COMP 5350 / 6350
Digital Forensics

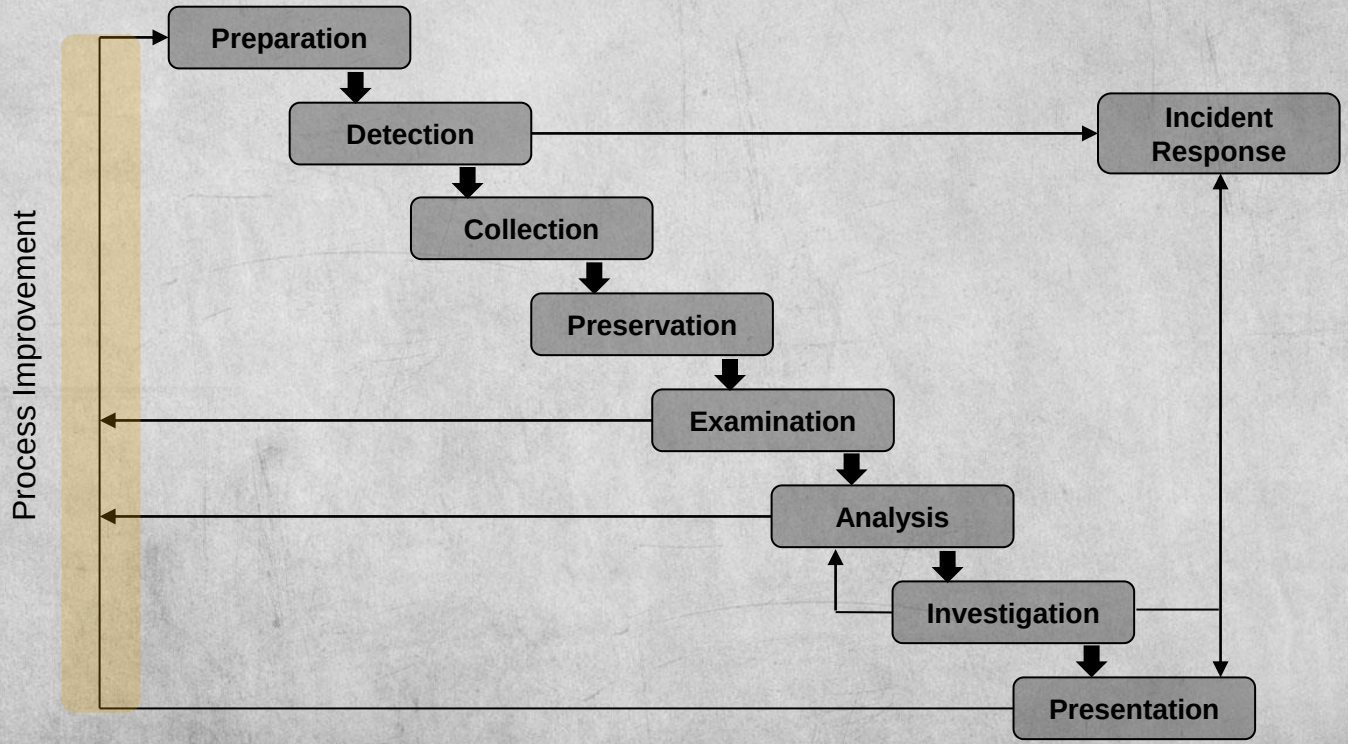
Network Forensics

Network Forensics

- A field within digital forensics that focuses on the collection, monitoring, and analysis of computer network traffic
- We must transition from disk level to network level forensics which requires expertise in several disciplines
  - ✓ Network Protocols
  - ✓ Network Architecture
  - ✓ Network Traffic Collection
  - ✓ Intrusion Detection
    - Behavioral
    - Anomaly
    - Signature
    - Heuristic

Network Forensics Process Model

- When conducting network forensics analysis, the following activities are conducted:

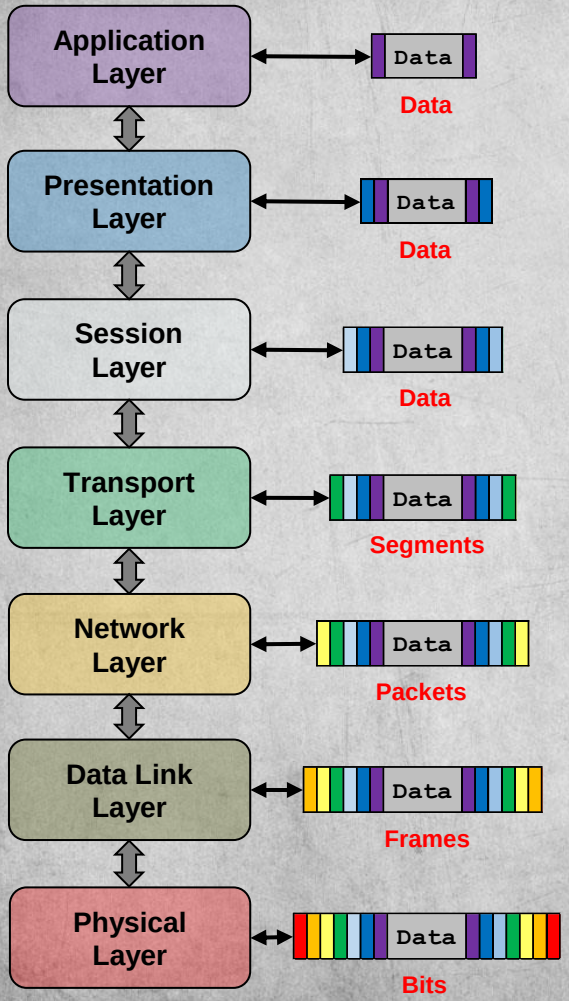


Introduction to Network Communication Protocols

Network Communication Protocols

- A protocol is a mechanism by which systems can communicate by establishing a set of rules and common message formats during transmission
- There will be two models that we will introduce relative to protocol utilization
  - ✓ Open Systems Interconnection (OSI) Model
  - ✓ Transmission Control Protocol / Internet Protocol (TCP/IP) Model
- When legacy protocols were initially created the emphasis was on usability not security so over time additional security features have been added

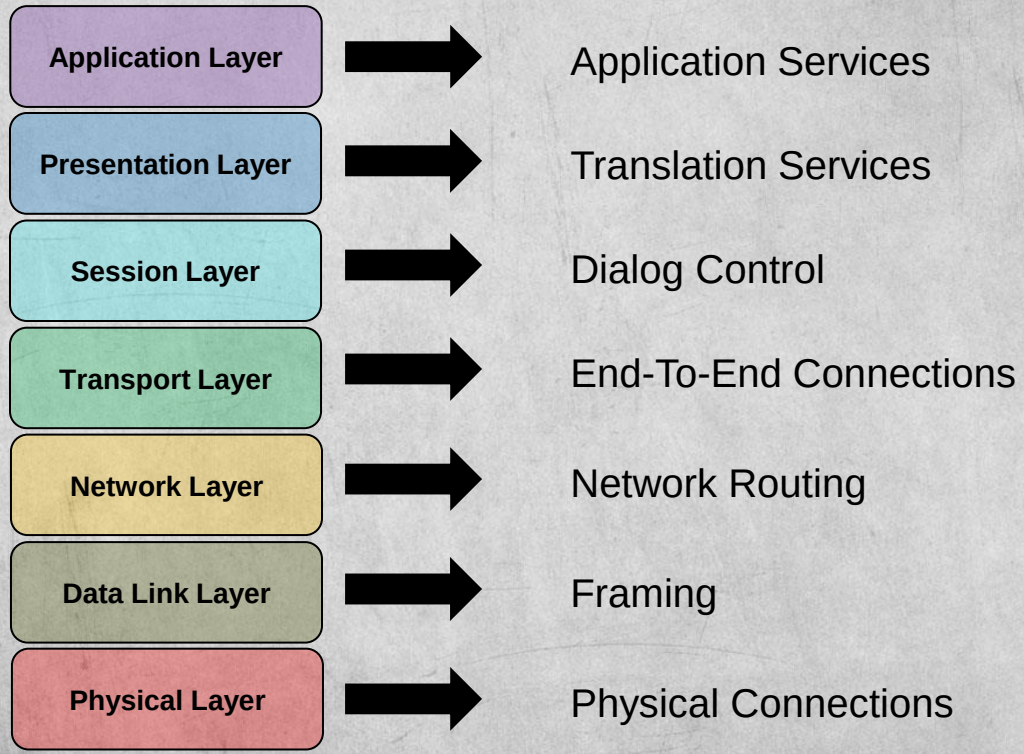
OSI Model



- The OSI model is composed of 7 layers
  - ✓ Layer 7 – Application Layer
  - ✓ Layer 6 – Presentation Layer
  - ✓ Layer 5 – Session Layer
  - ✓ Layer 4 – Transport Layer
  - ✓ Layer 3 – Network Layer
  - ✓ Layer 2 – Data Link Layer
  - ✓ Layer 1 – Physical Layer
- Lower layer services “provides” services to higher layers
- Higher layer services “uses” services of lower layers
- The following list of protocols are commonly used and should be studied in preparation for the exam

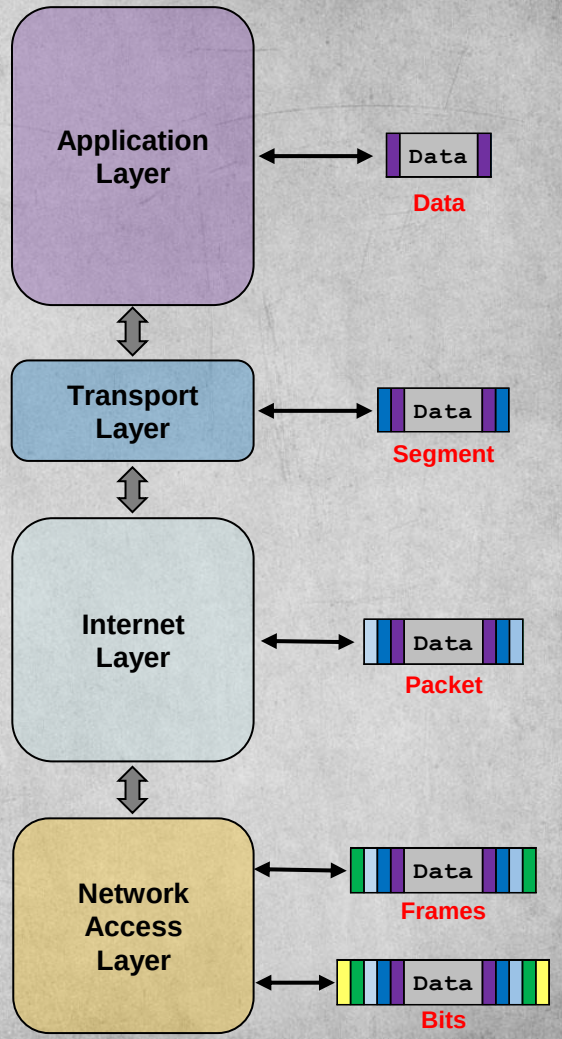
OSI Layer Functions

- Each layer of the OSI model provides a specific function

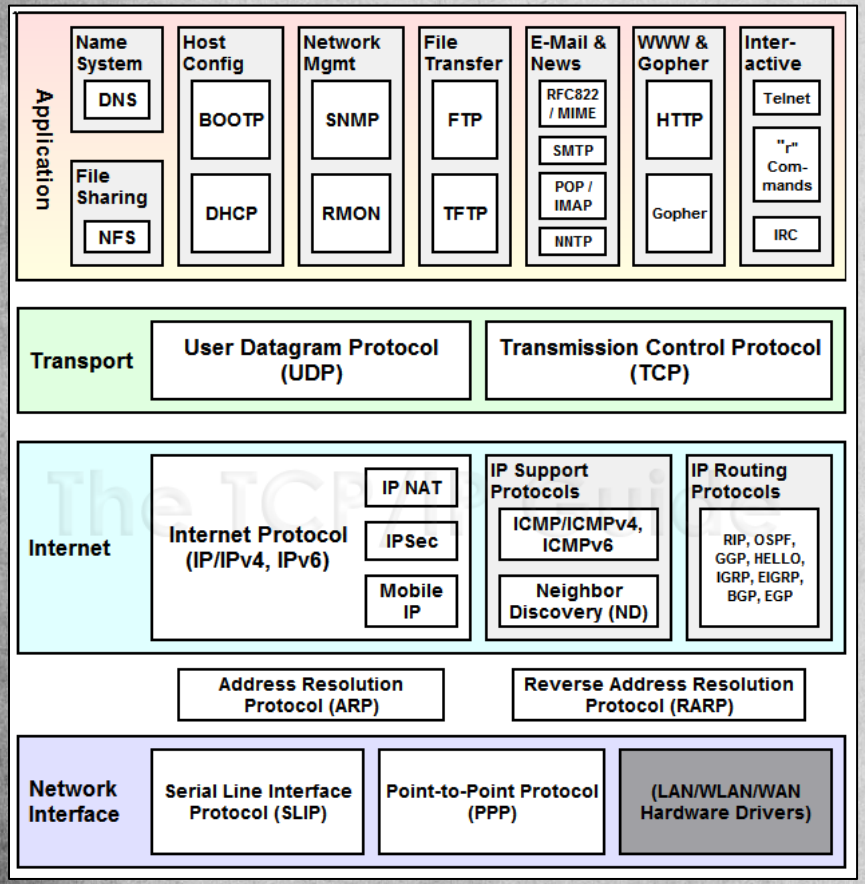


TCP / IP

- Although the OSI model provided an initial framework for network communication, eventually TCP/IP took its place and is now the underlying suite of protocols that aids in the creation, transmission, and reception of internet traffic
- TCP / IP contains four layers
  - ✓ Application Layer
    - Application Data
  - ✓ Transport Layer
    - Ports
  - ✓ Internet Layer
    - IP Addressing
  - ✓ Network Access Layer
    - MAC Addressing
- We will focus on the TCP/IP model of network communication for the rest of this session



