Security

Security Vulnerabilities

- All software has bugs
- Sometimes bugs lead to adversaries gaining privileges

Security Vulnerabilities

- An attacker may:
  - Steal data/money
  - Decrypt secret data
  - Impersonate another person
  - Gain control of machines
  - Cripple a machine
  - Deny a service to legitimate users

Example – Buffer Overflow

```
#include <stdio.h>


int main() {
    char buffer[16];

    scanf("%s", buffer);

    return 0;
}
```

Example – Buffer Overflow

- What happens when text input is greater than 15 characters?
- Memory is overwritten with excess input
- Attacker can make use of that to make the program do something else
- Like spawn a shell
- To run arbitrary commands

Some Common Vulnerabilities

- Buffer Overflow
- Shell Injection
- SQL Injection
- Cross Site Scripting (XSS)
- Cross Site Request Forgery (CSRF)
- Session Hijack
- Insufficient Randomization
- Denial of Service (DOS)
- (Social Engineering!)

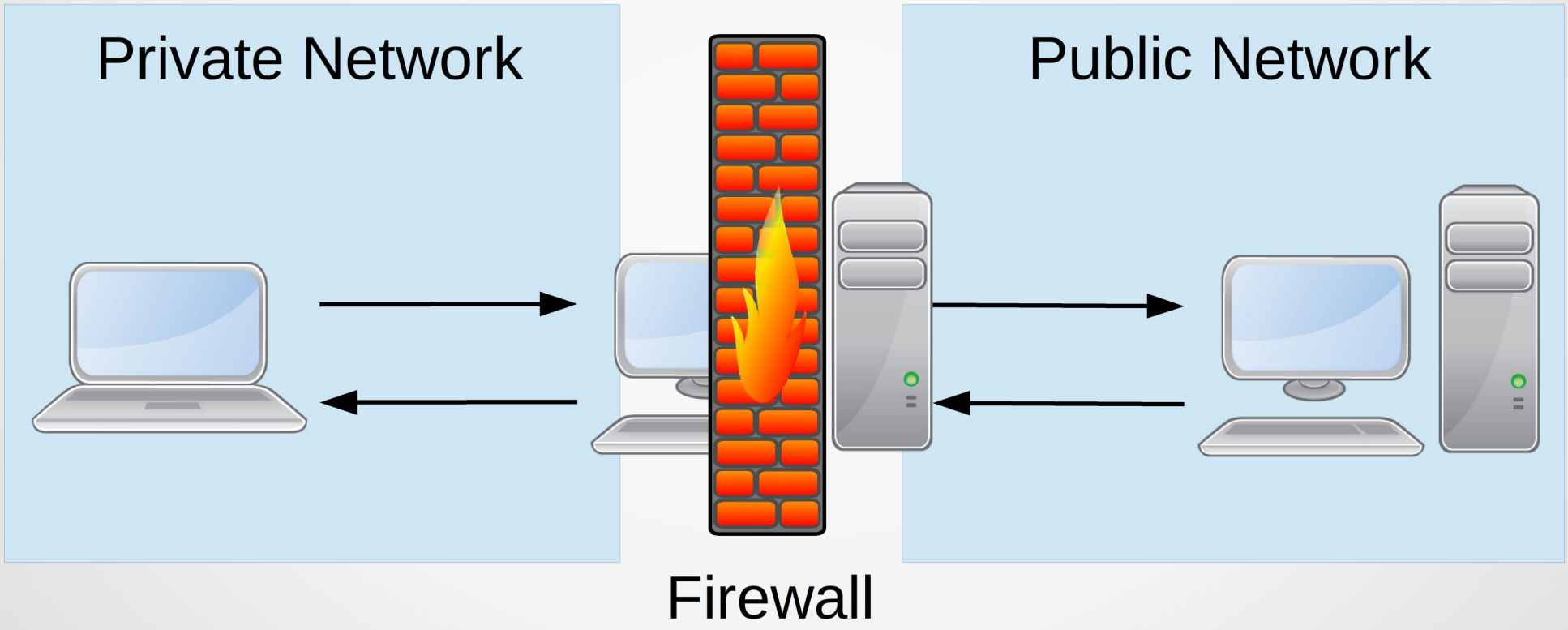
Security Updates

- Security problems in software are found regularly
- OS vendors give security updates to fix problems
- Install security updates regularly
- Setup automatic security updates
- If maintaining servers:
  - Subscribe to security announcement lists

Firewall

- Software/hardware to restrict incoming/outgoing network traffic
- Deployed between a trusted and untrusted network
- Deployed on a personal computer for additional security

