THE UNIVERSITY OF QUEENSLAND
AUSTRALIA
CREATE CHANGE

Securing Business Information

Week 08: Security Technology (Part 1)

- Kerberos, Firewalls, Proxies and the DMZ

Dr Alex Pudmenzky

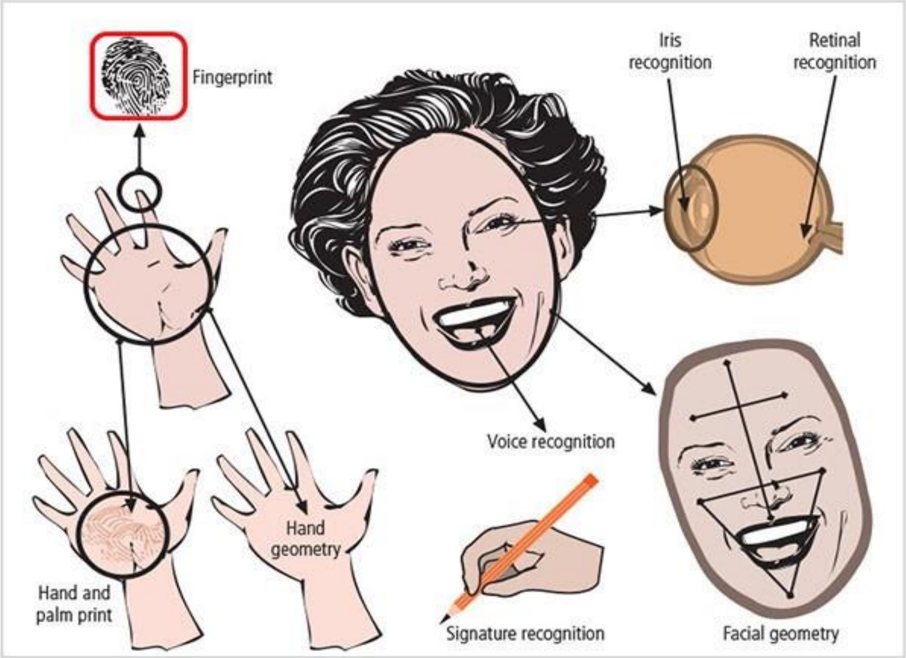
Semester 2, 2024

Overview

- Define **authentication** and explain the three commonly used authentication factors
- Describe the **Kerberos** protocol
- Describe **firewall** principles and the various approaches to firewall implementation
- Discuss the **proxy** firewall approach
- Describe the strategy that enables the business use & benefits of a **demilitarized zone** (DMZ)

Authentication

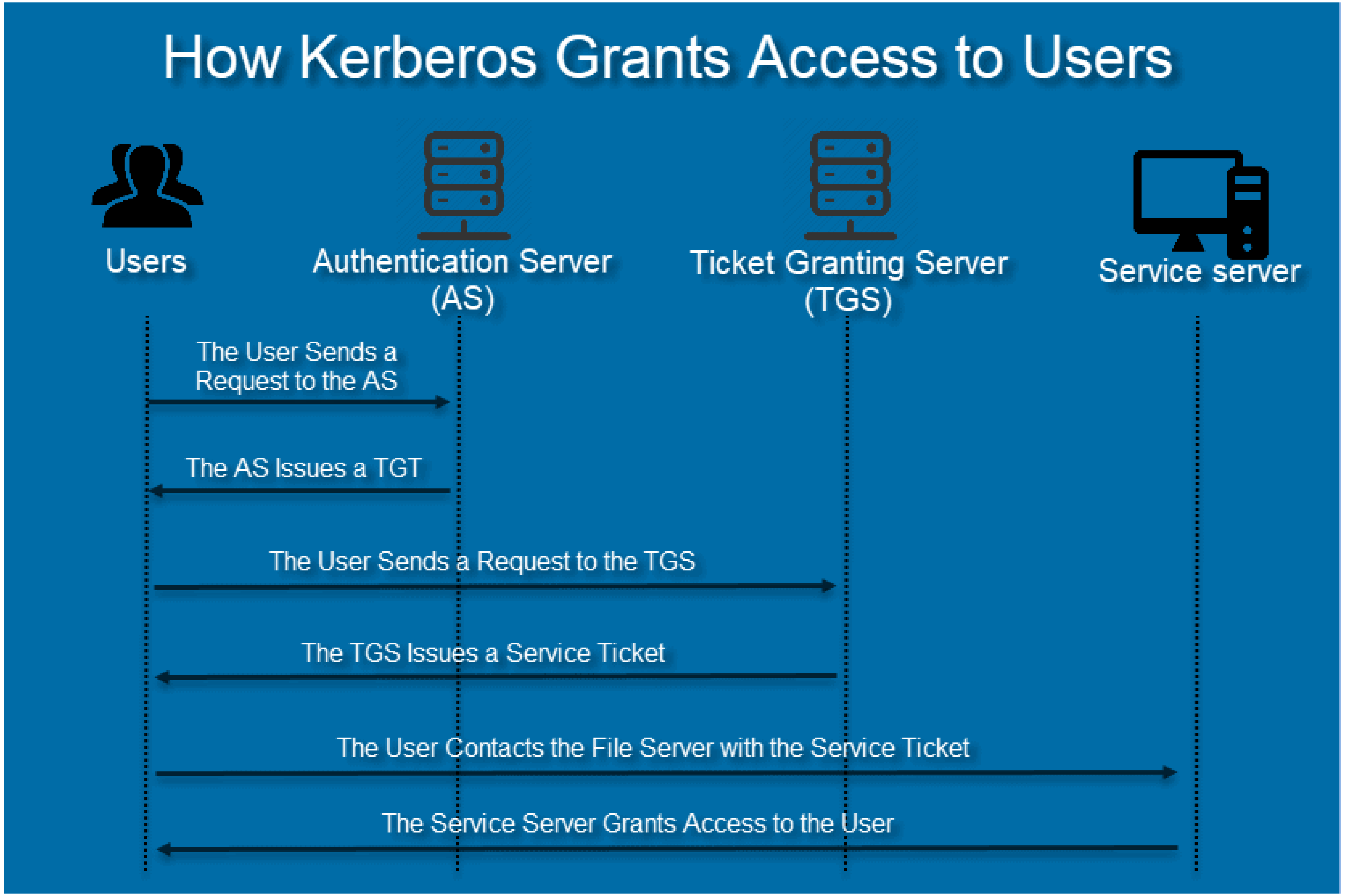
- **Authentication**: the process of validating a supplicant's[1] purported identity
- Authentication factors (from weakest to strongest methods)
  - Something a supplicant knows
    - Password: a private word or combination of characters that only the user should know
    - Passphrase: a sequence of words or other text typically longer than a password
  - Something a supplicant has
    - Smart card: contains a computer chip that can verify and validate information
    - Synchronous tokens
    - Asynchronous tokens
  - Something a supplicant is
    - Relies upon individual characteristics
    - Strong authentication



[1]Supplicant: Someone who's asking for something in a humble manner.

