SAND2010-4522P

# Introduction to TCP/IP V4 Networking

**J. Larry Dishman**
**Senior Member of Technical Staff**
**Sandia National Laboratories**

Sandia National Laboratories is a multi-program laboratory operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin company, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

# Class Objectives

- What is a network
- Basic network functions
- TCP/IP fundamentals
- OSI vs. TCP/IP model
- Common application protocols
- Basic troubleshooting

# Table of Content


- Chapter 1: History & Models
- Chapter 2: Local Area Network (LAN)
- Chapter 3: Ethernet & Connections
- Chapter 4: Wireshark & Protocols
- Chapter 5: Basic Troubleshooting

# History & Models

- Chapter 1: History & Models
- Chapter 2: Local Area Network (LAN)
- Chapter 3: Ethernet & Connections
- Chapter 4: Wireshark & Protocols
- Chapter 5: Basic Troubleshooting

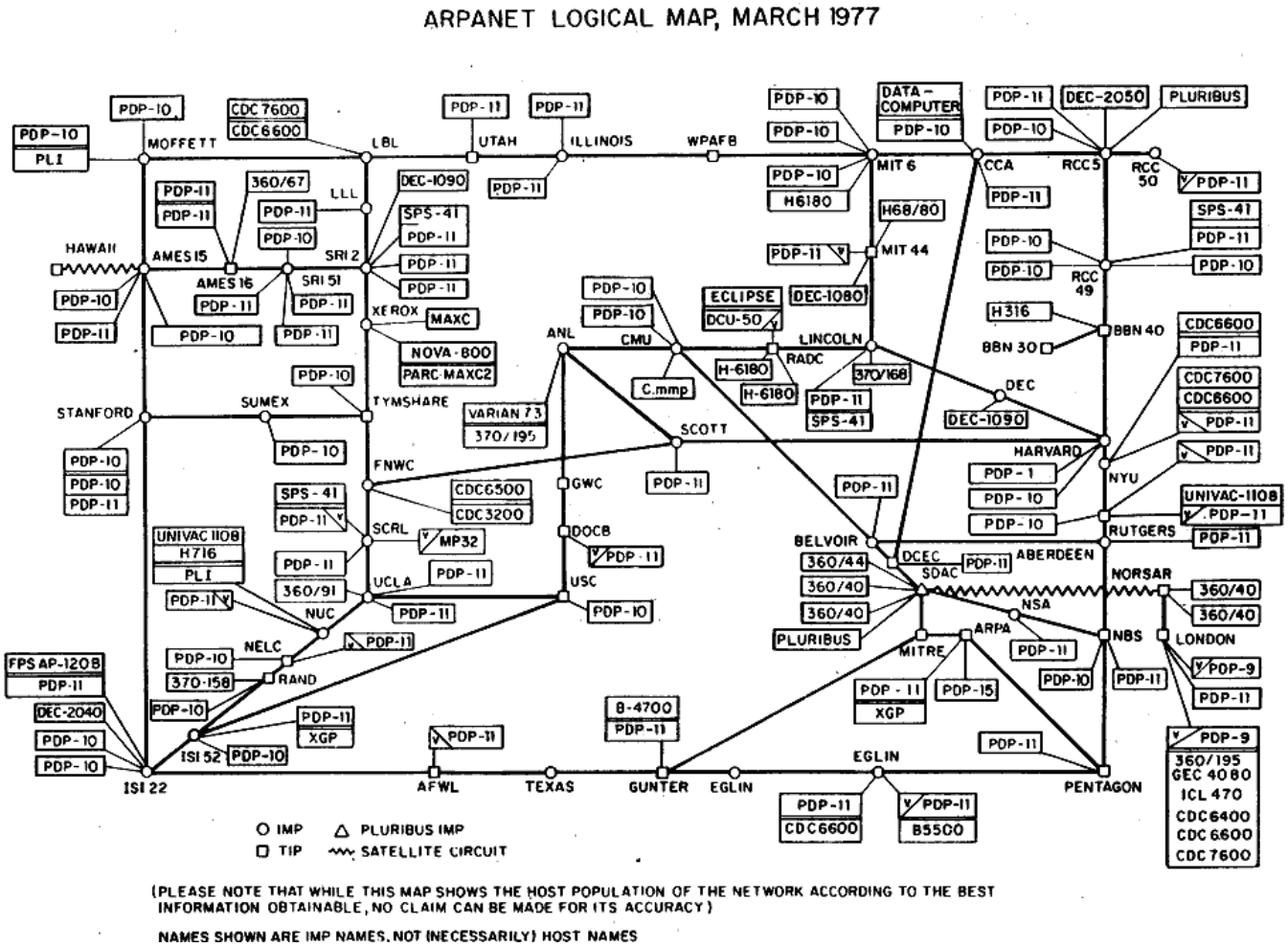
# History of the Internet

- 1962, Advanced Research Projects Agency (ARPA) wanted a way for all of its mainframes to communicate
- 1968, Packet switched network connected 4 sites at 50Kb/s. UCLA, UC Santa Barbra, Stanford and University of Utah. (ARPANet)
- 1972, ARPANet expands to 32 nodes and email program created

# Cont.

- 1972, ARPA renamed to Defense Advanced Research Project Agency, DARPA
- 1973, Transmission Control Protocol and Internet Protocol work started, TCP/IP
- 1976, Ethernet development started for Local Area Network, LAN
- 1976, Atlantic packet satellite network (SATNET) joined US and Europe

# Would you Believe




(PLEASE NOTE THAT WHILE THIS MAP SHOWS THE HOST POPULATION OF THE NETWORK ACCORDING TO THE BEST INFORMATION OBTAINABLE, NO CLAIM CAN BE MADE FOR ITS ACCURACY)

NAMES SHOWN ARE IMP NAMES, NOT (NECESSARILY) HOST NAMES

Sandia National Laboratories

# Cont.

- 1983, ARPANet split into public (ARPANet) and private (Milnet) network
- 1983, TCP/IP became the predominant protocol on ARPANet
- 1983, University of Wisconsin developed a way to use IP addresses or system names to identify computers, DNS

# National Science Foundation (NSF)

- 1985, National Science Foundation (NSF) upgrades backbone to T-1 (1.544 Mbps) → T-3 (45Mbps) → 145 Mbps (ATM)
- Two protocols introduced, Network News Transfer Protocol (NNTP) and Mail Exchange (MX) records added to DNS
- InterNIC came into being to maintain directory, registration and information services

# Cont

- 1990, Berners-Lee specified HTML and wrote the browser and server software (CERN)
- 1993, Mosaic web browser releases (NCSA)
- March 1999, “During my service in the United States Congress, I took the initiative in creating the Internet.” Al Gore

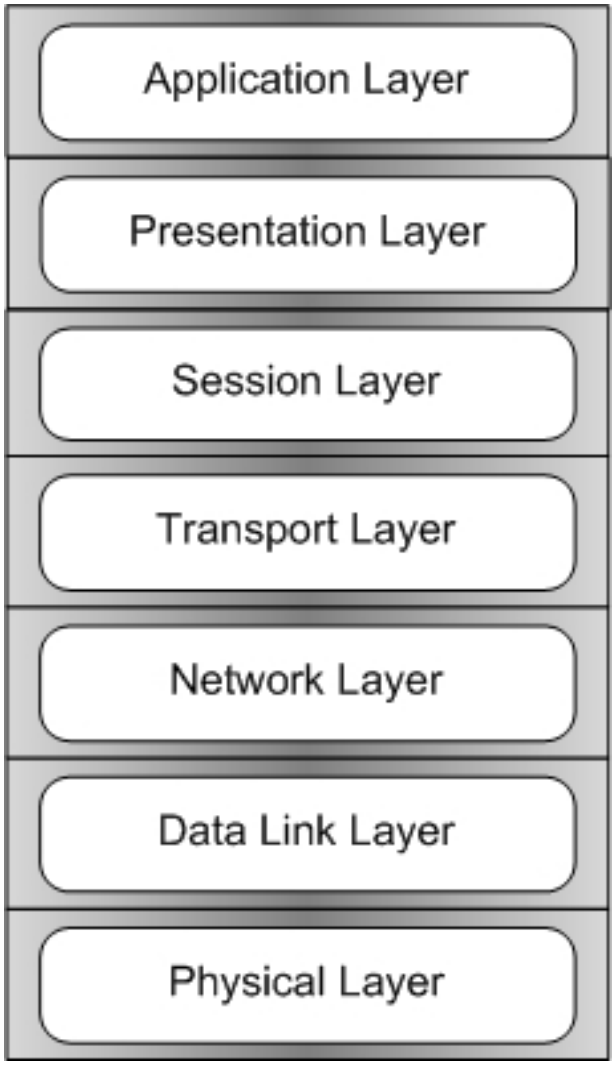
# Standards

- International Engineering Task Force (IETF) heads all research activities for TCP/IP including proposed changes
- Request For Comment (RFC) is how everything gets implemented
- Anyone can submit an RFC
- They can be viewed at http://www.ietf.org/
- RFC 2274 deals with RTFM (Real-Time-Flow-Monitor)
- International Organization for Standards (ISO), manufacturer, vendor, consumer and engineering driven

# Open System Interconnection Model (OSI)

- 1980, ISO developed a network reference model to help vendors create interoperability
- Used 7 layer model to describe how information is passed from the application to/from one host to another
- As the information passes through the layers header information is added/removed
- No present day operating system fully implements the OSI model

# OSI Model




Sandia National Laboratories

# OSI Model

| |
|---|
| **Application**<br>*Application interface to network* |
| **Presentation**<br>*Data compression, transformation, syntax and presentation* |
| **Session**<br>*Sets up/manages sessions between users* |
| **Transport**<br>*Creates/manages connections between sender/receiver* |
| **Network**<br>*Controls routing of information and packed congestion control* |
| **Data-Link**<br>*Ensures error free transmission by dividing data into frames and acknowledges receipt of frames* |
| **Physical**<br>*Transmits raw bits over communication medium* |

# OSI Simplest Terms

1. Physical – hardware layer
2. Data-Link – Interface to the physical layer
3. Network – Routes packets through the network
4. Transport – Ensures reliable communications
5. Session – Negotiates communication between two nodes on the network
6. Presentation – Presents data to the application layer in a useable format
7. Application – Interface to end-user applications

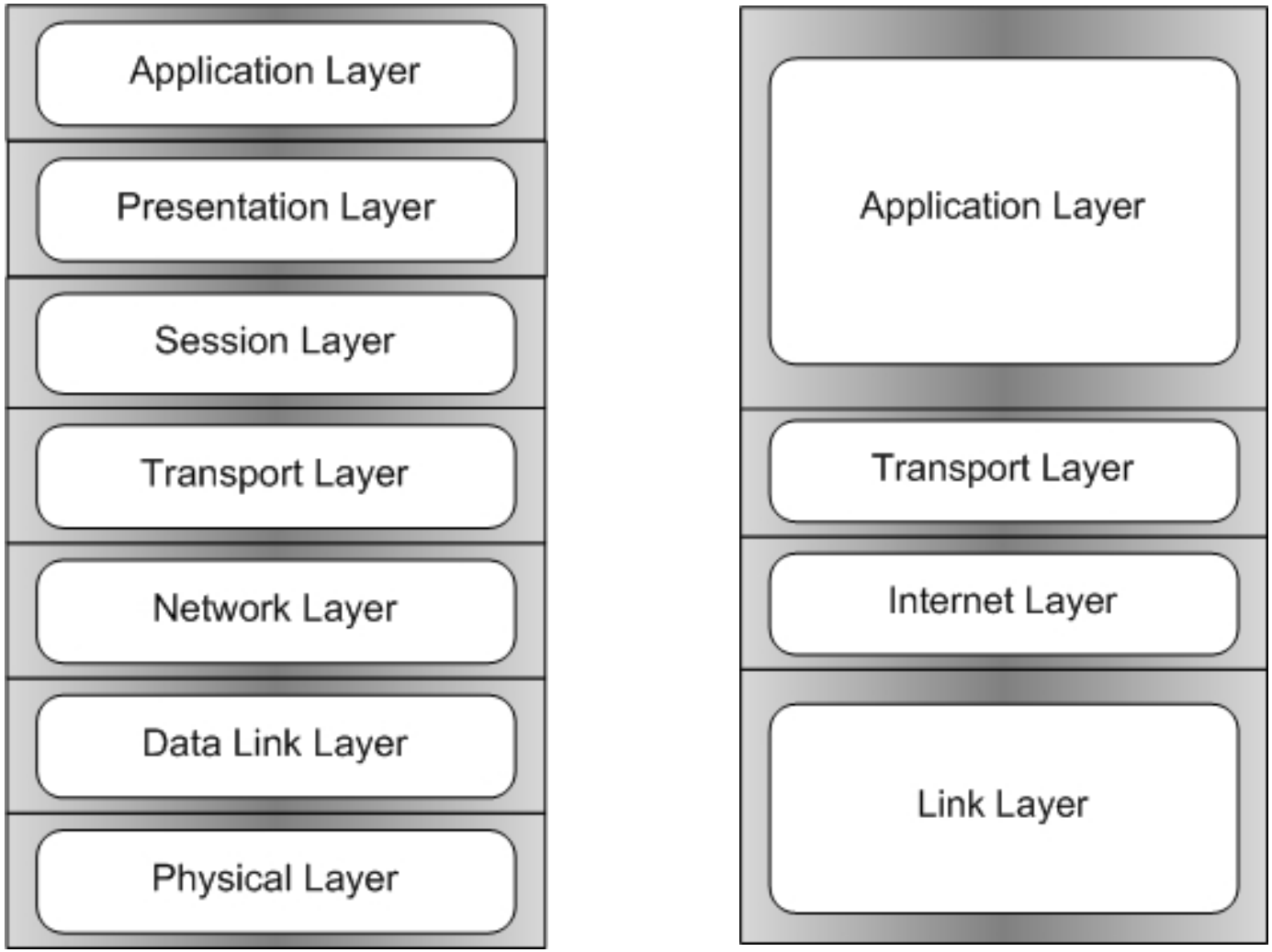
# OSI Model

- Remember: All People Seem To Need Data Processing = Application Presentation Session Transport Network Data-Link Physical

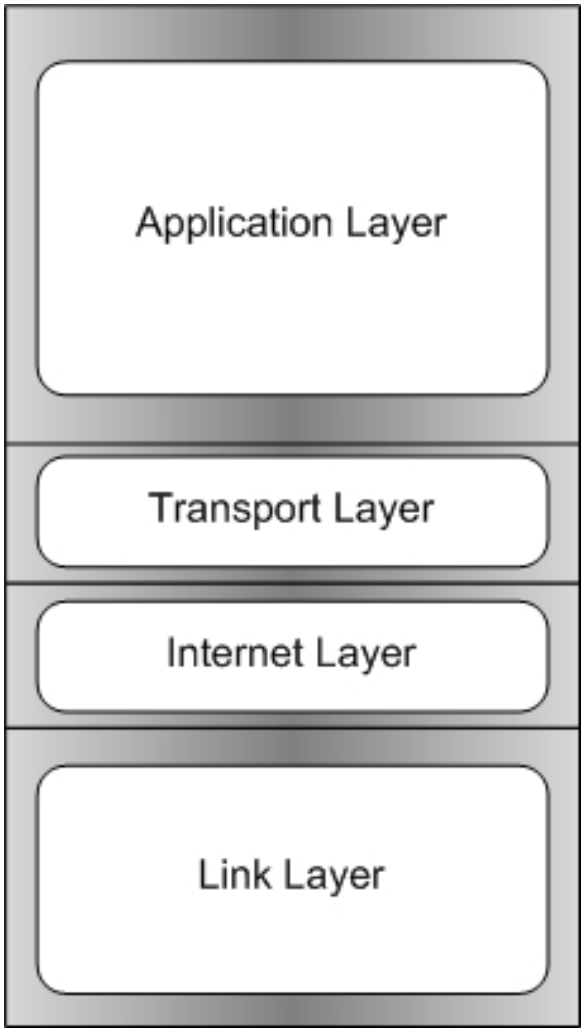
# TCP/IP Model

- Four layer model
- Application, Transport, Internet and Link
- TCP operates on the transport layer
- IP operates on the Internet layer

