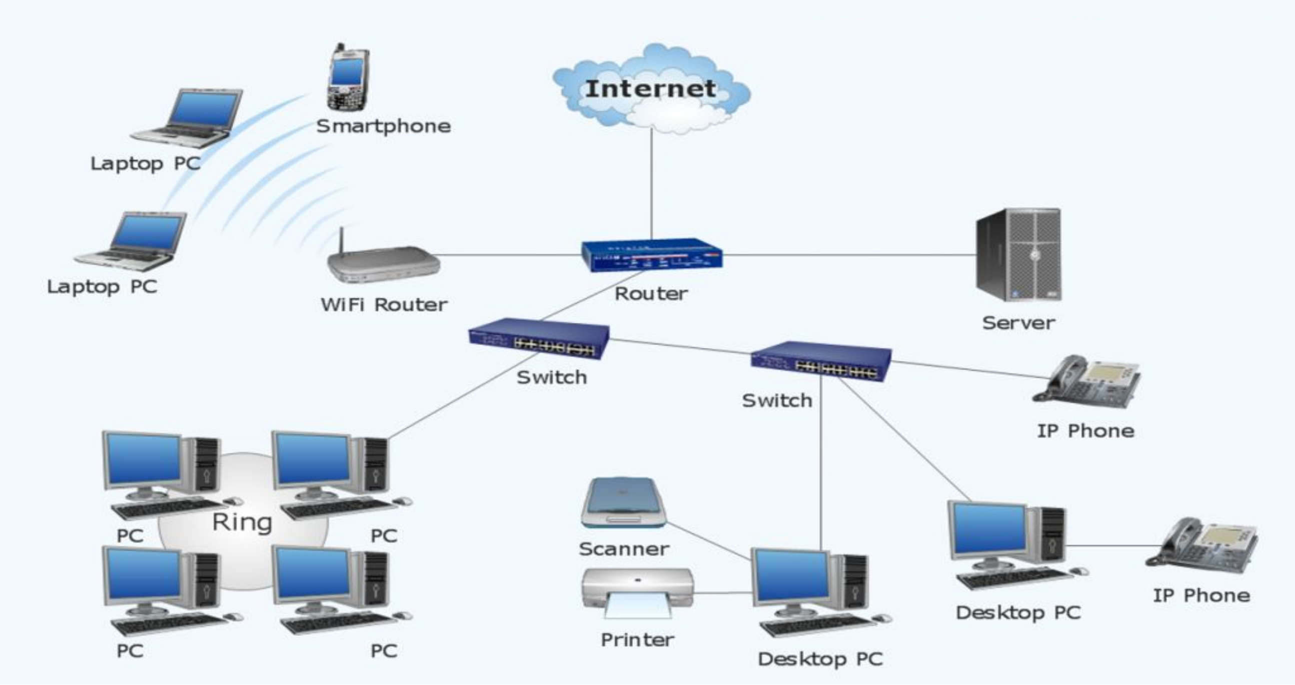
Network Security

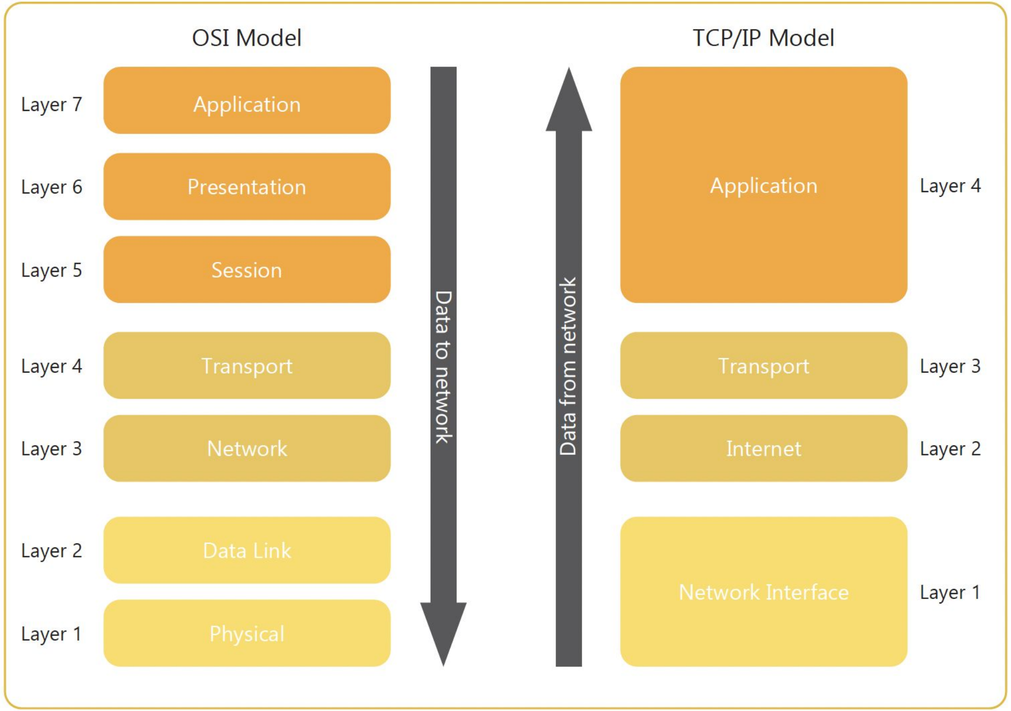
Computer Network



Types of Computer Networks

- Personal Area Network (PAN): Network of computer devices centered around an individual's workspace/home.
- Local Area Network (LAN): network that connects computers over a small geographical distance
- Metropolitan Area Network (MAN): network that connects computers over a larger distance such as within a city
- Wide Area Network (WAN): network that connects computers over a very large geographical distance

OSI Model and TCP/IP Model



OSI Model

Layer 7 Application

Layer 6 Presentation

Layer 5 Session

Layer 4 Transport

Layer 3 Network

Layer 2 Data Link

Layer 1 Physical

Data to network

Data from network

TCP/IP Model

Application Layer 4

Transport Layer 3

Internet Layer 2

Network Interface Layer 1

TCP/IP Model

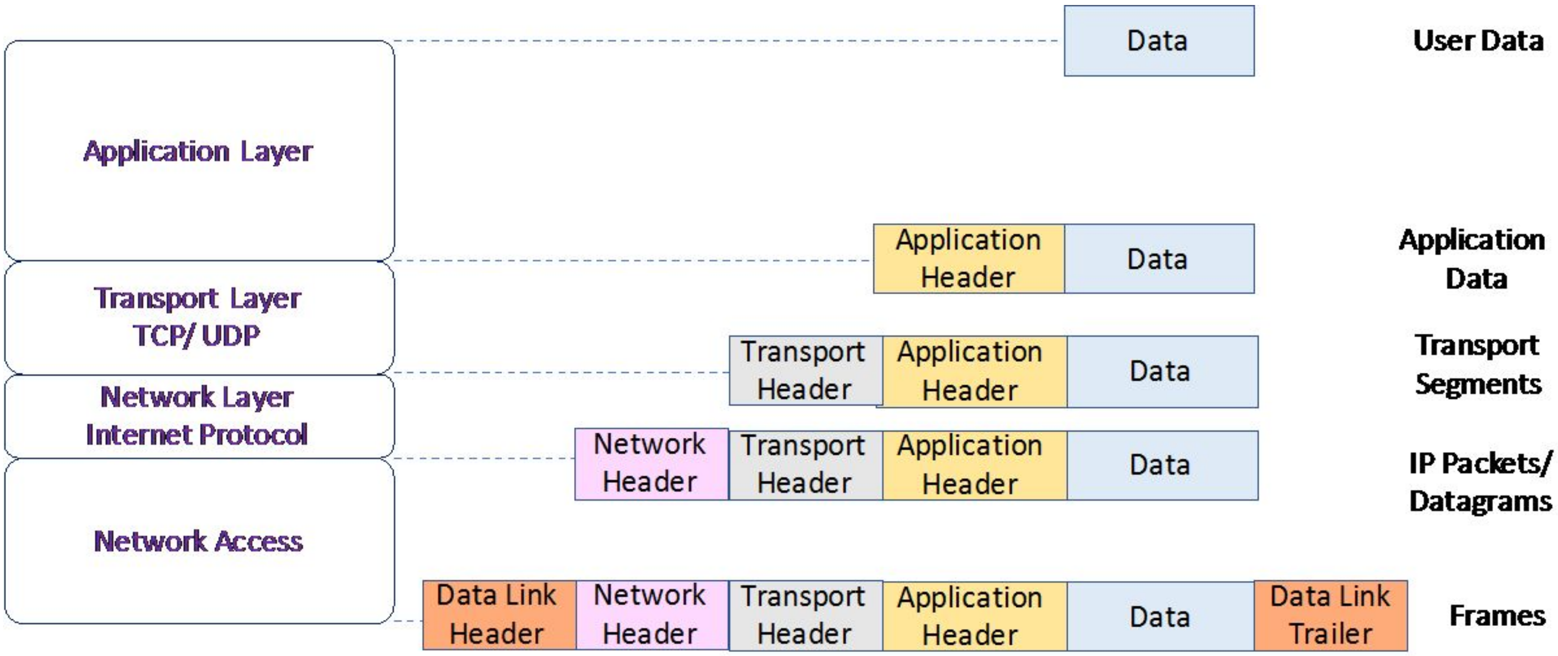
- **Application Layer:** Responsible for creating and processing user data between applications.

