# Network Security Protocols and Defensive Mechanisms

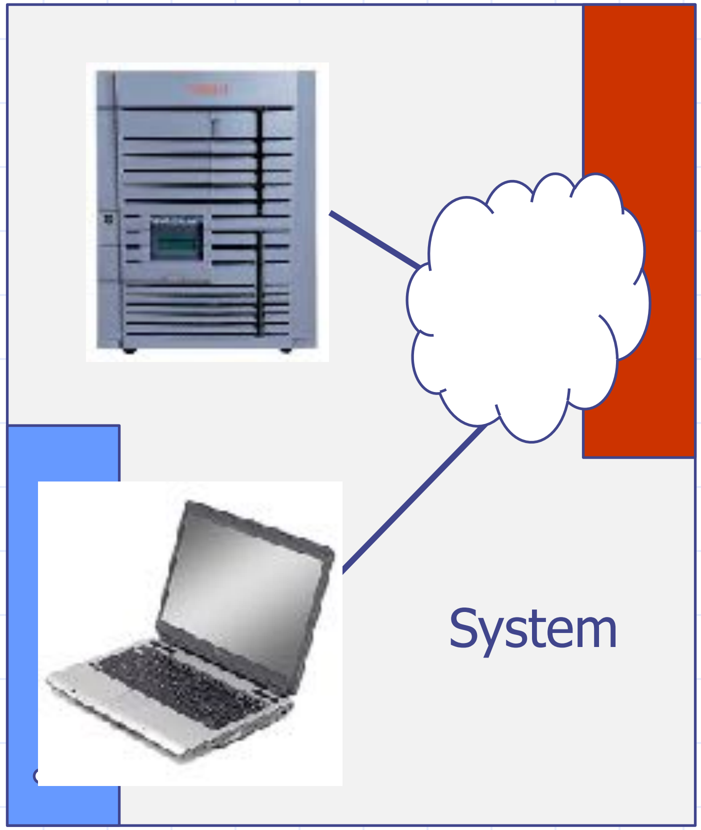
John Mitchell

*Acknowledgments: Lecture slides are from the Computer Security course taught by Dan Boneh and John Mitchell at Stanford University. When slides are obtained from other sources, a a reference will be noted on the bottom of that slide. A full list of references is provided on the last slide.*

# Network security

- What is the network for?
- What properties might attackers destroy?
  - Confidentiality : no information revealed to others
  - Integrity : communication remains intact
  - Availability : messages received in reasonable time

- Confidentiality
- Integrity
- Availability




Network Attacker

Intercepts and controls network communication

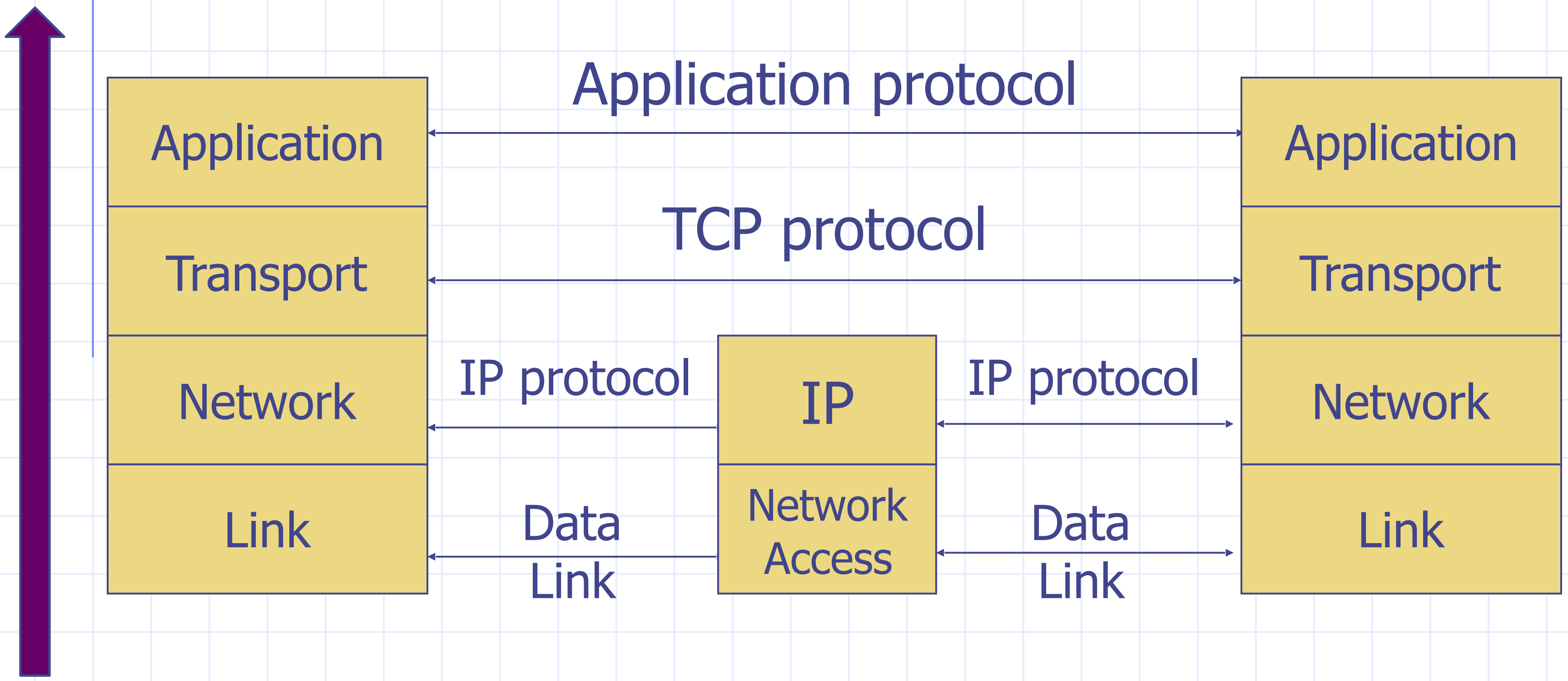
# Plan for today


- Protecting network connections
  - Wireless access– 802.11i/WPA2
  - IPSEC
- Perimeter network defenses
  - Firewall
    - Packet filter (stateless, stateful), Application layer proxies
  - Intrusion detection
    - Anomaly and misuse detection
- Network infrastructure security
  - BGP instability and S-BGP
  - DNS rebinding and DNSSEC

# Last lecture

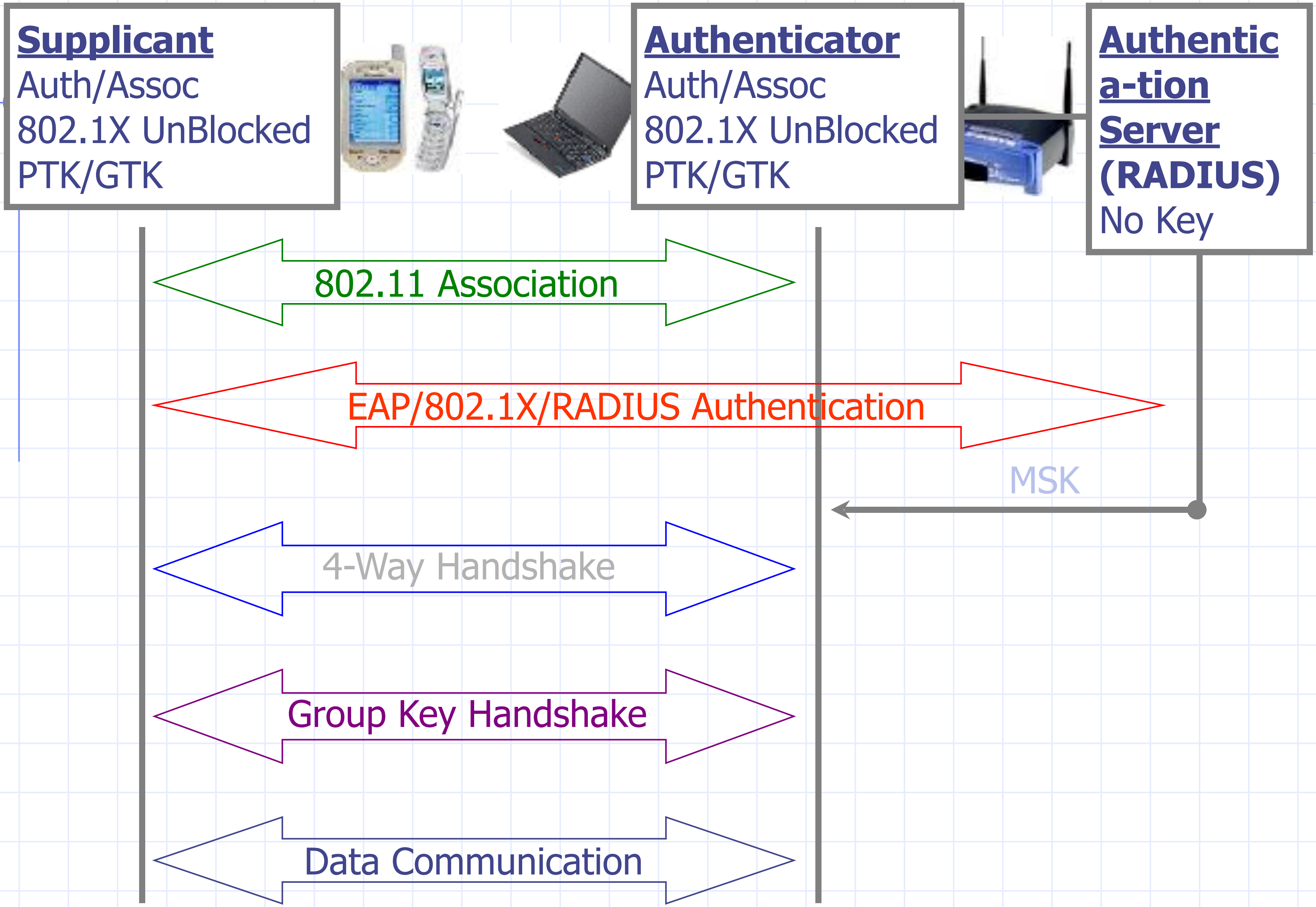
- Basic network protocols
  - IP, TCP, UDP, BGP, DNS
- Problems with them
  - TCP/IP
    - No SRC authentication: can't tell where packet is from
    - Packet sniffing
    - Connection spoofing, sequence numbers
  - BGP: advertise bad routes or close good ones
  - DNS: cache poisoning, rebinding
    - Web security mechanisms rely on DNS

# Network Protocol Stack



# Link-layer connectivity

# 802.11i Protocol



# TCP/IP CONNECTIVITY

How can we isolate our conversation from attackers on the Internet?

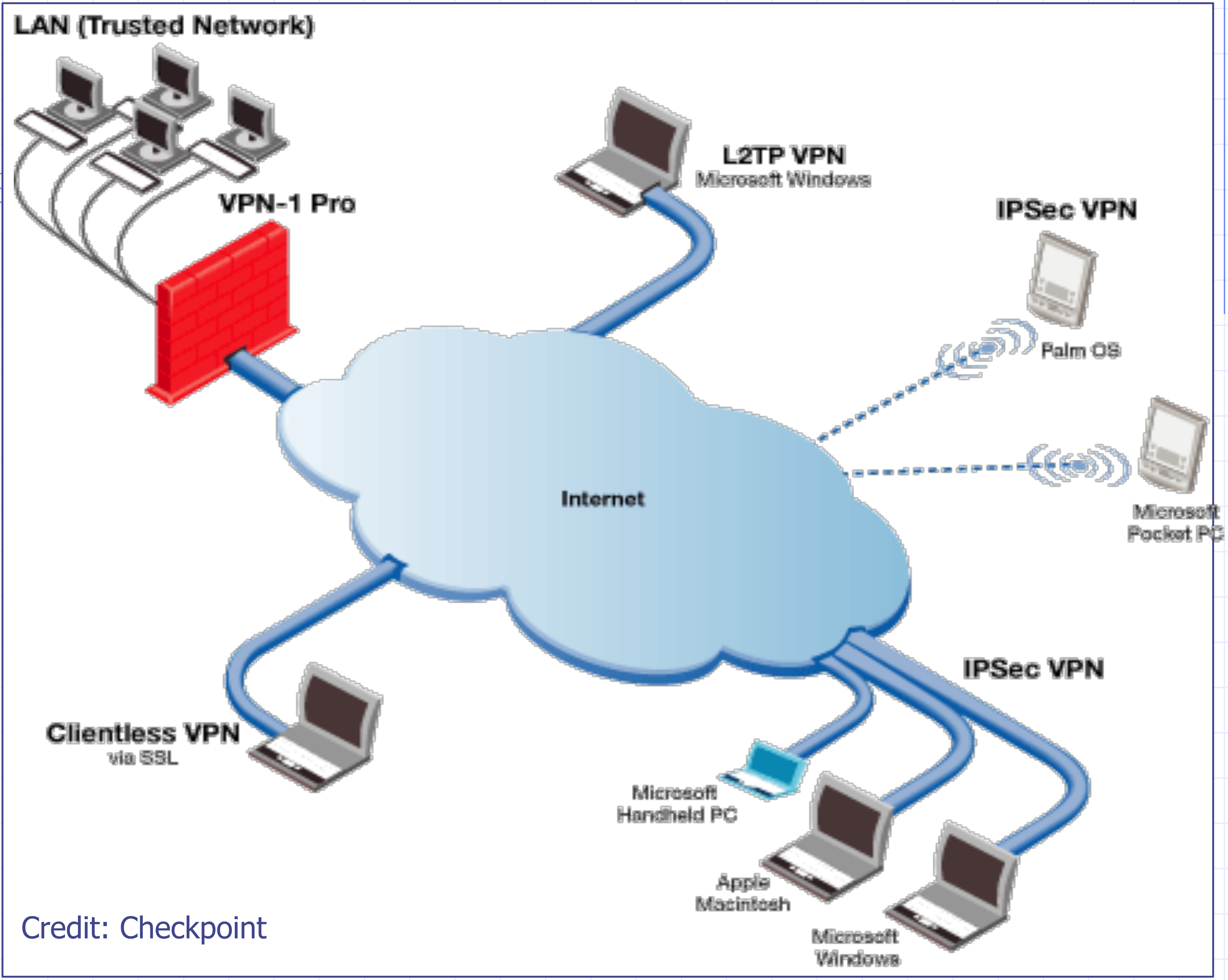
# Basic Layer 2-3 Security Problems

- Network packets pass by untrusted hosts
  - Eavesdropping, packet sniffing
  - Especially easy when attacker controls a machine close to victim
- TCP state can be easy to guess
  - Enables spoofing and session hijacking