Kerberos guarding Hades

Kerberos – Overview flowchart



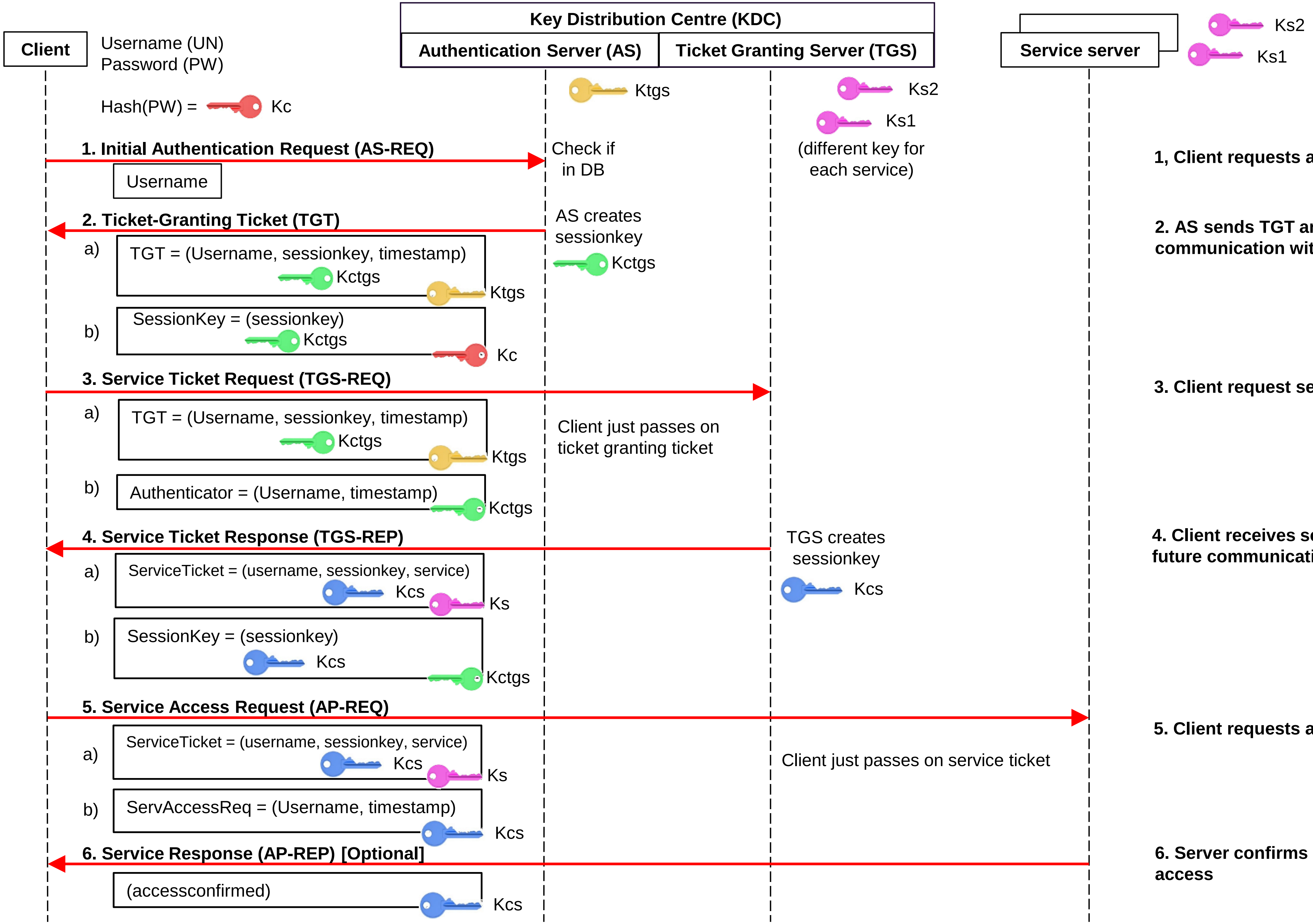
How Kerberos Grants Access to Users

Users
Authentication Server (AS)
Ticket Granting Server (TGS)
Service server
The User Sends a Request to the AS
The AS Issues a TGT
The User Sends a Request to the TGS
The TGS Issues a Service Ticket
The User Contacts the File Server with the Service Ticket
The Service Server Grants Access to the User

https://phoenixnap.com/blog/kerberos-authentication

Kerberos - Overview

- **Secure Third-Party Authentication**: Kerberos provides secure, third-party authentication for both users and services. It ensures **mutual authentication**—both the user and the service authenticate each other during the process, preventing attacks such as impersonation.
- **Symmetric Key Encryption**: Kerberos primarily relies on **symmetric key encryption** to authenticate users and grant access to various network resources. Each user and service shares a secret key with the Key Distribution Center (KDC), which facilitates secure communication.
- **Shared Secret for Authentication**: Authentication in Kerberos is based on the concept of a **shared secret** (symmetric key). This key is used to validate the identities of both clients (users) and servers (services), ensuring that only authorized entities can communicate.
- **Private Key Database**: The KDC maintains a secure **database of private keys**, which includes client and server keys. Client keys are typically derived from hashed passwords, while service keys are preconfigured during the setup phase. These keys are used for encrypting tickets and session keys.



Key Distribution Centre (KDC)

Client | Authentication Server (AS) | Ticket Granting Server (TGS) | Service server

Username (UN)
Password (PW)

Hash(PW) = Kc

Ktgs

Ks2
Ks1
(different key for each service)

Ks2
Ks1

Print Server, File Server, Web Server, Database Server, Email Server, Application Server, etc.

1. Initial Authentication Request (AS-REQ)

Username

Check if in DB

2. Ticket-Granting Ticket (TGT)

AS creates sessionkey Kctgs

a) TGT = (Username, sessionkey, timestamp) Kctgs Ktgs

b) SessionKey = (sessionkey) Kctgs Kc

3. Service Ticket Request (TGS-REQ)

a) TGT = (Username, sessionkey, timestamp) Kctgs Ktgs

b) Authenticator = (Username, timestamp) Kctgs

Client just passes on ticket granting ticket

4. Service Ticket Response (TGS-REP)

TGS creates sessionkey Kcs

a) ServiceTicket = (username, sessionkey, service) Kcs Ks

b) SessionKey = (sessionkey) Kcs Kctgs

5. Service Access Request (AP-REQ)

a) ServiceTicket = (username, sessionkey, service) Kcs Ks

b) ServAccessReq = (Username, timestamp) Kcs

Client just passes on service ticket

6. Service Response (AP-REP) [Optional]

(accessconfirmed) Kcs

1, Client requests authentication

2. AS sends TGT and session key for future communication with TGS

3. Client request service ticket

4. Client receives service ticket and session key for future communication with service

5. Client requests access to service

6. Server confirms client identity and grants access

Key names explained on next slide...

Names for the keys typically used in the Kerberos protocol:

Kc

Kc (Client Key): The symmetric key derived from the user's password, used to encrypt communication between the client and the Key Distribution Center (KDC) during initial authentication.

Kctgs

Kctgs (Client-TGS Session Key): A session key shared between the client and the Ticket Granting Server (TGS). It is used to encrypt communication between the client and the TGS after the client receives a Ticket-Granting Ticket (TGT).

Ktgs

Ktgs (TGS Secret Key): The symmetric key known only to the TGS and the Authentication Server (AS). It is used to encrypt the Ticket-Granting Ticket (TGT), allowing the TGS to decrypt and validate it.

Ks

Ks (Service Secret Key): The symmetric key specific to each service. It is used to encrypt the service ticket issued by the TGS so that only the service can decrypt it (Ks1, Ks2,...).

Kcs

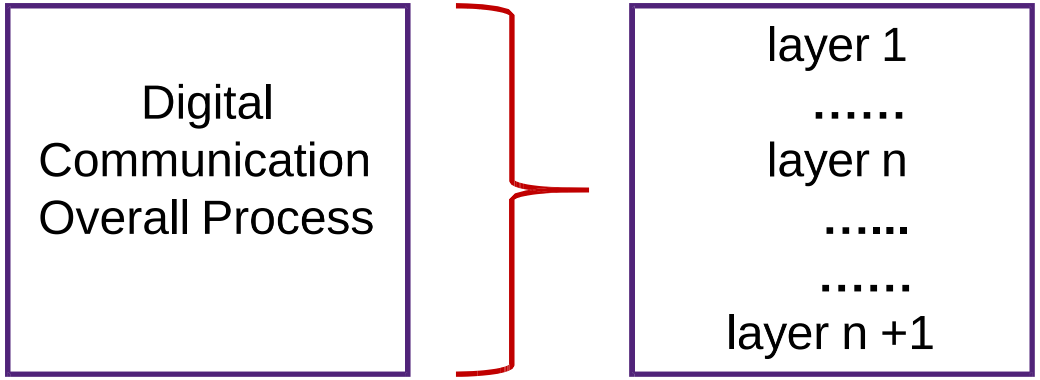
Kcs (Client-Service Session Key): A session key shared between the client and the service. It is used to encrypt communication between the client and the service during their interaction.