- **Transport Layer:** Responsible for data transfer between the application program running on the client and the application program running on the server.

- **Network (or Internetwork) Layer:** Responsible for transport of data from node to node in a network.

- **Network Interface/Link Layer:** Acts as the interface to the actual network hardware.
  This layer implements the actual topology of a local network that allows the internet layer to present an addressable interface.

TCP/IP Model



Application Layer

Transport Layer
TCP/ UDP

Network Layer
Internet Protocol

Network Access

Data — User Data

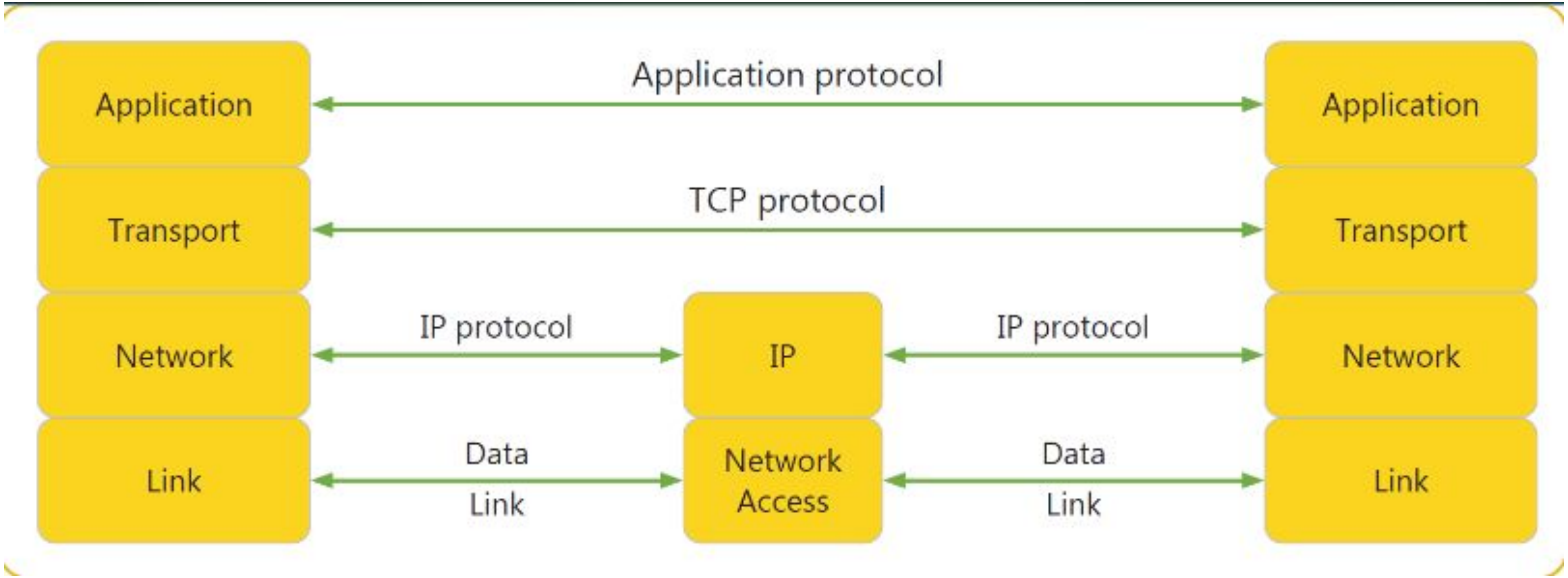
Application Header | Data — Application Data

Transport Header | Application Header | Data — Transport Segments

Network Header | Transport Header | Application Header | Data — IP Packets/ Datagrams

Data Link Header | Network Header | Transport Header | Application Header | Data | Data Link Trailer — Frames

TCP/IP Model


Application
Application protocol
Application
Transport
TCP protocol
Transport
Network
IP protocol
IP
IP protocol
Network
Link
Data
Link
Network
Access
Data
Link
Link

Protocols

- A protocol is a set of rules and standards that define a language that can be used to communicate.
- There are a great number of protocols used extensively in networking, and they are often implemented in different layers.

- Application Layer : HTTP, FTP, DNS, etc.,
- Transport layer: TCP, UDP, etc.,
- Network Layer: IP, ICMP, etc.,
- Network Interface Layer: PPP, ARP, etc.,

Addresses and Identifiers

- Network Access Layer : MAC Address
- Internet/Network Layer: IP Address
- Transport layer: Port Number

MAC Address

- Media Access Control (MAC) Address is a 6-byte (48-bits) address that is unique to each networking device/interface
- Also known as Physical/Hardware address
- Generally written as a hexadecimal number
- It has two parts. The first three bytes indicate the manufacturer of the Network Interface Card (NIC) and the last three bytes are a unique number assigned to the NIC by the manufacturer
- Randomized MAC: Introduced to provide privacy especially in case of mobiles and laptops.
  - Poses challenges for device tracking/authentication in organizations

IP Address

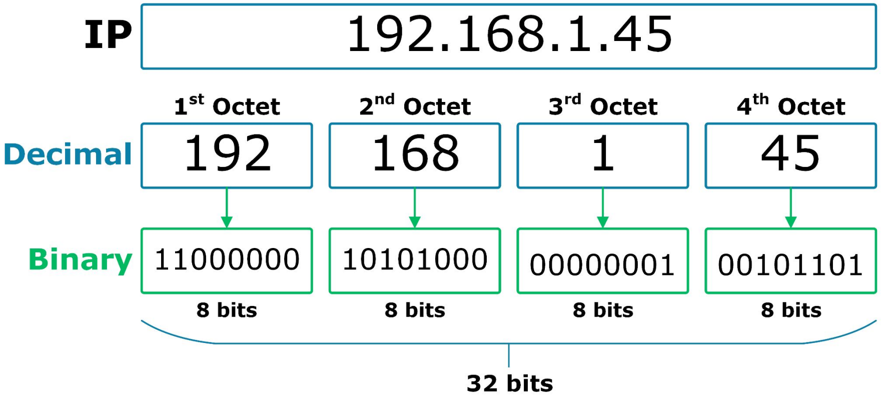
- Also known as Virtual Address
- So each device has a Physical address and a Virtual Address
- There are two versions of IP addresses: IPv4 and IPv6
- IPv4 uses 32 bit address
- Each address has two parts - network part and host part
- Generally, IP addresses are assigned by the ISP or a system administrator
- Public Vs Private IP addresses
- Dynamic Vs Static IP addresses

IP Address

- IP addresses are managed by the Internet Assigned Numbers Authority (IANA) which has overall responsibility for the IP address pool and by the Regional Internet Registries (RIRs) to which IANA distributes large blocks of addresses.
- Dynamic Host Configuration Protocol (DHCP) is a protocol that automatically provides an IP address to a host
- Loopback address is a virtual interface that loops back to the same host

IPv4 Address

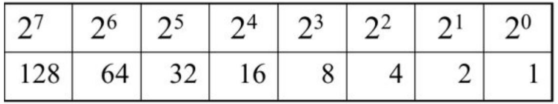
- A dotted decimal number made of 32 bits
- It is divided into 4 Octets
- Value of each octet ranges from 0 to 255



IP 192.168.1.45

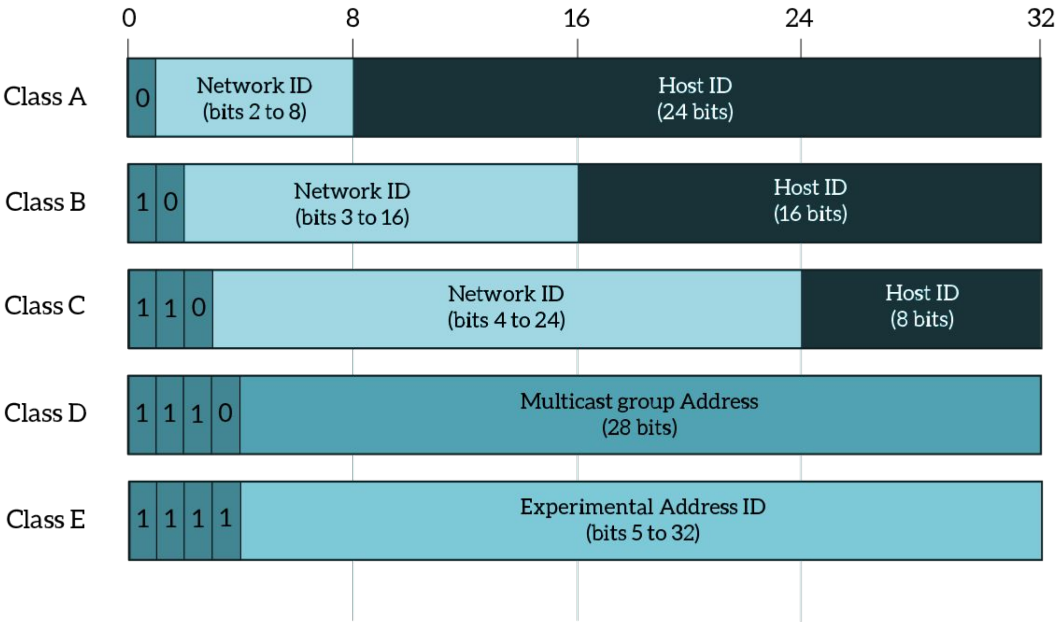
	1st Octet	2nd Octet	3rd Octet	4th Octet
Decimal	192	168	1	45
Binary	11000000	10101000	00000001	00101101
	8 bits	8 bits	8 bits	8 bits