# Virtual Private Network (VPN)

- Three different modes of use:
  - Remote access client connections
  - LAN-to-LAN internetworking
  - Controlled access within an intranet
- Several different protocols
  - PPTP – Point-to-point tunneling protocol
  - L2TP – Layer-2 tunneling protocol
  - IPsec (Layer-3: network layer)

(PPTP and L2TP: Data layer)

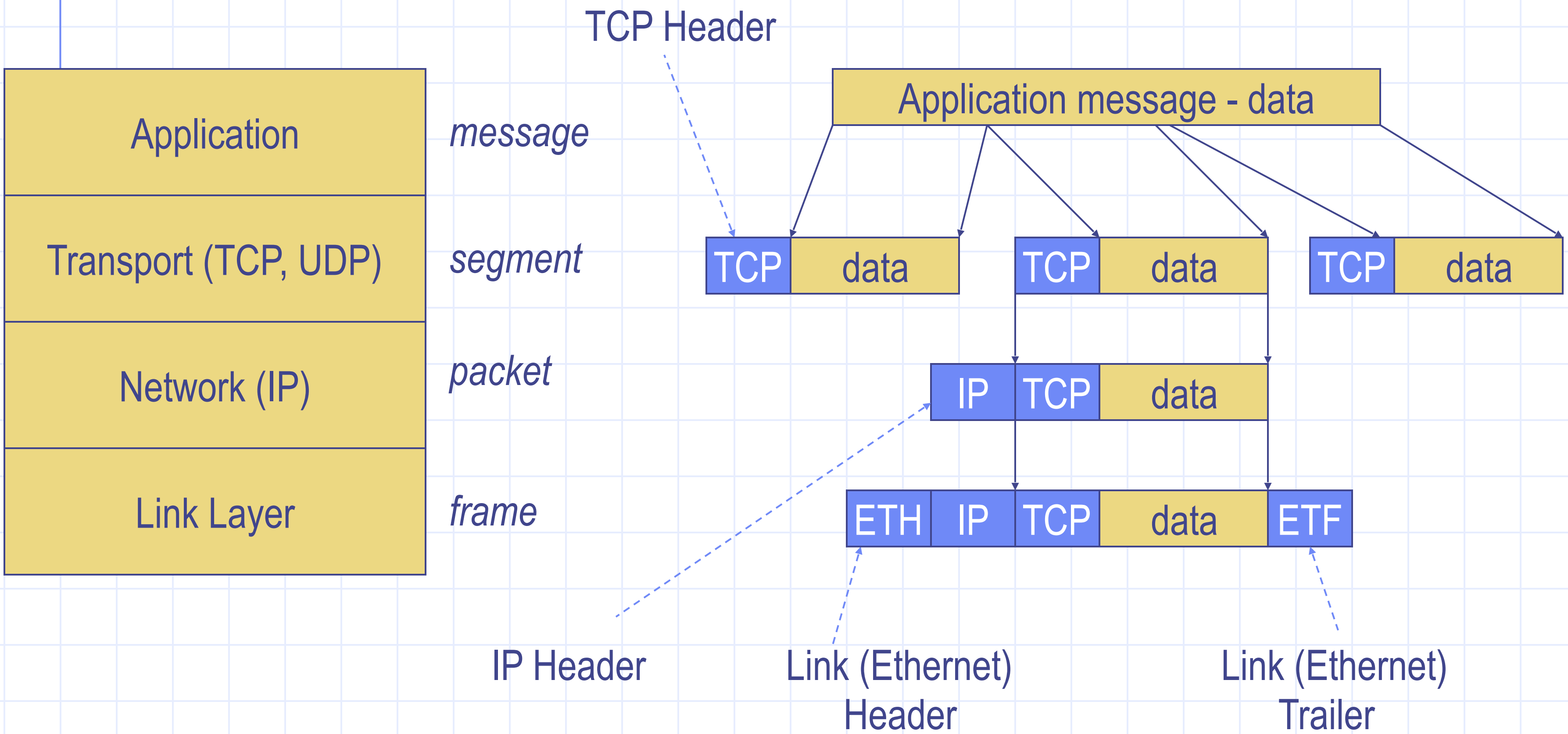



Credit: Checkpoint

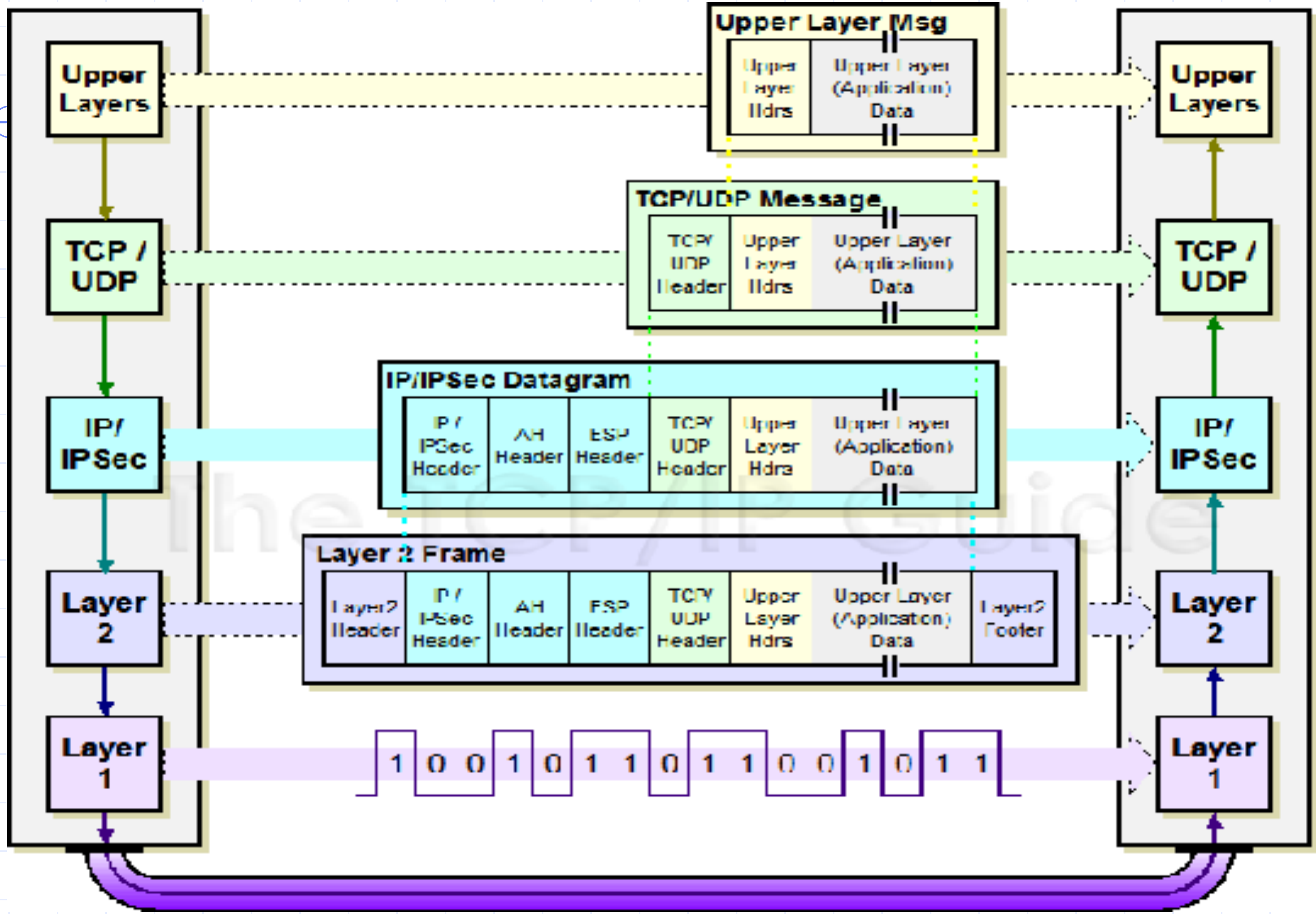
# IPSEC

- Security extensions for IPv4 and IPv6
- IP Authentication Header (AH)
  - Authentication and integrity of payload and header
- IP Encapsulating Security Protocol (ESP)
  - Confidentiality of payload
- ESP with optional ICV (integrity check value)
  - Confidentiality, authentication and integrity of payload

# Recall packet formats and layers



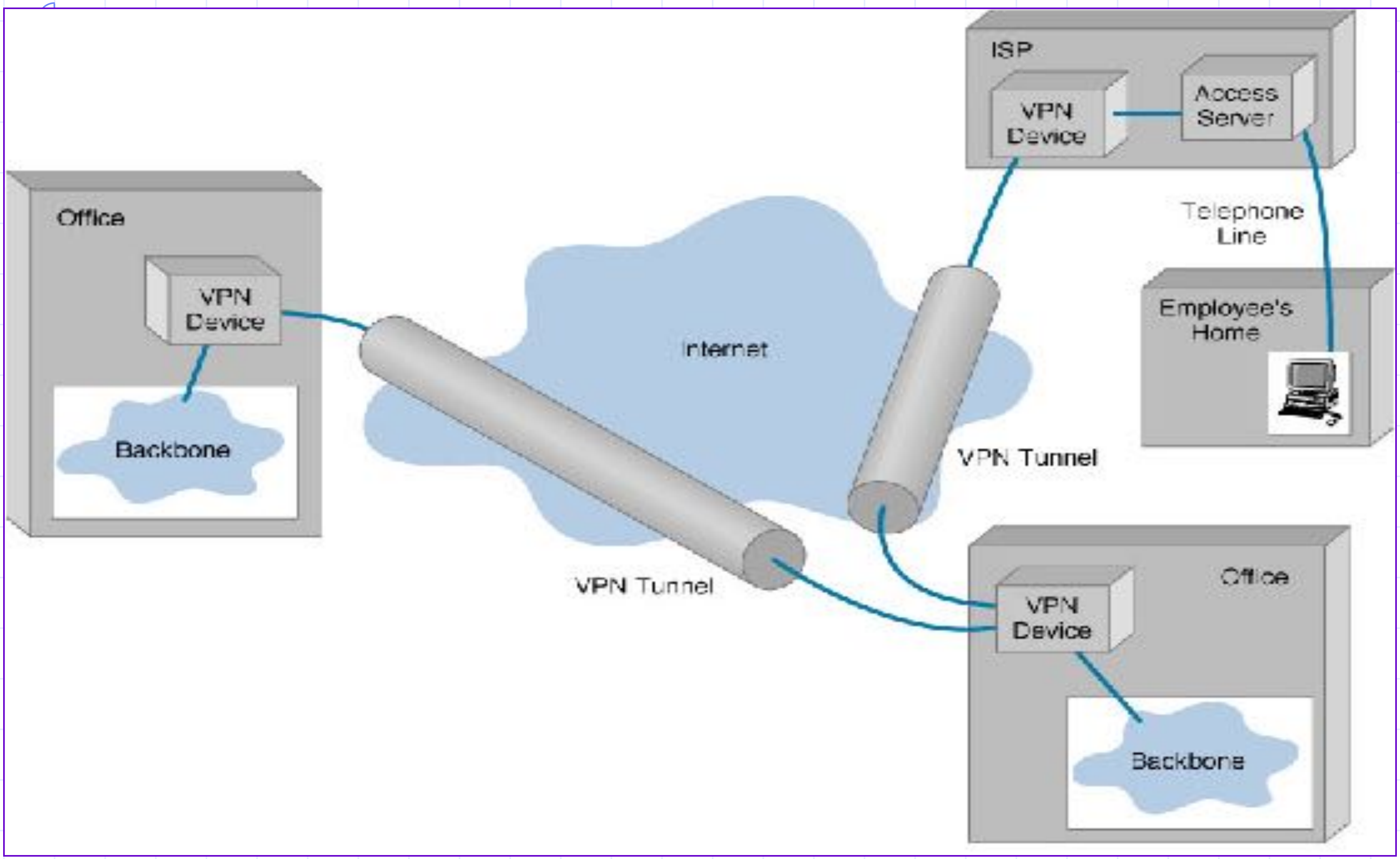
# IPSec Transport Mode: IPSEC instead of IP header