# OSI to TCP/IP Mapping

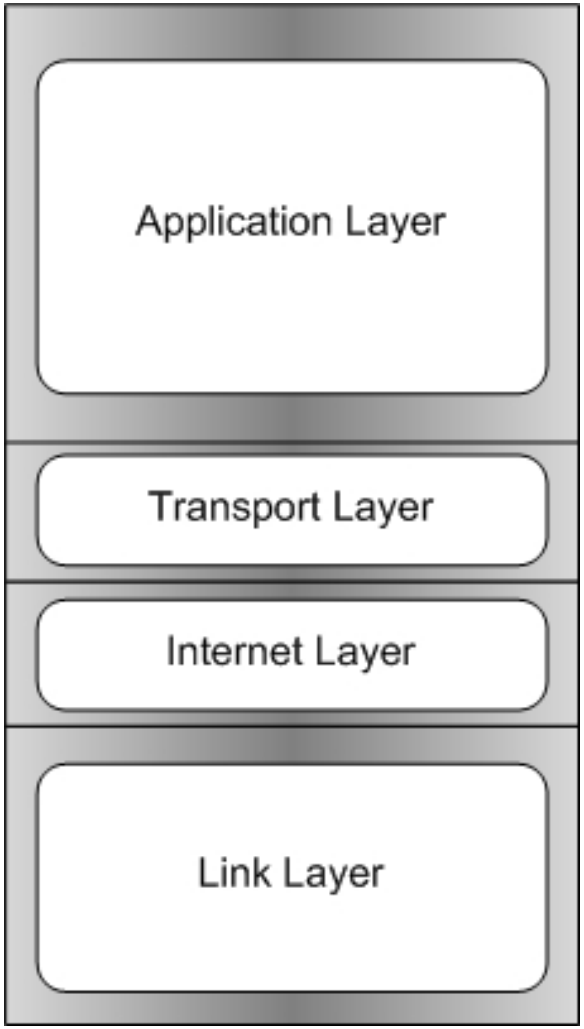


# Link Layer



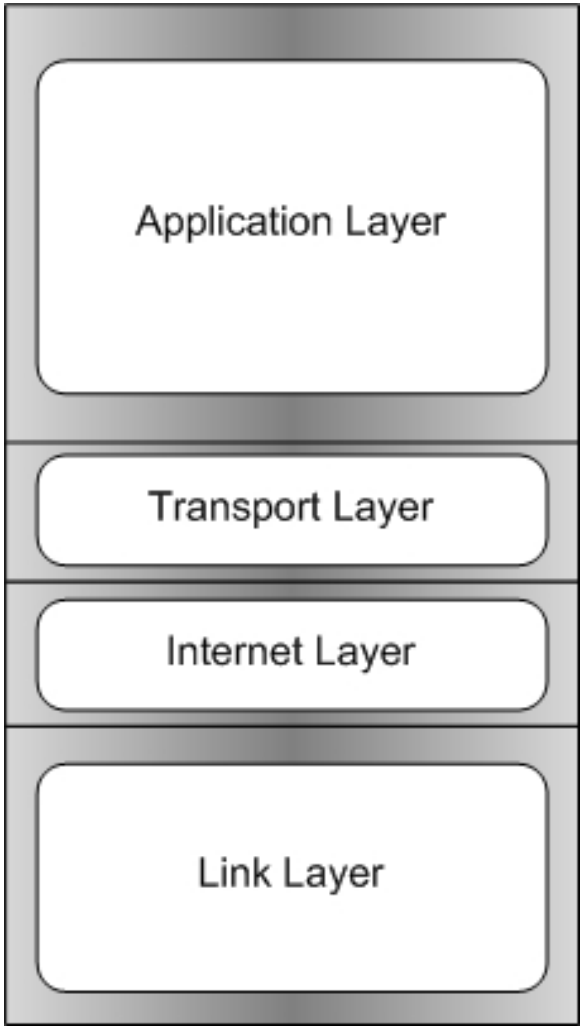

- Sometimes called: data-link or physical layer
- Specifies electrical and physical connection to the LAN
- Allows a host to send and receive bits
- Defines how the cable is attached to the Network Interface Card (NIC)
- Provides data encoding and bit synchronization
- DSL, RJ45, FDDI…
- Allow hosts on a common Data-Link layer to exchange frames – the Local Area Network
- Arrange bits from the Physical Layer into logical sequences
- Ethernet, Token Ring, PPP, ATM, ARP, Frame Relay…
- Data Type: Frames

# Internet Layer



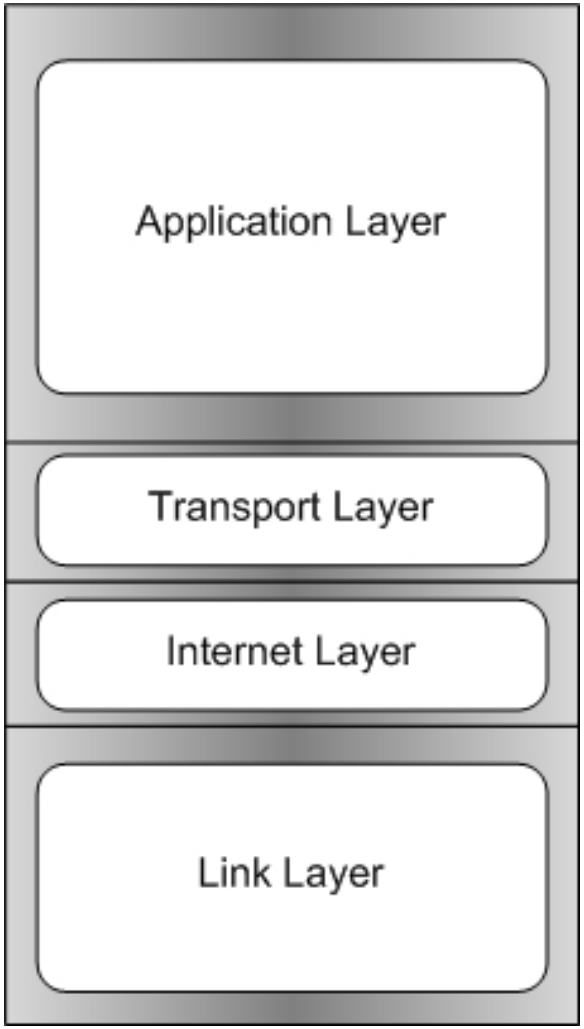

- Sometimes called the network layer
- Offers communication between LANs
- Delivery of packets end to end and implements a logical addressing scheme
- Examines the destination address, if it is its own address, it passes the data up the stack
- IP, IPX, ICMP, IPsec…

# Transport Layer



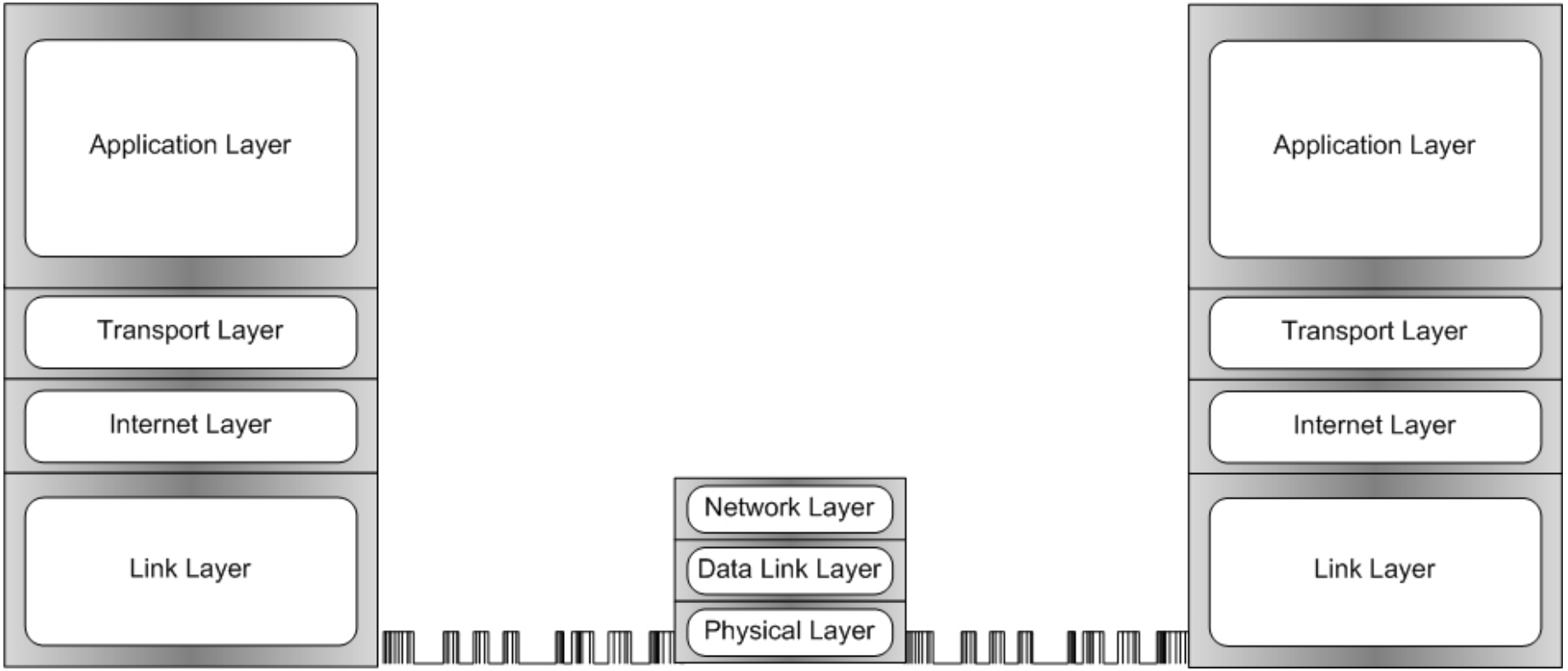

- Offers persistent connections
  - Specifies segmentation and reassembly of large sequences of data, and retransmits dropped segments
  - Multiple connections between hosts are possible
- Provides end-to-end reliable data transfer services to the upper layers
- TCP, UDP, PPTP, L2TP…

# Application Layer




- Offers persistent session-connections
  - Synchronization points offer session recovery if interrupted
  - May involve the use of multiple transport-connections
- Establishes, manages and terminates the connections between the local and remote application
- Manages the transaction sequencing until message is complete
- NetBIOS, AppleTalk SAP…
- SSL, TSL, JPEG, ASCII, GIF, TIFF…
- Client-Server software communication from one computer to another
- FTP, SMTP, DNS, Web Browser…
- Data Type: User Data, Session streams, Encoded Data

# TCP/IP Model at Work



# TCP/IP 4-Layer Model Review

- Application - Defines TCP/IP application protocols and how host programs interface with transport layer services to use the network.
  - HTTP, Telnet, FTP, TFTP, SNMP, DNS, SMTP, X Windows, other application protocols
- Transport - Provides communication session management between host computers. Defines the level of service and status of the connection used when transporting data.
  - TCP, UDP
- Internet - Packages data into IP packets/datagrams, which contain source and destination address information that is used to forward the packets/datagrams between hosts and across networks. Performs routing of IP packets/datagrams.
  - IP, ICMP, ARP, IPSEC
- Link Layer (Network interface) - Specifies details of how data is physically sent through the network, including how bits are electrically signaled by hardware devices that interface directly with a network medium, such as coaxial cable, optical fiber, or twisted-pair copper wire.
  - Ethernet, Token Ring, FDDI, X.25, Frame Relay, RS-232

# Internet Assigned Numbers Authority (IANA)

- Http://www.iana.org/
- Manages the DNS Root Zone and top level domains
- Coordinates the global IP space, allocates these to Regional Internet Registries
- Central repository for protocol name and number registries used in many Internet protocols

# International Corporation for Assigned Names and Numbers (ICANN)

- Manages DNS coordination
- Make sure all addresses are unique
- Sets minimum standards and accreditation for domain name registrars
- Operates http://www.internic.net/ a website for obtaining domain registration information

# Local Area Network (LAN)

- Chapter 1: History & Models
- Chapter 2: Local Area Network (LAN)
- Chapter 3: Ethernet & Connections
- Chapter 4: Wireshark & Protocols
- Chapter 5: Basic Troubleshooting

# Local Area Networks (LANs)

- LANs are small networks usually confined to a single location (building, floor, room…)
- Can be either wired or wireless
- Peer-to-peer (ad-hoc with no dedicated servers)
- Server based (dedicated servers provide services to clients)

# Simple Networks

- 10Base2 (thin ethernet) up to 200 meters on co-axial cable
- 10Base5 (thick ethernet) up to 500 meters on co-axial cable
- 10BaseT (star network) up to 100 meters on Unshielded Twisted Pair (UTP) RJ45 connector

# Unshielded Twisted Pair (UTP) Cabling

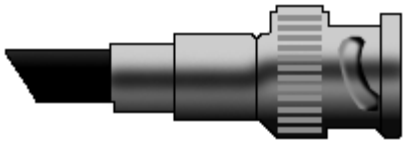
- Category 1: 1 Mbps, UTP for voice only
- Category 2: Data 4 Mbps
- Category 3: Data 10 Mbps
- Category 4: Data 20 Mbps
- Category 5: Data 100 Mbps
- Category 5e: Data 100 Mbps, some compatibility with 1,000 Mbps
- Category 6: Data 1,000 Mbps (250 MHz)
- Category 6e: Data 10,000 Mbps (500 MHz) Less than 30 Meters
- Category 7: Data 10GBaseT for 100 Meters (600 MHz, 1.2 GHz in pairs with Siemon connector)

# Unshielded Twisted Pair (UTP) Hub

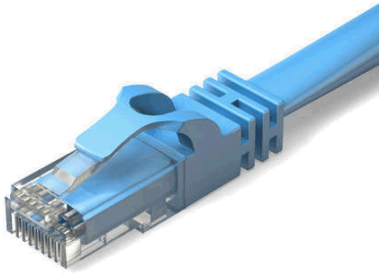
Carrier Sense Multiple Access with Collision Detection (CSMA/CD)

- Multiple Access - all hosts transmit/receive on same network
- If more than one host transmits at same time collision occurs
- Jam signal transmitted and host backs off for a random time
- Saturates at 40% utilization

# Copper Connectors



BNC – 10Base2 network, not used very often

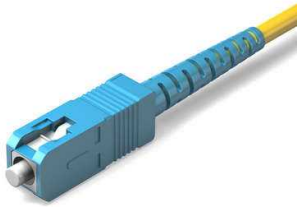


RJ45 – Up to 1000BaseT network, the most common cable used today

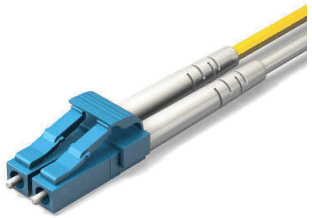
10GBaseT - 10Gb copper cable, IEEE Draft P802.3ak/DS.3 specification.