More on the topic of a ‘secure network‘

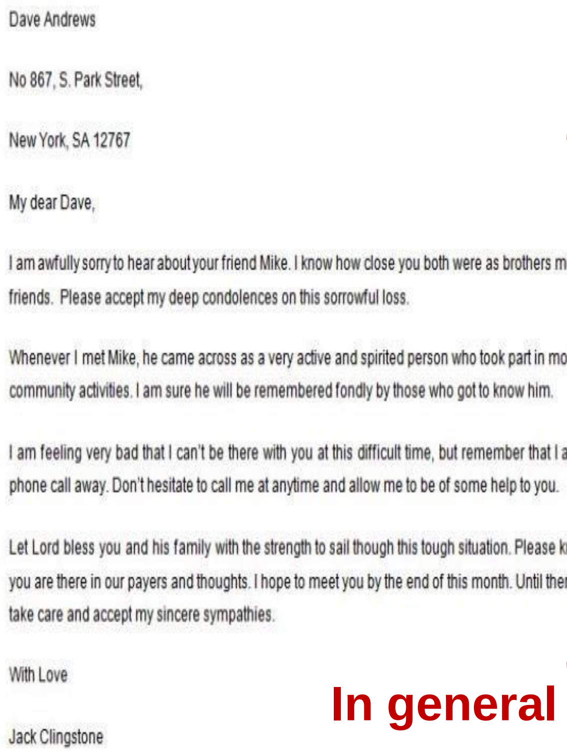
- Describe firewall principles and the various approaches to firewall implementation
- Discuss the proxy firewall approach
- Destribe the strategy that enables the business use & benefits of a demilitarized zone (DMZ)

Revision - communication protocols

- A protocol is a standard means for coordinating an activity between two or more entities.
- We have political protocols, and many other types – including communication protocols (and last week we talked of ‘secure communication protocols)
- Communications protocols are broken into **levels or layers**
  - for both ‘snail mail’ and for computer communication.



Revision - communication protocols ‘snail mail’



Dave Andrews

No 867, S. Park Street,

New York, SA 12767

My dear Dave,

I am awfully sorry to hear about your friend Mike. I know how close you both were as brothers more than friends. Please accept my deep condolences on this sorrowful loss.

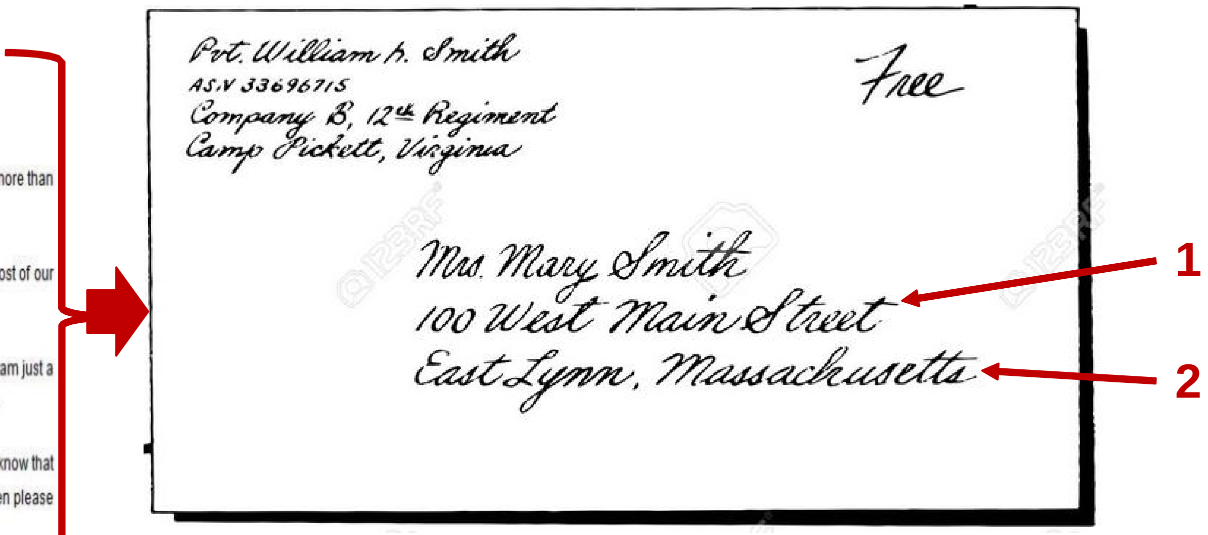
Whenever I met Mike, he came across as a very active and spirited person who took part in most of our community activities. I am sure he will be remembered fondly by those who got to know him.

I am feeling very bad that I can't be there with you at this difficult time, but remember that I am just a phone call away. Don't hesitate to call me at anytime and allow me to be of some help to you.

Let Lord bless you and his family with the strength to sail though this tough situation. Please know that you are there in our payers and thoughts. I hope to meet you by the end of this month. Until then please take care and accept my sincere sympathies.

With Love

Jack Clingstone



Pvt. William h. Smith
AS.N 33696715
Company B, 12th Regiment
Camp Pickett, Virginia

Free

Mrs. Mary Smith
100 West Main Street
East Lynn, Massachusetts

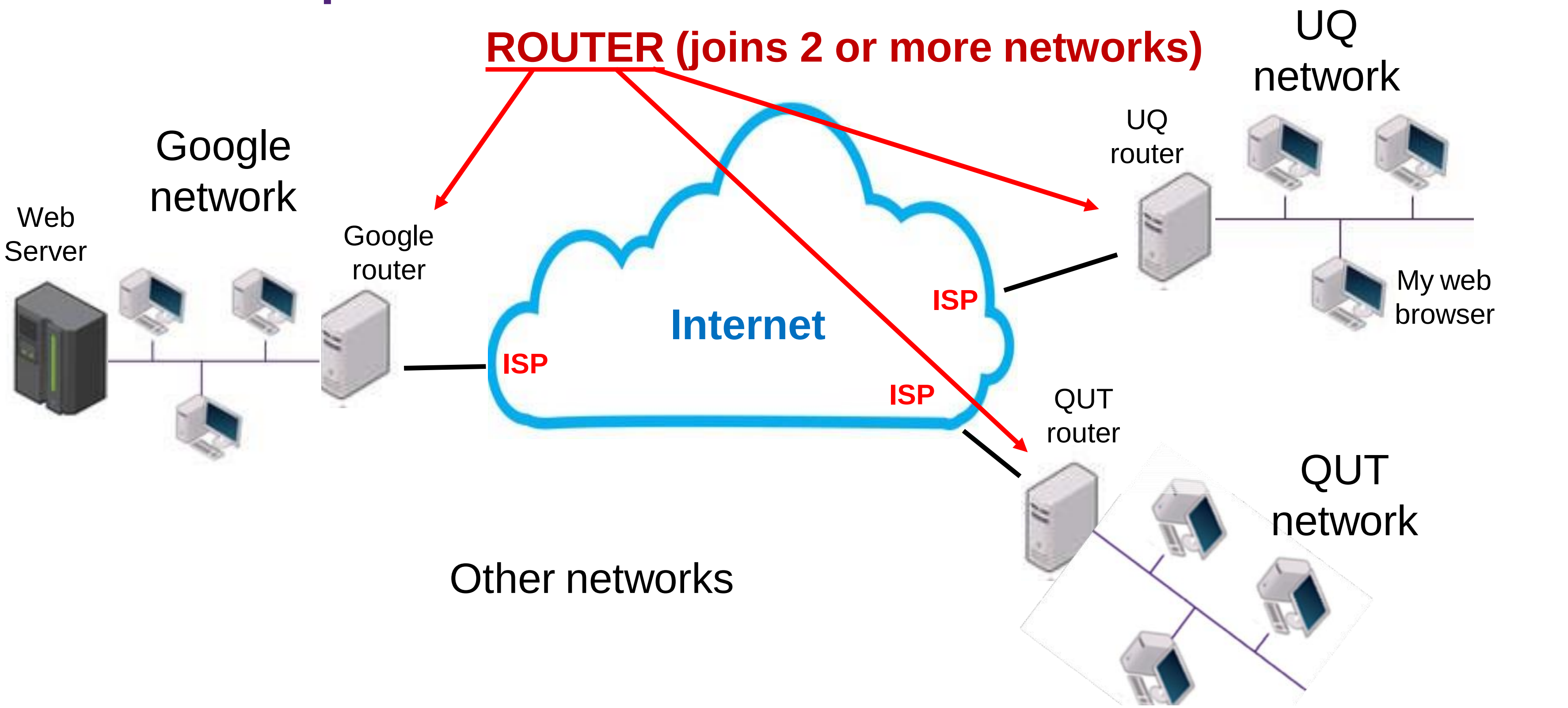
In general terms, <u>we need two levels of addressing</u>:

1) Street details

2) Suburb/town details

Remember this – it is very similar with digital communication!