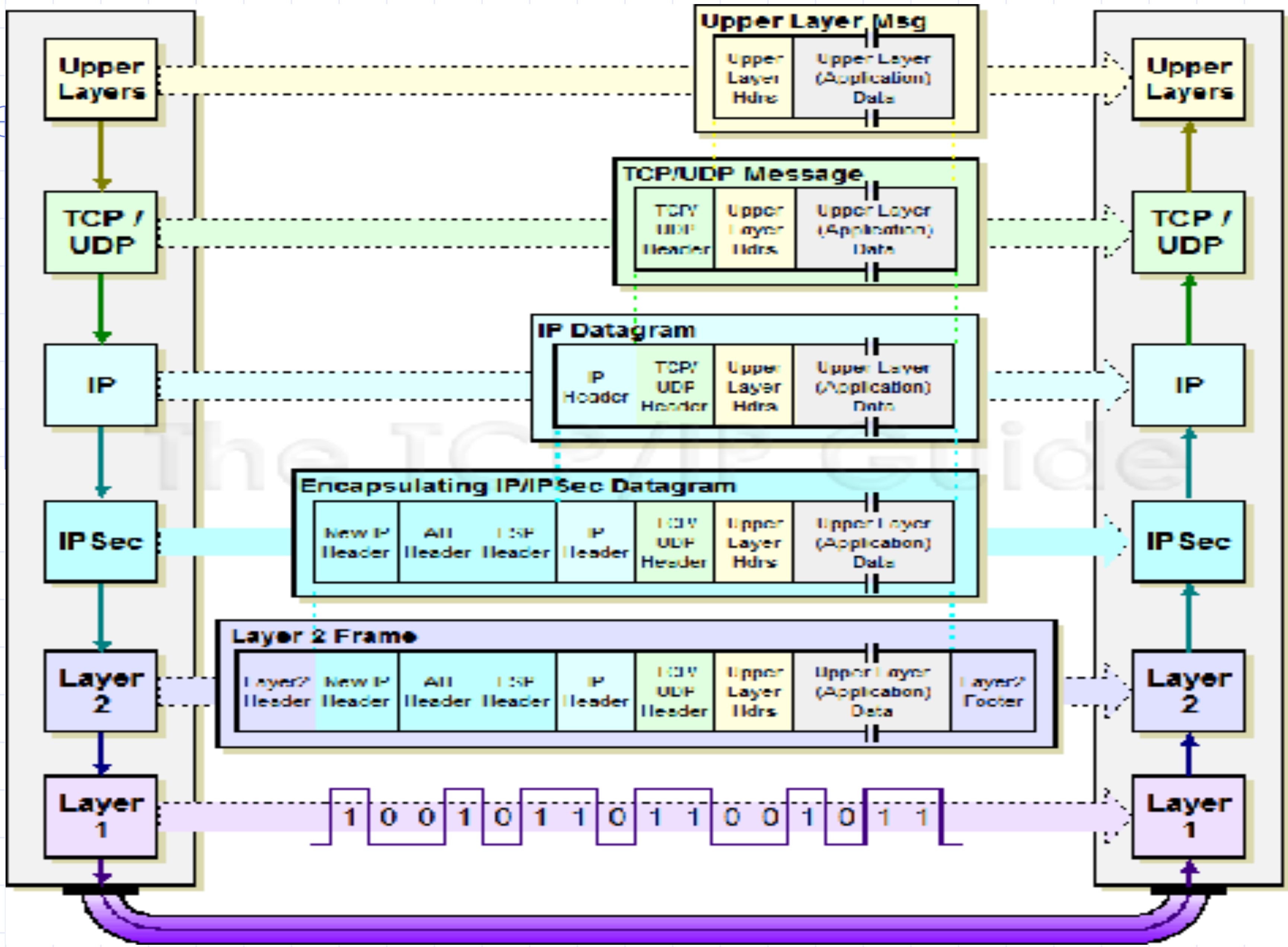



http://www.tcpipguide.com/free/t_IPSecModesTransportandTunnel.htm

# IPSEC Tunnel Mode



# IPSec Tunnel Mode: IPSEC header + IP header



# Summary of first section


- Protecting network connections
  - Wireless access– 802.11i/WPA2
    - Several subprotocols provide encrypted link between user device and wireless access point
  - IPSEC
    - Give external Internet connections equivalent security to local area network connections

# Second topic of today's lecture

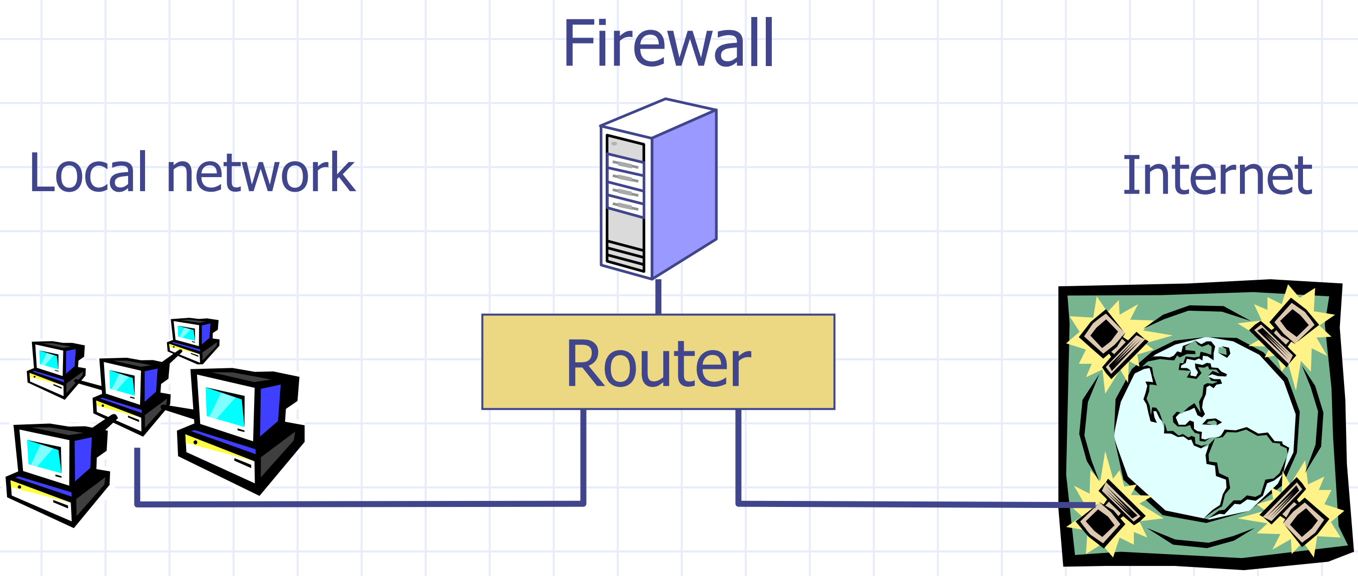
- Perimeter defenses for local networks
  - Firewall
    - Packet filter (stateless, stateful)
    - Application layer proxies
  - Intrusion detection
    - Anomaly and misuse detection

# LOCAL AREA NETWORK

How can we protect our local area network from attackers on the external Internet?

