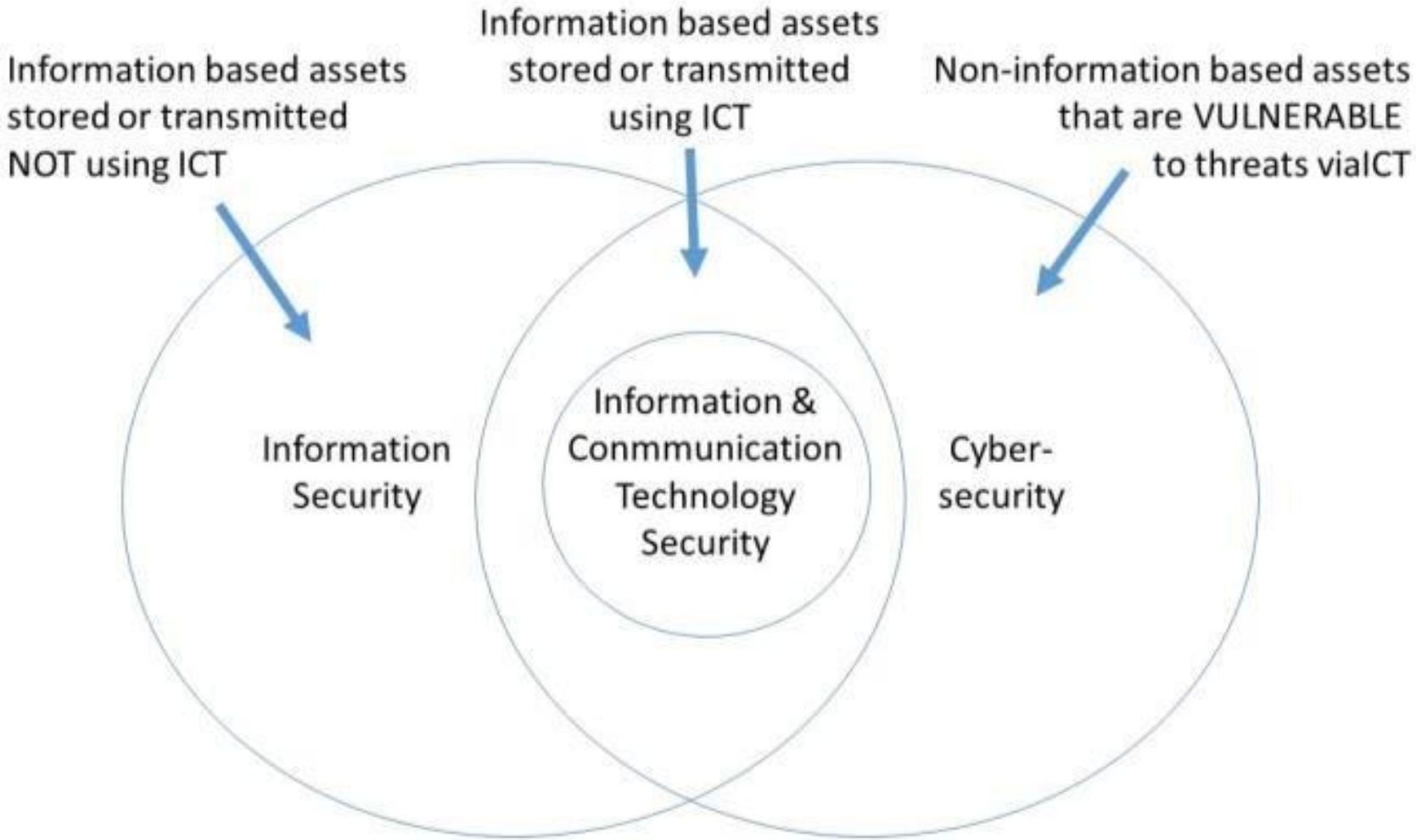
# Security


Information based assets stored or transmitted NOT using ICT
Information based assets stored or transmitted using ICT
Non-information based assets that are VULNERABLE to threats viaICT
Information Security
Information & Conmmunication Technology Security
Cyber-security


IDENTIFY
• Asset Management
• Vulnerability Assessment
• Governance, Risk & Compliance
RECOVER
• Continuity of Operations
• Disaster Recovery
• Threat Mitigation
PROTECT
• Threat Prevention
• Access Control
• Data Security
CYBER SECURITY
RESPOND
• Incident Response
• Malware Analysis
• Forensic Remediation
DETECT
• Continuous Monitoring
• Anomaly Detection
• User & Application Awareness

# Security

info sec.: C.I.A.

confidentiality: only sender, intended receiver should “understand” message contents

- sender encrypts message
- receiver decrypts message

message integrity: sender, receiver want to ensure message not altered (in transit, or afterwards) without detection (hash functions and digital signatures)

access and availability: services must be accessible and available to users (disaster recovery plan, redundancy)

non-repudiation: knowing who sent or received information (digital signatures)

authentication: sender, receiver want to confirm identity of each other (something you know, have, are)

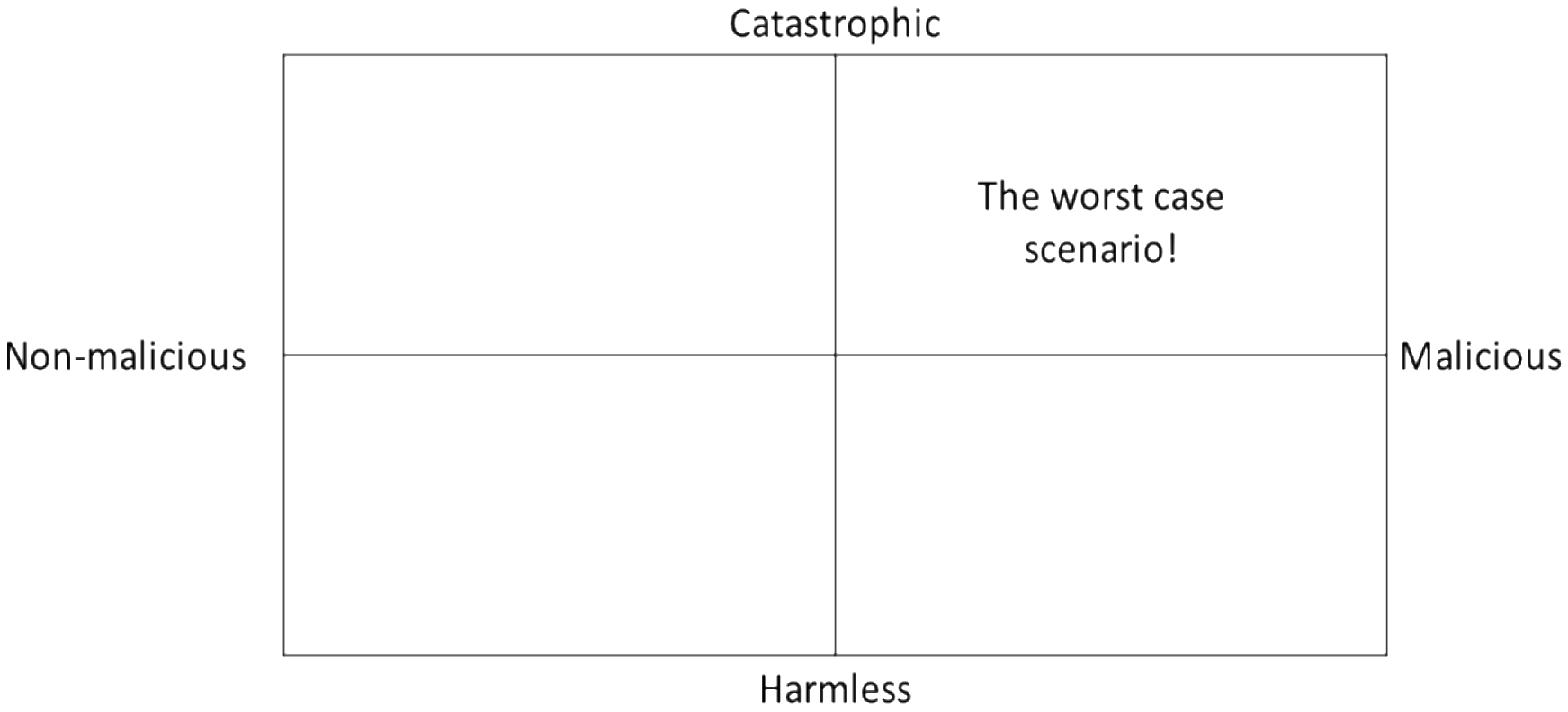
information assurance: C.I.A. + authentication + non-repudiation

authorization: determining if the client has permission to use or access a resource

# Cyber security

- Protecting cyber realm towards to cyber attacks and reducing the risks
- There are lots of hackers, cyber terrorists and spies
- Risks stem from errors of hardware & software

# Hardware & Software errors



# Cyber security

Three basic components

- Vulnerability
  - Weakness of a system
- Threat
  - Resolved when weakness is prevented
- Countermeasure
  - Resolving a vulnerability

# Cyber security


* BCP: Business Continuity Planning

# Cyber security

# Cyber resilience

- continuously deliver the intended outcome despite adverse cyber events
- collaboration of people, processes, technology and facilities
- cyber security and keeping things running

# Cyber security

**Attackers**

- **Amateur**
- **Hacker (Cracker)**
- **State-funded spy**
- **Terrorist**

# Hackers

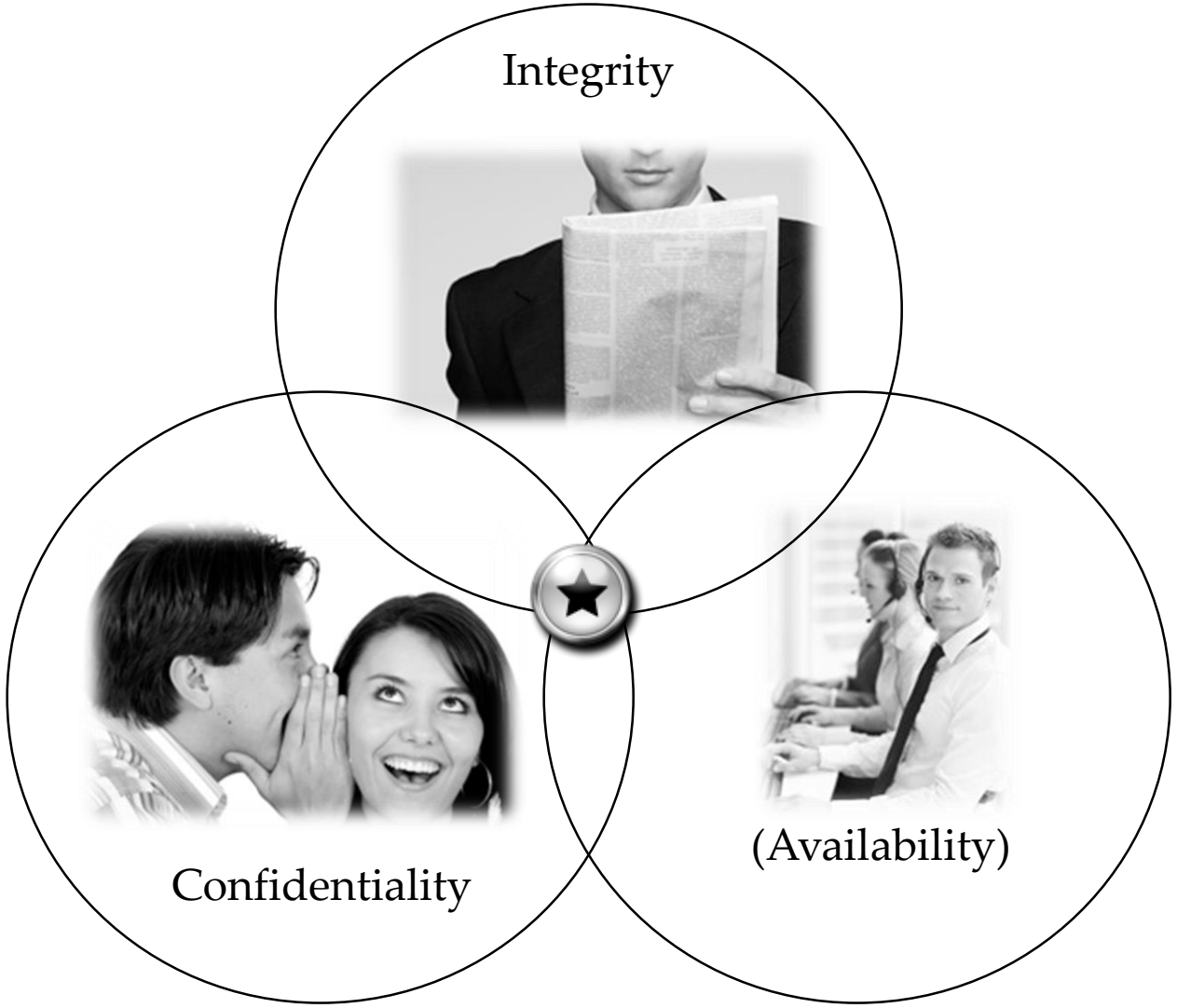
# Ethical hackers

- Professional and ethical values
- Get Out of Jail Free
- Report of the findings
- Respecting privacy
- No crashing tested systems