Revision - computer network communication



As with snail mail, computer communication requires two **levels/layers** of addressing for each message

1) right network
2) right machine on the right network
3) right software application on the right machine on the right network

But we only have one IP address like **192.168.1.10**?

Which part of the IP address specifies the network and which the host machine?

We use the **subnet mask** used to determine which part of the IP address specifies the network and which part specifies the host.

The mask uses a specific number of bits, set to 1, to identify the network portion of the IP address, and the remaining bits, set to 0, to identify the host portion, which separates the IP address into two distinct parts.

The subnet mask is a 32-bit number that is used in conjunction with the IP address to determine which part of the IP address specifies the network and which part specifies the host.

In binary form, the subnet mask is a series of 1s followed by a series of 0s. For example, the subnet mask **255.255.255.0** in binary form is **11111111.11111111.11111111.00000000**.

To determine the network and host portions of an IP address using the subnet mask, the following process is used:

1. Convert the IP address and subnet mask to binary form.
2. Perform a bitwise AND operation between the binary IP address and the binary subnet mask.
3. The resulting value is the network portion of the IP address.
4. The remaining bits are the host portion of the IP address.

Numerical example

For example, consider the IP address **192.168.1.10** and subnet mask **255.255.255.0**.

In binary form, the IP address is (4 octets, i.e. 4 times 8 binary digits)
11000000.10101000.00000001.00001010 and the subnet mask is
11111111.11111111.11111111.00000000.

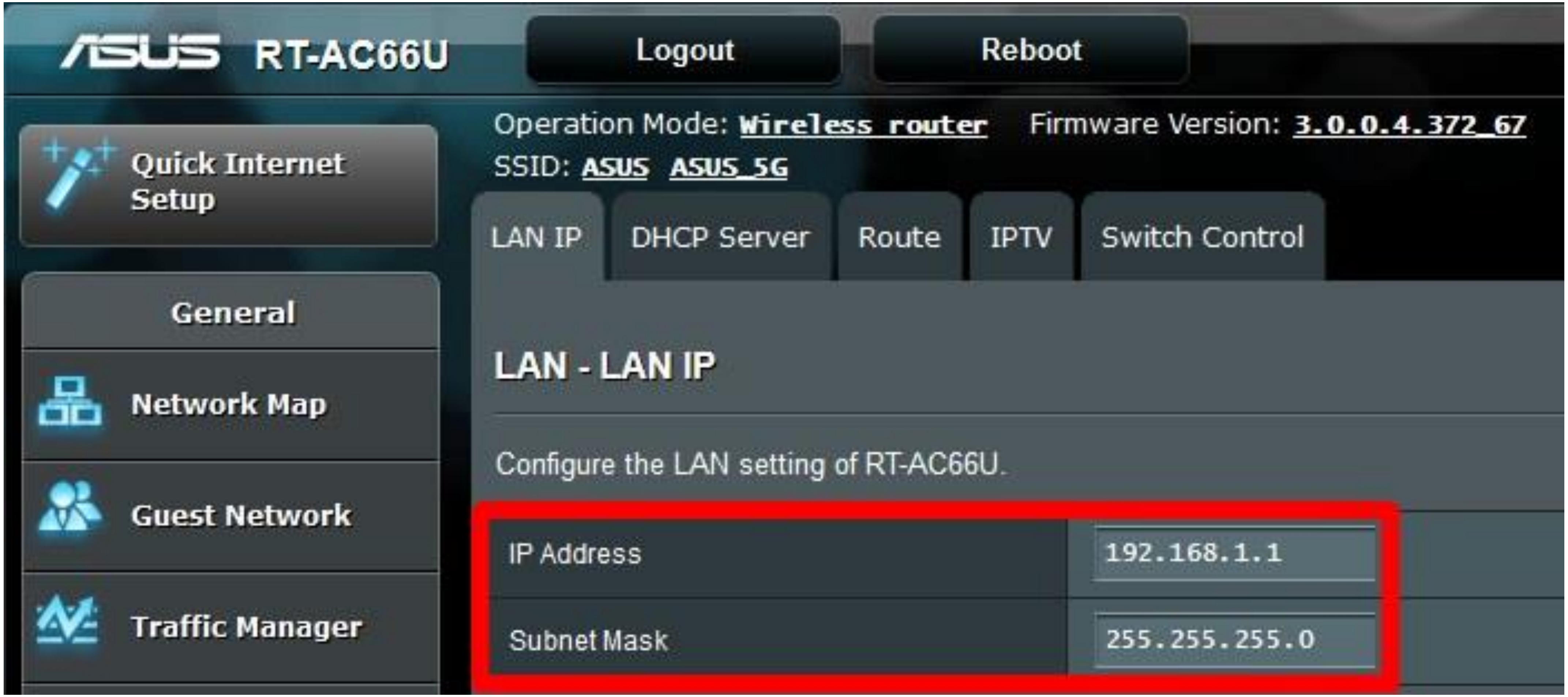
Performing a bitwise **AND** operation between the IP address and subnet mask yields the following result:

11000000.10101000.00000001.00001010 (IP address) AND
11111111.11111111.11111111.00000000 (Subnet mask) equals
11000000.10101000.00000001.00000000 (Network portion)
00000000.00000000.00000000.00001010 (Host portion)

So, the network portion of the IP address is **192.168.1.0**, and the host portion is **0.0.0.10**.

This means that any IP address with the same network portion as **192.168.1.0** (e.g. **192.168.1.54**) belongs to the same network, and traffic destined for that network can be forwarded using the information in the routing table.

Home router setup example



Operation Mode: Wireless router Firmware Version: 3.0.0.4.372_67

SSID: ASUS ASUS_5G

LAN - LAN IP

Configure the LAN setting of RT-AC66U.

IP Address	192.168.1.1
Subnet Mask	255.255.255.0

Automatic assignment of IP, subnet mask, gateway, lease time, DNS Servers can be done by Dynamic Host Configuration Protocol (DHCP).

