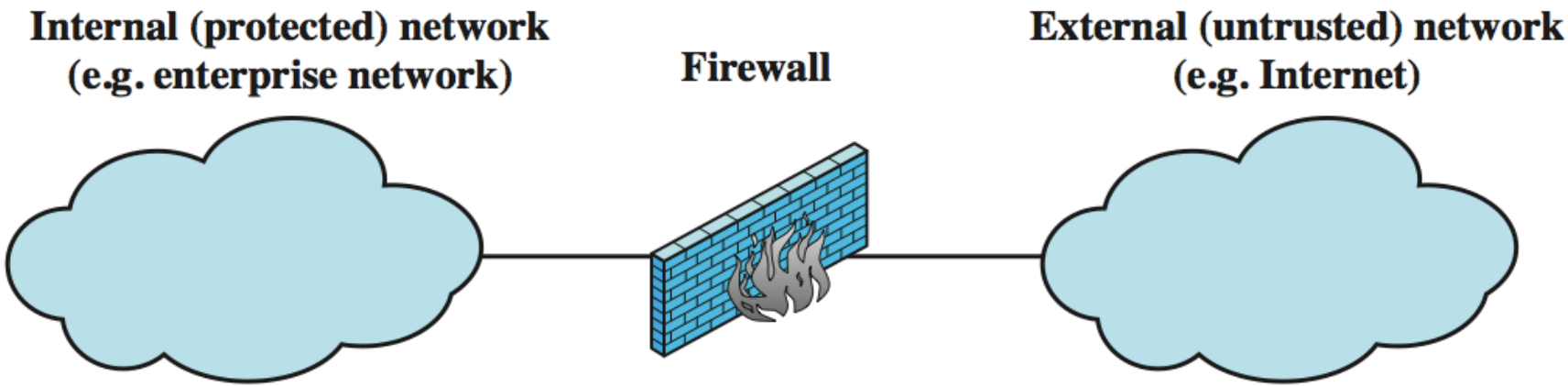
Lecture 25 – Network Defense 1: Firewalls & IPSec

Ryan Cunningham
University of Illinois
ECE 422/CS 461 – Fall 2017

Security News

- London Bridge Plastic Surgery hacked
- Elmedia Player infected with Trojan
- DUHK – Don’t Use Hard Coded Keys

Firewall Model



firewall: isolates organization's internal net from larger Internet, allowing some packets to pass while blocking others

Purpose of Firewalls (threats)

- Prevent denial of service attacks:
  - SYN flooding: attacker establishes many bogus TCP connections, no resources left for “real” connections.
- Prevent illegal modification/access of internal data
  - e.g., attacker replaces CIA’s homepage with something else
- Allow only authorized access to inside network (set of authenticated users/hosts)

Purpose of Firewalls (internal)

- Bandwidth control
  - Block high bandwidth applications
  - Netflix, BitTorrent
- Employee network usage control
  - Block games, pornography, non-business uses
- Privacy
  - Don’t let outside see what you have, how big you are, etc.
  - Similar to making corporate phone directory proprietary

Firewall Requirements

- All traffic between inside and outside passes through firewall
- Only authorized traffic is allowed through
- The firewall itself is immune to penetration

Filtering Characteristics

- IP address and protocol values
  - source *and* destination IP address, ports
  - outbound vs. inbound traffic
- Application protocol
  - Filter SMTP out spam, HTTP for unauthorized websites
- User Identity
  - Access control for secure authentication
- Network Activity
  - Time of day, rate of requests