# Hackers vs malicious users

- Hackers: External, unauthorized
- Malicious users: Internal, authorized

# Threats



# Main types of threats

- Disclosure
- Deception
- Disruption
- Usurpation

# Attacks

- Buffer overflow
- Brute force
- Replay
- Sniffing, man in the middle
- Session hijacking
- Denial of Service
- Phishing
- Malware

# Buffer overflow

```
#include <stdio.h>
#define MAX_IP_LENGTH 15
int main(void) {
  char file_name[] = "ip.txt";
  FILE *fp;
  fp = fopen(file_name, "r");
  char ch;
  int counter = 0;
  char buf[MAX_IP_LENGTH];
  while((ch = fgetc(fp)) != EOF) {
    buf[counter++] = ch;
  }
  buf[counter] = '\0';
  printf("%s\n", buf);
  fclose(fp);
  return 0;
}
```

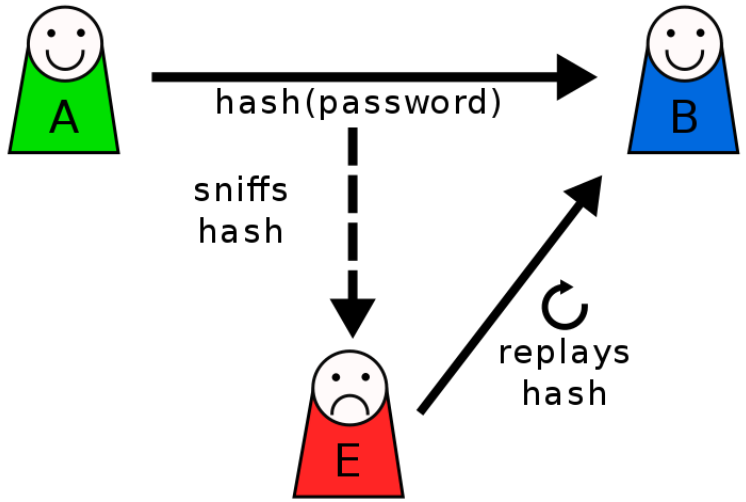
Proper: 255.255.255.255

Fake: 19222222222.16888888.0.1

# Brute force

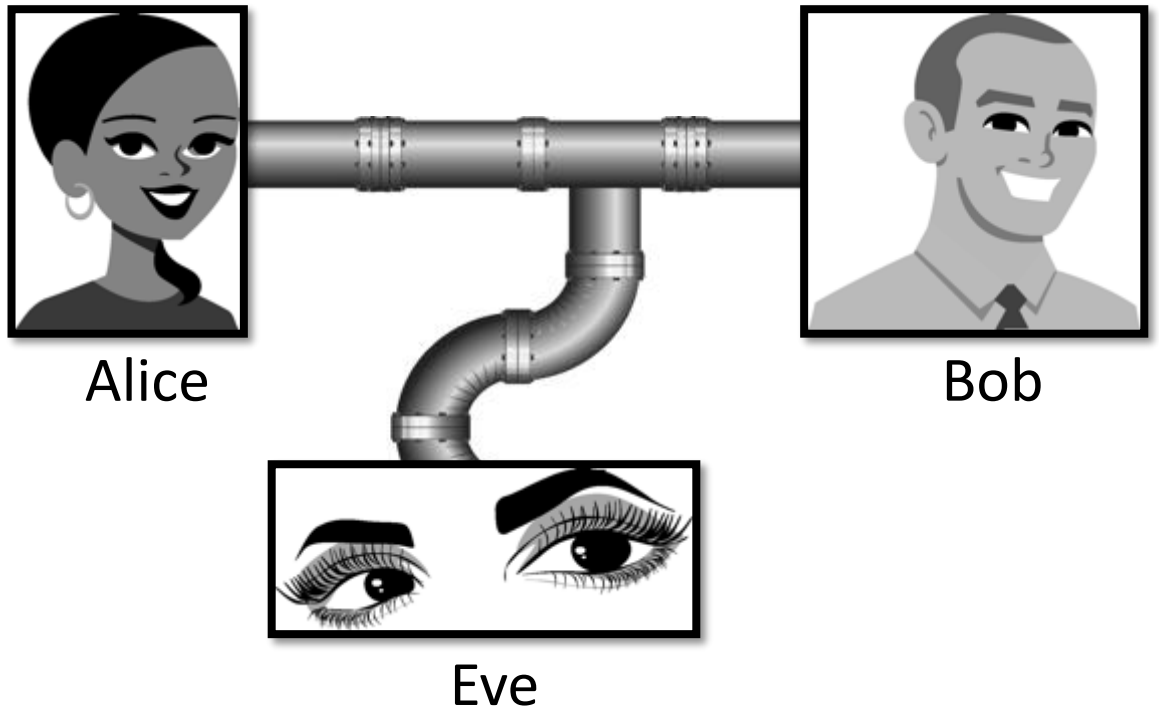
- An attempt to crack a password or username
- Trial and error approach

# Replay attack

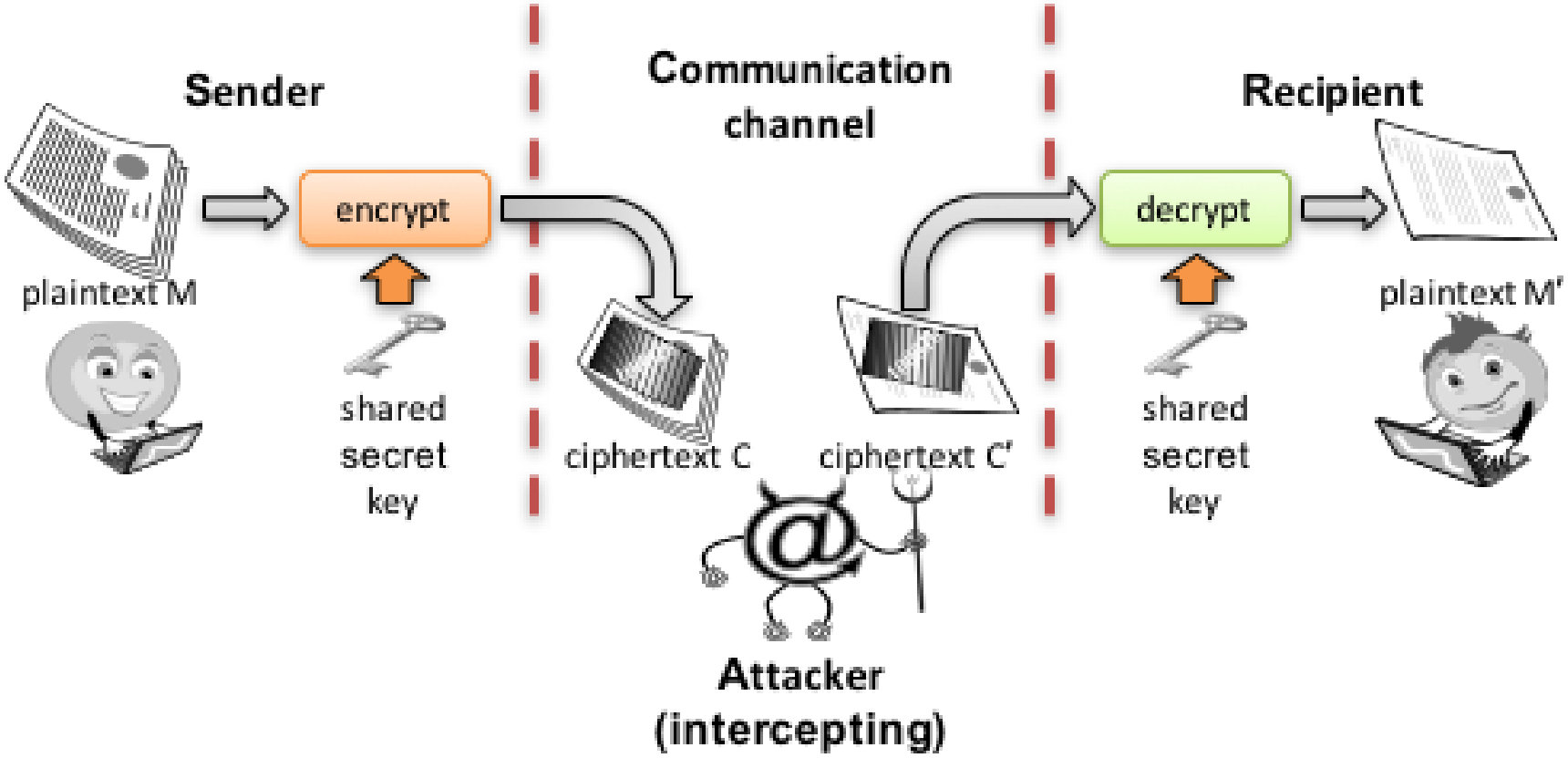


# Sniffing

- Eavesdropping
- Usually passive
- Acquisition of knowledge



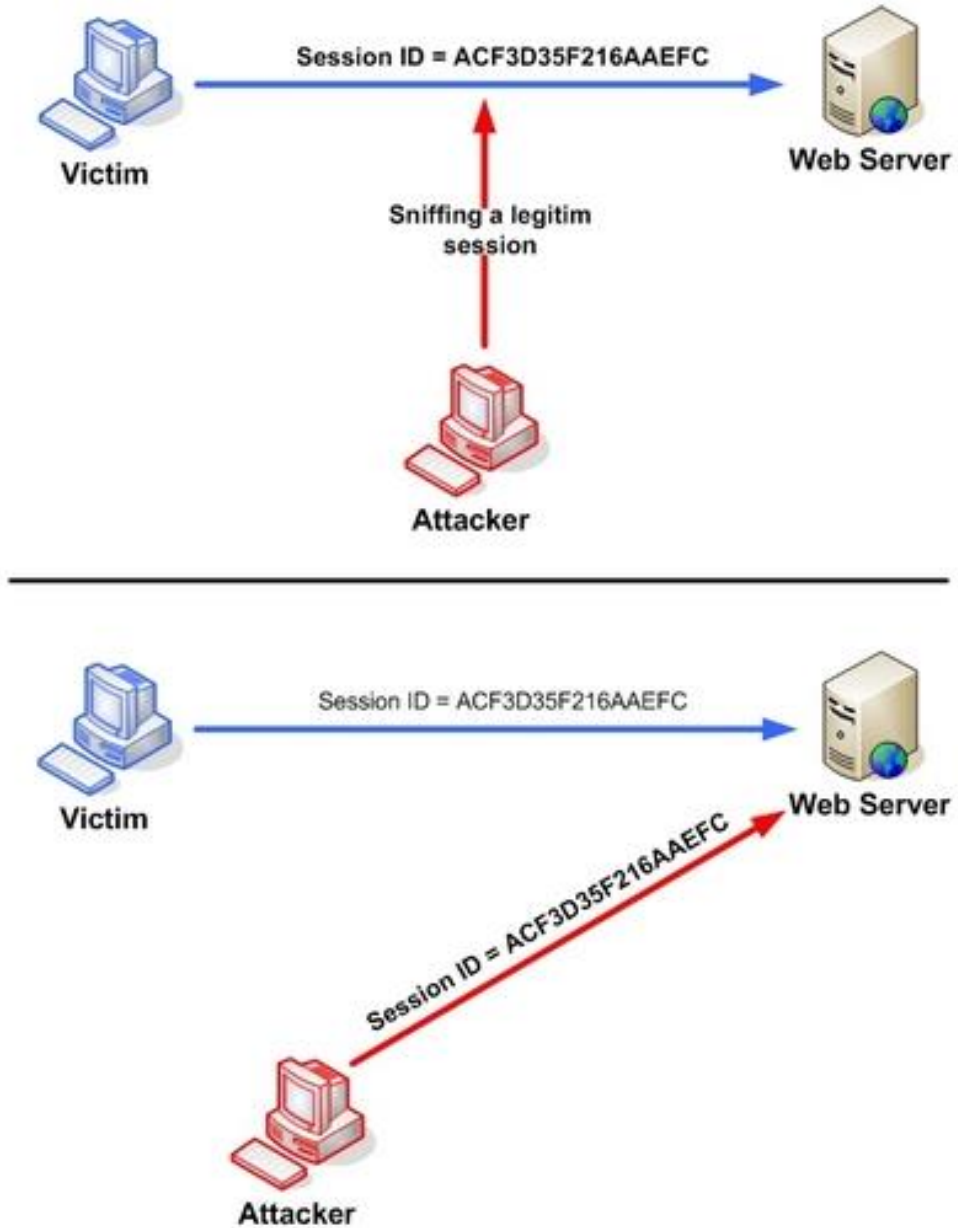
# Sniffing


Sender
Communication channel
Recipient
plaintext M
encrypt
shared secret key
ciphertext C
ciphertext C'
decrypt
shared secret key
plaintext M'
Attacker (intercepting)

