Network Security in Practice

Kai Shen

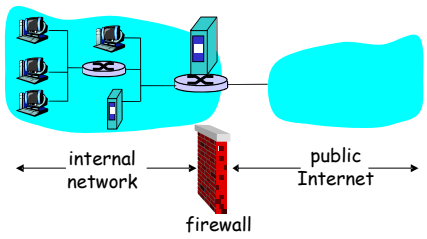
Practices of Network Security

- Access control: firewalls
- Attacks and counter measures
- Security protocol case studies

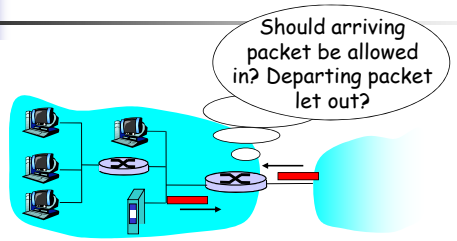
Access Control: Firewalls

firewall

isolates organization's internal network from the public Internet through filtering, allowing some data to pass, blocking others.



Network-layer Packet Filtering



- Firewall is built into the edge router connected to the Internet
- Router filters packet-by-packet, decision to forward/drop packet based on:
  - source IP address, destination IP address
  - TCP/UDP source and destination port numbers
  - TCP SYN and ACK bits

Policies in Network-layer Packet Filtering

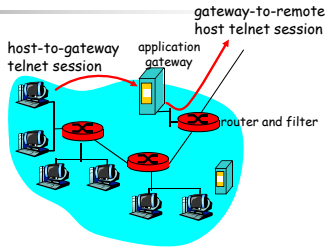
- Example 1: blocking all incoming TCP datagrams with dest port = 80
  - No external clients can access internal Web servers.
- Example 2: blocking all TCP datagrams with source or dest port = 23, except for those with source or dest IP = 128.151.67.155 (a particular internal machine)
  - All incoming and outgoing telnet connections have to go through a telnet gateway.
- Example 3: blocking all incoming TCP datagrams with ACK bit set to 0
  - Prevents external clients from initiating TCP connections with internal clients, but allows internal clients to connect to outside.

More on Network-layer Packet Filtering

- Advantage:
  - transparent to network applications
  - incurring little extra overhead/latency
- Limitation:
  - relying only on IP/TCP/UDP header info
    ⇒ not flexible enough, e.g., firewall can know the IP of the source, but not the "user"

Application-layer Gateways

- Access control according to application-layer information.
- Example: allow selected internal users to telnet outside.



1. Router filter blocks all telnet connections not originating from gateway ⇒ require all telnet users to telnet through gateway.
2. For authorized users, gateway sets up telnet connection to dest host.

Practices of Network Security

- Access control: firewalls
  - network-layer firewall
  - application-layer firewall
- Attacks and countermeasures
- Security protocol case studies

Network Security Threat: Mapping

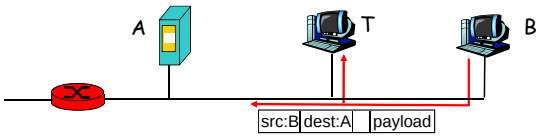
- Before attacking: "scout the area" – find out what services are implemented on network
- Try to determine what host addresses are valid on the network
- Port-scanning: try to establish TCP connection to each port in sequence (see what happens)

Countermeasures:

- Record traffic entering network
- Look for suspicious activity (e.g., IP addresses, ports being scanned sequentially)

Network Security Threat: Packet Sniffing

- promiscuous NIC reads all packets passing by a broadcast media (e.g. shared-link Ethernet)
- can read all unencrypted data (e.g. passwords)

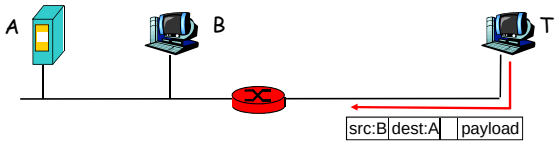


Countermeasures:

- checks periodically if host interface in promiscuous mode.
- one host per segment of broadcast media (switched Ethernet)
- encrypt all packets.

Network Security Threat: IP Spoofing

- with root privilege, one can generate "raw" IP packets with any value into IP source address field
- receiver can't tell if source is spoofed
- e.g.: T pretends to be B



Countermeasures:

- authentication
- ingress filtering – routers should not forward outgoing packets with invalid source addresses

Network Security Threat: Cross-Site Scripting

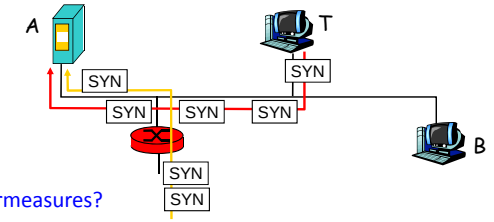
- Cross-site scripting:
  - duped to run script unintended by the original site
  - most significant vulnerability for web applications today
- Examples:
  - search engine FOOBAR displays the input search keywords in the return page; attacker prepares a search URL that includes HTML tag and JavaScript code; a user who trusts FOOBAR clicks the search URL
  - attacker supplies attack string as msg to a msg board
  - attacker embeds attack strings in machine names

Countermeasures?