# Fiber Connectors

ST - Straight Tip, 2.5mm ferrule with a round plastic or metal body, "twist-on/twist-off" bayonet-style mount



SC - Subscriber Connector 2.5mm ferrule with a push-on/pull-off mount. Usually seen in pairs with a clip holding them together



LC - Local Connector 1.25mm ferrule with a push-on/pull-off mount

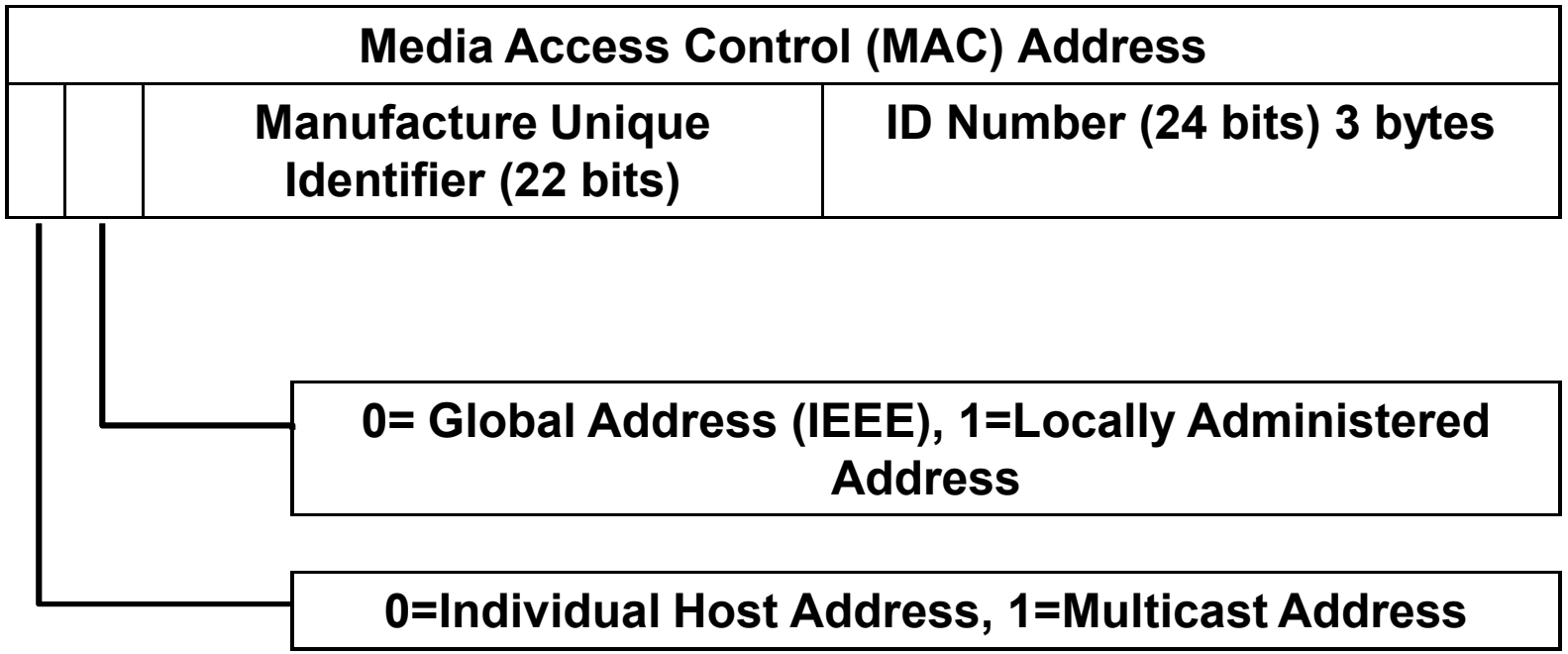
# Media Access Control (MAC)

- The MAC address is a 6-byte number most commonly represented as six two-digit hexadecimal numbers
- First half defines manufacturer
- Second half is the unique device number
- MAC addresses are also know as “Hardware Addresses”

# Media Access Control (MAC) Address

- Every network device has a pre-programmed MAC address unique to the manufacturer
- Length 3 bytes (24 bits) allows the manufacturer to create 16 million NICs
- First and second bits are special flags
  - First bit – Address Type, 0=individual, 1=multicast
  - Second bit – Address assignment, 0=global (IEEE), 1=locally assigned

# Media Access Control (MAC) Address



# Binary

- Binary digits use a value of 0 or 1 and are referred to as a bit
- To make it easier for humans to understand TCP/IP values are broken down into groups called octets (8 digits)
- TCP/IP V4 addresses are 4 octets in length

# Binary Translation

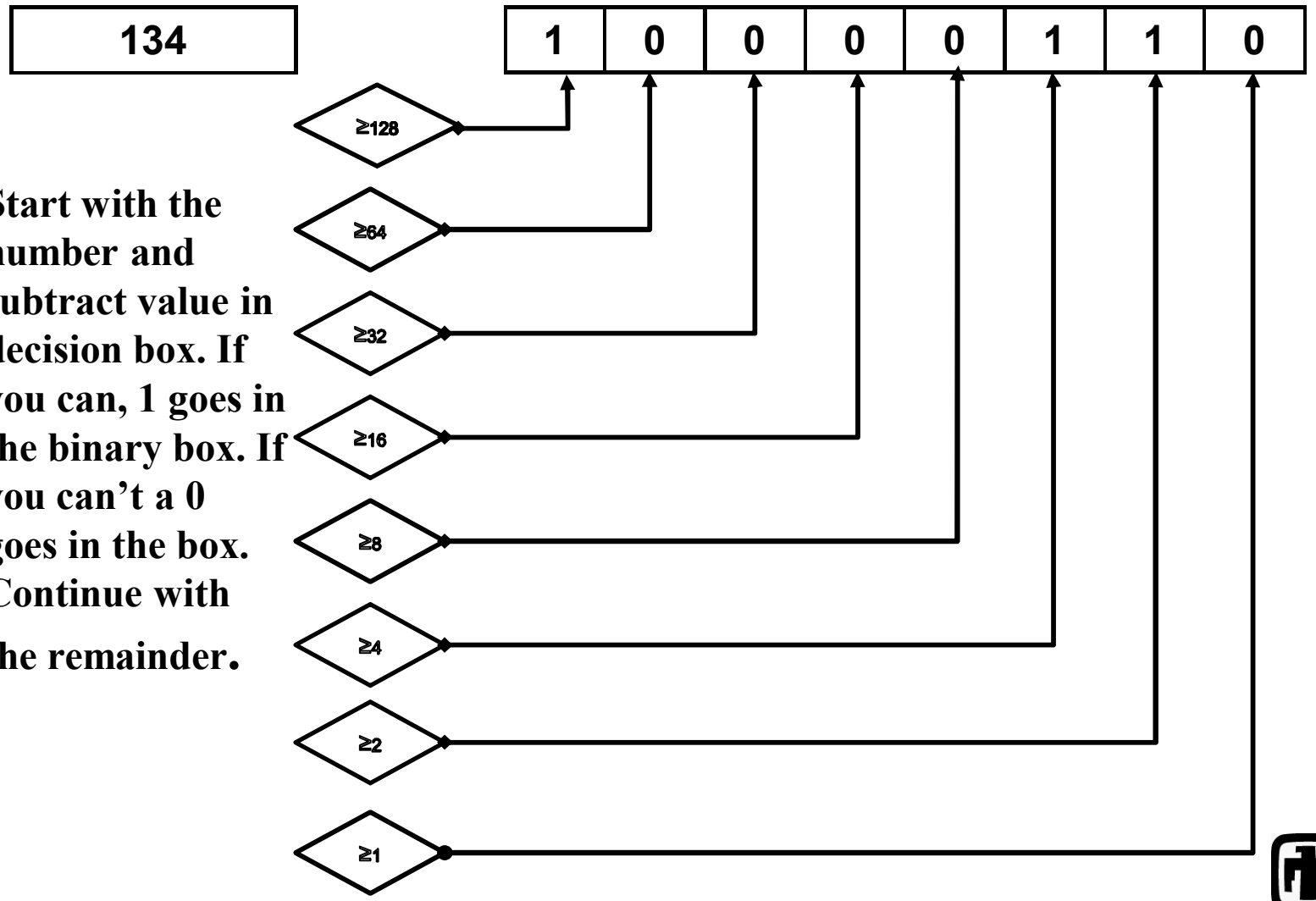
- People use decimal in everyday life
- In networking you need to know how to convert between binary and decimal
- Binary is a numerical notation to the base 2, in which each place of a number is expressed as 0 or 1, corresponds to a power of 2
- The decimal number 63 appears as 111111 in binary notation, $63 = 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$

# Translating Binary

- Using 8 bits (1 octet) you can represent values between 0 and 255 (decimal)
- Progression of values from highest to lowest, 128, 64, 32, 16, 8, 4, 2,1

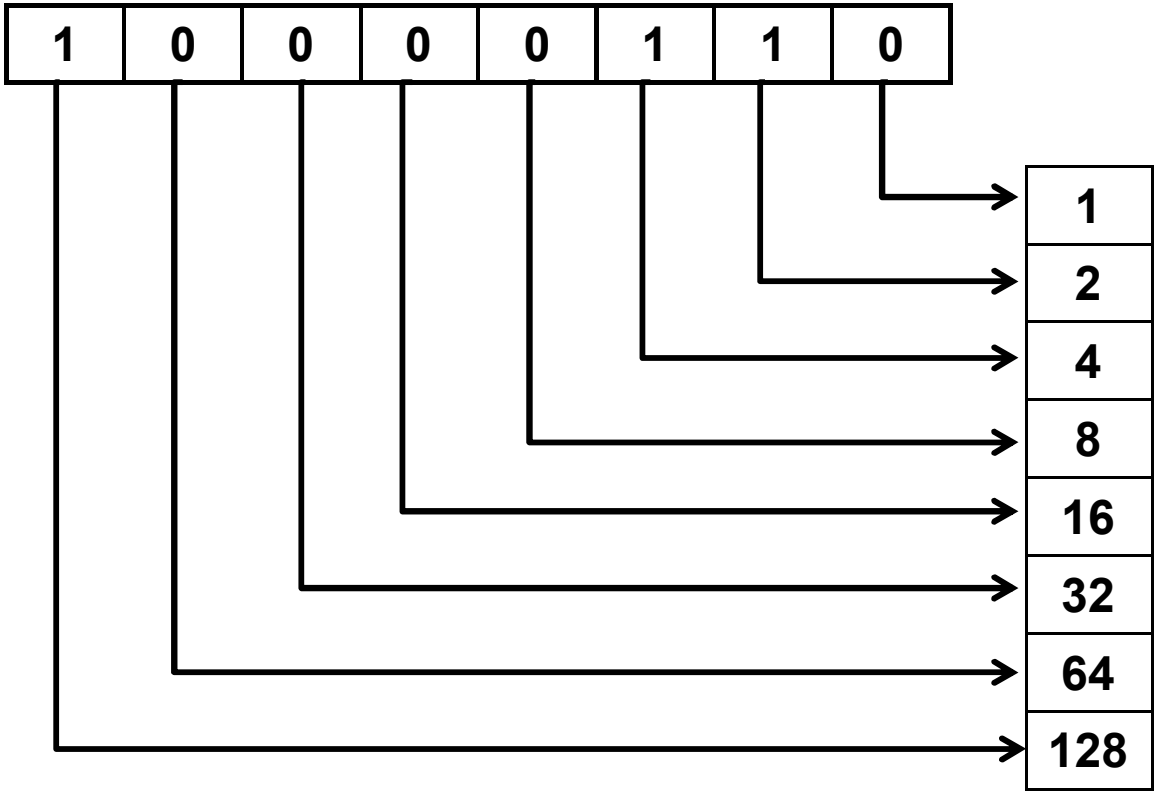
| | Dot-decimal notation | Binary form |
|---|---|---|
| IP address | 192.168.5.130 | 11000000.10101000.00000101.10000010 |
| Subnet Mask | 255.255.255.0 | 11111111.11111111.11111111.00000000 |
| Network Portion | 192.168.5.0 | 11000000.10101000.00000101.00000000 |
| Host Portion | 0.0.0.130 | 00000000.00000000.00000000.10000010 |

# Decimal to Binary




Sandia National Laboratories

# Binary to Decimal

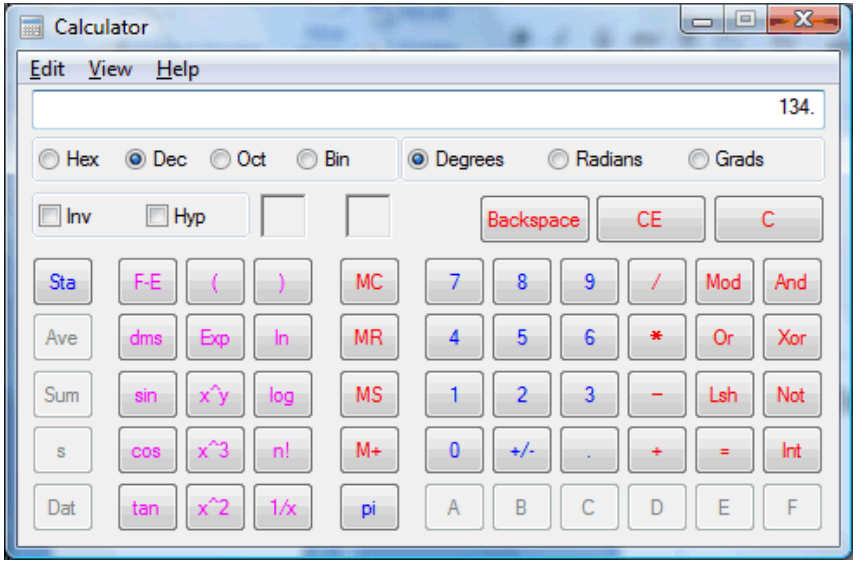



$$128+0+0+0+0+4+2+0 = 134$$

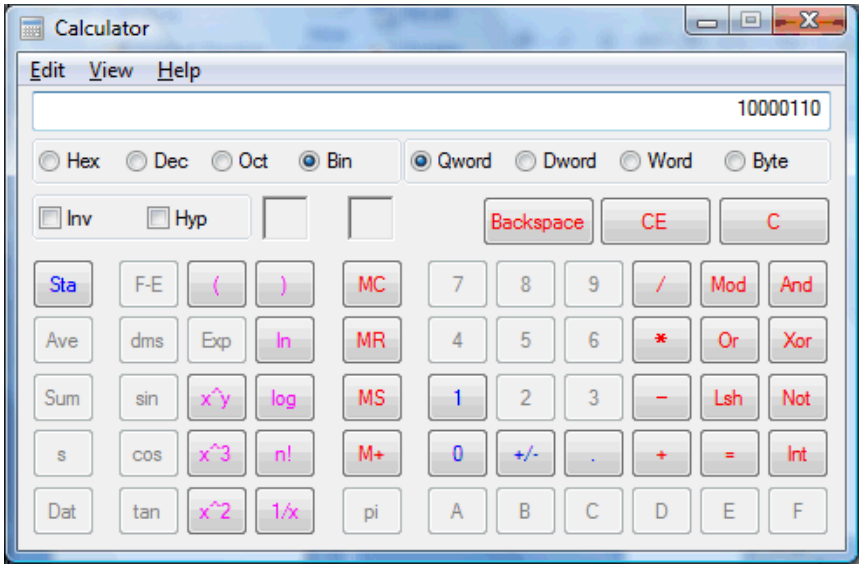
Sandia National Laboratories

# Do It The Easy Way

## Enter Decimal Number




## Click 'Bin'




Sandia National Laboratories

# Subnet Mask

- IP addresses have two parts
- Network ID and Host ID
- Subnet mask separates the network and host
- Subnet mask by class
  - Class A 255.0.0.0
  - Class B 255.255.0.0
  - Class C 255.255.255.0
  - Class D and E don’t have a defined mask

# Common IP Address Classes

- There are 5 classes, class A, B, C, D and E
- Class A, 0-126 126 networks
- Class B, 128-191 16,384 Networks
- Class C, 192-223 2,097,152 Networks
- Class D, 224-239 (Multicast)
- Class E, 240-254 (Experimental)
- The 127 network is reserved for loopback and can't be used

# Variable Length Subnet Mask (VLSM)

- Used when you want to divide the network not on a common class boundary
- Every network has two reserved addresses
- All 0's – Network ID
- All 1's - Broadcast

# Number of Host on a Network

- From a single Class A we can use the subnet mask to divide it up
  - 255.0.0.0 $2^{24}-2$ (About 16 million Hosts, 1 Network)
  - 255.255.0.0 $2^{16}-2$ (65,534 Hosts, 256 Networks)
  - 255.255.255.128 $2^{15}-2$ (32,766 Hosts, 512 Networks)
  - 255.255.255.0 $2^{8}-2$ (254 Hosts, 65,536 Networks)
  - 255.255.255.240 $2^{4}-2$ (14 Hosts, 1 Million Networks)
  - 255.255.255.252 $2^{2}-2$ (2 Hosts, 4 Million Networks)

# How Do We Get Around Address Loss?

- In the original TCP/IP specifications when subnetting you lose the 1st and last networks (not used today)
- Example: 40.45.0.0 network using 255.255.192.0 as a subnet mask
  - 40.45.0.0 Can't use, overlaps original network
  - 40.45.64.0
  - 40.45.128.0
  - 40.45.192.0 Can't use, overlaps broadcast address

# Classless Inter-Domain Routing (CIDR)

- Introduced in 1993 as the official way to subnet
- CIDR abbreviations for subnet mask
  - 255.0.0.0 8 bits for Network ID /8
  - 255.255.0.0 16 bits for Network ID /16
  - 255.255.255.0 24 bits for Network ID /24
  - Essentially how many ‘1s’ do I have in the subnet mask

| Network | Range |
|---|---|
| 192.168.2.0/25 | 192.168.2.1 - 192.168.2.126 |
| 192.168.2.128/27 | 192.168.2.129 - 192.168.2.158 |
| 192.168.2.160/27 | 192.168.2.160 - 192.168.2.190 |

# Private IP Address Ranges

- IP ranges set aside for everyone's use
  - 10.0.0.0 Mask: 255.0.0.0
  - 172.16.0.0 to 172.31.0.0 Mask: 255.255.0.0
  - 192.168.0.0 to 192.168.255.0 Mask: 255.255.255.0
  - By convention these ranges are not routed on the open internet
  - Border routers should be configured to not pass these ranges
  - 169.254.0.0/16 is defined as Link-Local (Zeroconf/auto-config) and shouldn't be passed by any router

| IP address range | number of addresses | *classful* description | largest CIDR block (subnet mask) | host id size |
|---|---|---|---|---|
| 10.0.0.0 – 10.255.255.255 | 16,777,216 | single class A | 10.0.0.0/8 (255.0.0.0) | 24 bits |
| 172.16.0.0 – 172.31.255.255 | 1,048,576 | 16 contiguous class B's | 172.16.0.0/12 (255.240.0.0) | 20 bits |
| 192.168.0.0 – 192.168.255.255 | 65,536 | 256 contiguous class C's | 192.168.0.0/16 (255.255.0.0) | 16 bits |