# Types of MITM Attacks

- **Rogue Access Point**
- **ARP Spoofing**
- **DNS Spoofing**

# Session hijacking



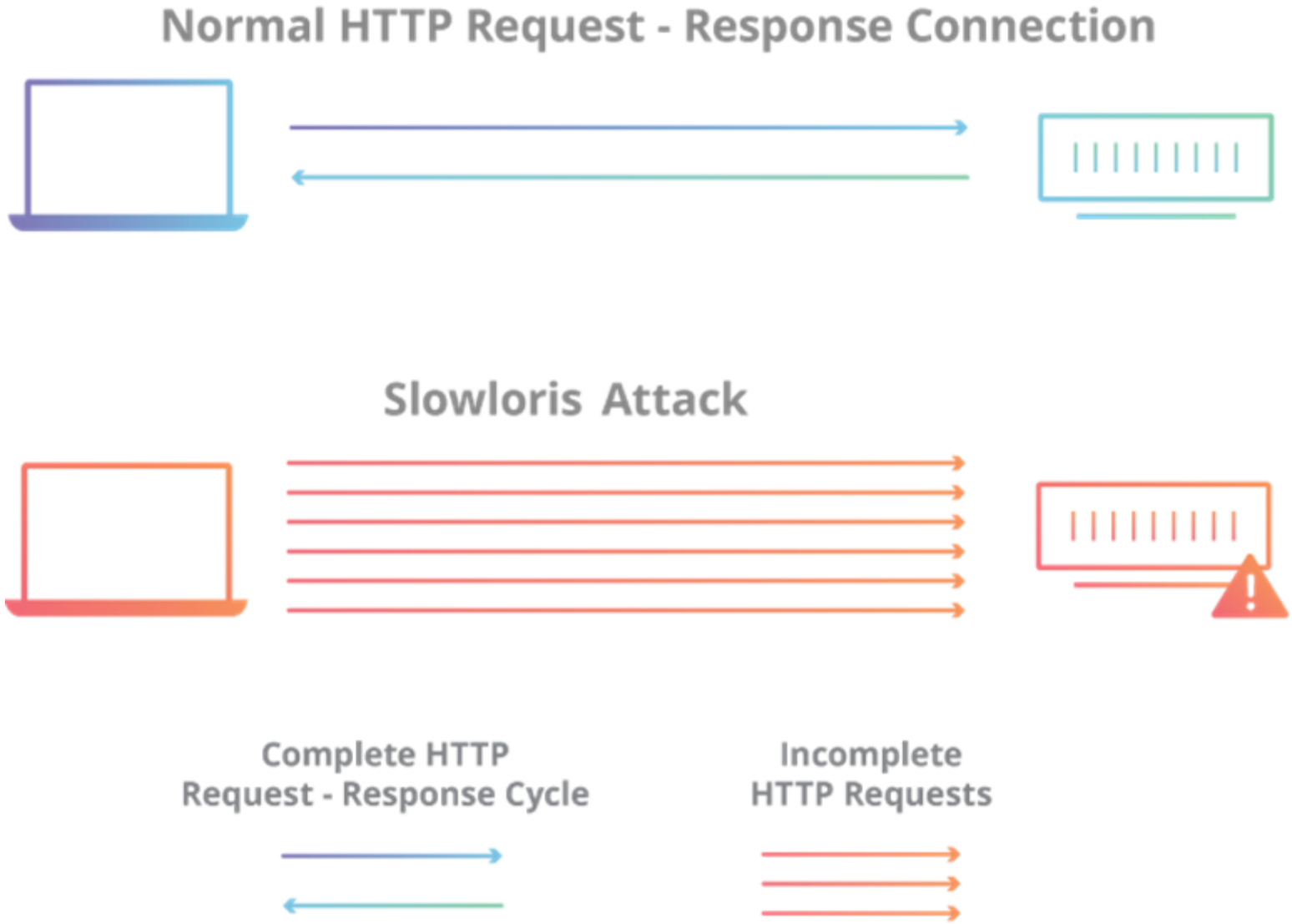
# Denial of service

- Shut down a machine or network
- Can cost the victim a great deal of time and money to handle

# Denial of service

- Flooding services
- Crashing services
- DDoS

# Denial of service



# Denial of service

## How to mitigate?

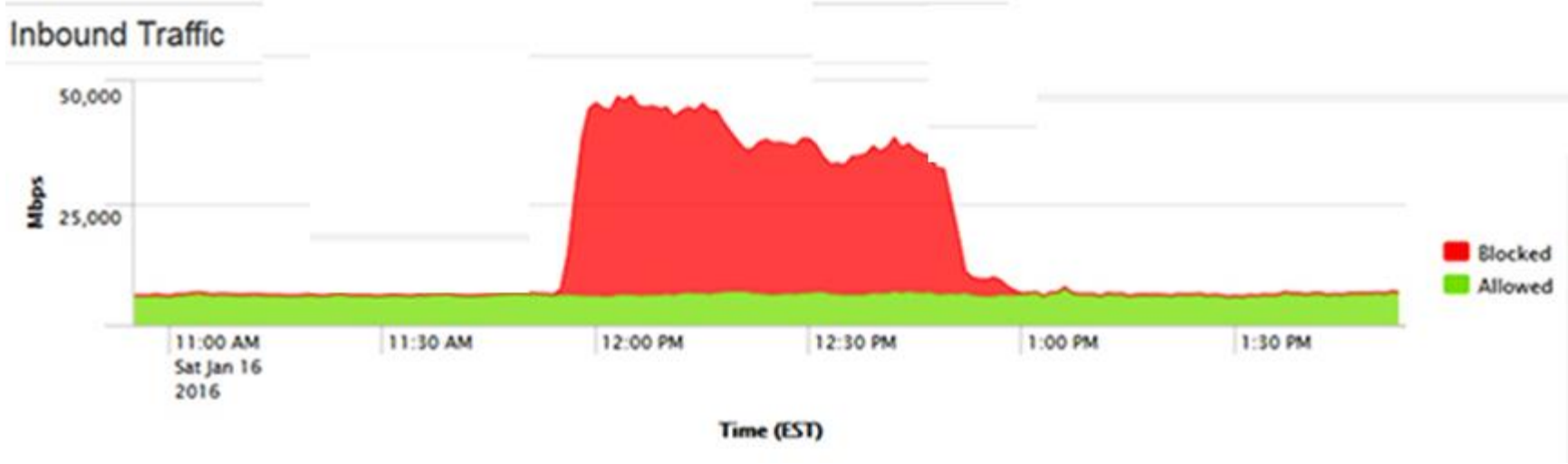
- **Increase server availability**
- **Rate limit incoming requests**
- **Cloud-based protection**

# Denial of service

## DDoS Types

- volumetric (bps---bits per second)

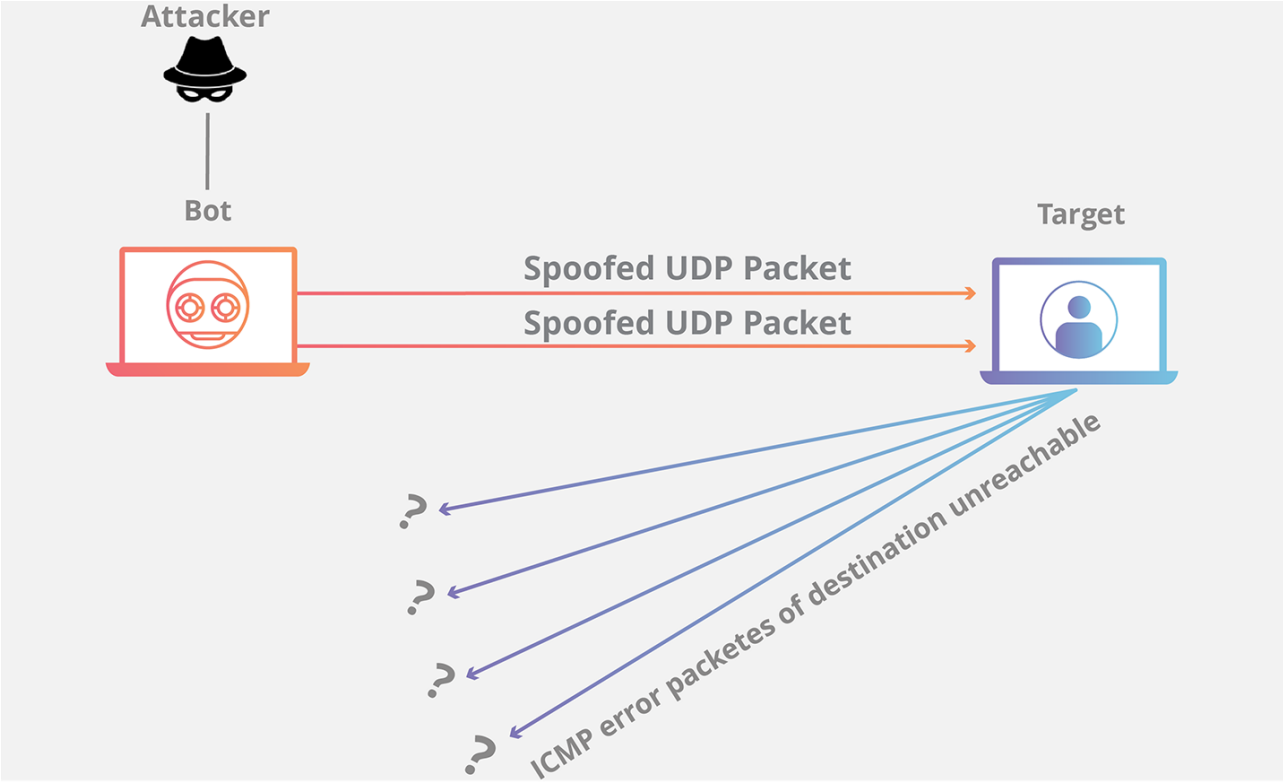
  - DSL routers, surveillance cameras, and IoT devices can be used




- protocol (pps---packets per second) : OSI Layer 3 or Layer 4
- application layer(rps---requests per second): OSI Layer 7