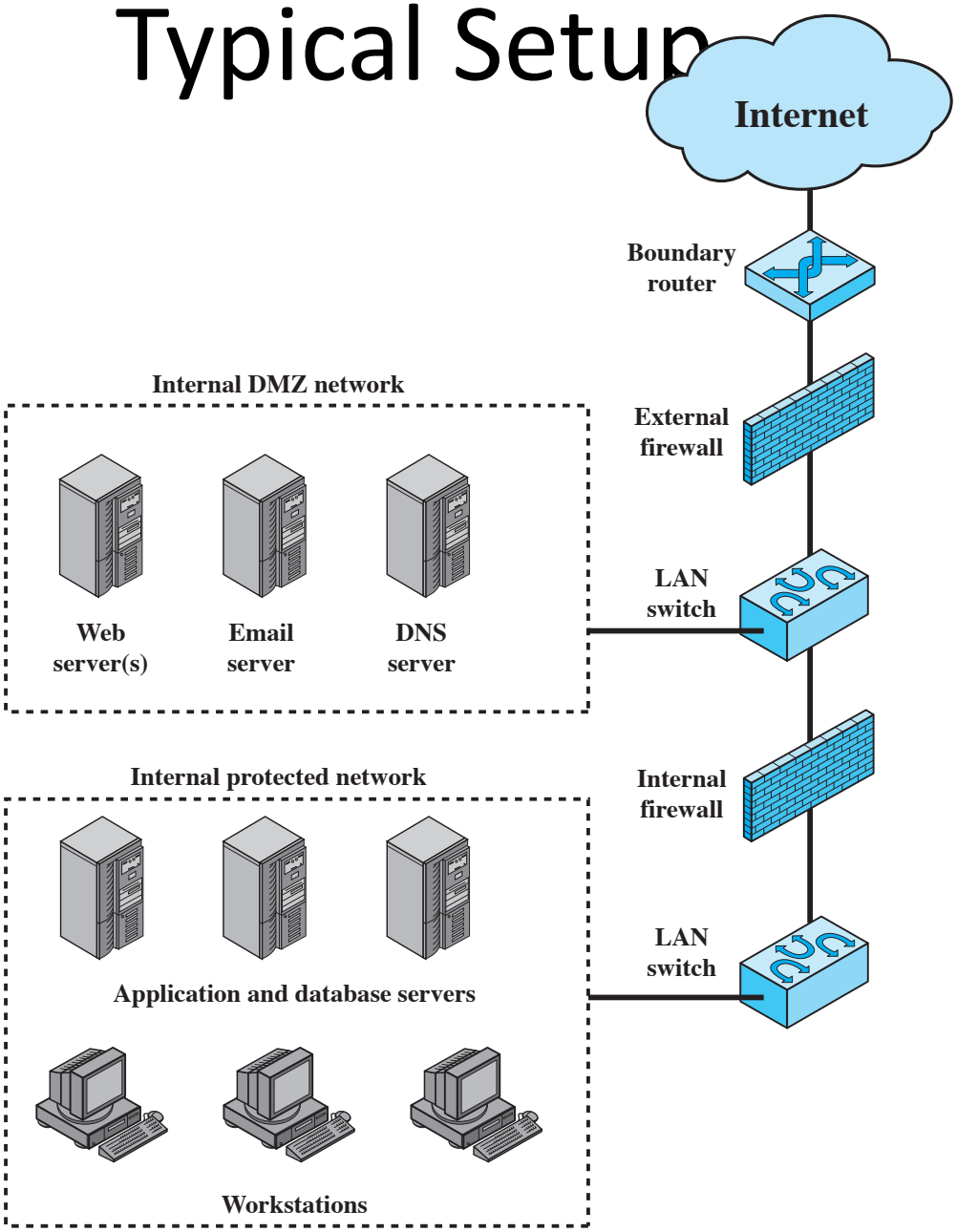
Firewall Capabilities

1. Single choke point to consolidate security services
2. Location for monitoring security events
3. Can perform NAT between outside/inside
4. Can provide IPSec platform to build VPN

Firewall Limitations

1. Cannot protect against attacks that bypass firewall
2. Cannot fully protect against internal threats
3. Can be difficult to isolate wireless networks
4. Cannot protect against devices that physically move on and off (e.g. phones and laptops)

Typical Setup



Types of Firewalls

1. Packet filters
2. Stateful firewalls
3. Application-level gateway

1) Packet Filters

- Inspect packets one at a time based on:
  - Source IP address
  - Destination IP address
  - Source port
  - Destination port
  - Protocol field (i.e. TCP vs. UDP)
- Set of rules define which packets get through
  - Default = forward or discard?