32 bits



2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1

IPv4 Address Classes



0 8 16 24 32

Class A: 0 | Network ID (bits 2 to 8) | Host ID (24 bits)

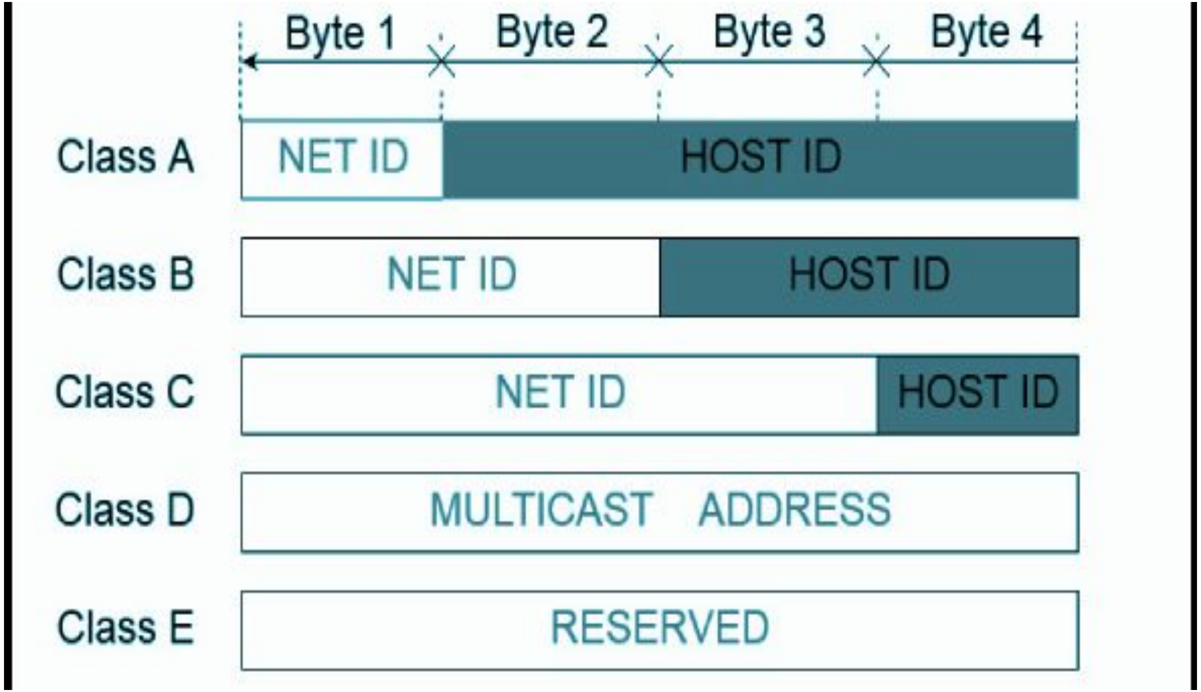
Class B: 1 0 | Network ID (bits 3 to 16) | Host ID (16 bits)

Class C: 1 1 0 | Network ID (bits 4 to 24) | Host ID (8 bits)

Class D: 1 1 1 0 | Multicast group Address (28 bits)

Class E: 1 1 1 1 | Experimental Address ID (bits 5 to 32)

IPv4 Address Classes



Byte 1 Byte 2 Byte 3 Byte 4

Class A: NET ID | HOST ID

Class B: NET ID | HOST ID

Class C: NET ID | HOST ID

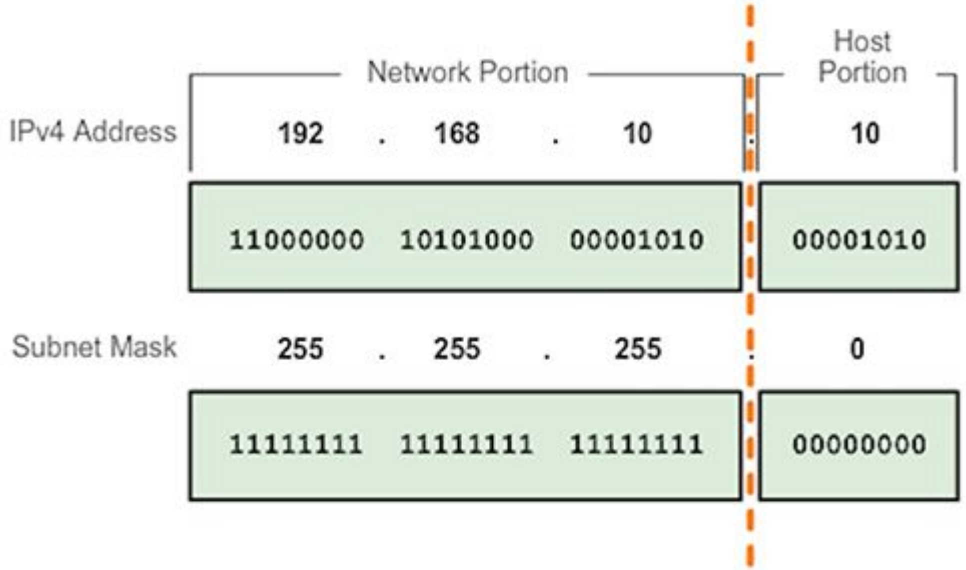
Class D: MULTICAST ADDRESS

Class E: RESERVED

IP Subnetting

- Dividing a network into smaller networks
- Subnet mask is used to differentiate between the network ID and host ID
- Length of the subnet mask (Number of 1s) is added as a suffix to the IP address
- Example: 172.30.26.12/18   (here the first 18 bits represent the network portion)

IPv4 Network Address



Private IP Addresses

- Addresses within this private address space are only unique within a given private network.
- An IP address within these ranges is therefore considered non-routable, as it is not unique. Any private network that needs to use IP addresses internally can use any address within these ranges without any coordination with IANA or an Internet registry.
- Private IP Address Ranges
  - Class A: 10.0.0.0 to 10.255.255.255
  - Class B: 172.16.0.0 to 172.31.255.255
  - Class C: 192.168.0.0 to 192.168.255.255

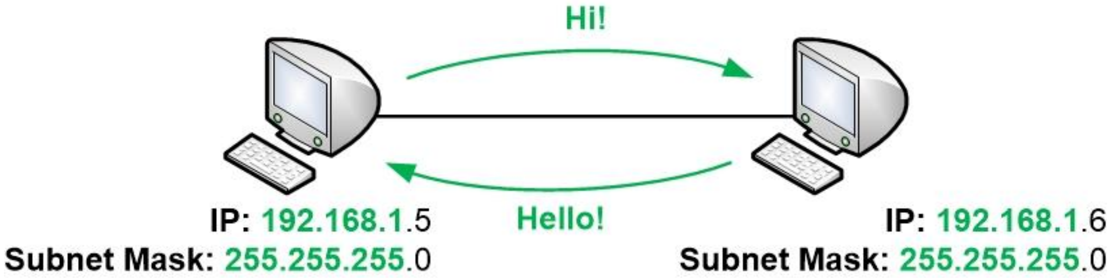
Gateway

- Gateway is a node located at the boundary of a network and manages all data that inflows or outflows from that network.
- It forms a passage between two different networks operating with different transmission protocols.
- IP address of the Gateway should be part of the network that it is connecting

Port Numbers

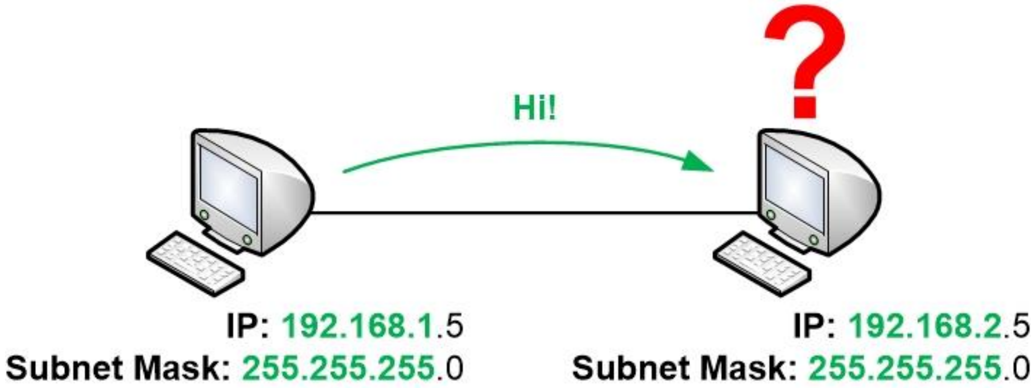
- A port is an address on a network device that can be associated to a specific piece of software.
- It is not a physical interface or a location, but it allows the server to be able to communicate using more than one application.
- It is a 16 bits number. Ranges from 0 to 65535
- Numbers 0 to 1023 are reserved for common applications. These are known as well-known ports