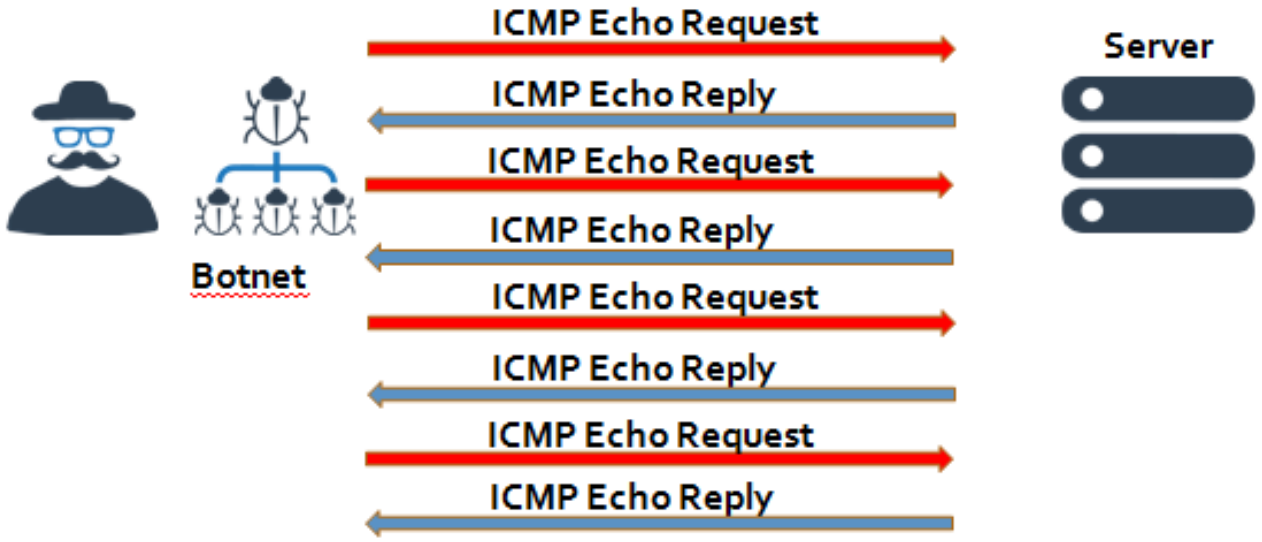
# Denial of service

## UDP Flood (Vol.)



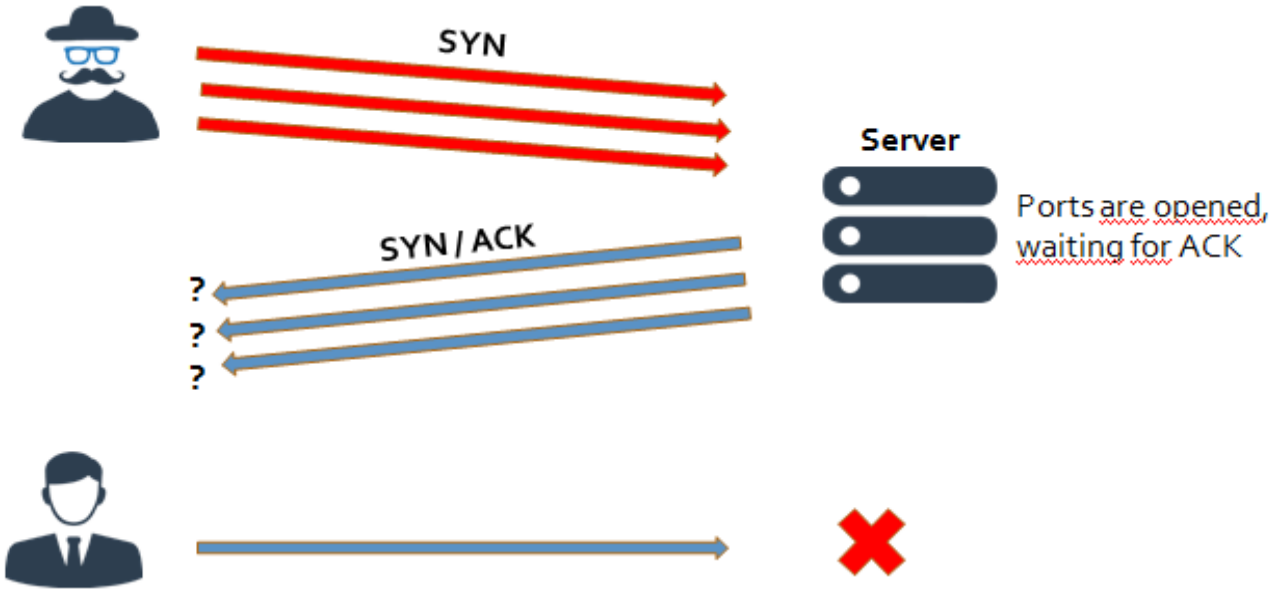
# Denial of service

## ICMP (Ping) Flood (Vol.)



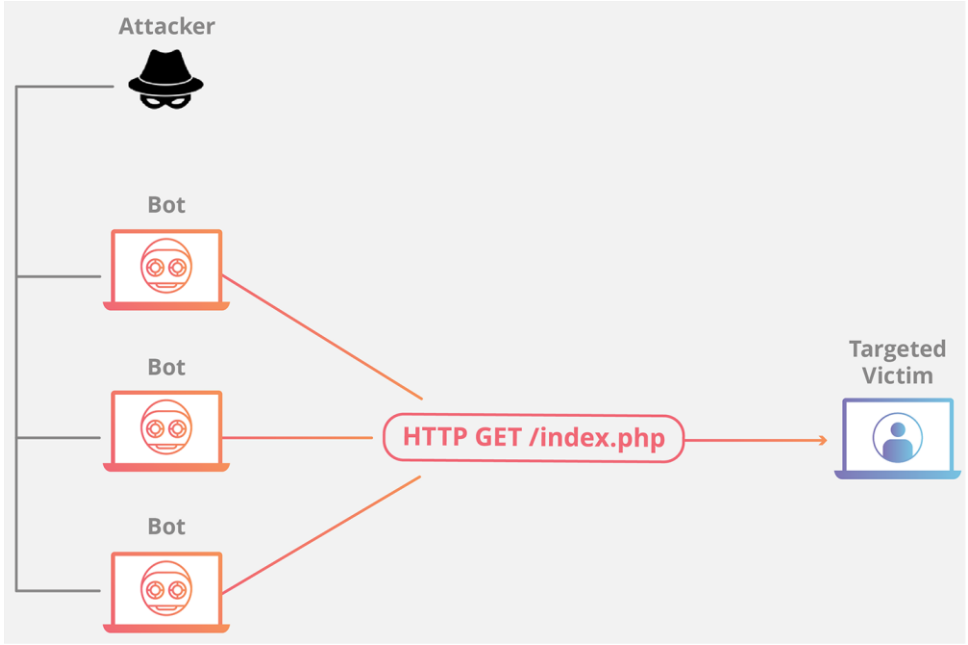
# Denial of service

## Syn Flood (Protocol)



# Denial of service

## HTTP Flood (App. Layer)



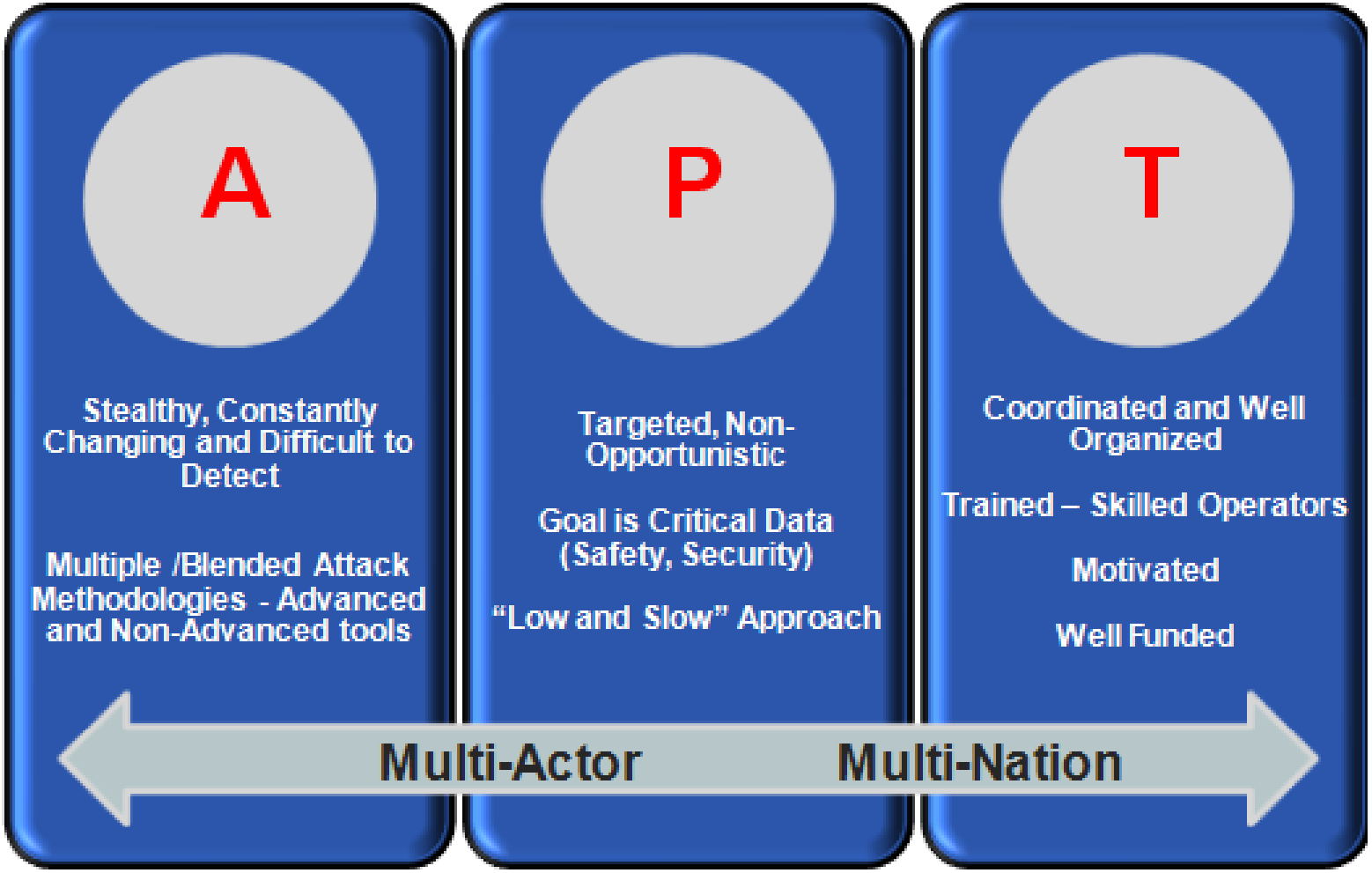
# Phishing

- Social engineering
- Used to steal data
- Tricky email, instant message, or text message
- Clicking a malicious link

# Malware

- Virus
- Worm
- Trojan
- Ransomware

# APT


A
Stealthy, Constantly Changing and Difficult to Detect
Multiple /Blended Attack Methodologies - Advanced and Non-Advanced tools
P
Targeted, Non-Opportunistic
Goal is Critical Data (Safety, Security)
"Low and Slow" Approach
T
Coordinated and Well Organized
Trained – Skilled Operators
Motivated
Well Funded
Multi-Actor
Multi-Nation

# APT


**1. infection**
Stuxnet enters a system via a USB stick and proceeds to infect all machines running Microsoft Windows. By brandishing a digital certificate that seems to show that it comes from a reliable company, the worm is able to evade automated-detection systems.

**2. search**
Stuxnet then checks whether a given machine is part of the targeted industrial control system made by Siemens. Such systems are deployed in Iran to run high-speed centrifuges that help to enrich nuclear fuel.

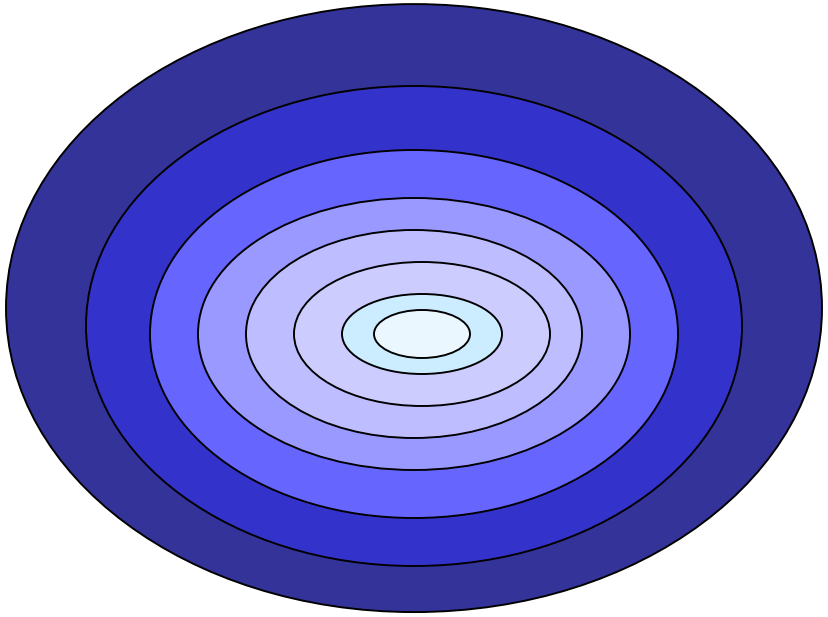
**3. update**
If the system isn't a target, Stuxnet does nothing; if it is, the worm attempts to access the Internet and download a more recent version of itself.

