Network Firewalls

Don Porter
Firewalls: An Essential Tool

• Previous Lectures: Every service on a system visible to the outside world is a potential attack vector

• Observations:
  – It is really hard to police every single system for insecure software (although you should do this)
  – Some network services are intended only for use inside your network

• Idea: Filter incoming network connections
Example

Da’ Internet

Router

Switch

Public web server

Super-Secret Internal Database Server

80

3306

How to let users access database, but not bad guy?
Example

Incoming:
Allow: Web server, port 80
Else Deny

Outgoing:
Allow all

Direct outside connections to database blocked
Example Recap

• A firewall (aka packet filter) looks at packet headers and filters them based on attributes such as IP address and port number
• Can filter incoming and outgoing traffic
• Can log dodgy packets for further inspection
Types of Firewalls

• Most personal computers include firewall software
  – Linux: iptables
  – Windows: part of Microsoft Security Essentials

• For enterprise deployments, you can buy stand-alone firewall boxes from companies like Cisco

• For smaller deployments, a Linux system can also act as a firewall, using same software
  – In fact, many personal router/firewall/access point boxes run a lightweight Linux build + iptables
More Layered Security

• Some servers are intended to be publicly accessible (e.g., the web server)
• Others are for internal-use only and contain sensitive information (e.g., the database server)
• What happens if the web server is compromised?
  – Web server is inside the firewall
  – Can access the sensitive database server
  – Attacker can use web server to attack database
Refinement: DMZ

• Idea: Put a second firewall between public and private services
• We call the public part of the network the Demilitarized Zone (DMZ)
Example

Da’ Internet

Allow 80 in

Router

Switch

DMZ

Deny All in

DMZ

80

Public web server

Super-Secret Internal Database Server

3306
DMZ Recap

• Best practice: 2 firewalls
  – One between you and internet
  – One between public and private servers

• Limits damage if your web server is compromised
Incoming Traffic Caveats

• As presented, the rules are pretty simple:
  – E.g., block everything except traffic to web server

• But what about responses to external traffic?
  – E.g., http GET of www.google.com?

• Firewalls generally track some connection-level state, allow incoming responses to requests from inside the firewall
  – Sometimes called stateful inspection
  – States of note: Established and Related
Firewall Overview Summary

• Placing packet filters near your router can protect your network
  – Block access to private systems
  – Mitigate risk of user running a vulnerable service without your knowledge

• Multiple firewalls can be useful
  – DMZ
  – Host-level firewall

• Only one piece of the puzzle!
  – Disabling vulnerable services, security patches, etc., still matter
iptables

• Let’s walk through how you might configure iptables on a Linux machine
Key abstractions

- **Rules**: If packet matches X, take action Y

- Examples:
  
  - `-p tcp --dport 80 -j ACCEPT`
    - (If the packet is a TCP packet destined for port 80, allow)
  
  - `-s 87.84.250.101 -j DROP`
    - (If the packet comes from IP address 87.84.250.101, silently drop)
  
  - `-p icmp --limit 2/sec -j ACCEPT`
    - (Limit incoming pings to 2 per second)
Key Abstractions

• **Chains**: An ordered list of rules
  – Evaluation stops on a match

• Generally has the structure:
  
  If A, Accept
  If B, Accept
  ... (more accept rules)
  Drop everything else
Key Abstractions

• Tables: Collection of chains
  – Each chain applied to different stages of packet processing

• Default table: “filter”, has 3 chains:
  – INPUT – chain of rules for packets coming into local machine
  – OUTPUT – chain of rules for packets leaving the local machine (and that originated on the machine)
  – FORWARD – chain of rules for routed packets
    • I.e., packets that enter one device and leave on another
Detailed Example (command line)

```bash
iptables -F
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT
iptables -A INPUT --state RELATED,ESTABLISHED -j ACCEPT
iptables -A INPUT -p icmp --limit 2/sec -j ACCEPT
iptables -A INPUT -i lo -j ACCEPT
Iptables -p tcp --dport 22 -j ACCEPT
```
How to automatically reload?

- You can just type `sudo iptables -L` to see current state.
- You can dump the current iptables state using:
  ```bash
  sudo iptables-save > saved-rules
  ```
- You can restore the same rules again using:
  ```bash
  sudo iptables-restore < saved-rules
  ```
As part of boot...

• You can generally configure rules when a machine is brought up in /etc/network/interfaces
  – This is the standard network configuration file
  – Directive: pre-up

• Example:

  auto eth0
  iface eth0 inet dhcp
  pre-up iptables-restore < /etc/iptables.up.rules
Summary

• Firewalls can harden your network
  – But are not a panacea
• In fact, use 2 firewalls, and have a DMZ for public systems
• Iptables is good to have in your toolbox