IP Addressing

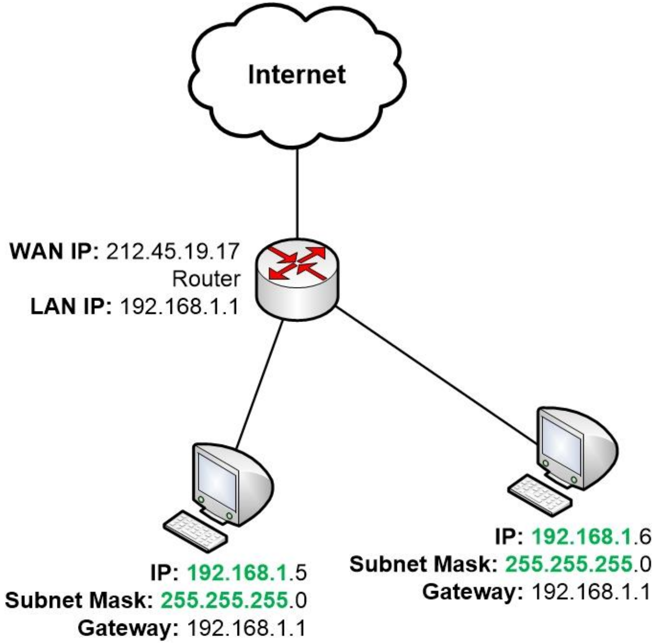


https://www.homenethowto.com/basics/default-gateway-finding-other-ip-networks/

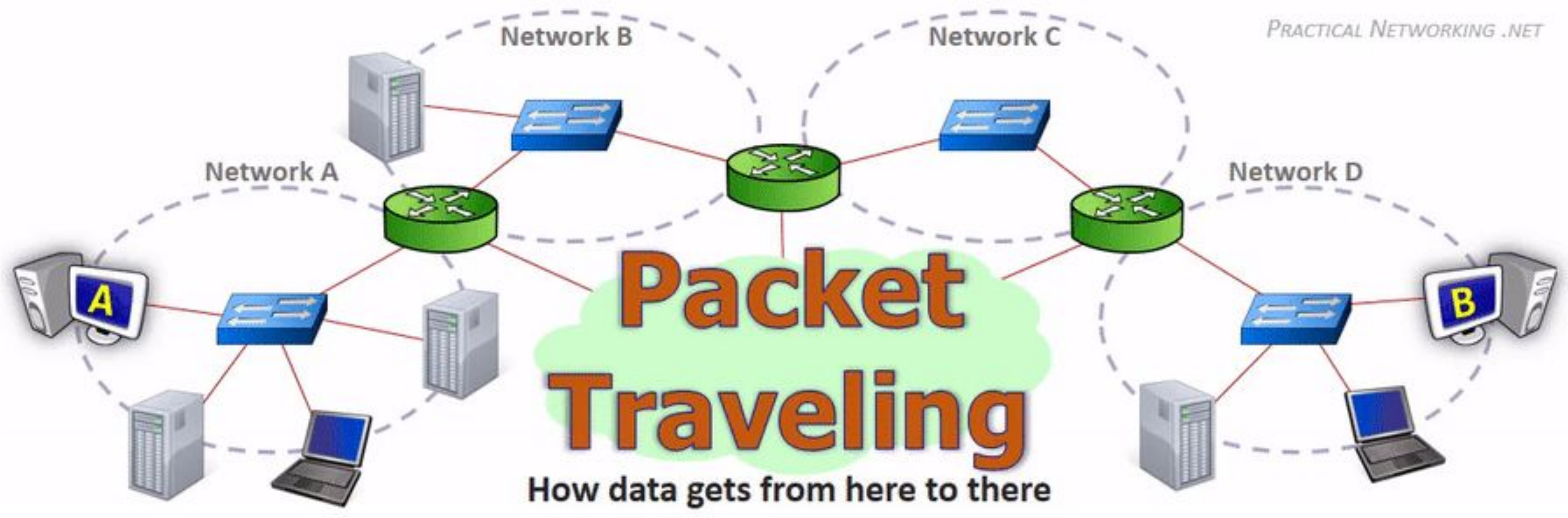
IP Addressing



IP Addressing

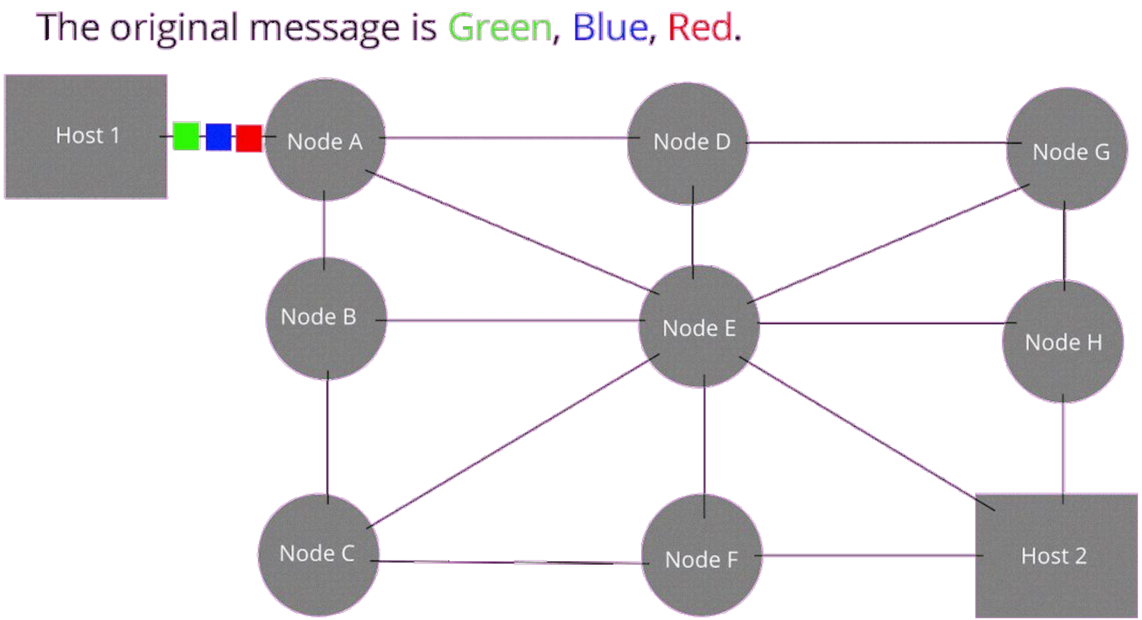


Packet Travelling



https://www.practicalnetworking.net/series/packet-traveling/packet-traveling/

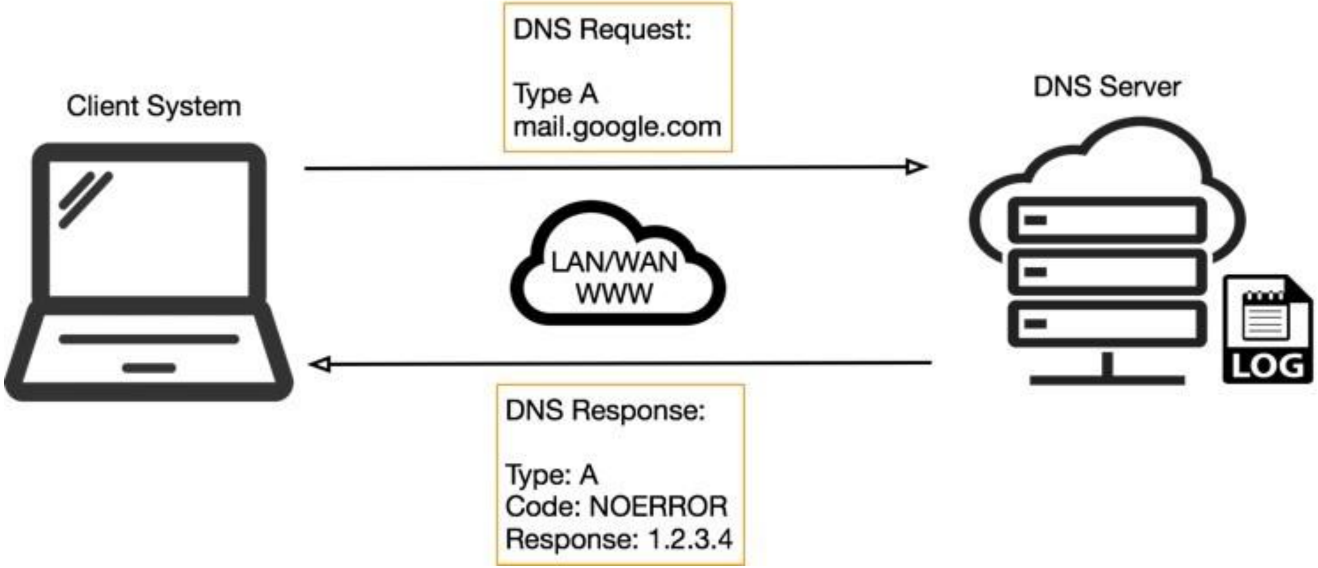
Packet Travelling



The original message is Green, Blue, Red.