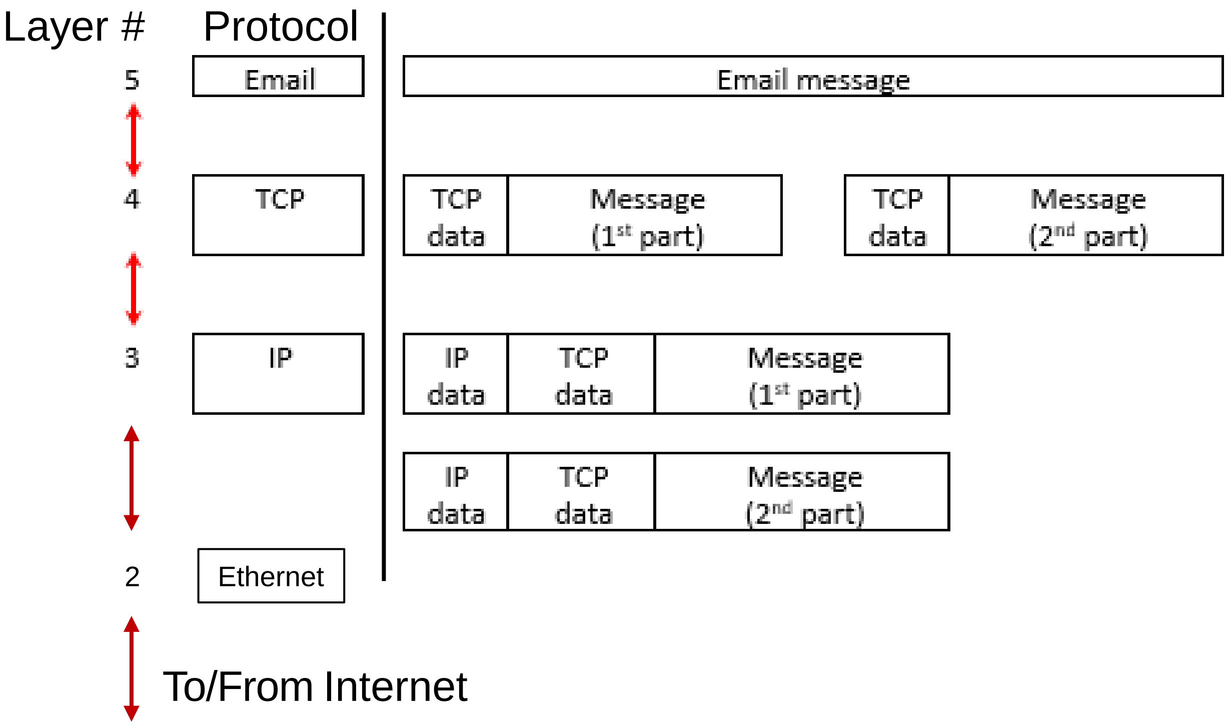
Revision - TCP/IP and OSI Architecture

TCP/IP Model			
Layer	**Name**	**Addresses**	**Comments**
7	**Application**	**Email, Web, others**	**Important – firewall focus**
4	**Transport**	**TCP, UDP**	**Important – firewall focus**
3	**Network**	**IP**	**Important – firewall focus**
2	Data link	MAC Address	**not discussed**
1	Physical	Wireless/wired	**not discussed**

right software application

right network, right machine

Example



Layer # | Protocol

5 Email | Email message

4 TCP | TCP data | Message (1st part) | TCP data | Message (2nd part)

3 IP | IP data | TCP data | Message (1st part)

IP data | TCP data | Message (2nd part)

2 Ethernet

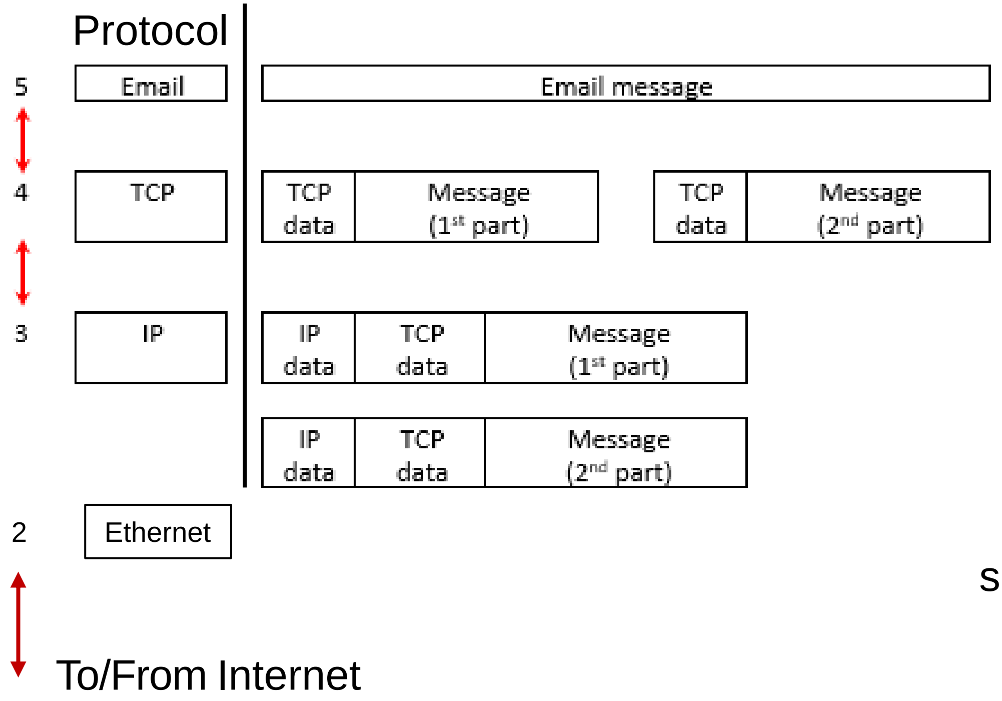
To/From Internet

TCP data identifies the sending/receiving applications via a simple number (**a port number**)

IP data identifies the sending/receiving networks and machine via an **IP address**

MAC data identifies the sending/receiving machine via a **MAC address**

TCP/IP and OSI Architecture – Firewall implications



TCP data identifies the sending/receiving applications via a simple number (**a port number**)

IP data identifies the sending/receiving networks and machine via an **IP address**

MAC data identifies the sending/receiving machine via a **MAC address**

Proxy FW

Stateful Inspection FW

Packet filtering FW

Switches, Bridges, WAP, NIC

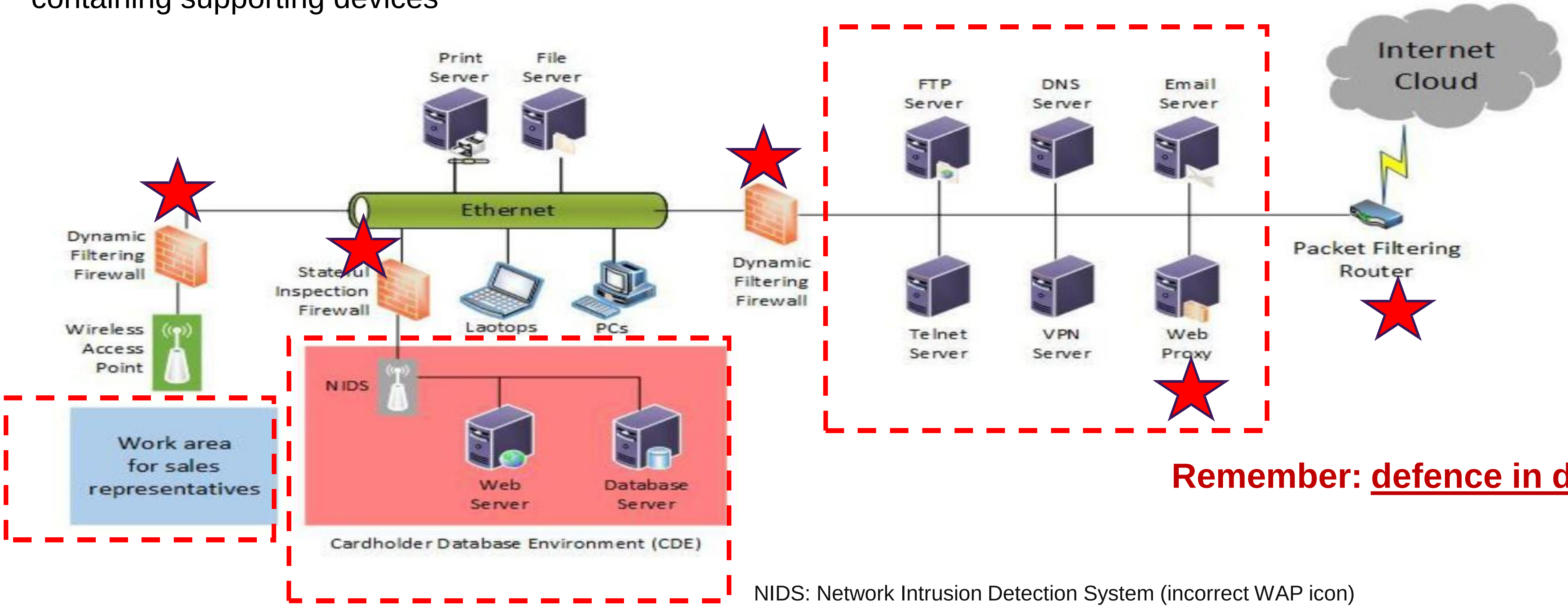
1. TCP is designed to process **CONNECTIONS** (related groups of packets)