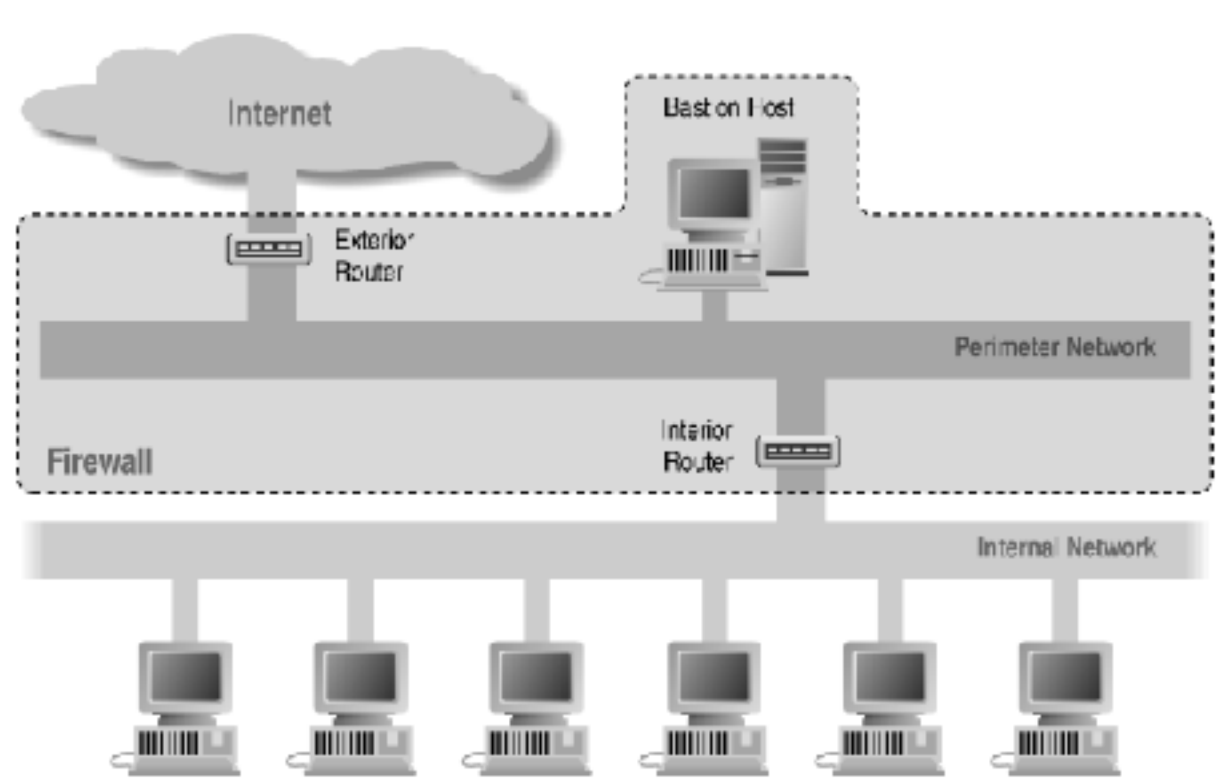
# Basic Firewall Concept

- Separate local area net from internet

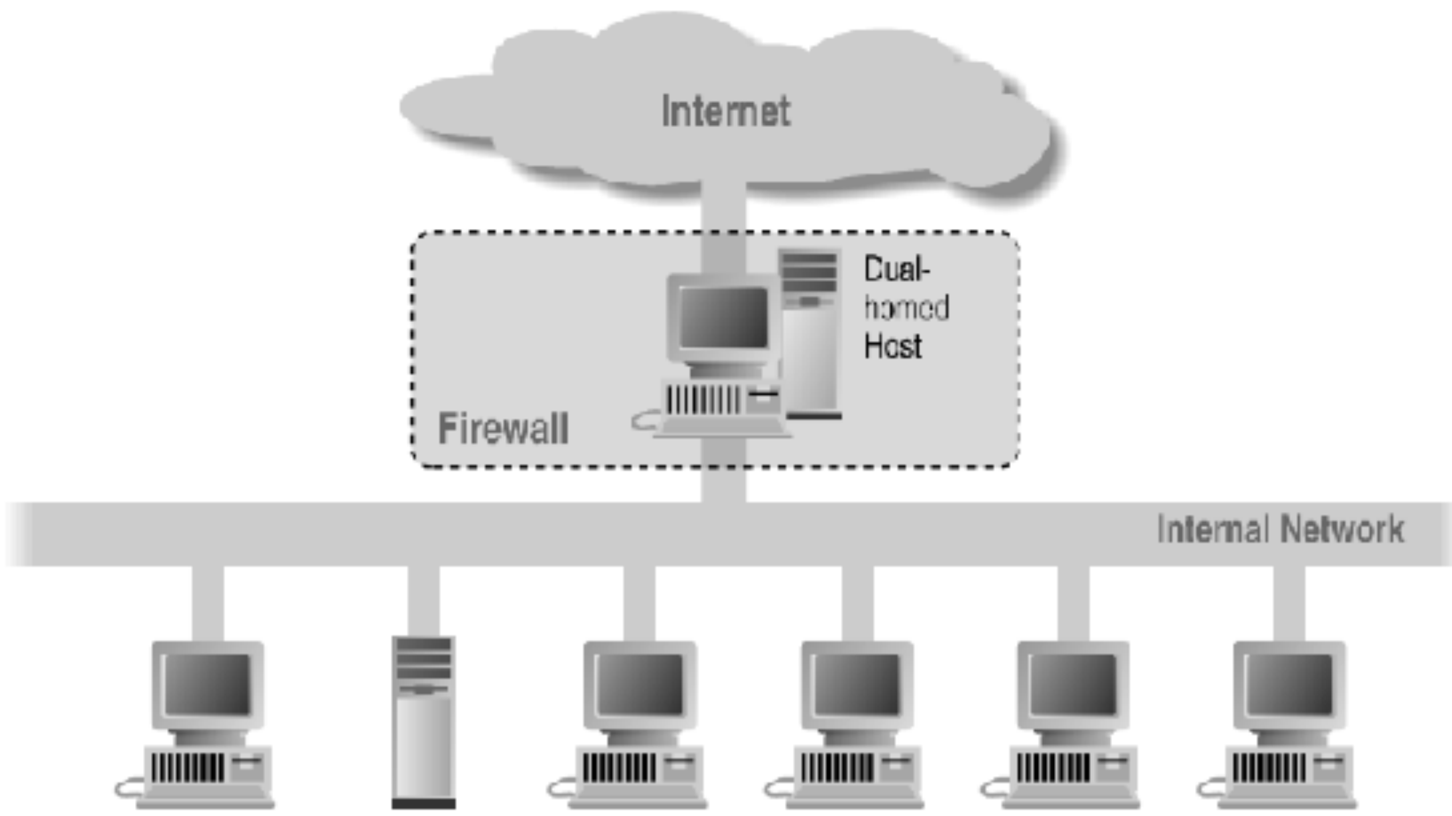



All packets between LAN and internet routed through firewall

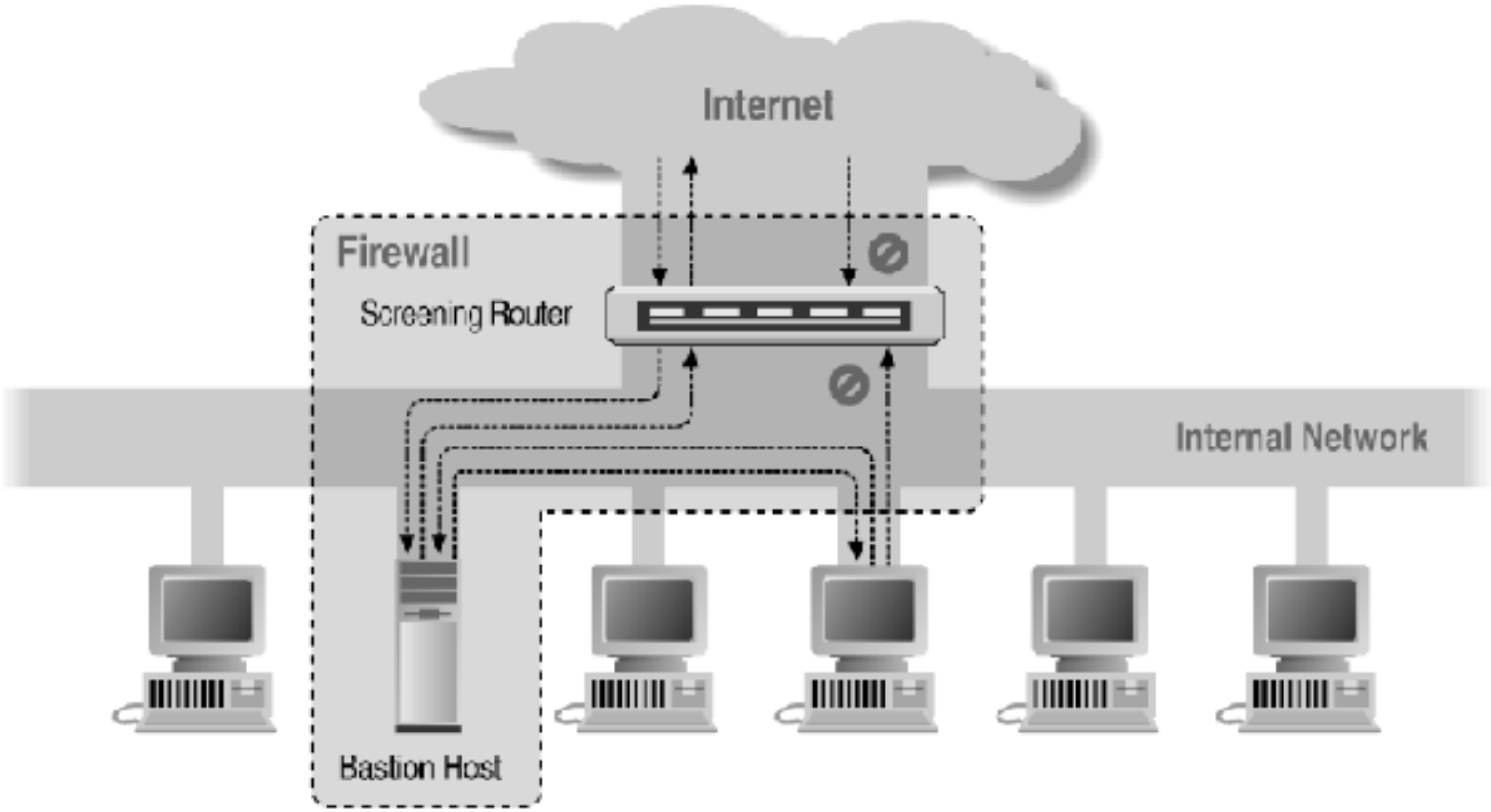
# Screened Subnet Using Two Routers



# Alternate 1: Dual-Homed Host



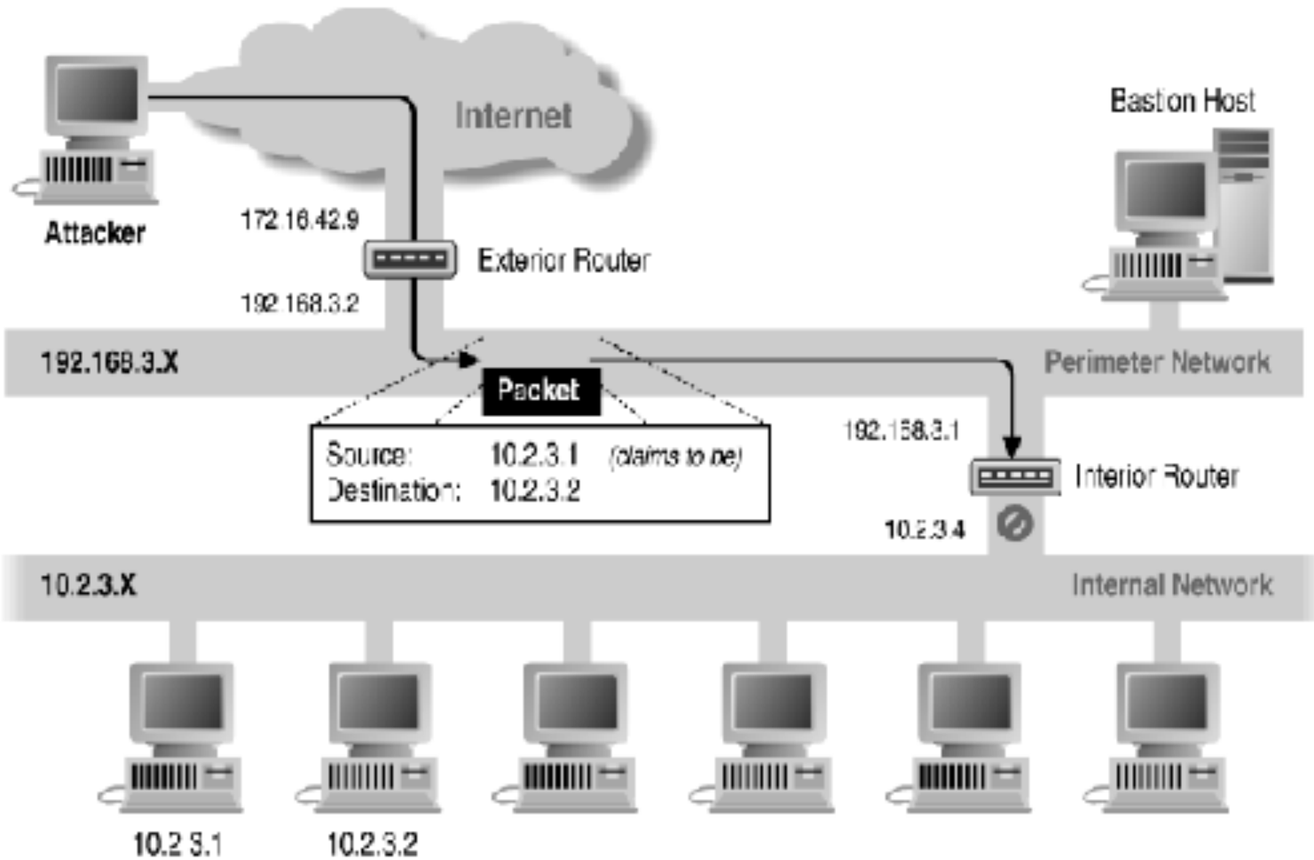
# Alternate 2: Screened Host



# Basic Packet Filtering

- Uses transport-layer information only
  - IP Source Address, Destination Address
  - Protocol (TCP, UDP, ICMP, etc)
  - TCP or UDP source & destination ports
  - TCP Flags (SYN, ACK, FIN, RST, PSH, etc)
  - ICMP message type
- Examples
  - DNS uses port 53
    - Block incoming port 53 packets except known trusted servers
- Issues
  - Stateful filtering
  - Encapsulation: address translation, other complications
  - Fragmentation

# Source-Address Forgery

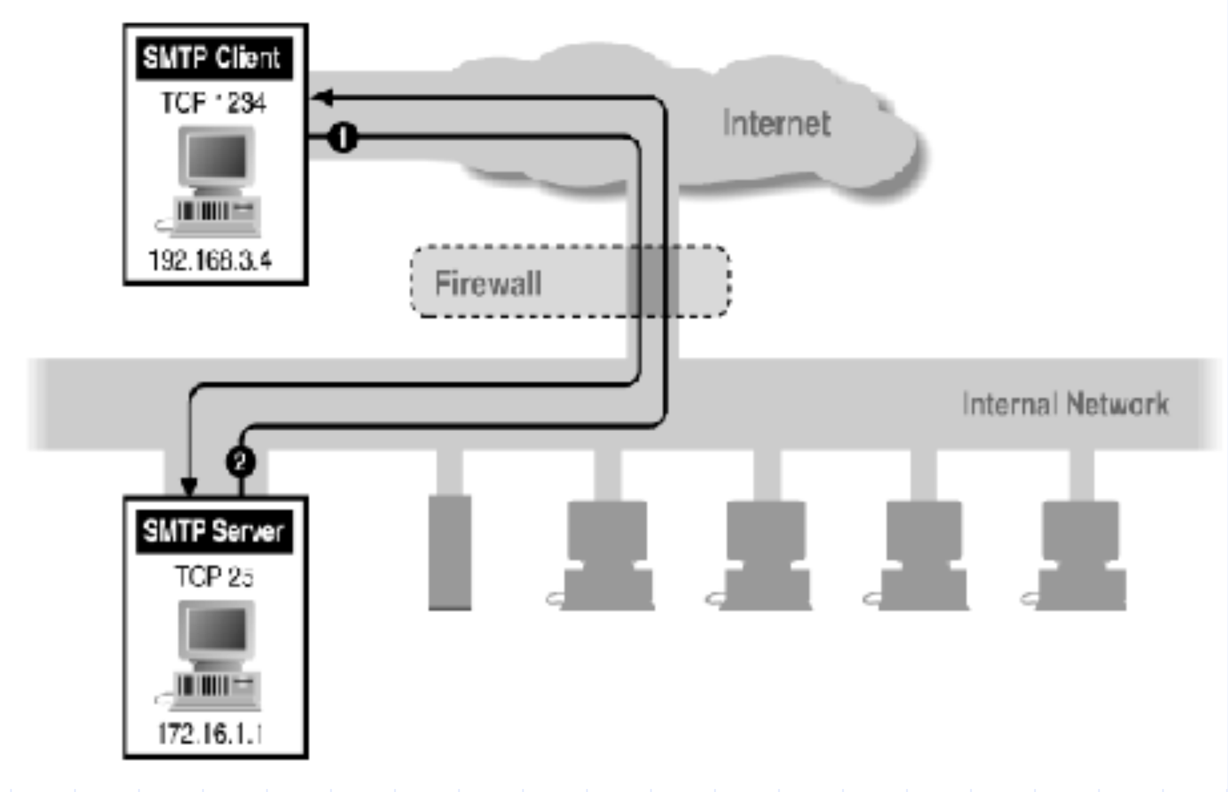


# More about networking: port numbering

- TCP connection
  - Server port uses number less than 1024
  - Client port uses number between 1024 and 16383
- Permanent assignment
  - Ports <1024 assigned permanently
    - 20,21 for FTP 23 for Telnet
    - 25 for server SMTP 80 for HTTP
- Variable use
  - Ports >1024 must be available for client to make connection
  - Limitation for stateless packet filtering
    - If client wants port 2048, firewall must allow incoming traffic
  - Better: stateful filtering knows outgoing requests
    - Only allow incoming traffic on high port to a machine that has initiated an outgoing request on low port

# Filtering Example: Inbound SMTP

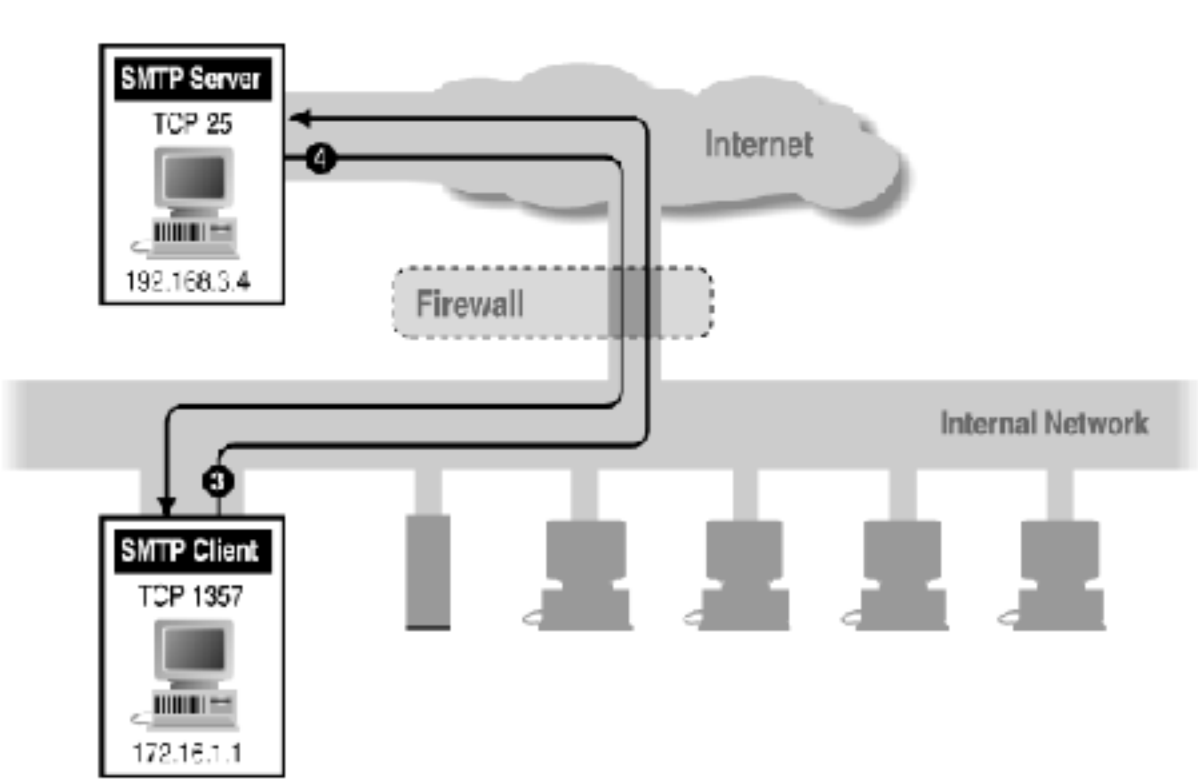
Assume we want to block internal server from external attack