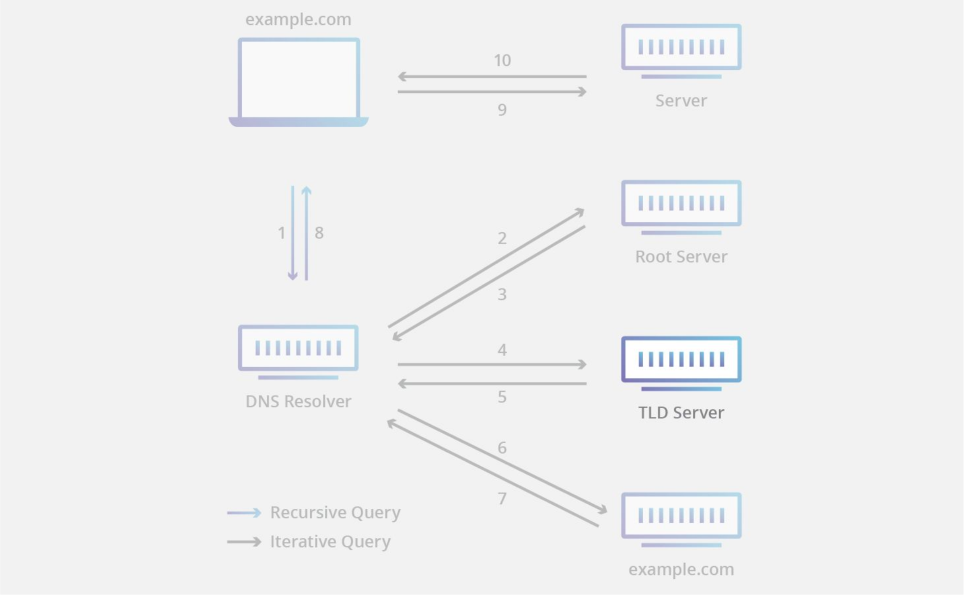
https://commons.wikimedia.org/wiki/File:Packet_Switching.gif

DNS Protocol

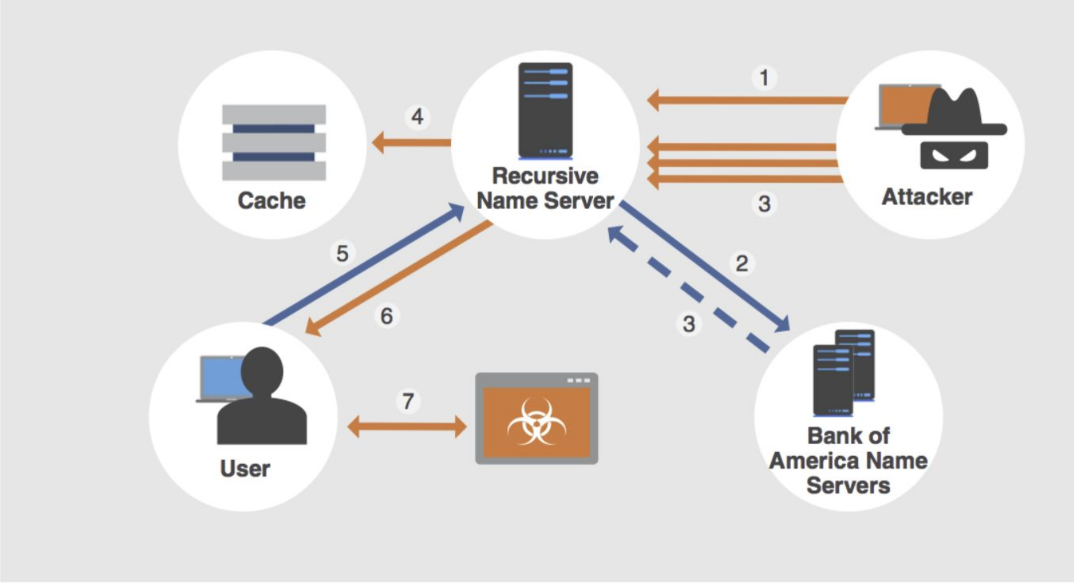


DNS Protocol

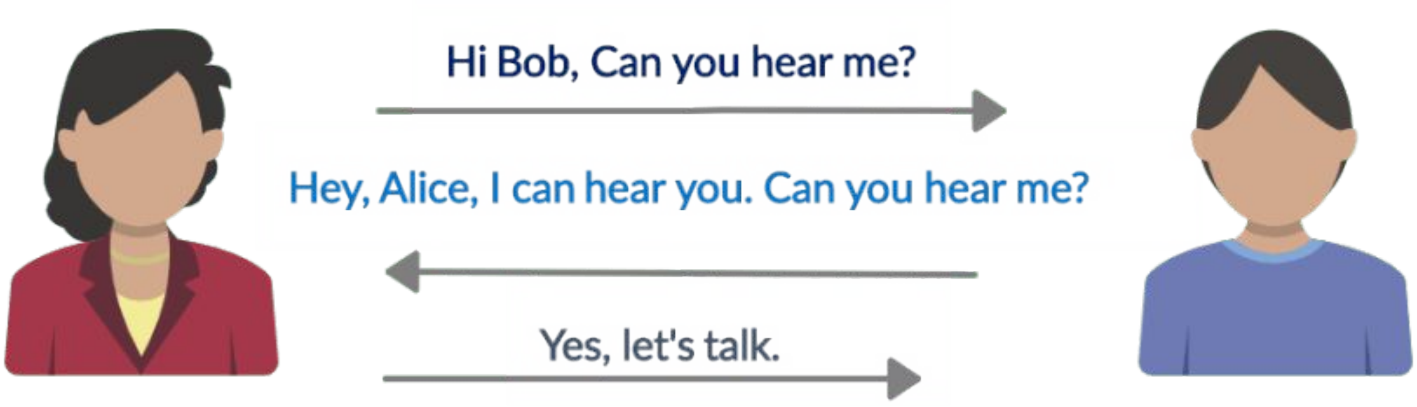
DNS Protocol



DNS Cache Poisoning Attack



Establish Connection

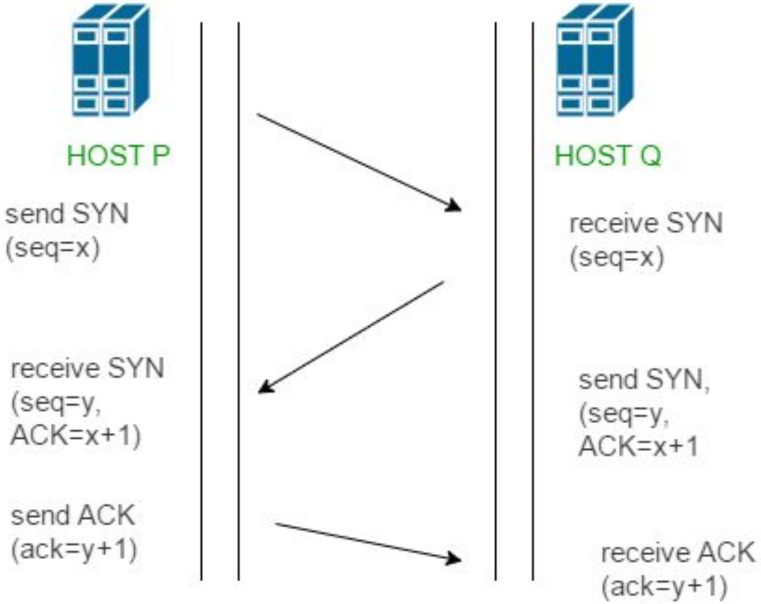


Hi Bob, Can you hear me?

Hey, Alice, I can hear you. Can you hear me?

Yes, let's talk.

Establish Connection using 3-way Handshake



HOST P

send SYN
(seq=x)

receive SYN
(seq=y,
ACK=x+1)

send ACK
(ack=y+1)

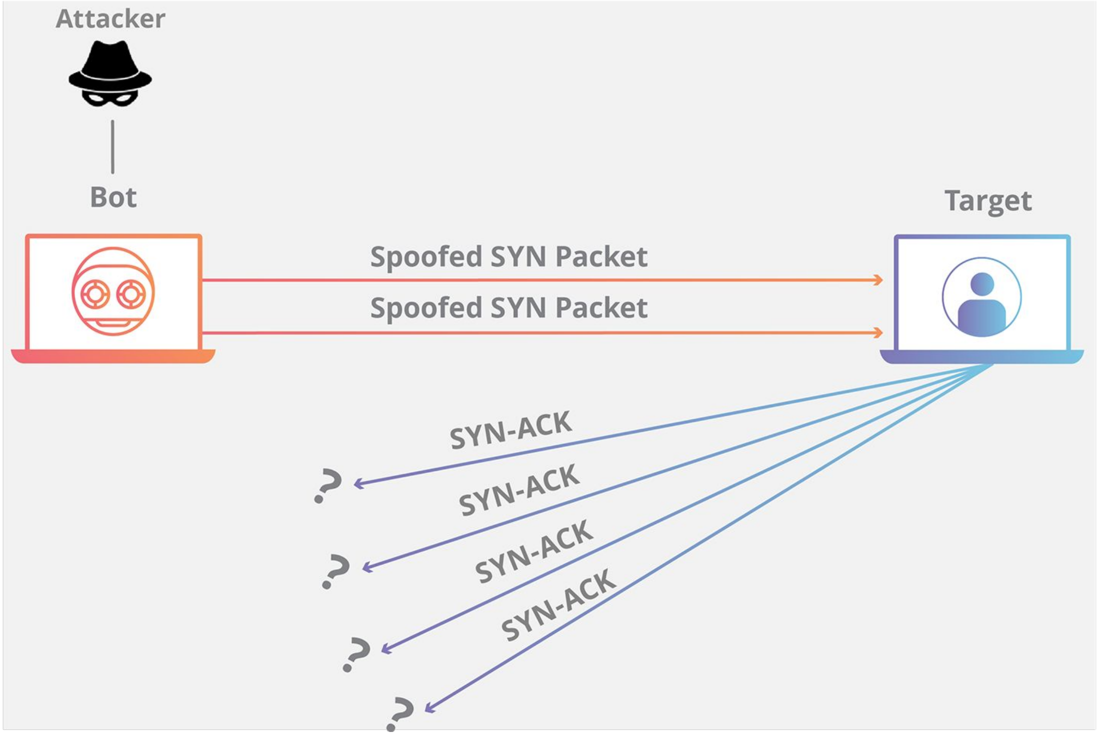
HOST Q

receive SYN
(seq=x)

send SYN,
(seq=y,
ACK=x+1

receive ACK
(ack=y+1)