TCP/IP Protocols



- Each layer of the TCP/IP model will have protocols that provide different capabilities
- Users are most familiar with application layer protocols since they provide the user interface
- In order to conduct an effective network forensics analysis, it is first necessary to learn what these protocols are and how they work
- We will focus on the TCP/IP model of network communication for the rest of this session

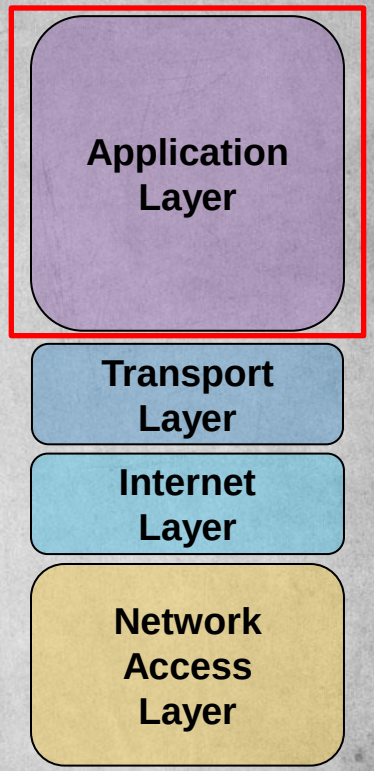
Request For Comments

- A Request for Comments (RFC) is a type of publication from the Internet Engineering Task Force (IETF) and the Internet Society (ISOC), the principal technical development and standards-setting bodies for the Internet
- RFCs define how different protocols will behave to ensure consistent operation and provide all relevant technical information
- Some of common protocol RFC’s
  - ✓ RFC 2616 – Hypertext Protocol (HTTP)
  - ✓ RFC 959 – File Transfer Protocol (FTP)
  - ✓ RFC 821 – Simple Mail Transfer Protocol (SMTP)

Network Protocols by TCP/IP Layer

Application Layer Protocols

- Application layer protocols are focused on shared communications protocols used by hosts in a communications network
- Every network-based protocol is defined by the Internet Engineering Task Force (IETF), which is the organization that publishes technical documentation known as a Request for Comment (RFC)
- Each protocol that we discuss will have its own RFC



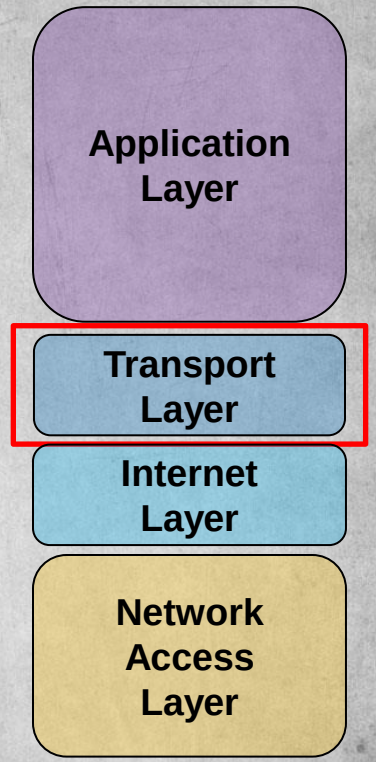
Application Layer Protocols

Protocol	Name
FTP	File Transfer Protocol – Data
FTP	FTP – Connection
SSH	Secure Shell
SFTP	SSH FTP
SCP	Secure Copy
Telnet	Telnet
SMTP	Simple Mail Transfer Protocol
DNS	Domain Name System
DHCP	Dynamic Host Configuration Protocol

Protocol	Name
TFTP	Trivial FTP
POP3	Post Office Protocol
HTTP	Hypertext Transfer Protocol
POP3	Post Office Protocol
IMAP	Internet Message Access Protocol
SNMP	Simple Network Management Protocol
HTTPS	HTTP Secure
FTPS	FTP over SSL
RDP	Remote Desktop Protocol

Transport Layer Protocols

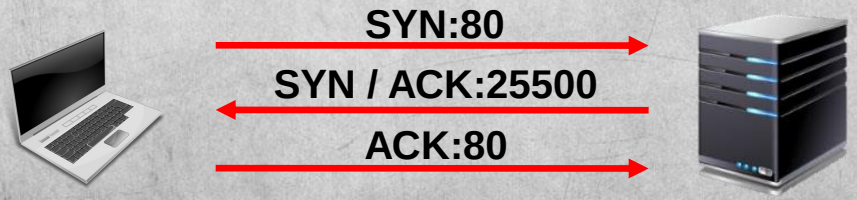
- Transport layer protocols establish host-to-host communication services for applications and services including connection-oriented communication, reliability, flow control, and multiplexing
- The RFC for each application contains information about the ports used to convey information between hosts and applications



Protocol	Name
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

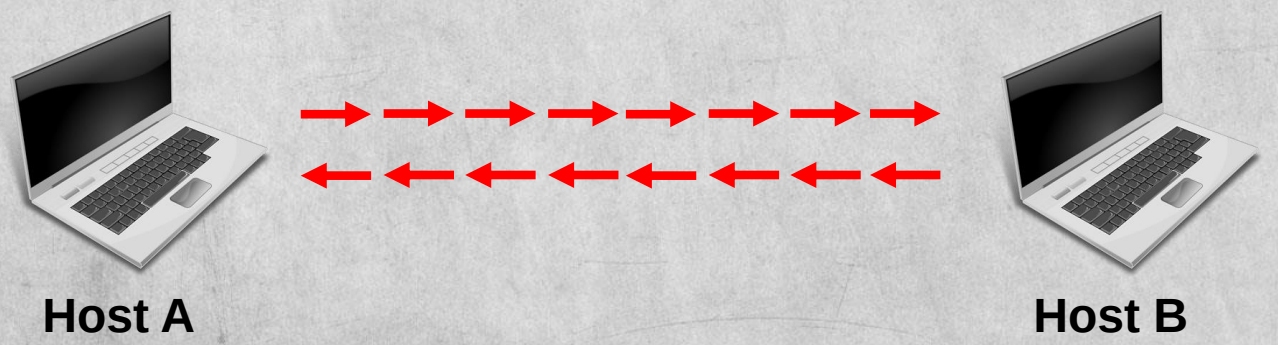
TCP Connection Process

- TCP ensures reliable data delivery through error checking, acknowledgements, and if necessary, retransmission
- In order to communicate with TCP, hosts must first establish a connection which is known as a “virtual circuit”
- A successful “three-way” handshake is required before the virtual circuit is established
- The handshake process also establishes acknowledgement and windowing parameters during transmission



UDP Process

- UDP is called a “best effort” communication process
- Datagrams are sent with no regard to reception of each packet

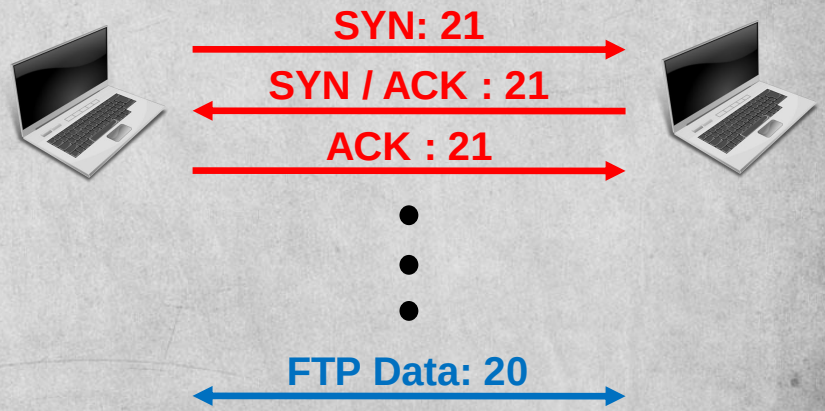


Introduction to Protocols

- Now that we have introduced TCP/IP application and transport layer details, we can highlight common protocols seen during a forensics investigation
- Network forensics can establish user or organizational objectives
- For each protocol we will identify
  - ✓ Protocol Definitions
  - ✓ Protocol Capabilities
  - ✓ Port Information

File Transfer Protocol

- FTP provides file transfer between hosts
- As with any TCP-based protocol, FTP must first establish a connection between hosts
- FTP Ports:
  - ✓ 20/TCP – FTP Data
  - ✓ 21/TCP – FTP Control
- FTPS Port:
  - ✓ 990/TCP

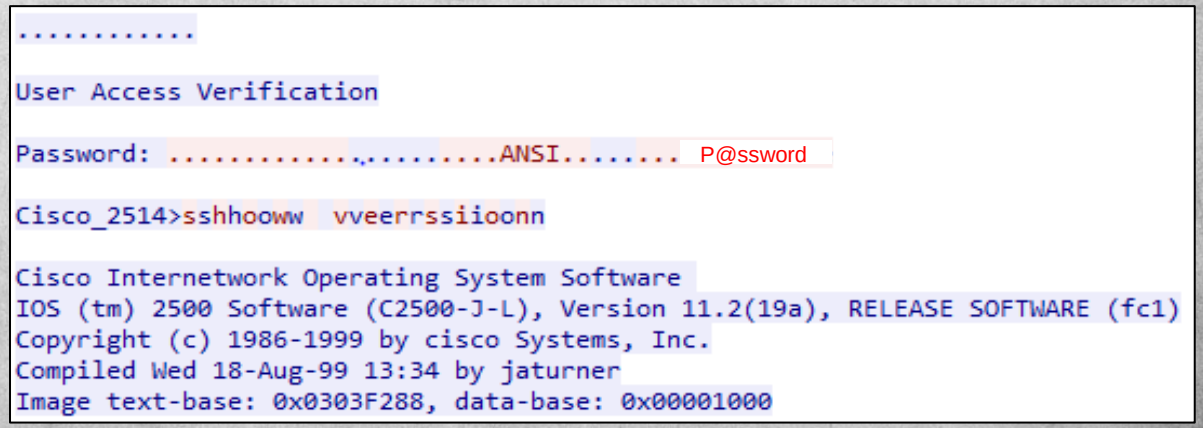


Secure Shell / Secure FTP / Secure Copy

- Many remote connection protocols are unencrypted and do not provide confidentiality of data
- The following protocols are purposely designed to provide traffic encryption
- SSH / SFTP / SCP Port:
  - ✓ 22/TCP

Telnet

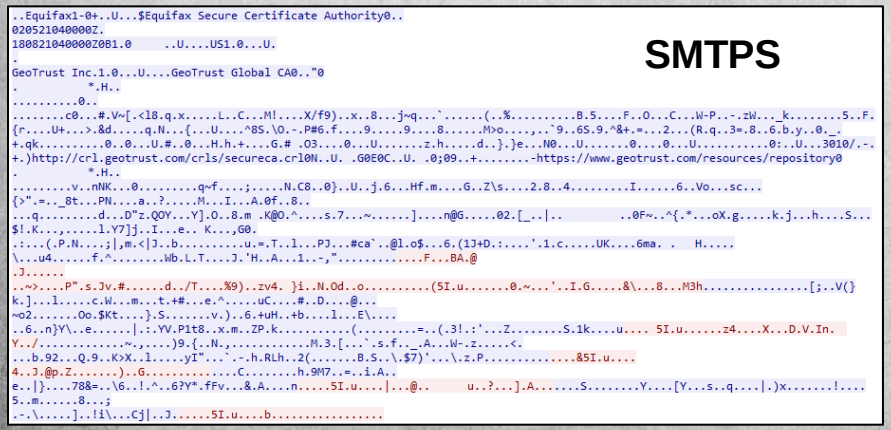
- A legacy protocol used to establish basic connections between hosts
  - ✓ Network Devices
    - Switches
    - Routers
- Telnet provides an unencrypted channel to establish connections
- Telnet Port:
  - ✓ 23/TCP



Simple Message Transfer Protocol

- SMTP was one of the first protocols dedicated to electronic mail transmission
- SMTP Port:
  - ✓ 25/TCP
- SMTPS Port:
  - ✓ 465/TCP