**4. compromise**
The worm then compromises the target system's logic controllers, exploiting "zero day" vulnerabilities-software weaknesses that haven't been identified by security experts.

**5. control**
In the beginning, Stuxnet spies on the operations of the targeted system. Then it uses the information it has gathered to take control of the centrifuges, making them spin themselves to failure.

**6. deceive and destroy**
Meanwhile, it provides false feedback to outside controllers, ensuring that they won't know what's going wrong until it's too late to do anything about it.

# Security: Defense in Depth



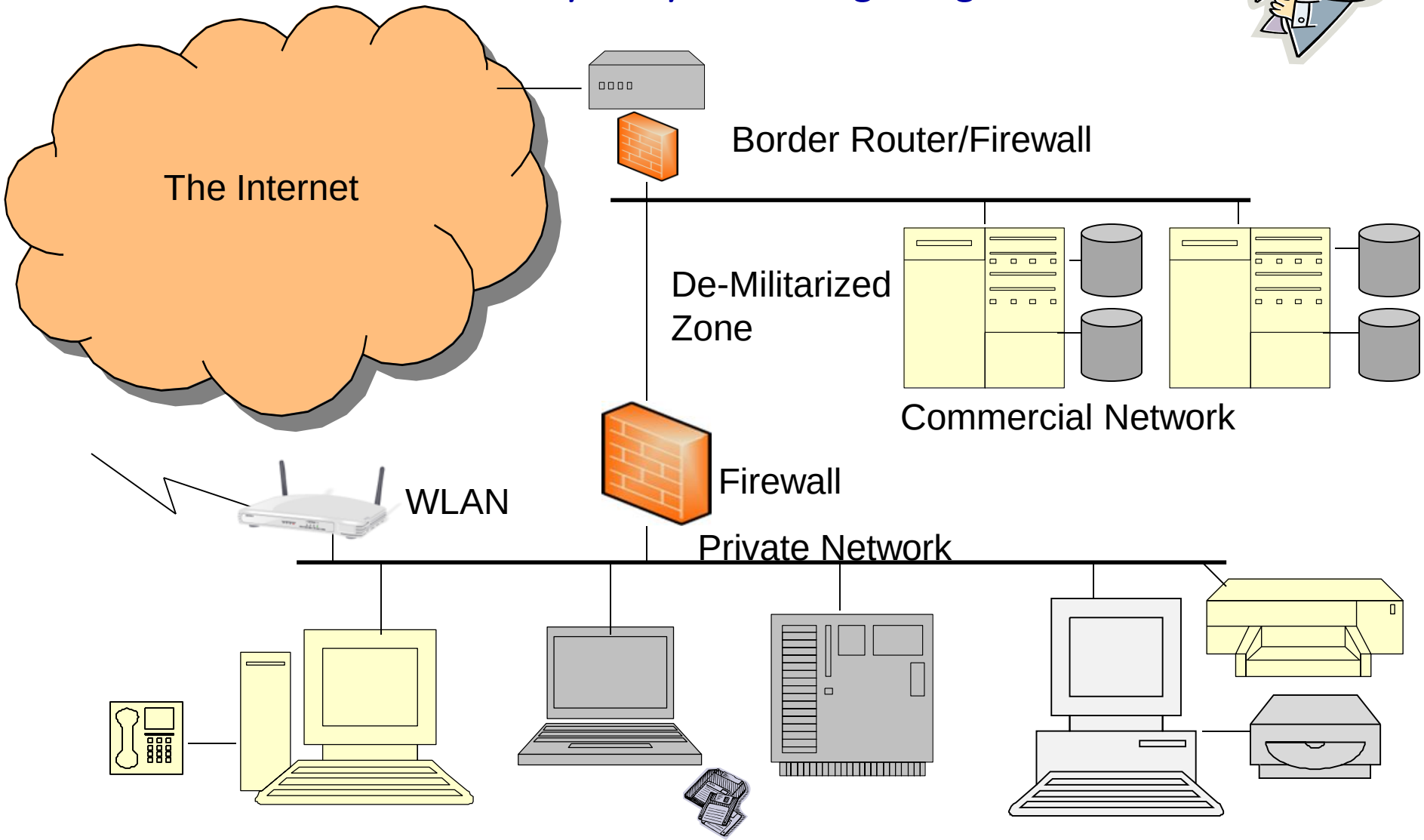
**Border Router**
**Perimeter firewall**
**Internal firewall**
**Intrusion Detection System**
**Policies & Procedures & Audits**
**Authentication**
**Access Controls**

# Bastion Host

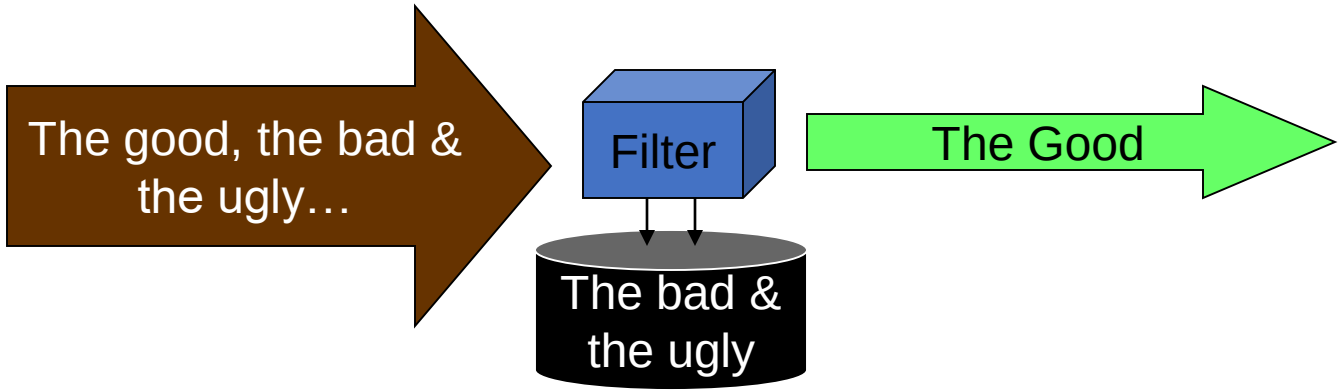
* Computer fortified against attackers
* Applications turned off
* Operating system patched
* Security configuration tightened

# Attacking the Network

## What ways do you see of getting in?



# Filters: Firewalls & Routers




Route Filter:  Verifies source/destination IP addresses

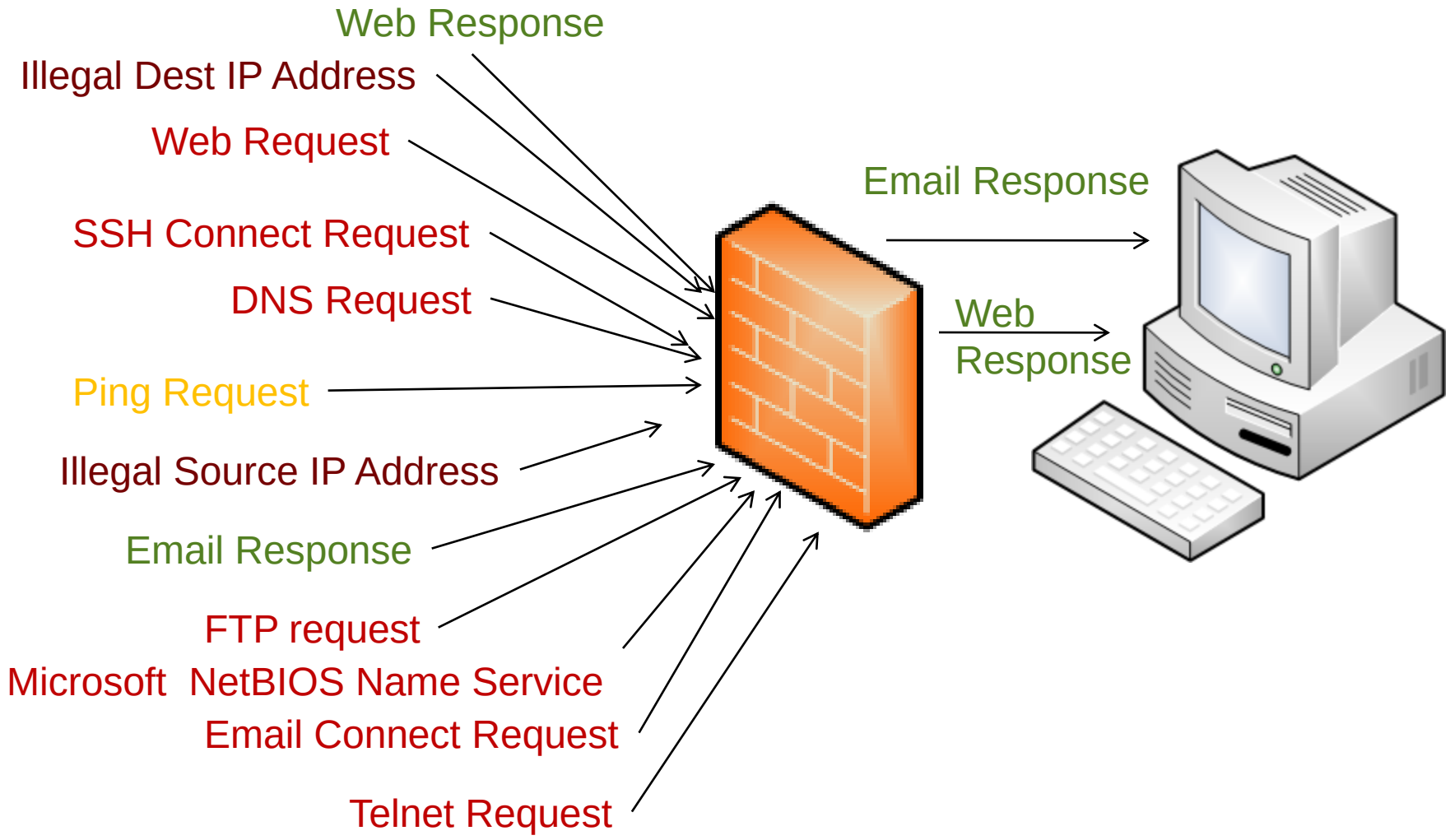
Packet Filter: Scans headers of packets

Content Filter: Scans contents of packet (e.g., IPS)