Can block external request to internal server based on port number

# Filtering Example: Outbound SMTP

Assume we want to allow internal access to external server

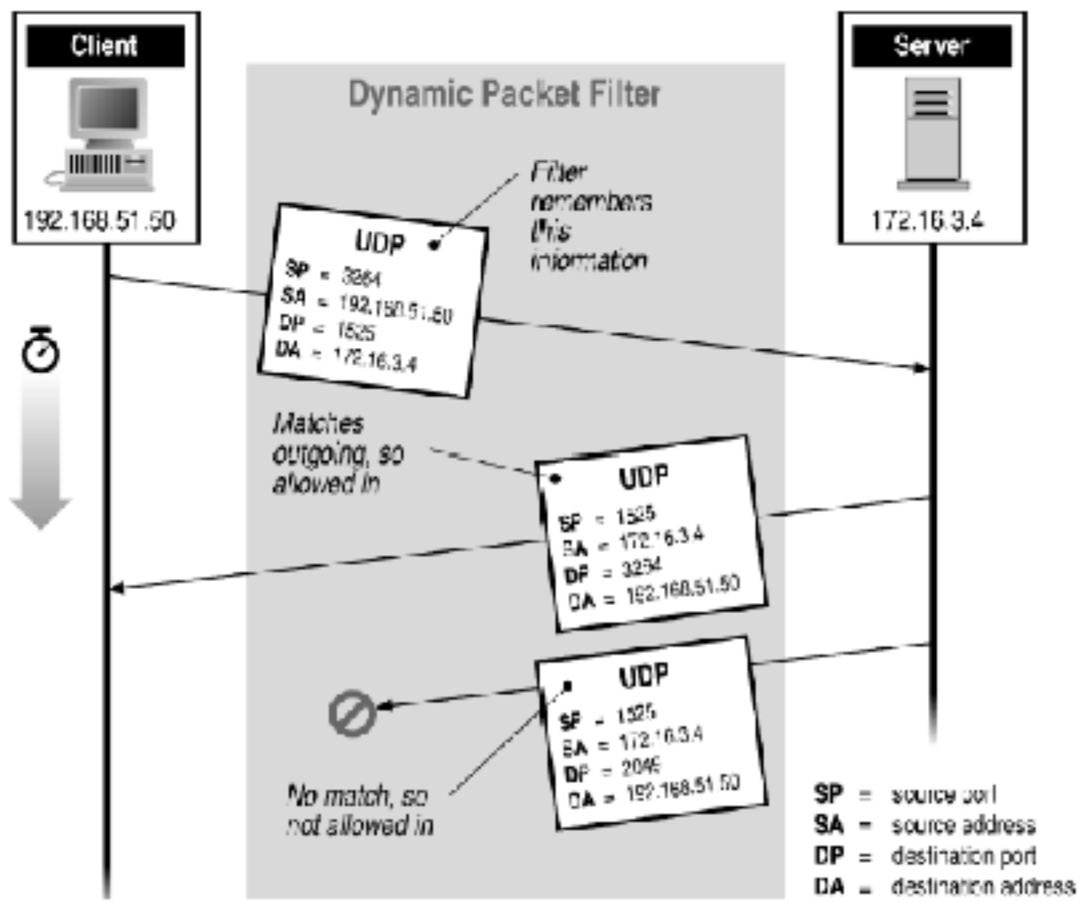



Known low port out, arbitrary high port in

If firewall blocks incoming port 1357 traffic then connection fails

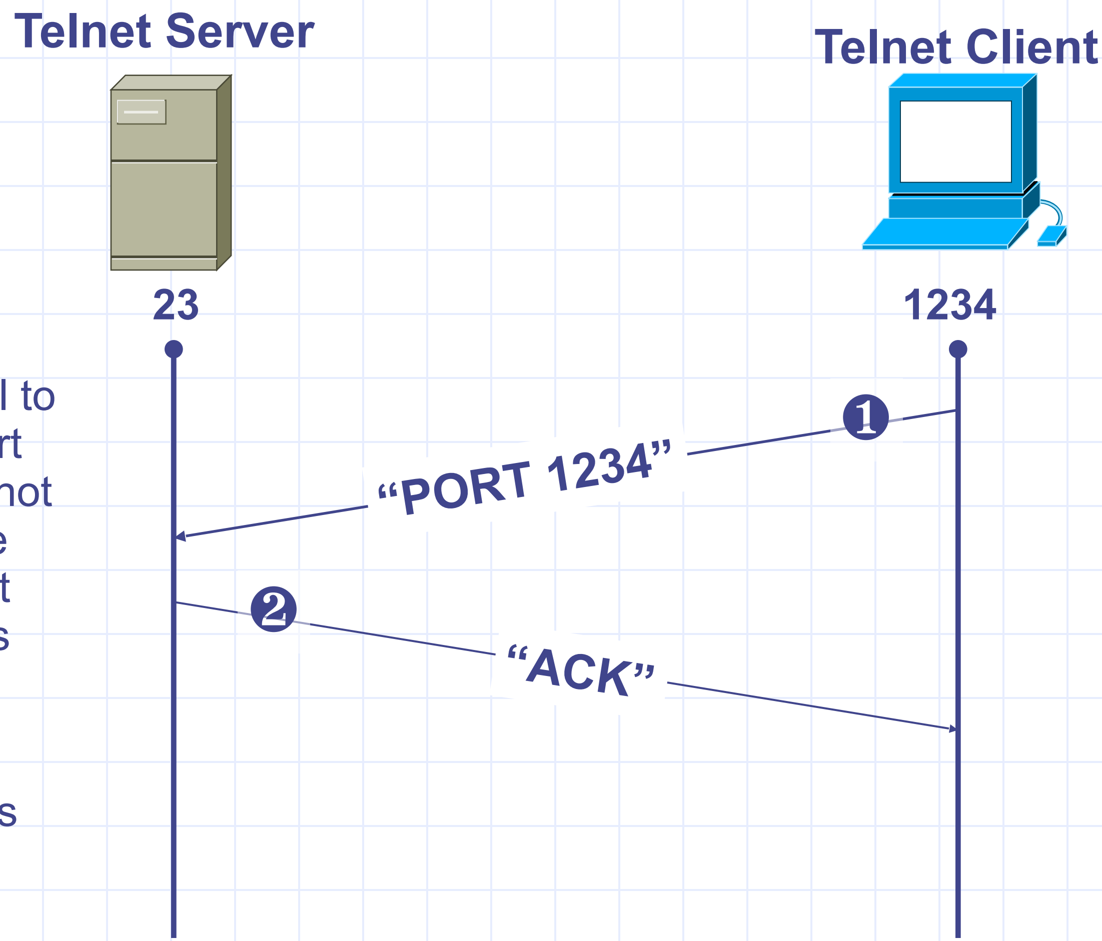
# Stateful or Dynamic Packet Filtering

Assume we want to allow external UDP only if requested



# Telnet

How can stateful filtering identify legitimate session?



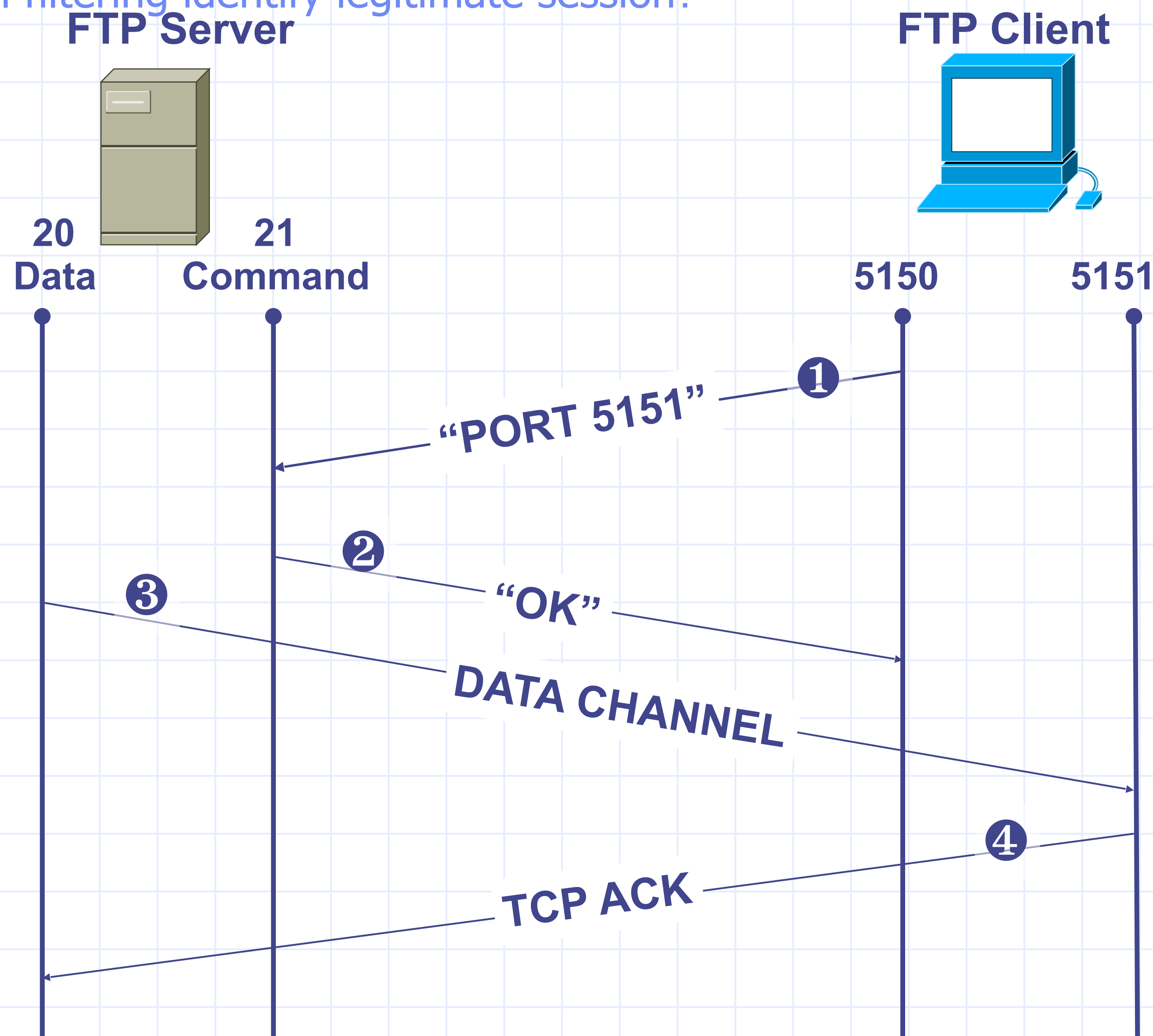

❶ Client opens channel to server; tells server its port number. The ACK bit is not set while establishing the connection but will be set on the remaining packets

❷ Server acknowledges

Stateful filtering can use this pattern to identify legitimate sessions

# FTP

How can stateful filtering identify legitimate session?




❶ Client opens command channel to server; tells server second port number

❷ Server acknowledges

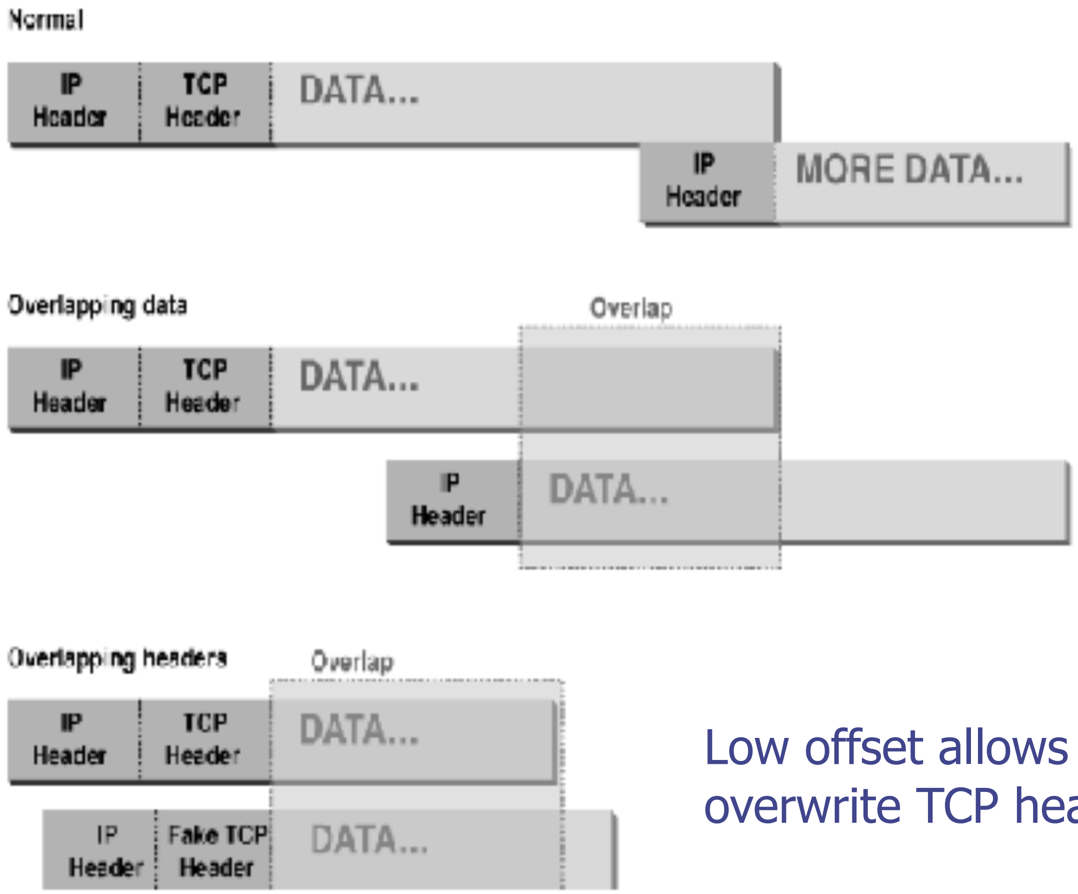
❸ Server opens data channel to client's second port