2. IP is designed to process individual **PACKETS** (each packet individually)

3. Some firewalls work at the IP level, some at the TCP level, some at the application level. This 'level of operation' significantly determines the level of security a firewall can introduce into a network and **its use in the network**

Firewalls

- **Software** – running on some type of computer configuration
- Prevent specific types of information from moving **between** the outside world (untrusted network) and the inside world (trusted network)
- Maybe: separate computer system; software service running on existing router or server; or separate network containing supporting devices



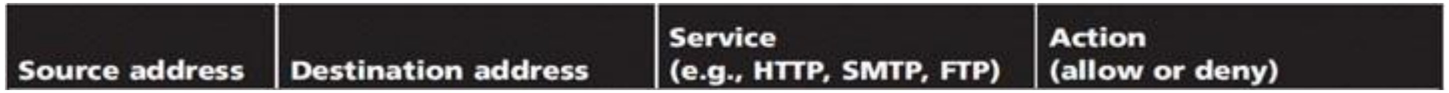
Remember: defence in depth

NIDS: Network Intrusion Detection System (incorrect WAP icon)

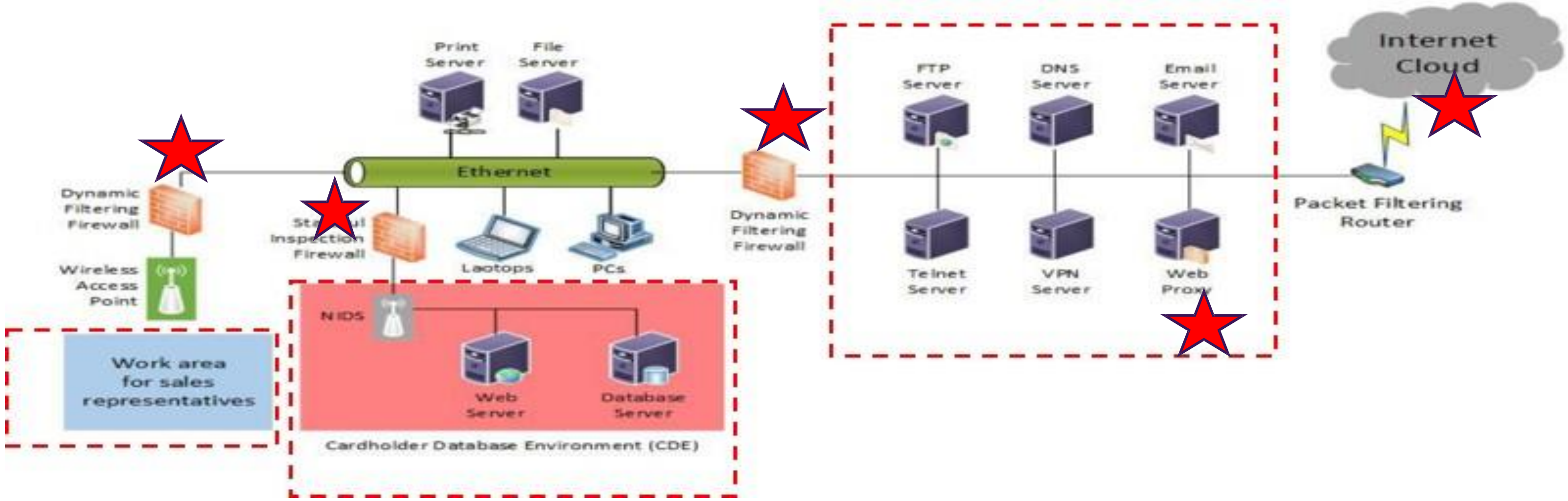
Configuring and managing – firewall **rules**

- Firewall **rules**
  - Operate by examining data packets and performing comparison with predetermined ***rules***
  - Most firewalls use packet data/header OR connection data/header information to determine whether specific packet should **be allowed or denied**

RULE STRUCTURE

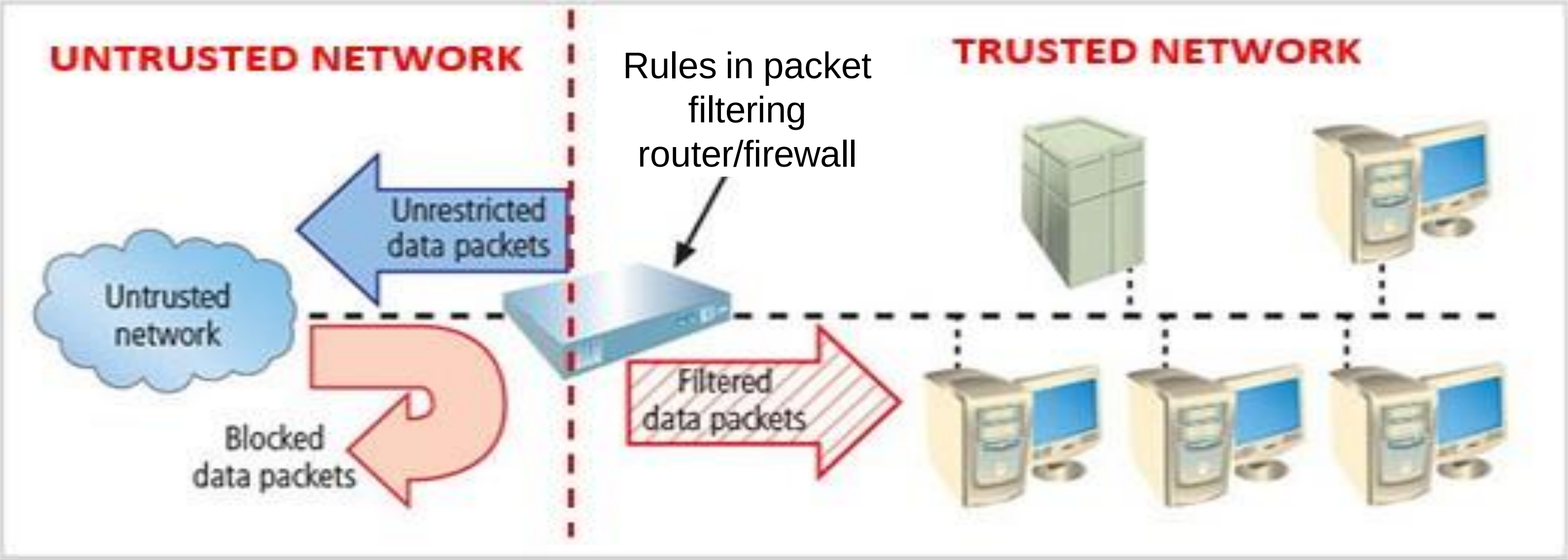


Source address	Destination address	Service (e.g., HTTP, SMTP, FTP)	Action (allow or deny)



Packet filtering & rule examples

Source Address	Destination Address	Service (e.g. HTTP, SMTP, FTP)	Action (Allow or Deny)
Nasty site	My business	Any	Deny
My supplier #1	My business	Email (SMTP)	Allow
My supplier #2	My business	Email (SMTP)	Allow