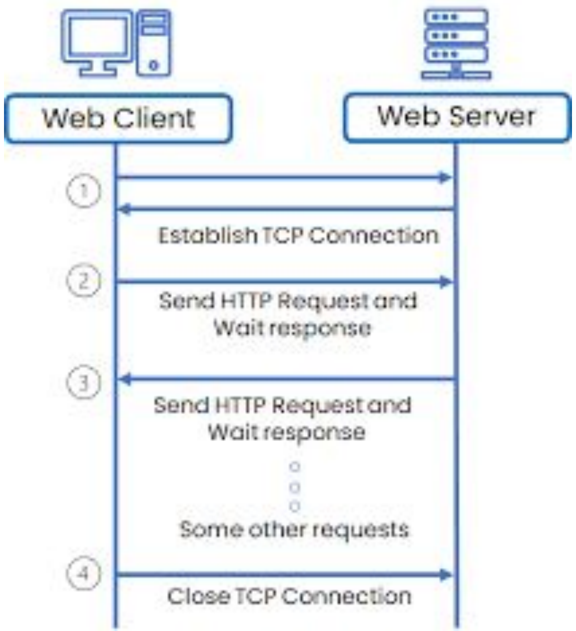
SYN Flood Attack

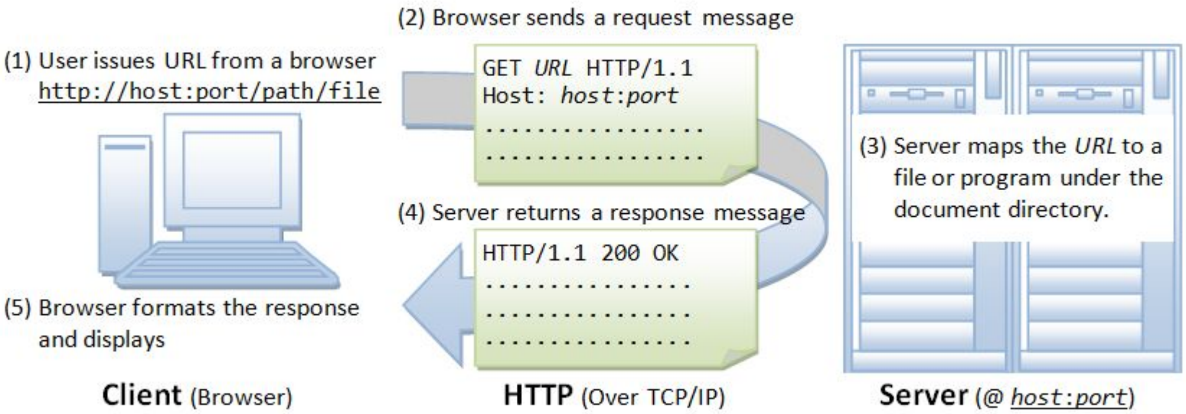


Different Ways of DoS Attack

- Transmission Failure
- Traffic Redirection
- DNS Attack
- Connection Flooding

Exchanging Data using HTTP




(1) User issues URL from a browser
http://host:port/path/file
(2) Browser sends a request message
(3) Server maps the URL to a file or program under the document directory.
(4) Server returns a response message
(5) Browser formats the response and displays
Client (Browser)
HTTP (Over TCP/IP)
Server (@ host:port)

```
GET URL HTTP/1.1
Host: host:port
.................
.................
```

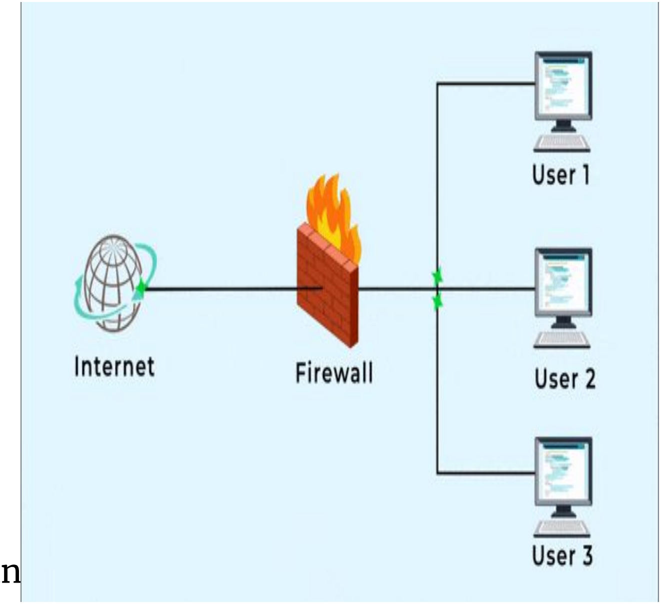
```
HTTP/1.1 200 OK
................
................
................
```

Security Features provided by TLS

- Confidentiality
- Integrity
- Authentication

Firewall

- First line of defence in a network
- Prevents unauthorised outsiders from accessing internal resources
- Prevents insiders from transferring sensitive information outside the network and accessing unsecured resources
- It can be a software or hardware or both
- Security measure that filters incoming and outgoing traffic based on predefined rules
- Rules are generally specified in terms of IP addresses, ports, etc
- These rules form the firewall policy
- Firewall policy must be carefully configured and frequently evaluated an updated
- Can also use multiple network security perimeter



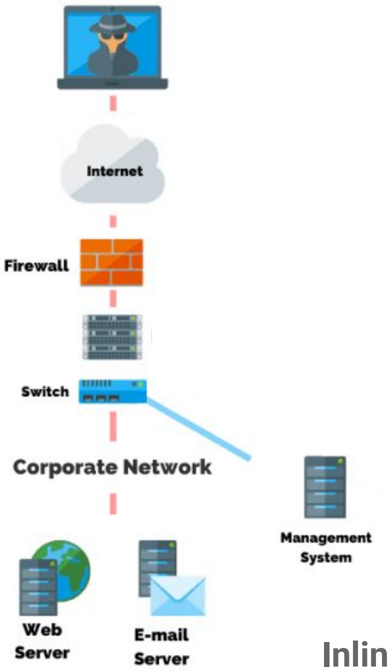
Types of Firewalls

- Packet Filtering Firewalls: Simple firewalls. Inspect packets based on IP, protocol, port, etc
- Stateful Inspection Firewalls: More advanced firewalls. Inspect complete connections and sessions.
- Web Application Firewalls: used to protect websites/web applications
- Personal Firewall: an application which controls network traffic to and from a computer, permitting or denying communications based on a security policy

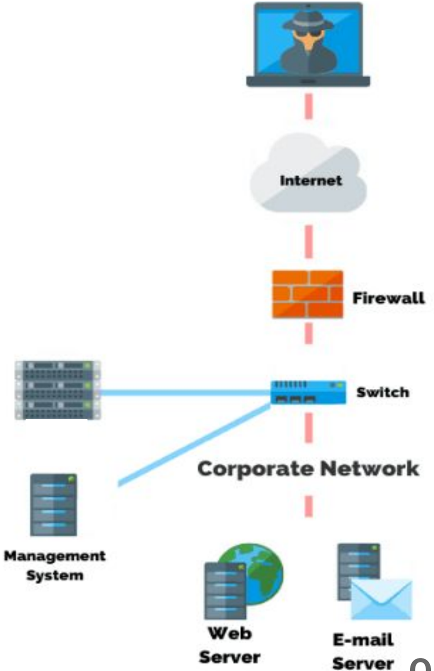
Intrusion Detection and Prevention Systems

- Intrusion Detection Systems (IDS): Security measure that monitors the traffic for any malicious activities or policy violations and sends an alert if detected.
- Intrusion Prevention Systems (IPS): Measure that inspects the traffic and proactively stops any malicious traffic
- Can work in inline or out-of-band/end host mode
- Can use anomaly-based detection or signature based detection
- There are two main types:
    - Network Intrusion Detection and Prevention System (NIDPS)
    - Host Intrusion Detection and Prevention System (HIDPS)