❹ Client acknowledges

# Normal IP Fragmentation


Flags and offset inside IP header indicate packet fragmentation

# Abnormal Fragmentation

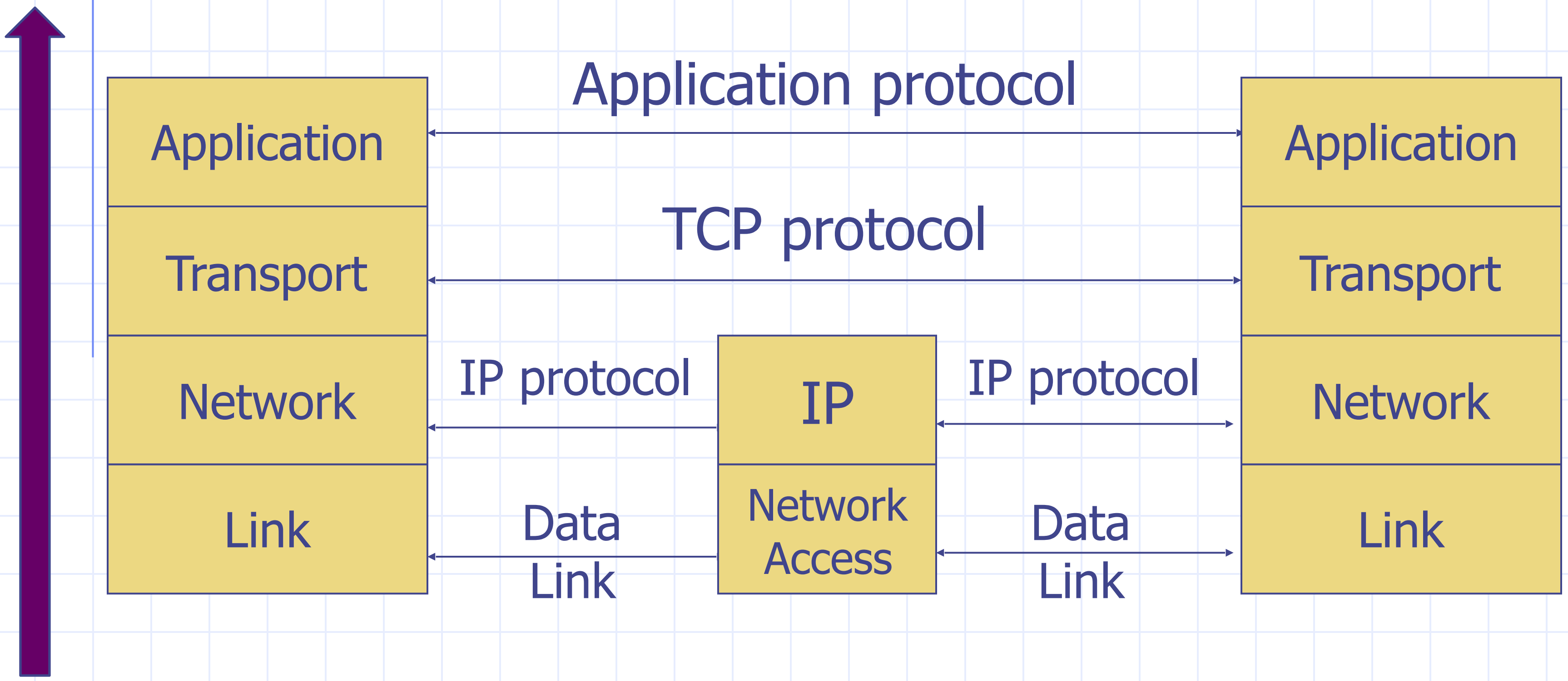



Low offset allows second packet to overwrite TCP header at receiving host

# Packet Fragmentation Attack

- Firewall configuration
  - TCP port 23 is blocked but SMTP port 25 is allowed
- First packet
  - Fragmentation Offset = 0.
  - DF bit = 0 : "May Fragment"
  - MF bit = 1 : "More Fragments"
  - Destination Port = 25. TCP port 25 is allowed, so firewall allows packet
- Second packet
  - Fragmentation Offset = 1: second packet overwrites all but first 8 bits of the first packet
  - DF bit = 0 : "May Fragment"
  - MF bit = 0 : "Last Fragment."
  - Destination Port = 23. Normally be blocked, but sneaks by!
- What happens
  - Firewall ignores second packet "TCP header" because it is fragment of first
  - At host, packet reassembled and received at port 23

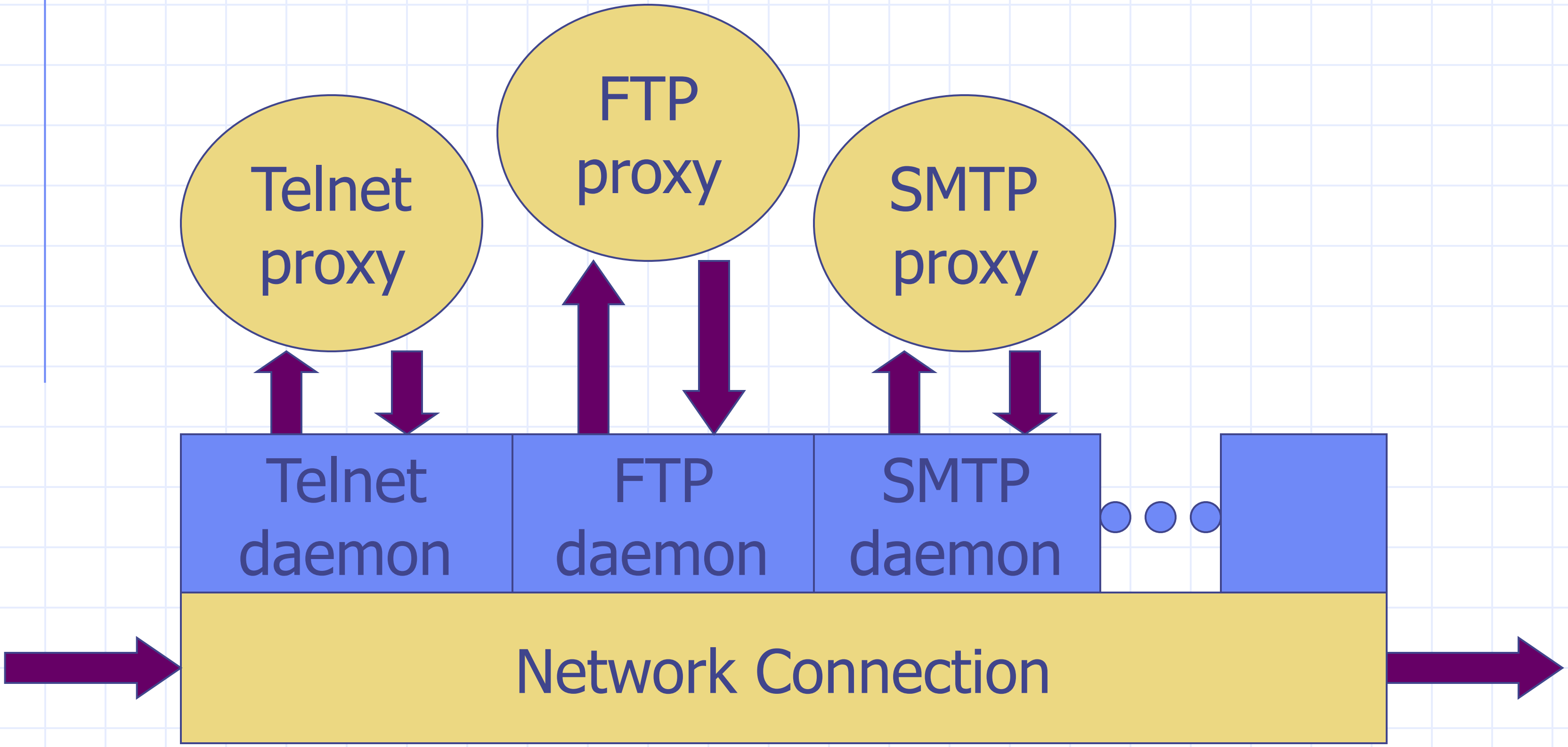
# TCP Protocol Stack



# Proxying Firewall

- Application-level proxies
  - Tailored to http, ftp, smtp, etc.
  - Some protocols easier to proxy than others
- Policy embedded in proxy programs
  - Proxies filter incoming, outgoing packets
  - Reconstruct application-layer messages
  - Can filter specific application-layer commands, etc.
    - Example: only allow specific ftp commands
    - Other examples: ?
- Several network locations – see next slides

# Firewall with application proxies




Daemon spawns proxy when communication detected …

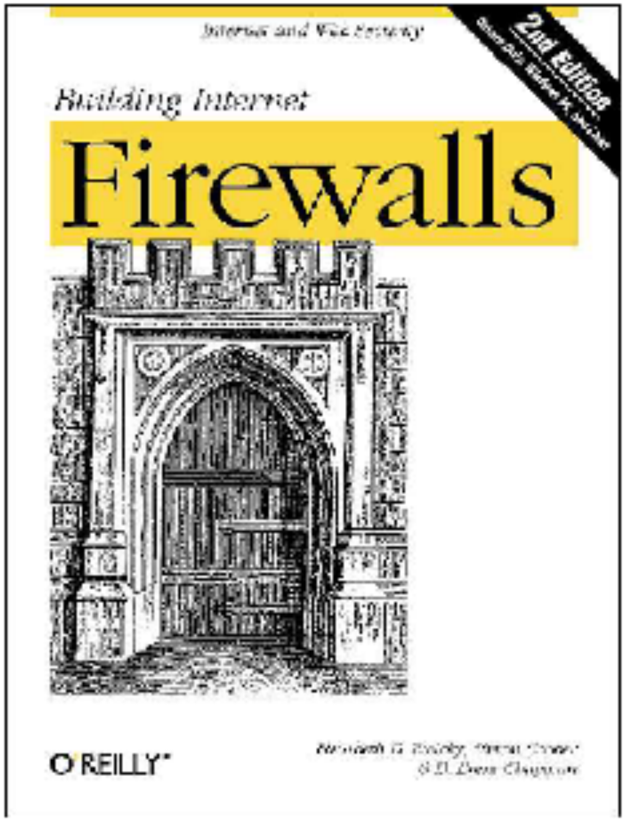
# Application-level proxies

- Enforce policy for specific protocols
  - E.g., Virus scanning for SMTP
    - Need to understand MIME, encoding, Zip archives
  - Flexible approach, but may introduce network delays
- "Batch" protocols are natural to proxy
  - SMTP (E-Mail) NNTP (Net news)
  - DNS (Domain Name System) NTP (Network Time Protocol)
- Must protect host running protocol stack
  - Disable all non-required services; keep it simple
  - Install/modify services you want
  - Run security audit to establish baseline
  - Be prepared for the system to be compromised

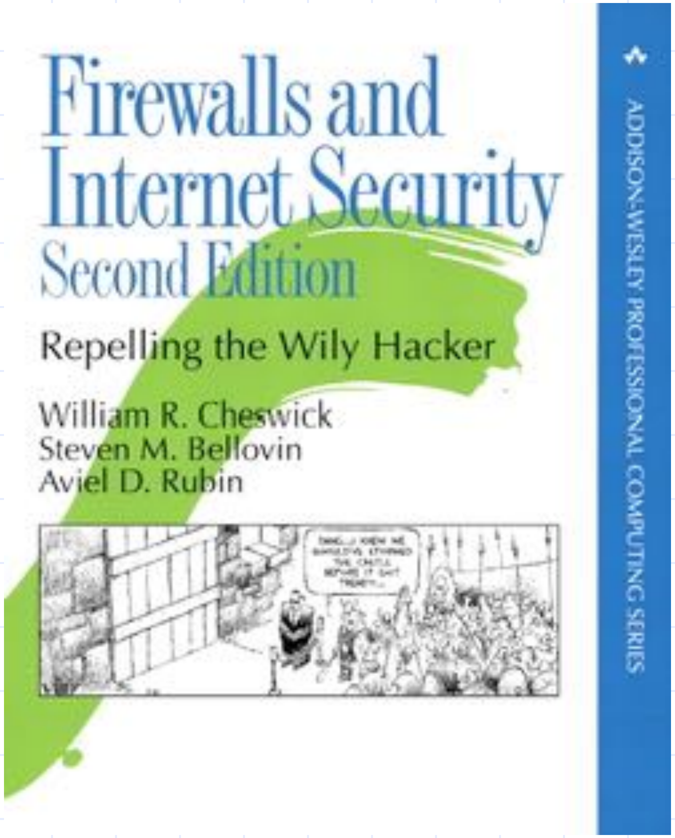
# Web traffic scanning

- Intercept and proxy web traffic
  - Can be host-based
  - Usually at enterprise gateway
- Block known bad sites
- Block pages with known attacks
- Scan attachments
  - Virus, worm, malware, …

# Firewall references




Elizabeth D. Zwicky
Simon Cooper
D. Brent Chapman



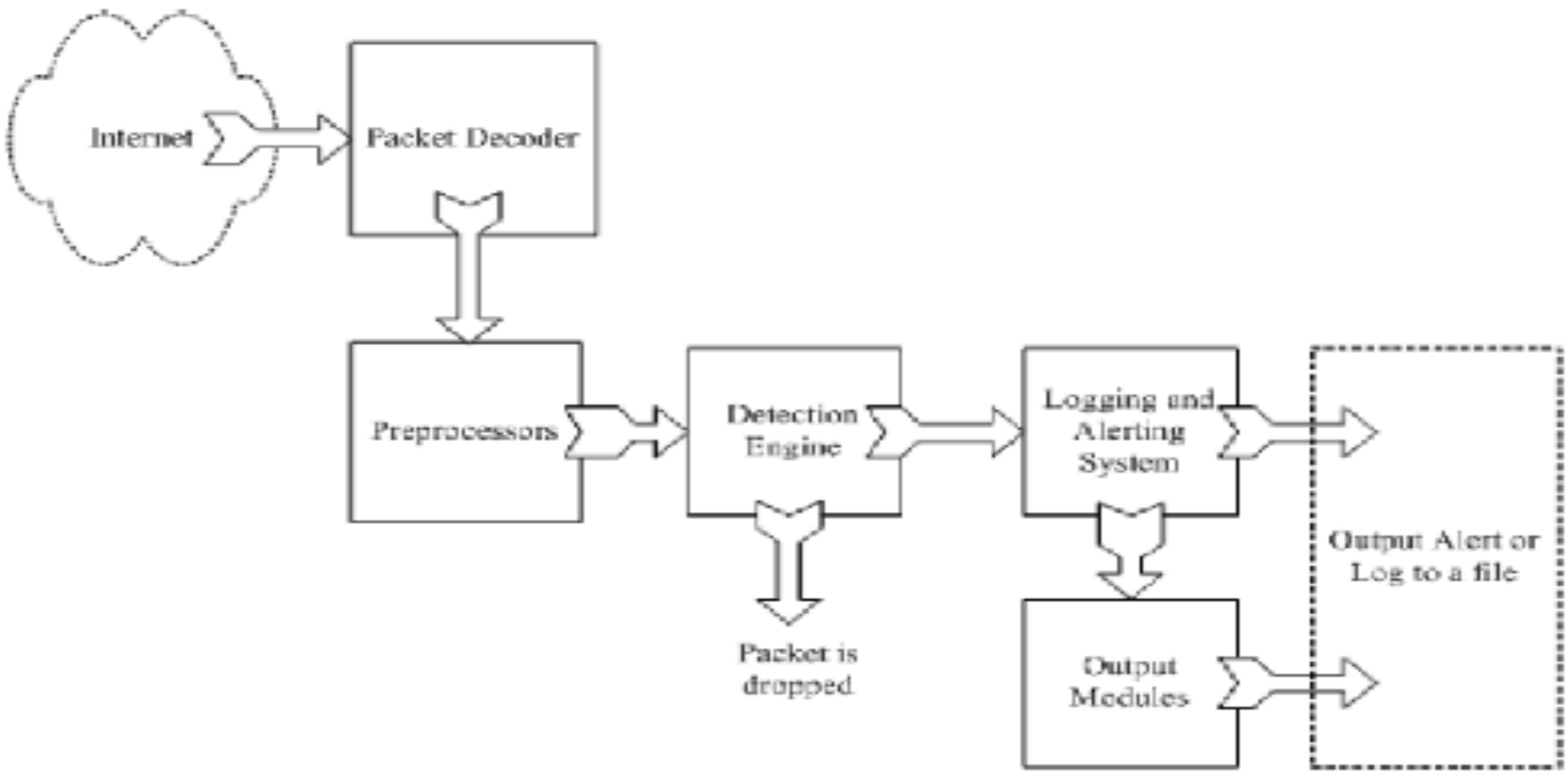

William R Cheswick
Steven M Bellovin
Aviel D Rubin

# Intrusion detection

- Many intrusion detection systems
  - Network-based, host-based, or combination
- Two basic models
  - Misuse detection model
    - Maintain data on known attacks
    - Look for activity with corresponding signatures
  - Anomaly detection model
    - Try to figure out what is "normal"
    - Report anomalous behavior
- Fundamental problem: too many false alarms