SMTPS

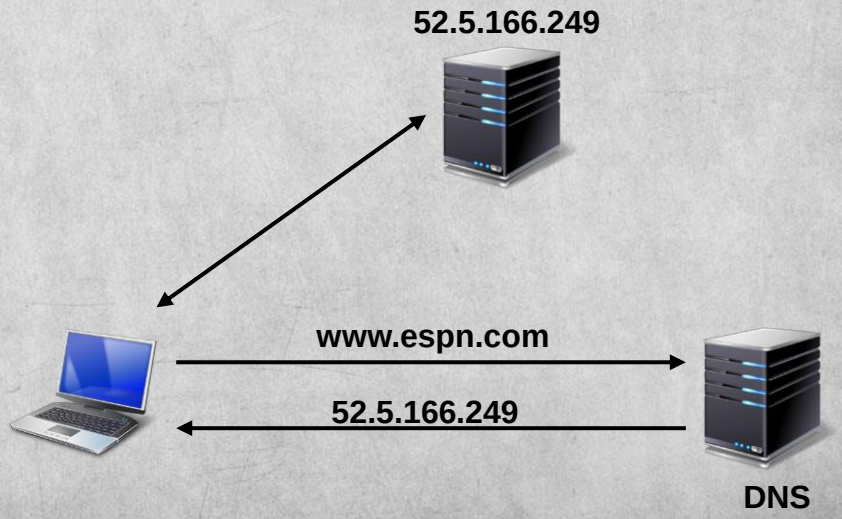


SMTP

```
220 mx100.stngva01.us.mxservers.net ESMTP mxl_mta-1.3.8-10p4; Mon, 15 Jan 2007 16:49:50 -0500 (EST); NO UCE
EHLO Vaio
250-mx100.stngva01.us.mxservers.net
250-SIZE 0
250 PIPELINING
MAIL FROM: <breaddy16@packet-level.com>
250 Sender Ok
RCPT TO: <bbelch@packet-level.com>
250 bbelch@packet-level.com ok (normal)
DATA
354 Start mail input; end with <CRLF>.<CRLF>
From: "Brian Readdy16" <breaddy16@packet-level.com>
To: "'Barnel Belch'" <bbelch@packet-level.com>
Subject: Test email
Date: Mon, 15 Jan 2007 13:55:23 -0800
Message-ID: <006d01c738ef$e544a990$e522a143@hq.wnbnet>
MIME-Version: 1.0
Content-Type: multipart/alternative;
        boundary="----=_NextPart_000_006E_01C738AC.D7216990"
X-Mailer: Microsoft Office Outlook 11
thread-index: Acc479ph2gILPVD6QgiKnNrF3RpHpQ==
X-MimeOLE: Produced By Microsoft MimeOLE V6.00.2900.3028

This is a multi-part message in MIME format.

------=_NextPart_000_006E_01C738AC.D7216990
Content-Type: text/plain;
        charset="us-ascii"
Content-Transfer-Encoding: 7bit

This is just a test email - you don't need to respond.


Brain

This message is intended only for the use of the addressee and may contain
information that is privileged and confidential. If you are not the intended
recipient, you are hereby notified that any use and/or dissemination of this
communication is strictly prohibited. If you have received this
communication in error, please delete all copies of the message and its
attachments and notify the sender immediately.
```

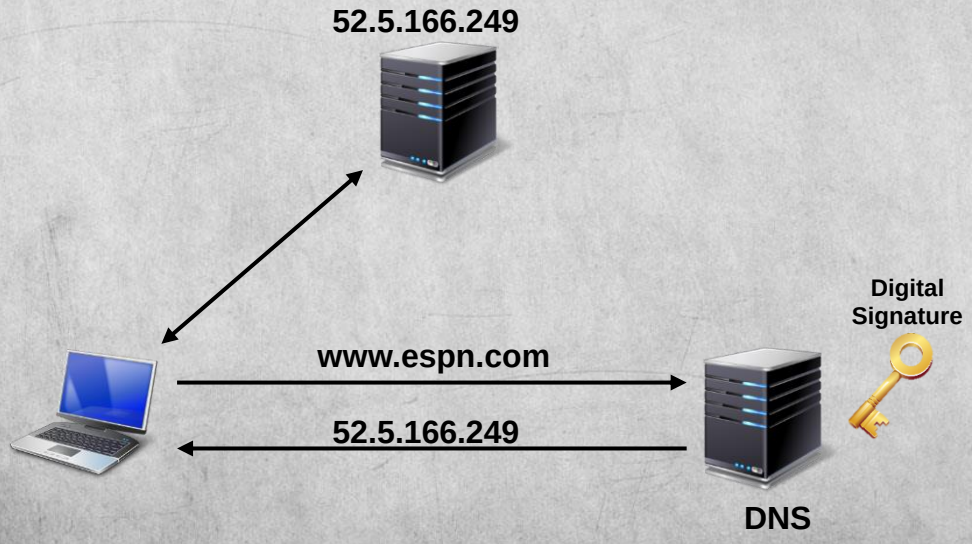
DNS

- Domain Name Service translates Fully Qualified Domain Names into an IP address and is critical for web communication
- DNS Ports:
  - ✓ 53/TCP – DNS Zones
  - ✓ 53/UDP – NSLookups
- DNS provides a number of records that identify key information including:
  - ✓ A – IPv4 Address
  - ✓ AAAA – IPv6 Address
  - ✓ CNAME – Canonical Name
  - ✓ MX – Mail Exchange
  - ✓ PTR – Pointer Record
  - ✓ NS – Name Server
  - ✓ SOA – Start of Authority
  - ✓ SRV – Service
  - ✓ TXT – Text



DNSSEC

- DNS is insecure because it does not validate DNS responses
- DNS Security Extensions (DNSSEC) does not encrypt transmissions, but provides integrity by digitally signing DNS responses
- DNSSEC require authentication keys for each DNS server



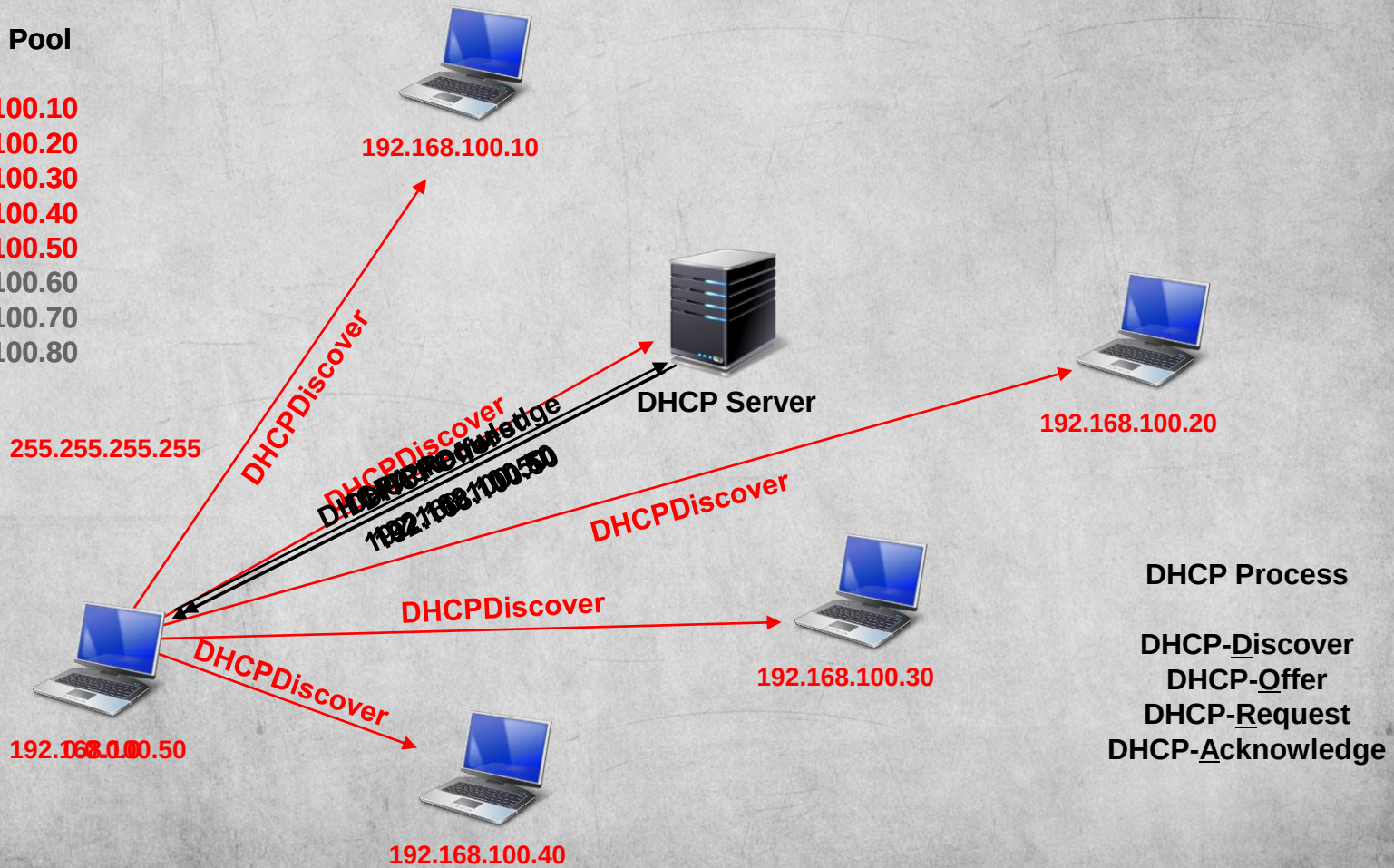
Dynamic Host Configuration Protocol

- As systems are turned on and go through the boot process, there will be a time when they will need to acquire local IP addressing to communicate on a local area network
- Dynamic Host Configuration Protocol (DHCP) is made up of a 4-step process to assign IP addresses
  - ✓ DHCP Discover
  - ✓ DHCP Offer
  - ✓ DHCP Request
  - ✓ DHCP Acknowledge
- DHCP Ports:
  - ✓ DHCP Servers
    - 67/UDP
  - ✓ DHCP Clients
    - 68/UDP

Dynamic Host Configuration Protocol

DHCP IP Pool

192.168.100.10
192.168.100.20
192.168.100.30
192.168.100.40
192.168.100.50
192.168.100.60
192.168.100.70
192.168.100.80



DHCP Process

DHCP-Discover
DHCP-Offer
DHCP-Request
DHCP-Acknowledge

Post Office Protocol

- Unlike SMTP, POP3 is an electronic mail protocol that is designed to pull emails from a remote server and deliver it to a host-based email application
- POP3 Port:
  - 110/TCP
- POP3S Port:
  - 995/TCP

SNMP

- Simple Network Management Protocol manages and monitors IP addressed devices on a network
  - ✓ Hubs
  - ✓ Switches
  - ✓ Routers

- SNMP Port:
  - ✓ 161/UDP

- SNMP unencrypted versions: SNMPv1, SNMPv2c

- SNMP encrypted version: SNMPv3

Web Related Protocols

- HTTP – Hypertext Transfer Protocol
  - ✓ Establishes client-server communication with linked content
  - ✓ 80/TCP
- HTTPS – HTTP with SSL / TLS Encryption
  - ✓ 443/TCP
- SSL – Secure Socket Layer
  - ✓ Transport layer encryption utilizing public key cryptography
  - ✓ SSL 1.0, 2.0, and 3.0
  - ✓ SSL 3.0 transitioned to TLS 1.0
- TLS – Transport Layer Security
  - ✓ Transport layer replacing SSL
  - ✓ TLS 1.0, 1.1, 1.2

Remote Access Protocols

- Some of the more common remote access protocols that may be observed during a forensics investigation include:

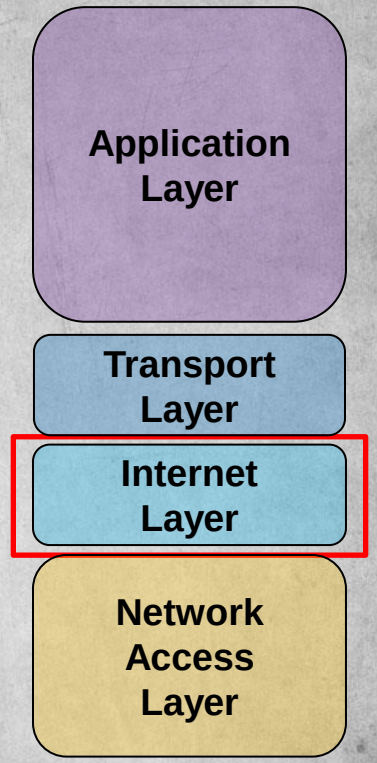
  - ✓ Remote Desktop Protocol (RDP / xRDP)
    - TCP/3389
  - ✓ Virtual Network Computing (VNC)
    - TCP/5900

Application / Transport Protocol Summary

Protocol	Transport Protocol	Transport Layer Port	Name
FTP	TCP	20 / 21	File Transfer Protocol – Data / Connection
SSH	TCP	22	Secure Shell / Secure FTP / Secure Copy
Telnet	TCP	23	Telnet
SMTP	TCP	25 / 465	Simple Mail Transfer Protocol / SMTPS
DNS	UDP / TCP	53	Domain Name System – Lookups / Zones
DHCP	UDP	67 / 68	Dynamic Host Configuration Protocol – Server / Client
HTTP	TCP	80 / 443	HTTP / HTTPS
POP3	TCP	110 / 995	Post Office Protocol / POPS
SNMP	UDP	161	Simple Network Management Protocol (v1, v2)
RDP	TCP	3389	Remote Desktop Protocol