Intrusion Detection and Prevention Systems

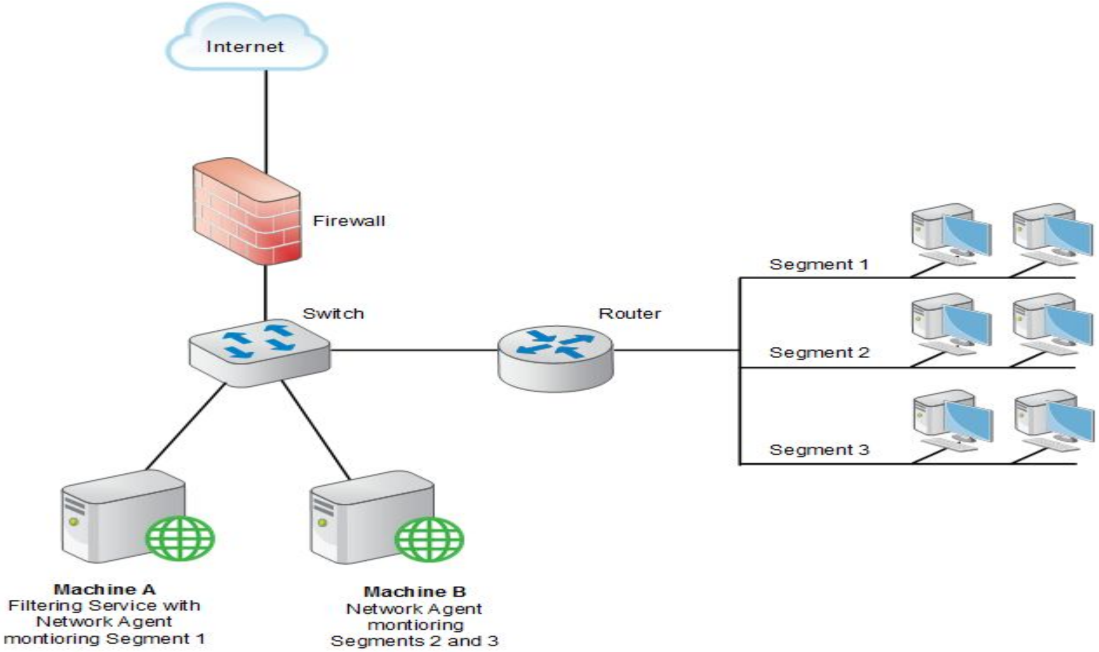


Inline Systems



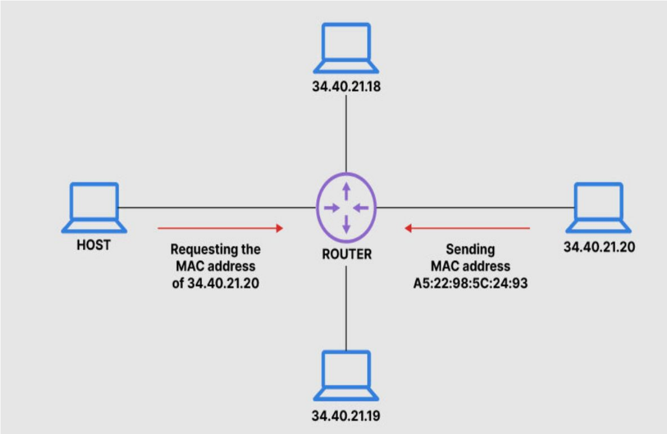
Out-of-Band Systems

Network Segmentation

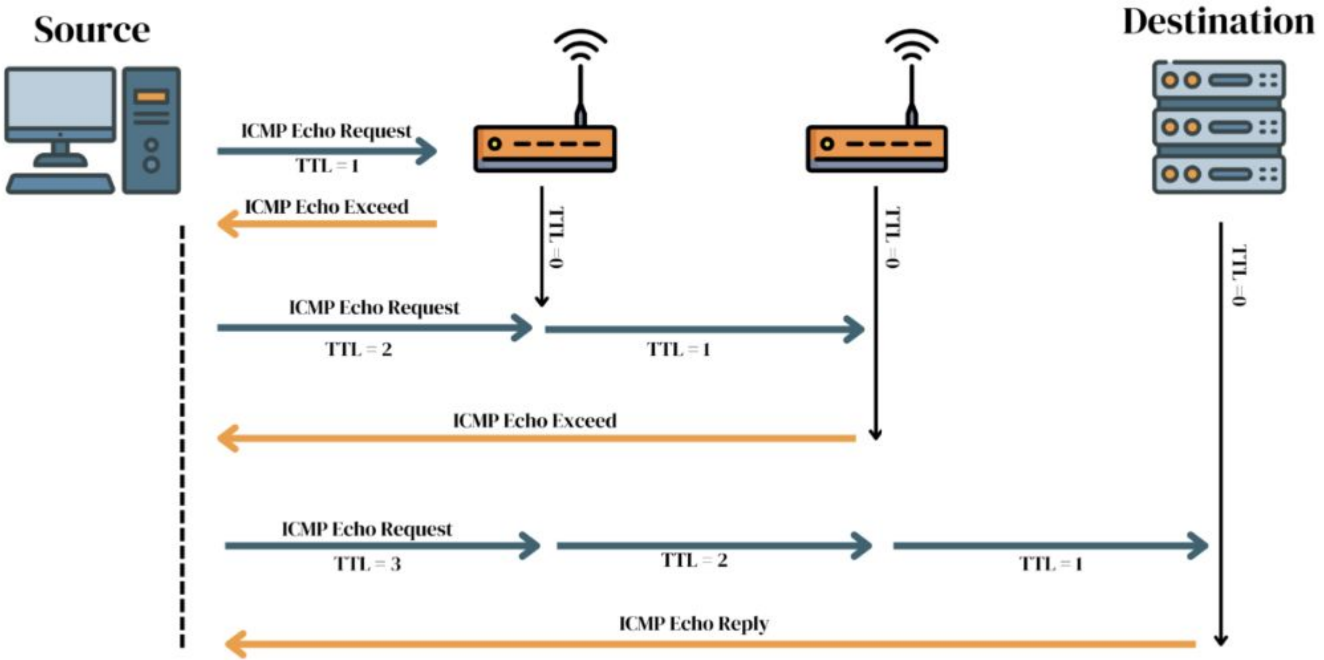


Address Resolution Protocol

- Maps IP address to MAC address
- A host broadcasts an ARP request asking for a MAC address
- The corresponding system will respond with its MAC address
- There is no verification of the responder
- This needs to ARP Spoofing/ARP Poisoning
- IPv6 uses Neighbor Discovery Protocol (NDP) that uses cryptographic keys to verify host identities


34.40.21.18
HOST
Requesting the MAC address of 34.40.21.20
ROUTER
Sending MAC address A5:22:98:5C:24:93
34.40.21.20
34.40.21.19

Traceroute Command



https://www.cloudns.net/blog/traceroute-command-tracert/

Port Scanning

- Involves scanning one or more IP addresses and recording open ports and known vulnerabilities present in them
- It is useful for network administrators to monitor the network
- But it can also be used by attackers to analyse victim's network
- Many port scanning tools are available

Wireshark

- Open source network protocol analyser
- Filters Traffic by
  - Protocols
  - A specific port
  - Specific direction
  - Network address
  - Port range

- More user-friendly than tcpdump

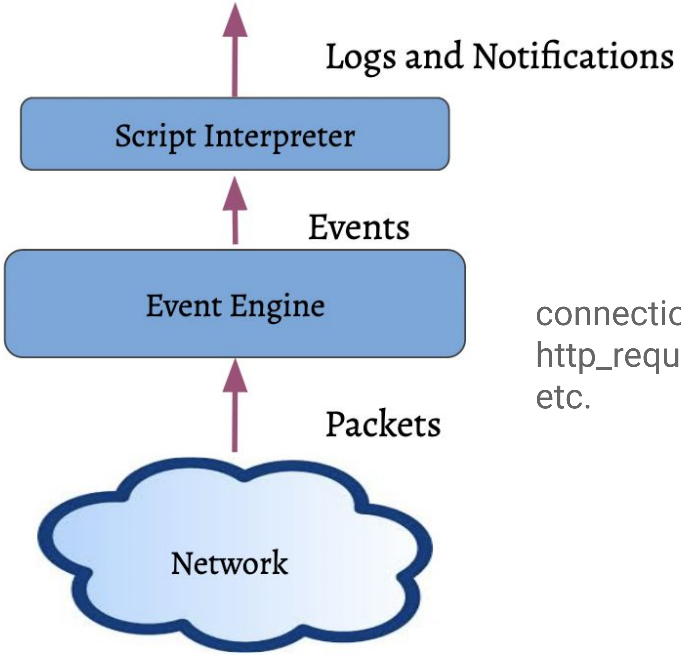
Wireshark

- Can be used for
  - Understanding network protocols
  - Network troubleshooting
  - Security and Incident Response

Zeek

- Network Network Monitoring System
- Open source
- Previously known as Bro
- Developed in 1995 at International Computer Science Institute (ICSI), Berkeley
- Converts raw network traffic into comprehensive logs
- Out-of-band analysis
- Good for threat hunting
- Generates compact logs
- Reduces memory requirement
- Packet capture + Traffic filtering + Scripting

Zeek Architecture



connection_attempt, http_request, http_response, etc.

Zeek Events

- new_connetion
- http_request
- http_response
- dns_query
- …….