Default Deny: Any packet not explicitly permitted is rejected

Fail Safe or Fail Secure

# Packet Filter Firewall



# Firewalls


## Commercial

- Palo Alto Networks
- Check Point
- Fortinet
- Cisco

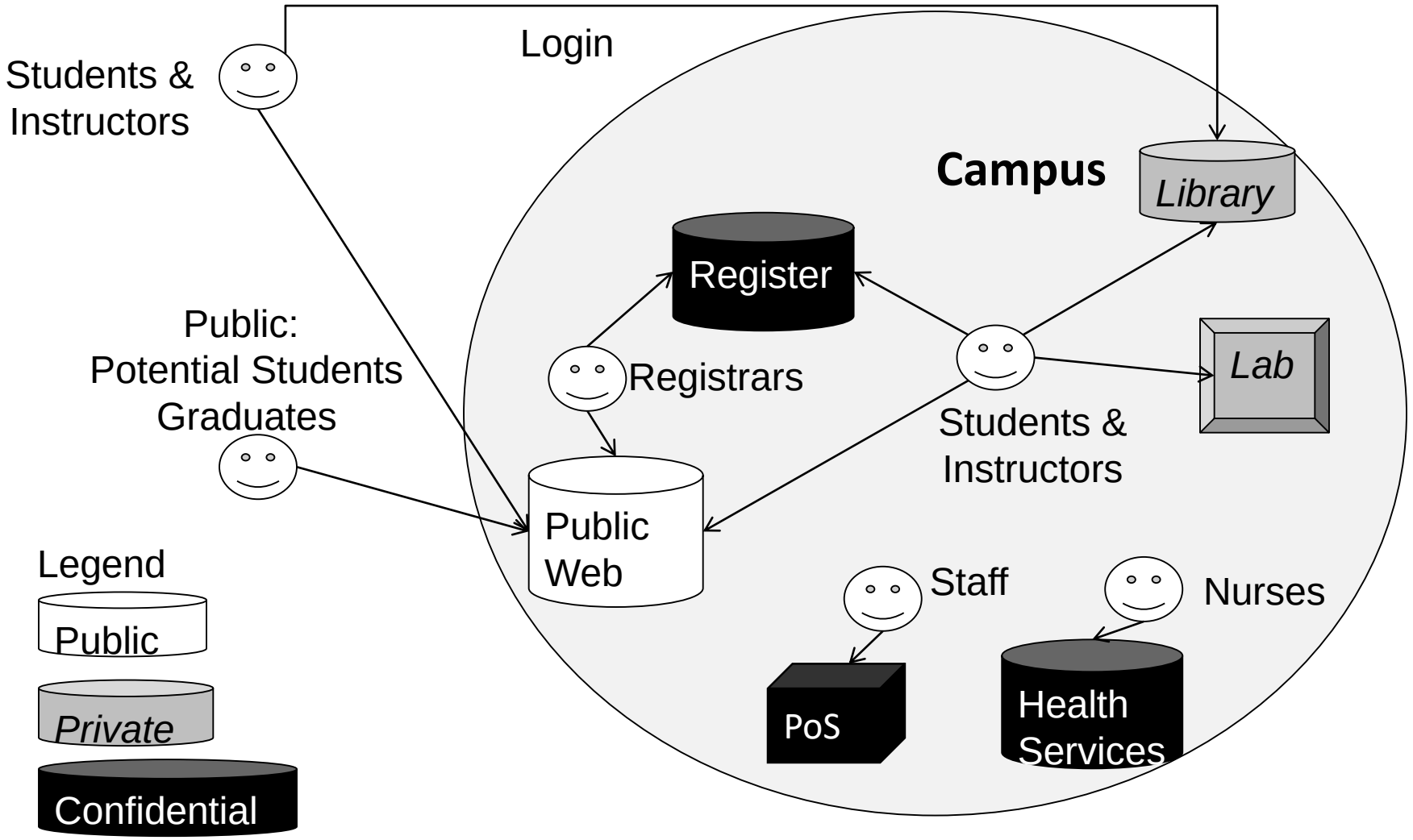
## Open Source

- iptables
- pfSense

# Firewalls – Next Generation

**Packet Filter FW**
**IDS/IPS**
**Application Control**
**Anti Virus**
**Anti Bot**
**SSL inspection**
**DLP**
**…**

# Informal Path of Logical Access



# Determine Services

| Service<br>(e.g., web, sales database) | Source<br>(e.g., home, world, local computer) |
|---|---|
| Registration | Registrars: On campus |
| Library databases | On campus students and staff.<br>Off-campus requires login |
| Health Services | On campus: nurses office |
| External (Internet) web services | On campus: Campus labs, dorms, faculty offices |

# Allocate Network Zones

| Zone | Services | Zone Description |
|---|---|---|
| Internet | | This zone is external to the organization. |
| DMZ | Web, Email, DNS | This zone houses services that the public are allowed to access in our network. |
| Wireless Network | Wireless local employees | This zone connects wireless/laptop employees/students (and crackers) to our internal network. They have wide access. |
| Private Server Zone | DBs | This zone hosts our student learning databases, faculty servers, and student servers. |
| Confidential Zone | Payment card, health, grades info | This highly-secure zone hosts databases with payment and other confidential (protected by law) information. |
| Private User Zone | Wired staff/ students | This zone hosts our wired/fixed employee/classroom computer terminals. |

# Define Controls

| Zone | Service | Required Controls |
|---|---|---|
| DMZ | Web,<br>Email,<br>DNS | Hacking: Intrusion Prevention System, Monitor alarm logs, Anti-virus software within Email package. |
| Wireless Network | Wireless local users | Confidentiality: WPA2 Encryption<br>Authentication: WPA2 Authentication |
| Private Server Zone | Classroom software, Faculty & student storage. | Confidentiality: Secure Web (HTTPS), Secure Protocols (SSH, SFTP).<br>Authentication: Single Sign-on through Radius<br>Hacking: Monitor alarm logs |

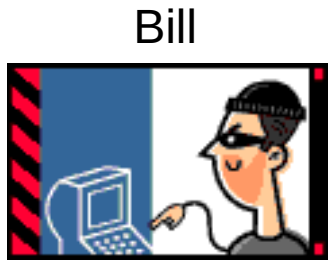
# Data Privacy

Confidentiality: Unauthorized parties cannot access information
(->Secret Key Encryption)

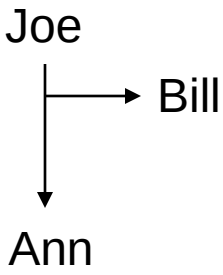
Authenticity: Ensures claimed sender = actual sender.
(->Public Key Encryption)

Integrity: Ensures the message is not modified in transmission.
(->Hashing)

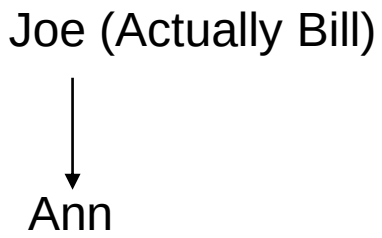
Nonrepudiation: Ensures sender cannot later deny sending message.
(->Digital Signature)



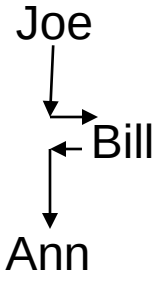

**Confidentiality**



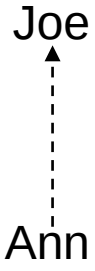

**Authenticity**




**Integrity**




**Non-Repudiation**



# Confidentiality:
# Encryption – Secret Key
## Examples: DES, AES


Sender, Receiver have IDENTICAL keys

Plaintext = Decrypt($K_{secret}$, Encrypt($K_{secret}$,Plaintext))

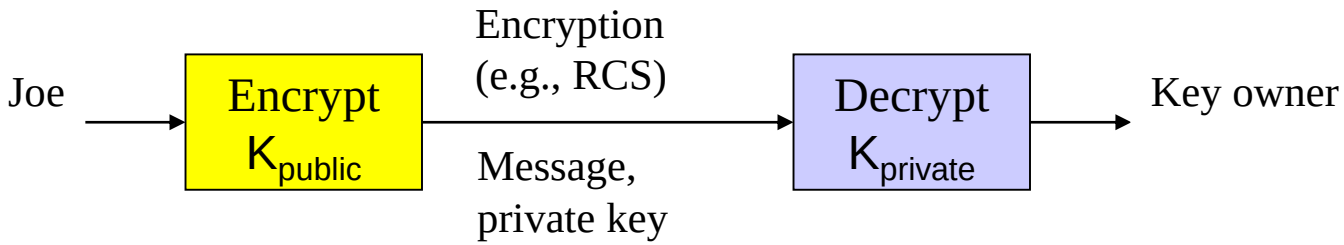
## Confidentiality, Authentication, Non-Repudiation

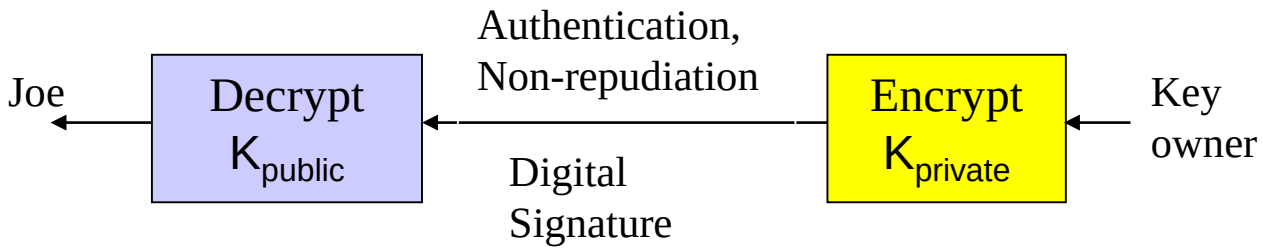
# Public Key Encryption

## Examples: RSA, ECC, Quantum

Sender, Receiver have Complimentary Keys

Plaintext = Decrypt($k_{PRIV}$, Encrypt($k_{PUB}$,Plaintext))

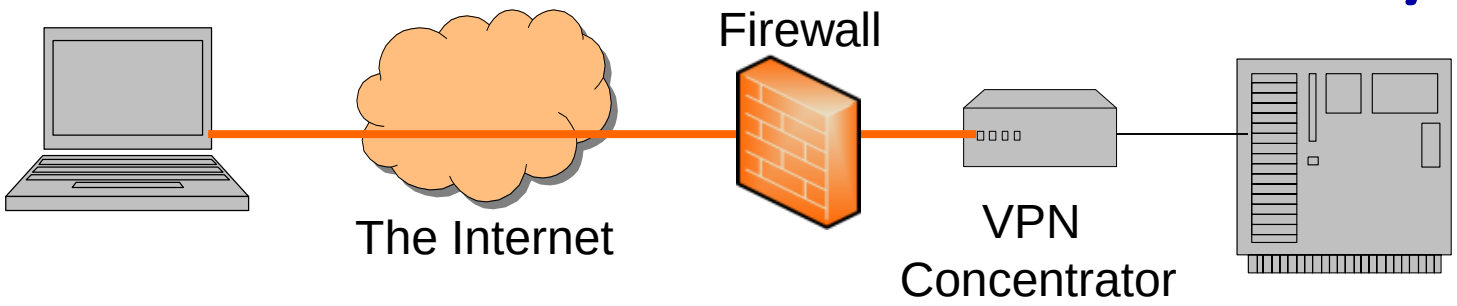



Plaintext = Decrypt($k_{PUB}$, Encrypt($k_{PRIV}$,Plaintext))

## Confidentiality:

# Remote Access Security




Virtual Private Network (VPN) often implemented with IPSec

Can authenticate and encrypt data through Internet (red line)

Easy to use and inexpensive

Difficult to troubleshoot

Susceptible to malicious software and unauthorized actions

Often router or firewall is the VPN endpoint

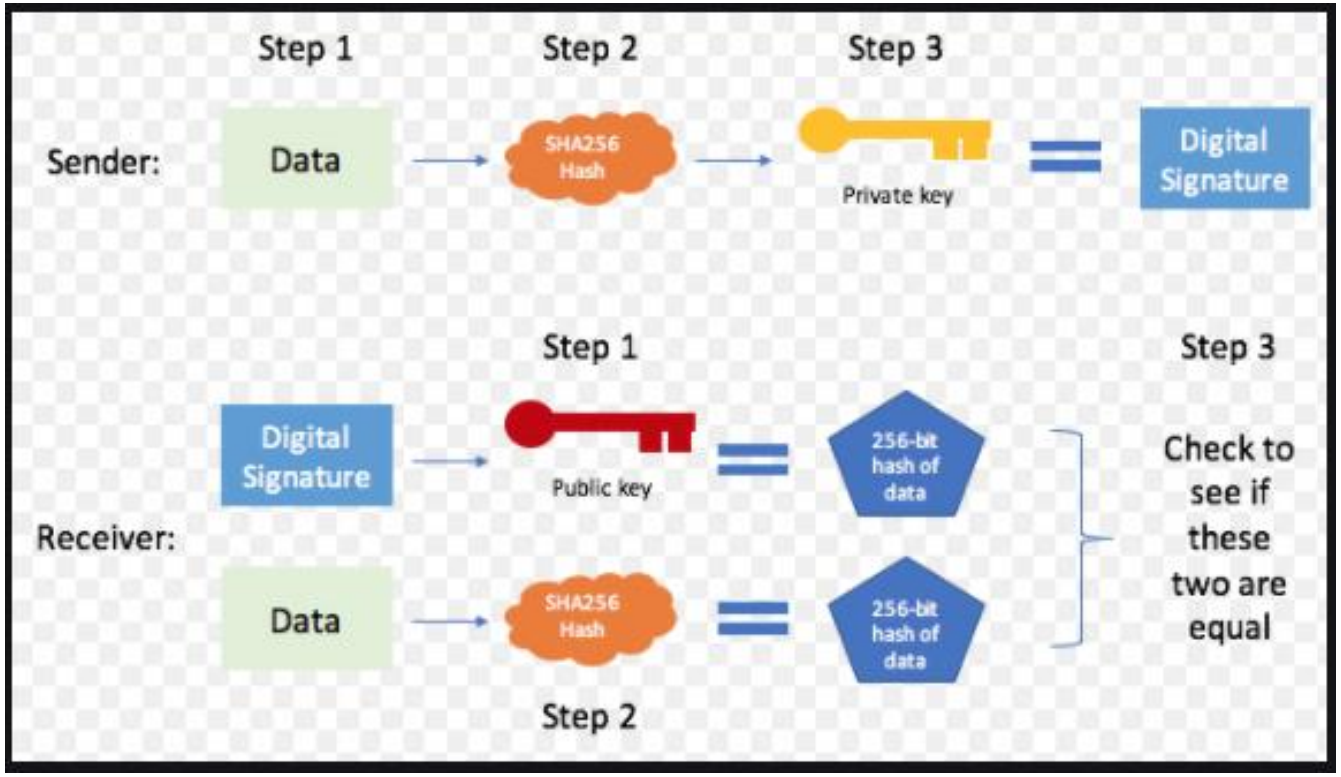
# Integrity:
# Hash Functions
## Examples: SHA-2, SHA-3

