Firewalls

IT443 – Network Security Administration

Slides courtesy of Bo Sheng
Internet Security Mechanisms

- **Goal:** prevent if possible; detect quickly otherwise; and confine the damage

- **Prevent:** Firewall, IPsec, SSL
- **Detect:** Intrusion Detection
- **Survive/Response:** Recovery, Forensics
Firewalls

- Provides secure connectivity between networks

- Implements and enforces a security policy for communication between networks
Firewalls

• Many organizations have distinct needs
  – access by anyone to public data concerning the company
  – access only by employees to internal data

• Solution: inner and outer (DMZ) networks
Firewall Functions

• Controlled access
  – restrict incoming and outgoing traffic according to security policy

• Others
  – log traffic, for later analysis
  – network address translation
  – encryption / decryption
  – application (payload) transformations
Limitations of Firewalls

• Cannot protect against traffic that does not cross it
  – i.e., there may be other ingress points to the network, such as modems or wireless access points, that bypass the firewall
  – doesn’t protect against “inside” attacks

• Configuration of firewalls to accomplish a desired high-level security policy is non-trivial
Filtering

• Compare traffic to patterns, then process traffic according to rules if matched

• Two styles
  – packet filtering
  – session filtering
Packet Filtering

- Patterns specify values in the header of a single packet, e.g.,
  - source IP address and port number
  - destination IP address and port number
  - transport protocol type
Packet Filtering

• Decisions made on a per-packet basis
  – no state information (about previous packets) is maintained or used

• Assessment
  – easy to implement
  – but limited capabilities

• May be subject to tiny-fragment attack
  – first fragment has only a few bytes
  – rest of TCP header in a second fragment, not examined by firewall
Session Filtering

• Packet decisions are made in the context of a connection or flow of packets

• If packet is the start of a new connection…
  – check against rules for new connections

• If packet is part of an existing connection…
  – check against state-based rules for existing connections
  – update state of this connection
Session Filtering

• Assessment
  – more powerful than packet filtering, can recognize more sophisticated threats or implement more complex policies
  – also more expensive to implement
iptables

• Tables
  – Filter
    • Packet filtering, default table
  – Nat
    • Rewrite packet source/destination
  – Mangle
    • Alter packet header/content
  – Raw
    • Avoid connection track
iptables

• Build-in chains
  – INPUT
  – OUTPUT
  – FORWARD
  – PREROUTING
  – POSTROUTING
iptables

• Basic syntax
  - iptables [-t table] [-[AD] chain rule-spec [options]]

  - Rules
    • Match condition
      - E.g., -s 192.168.1.102, --dport 80
    • Target (-j): ACCEPT, DROP/REJECT, QUEUE, or RETURN

  - iptables -L INPUT
  - iptables -A INPUT -p tcp --dport 22 -j ACCEPT
iptables

• Basic syntax
  – Insert: `iptables -I INPUT 2 ...`
  – Delete:
    • `iptables -D INPUT -p tcp --dport 22 -j ACCEPT`
    • `iptables -D INPUT 2`
    • `iptables -t filter -F INPUT`
iptables

• Examples
  – Network setting:
    • Server (VM): 172.16.190.131
    • Client 1(VM): 172.16.190.132
    • Client 2(Host): 172.16.190.1
    • sudo apt-get install openssh-server telnetd
  – Block ping
    • iptables -A INPUT -p icmp -j DROP
  – Block ping from client 1
    • iptables -A INPUT -s 172.16.190.132 -p icmp -j DROP
iptables

• Examples
  – Network setting:
    • Server (VM): 172.16.190.131
    • Client 1(VM): 172.16.190.132
    • Client 2(Host): 172.16.190.1

  – Block all requests from client 2 except ssh
    • `iptables -A INPUT -s 172.16.190.1 -j DROP`
    • `iptables -A INPUT -p tcp -s 172.16.190.1 --dport 22 -j ACCEPT`
      WILL NOT WORK! 1st rule shadows second!
      ORDER MATTERS!
iptables

• Examples
  – Network setting:
    • Server (VM): 172.16.190.131
    • Client 1(VM): 172.16.190.132
    • Client 2(Host): 172.16.190.1
  – Allow at most 1 telnet login from each client
    • `iptables -A INPUT -p tcp --syn --dport 23 -m connlimit --connlimit-above 1 -j DROP`
  – Limit the rate of ping to at most once per second
    • `iptables -A INPUT -p icmp -m limit --limit 1/s -- limit-burst 2 -j ACCEPT`
    • `iptables -A INPUT -p icmp -j DROP`