Zeek Scripts

- With Zeek installation we get a collection of preloaded scripts which generate log files and alarms

- By default, the scripts in the base directory will be used

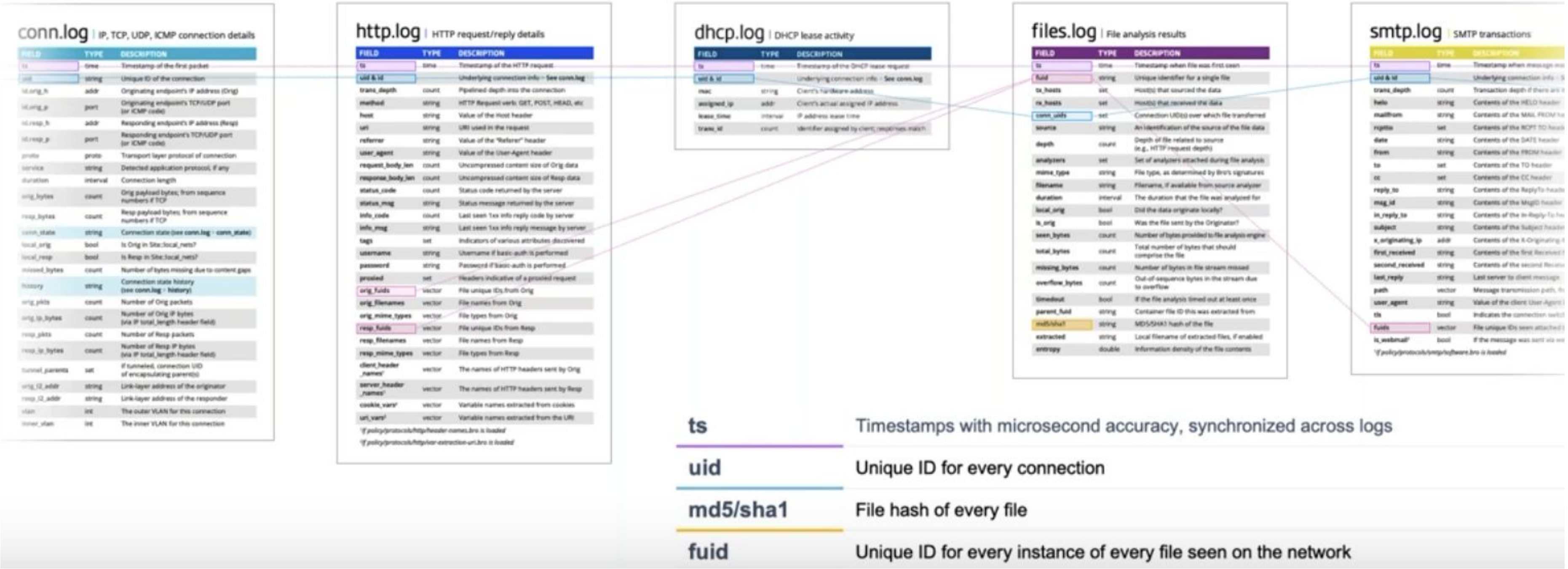
- We can also import the scripts from other directories

- Depending on the traffic and the scripts used, log files will be generated for different protocols and notices

Log Files

- **Protocol logs**
  - Conn.log : TCP/UDP/ICMP connections
  - http: HTTP requests and replies
  - dns.log : DNS activity
  - dhcp.log : DHCP leases
  - ...
- **File Logs**
  - files.log : File analysis results
  - pe.log : Portable Executable (PE)
  - X509.log : X.509 certificate info
- **Detection logs**
  - intel.log : Intelligence data matches
  - notice.log: Zeek notices
  - notice_alarm.log: The alarm stream
  - ...

Zeek Logs Interlinked



conn.log | IP, TCP, UDP, ICMP connection details

FIELD	TYPE	DESCRIPTION
ts	time	Timestamp of the first packet
uid	string	Unique ID of the connection
id.orig_h	addr	Originating endpoint's IP address (Orig)
id.orig_p	port	Originating endpoint's TCP/UDP port (or ICMP code)
id.resp_h	addr	Responding endpoint's IP address (Resp)
id.resp_p	port	Responding endpoint's TCP/UDP port (or ICMP code)
proto	proto	Transport layer protocol of connection
service	string	Detected application protocol, if any
duration	interval	Connection length
orig_bytes	count	Orig payload bytes; from sequence numbers if TCP
resp_bytes	count	Resp payload bytes; from sequence numbers if TCP
conn_state	string	Connection state (see conn.log > conn_state)
local_orig	bool	Is Orig in Site::local_nets?
local_resp	bool	Is Resp in Site::local_nets?
missed_bytes	count	Number of bytes missing due to content gaps
history	string	Connection state history (see conn.log > history)
orig_pkts	count	Number of Orig packets
orig_ip_bytes	count	Number of Orig IP bytes (via IP total_length header field)
resp_pkts	count	Number of Resp packets
resp_ip_bytes	count	Number of Resp IP bytes (via IP total_length header field)
tunnel_parents	set	If tunneled, connection UID of encapsulating parent(s)
orig_l2_addr	string	Link-layer address of the originator
resp_l2_addr	string	Link-layer address of the responder
vlan	int	The outer VLAN for this connection
inner_vlan	int	The inner VLAN for this connection

http.log | HTTP request/reply details

FIELD	TYPE	DESCRIPTION
ts	time	Timestamp of the HTTP request
uid & id		Underlying connection info > See conn.log
trans_depth	count	Pipelined depth into the connection
method	string	HTTP Request verb: GET, POST, HEAD, etc
host	string	Value of the Host header
uri	string	URI used in the request
referrer	string	Value of the "Referer" header
user_agent	string	Value of the User-Agent header
request_body_len	count	Uncompressed content size of Orig data
response_body_len	count	Uncompressed content size of Resp data
status_code	count	Status code returned by the server
status_msg	string	Status message returned by the server
info_code	count	Last seen 1xx info reply code by server
info_msg	string	Last seen 1xx info reply message by server
tags	set	Indicators of various attributes discovered
username	string	Username if basic-auth is performed
password	string	Password if basic-auth is performed
proxied	set	Headers indicative of a proxied request
orig_fuids	vector	File unique IDs from Orig
orig_filenames	vector	File names from Orig
orig_mime_types	vector	File types from Orig
resp_fuids	vector	File unique IDs from Resp
resp_filenames	vector	File names from Resp
resp_mime_types	vector	File types from Resp
client_header_names¹	vector	The names of HTTP headers sent by Orig
server_header_names¹	vector	The names of HTTP headers sent by Resp
cookie_vars²	vector	Variable names extracted from cookies
uri_vars²	vector	Variable names extracted from the URI

¹if policy/protocols/http/header-names.bro is loaded

²if policy/protocols/http/var-extraction-uri.bro is loaded

dhcp.log | DHCP lease activity

FIELD	TYPE	DESCRIPTION
ts	time	Timestamp of the DHCP lease request
uid & id		Underlying connection info > See conn.log
mac	string	Client's hardware address
assigned_ip	addr	Client's actual assigned IP address
lease_time	interval	IP address lease time
trans_id	count	Identifier assigned by client; responses match

files.log | File analysis results

FIELD	TYPE	DESCRIPTION
ts	time	Timestamp when file was first seen
fuid	string	Unique identifier for a single file
tx_hosts	set	Host(s) that sourced the data
rx_hosts	set	Host(s) that received the data
conn_uids	set	Connection UID(s) over which file transferred
source	string	An identification of the source of the file data
depth	count	Depth of file related to source (e.g., HTTP request depth)
analyzers	set	Set of analyzers attached during file analysis
mime_type	string	File type, as determined by Bro's signatures
filename	string	Filename, if available from source analyzer
duration	interval	The duration that the file was analyzed for
local_orig	bool	Did the data originate locally?
is_orig	bool	Was the file sent by the Originator?
seen_bytes	count	Number of bytes provided to file analysis engine
total_bytes	count	Total number of bytes that should comprise the file
missing_bytes	count	Number of bytes in file stream missed
overflow_bytes	count	Out-of-sequence bytes in the stream due to overflow
timedout	bool	If the file analysis timed out at least once
parent_fuid	string	Container file ID this was extracted from
md5/sha1	string	MD5/SHA1 hash of the file
extracted	string	Local filename of extracted files, if enabled
entropy	double	Information density of the file contents

smtp.log | SMTP transactions

FIELD	TYPE	DESCRIPTION
ts	time	Timestamp when message wa
uid & id		Underlying connection info > S
trans_depth	count	Transaction depth if there are m
helo	string	Contents of the HELO header
mailfrom	string	Contents of the MAIL FROM he
rcptto	set	Contents of the RCPT TO heade
date	string	Contents of the DATE header
from	string	Contents of the FROM header
to	set	Contents of the TO header
cc	set	Contents of the CC header
reply_to	string	Contents of the ReplyTo heade
msg_id	string	Contents of the MsgID header
in_reply_to	string	Contents of the In-Reply-To hea
subject	string	Contents of the Subject header
x_originating_ip	addr	Contents of the X-Originating-I
first_received	string	Contents of the first Received h
second_received	string	Contents of the second Receive
last_reply	string	Last server to client message
path	vector	Message transmission path, fr
user_agent	string	Value of the client User-Agent
tls	bool	Indicates the connection switc
fuids	vector	File unique IDs seen attached t
is_webmail¹	bool	If the message was sent via we

¹if policy/protocols/smtp/software.bro is loaded

ts	Timestamps with microsecond accuracy, synchronized across logs
uid	Unique ID for every connection
md5/sha1	File hash of every file
fuid	Unique ID for every instance of every file seen on the network

Advantages of Zeek

- Network traffic analysis
- Protocol analysis
- Threat detection
- Forensic analysis
- Integration with other security tools
- Customization and extensibility

References

- https://www.cloudflare.com/learning/dns/dns-server-types/
- https://www.iana.org/domains/root/servers
- https://docs.zeek.org/en/master/
- https://www.fortinet.com/resources/cyberglossary/intrusion-detection-system
-