1) Packet Filters

Rule	Direction	Src address	Dest address	Protocol	Dest port	Action
1	In	External	Internal	TCP	25	Permit
2	Out	Internal	External	TCP	>1023	Permit
3	Out	Internal	External	TCP	25	Permit
4	In	External	Internal	TCP	>1023	Permit
5	Either	Any	Any	Any	Any	Deny

1. Allow inbound mail
2. Allow response from outbound SMTP connection
3. Allow outbound mail
4. Allow response from inbound SMTP connection
5. Default policy: deny everything else

1) Packet Filters

Rule	Direction	Src address	Dest address	Protocol	Dest port	Action
1	In	External	Internal	TCP	25	Permit
2	Out	Internal	External	TCP	>1023	Permit
3	Out	Internal	External	TCP	25	Permit
4	In	External	Internal	TCP	>1023	Permit
5	Either	Any	Any	Any	Any	Deny

NOT GOOD ENOUGH!

1) Packet Filters

Rule	Direction	Src address	Dest address	Protocol	Dest port	Action
1	In	External	Internal	TCP	25	Permit
2	Out	Internal	External	TCP	>1023	Permit
3	Out	Internal	External	TCP	25	Permit
4	In	External	Internal	TCP	>1023	Permit
5	Either	Any	Any	Any	Any	Deny

- Might do better if we inspect flags
  - (e.g. SYN, ACK, FIN)
- Might do better if we track source ports

1) Packet Filters (Disadvantages)

- Cannot detect application level commands
- Logs limited data (same as access control decisions)
- Doesn’t support user authentication
- IP addresses can be spoofed
- Can be vulnerable to fragment attacks (tiny packets force header to be split into multiple packets)
- Can easily be misconfigured

2) Stateful Firewalls

- Sometimes, *context* of a packet is important
- If we know the *state* of a protocol, we can filter better
  - e.g. track each SMTP connection
  - e.g. deny SYN reflection attacks
- Stateful firewall tracks *same info* as packet filter
- Keep track of connections in a *connection state table*
- Can make decisions based on packet rules *and* state of connection

2) Stateful Firewalls

Source Address	Source Port	Destination Address	Destination Port	Connection State
192.168.1.100	1030	210.9.88.29	80	Established
192.168.1.102	1031	216.32.42.123	80	Established
192.168.1.101	1033	173.66.32.122	25	Established
192.168.1.106	1035	177.231.32.12	79	Established
223.43.21.231	1990	192.168.1.6	80	Established
219.22.123.32	2112	192.168.1.6	80	Established
210.99.212.18	3321	192.168.1.6	80	Established
24.102.32.23	1025	192.168.1.6	80	Established
223.21.22.12	1046	192.168.1.6	80	Established

2) Stateful Firewalls (Disadvantages)

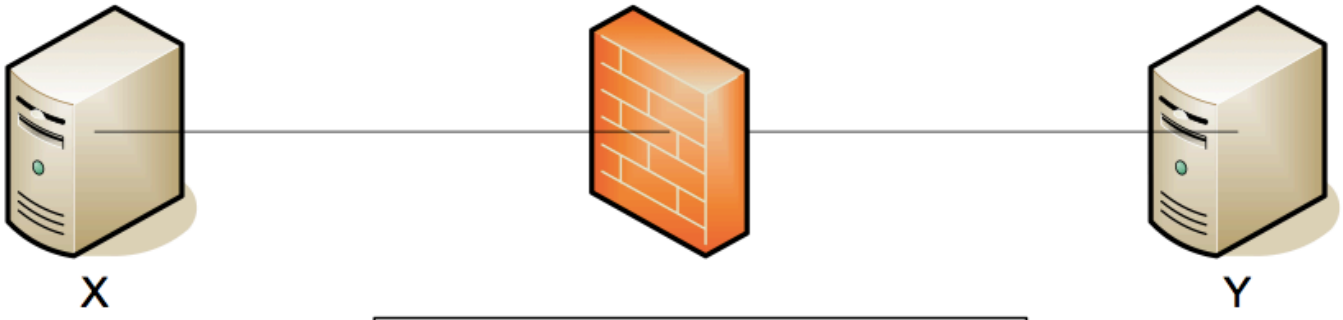
- Largely the same as packet filters
- More overhead than packet filters
- More difficult to configure
- Possibly vulnerable to attack
- Still can’t filter at application layer!