Sandia National Laboratories

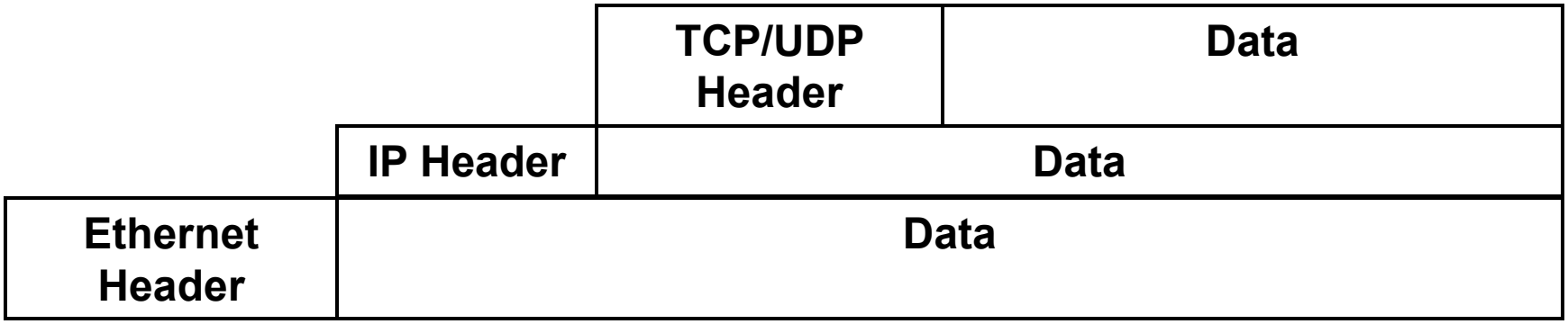
# Lets Have Some Fun

- If the Network ID of two hosts match they are on the same network
- Are these two addresses on the same network?
  - 134.232.5.6 & 134.232.8.6 Mask: 255.255.0.0
  - 192.168.250.3 & 192.168.250.250 Mask:255.255.255.0
  - 67.103.4.240 & 67.103.5.241 Mask: 255.255.255.0
  - 67.103.4.240 Mask: 255.255.0.0 & 67.103.8.240 Mask: 255.255.255.0
  - 80.40.10.4 & 80.40.10.250 Mask: Unknown/Not Available

# Ethernet & Connections

- Chapter 1: History & Models
- Chapter 2: Local Area Network (LAN)
- Chapter 3: Ethernet & Connections
- Chapter 4: Wireshark & Protocols
- Chapter 5: Basic Troubleshooting

# What does it look like on the Network?




- As the packet goes up/down the stack header information is added/removed
- Data in the lower levers contains information (headers) for upper layers

# Ethernet Frame 802.3

- Frame built/used by layer 2
- Preamble to sync clock (length 1-7)
- Destination and source MAC (6 + 6 bytes)
- Size of data (2 bytes)
- Data (64 – 1500 bytes)
- Cyclic redundancy check (4 bytes)

# Ethernet Frame Types

## RAW

| Preamble/ Start | Destination MAC | Source MAC | Length | Data | CRC |
|---|---|---|---|---|---|
| 7/1 bytes | 6 bytes | 6 bytes | 2 bytes | 46 bytes min – 1500 bytes max | 4 bytes |

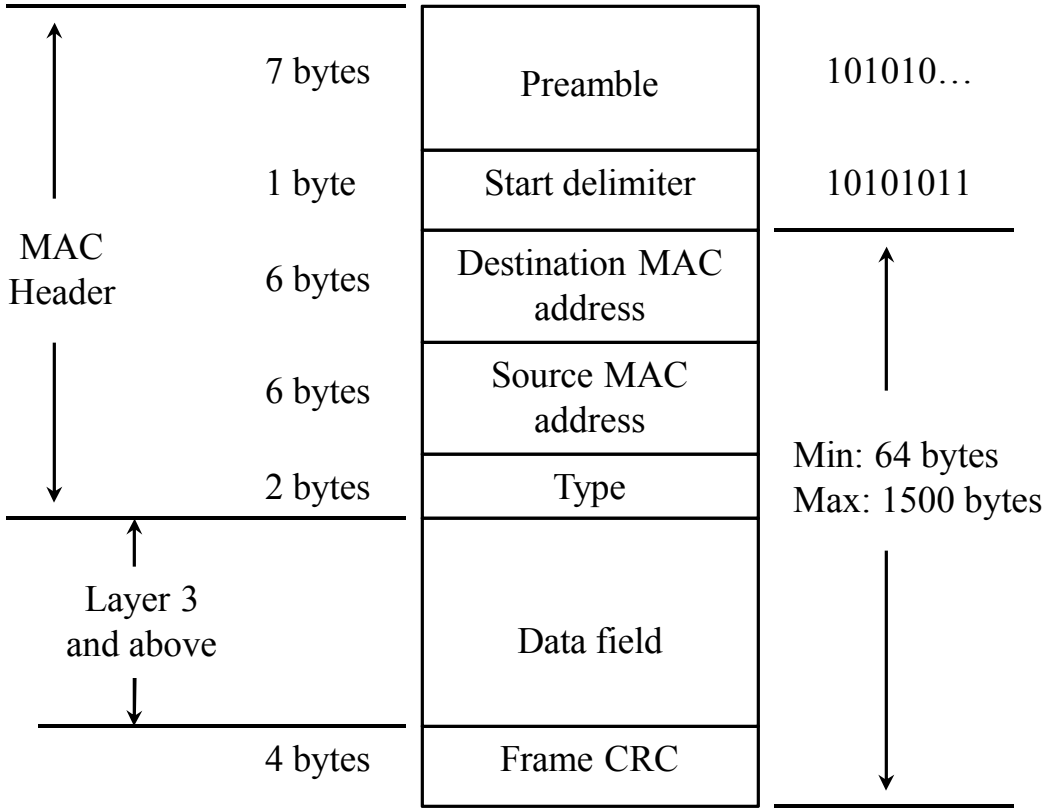
## Type II

| Preamble/ Start | Destination MAC | Source MAC | Type | Data | CRC |
|---|---|---|---|---|---|
| 7/1 bytes | 6 bytes | 6 bytes | 2 bytes (0x80 for IP V4) | 46 bytes min – 1500 bytes max | 4 bytes |

# Ethernet Frame Type II

- Frame is built/used by the link layer
- The preamble is always 7 bytes of "10101010" followed by one byte of "10101011"
- Destination and source are identified by their MAC address
- Data field contains information for upper layers
- Cyclic Redundancy Check (CRC) used for integrity check

# Ethernet Frame Type II




Sandia National Laboratories

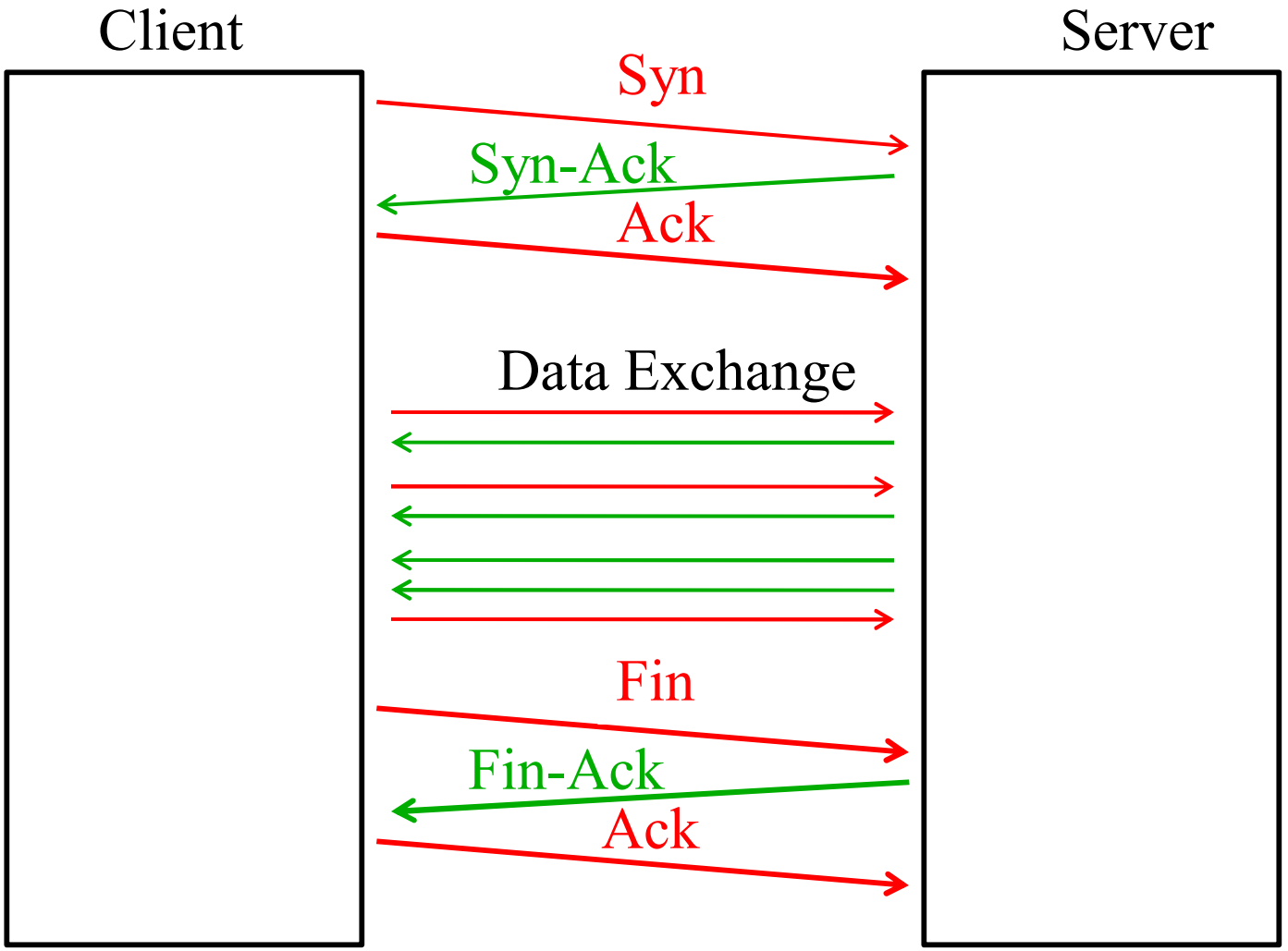
# Transmission Control Protocol (TCP)

- It's the TCP in TCP/IP ☺
- TCP setup starts with a 3-way handshake in which both parties acknowledge a connection
  1. Sending host to receiving host — Synchronize request (syn)
  2. Receiving host to sending host — Synchronize request (syn) + Synchronize Acknowledge (ack)
  3. Sending host to receiving host — Synchronize Acknowledge (ack)
- As with the initial handshake, all TCP segments from the sender require acknowledgement from the receiver

# Closing the TCP session

- To close the connection
  - A FIN packet is sent to the destination
  - The destination replies with a FIN + ACK packet
  - Sender replies with an ACK packet closing the connection
  - If this isn't done, the operating system must depend on its time out value to close the connection

# Normal TCP Connection



# TCP Structure

**TCP headers specify the connection options and dictates the flow of data**

| Source Port | Destination Port | Sequence Number | Acknowledgement Number | Data Offset | Reserved |
|---|---|---|---|---|---|

| URG | ACK | PSH | RST | SYN | FIN | Window | Checksum | Urgent Pointer | Options |
|---|---|---|---|---|---|---|---|---|---|

| Padding | Data |
|---|---|

Sandia National Laboratories

# TCP Structure

- Source and Destination Port – 16 bit values
- Sequence Number – Incrementing 32-bit value that indicates the order in which this segment should be reassembled
- Acknowledgement Number – 32 bit value, if the ACK flag is set, the value of this field is the next sequence number that the receiver is expecting
- Data Offset – 4 bit field with size of TCP header
- Reserved – 6 bit field set to all 0s

# TCP Structure

- Urgent Control Bit (URG) – If set to 1, urgent field contains data
- Acknowledgement Control Bit (ACK) - If set to 1, acknowledgement number field contains data
- Push Control Bit – If set to 1, send data immediately instead of waiting for buffer to fill up

# TCP Structure

- Reset Control Bit – If set to 1, TCP packet requests a reset of the connection
- Synchronize Control Bit (SYN) – Request to initiate connection and exchange initial sequence numbers (ISN)
- Finish Control Bit (FIN) – If set to 1, sending host has no more data to send
- Window – 16 bit field used to say how much data the receiving host can accept at a time

# TCP Structure

- Checksum – Used to check the integrity of the header and payload
- Options – Variable length field containing end of option list, No Operation or Maximum Segment Size
- Padding – Data is padded to land on a 32 bit boundary
- Data – Data being transmitted

# User Datagram Protocol (UDP)

- Transport layer protocol used for connectionless data transmission
- Best effort transmission between hosts
- Relies on higher level layers to provide error detection and correction if needed
- Normally used for Domain Name System (DNS) request, Simple Network Management Protocol (SNMP) and streaming data. Internet radio and video are two examples
- UDP is faster that TCP
  - No connection overhead (syn/ack/fin)
  - Packet is addressed and sent

# UDP Structure

| Source Port | Destination Port | Length | Checksum | Data |
|---|---|---|---|---|

- Source and Destination Port – 16 bits
- Length – 16 bits
- Checksum – 16 bit value for integrity check
- Data – Variable length
- Theoretical limit of 65,535 bytes (8 byte header + 65527 bytes of data)
- Practical limit for the data length which is 65,507 bytes
- (MTU 1500 minus above)

# Internet Protocol Basics

- IP addresses are used to identify each host on a TCP/IP network
- Each host must have a unique ID
- Host can be computers, printers, routers, switches, sniffers, or any type of network device

# Internet Protocol (IP)

| Version | Length | Service Type | Packet Length | Identification | Flags |
|---|---|---|---|---|---|
| Fragment Offset | Time-To-Live TTL | Protocol | Header Checksum | Source Address | Destination Address |
| Options | Padding | Data | | | |

# Internet Protocol (IP)

- Version: V4 or V6, V4 is dominant at the moment
- Length: Length of IP Header
- Service Type: Informs IP how to handle the frame; Delay, Reliability, Throughput…
- Frame/Packet length: Total length including headers
- Identification, Flags and Frame Offset: Used for fragment fields and large packet reassembly

# Internet Protocol (IP)

- Time-To-Live: TTL decrements once for every router the packet goes through. When TTL reaches zero the router drops the packet and sends back an ICMP error message to the sender
- Protocol: Identifies higher level protocol used in packet
- Header Checksum: Integrity check of the header
- Source & Destination Addresses: IP address of source and destination

# Internet Protocol (IP)

- Options: Optional 8 bit code that can appear in the packet
  - Copy bit, if set to 1, copy options into fragmented packets
  - Option Class, datagram, network control, debugging and measurement
  - Option number code, additional information if needed

