

Network Security in Practice

Kai Shen

12/8/2014 CSC 257/457 - Fall 2014 1

Practices of Network Security

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

12/8/2014 CSC 257/457 - Fall 2014 2

Access Control: Firewalls

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

12/8/2014 CSC 257/457 - Fall 2014 3

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 flags (SYN and ACK bits)

12/8/2014 CSC 257/457 - Fall 2014 4

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.

12/8/2014

CSC 257/457 - Fall 2014

5

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”

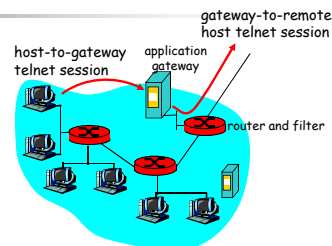
12/8/2014

CSC 257/457 - Fall 2014

6

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.

12/8/2014

CSC 257/457 - Fall 2014

7

Practices of Network Security

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

12/8/2014

CSC 257/457 - Fall 2014

8

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)

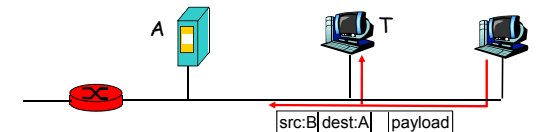
12/8/2014

CSC 257/457 - Fall 2014

9

Network Security Threat: Packet Sniffing

- Promiscuous NIC reads all packets passing by a broadcast media (e.g. shared-link Ethernet, WiFi)
- 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.

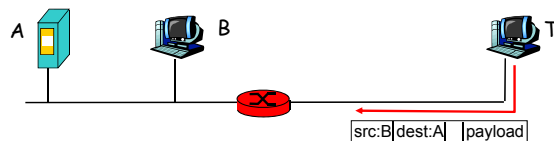
12/8/2014

CSC 257/457 - Fall 2014

10

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

12/8/2014

CSC 257/457 - Fall 2014

11

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:
 - attacker supplies attack string (including HTML tag and JavaScript code) as msg to msg board FOOBAR; a user who trusts FOOBAR views msgs and his/her browser would run attack JavaScript
 - search engine FOOBAR displays the input search keywords in the return page; attacker prepares a search query with attack string; a user who trusts FOOBAR clicks the search
 - attacker embeds attack strings in machine names

Countermeasures?

- Careful input checking. Dependence tracking through tainting.

12/8/2014

CSC 257/457 - Fall 2014

12

Network Security Threat: Denial-of-service Attack

- SYN flooding: attacker establishes many bogus TCP connections, flood of maliciously generated packets “swamp” receiver
 - Resource use at victim; resource use by the attacker
- Distributed DOS (DDOS): multiple coordinated sources swamp receiver
 - e.g., T and remote host SYN-attack A

Countermeasures?

12/8/2014 CSC 257/457 - Fall 2014 13

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

12/8/2014 CSC 257/457 - Fall 2014 14

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


12/8/2014 CSC 257/457 - Fall 2014 15

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?

12/8/2014 CSC 257/457 - Fall 2014 16




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?
 - timeout management for continuation state is a bit tricky
- also useful for transparent server fail-over/migration/load-balancing

12/8/2014 CSC 257/457 - Fall 2014 17



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).

12/8/2014 CSC 257/457 - Fall 2014 18