3) Application Proxy Firewall

- Firewall runs in application space of firewall server
- Filter connections based on application behavior
  - Block Java
  - Filter out visits to bad URLs
  - Block suspicious protocols

3) Application Proxy Firewall

Traffic reconstruction



GET command causes firewall to dynamically open data channel initiate from Y to X

Might have filter for files to block, like /etc/passwd

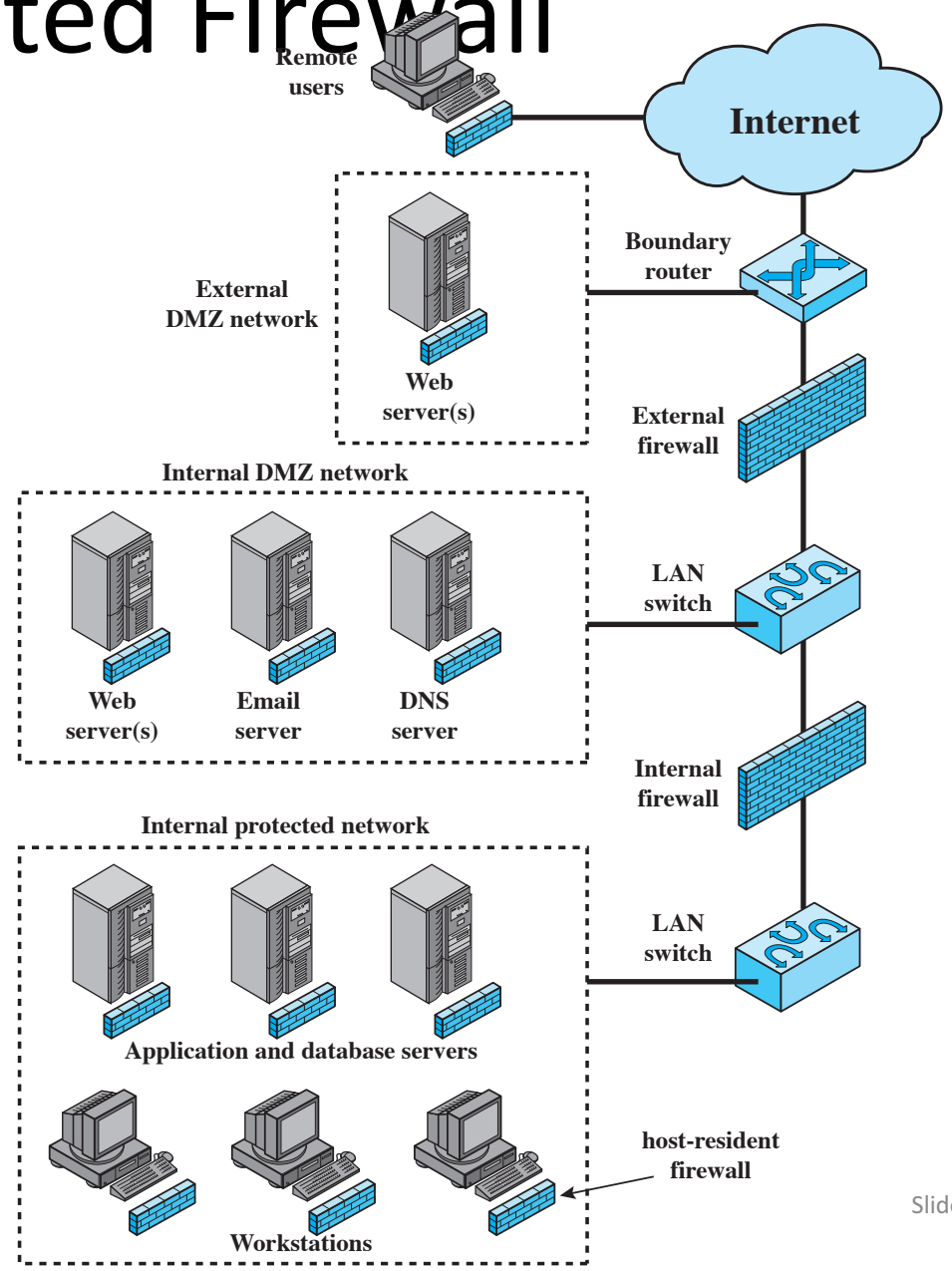
3) Application Proxy Firewall (Disadvantages)

- *Much* more overhead than packet filter or stateful firewall
- Adds latency to all traffic
- Can be circumvented via encryption or steganography
- Much more complicated to set up