Internet Layer Protocols

- The Internet Layer Provides:
  - ✓ Packet Routing
  - ✓ IP Address Identification
- Routing protocols include:
  - ✓ IGP – RIP, IGRP, OSPF, IS-IS
  - ✓ EGP – BGP
- Although significant forensics information can be found in routing protocols, we will focus on IP



Protocol	Name
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6

IPv4 Addressing Rules

- An IP address must be assigned to a host to communicate with other hosts
- An IP address is a 32-bit value formatted in a dotted-decimal notation containing 4 octets
- Each network interface card (NIC) is assigned an IP address before communicating over a network, LAN or WAN
- There are a total of 32-bits in an IPv4 address which means there are 2^32 or 4,294,967,296 potential IP addresses available for use

204.17.125.47

11001100.00010001.1111101.00101111

IPv4 Address Types

- There are three modes to consider when considering IP communications
  - ✓ Unicast: 1-to-1
  - ✓ Multicast: 1-to-Many
  - ✓ Broadcast: 1-to-All

- We will introduce the following IPv4 address types:
  - ✓ Class A, B, C, D, & E
  - ✓ Private
  - ✓ Loopback
  - ✓ Broadcast
  - ✓ APIPA

IPv4 Network Classes

- There are 5 IPv4 network classes with the following IP address ranges:
  - ✓ Class A

    0.0.0.0 – 127.255.255.255
  - ✓ Class B

    128.0.0.0 – 191.255.255.255
  - ✓ Class C

    192.0.0.0 – 223.255.255.255
  - ✓ Class D (Multicast)

    224.0.0.0 – 239.255.255.255
  - ✓ Class E

    240.0.0.0 – 247.255.255.255
- The determination of which IPv4 network class an address falls into, is based solely on the value of the first octet

Private IP Addresses

- There are a range of IP addresses assigned solely for internal IP addressing
- These addresses are locally used and non-routable across the internet
- RFC 1918 documents private IP addresses

Class A
10.0.0.0 – 10.255.255.255

Class B
172.16.255.255 - 172.31.255.255

Class C
192.168.0.0 – 192.168.255.255

Loopback Address

- A loopback address is a network interface configuration that allows for signals to remain within a host
- Loopback addresses can be used to debug traffic before they leave the confines of a network
- IPv4 loopback address can be with the following range:

127.0.0.0 – 127.255.255.255

Broadcast Address

- A broadcast is used to send messages to all hosts on a network segment
- A broadcast address is identified when 2 or more concurrent octets are designated with all 1’s

255.255.255.255	Indicates a broadcast for all hosts in a network
172.16.255.255	Indicates a broadcast for all subnets and hosts in the 172.16.0.0 network
10.255.255.255	Indicates a broadcast for all subnets and hosts in the 10.0.0.0 network

APIPA Addresses

- Automatic Private IP Addressing is assigned to any host not receiving a proper IP address during the DHCP process
- Used in LANs and non-routable
- APIPA address range:

169.254.0.1 – 169.254.255.254

Network Forensic Analysis

Libpcap Introduction

- With a basic introduction to protocols completed, we will now talk about the tools that will help to conduct network forensics
- libpcap is an open-source C-language library for capturing, creating, and manipulating network packets
  - ✓ Npcap and WinPcap are the Windows-based versions of the libpcap library
- All tools that we will use from a network forensic analysis standpoint use the libpcap format and

Network Traffic Analysis

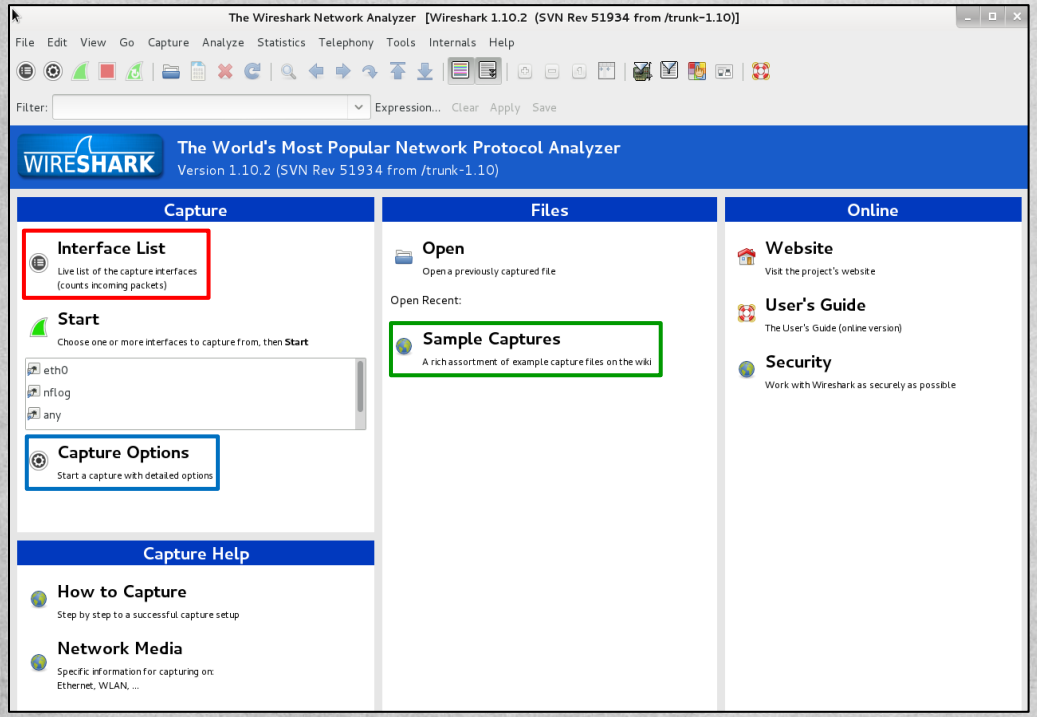
- During a digital forensics investigation, it may be necessary to analyze network traffic for malicious activities
- Network traffic can help to identify all level of malicious activities
  - ✓ Network Misconfigurations
  - ✓ User Activities
  - ✓ Malware Deployment
- Some of the more common network forensic analysis tools include:
  - ✓ Wireshark / Tshark*
  - ✓ TCPDump
  - ✓ Network Miner
  - ✓ Passive Asset Detection System (PADS)
  - ✓ System for Internet-Level Knowledge (SiLK)
  - ✓ Snort
  - ✓ Suricata

Wireshark Introduction

- Although Wireshark can be used to determine network functionality and traffic flow, some techniques provide significant capability within an organization
  - ✓ Traffic Capture for Forensic Analysis
  - ✓ Traffic Filtering
  - ✓ Encrypted Packet Decryption
  - ✓ Identification of System Misconfigurations
  - ✓ Rebuild Sessions
  - ✓ Identifying Signatures for IDS and IPS systems

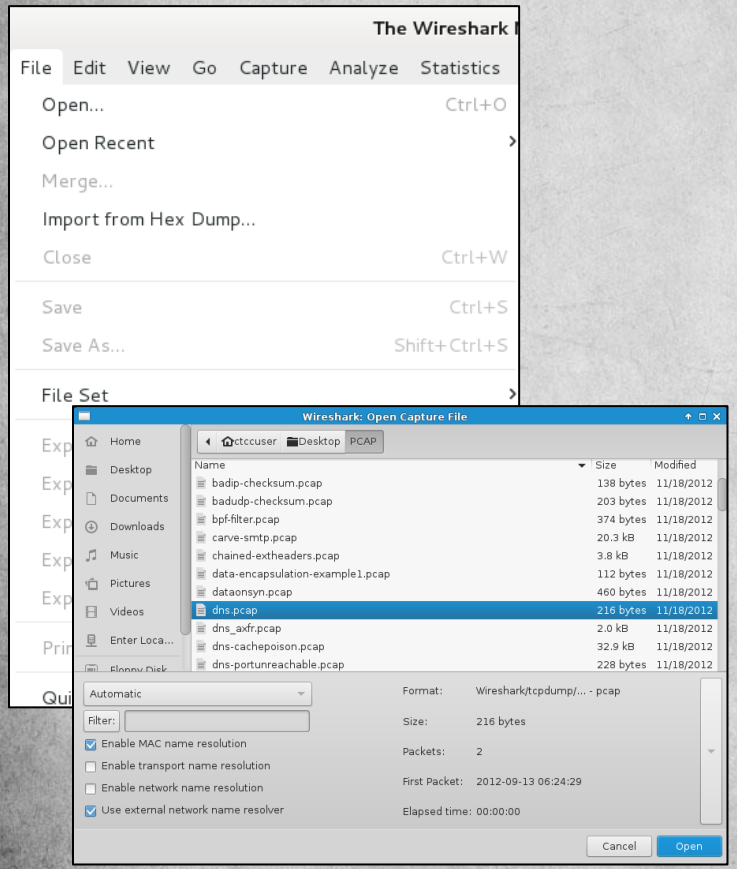
Wireshark User Interface

Wireshark User Interface



- Interface List
- Capture Options
- Sample Captures

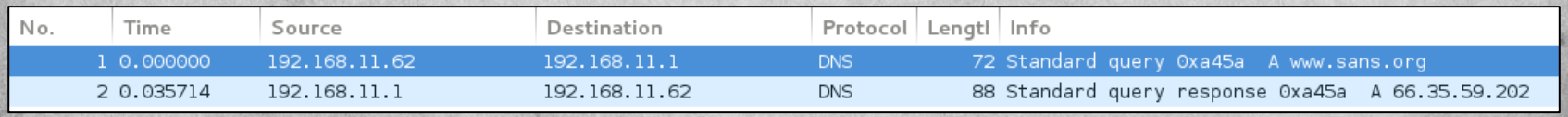
Traffic Selection



- Previously recorded traffic can be loaded into Wireshark for analysis
- These “traces” are not played back with the same timing as they were collected
- Once selected, display options can be configured
- Other tools are required to playback traces with timing
  - ✓ tcpreplay
  - ✓ bittwist
  - ✓ scapy

Wireshark Default Fields

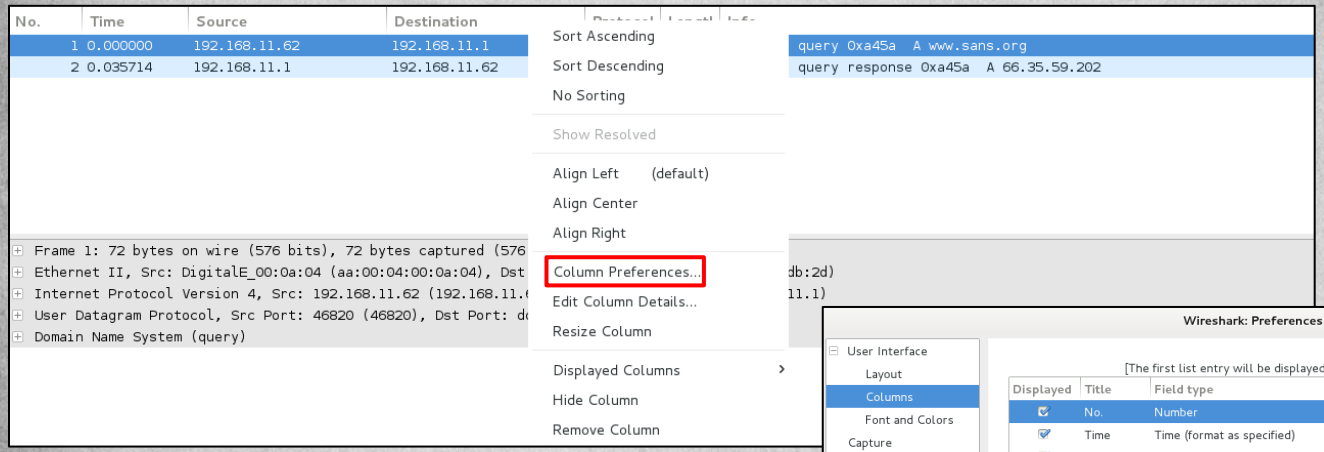
- Wireshark provides a default layout that provides:
  - ✓ Packet Number
  - ✓ Time
  - ✓ Source IP
  - ✓ Destination IP
  - ✓ Protocol
  - ✓ Packet Length in Bytes
  - ✓ Information
- For a more rigorous forensics analysis, it will be necessary to add additional fields

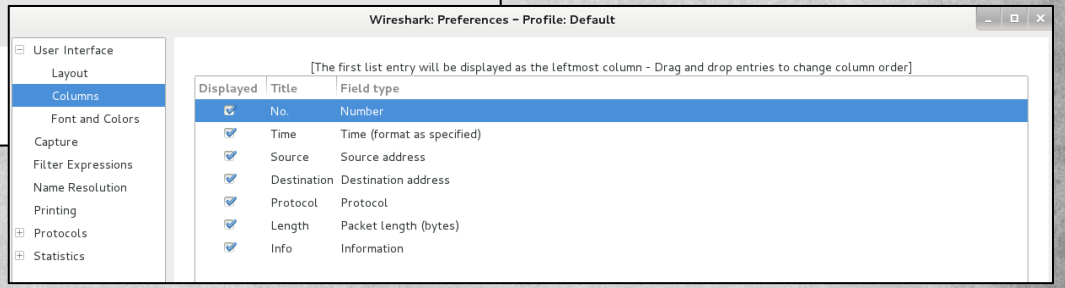


No.	Time	Source	Destination	Protocol	Lengtl	Info
1	0.000000	192.168.11.62	192.168.11.1	DNS	72	Standard query 0xa45a A www.sans.org
2	0.035714	192.168.11.1	192.168.11.62	DNS	88	Standard query response 0xa45a A 66.35.59.202