Ensures the message was not modified during transmission


H = Hash Algorithm

H=Hashed Value

# Non-Repudiation:
# Digital Signature

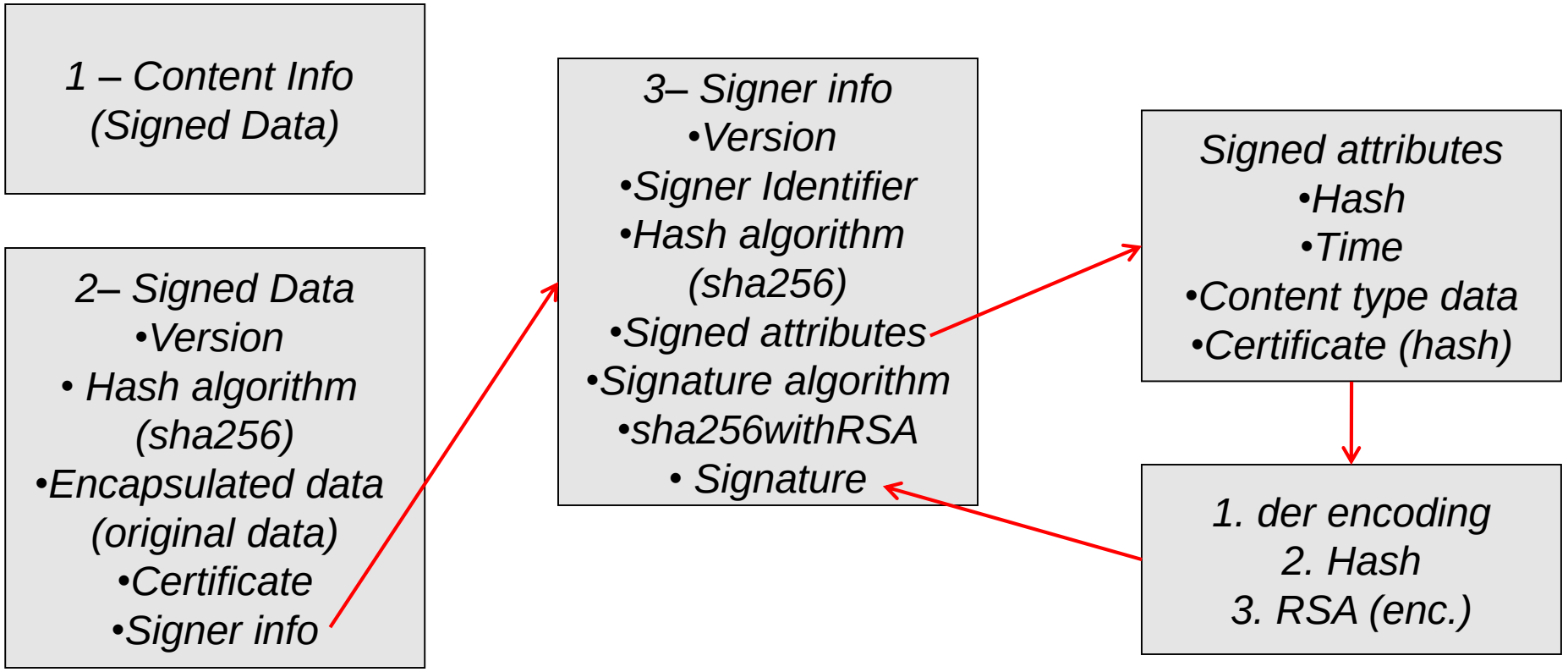



Public key algorithm

Verifies integrity of data

Verifies identity of sender: non-repudiation

# Non-Repudiation:
# Digital Signature



Authentication:

# Public Key Infrastructure (PKI)

7. Tom confirms Sue's DS

**Tom**

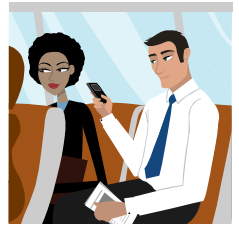
5. Tom requests Sue's DC →
6. CA sends Sue's DC ←

**Certificate Authority (CA)**

**Digital Certificate**
User: Sue
Public Key: 2456

4. Sue sends Tom message signed with Digital Signature

3. Send approved Digital Certificates



**Sue**

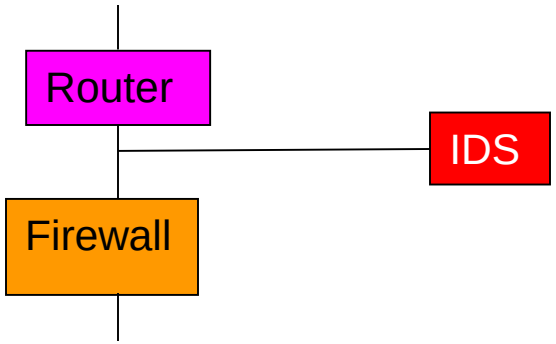
1. Sue registers with CA through RA

Register(Owner, Public Key)

2. **Registration Authority** (RA) verifies owners

## Hacking Defense:

# Intrusion Detection/Prevention Systems (IDS or IPS)

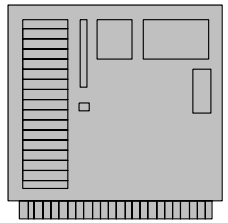



Network IDS=NIDS

Examines packets for attacks

Can find worms, viruses, or defined attacks

Warns administrator of attack



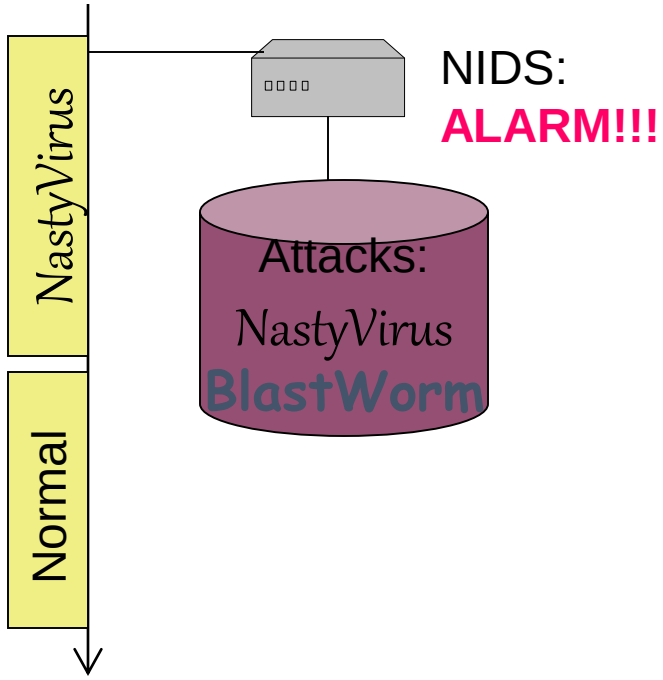
Host IDS=HIDS

Examines actions or resources for attacks

Recognize unusual or inappropriate behavior

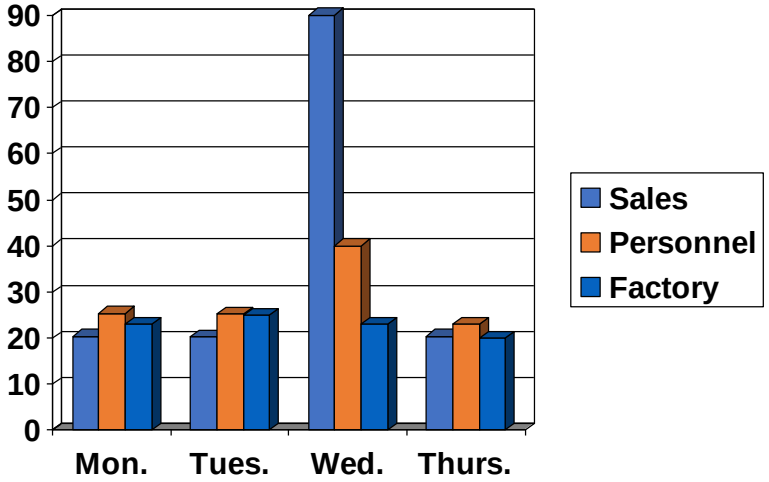
E.g., Detect modification or deletion of special files

# Hacking Defense:
# IDS/IPS Intelligence Systems




Signature-Based:

Specific patterns are recognized as attacks




Statistical-Based:

The expected behavior of the system is understood

If variations occur, they may be attacks (or maybe not)

Neural Networks:

Statistical-Based with self-learning (or artificial intelligence)

Recognizes patterns

# Hacking Defense:
# IDS/IPS


Commercial

- Cisco
- Intel Security (McAfee)
- Trend Micro (Tipping Point)

Open Source

- Snort
- Suricata

# Hacking Defense:
# WAF

**SQL injection**
**Cross-site scripting**
**Local File Inclusion**
**Remote File Inclusion**
**Remote Code Execusion**
**PHP Code Inclusion**
**….**

# Hacking Defense:

# WAF

Commercial

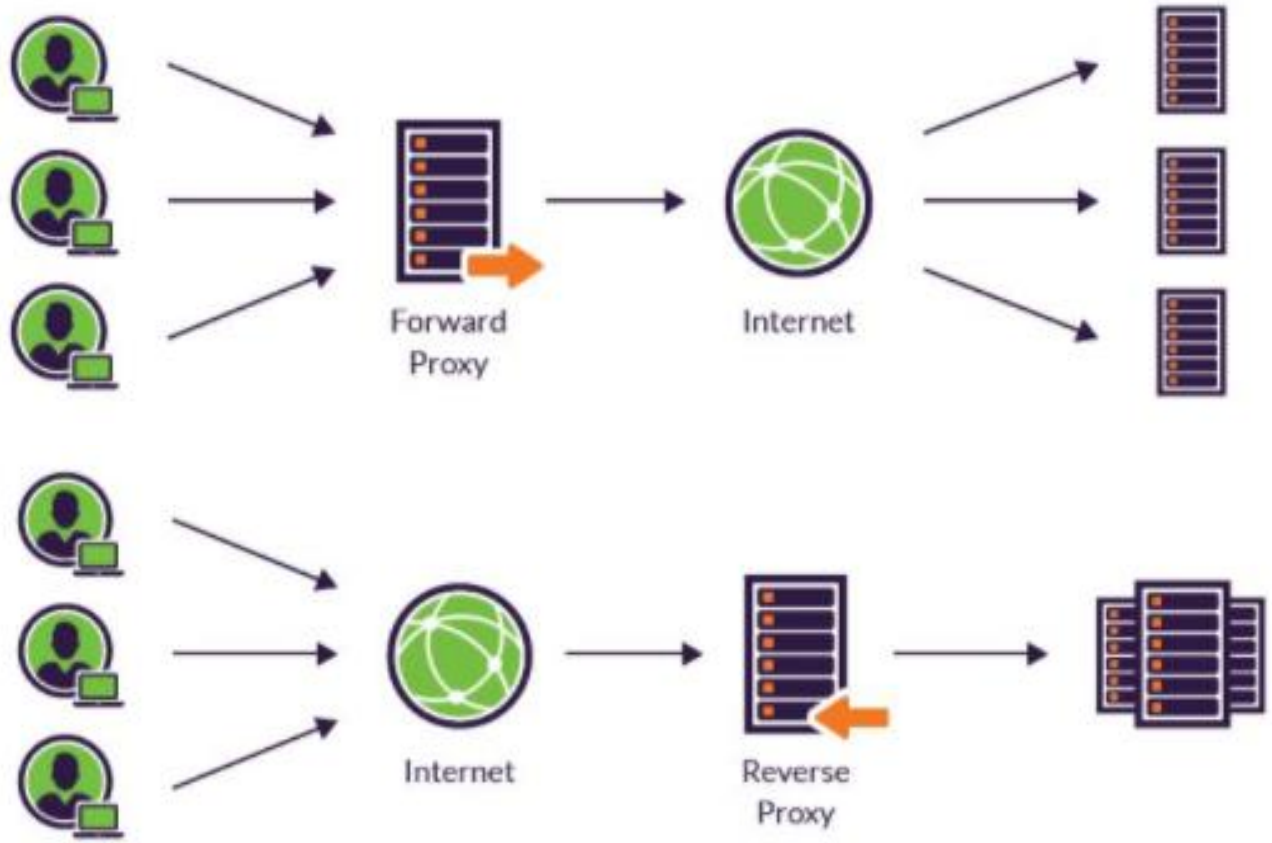
- Imperva
- F5
- Akamai

Open Source

- ModSecurity
- IronBee

Hacking Defense:

# Web Proxy (Web Gateway)



# Hacking Defense:
# Web Proxy

Commercial
- Symantec
- Zscaler

Open Source
- Squid
- Varnish

## Hacking Defense:

# Honeypot & Honeynet

Honeypot: A system with a special software application which appears easy to break into

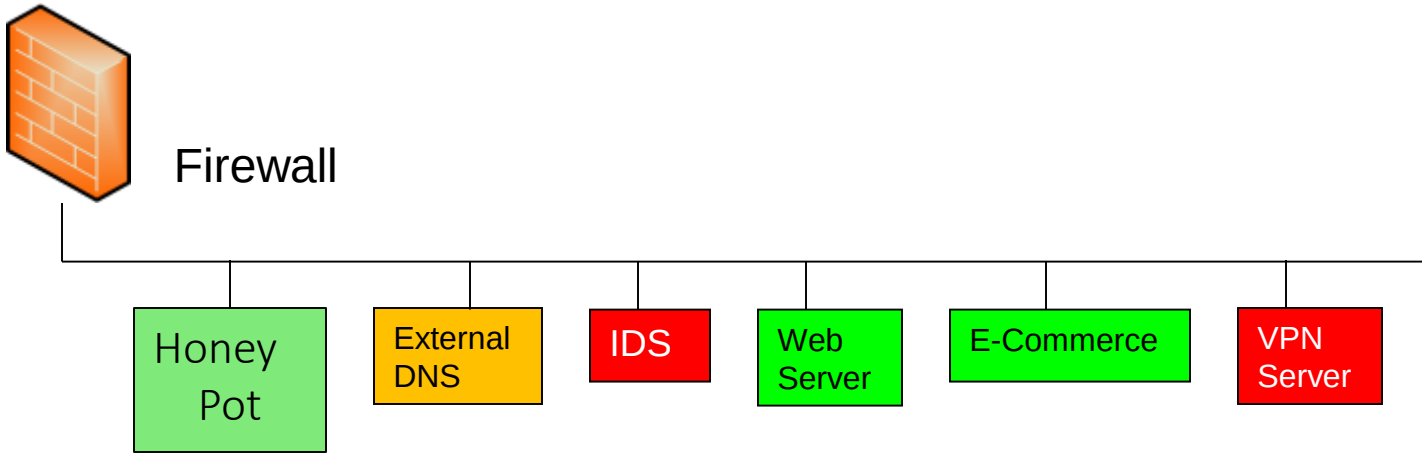
Honeynet: A network which appears easy to break into

Purpose: Catch attackers

All traffic going to honeypot/net is suspicious

If successfully penetrated, can launch further attacks

Must be carefully monitored



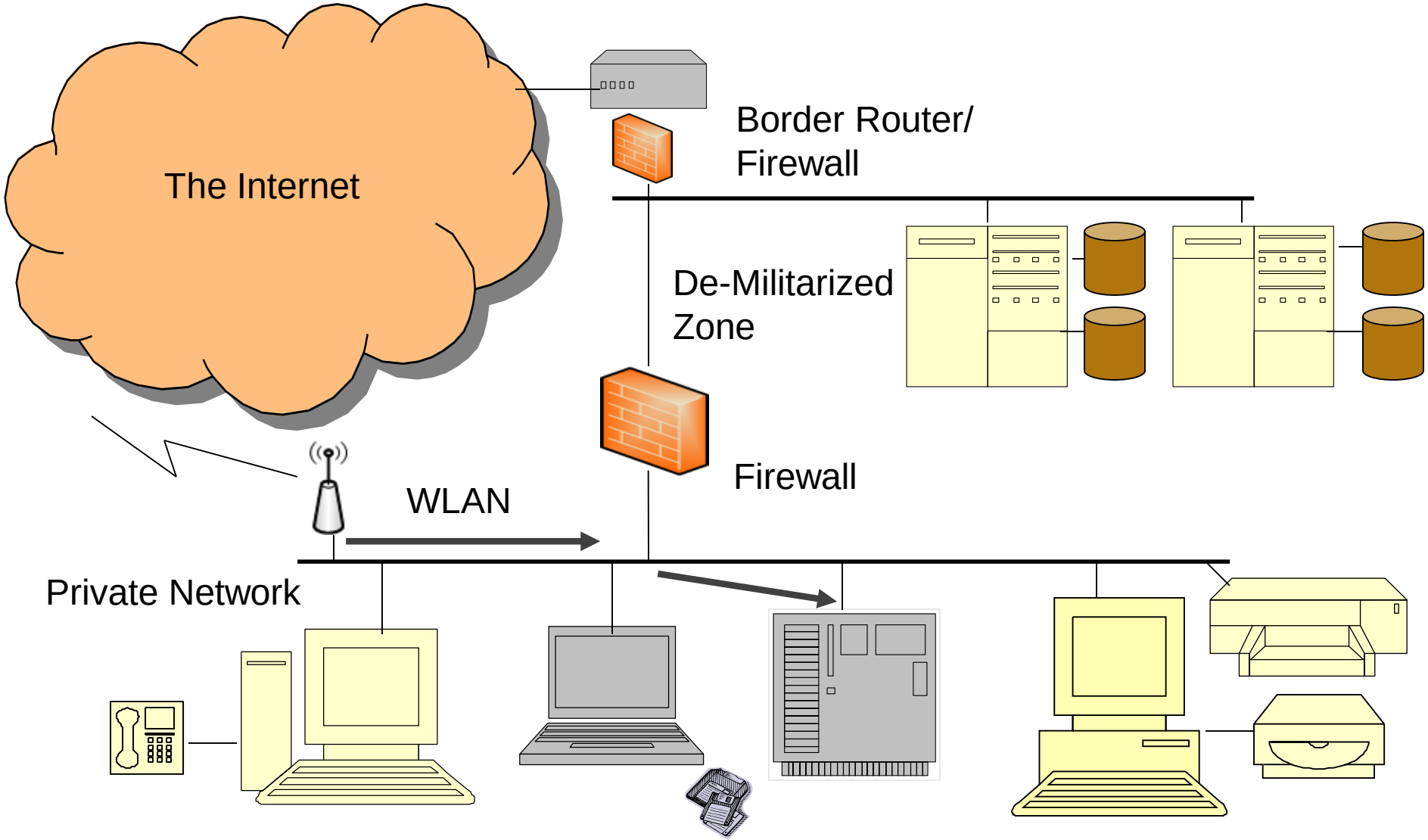
## Hacking Defense:

# Vulnerability Assessment

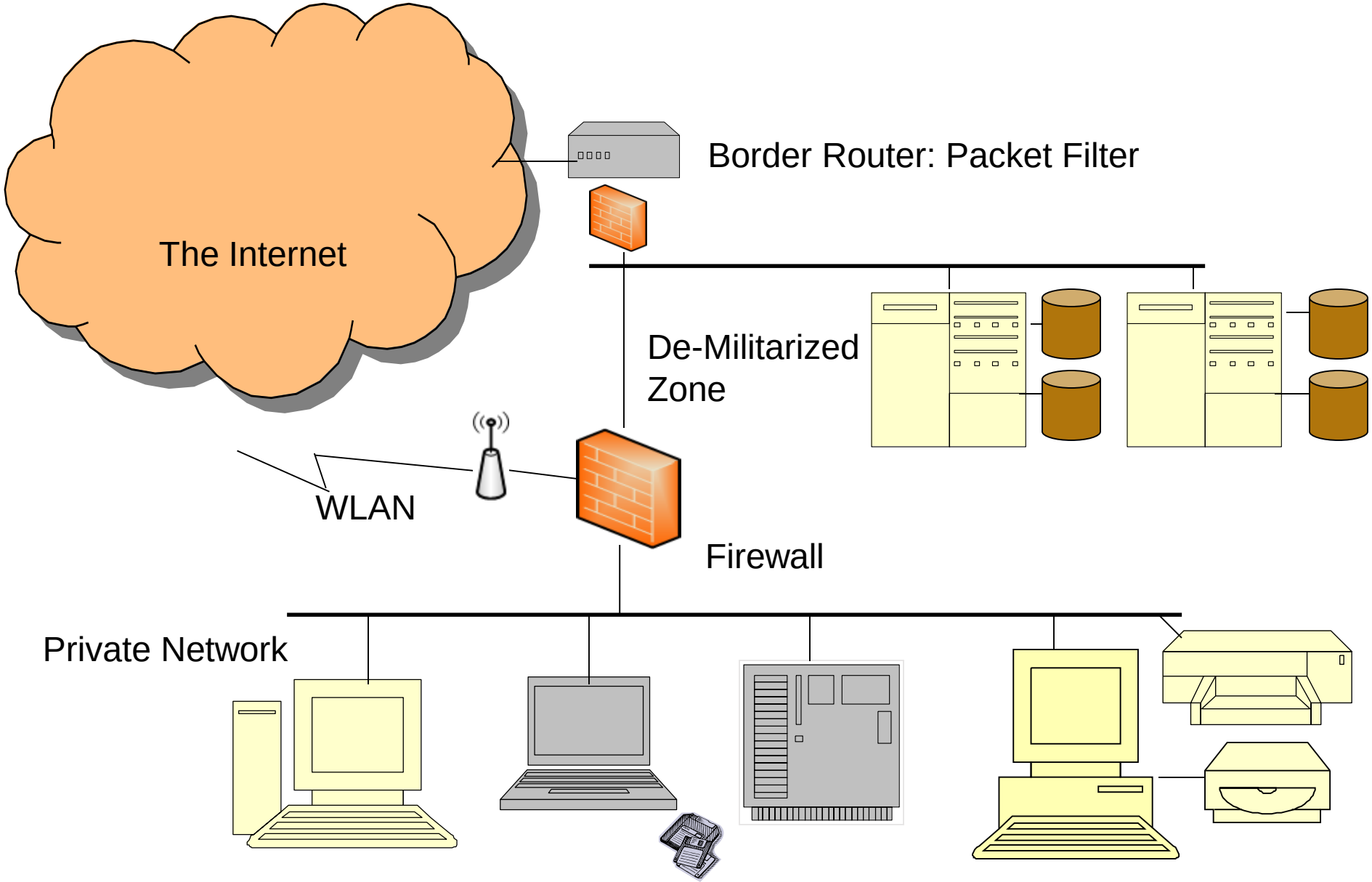
* Scan servers, work stations, and control devices for vulnerabilities
* Open services, patching, configuration weaknesses
* Testing controls for effectiveness
* Adherence to policy & standards
* Penetration testing

# Path of Logical Access

## How would access control be improved?



# Protecting the Network



# End User Security Systems

Host FW
Host IPS
Anti Virus, Endpoint Security Systems
Endpoint Detection and Response (EDR)
DLP
Sandbox
Application Control
Encryption

...