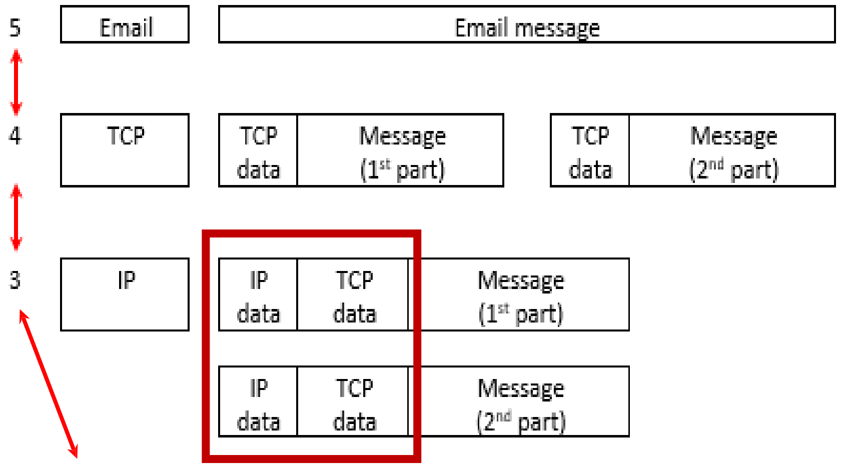


Rules in packet filtering router/firewall

Firewalls and Network Devices by OSI Layer

- **L7 Application Layer**: Application Layer Firewalls (Proxy Firewalls)
- **L4 Transport Layer:** Stateful Inspection Firewalls, Intrusion Prevention Systems (IPS)
- **L3 Network Layer:** Packet-Filtering Firewalls, Routers, VPN gateways
- **L2 Data link layer:** Switches, Bridges, WAPs, NICs
- **L1 Physical layer:** HUBs

Firewall processing modes

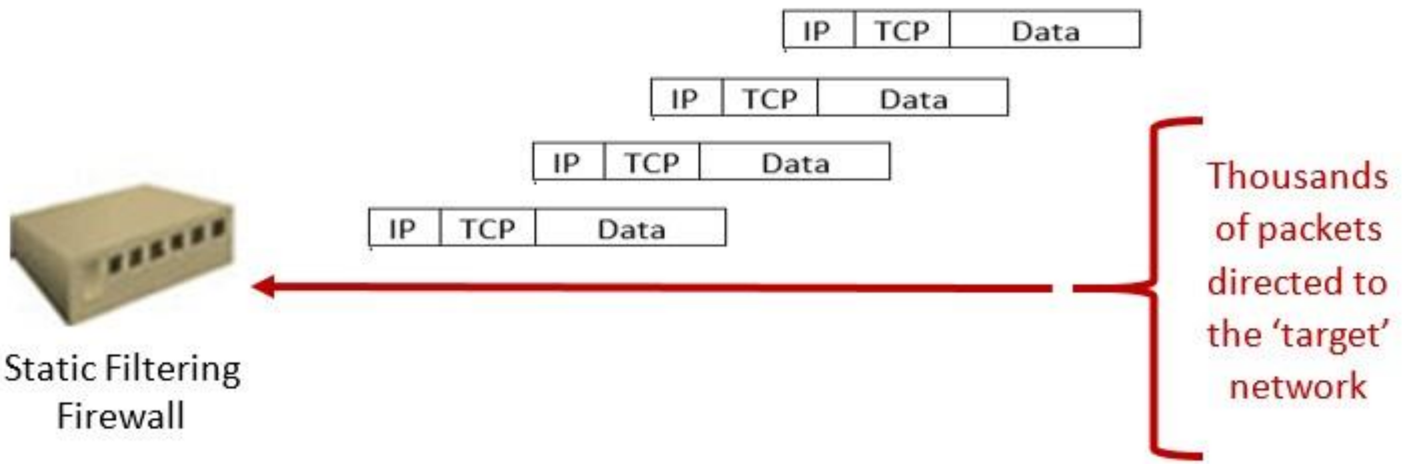


- **Packet filtering firewalls examine header information of data packets**
- Most often based on combination of:
  - Internet Protocol (IP) source and destination address
  - Direction (inbound or outbound) – both directions are critical!
  - Transmission Control Protocol (TCP) source and destination port requests
- Three subsets of packet filtering firewalls (in order of **increasing** level of security):
  - **Static filtering**: requires that filtering rules governing how the firewall decides which packets are allowed and which are denied are developed and installed – work at PACKET level
  - **Dynamic filtering**: allows firewall to react to emergent event and update or create rules to deal with event – work at PACKET level
  - **Stateful inspection**: firewalls that keep track of each network **CONNECTION** between internal and external systems using a state table

Static filtering – fastest – most limited security

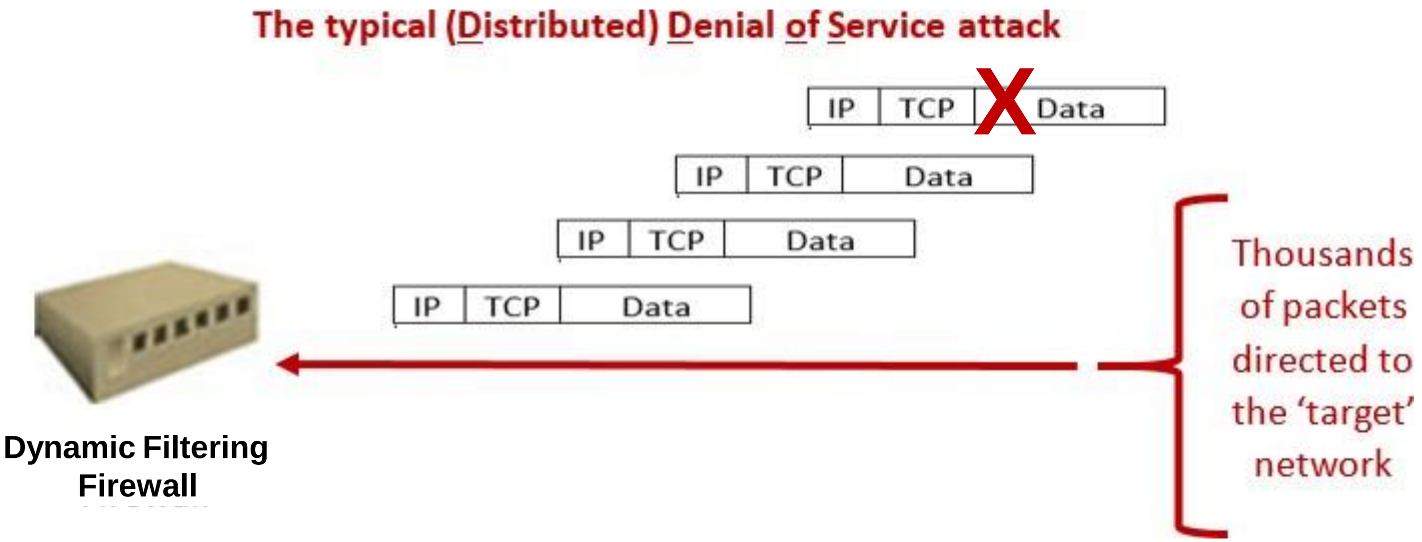
- **Static filtering**: requires that filtering rules governing how the firewall decides which packets are allowed and which are denied are developed and installed – AND ARE STATIC (cannot be changed until firewall is reprogrammed!
- It is **very simple** in its capabilities but it is the **quickest** of all firewalls. **It sees all traffic!**
- A static filtering firewall can (easily) be overwhelmed by 'unexpected' increases in workload – the firewall can be 'crashed' and therefore service is 'denied' to all legitimate users.