Wireshark Column Configuration

- Other fields that should be added include SRC MAC, DST MAC, SRC IP, DST IP, SRC Port, DST Port, Protocol, Info, Length

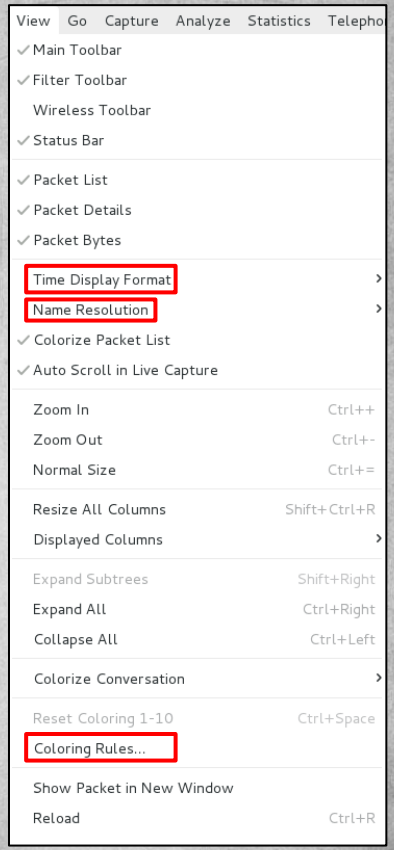




No.	Time	SRC MAC	DST MAC	SRC IP	DST IP	SRC Port	DST Port	Protocol	Info	Length
1	0.000000	aa:00:04:00:0a:04	4c:e6:76:40:db:2d	192.168.11.62	192.168.11.1	46820	53	DNS	Standard query 0xa45a A www.sans.org	72
2	0.035714	4c:e6:76:40:db:2d	aa:00:04:00:0a:04	192.168.11.1	192.168.11.62	53	46820	DNS	Standard query response 0xa45a A 66.35.59.202	88

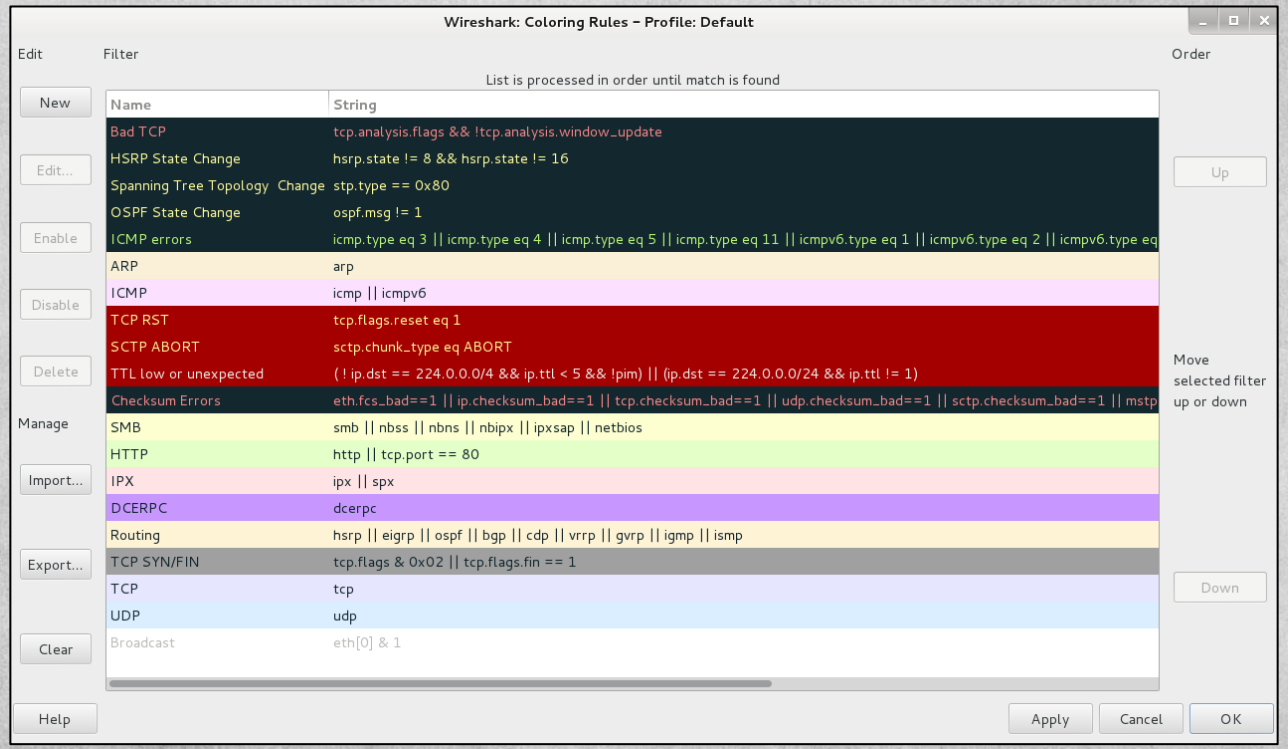
Wireshark View Menu

- There are several view options that can provide additional information within a network trace
- Time Display Format can format packets into any time that is required
  - ✓ Time format is based on local time
- Name Resolution can be configured to resolve
  - ✓ MAC Address Resolution (OUI)
  - ✓ Transport Layer Resolution (Ports)
  - ✓ Network Layer Resolution (IP)
- Coloring Rules

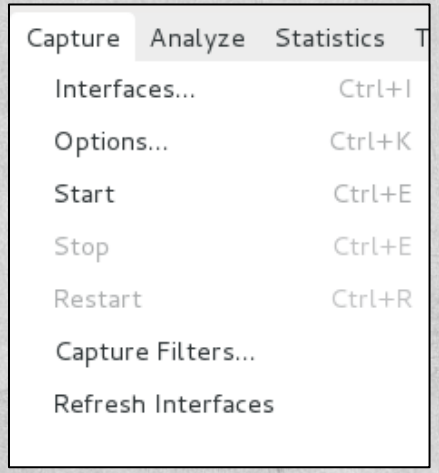


Coloring Rules

- Coloring rules can aid in identifying specific activities that are of interested during a forensics investigation

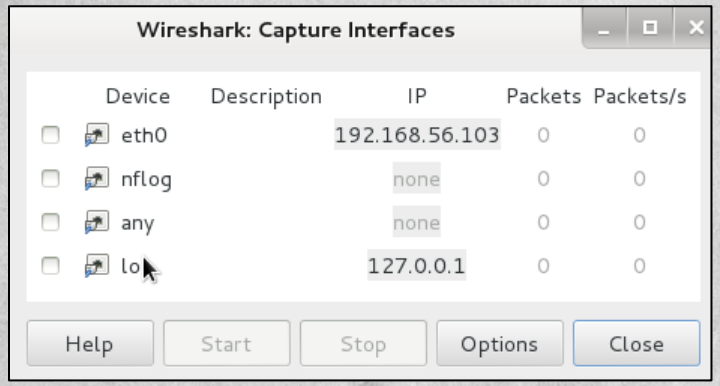


Capture Menu



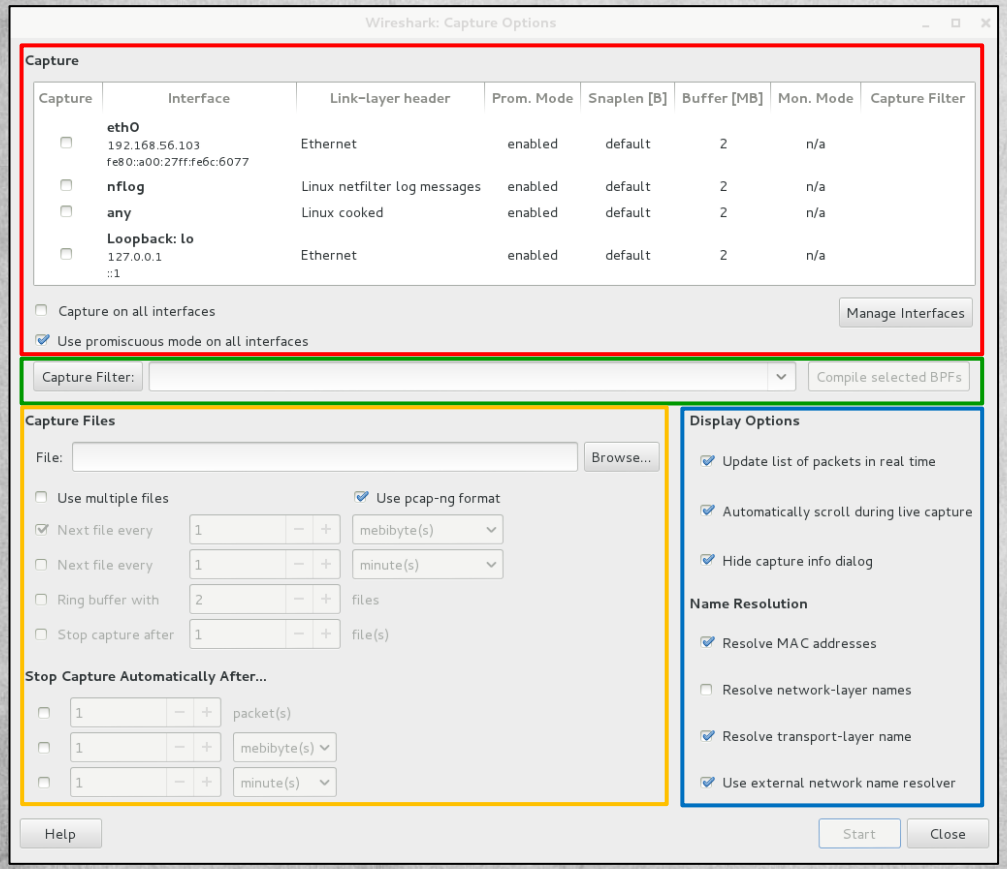
- Interfaces
- Options
- Capture Filters

Capture Interfaces



- Select capture one or all interfaces available
- Capture Options are available

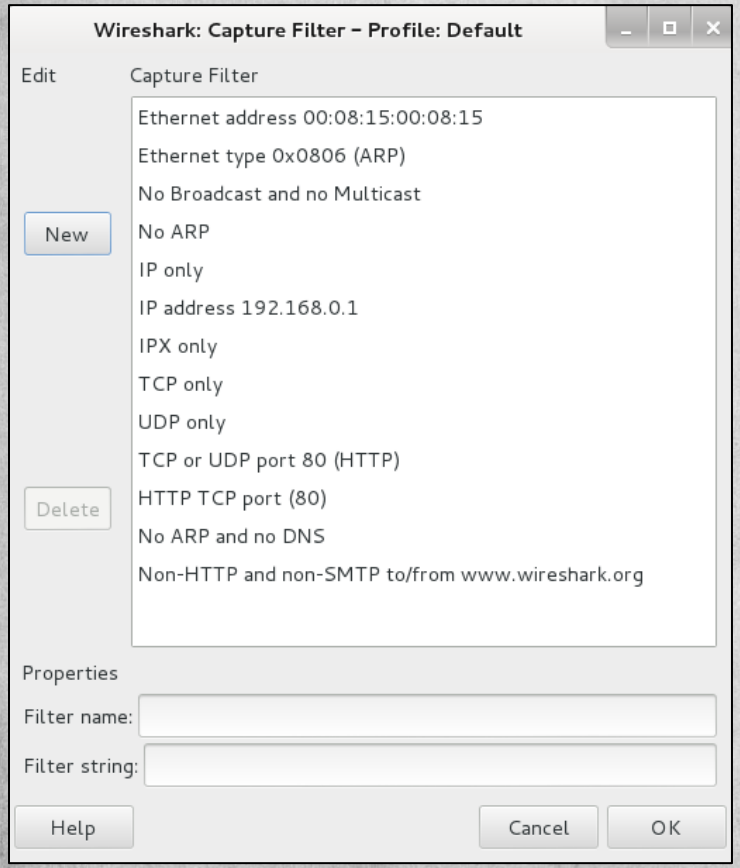
Capture Options



- Interface Selection
- Promiscuous Mode
- Capture Filters
- Capture File Configuration
- Display Options

Wireshark Traffic Filtering

Capture Filtering



- A capture filter can be used to specific traffic, prior to display
- A capture filter is used to reduce the size of the capture files
- Although some of the defaults might be helpful, designing capture filters will provide greater flexibility
- Select “New” to create a new capture filter

Capture Filter Protocols

- Capture filters will filter anything that does not meet minimum traffic requirements
- Capture filters can be configured to identify specific protocols, traffic flows, hosts, and networks

Capture Filter Protocols	
ether - Ethernet	lat – Local Area Transport
fddi - Fiber Distributed Data Interface	sca – Systems Common Architecture
ip – Internet Protocol	moprc – Maintenance Operation Protocol
arp – Address Resolution Protocol	mopdl – Maintenance Operation Protocol
rarp – Reverse Address Resolution Protocol	tcp – Transfer Control Protocol
decnet – Obsolete Decnet Protocol	udp – User Datagram Protocol