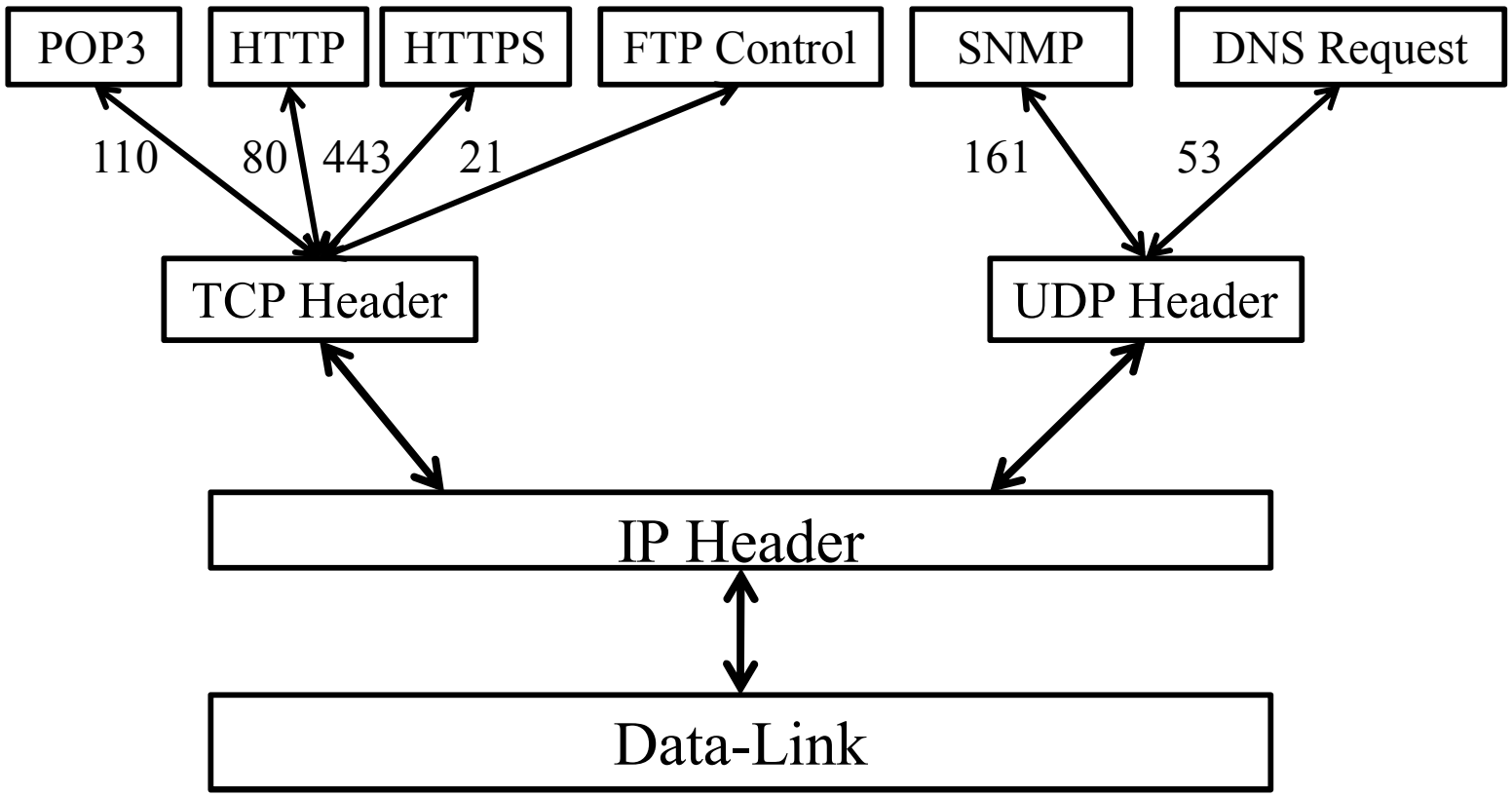
# Internet Protocol (IP)

- Padding: Used to ensure even number of bytes in the header
- Data: payload of packet

# Software Ports (Transport Layer)

- Used as numerical identifier for the endpoints for host-to-host communications
- Ephemeral  ports (0-1023), sometimes also known as trusted ports, can only be used by system (or root) processes or by programs executed by privileged users on most systems
- Registered ports (1024-49151) are listed by IANA at the request of a software vendor. Examples: pcAnywhere, MySQL Proxy, Embedded Light Control Network…
- Dynamic and/or private ports (49152-65535). Dynamically assigned ports are opened and closed by the server as needed.
- Port information http://www.iana.org/assignments/port-numbers or http://insecure.org/nmap/data/nmap-services
- On Windows look at the 'services' file in C:\Windows\System32\drivers\etc

# Software Ports



# Software Port Got-Yas

- Be careful and don't confuse the client and server in the conversation
- The only way to be sure is if you see the initial handshake
- It's easy to confuse the two if both ports are in the Registered Ports (1024-49151) area
- Older operating systems like Windows XP and some *NIX systems
  - The first connection from the client usually starts at 1024 and increments every time it makes a new connection
  - When it gets to 4999 it usually starts at 1024 again
- Newer operating system like Windows Vista and newer *NIX systems use a larger range (49152-65535)

# Common Software Ports

| | | | |
|---|---|---|---|
| • ftp-data | 20/tcp | • rpc | 111/tcp/udp |
| • ftp-cont | 21/tcp | • nntp | 119/tcp |
| • telnet | 23/tcp | • ntp | 123/udp |
| • smtp | 25/tcp | • netbios | 137-139/tcp/udp |
| • time | 37/tcp | • imap2 | 143/tcp |
| • DNS | 53/udp/tcp | • ldap | 389/tcp |
| • tftp | 69/tcp | • https | 443/tcp |
| • http | 80/tcp | • syslog | 514/udp |
| • pop3 | 110/tcp | • mDNS | 5353/udp |

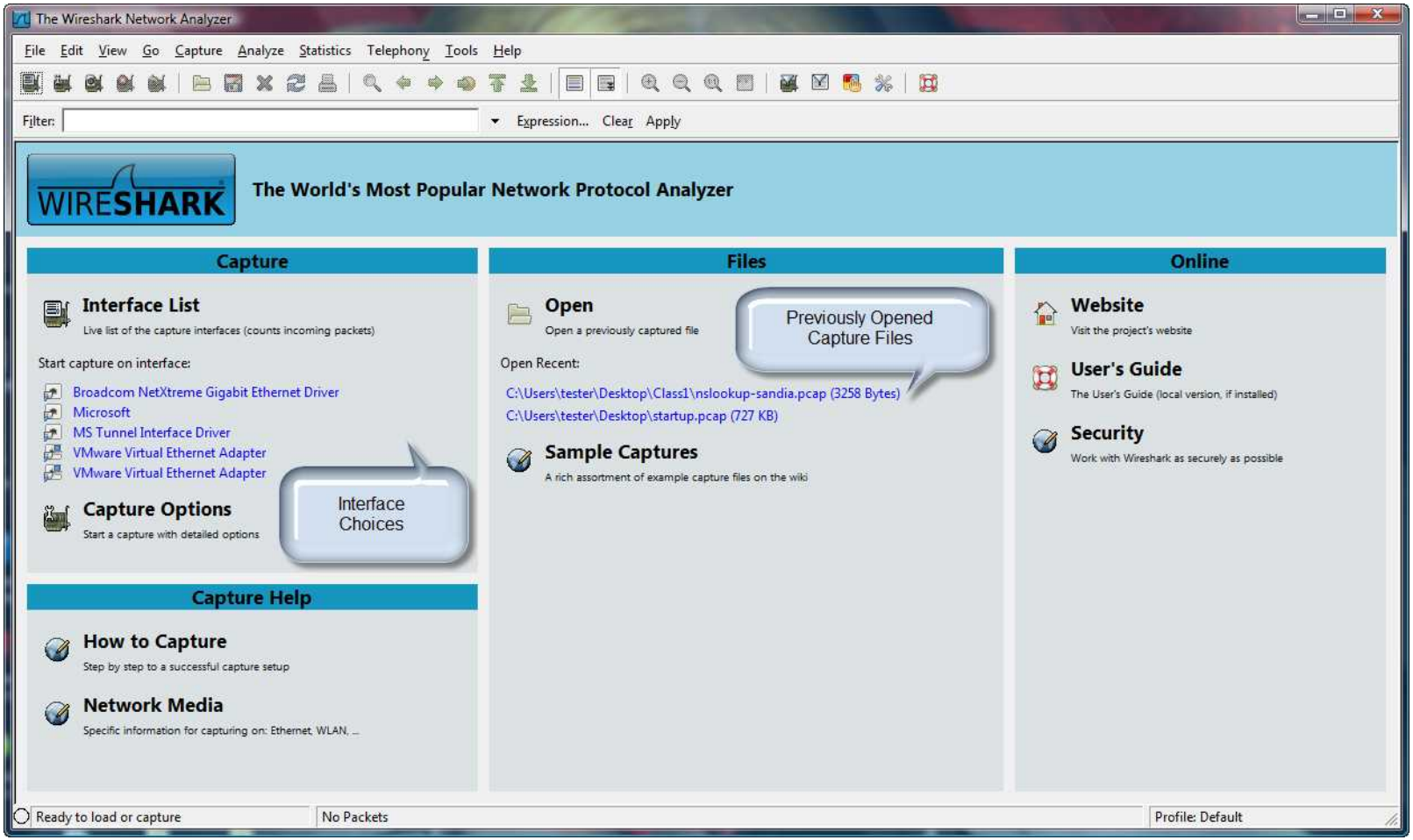
# Wireshark & Protocols

- Chapter 1: History & Models
- Chapter 2: Local Area Network (LAN)
- Chapter 3: Ethernet & Connections
- Chapter 4: Wireshark & Protocols
- Chapter 5: Basic Troubleshooting

# Wireshark

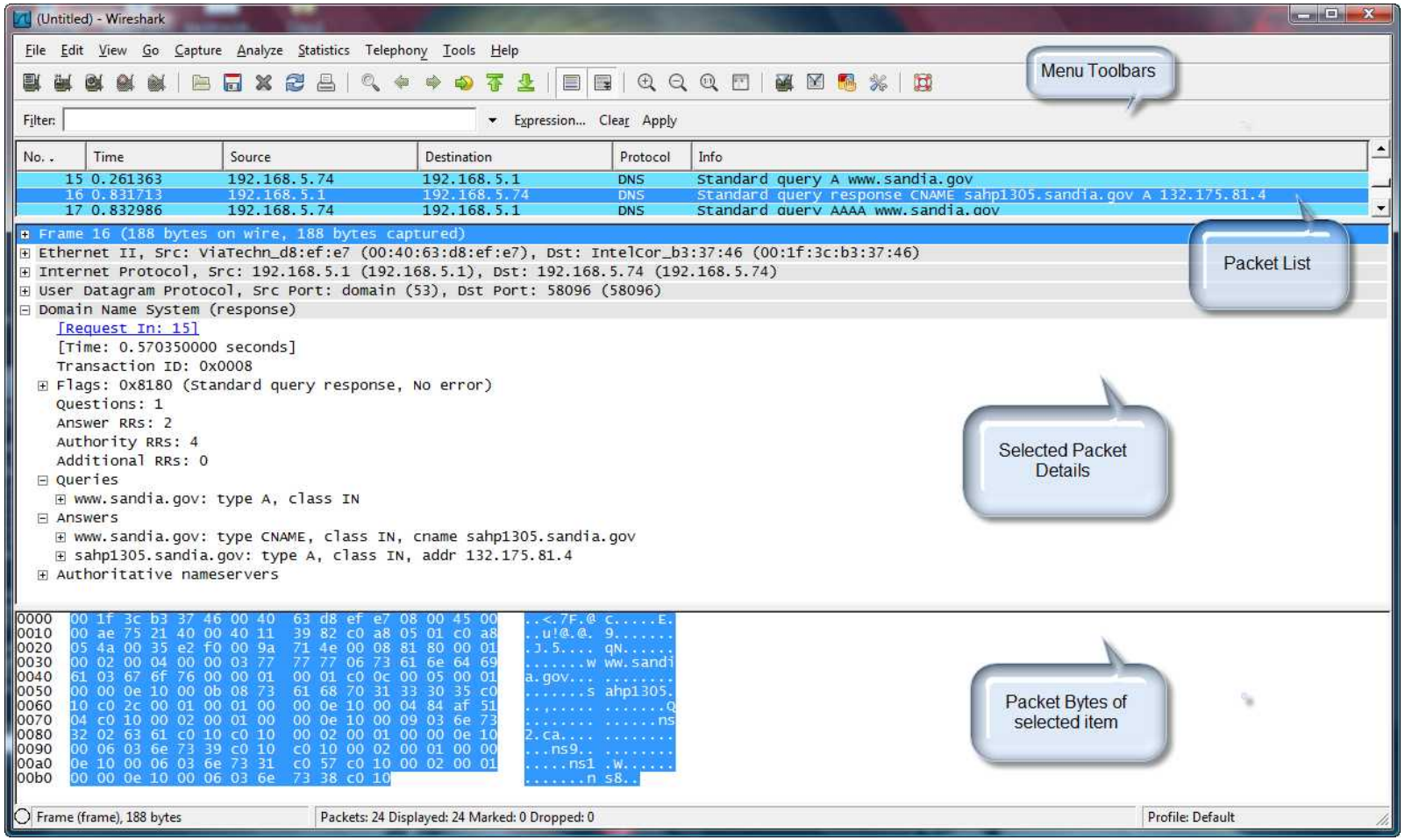
- Open source network protocol analyzer
- Available at http://www.wireshark.org/
- Training and user guide available online at http://www.wireshark.org/docs/
- I've included the user guide on the class DVD

# Wireshark Opening Screen




Sandia National Laboratories

# Wireshark




Sandia National Laboratories

# Common Protocols seen on the Network

- Address Resolution Protocol (ARP)
- Dynamic Host Configuration Protocol (DHCP)
- Domain Name System (DNS)
- File Transfer Protocol (FTP)
- HyperText Transfer Protocol (HTTP)
- Internet Control Message Protocol ICMP echo request (PING)

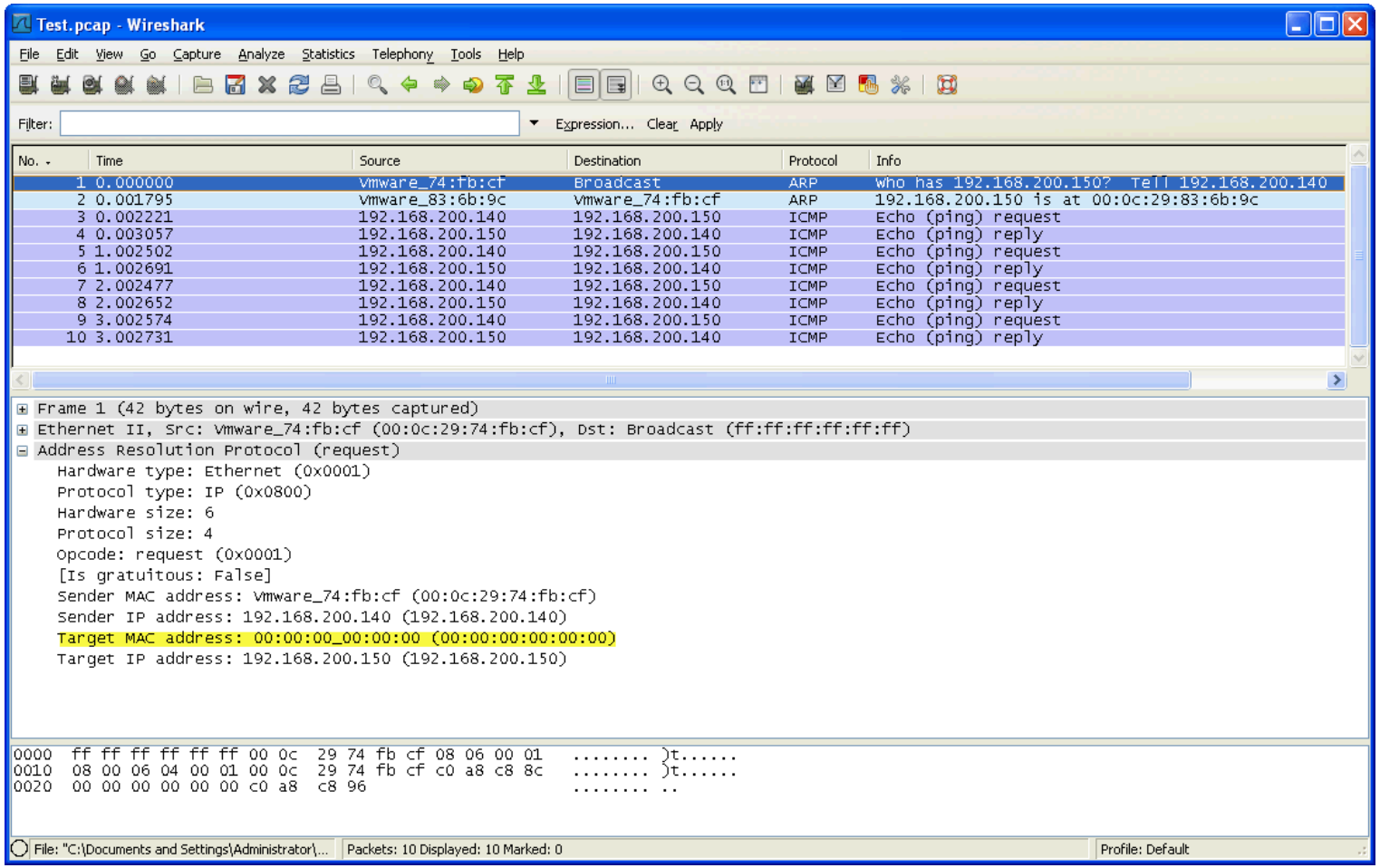
# Address Resolution Protocol (ARP)

- Ask the question, “What is the MAC address of the node that has this IP address?”
- Simple request/reply protocol that is used to find a destination’s hardware address when it’s unknown
- Used when the destination is on the same network segment
- There are four addresses involved in this protocol
  - Sender Hardware Address
  - Sender IP Address
  - Destination Hardware Address
  - Destination IP Address
- The ARP answer contains both the sender and the original ARP requester records

