eecs489 COMPUTER NETWORKS

Lecture 14:
IPSec, Firewall

At Which Layer to Put Security?

Link-oriented vs. end-to-end

Which layer?

- application layer: secure email (PGP), SSH, DNSSec
- above TCP: Secure Socket Layer (SSL) Netscape, 1994, used by HTTPS
- IPsec: Authentication Header (AH) and Encapsulating Security Payload (ESP)

Security at the Network Layer

There are security concerns that apply to multiple applications and cut across protocol layers

Shouldn't security be implemented by the network layer for all applications?

Benefits of network-layer security:

- below transport layer: transparent to applications
- can be transparent to end users
- helps secure routing architecture

[after Rexford]

IPSec: Network Layer Security

Provides:

- network-layer authentication: destination host can authenticate source IP address
- network-layer confidentiality and integrity:
  - sending host encrypts the data in IP datagram

Two principle protocols:

- authentication header (AH) protocol
- encapsulation security payload (ESP) protocol
- mandatory in IPv6, optional in IPv4

IPSec: Network Layer Security

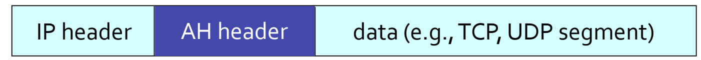
To use either AH or ESP, source and destination hosts perform connection handshake:

- to create a network-layer end-to-end logical channel called a security association (SA)
- SA sets up a shared secret between the two hosts
- each SA is unidirectional, uniquely determined by:
  - security protocol (AH or ESP)
  - source IP address
  - 32-bit connection ID

Authentication Header (AH) Protocol

Provides source authentication and data integrity, but no confidentiality/secrecy

AH header (IP protocol# 51) inserted between IP header and payload



Intermediate routers process datagrams as usual

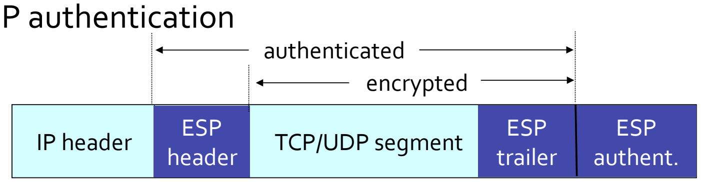
AH header includes:

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

ESP Protocol

Provides source authentication, data integrity, and confidentiality/secrecy

ESP header (IP protocol# 50) inserted between IP header and payload, followed by ESP trailer and ESP authentication



- payload and ESP trailer are encrypted
- next header field is in ESP trailer
- optional authentication is similar to AH's

ESP is used to provide VPNs

What is a VPN?

Makes a shared network look like a private network

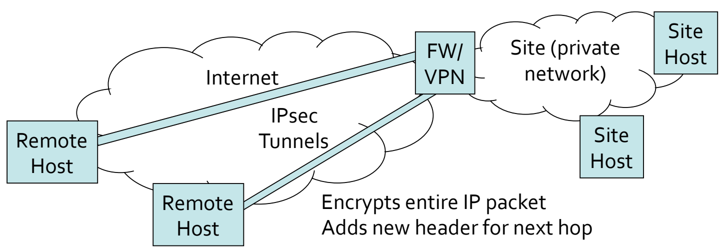
Why VPN?

- VPN makes separated IP sites look like one private IP network
- private addresses and domain names (useful for authorization)
- security
- bandwidth guarantees, quality of service (QoS), service-level agreement (SLA) across ISP
- simplified network operation: ISP can do the routing for you
- building a real private network is expensive (cheaper to use shared resources rather than to have dedicated resources)

End-to-end VPNs

Solves the problem of connecting remote hosts to a firewalled network

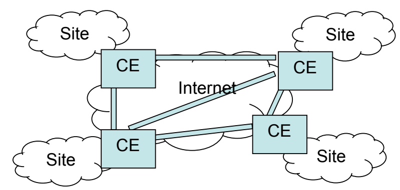
- commonly used for roaming
- benefits in the form of security and private addresses only
  - no simplicity or QoS benefit



Network VPNs

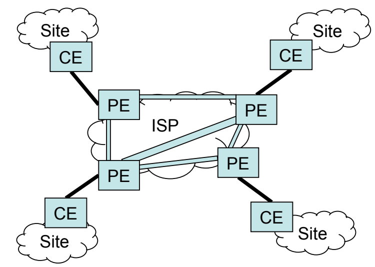
Customer based:

- customer buys own equipment, configures IPSec tunnels across the global Internet, manages addressing and routing
- ISP plays no role
- customer has more control over security and ISP choices, but requires skills



Provider based:

- provider manages all the complexity of the VPN, usually with MPLS (see lecture on MPLS)
- customer simply connects to the provider equipment