Firewall



Firewall Rule – Example

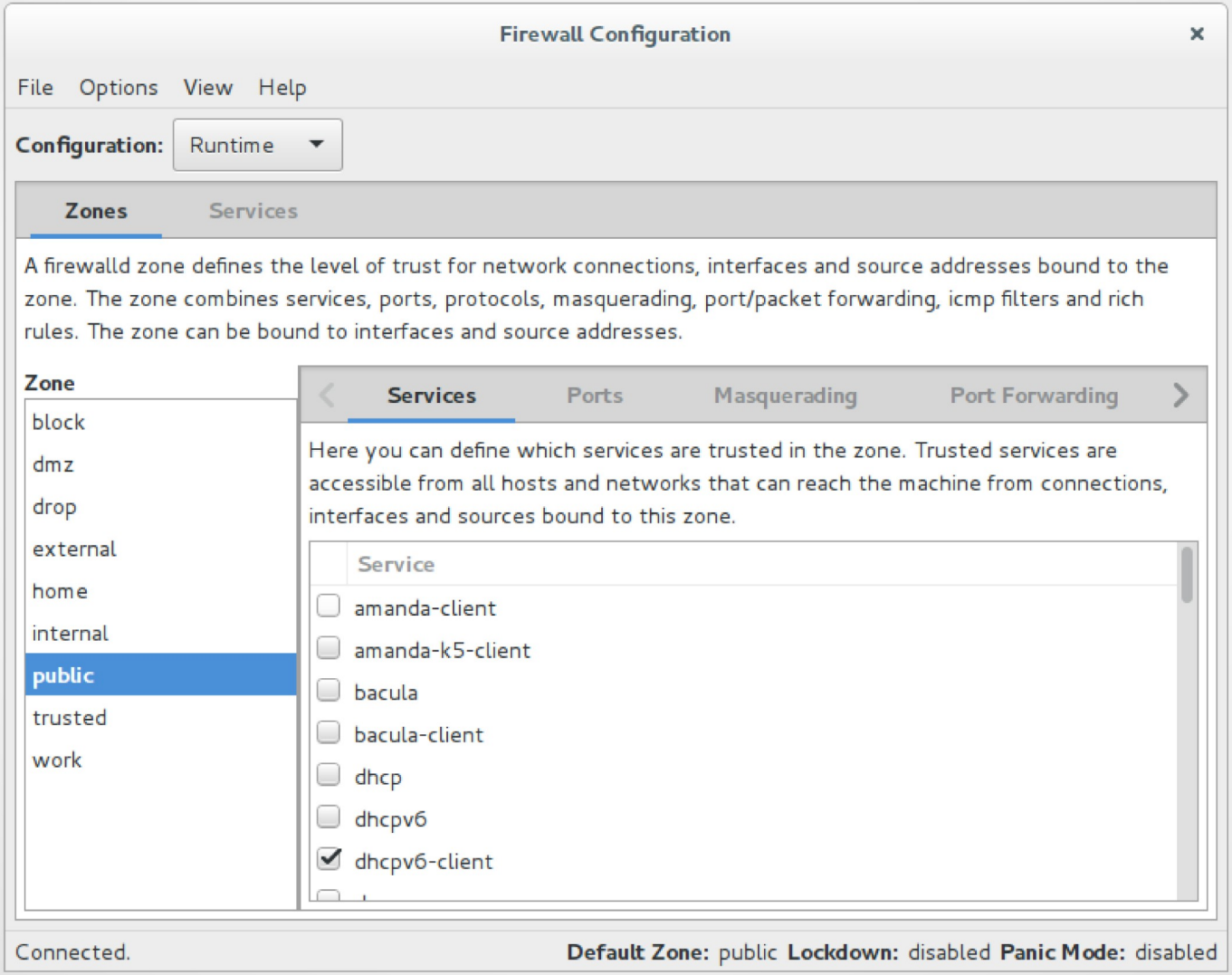
- Block all incoming traffic on port 80

```
$ iptables -A INPUT -p tcp --dport 80 -j DROP
$ iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt   source      destination
DROP       tcp  --    0.0.0.0/0   0.0.0.0/0    tcp dpt:80

Chain FORWARD (policy ACCEPT)
target     prot opt   source      destination

Chain OUTPUT (policy ACCEPT)
target     prot opt   source      destination
```

Easy Configuration - FirewallD



Port Scanning

- Send connection requests to all common ports on a computer
- Gather information based on the response

Port Scanning

- List machines on a network
- Identify operating system of a machine
- List ports open on machine
- Identify the service running on a port
- Identify the software and version for a service

Port Scanning – Examples

```
kirk@ent:~$ nmap -v -n -sn 192.168.1.0/24
Starting Nmap 6.47 ( http://nmap.org ) ...
Nmap scan report for 192.168.1.0 [host down]
Nmap scan report for 192.168.1.1
Host is up (0.00023s latency).
Nmap scan report for 192.168.1.2 [host down]
...
Nmap scan report for 192.168.1.255 [host
down]
```

Port Scanning – Examples

```
kirk@ent:~$ nmap -n -sT 192.168.1.6
Starting Nmap 6.47 ( http://nmap.org ) at ...
Nmap scan report for 192.168.1.6
Host is up (0.00055s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
8080/tcp open  http-proxy
Nmap done: 1 IP address (1 host up) scanned in
0.08 seconds
```

Penetration Testing

- Check a machine/network for known security vulnerabilities
- Also some potential problems
- After a secure setup, scan regularly
- Example: Metasploit Framework, W3af

References

- Iptables Documentation: http://nmap.org/book/toc.html
- Nmap Network Scanning: http://nmap.org/book/toc.html
- The Open Web Application Security Project: https://www.owasp.org
- The Web Application Hacker's Handbook: Discovering and Exploiting Security Flaws
- Metasploit: The Penetration Tester's Guide