# Address Resolution Protocol (ARP)

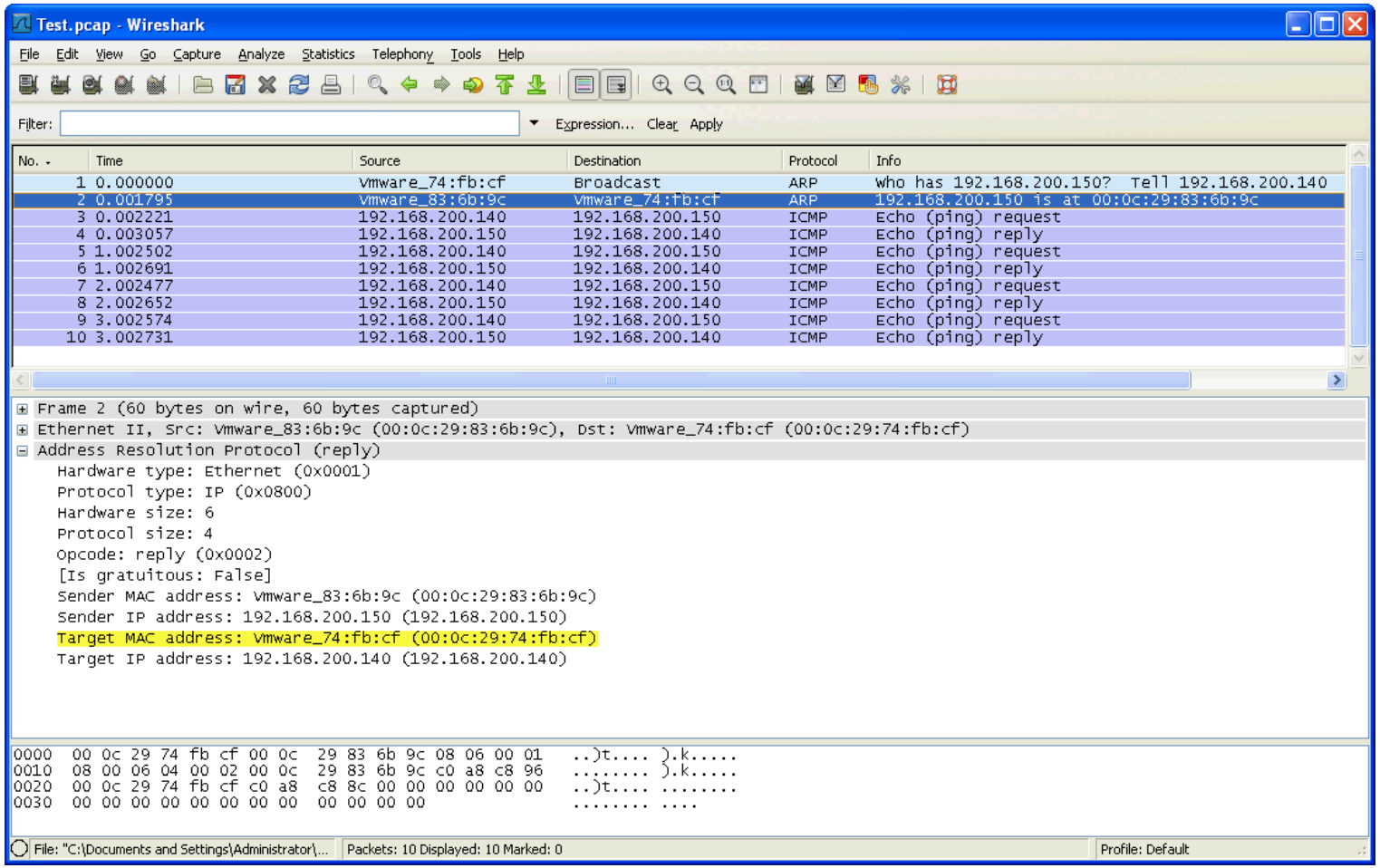
- Sequence of events between nodes
  1. Source checks it's cache
  2. Source generates ARP request
  3. Source broadcasts ARP request
  4. Local devices process the ARP broadcast
  5. Destination generates ARP reply
  6. Destination updates it's ARP cache with source information
  7. Destination sends ARP reply to source
  8. Source processes the ARP reply
  9. Source updates it's ARP cache

# Address Resolution Protocol (ARP) Request




Sandia National Laboratories

# Address Resolution Protocol (ARP) Reply




Sandia National Laboratories

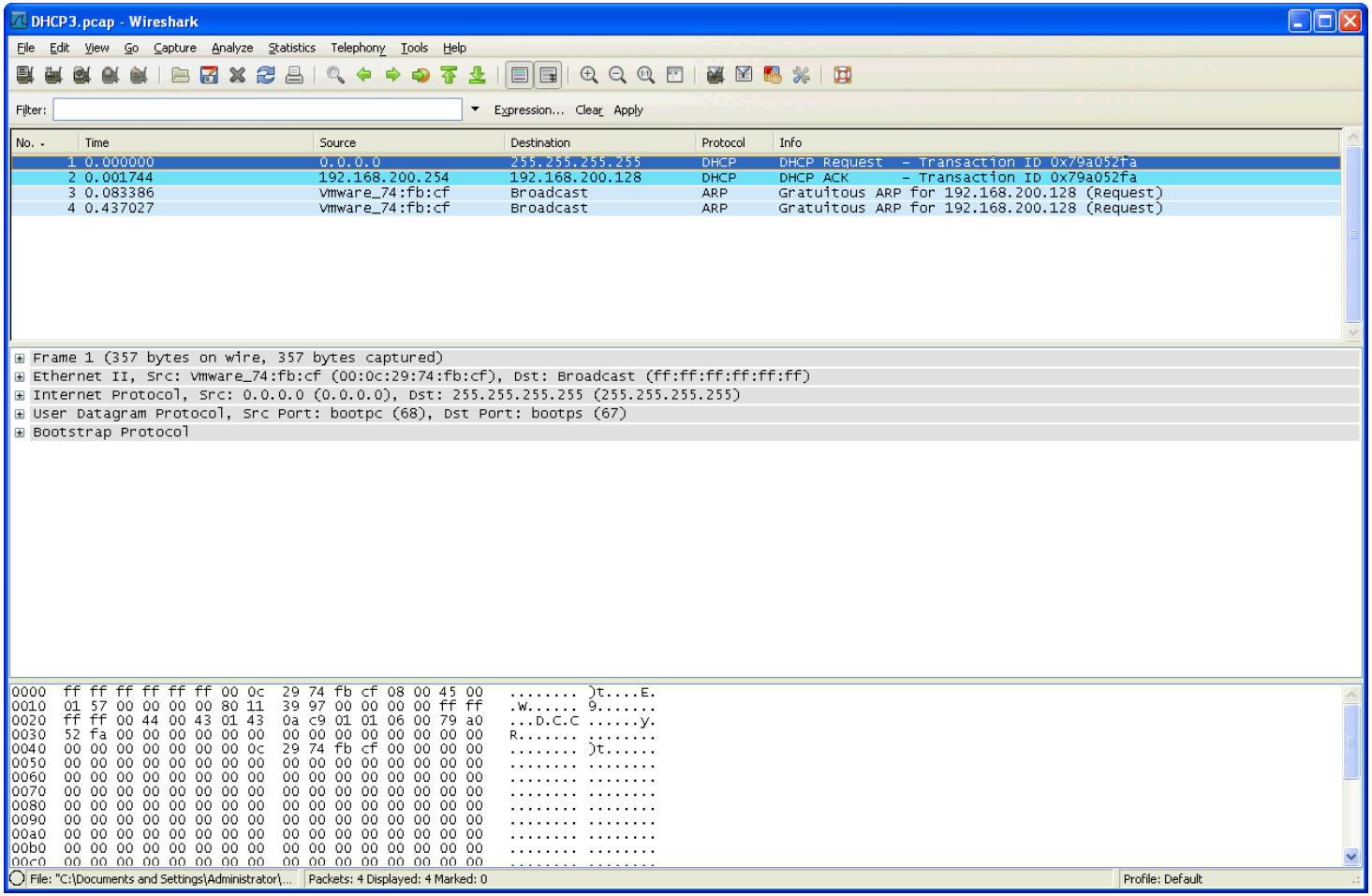
# Dynamic Host Configuration Protocol (DHCP)

- Allows a host to connect to a network without any manual configuration
- Used between client/server for client IP address configuration
- RFC 2131 is the current definition for IPv4
- Information usually passed to client in “Options” fields
  - IP Address for client to use
  - Subnet Mask
  - DNS Servers
  - Router/Gateway Address
  - IP Lease time

# Dynamic Host Configuration Protocol (DHCP)

- Sequence between client/server
  - Client sends DHCP discovery
  - Server offers IP address to client
  - Client sends a request back to the server letting it know it is going to use the IP offered
  - Server sends acknowledgement

# Dynamic Host Configuration Protocol (DHCP)




Sandia National Laboratories

# Domain Name System (DNS)

- DNS translates a name, such as "google.com" into an IP address that a computer can then connect to
- RFC 1034 and 1035
- Domain name parts are separated by a “.”and use the form hostname.subdomain.tld (top-level-domain)
- Example, in fred.mydomain.org
  - org is the top level domain
  - mydomain is a subdomain of org (subdomains can as deep as 127 levels)
  - fred is the host name
- Top level domains are ‘Root Domains’ com, edu, gov, mil, org…
- The root servers only know about the next portion of the name

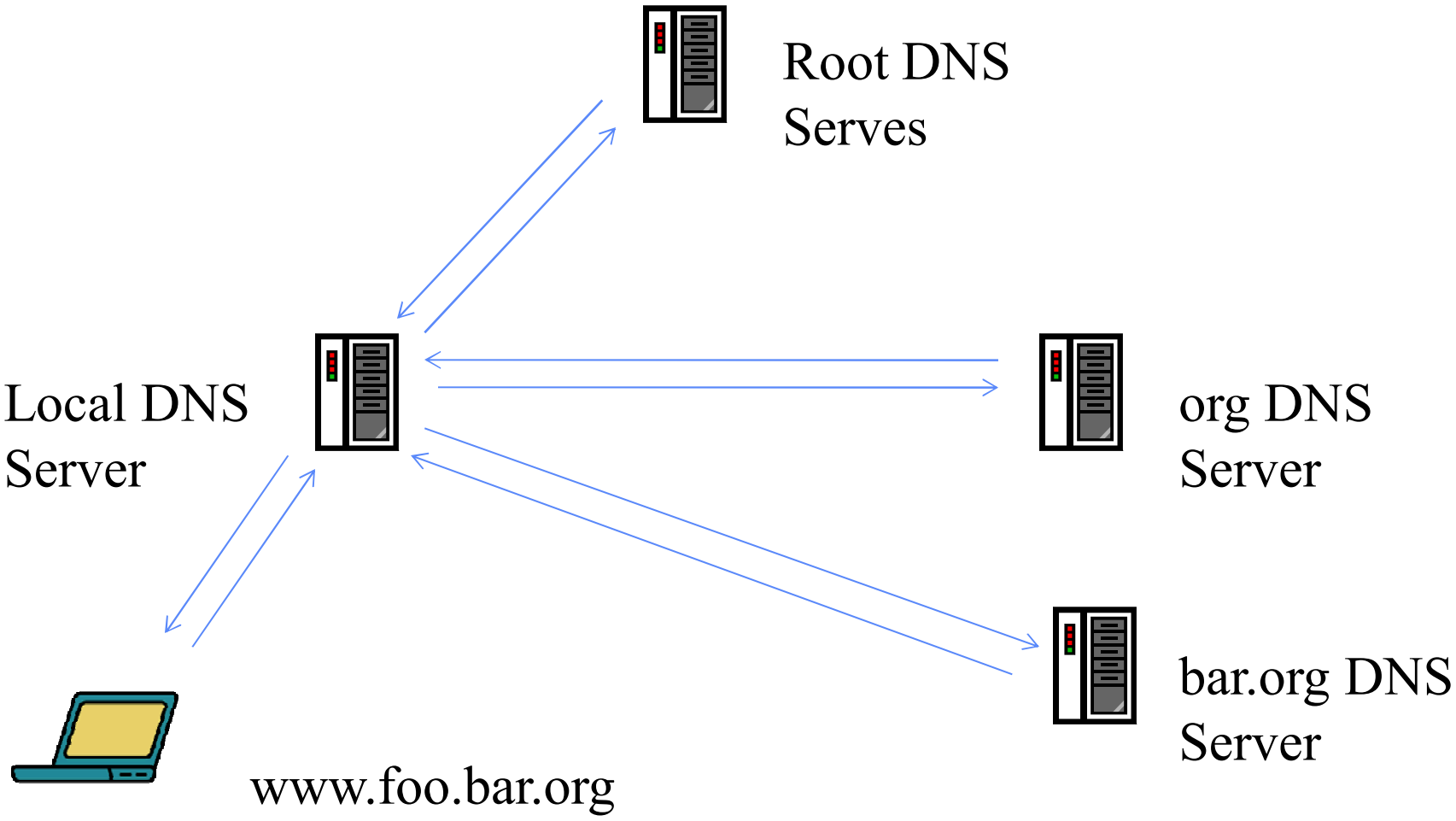
# DNS Server

- Contains a database with hostnames and IP addresses
- Can lookup either hostname or IP
- If request isn't in its local database, It can do recursive queries to other DNS servers

# DNS Resolver

- Software that resides on the client
- Responsible for initiating queries to DNS server of record
- Translates domain name into an IP address

# How DNS Queries Work



# Hosts File

- Hosts files are a very good way to map IP to Host names without using DNS
- Location on Windows systems, C:\Windows\System32\drivers\etc
- Edit using Notepad

```
# For example:
#
#      102.54.94.97     rhino.acme.com          # source server
#       38.25.63.10     x.acme.com              # x client host

127.0.0.1       localhost
::1             localhost
123.123.123.123         myweb.server.org
```

# The Good and Bad

- It can be used to solve name resolution problems
- Programs can use it to force request from a certain site
- Hackers can use it to launch man-in-the-middle attacks 
- Malware can use it to redirect you to sites they own for tracking, pop-ups, redirect you to a phony site or installing software 

# File Transfer Protocol (FTP)

- FTP is used to  exchange files over the network using TCP/IP
- Client initiates request to the server
- Initial connection is established on TCP port 21 (control port) of the FTP server
- Client uses any port greater than 1023, once connected the data is transferred over a different port

# File Transfer Protocol (FTP)

- There are two modes of operation, active and passive
- In both cases the client makes the first connection on TCP port 21 (control port)
- Active Mode – Client opens a second connection on port 20 (data port), data is sent/received on this port
- Passive Mode – Server selects random port greater than 1023 and the client connects to it for data transfer

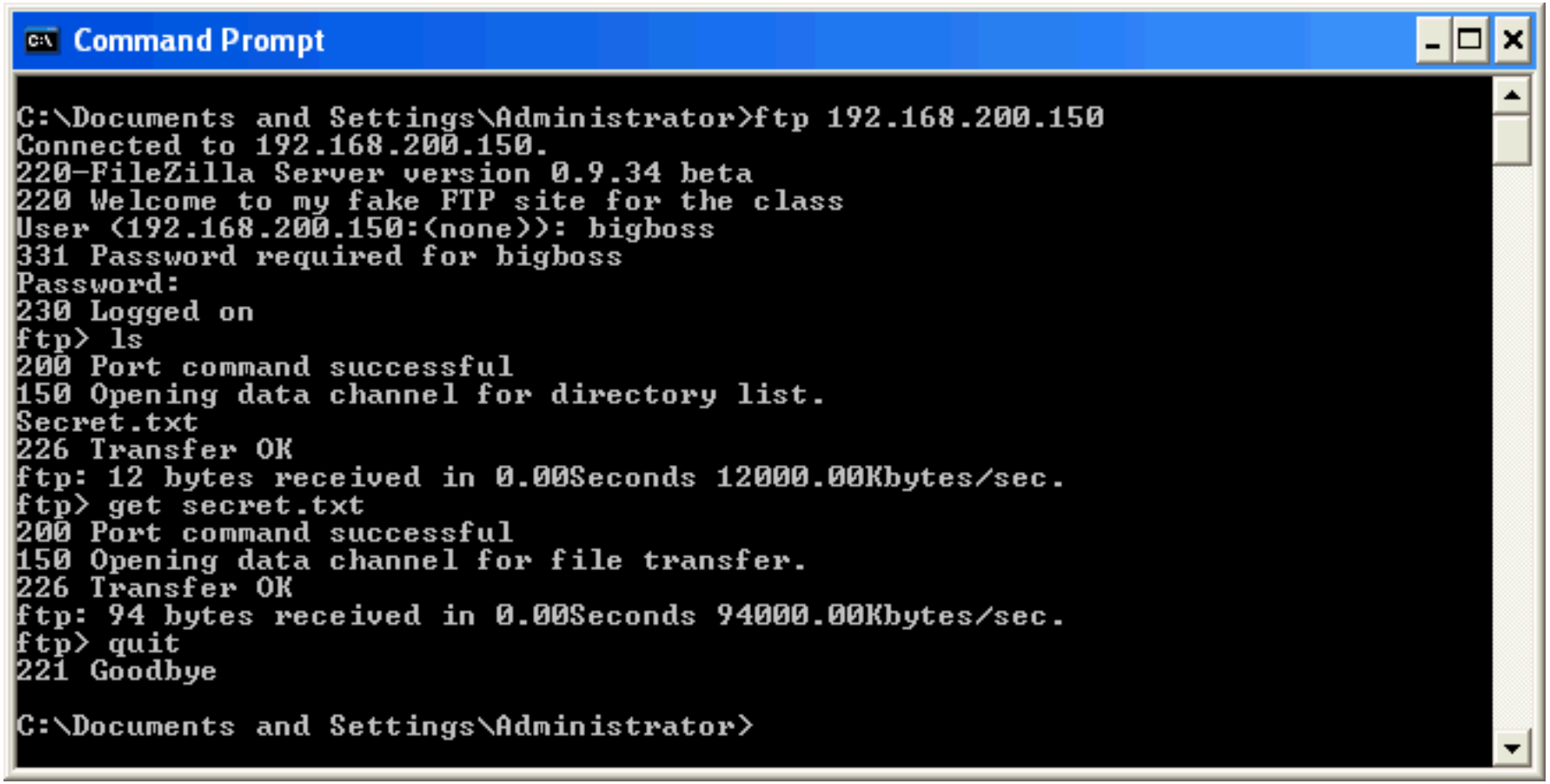
# File Transfer Protocol (FTP)