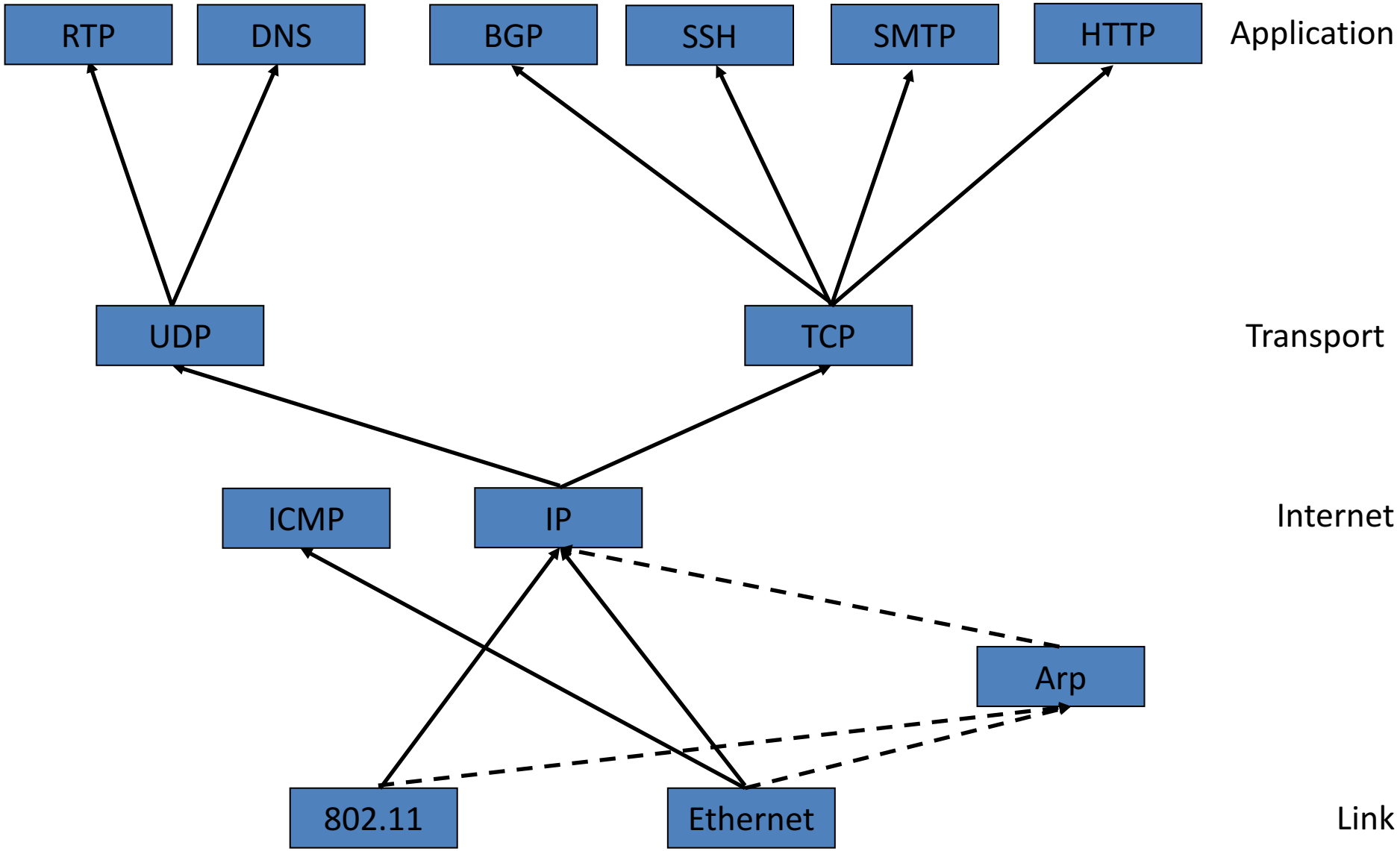
Firewall Basing

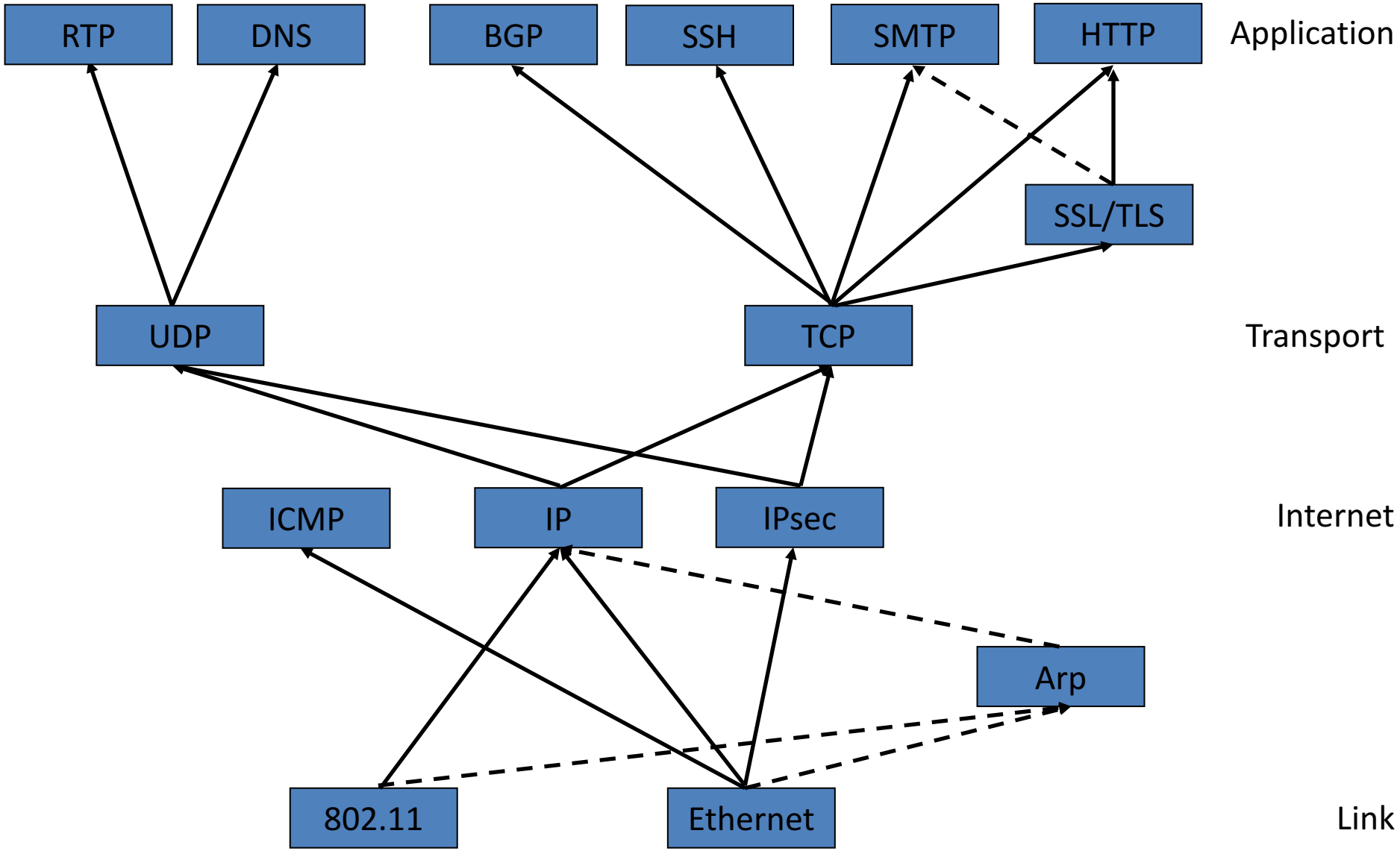
- Router/LAN Switch
- UNIX/LINUX server
- Bastion host
  - Hardened system living in the DMZ
    - e.g. performs no writes after boot up!
  - Can be used to host application proxy firewall
- Host-based firewall
- Personal firewall

Distributed Firewall



Secure Protocols


RTP
DNS
BGP
SSH
SMTP
HTTP
Application
UDP
TCP
Transport
ICMP
IP
Internet
Arp
802.11
Ethernet
Link


RTP
DNS
BGP
SSH
SMTP
HTTP
Application
SSL/TLS
UDP
TCP
Transport
ICMP
IP
IPsec
Internet
Arp
802.11
Ethernet
Link

SSL/TLS

- transport layer security to any TCP-based app using SSL services.
- used between Web browsers, servers for e-commerce (shttp).
- security services:
  - server authentication
  - data encryption
  - client authentication (optional)

- server authentication:
  - SSL-enabled browser includes public keys for trusted CAs.
  - Browser requests server certificate, issued by trusted CA.
  - Browser uses CA's public key to extract server's public key from certificate.
- check your browser's security menu to see its trusted CAs.