Denial of Service (DoS) Attack

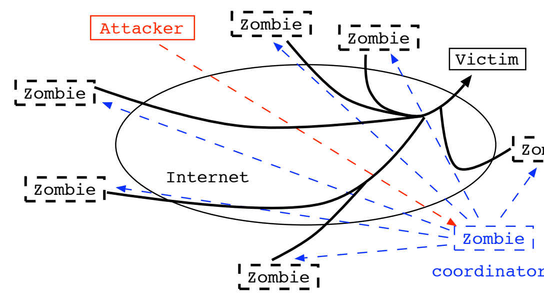
An attacker inundates its victim with otherwise legitimate service requests or traffic such that victim's resources are overloaded and overwhelmed to the point that the victim can perform no useful work and legitimate users are denied service

Examples:

- SYN flood: send lots of TCP SYN to fill up victim's listen queue
- smurf attack: broadcast ICMP echo requests with the source address spoofed to victim's
- IP fragmentation
- TCP reassembly

Distributed DoS (DDoS) Attack

Attacker commandeers systems (zombies) distributed across the Internet, forming a botnet, to send correlated service requests or traffic to the victim to overload the victim

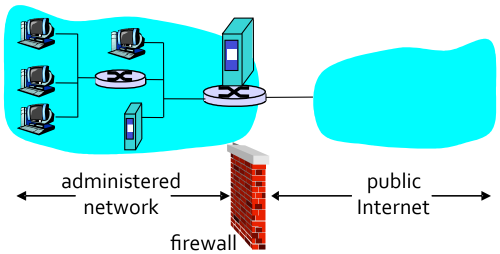


Firewalls

One way to protect against hosts being taken over and made zombies

A firewall: a barrier that restricts the free flow of data between the "inside" and the "outside"

A border checkpoint to monitor traffic for known attack patterns (intrusion detection)



Firewalls

Goals:

- to prevent illegal modification or access of internal data
- to allow only authorized access to internal network (set of authenticated users/hosts)
- to prevent host intrusion for launching a DDoS attack
- in general, to isolate an organization's internal network from the larger Internet, allowing some packets to pass through, blocking others

Design criteria:

- all traffic must go through the firewall
- only authorized traffic will be allowed to pass
- authorization must be a local security policy
- the firewall itself must be immune to penetration

Advantages of Firewalls

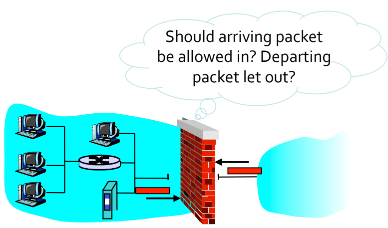
Firewall vs. security hardening of each host:

- more convenient: border checkpoint
- more secure: a firewall is not a general purpose machine
  - administrator of firewall is more security conscious (hopefully)
  - firewall has restricted access, usually can be made less convenient to use (e.g., requiring one-time passcode)
- central point of control for mail, ftp, web administration
- can also be used internally (between departments)

Types of Firewall

Three types of firewall:

1. packet filter
2. circuit-level gateway
3. application-level gateway



Packet-filter:

- drop packet not matching any given header patterns
- filter usually constructed out of source address, destination address, source port, destination port, TCP flag fields, ICMP message type, deep packet inspection (DPI) of packet contents
- allow for wild-card values

Packet Filtering Examples

Block all packets with IP protocol field = 17 and with either source or destination port = 23

- all incoming and outgoing UDP flows are blocked
- all Telnet connections are blocked

Block inbound TCP packets with SYN flag set but ACK flag not set

- blocks external clients from making TCP connections with internal clients
- but allows completion of internally initiated TCP connections

Block all packets with TCP port of Quake

[after Rexford]

Packet Filter Configuration

Firewall applies a set of rules to each packet

- to decide whether to allow or block the packet

Each rule is a test on the packet

- comparing IP and TCP/UDP header fields to determine whether to allow or block

Order matters

- once a packet matches a rule, the decision is made

[after Rexford]

Packet Filter Configuration Example

Alice runs a network with address prefix `222.22.0.0/16`

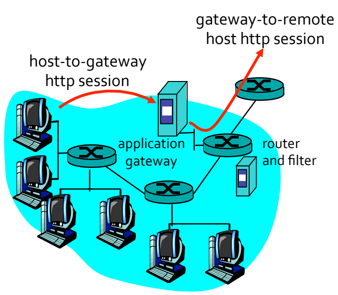
- she wants to allow Bob's school to access hosts on her subnet `222.22.22.0/24`
  - Bob is on address prefix `111.11.0.0/16`
- Alice doesn't trust Trudy, who is inside Bob's network
  - Trudy is on address prefix `111.11.11.0/24`
- Alice doesn't allow any other traffic from the Internet

Rules

1. Don't let Trudy's machines in
   - `block(src = 111.11.11.0/24, dst = 222.22.0.0/16)`
2. Let the rest of Bob's network in to subnet `222.22.22.0/24`
   - `allow(src=111.11.0.0/16, dst = 222.22.22.0/24)`
3. Block the rest of the world
   - `block(src = 0.0.0.0/0, dst = 0.0.0.0/0)`

[after Rexford]