- Two modes of transmitting data, ASCII and Binary
- ASCII for transferring plain text data
- Binary for transferring binary data
- See http://www.nsftools.com/tips/RawFTP.htm for a full list of commands
- Common client commands
  - GET – Download a file from the server
  - PUT – Upload a file to the server
  - CD, LS, DIR – For moving around the directory structure

# FTP Commands

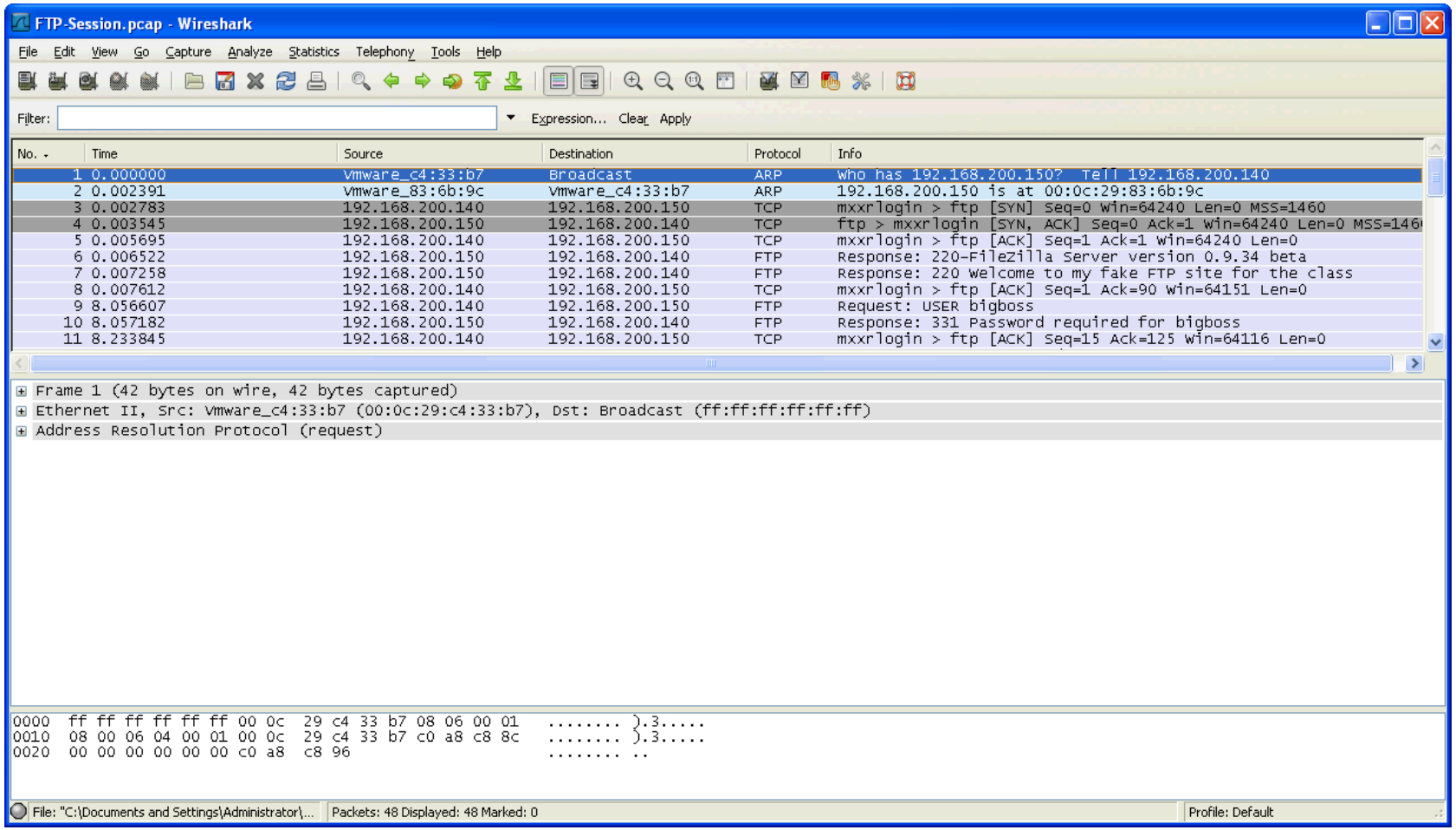
- ABOR - abort a file transfer
- CWD - change working directory
- DELE - delete a remote file
- LIST - list remote files
- MDTM - return the modification time of a file
- MKD - make a remote directory
- NLST - name list of remote directory
- PASS - send password
- PASV - enter passive mode
- PORT - open a data port
- PWD - print working directory
- QUIT - terminate the connection
- RETR - retrieve a remote file
- RMD - remove a remote directory
- RNFR - rename from
- RNTO - rename to
- SITE - site-specific commands
- SIZE - return the size of a file
- STOR - store a file on the remote host
- TYPE - set transfer type
- USER - send username
- ACCT* - send account information
- APPE - append to a remote file
- CDUP - CWD to the parent of the current directory
- HELP - return help on using the server
- MODE - set transfer mode
- NOOP - do nothing
- REIN* - reinitialize the connection
- STAT - return server status
- STOU - store a file uniquely
- STRU - set file transfer structure
- SYST - return system type

# FTP Client Side
# Text Transfer



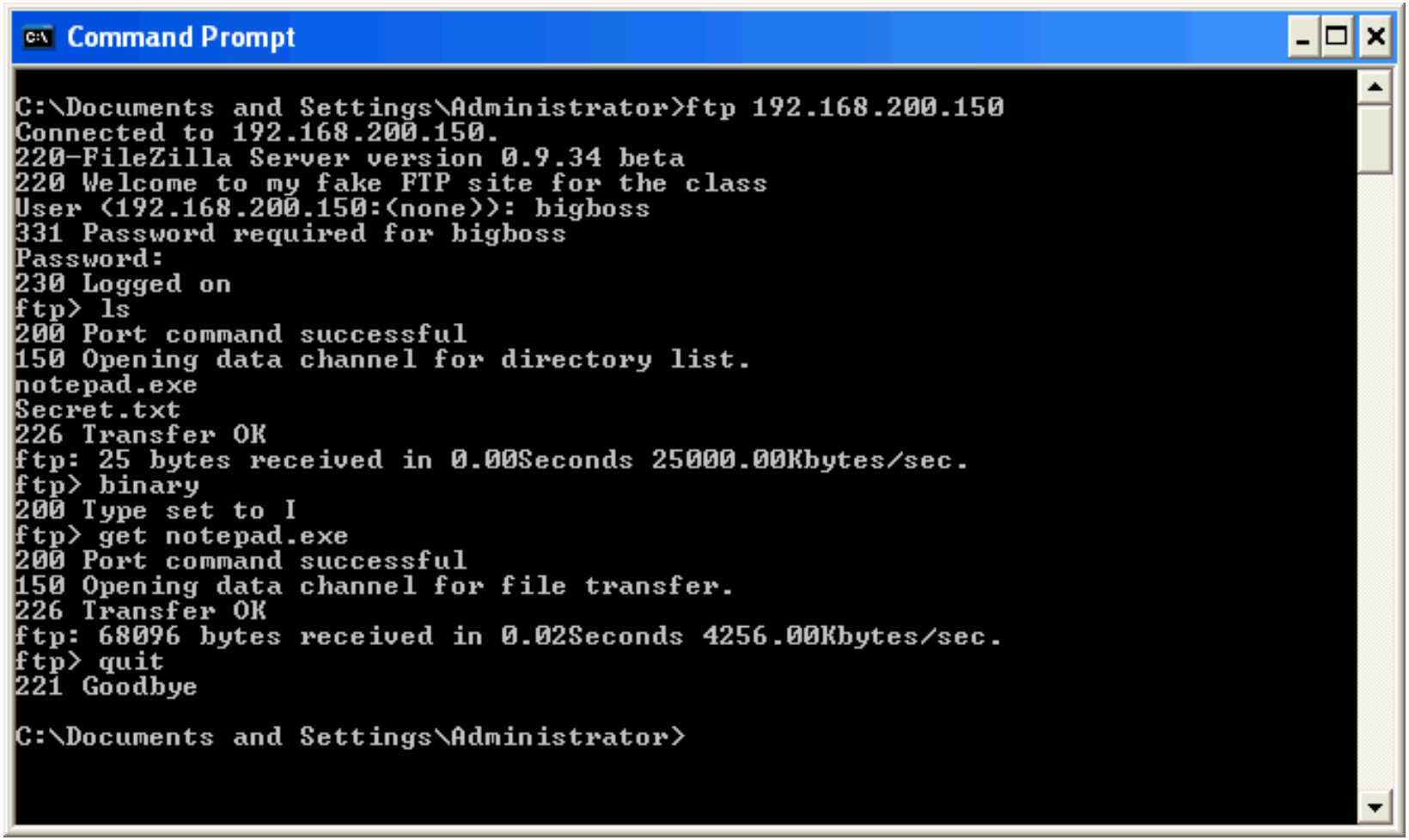

Sandia National Laboratories

# FTP Session Capture
# Text Transfer

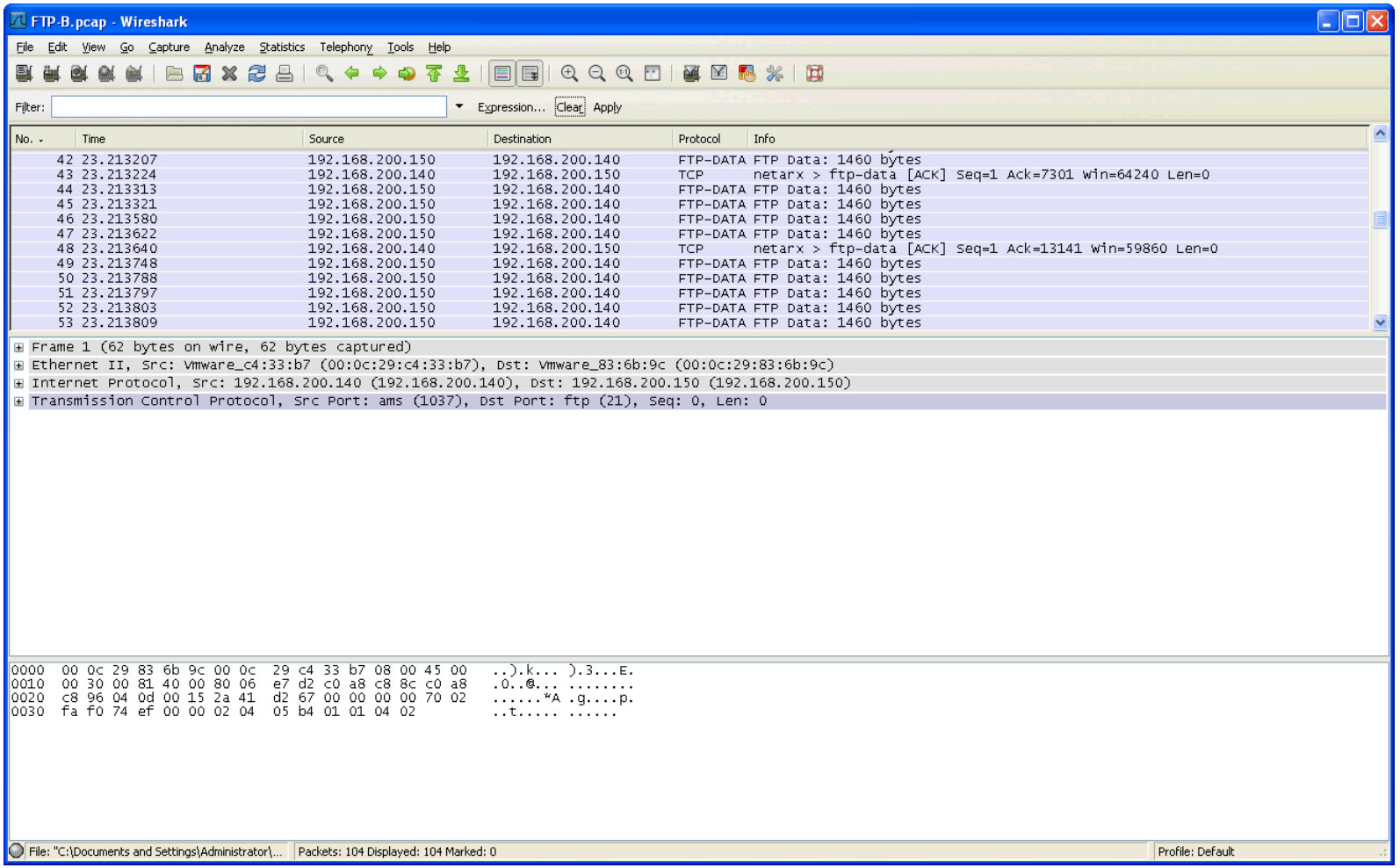



Sandia National Laboratories

# FTP Client Side
# Binary Transfer



# FTP Session Capture
# Binary Transfer




Sandia National Laboratories

# HyperText Transfer Protocol (HTTP)

- Used to transfer information between clients and servers
- Most widely used version HTTP/1.1
- Usually bound to TCP port 80
- Secure Sockets Layer (SSL) version (HTTPS) bound to TCP port 443
- Full list of commands at http://www.networksorcery.com/enp/protocol/http.htm
- Common requests: (majority of traffic)
  - GET – Get the requested document
  - POST – Upload data to be processed (Forms)
- Other requests:
  - HEAD – Request document header information
  - CONNECT - Used to tunnel HTTPS through a proxy server

# HyperText Transfer Protocol (HTTP)

- Error codes:
  - 1XX – Informational – Request received and processed
  - 2XX – Success – Action understood and accepted
  - 3XX – Redirection – Client needs to go to a different information source
  - 4XX – Client errors – Request contains bad syntax and can’t be processed
  - 5XX – Server Error – Server couldn’t process a valid request

# HTTP Client Headers

- **The web browser sends headers indicating the desired web server and what types of data it can accept:**