Capture Filter Host Filtering

- Capture filters will filter anything that does not meet minimum traffic requirements
- Capture filters can be configured to identify specific protocols, traffic flows, hosts, and networks

Traffic Flow	Capture Filter
ether src host	ether src 58:C2:32:23:F2:01
ether dst host	ether dst 58:C2:32:23:F2:01
src host <IPAddress>	src host 154.28.153.21
dst host <IPAddress>	dst host 154.28.153.21
src net <Network>	src net 134.81.23.0/18
dst net <Network>	dst net 192.168.1.0/24
src host <IPAddress> and dst net <Network>	src host 210.1.3.1 and dst net 192.168.0.0/16

Capture Filter Options

- Capture Filters can also be configured to isolate specific ports and / or protocols
- During a forensic analysis of network traffic, keep in mind standard ports may not be used

Example	Capture Filter
DNS Traffic	port 53
SMTP and HTTP Traffic	port 25 and port 80
No ARP Traffic	not arp
Well Known Ports	tcp portrange 1 - 1024
NetBIOS Ports	port 137 or port 138 or port 139
MSSQL Ports	port 1433 or port 1434

Function	Example
Negation	! or not
Concatenation	&& or and
Alternation	|| or port range

Capture Filter Examples

- Create a capture filter that filters traffic originating from 210.45.21.2 and communicates to an external DNS server

  src host 210.45.21.2 and dst port 53

- Create a capture filter that isolates MSSQL traffic to or from 198.100.32.10

  host 198.100.32.10 and port 3306

- Create a capture filter that filters outbound FTP traffic from the 192.168.20.0/23 network

  net 192.168.20.0/23 and dst port 21

Display Filters

Filter	Explanation
ip.addr == 192.168.20.10	Filters on 1 IP address
ip.addr == 192.168.20.10 && ip.addr == 210.45.21.2	Filters on 2 IP addresses
http || arp	Displays HTTP or ARP Traffic
tcp.port == 25	Displays TCP Port 25
tcp.flags.ack	Displays Packets with ACK Bit Set
tcp.flags.reset == 1	Displays all TCP Resets
http.request.method == GET or http.request	Displays GET requests
tcp contains <SomeWord>	Displays Packet That Contain <SomeWord>
!icmp	Filters out ICMP Traffic
udp contains AB:CD:EF	Filters on the HEX values AB:CD:EF
http.request.method == "GET"	Search only for HTTP GET requests
http.request.full_uri contains "index.php?"	Search for a URI with “index.php?”

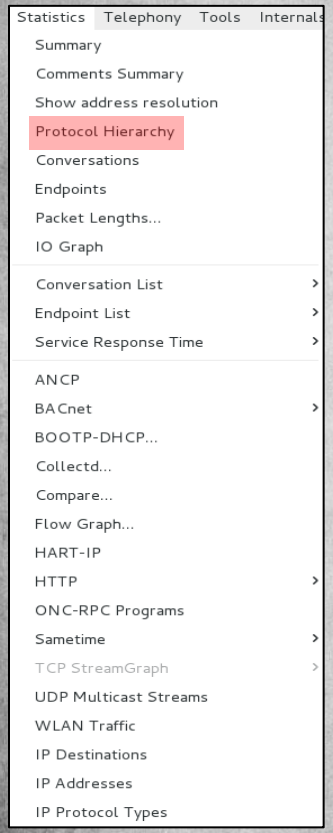
PCAP Analysis

Analysis Exercise

- You are given a packet capture named “Analysis1”, answer the following questions:
  1) What protocols were used during the session?
  2) What kind of files were downloaded during the session?
  3) What display filter can be used filter out file downloads?
  4) List the files downloaded during the session?

Protocol Identification

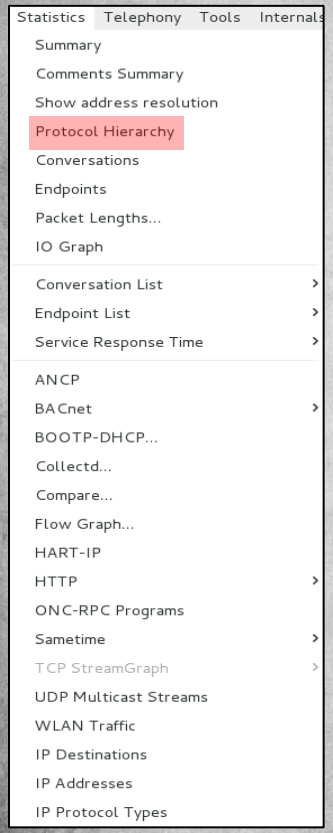
What protocols were used during the session?



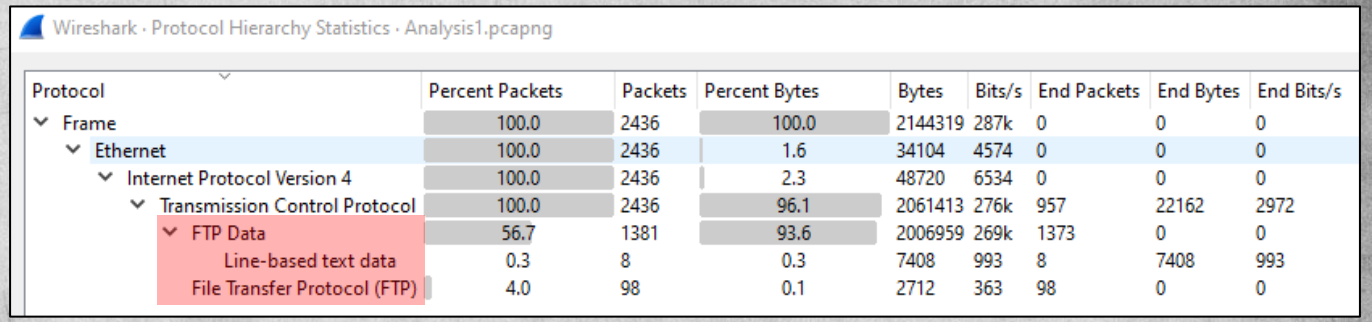
Wireshark · Protocol Hierarchy Statistics · Analysis1.pcapng

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	2436	100.0	2144319	287k	0	0	0
Ethernet	100.0	2436	1.6	34104	4574	0	0	0
Internet Protocol Version 4	100.0	2436	2.3	48720	6534	0	0	0
Transmission Control Protocol	100.0	2436	96.1	2061413	276k	957	22162	2972
FTP Data	56.7	1381	93.6	2006959	269k	1373	0	0
Line-based text data	0.3	8	0.3	7408	993	8	7408	993
File Transfer Protocol (FTP)	4.0	98	0.1	2712	363	98	0	0

File Transfer Identification



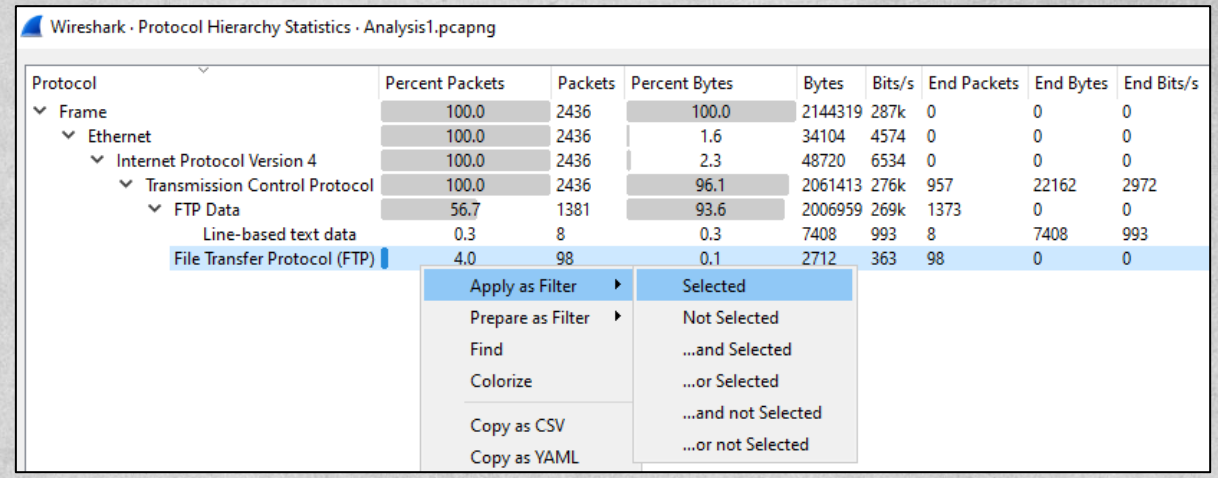
What protocol(s) were used to download files during the session?

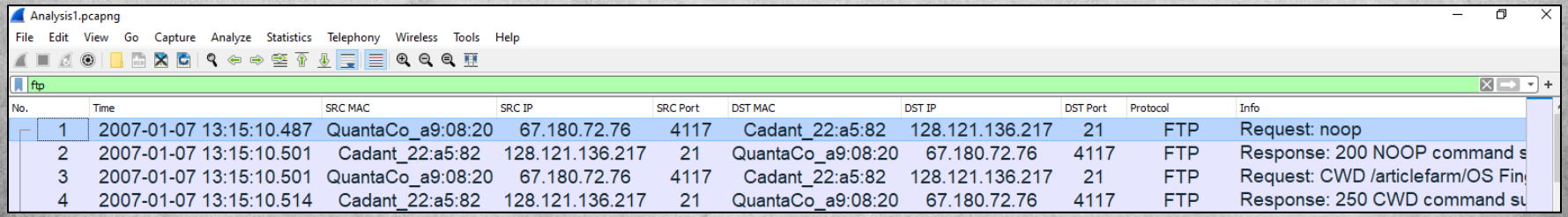


Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	2436	100.0	2144319	287k	0	0	0
Ethernet	100.0	2436	1.6	34104	4574	0	0	0
Internet Protocol Version 4	100.0	2436	2.3	48720	6534	0	0	0
Transmission Control Protocol	100.0	2436	96.1	2061413	276k	957	22162	2972
FTP Data	56.7	1381	93.6	2006959	269k	1373	0	0
Line-based text data	0.3	8	0.3	7408	993	8	7408	993
File Transfer Protocol (FTP)	4.0	98	0.1	2712	363	98	0	0

Display Filters

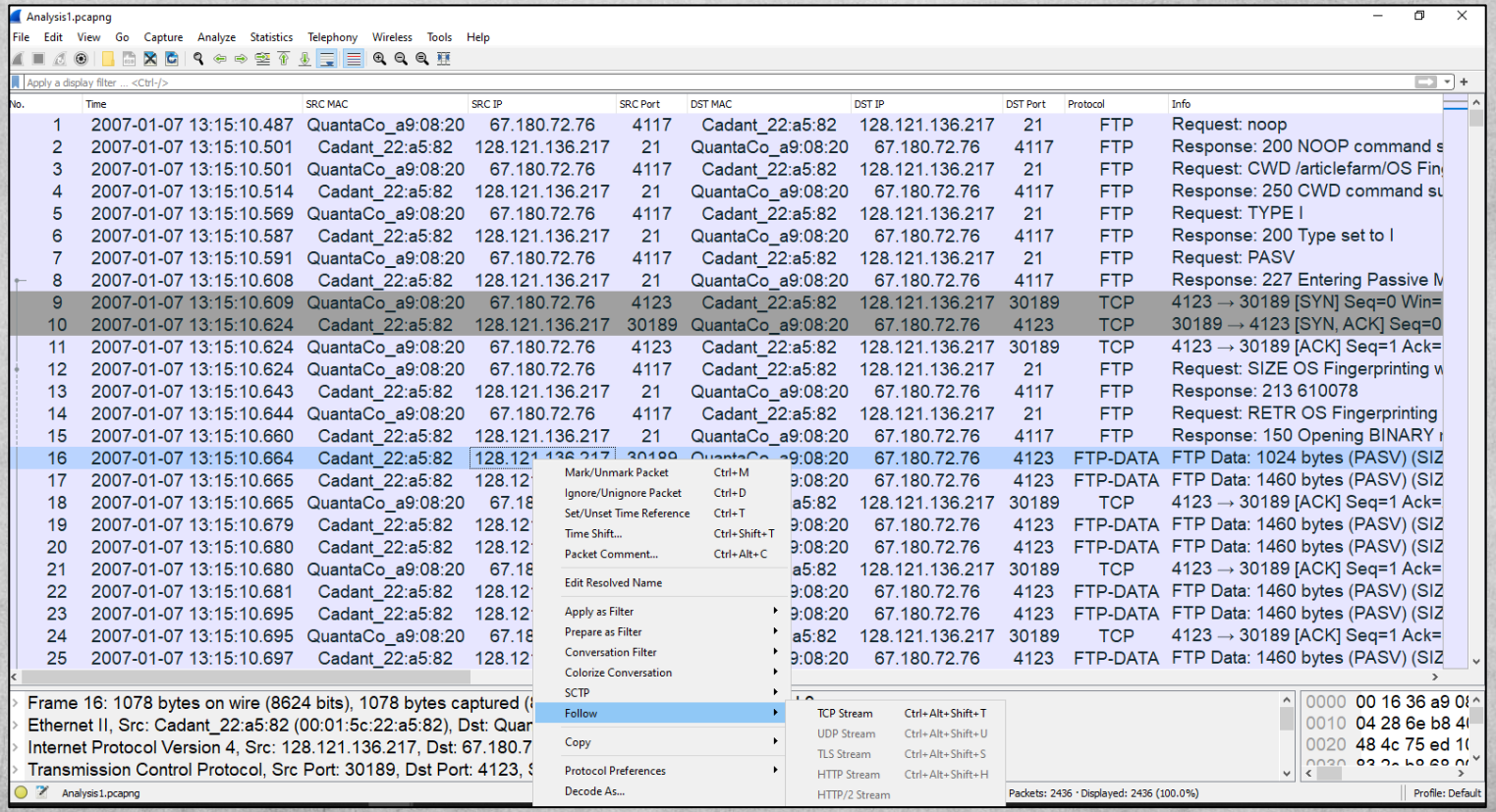
What display filter can be used filter out file downloads?