# Example: Snort

http://www.snort.org/

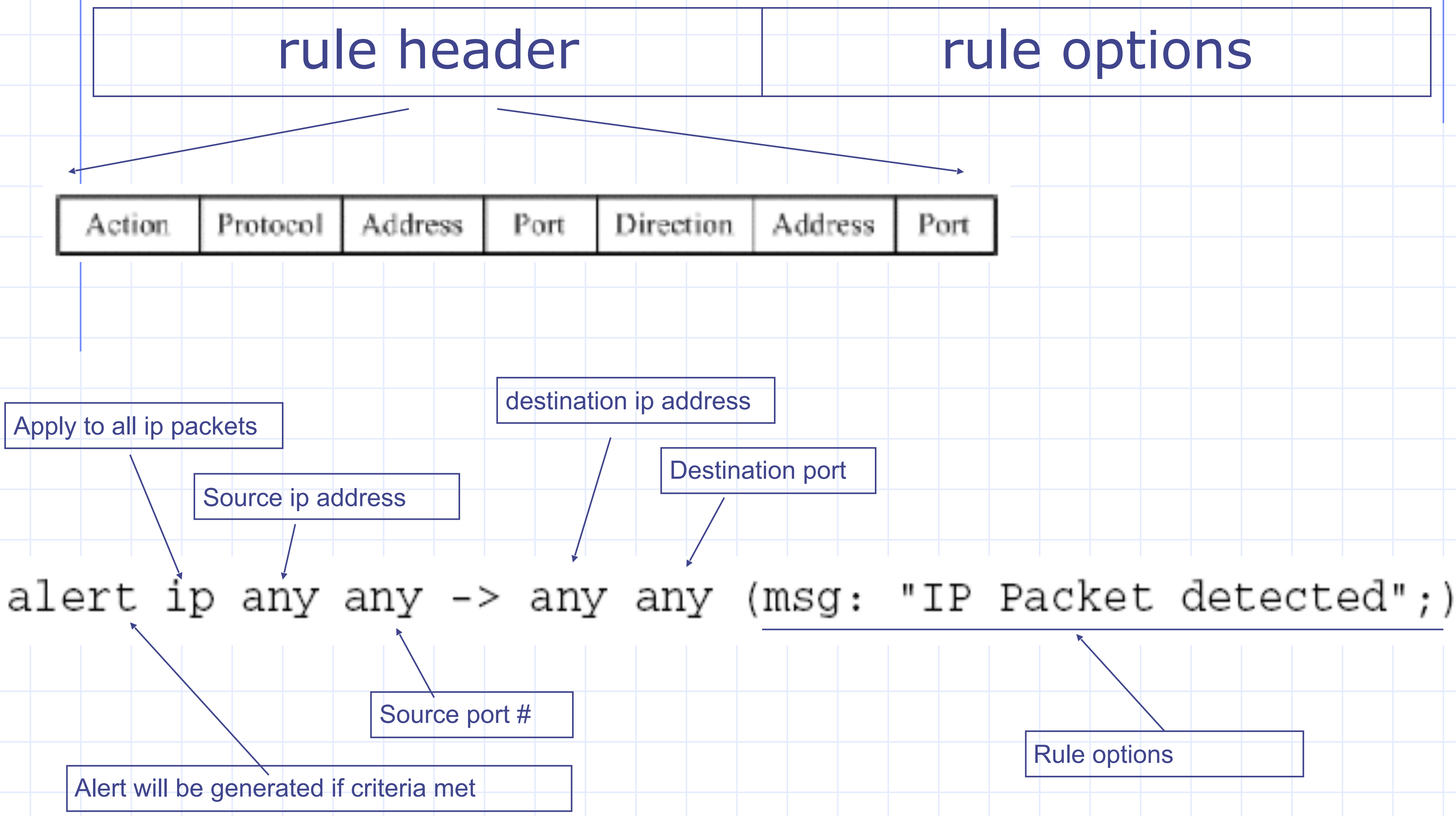



From: Rafeeq Ur Rehman, *Intrusion Detection Systems with Snort: Advanced IDS Techniques with Snort, Apache, MySQL, PHP, and ACID.*

# Snort components

- Packet Decoder
  - input from Ethernet, SLIP, PPP…
- Preprocessor:
  - detect anomalies in packet headers
  - packet defragmentation
  - decode HTTP URI
  - reassemble TCP streams
- Detection Engine: applies rules to packets
- Logging and Alerting System
- Output Modules: alerts, log, other output

# Snort detection rules



# Additional examples

alert tcp any any -> 192.168.1.0/24 111
(content:"|00 01 86 a5|"; msg: "mountd access";)

alert tcp !192.168.1.0/24 any -> 192.168.1.0/24 111
(content: "|00 01 86 a5|"; msg: "external mountd access";)

! = negation operator in address

content - match content in packet

192.168.1.0/24 - addr from 192.168.1.1 to 192.168.1.255

https://www.snort.org/documents/snort-users-manual

# Snort challenges

- Misuse detection – avoid known intrusions
  - Database size continues to grow
    - Snort version 2.3.2 had 2,600 rules
  - Snort spends 80% of time doing string match
- Anomaly detection – identify new attacks
  - Probability of detection is low

# Difficulties in anomaly detection

- Lack of training data
  - Lots of "normal" network, system call data
  - Little data containing realistic attacks, anomalies
- Data drift
  - Statistical methods detect changes in behavior
  - Attacker can attack gradually and incrementally
- Main characteristics not well understood
  - By many measures, attack may be within bounds of "normal" range of activities
- False identifications are very costly
  - Sys Admin spend many hours examining evidence

# Summary of this section

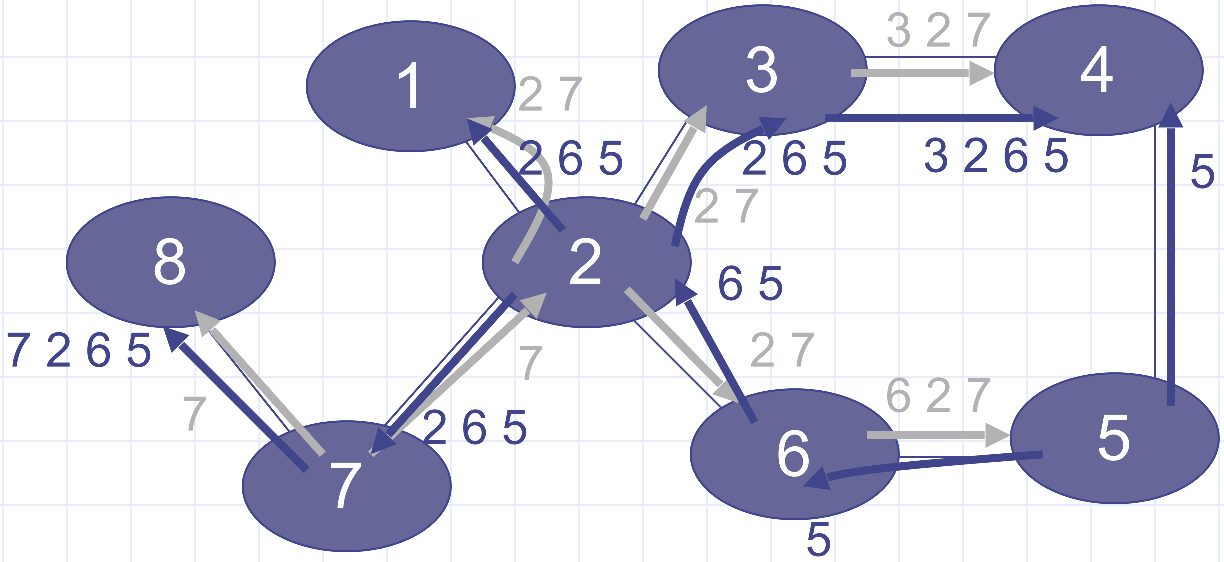
- Perimeter defenses for local networks
  - Firewall
    - Packet filter (stateless, stateful), Application layer proxies
  - Intrusion detection
    - Anomaly and misuse detection

# Last section of today's lecture

- Network infrastructure protocols
  - BGP vulnerabilities and S-BGP
  - DNS security, cache poisoning and rebinding attacks

# INFRASTRUCTURE PROTOCOLS: BGP, DNS

# BGP example




- Transit: 2 provides transit for 7
- Algorithm seems to work OK in practice
  - BGP is does not respond well to frequent node outages

Figure: D. Wetherall

# BGP Security Issues

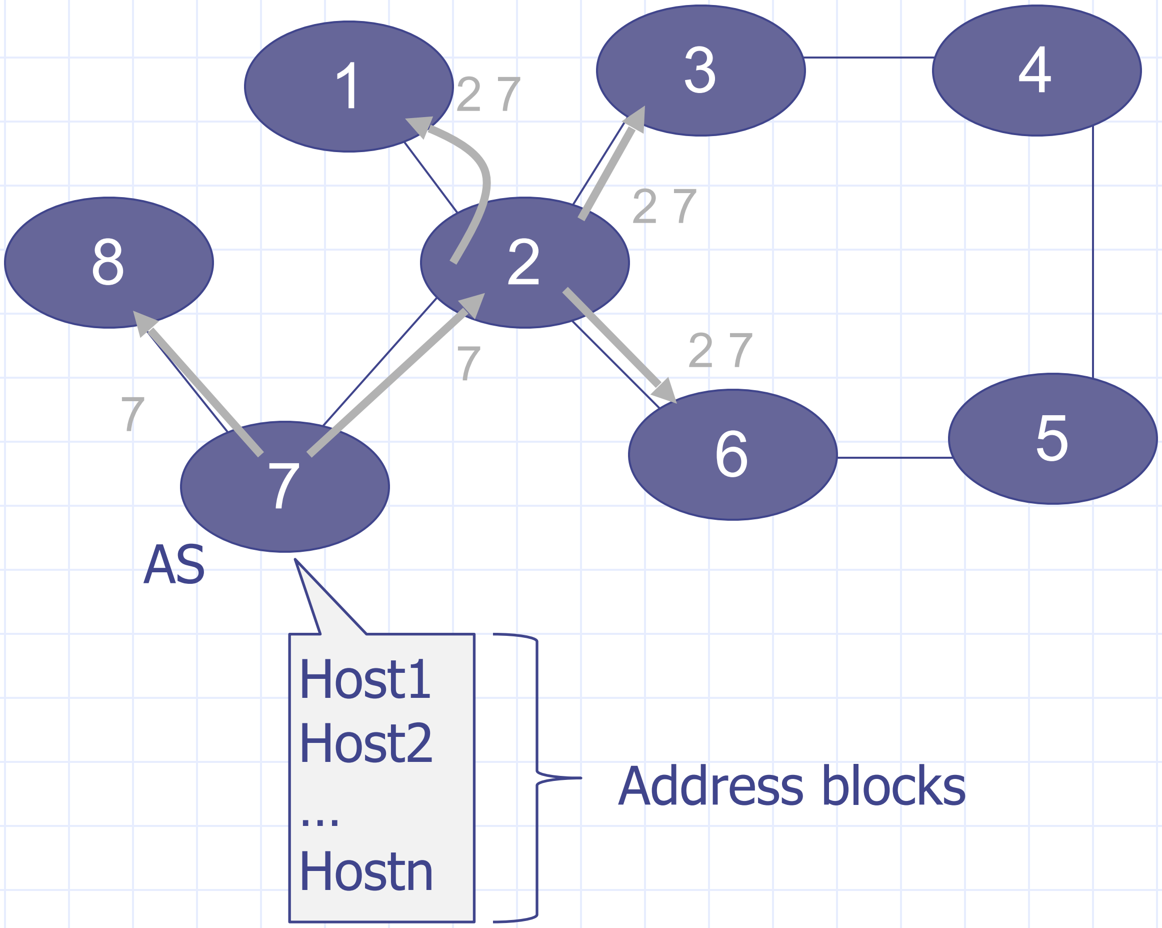
- BGP is used for all inter-ISP routing
- Benign configuration errors affect about 1% of all routing table entries at any time
- Highly vulnerable to human errors, malicious attacks
  - Actual routing policies can be very complicated
- MD5 MAC is rarely used, perhaps due to lack of automated key management, addresses only one class of attacks

# S-BGP Design Overview

- IPsec: secure point-to-point router communication
- Public Key Infrastructure: authorization for all S-BGP entities
- Attestations: digitally-signed authorizations
  - Address: authorization to advertise specified address blocks
  - Route: Validation of UPDATEs based on a new path attribute, using PKI certificates and attestations
- Repositories for distribution of certificates, CRLs, and address attestations
- Tools for ISPs to manage address attestations, process certificates & CRLs, etc.