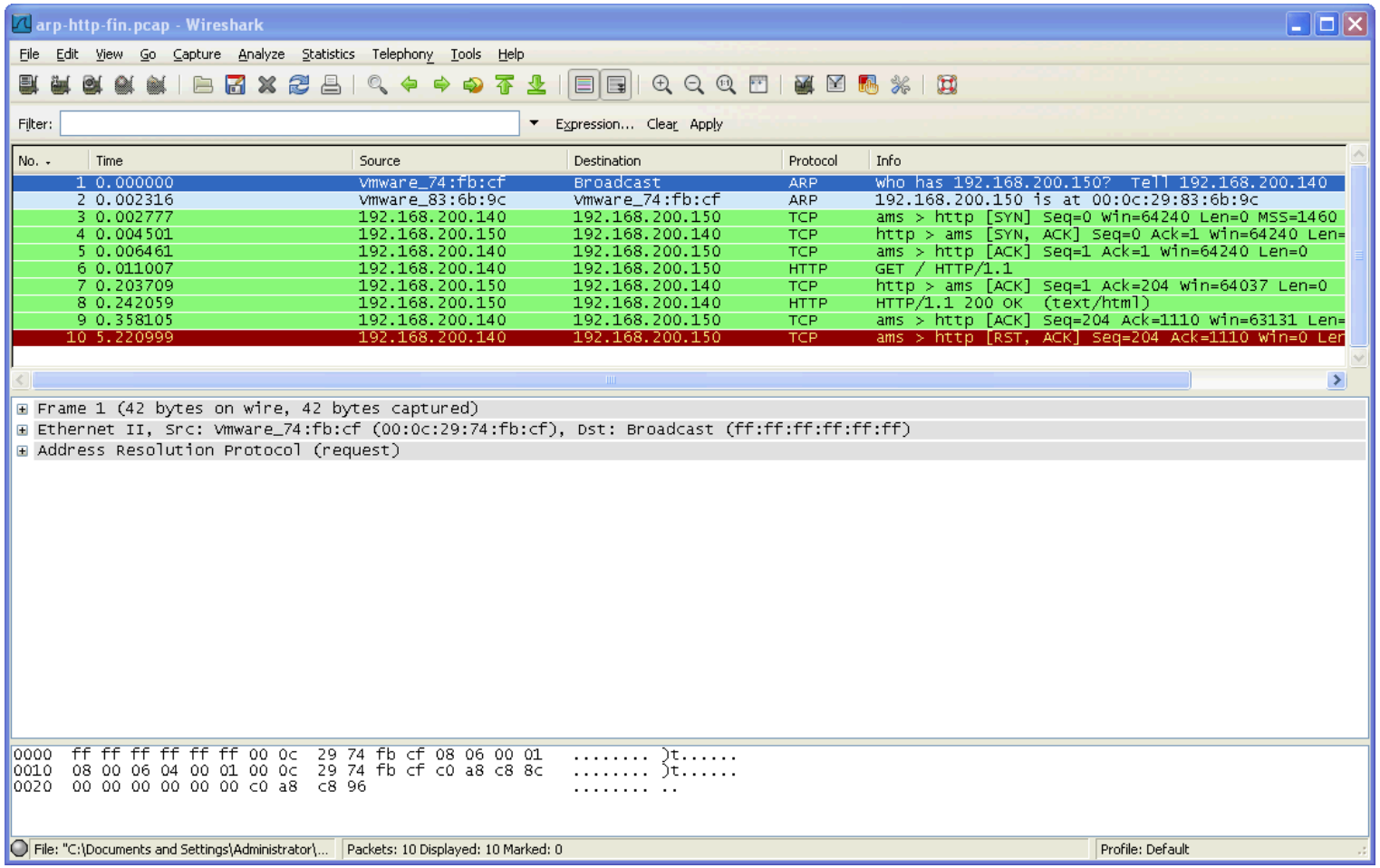
```
GET http://www.google.com/ HTTP/1.1
Host: www.google.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.0; en-US; rv:1.9.1.6)
Gecko/20091201 Firefox/3.5.6 (.NET CLR 3.5.30729)
Accept: Text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Proxy-Connection: keep-alive
```

# HTTP Server Headers

- **The web server precedes its responses with its own headers, which include the content type and length, and usually the server type and modification date:**

```
HTTP/1.1 200 OK
Date: Fri, 08 Jan 2010 17:01:49 GMT
Server: Apache/1.3.3.7 (Unix) (Red-Hat/Linux)
Last-Modified: Wed, 08 Nov 2009 23:11:55 GMT
Etag: "3f80f-1b6-3e1cb03b"
Accept-Ranges: bytes
Content-Length: 438
Connection: close
Content-Type: text/html; charset=UTF-8
```

# HTTP Capture

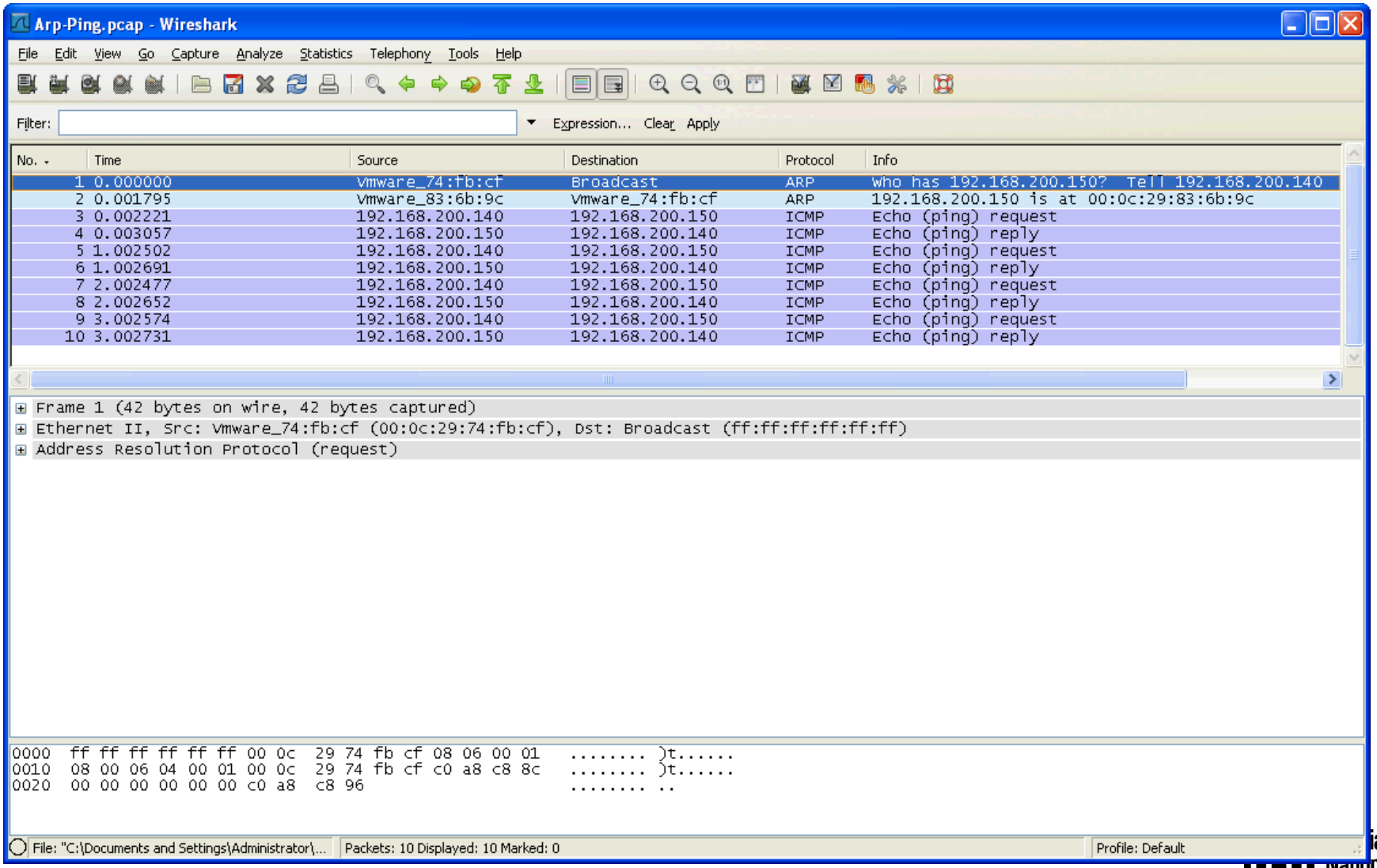



Sandia National Laboratories

# Internet Control Message Protocol (ICMP)

- Performs control, error reporting and information functions
  - Reporting unreachable destinations
  - Redirecting routing
  - Flow control
  - Best know program (PING)
  - ICMP parameters listed at http://www.iana.org/assignments/icmp-parameters
  - Common parameters – 0 echo reply, 3 destination unreachable, 8 echo request, 11 time exceeded

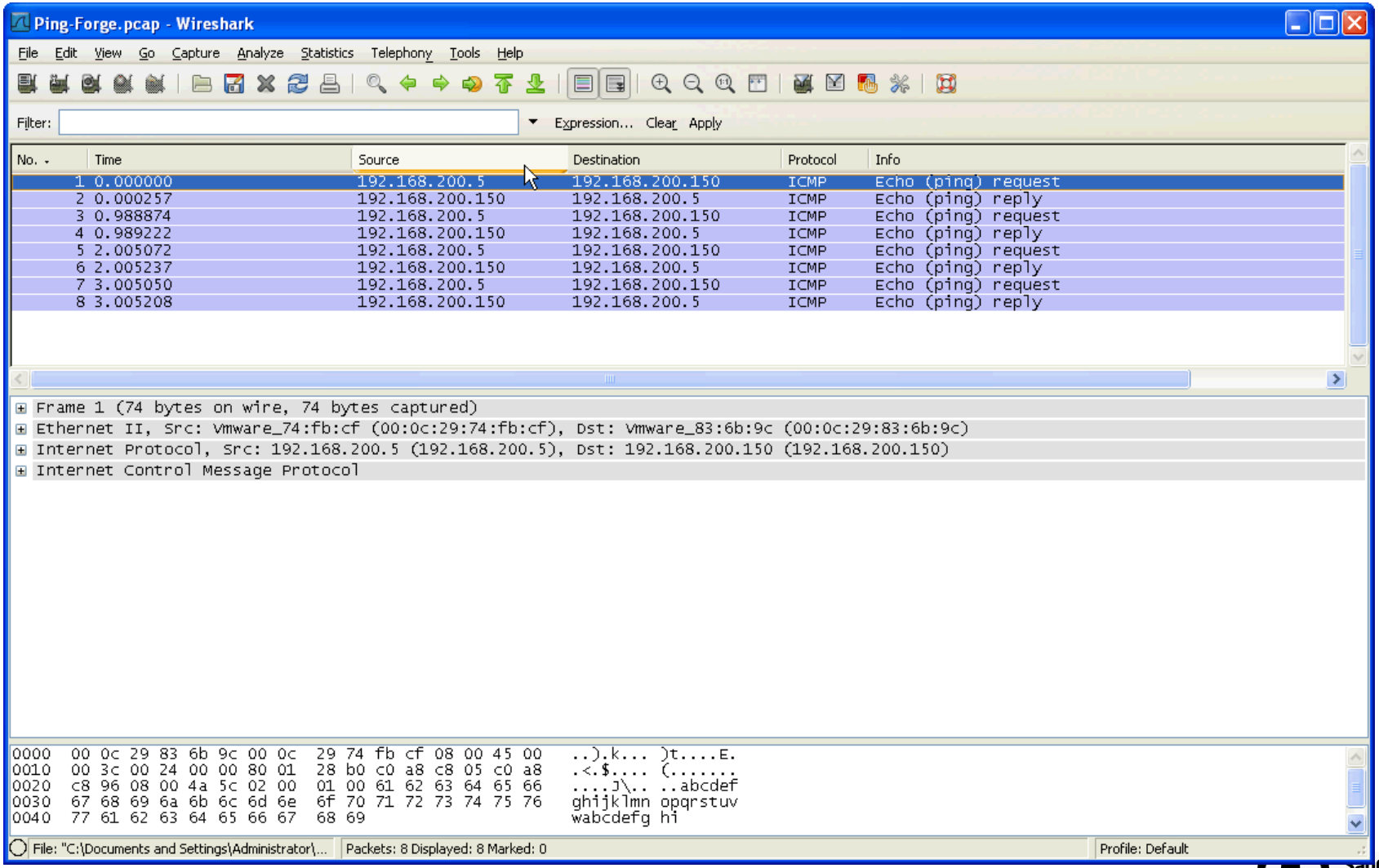
# PING



# PING Demo

- Can you believe everything you see on the network?

# PING Forge




Sandia National Laboratories

# Basic Troubleshooting

- Chapter 1: History & Models
- Chapter 2: Local Area Network (LAN)
- Chapter 3: Ethernet & Connections
- Chapter 4: Wireshark & Protocols
- Chapter 5: Basic Troubleshooting

Sandia National Laboratories

# Troubleshooting

- Tools you should take with you:
  - Laptop set to DHCP and known to work
  - Known good network cable
- Things to have around:
  - Install CD/DVD for the operating system
  - Known good NIC
  - Driver CD for NIC
  - Network analyzer or tester
  - Wireshark

# Troubleshooting

- Check to make sure everything is plugged in, turned on and you have a link light on the NIC
- Open a command prompt
- Type ipconfig /all make sure the system has an IP, subnet mask, gateway and DNS
- If the IP is 169.254.xxx.xxx (zeroconfig/autoconfig) then you have a problem
- Make sure the NIC is turned on in the BIOS
- If DHCP is used, try ipconfig /release then ipconfig /renew
- Check for speed/duplex mismatch

# Troubleshooting

- If you don't get a valid IP try:
  - ping 127.0.0.1 to make sure the NIC is working
  - ping the IP address of the system to check and make sure the stack is bound to the NIC
  - Try the laptop and make sure it works
- If you do get a valid IP try:
  - ping the IP of another system on the same network
  - ping the default gateway's IP
  - ping the DNS server

# Useful Commands

- arp -d * Flushes the ARP cache
- nbtstat -R  Flush the NetBIOS cache
- ipconfig /flushdns  Flushes the DNS cache
- nbtstat -RR  Re-registers with WINS
- ipconfig /registerdns Re-registers with DNS

# Windows, If all else fails

- First you can test your stack bindings by using the GUI (graphic user interface) program:  netsh diag gui (Windows XP)
- Repair and reset winsock in Windows Vista/7:
  - Start cmd.exe as administrator
  - Type netsh winsock reset - press enter
  - Restart the computer
- Reinstalling the TCP/IP protocol in Vista/7 can often correct issues that can't be resolved any other way:
  - Start cmd.exe as administrator
  - Type  netsh int ip reset c:\resetlog.txt - press enter
  - Restart the computer
  - Look at resetlog.txt for errors

# Other Useful Commands

- nslookup
- tracert

# nslookup

```
C:\>nslookup www.google.com
Server:  tester.Belkin
Address:  192.168.2.1
Non-authoritative answer:
Name:    www.l.google.com
Addresses:  209.85.225.103
          209.85.225.99
          209.85.225.147
          209.85.225.106
          209.85.225.105
          209.85.225.104
Aliases:  www.google.com
```

# Traceroute

```
C:\>
C:\>tracert www.google.com

Tracing route to www.l.google.com [209.85.225.104]
over a maximum of 30 hops:
  1    38 ms     1 ms     1 ms  tester.Belkin [192.168.2.1]
  2     8 ms     7 ms     7 ms  73.110.72.1
  3     7 ms     7 ms     7 ms  ge-4-7-ur01.albuquerque.nm.albuq.comcast.net [68.85.224.1]
  4     6 ms    26 ms     7 ms  te-8-4-ar01.albuquerque.nm.albuq.comcast.net [68.86.182.25]
  5    18 ms    16 ms    17 ms  te-0-6-0-4-cr01.denverqwest.co.ibone.comcast.net [68.86.91.25]
  6    18 ms    17 ms    33 ms  pos-0-10-0-0-cr01.denver.co.ibone.comcast.net [68.86.86.22]
  7    43 ms    41 ms    41 ms  pos-0-14-0-0-cr01.chicago.il.ibone.comcast.net [68.86.85.117]
  8    45 ms    44 ms    44 ms  pos-0-0-0-0-pe01.350ecermak.il.ibone.comcast.net [68.86.86.34]
  9    67 ms    66 ms    67 ms  as15169-1.350ecermak.il.ibone.comcast.net [75.149.230.198]
 10    67 ms    65 ms    66 ms  216.239.48.154
 11    66 ms    77 ms   123 ms  72.14.232.141
 12    66 ms    67 ms    69 ms  209.85.241.37
 13    73 ms    71 ms    71 ms  72.14.239.18
 14    68 ms    66 ms    65 ms  iy-in-f104.1e100.net [209.85.225.104]

Trace complete.
```

Sandia National Laboratories

# Recommended Reading

- "TCP/IP Illustrated, Volume 1, The Protocols" W. Richard Stevens, Addison-Wesley, 1994
- "TCP/IP Illustrated, Volume 2, The Implementation" Gary R. Wright and W. Richard Stevens, Addison-Wesley, 1995
- "TCP/IP Illustrated, Volume 3, TCP for Transactions, HTTP, NNTP, and the UNIX Domain Protocols" W. Richard Stevens, Addison-Wesley, 1996
- "Where wizards stay up late, the origins of the internet" Katie Hafner, Matthew Lyon, 1998
- "Wireshark Network Analysis" Laura Chappell, 2010
- Al Gore and the Internet, Robert Kahn and Vinton Cerf http://www.eecs.umich.edu/~fessler/misc/funny/gore,net.txt

# References

- http://www.tcpipguide.com/free/index.htm

# Questions?

Sandia National Laboratories