View Session – TCP Stream

What kind of files were downloaded during the session?



File Analysis

What kind of files were downloaded during the session?

```
00000000  50 4b 03 04 14 00 00 00  08 00 75 67 ef 2e 1f da   PK..... ..ug....
00000010  e1 7f 1d dc 01 00 8e 8c  02 00 14 00 00 00 69 63   ........ ......ic
00000020  6d 70 2d 66 69 6e 67 65  72 70 72 69 6e 74 2e 6a   mp-finge rprint.j
00000030  70 67 ec fd 07 54 94 c1  d2 2e 8c be 08 02 4a 52   pg...T.. ......JR
```

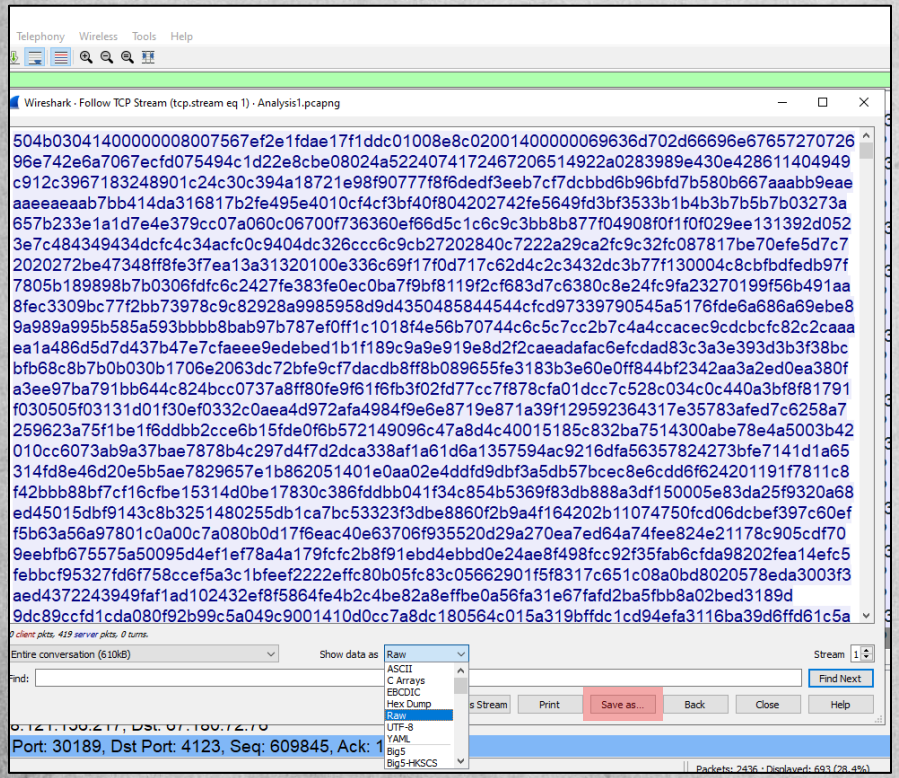
Header

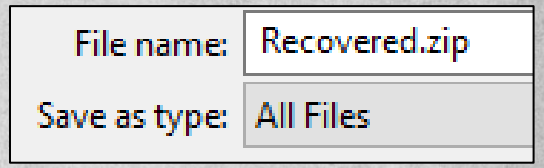
```
00094EF4  69 6e 74 69 6e 67 20 77  69 74 68 20 49 43 4d 50   inting w ith ICMP
00094F04  2e 64 6f 63 50 4b 05 06  00 00 00 00 04 00 04 00   .docPK.. ........
00094F14  31 01 00 00 d7 4d 09 00  00 00                     1....M.. ..
```

Footer

50 4B 03 04		PK..
	ZIP	PKZIP archive file (Ref. 1 | Ref. 2)
		Trailer: *filename* 50 4B *17 characters* 00 00 00
		Trailer: (*filename* PK *17 characters* . . .)

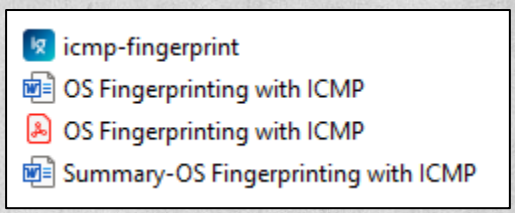
File Recovery – Raw Data

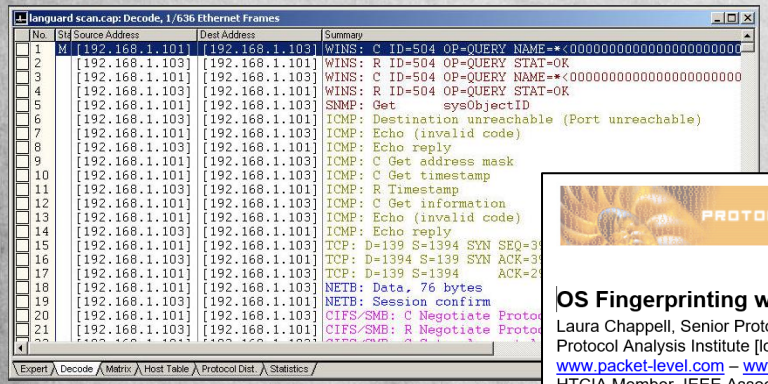


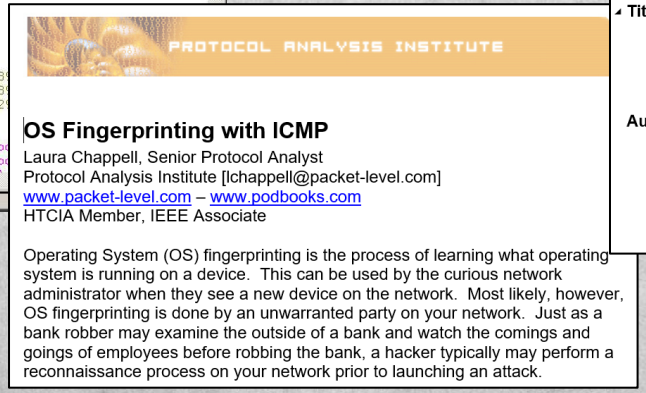


File Recovery – Raw Data

List the files downloaded during the session?



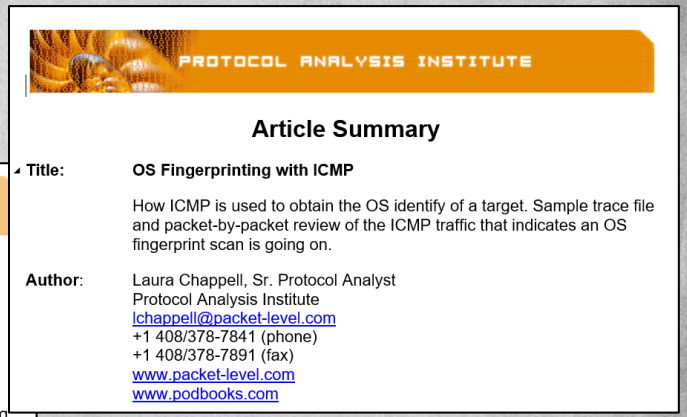



PROTOCOL ANALYSIS INSTITUTE

OS Fingerprinting with ICMP

Laura Chappell, Senior Protocol Analyst
Protocol Analysis Institute [lchappell@packet-level.com]
www.packet-level.com – www.podbooks.com
HTCIA Member, IEEE Associate

Operating System (OS) fingerprinting is the process of learning what operating system is running on a device. This can be used by the curious network administrator when they see a new device on the network. Most likely, however, OS fingerprinting is done by an unwarranted party on your network. Just as a bank robber may examine the outside of a bank and watch the comings and goings of employees before robbing the bank, a hacker typically may perform a reconnaissance process on your network prior to launching an attack.


PROTOCOL ANALYSIS INSTITUTE

Article Summary

Title: **OS Fingerprinting with ICMP**

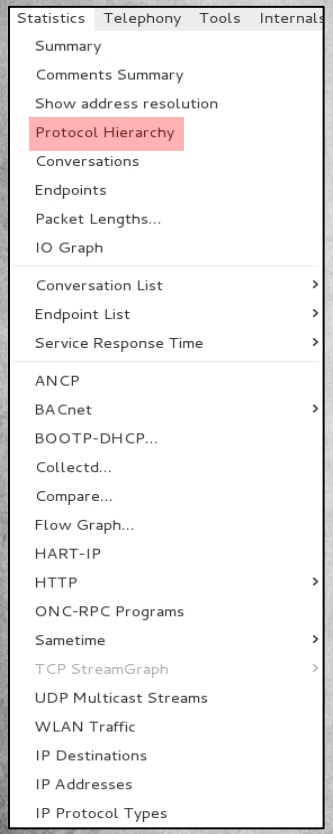
How ICMP is used to obtain the OS identify of a target. Sample trace file and packet-by-packet review of the ICMP traffic that indicates an OS fingerprint scan is going on.

Author: Laura Chappell, Sr. Protocol Analyst
Protocol Analysis Institute
lchappell@packet-level.com
+1 408/378-7841 (phone)
+1 408/378-7891 (fax)
www.packet-level.com
www.podbooks.com

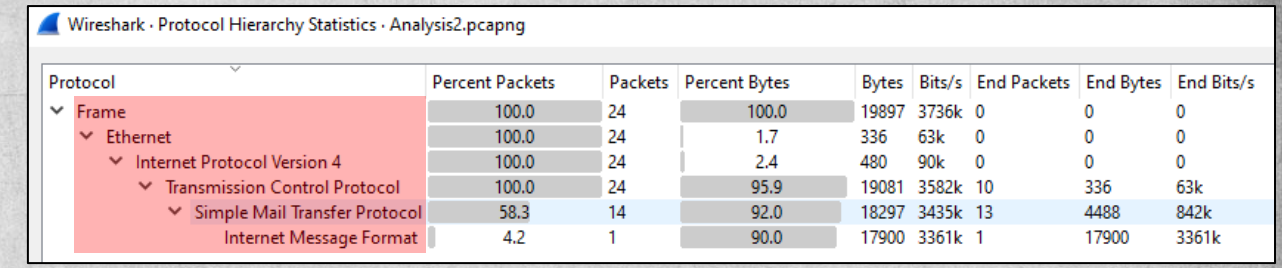
Analysis Exercise

- You are given a packet capture file named “Analysis2”, answer the following questions:
  1) What protocols were used during the session?
  2) Can we identify users in this session?
  3) Were any files transferred between these users?

Protocol Statistics



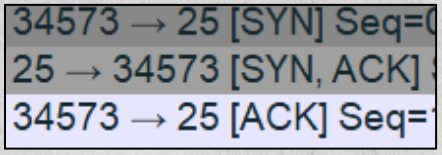
What protocols were used during the session?



Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	24	100.0	19897	3736k	0	0	0
Ethernet	100.0	24	1.7	336	63k	0	0	0
Internet Protocol Version 4	100.0	24	2.4	480	90k	0	0	0
Transmission Control Protocol	100.0	24	95.9	19081	3582k	10	336	63k
Simple Mail Transfer Protocol	58.3	14	92.0	18297	3435k	13	4488	842k
Internet Message Format	4.2	1	90.0	17900	3361k	1	17900	3361k

User Identification

Can we identify users in this session?