SSL/TLS (continued)

Encrypted SSL session:

- Browser generates *symmetric session key*, encrypts it with server's public key, sends encrypted key to server.
- Using private key, server decrypts session key.
- Browser, server know session key
  - All data sent into TCP socket (by client or server) encrypted with session key.

- SSL: basis of IETF Transport Layer Security (TLS).
- SSL can be used for non-Web applications, e.g., IMAP.
- Client authentication can be done with client certificates.

IPsec: Network Layer Security

- Network-layer secrecy:
  - sending host encrypts the data in IP datagram
  - TCP and UDP segments; ICMP and SNMP messages
- Network-layer authentication
  - destination host can authenticate source IP address
- Two principle protocols:
  - authentication header (AH) protocol
  - encapsulation security payload (ESP) protocol

- For both AH and ESP, source, destination handshake:
  - create network-layer logical channel called a security association (SA)
- Each SA is unidirectional
- Uniquely determined by:
  - security protocol (AH or ESP)
  - source IP address
  - 32-bit connection ID

IPSec Applications

- Establish secure network over internet (e.g. VPN)
  - Remote access, e-commerce
- Transparent to application layer
- Can be implemented by routers
  - Protect user data without their even knowing
- Also secures routing itself
  - Routers themselves are authorized

Authentication Header (AH) Protocol

- provides source authentication, data integrity, no confidentiality
- AH header inserted between IP header & data field
- protocol field: 51
- intermediate routers process datagrams as usual

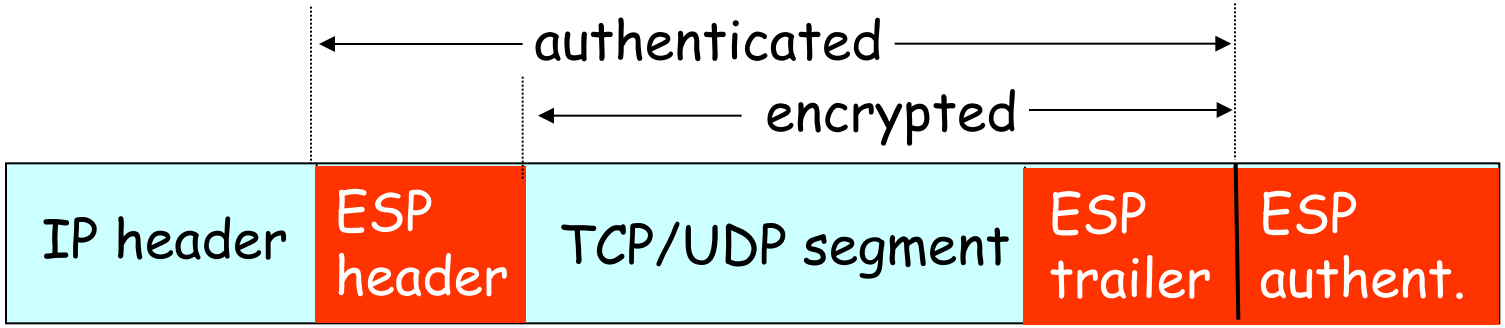
AH header includes:

- connection identifier
- authentication data: source-signed message digest calculated over original IP datagram
- next header field: specifies type of data (e.g., TCP, UDP, ICMP)

ESP Protocol

- provides secrecy, host authentication, data integrity
- data, ESP trailer encrypted
- next header field is in ESP trailer

- ESP authentication field is similar to AH authentication field
- Protocol = 50.



Transport vs. Tunneling

- Two IPSec modes of operation:

1. Transport: encrypt/authenticate payload only
2. Tunneling: wrap entire IP packet in new IPsec packet

- Tunneling allows for NAT
- Tunneling allows for VPNs