Slide: Steve Kent

# BGP example



# Address Attestation

- Indicates that the final AS listed in the UPDATE is authorized by the owner of those address blocks
- Includes identification of:
  - owner’s certificate
  - AS to be advertising the address blocks
  - address blocks
  - expiration date
- Digitally signed by owner of the address blocks
- Used to protect BGP from erroneous UPDATEs (authenticated but misbehaving or misconfigured BGP speakers)

# Route Attestation

- Indicates that the speaker or its AS authorizes the listener's AS to use the route in the UPDATE
- Includes identification of:
  - AS's or BGP speaker's certificate issued by owner of the AS
  - the address blocks and the list of ASes in the UPDATE
  - the neighbor
  - expiration date
- Digitally signed by owner of the AS (or BGP speaker) distributing the UPDATE, traceable to the IANA ...
- Used to protect BGP from erroneous UPDATEs (authenticated but misbehaving or misconfigured BGP speakers)

# Validating a Route

- To validate a route from $AS_n$, $AS_{n+1}$ needs:
  - address attestation from each organization owning an address block(s) in the NLRI
  - address allocation certificate from each organization owning address blocks in the NLRI
  - route attestation from every AS along the path ($AS_1$ to $AS_n$), where the route attestation for $AS_k$ specifies the NLRI and the path up to that point ($AS_1$ through $AS_{k+1}$)
  - certificate for each AS or router along path ($AS_1$ to $AS_n$) to check signatures on the route attestations
  - and, of course, all the relevant CRLs must have been checked

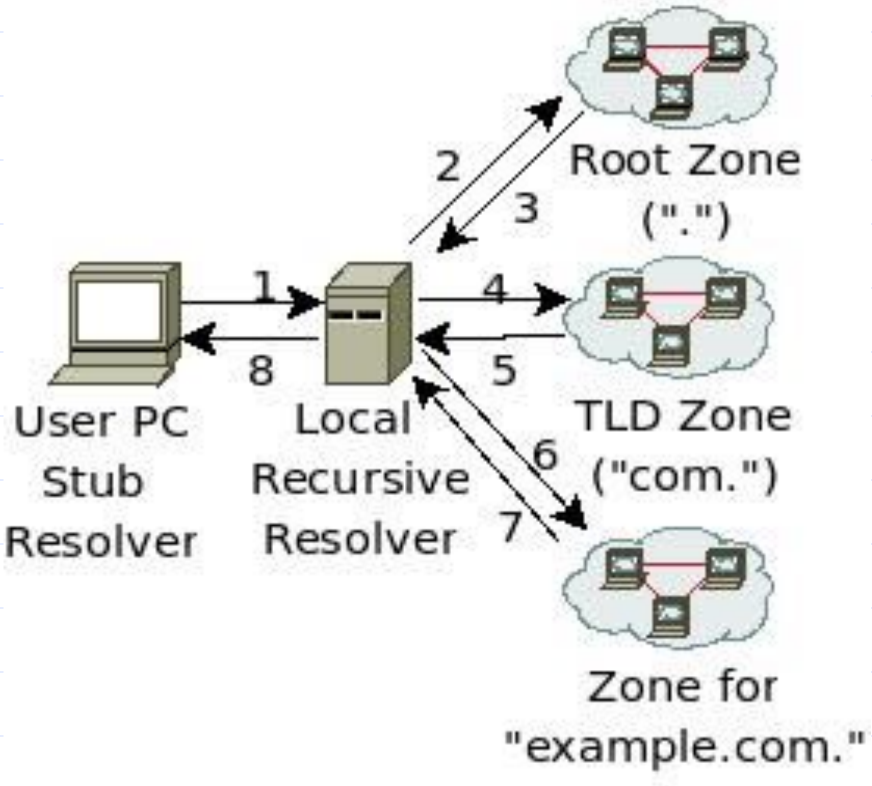
Slide: Kent et al.

# INFRASTRUCTURE PROTOCOLS: BGP, DNS

# Recall: DNS Lookup

Query: "www.example.com A?"

| Reply | Resource Records in Reply |
|---|---|
| 3 | "com. NS a.gtld.net"<br>"a.gtld.net A 192.5.6.30" |
| 5 | "example.com. NS a.iana.net"<br>"a.iana.net A 192.0.34.43" |
| 7 | "www.example.com A 1.2.3.4" |
| 8 | "www.example.com A 1.2.3.4" |




Local recursive resolver caches these for TTL specified by RR

# DNS is Insecure

- Packets sent over UDP, < 512 bytes
- 16-bit TXID, UDP Src port are only "security"
- Resolver accepts packet if above match
- Packet from whom? Was it manipulated?

- Cache poisoning
  - Attacker forges record at resolver
  - Forged record cached, attacks future lookups
  - Kaminsky (BH USA08)

# DNSSEC Goal

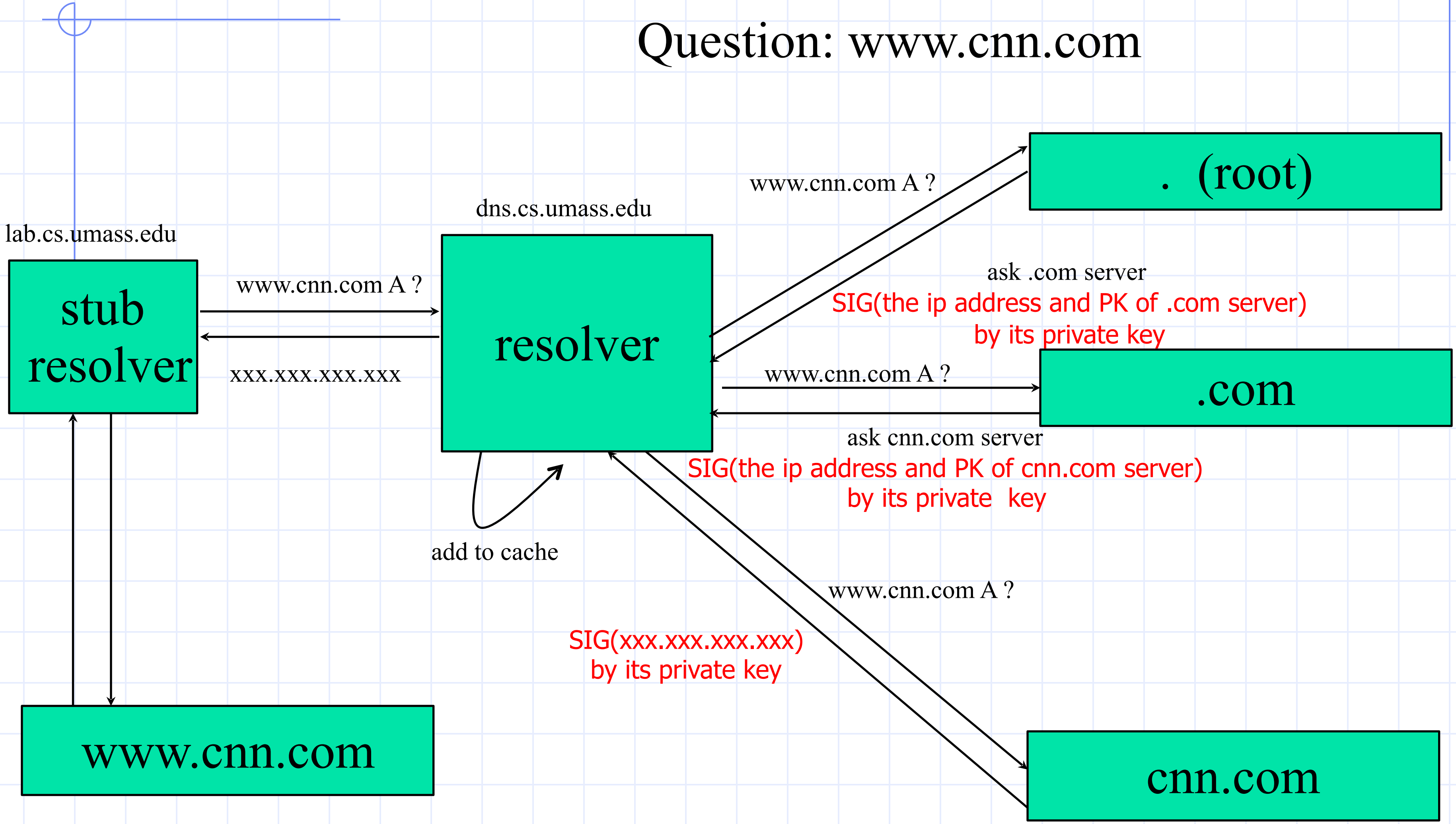
"The Domain Name System (DNS) security extensions provide origin authentication and integrity assurance services for DNS data, including mechanisms for authenticated denial of existence of DNS data."

-RFC 4033

# DNSSEC

- Basically no change to packet format
  - Goal is security of DNS data, not channel security
- New Resource Records (RRs)
  - RRSIG : signature of RR by private zone key
  - DNSKEY : public zone key
  - DS : crypto digest of child zone key
  - NSEC / NSEC3 authenticated denial of existence
- Lookup referral chain (unsigned)
- Origin attestation chain (PKI) (signed)
  - Start at pre-configured trust anchors
    - DS/DNSKEY of zone (should include root)
  - DS → DNSKEY → DS forms a link

# Verifying the tree

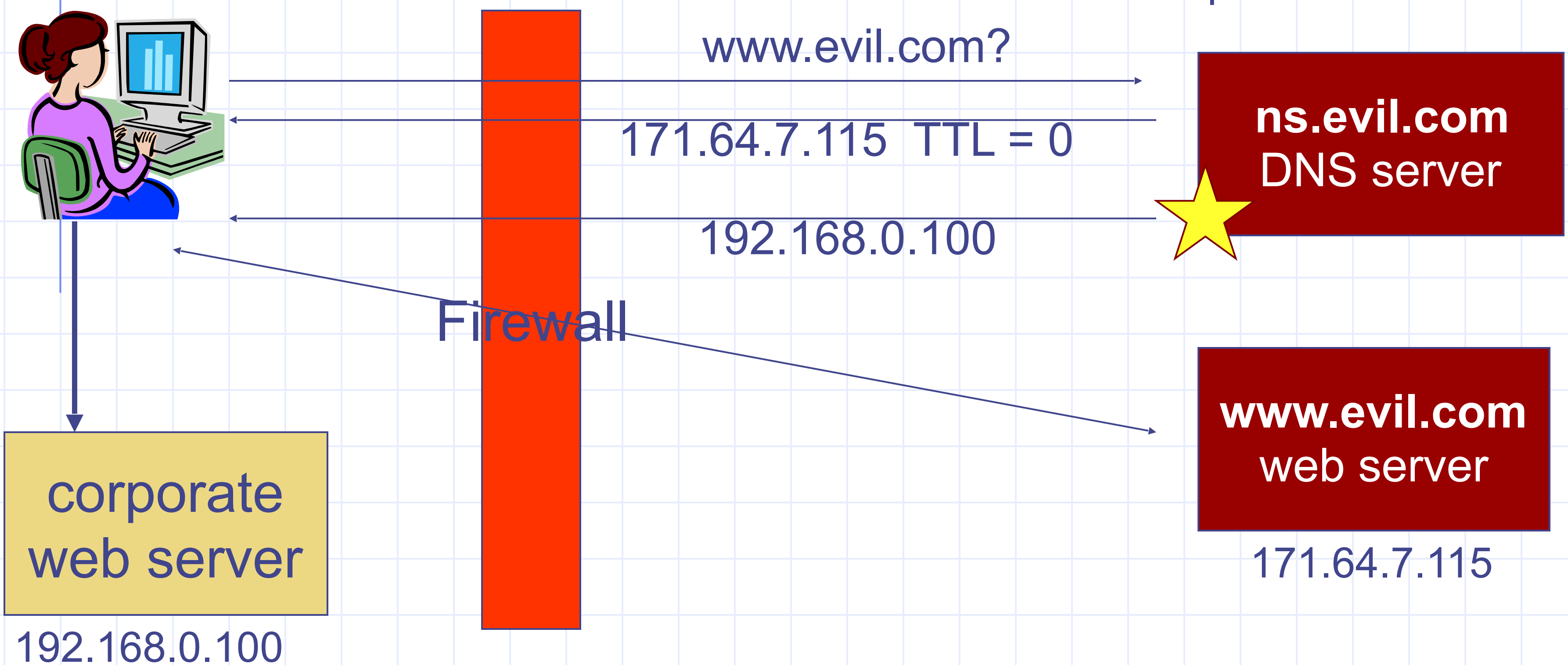


[DWF’96, R’01]

# DNS Rebinding Attack

<iframe src="**http://www.evil.com**">

DNSSEC cannot stop this attack




Read permitted: it’s the “same origin”

# DNS Rebinding Defenses

- Browser mitigation: DNS Pinning
  - Refuse to switch to a new IP
  - Interacts poorly with proxies, VPN, dynamic DNS, …
  - Not consistently implemented in any browser
- Server-side defenses
  - Check Host header for unrecognized domains
  - Authenticate users with something other than IP
- Firewall defenses
  - External names can’t resolve to internal addresses
  - Protects browsers inside the organization

# Summary of this section

- Network infrastructure protocols
  - BGP vulnerabilities and S-BGP
    - Security can be achieved by applying cryptography and basic network connection security to every step
    - Heavyweight solution, but illustrates the ways BGP can be vulnerable
  - DNS security, rebinding attack
    - Domain-name security achieved by additional infrastructure
    - Most complicated part is addressing non-existence

# Summary

- Protecting network connections
  - Wireless security – 802.11i/WPA2
  - IPSEC
- Perimeter network perimeter defenses
  - Firewall
    - Packet filter (stateless, stateful),
    - Application layer proxies
  - Intrusion detection
    - Anomaly and misuse detection
- Network infrastructure security
  - BGP vulnerability and S-BGP
  - DNSSEC, DNS rebinding