- Careful input checking

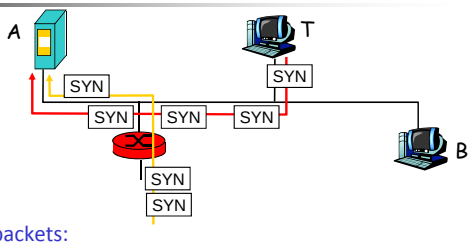
Network Security Threat: Denial-of-service Attack

- SYN flooding: attacker establishes many bogus TCP connections, flood of maliciously generated packets "swamp" receiver
- Distributed DOS (DDOS): multiple coordinated sources swamp receiver
- e.g., T and remote host SYN-attack A



Countermeasures?

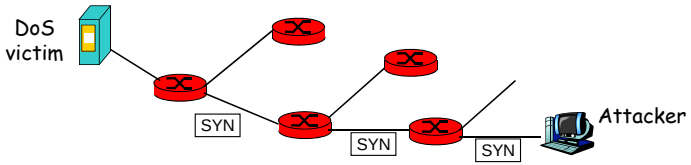
Countermeasure 1: Packet Filtering



Filtering out attack packets:

- attack packets carry spoofed IP addresses – hard to filter based on IP address
- if filtering out all SYN packets, then no good connections
- if filtering out some SYN packets, throw out good and bad connections

Countermeasure 2: Trace Back



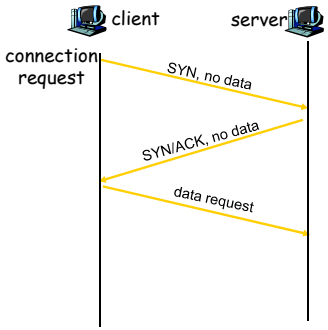
Trace back to flood source:

- attack packets with spoofed IPs
- trace back through network statistics
- sources are most likely innocent, compromised machines

Countermeasure 3: Delayed Processing

Delayed processing or resource allocation:

- Data structure allocation and initialization at receipt of real data request, not at receipt of first SYN
- What if attacker sends SYN, waits for SYNACK, and then sends some dummy data?



Stateless TCP

Stateless TCP [Shieh et al. NSDI 2005]:

- server side maintains no state about TCP connections
- advantage: TCP connections only require temporary space during packet processing
- state for a TCP connection:
  - receive buffer
  - send buffer
  - various control parameters and network statistics
- how to avoid maintaining such state at server side?
- also useful for transparent server fail-over/migration

Practices of Network Security

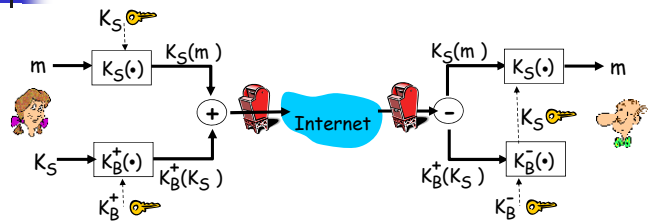
- Access control: firewalls
- Attacks and counter measures
  - mapping, sniffing, spoofing, cross-site scripting, DOS attack
- Security protocol case studies
  - Application-layer PGP: secure email
  - Transport-layer SSL: secure sockets
  - Network-layer IPsec: secure networking

Secure Email: Confidentiality

Alice wants to send confidential e-mail, m, to Bob.

- encrypts message with Bob's public key

Secure Email: Confidentiality



Alice:

- generates *symmetric* key, K_S
- encrypts message with K_S
- encrypts K_S with Bob's public key
- sends both $K_S(m)$ and $K_B^+(K_S)$ to Bob

Bob:

- uses his private key to decrypt and recover K_S
- uses K_S to decrypt $K_S(m)$ to recover m

Secure Email: Sender Authentication and Message Integrity

- Sender authentication and message integrity:
  - generates a digital signature of the message digest using its private key
- Put everything together
  - uses one-time session key and the receiver's public key to encrypt a digitally signed message
  - supports confidentiality, sender authentication, and message integrity
  - PGP (pretty good privacy) for Internet email

Secure Sockets Layer (SSL)/ Transport Layer Security (TLS)

- SSL/TLS: transport layer security service to any TCP-based applications
  - used for remote terminal access (SSH).
  - used between Web browsers, servers for e-commerce (https).
  - used between IMAP clients and servers.
- Security services:
  - CA-certified public keys.
  - data confidentiality by encryption using a symmetric session key, key encrypted with server's public key.
  - source authentication & data integrity by signed message digests.

Network Layer Security Protocol IPsec

- Like before:
  - data confidentiality by encryption using a symmetric session key
  - source authentication & data integrity by signed message digests

- Done in a way that is compatible with basic IP routing functions
  - easy deployment – require no router changes

More on IPsec

- Transport mode:

- Tunnel mode:

- Transport mode is more natural for a host-to-host secure conn; tunnel mode is better fit for intermediate segment of secure conn between two routers.
- (Virtual Private Network) VPN:

Network Security (summary)

Basic techniques……

- cryptography (symmetric and public)
- authentication
- message integrity
- key distribution

…. network security in practice

- firewall
- attacks and countermeasures
- secure application (PGP for email)
- secure transport (SSL/TLS)
- secure network (IPsec)

Disclaimer

- Parts of the lecture slides contain original work of James Kurose, Larry Peterson, and Keith Ross. The slides are intended for the sole purpose of instruction of computer networks at the University of Rochester. All copyrighted materials belong to their original owner(s).