Gateways



Circuit-level gateways: force every connection to go through the gateway

Circuit-level gateways usually do not look at the bytes sent, it serves mainly for logging of allowed connections

Example:

1. require all web users to http through gateway
2. for authorized users, gateway sets up http connection to remote host
3. gateway relays data between 2 connections
4. router's filter blocks all http connections not originating from gateway

Application-level gateways further parse traffic and allow only known operations

A Variation: Traffic Management

Allow vs. block is too binary a decision

- maybe better to classify the traffic based on rules and then handle the classes of traffic differently

Traffic shaping (rate limiting)

- limit the amount of bandwidth certain traffic may consume, e.g., rate limit Web or P2P traffic

Separate queues

- use rules to group related packets and then do round-robin scheduling across groups, e.g., separate queue for each internal IP address

[Rexford]

Firewall Implementation Challenges

Per-packet handling

- must inspect every packet
- challenging on very high-speed links

Complex filtering rules

- may have large number of rules
- may have very complicated rules
- filter specification language could make certain policies hard to express
- filters interaction could give rise to unintended policies (bugs in filter specifications)
- filter specification for protocols without fixed port number difficult
- not very effective against connectionless UDP (filters often use all or nothing policy for UDP)

[after Rexford]

Firewall Deployment Challenges

Location of firewalls

- complex firewalls near the edge, at low speed
- simpler firewalls in the core, at higher speed

Other limitations:

- IP spoofing: firewall can't know if data "really" comes from claimed source
- if multiple apps need special treatment, each needs its own app gateway
- client software must know how to contact gateway, e.g., must set IP address of proxy in Web browser

Tradeoff: degree of communication with outside world vs. level of security

Clever Users Subvert Firewalls

Many highly protected sites still suffer from attacks

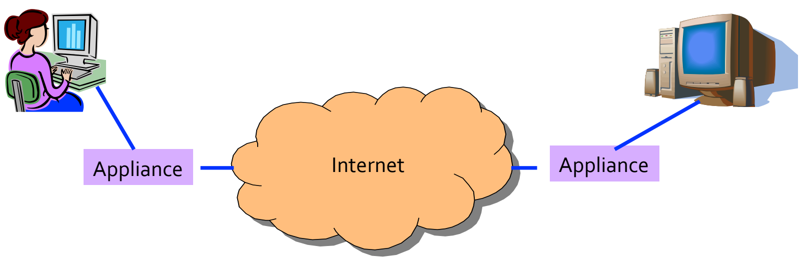
Packet-filter and circuit-level gateways can be (ab)used for tunneling under the firewall

- The Great Firewall of China prevents access to certain IP addresses and port numbers
  - users may log into another machine and then onward to the blocked servers
  - can be automated by the use of VPN or TOR
- firewall allows only port 80 (e.g., Web) traffic
  - p2p client uses port 80

Middleboxes

Firewalls, NAT boxes, traffic shapers, web caches, tunnel (VPN) endpoints are examples of so-called "middleboxes"

- intermediaries interposed in-between communicating hosts
- often without the knowledge of one or both parties



[after Rexford]

Advantages of Middleboxes

Improve performance between edge networks

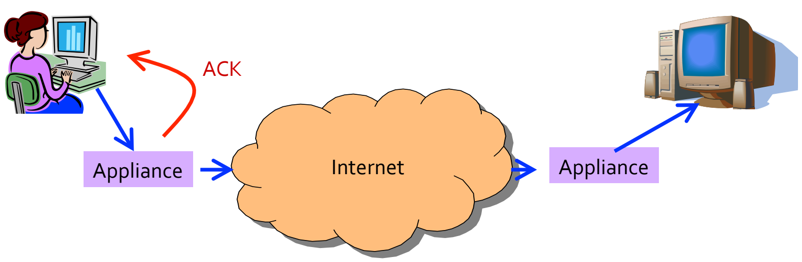
- e.g., multiple sites of the same company or between residential access network and data centers
- through buffering, compression, encryption, caching, etc.

Incrementally deployable

- no changes needed at the end hosts or the rest of the Internet
- inspects packets as they go by and takes necessary action transparent to the user

[after Rexford]

Example: WAN Accelerators



Appliance with a lot of local memory

Sends ACK packets quickly to the sender

Overwrites receive window with a large value

Or, even run a new and improved version of TCP

[after Rexford]

Two Views of Middleboxes

A practical necessity that

- solves real and pressing problems
- meets needs that are not likely to go away

An abomination

- a violation of layering
- causes confusion in reasoning about the network
- is responsible for many subtle bugs

[after Rexford]

Network Security (Summary)

Basic techniques
- cryptography (symmetric and public)
- authentication, message integrity, digital signature
- key distribution

used in many different security scenarios
- secure email (PGP), secure connection (SSH)
- secure transport (SSL)
- IPsec
- 802.11 (WPA)

Network Security (Summary)

Network security is an ongoing arms race
- breaking things is fun to some
- *ad hoc* approaches

And then there's unwanted traffic, spam, phising, and user tracking and privacy violation, etc.