Locate the three-way handshake

Analysis2.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/>

No.	Time	SRC MAC	SRC IP	SRC Port	DST MAC	DST IP	DST Port	Protocol	Info
1	2012-09-28 11:12:04.884	00:00:00_00:00:00	10.10.10.10	34573	00:00:00_00:00:00	10.10.10.25	25	TCP	34573 → 25 [SYN] Seq=0 Win=32792 L
2	2012-09-28 11:12:04.886	00:00:00_00:00:00	10.10.10.25	25	00:00:00_00:00:00	10.10.10.10	34573	TCP	25 → 34573 [SYN, ACK] Seq=0 Ack=1 \
3	2012-09-28 11:12:04.887	00:00:00_00:00:00	10.10.10.10	34573	00:00:00_00:00:00	10.10.10.25	25	TCP	34573 → 25 [ACK] Seq=1 Ack=1 Win=3
4	2012-09-28 11:12:04.889	00:00:00_00:00:00	10.10.10.25	25	00:00:00_00:00:00	10.10.10.10	34573	SMTP	S: 220 JSmith-desktop ESMTP Postfix (
5	2012-09-28 11:12:04.891	00:00:00_00:00:00	10.10.10.10	34573	00:00:0		25	TCP	34573 → 25 [ACK] Seq=1 Ack=44 Win=
6	2012-09-28 11:12:04.893	00:00:00_00:00:00	10.10.10.10	34573	00:00:0		25	SMTP	C: EHLO JSmith-desktop
7	2012-09-28 11:12:04.894	00:00:00_00:00:00	10.10.10.25	25	00:00:0		34573	TCP	25 → 34573 [ACK] Seq=44 Ack=22 Wir
8	2012-09-28 11:12:04.897	00:00:00_00:00:00	10.10.10.25	25	00:00:0		34573	SMTP	S: 250-JSmith-desktop | PIPELINING |
9	2012-09-28 11:12:04.899	00:00:00_00:00:00	10.10.10.10	34573	00:00:0		25	SMTP	C: MAIL FROM:<JSmith@comcast.net>
10	2012-09-28 11:12:04.901	00:00:00_00:00:00	10.10.10.25	25	00:00:0		34573	SMTP	S: 250 2.1.0 Ok
11	2012-09-28 11:12:04.903	00:00:00_00:00:00	10.10.10.10	34573	00:00:0		25	SMTP	C: RCPT TO:<jesse@myheart.com>
12	2012-09-28 11:12:04.905	00:00:00_00:00:00	10.10.10.25	25	00:00:0		34573	SMTP	S: 250 2.1.5 Ok
13	2012-09-28 11:12:04.907	00:00:00_00:00:00	10.10.10.10	34573	00:00:0		25	SMTP	C: DATA
14	2012-09-28 11:12:04.909	00:00:00_00:00:00	10.10.10.25	25	00:00:0		34573	SMTP	S: 354 End data with <CR><LF>.<CR><
15	2012-09-28 11:12:04.912	00:00:00_00:00:00	10.10.10.10	34573	00:00:0		25	SMTP	C: DATA fragment, 4096 bytes
16	2012-09-28 11:12:04.913	00:00:00_00:00:00	10.10.10.25	25	00:00:0		34573	TCP	4573 [ACK] Seq=246 Ack=4185
17	2012-09-28 11:12:04.915	00:00:00_00:00:00	10.10.10.10	34573	00:00:0		25	SMTP	Smith@comcast.net, subject: tes
18	2012-09-28 11:12:04.917	00:00:00_00:00:00	10.10.10.25	25	00:00:0		34573	TCP	4573 [ACK] Seq=246 Ack=1799
19	2012-09-28 11:12:04.919	00:00:00_00:00:00	10.10.10.25	25	00:00:0		34573	SMTP	2.0.0 Ok: queued as 4CF931B5
20	2012-09-28 11:12:04.921	00:00:00_00:00:00	10.10.10.10	34573	00:00:0		25	TCP	T
21	2012-09-28 11:12:04.923	00:00:00_00:00:00	10.10.10.25	25	00:00:00_00:00:00	10.10.10.10	34573	SMTP	S: 221 2.0.0 Bye

Mark/Unmark Packet Ctrl+M
Ignore/Unignore Packet Ctrl+D
Set/Unset Time Reference Ctrl+T
Time Shift... Ctrl+Shift+T
Packet Comment... Ctrl+Alt+C
Edit Resolved Name
Apply as Filter
Prepare as Filter
Conversation Filter
Colorize Conversation
SCTP
Follow
Copy
Protocol Preferences
Decode As...
Show Packet in New Window

TCP Stream Ctrl+Alt+Shift+T
UDP Stream Ctrl+Alt+Shift+U
TLS Stream Ctrl+Alt+Shift+S
HTTP Stream Ctrl+Alt+Shift+H
HTTP/2 Stream
QUIC Stream

User Identification

Can we identify users during this session?

```
220 JSmith-desktop ESMTP Postfix (Ubuntu)
EHLO JSmith-desktop
250-JSmith-desktop
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
MAIL FROM:<JSmith@comcast.net>
250 2.1.0 Ok
RCPT TO:<jesse@myheart.com>
250 2.1.5 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
Date: Fri, 28 Sep 2012 11:33:17 -0400
To: jesse@myheart.com
From: JSmith@comcast.net
Subject: test Fri, 28 Sep 2012 11:33:17 -0400
X-Mailer: swaks v20061116.0 jetmore.org/john/code/#swaks
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="----=_MIME_BOUNDARY_000_11181"
```

File Recovery

Were any files transferred between the users?

```
Date: Fri, 28 Sep 2012 11:33:17 -0400
To: jesse@myheart.com
From: JSmith@comcast.net
Subject: test Fri, 28 Sep 2012 11:33:17 -0400
X-Mailer: swaks v20061116.0 jetmore.org/john/code/#swaks
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="----=_MIME_BOUNDARY_000_11181"

------=_MIME_BOUNDARY_000_11181
Content-Type: text/plain

This is a test mailing
------=_MIME_BOUNDARY_000_11181
Content-Type: application/octet-stream
Content-Transfer-Encoding: BASE64
Content-Disposition: attachment

JVBERi0xLjQKJcfsj6IKNSAwIG9iago8PC9MZW5ndGggNiAwIFIvRmlsdGVyIC9GbGF0ZURIY29k
ZT4+CnN0cmVhbQp4nKVZ23LbOBJ911fgLVSVhRAAr3nLOM6Wd32ZiZVJtpx5oCna5oYiFZKy4n+Y
X/E/TjdAgABlxdlKuVy2RKDR19OnwW/Ep4z4+DP8zdez1x9ictfNfPIv+L2bfZsxuYAMf/l1+W0J
ixL8Ynk7U/sYiTmJU0GTlCzXMxamNAnIcjfzVk3Ht3X5/YvHvswJ95m/8IOFL4jzYL7832zhU85g
ew7SU5BO0xAPEBFZns28i7fzV6AQPGUhDVMBhy5XM+/8BLeO6nCfBn5MFozjY99RgTyRd5dXr8/f
4hZOQxSWhDRRoq69437OQ99rPs7/Wv5bncUjGkVpqldcnH4mWb0iD2VekIc5aOL7sfCKtstIX5jP
33vyd1WQ26ZdZz3Jm1oKBPtYqAwIaRQOBjwUqI08LKU8ScFt6kHbF62yzXLGQjBBAwb2RaA3Lrv6
78Xl71enV2op8wc/RL72wWj+dbPpy6bu/iLXTzk5vrz48/zy3Ql+asj707MTQinFTzU5vZCfLz8u
zfcoHwVCKFAYB6m/KPDVxDSpsjbr3cnV8YfT35enlxdugCEddHBpjP9AgJf3BXmnrdxk+dfsriBl
```

File Recovery – Encoding

Were any files transferred between the users?

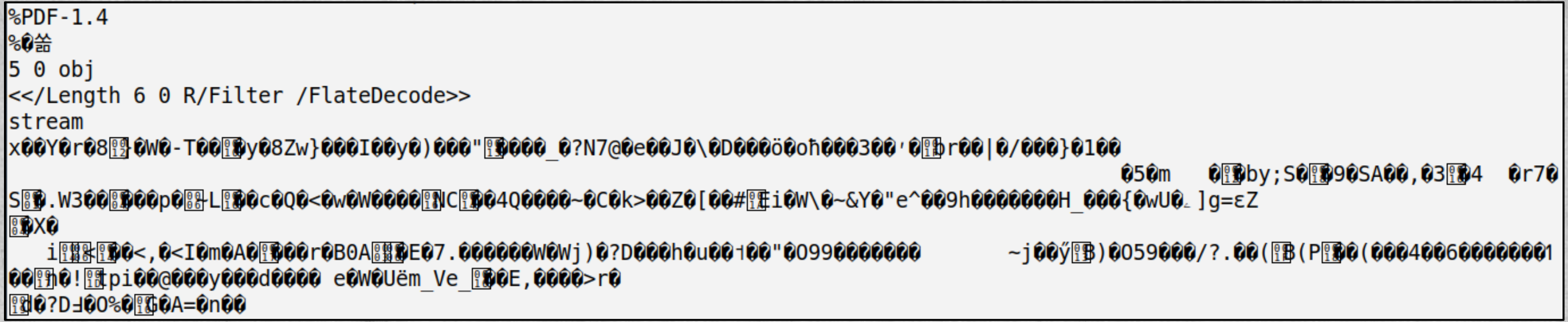
```
Date: Fri, 28 Sep 2012 11:33:17 -0400
To: jesse@myheart.com
From: JSmith@comcast.net
Subject: test Fri, 28 Sep 2012 11:33:17 -0400
X-Mailer: swaks v20061116.0 jetmore.org/john/code/#swaks
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="----=_MIME_BOUNDARY_000_11181"

------=_MIME_BOUNDARY_000_11181
Content-Type: text/plain

This is a test mailing
------=_MIME_BOUNDARY_000_11181
Content-Type: application/octet-stream
Content-Transfer-Encoding: BASE64
Content-Disposition: attachment

JVBERi0xLjQKJcfsj6IKNSAwIG9iago8PC9MZW5ndGggNiAwIFIvRmlsdGVyIC9GbGF0ZURlY29k
ZT4+CnN0cmVhbQp4nKVZ23LbOBJ911fgLVSVhRAAr3nLOM6Wd32ZiZVJtpx5oCna5oYiFZKy4n+Y
X/E/TjdAgABlxdlKuVy2RKDR19OnwW/Ep4z4+DP8zdez1x9ictfNfPlv+L2bfZsxuYAMf/I1+W0J
ixL8Ynk7U/sYiTmJU0GTlCzXMxamNAnIcjfzVk3Ht3X5/YvHvswJ95m/8IOFL4jzYL7832zhU85g
ew7SU5BO0xAPEBFZns28i7fzV6AQPGUhDVMBhy5XM+/8BLeO6nCfBn5MFozjY99RgTyRd5dXr8/f
4hZOQxSWhDRRoq69437OQ99rPs7/Wv5bncUjGkVpqIdcnH4mWb0iD2VekIc5aOL7sfCKtstIX5jP
33vyd1WQ26ZdZz3Jm1oKBPtYqAwIaRQOBjwUqI08LKU8ScFt6kHbF62yzXLGQjBBAwb2RaA3Lrv6
78XI71enV2op8wc/RL72wWj+dbPpy6bu/iLXTzk5vrz48/zy3Ql+asj707MTQinFTzU5vZCfLz8u
zfcoHwVCKFAYB6m/KPDVxDSpsjbr3cnV8YfT35enlxdugCEddHBpjP9AgJf3BXmnrdxk+dfsriBl
```

```
RkZCRjhFRDgzM0RFNTZEPjxBQ0U1OTkyODQzQTY3MUM4NUZGQkY4RUQ4MzNERTU2RD5dCj4+CnN0
YXJ0eHJlZgoxMTg2MAolJUVPRgo=" | base64 -d
```

File Recovery – Encoding

Were any files transferred between the users?


```
RkZCRjhFRDgzM0RFNTZEPjxBQ0U1OTkyODQzQTY3MUM4NUZGQkY4RUQ4MzNERTU2RD5dCj4+CnN0
YXJ0eHJlZgoxMTg2MAolJUVPRgo=" | base64 -d
```


```
+DOKBTaM7ivmphkDuPKR1yZ74zD2y+TnIoRL40QT9RC37WSsZG8kL/ubxYgjhqc7IU5RuC8OhhhJ
YXJ0eHJlZgoxMTg2MAolJUVPRgo=" | base64 -d > Recovered.pdf
```

File Recovery

Were any files transferred between the users?

dos2unix(1) 2010-04-03 dos2unix(1)

NAME

dos2unix − DOS/MAC to UNIX and vice versa text file format converter

SYNOPSIS

```
dos2unix [options] [-c CONVMODE] [-o FILE ...] [-n INFILE OUTFILE ...]
unix2dos [options] [-c CONVMODE] [-o FILE ...] [-n INFILE OUTFILE ...]
```

DESCRIPTION

The Dos2unix package includes utilities `dos2unix` and `unix2dos` to convert plain text files in DOS or MAC format to UNIX format and vice versa. Binary files and non-regular files, such as soft links, are automatically skipped, unless conversion is forced.

Dos2unix has a few conversion modes similar to dos2unix under SunOS/Solaris.

In DOS/Windows text files line endings exist out of a combination of two characters: a Carriage Return (CR) followed by a Line Feed (LF). In Unix text files line endings exists out of a single Newline character which is equal to a DOS Line Feed (LF) character. In Mac text files, prior to Mac OS X, line endings exist out of a single Carriage Return character. Mac OS X is Unix based and has the same line endings as Unix.

Summary

- Network forensics takes practice!
- The key is understanding how the protocols are supposed to work, versus how they are used
- Protocol analysis is greatly aided using RFC’s
- Analysis becomes more complex as encoding and encryption are used

References

- PCAP Files
  - ✓ http://wiki.wireshark.org/SampleCaptures
  - ✓ http://www.netresec.com/?page=PcapFiles
  - ✓ http://chrissanders.org/packet-captures
- Wireshark Filters
  - ✓ http://wiki.wireshark.org/CaptureFilters
  - ✓ http://wiki.wireshark.org/DisplayFilters
- Libpcap
  - ✓ http://books.gigatux.nl/mirror/networksecuritytools/0596007949/networkst-CHP-10-SECT-1.html