance that business objectives will be achieved and that undesired events will be prevented or detected and corrected.

IT Control Objective: A statement of the desired result or purpose to be achieved by implementing control procedures in a particular IT activity.

Risk: The potential that a given threat will exploit vulnerabilities of an asset or group of assets to cause loss or damage to the assets.

Evidence: Evidence is any information used by the auditors whether the entity or data being audited follows the established audit criteria or objective.

IT Governance: A structure of relationships and processes to direct and control the enterprise in order to achieve the enterprise's goals by adding value while balancing risk versus return over IT and its processes

Control Self-Assessment (CSA)

- Control Assessment can be defined as a “management technique that assures stakeholders, customers and other parties that internal control system of the organization is reliable. It also ensures that employees are aware of the risks to the business and they conduct periodic, proactive reviews of control.

Substantive v Compliance Testing

- **Compliance Testing:**

- Are controls in place and consistently applied?
- Access control
- Program change control
- Procedure documentation
- Program documentation
- Software license audits
- System log reviews
- Exception follow-ups

- **Substantive Testing:**

- Are transactions processed accurately?
- Are data correct and accurate?
- Double check processing
- Calculation validation
- Error checking
- Operational documentation
- If Compliance results are poor, Substantive testing should increase in type and sample number



Compliance Testing

- Control: Is production software controlled?
- Test: Are production executable files built from production source files?
- Test: Were proper procedures followed in their release?
- Control: Is Sales DB access constrained to Least Privilege?
- Test: Are permissions allocated according to documentation?
- Test: When sample persons access DB, can they access only what is allowed?

Substantive Testing

Audit: Is financial statement section related to sales accurate?

Test: Track processing of a sample transactions through the system, performing calculations manually

Test: Test error conditions

Audit: Is tape inventory correct?

Test: Search for sample days and verify complete documentation and tape completeness

Sampling



Statistical Sampling:

- N% of all items randomly tested
- Should represent population distribution

Variable Sampling: How accurate is the sample population in matching the full population?

- Determine appropriateness of sampling: (e.g., \$, weight, amount):
Sample average \$24.50, Real average: \$26.99

No statistical (or Judgment) Sampling:

- Auditor justifies another distribution for sample selection
- Which items are most risky?



Sampling

- **Tolerable Error Rate:** The maximum allowable error rate (e.g., inappropriately documented changes)
- **Non-Statistical Sampling** includes:
 - **Discovery Sampling:** A minimal testing model used when the expected occurrence rate is extremely low (e.g., find fraud, break laws)
 - **Stop-or-Go Sampling:** If the first 20 have zero errors, then stop. Else if the first 100 have < 10 errors, stop. Else...
 - **Attribute Sampling:** How many of X have Y attribute?
- E.g. How many changes are appropriately documented?