The typical (Distributed) Denial of Service attack


IP
TCP
Data
IP
TCP
Data
IP
TCP
Data
IP
TCP
Data
Thousands of packets directed to the 'target' network
Static Filtering Firewall

Dynamic filtering – next level (up) of sophistication

- **Dynamic filtering**: filtering rules can be changed DYNAMICALLY by the firewall itself (more intelligent). A dynamic filtering firewall can detect ‘emergent’ events – implement a consequential rule – deal with more ‘situations’.
- **This firewall, however, does not view the traffic as ‘connections. It sees all traffic!**
- We can move to a **more sophisticated design**



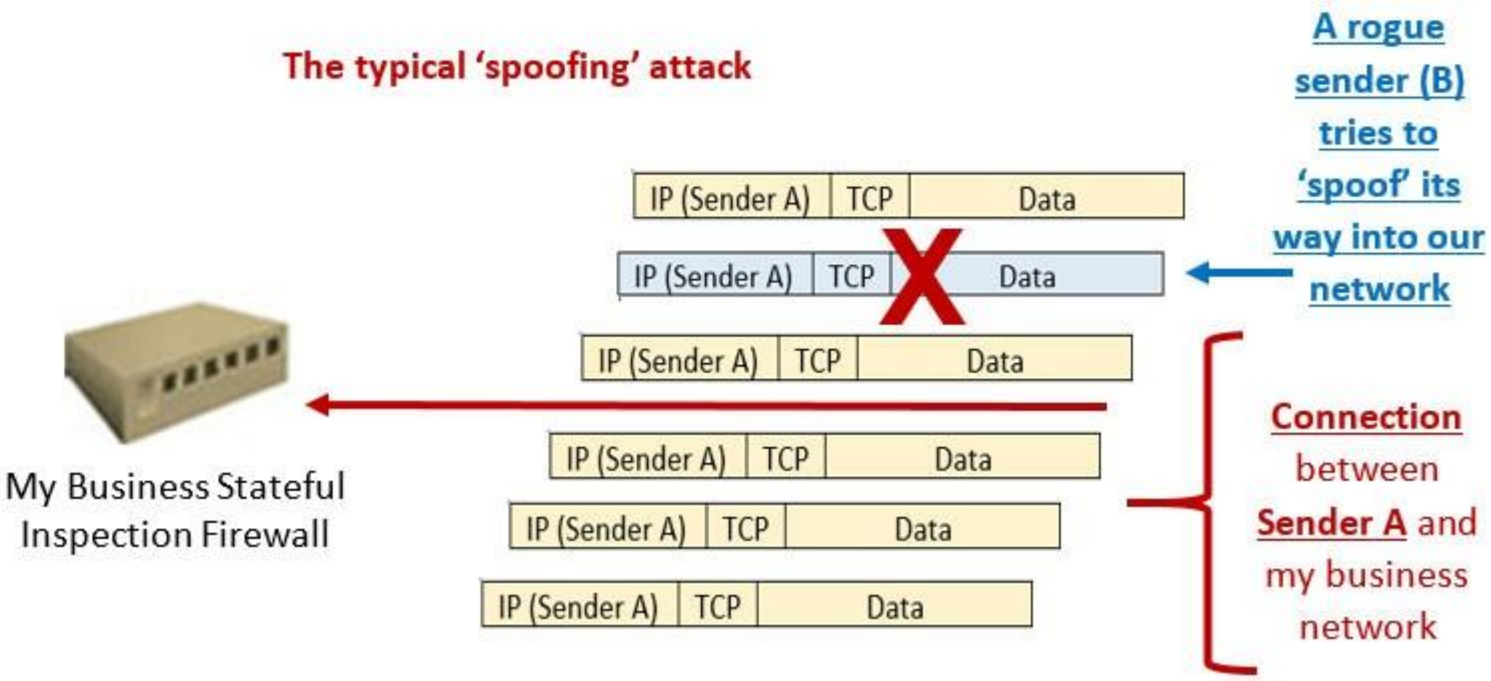
The typical (Distributed) Denial of Service attack

Thousands of packets directed to the ‘target’ network

Dynamic Filtering Firewall

Stateful inspection – top level of sophistication

- **Stateful inspection**: firewalls that keep track of each network **CONNECTION** between internal and external systems using a state table. This is the most sophisticated of the layer 3/4 firewalls – it can deal with attacks such as 'spoofing' (see below)
- **It sees (examines) all traffic!**


The typical 'spoofing' attack
IP (Sender A)
TCP
Data
IP (Sender A)
TCP
Data
IP (Sender A)
TCP
Data
IP (Sender A)
TCP
Data
IP (Sender A)
TCP
Data
IP (Sender A)
TCP
Data
A rogue sender (B) tries to 'spoof' its way into our network
Connection between Sender A and my business network
My Business Stateful Inspection Firewall

Firewall architectures (*how we position firewalls*)

- Firewall devices can be configured in a number of network connection architectures
- Best configuration depends on three factors:
  - Objectives of the network
  - Organization’s ability to develop and implement architectures
  - Budget available for function

Four common architectural implementations of firewalls:

(1) packet filtering routers/firewalls,

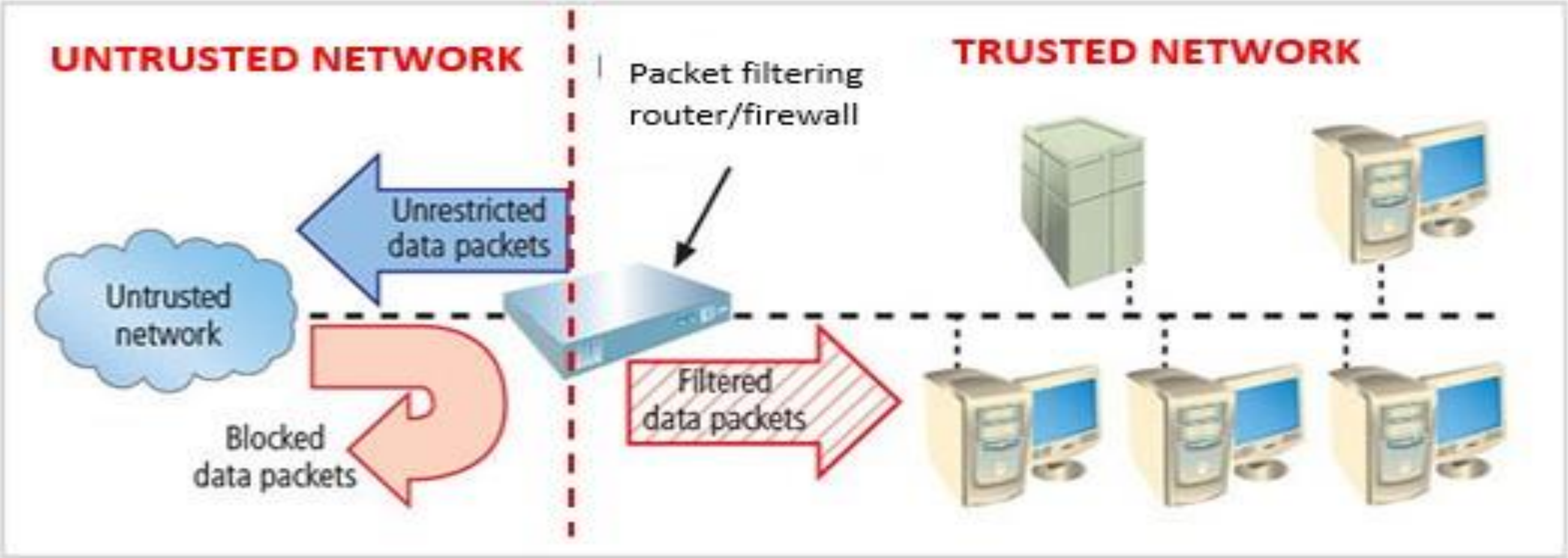
(2) screened host firewalls & (3) dual-homed firewalls : *NOT DISCUSSED*,

(4) screened subnet firewalls (DMZ) – these work with **proxies** (application gateways)

Firewall architectures (continued)

<u>Packet filtering firewalls/routers</u> (mainly work at layer 3)

- Most commonly deployed for small, uncomplicated sites – but is problematic
- Blocks packets from entry – can allow selective access to systems and services – depending on the policy
- <u>Strengths</u>: fast processing (not much of each packet to inspect) – good on 'main entrances' to networks
- <u>Drawbacks</u>: a lack of auditing (no logging), rules are difficult to test thoroughly, rules may become unmanageable, and strong authentication missing



Firewall architectures (continued)

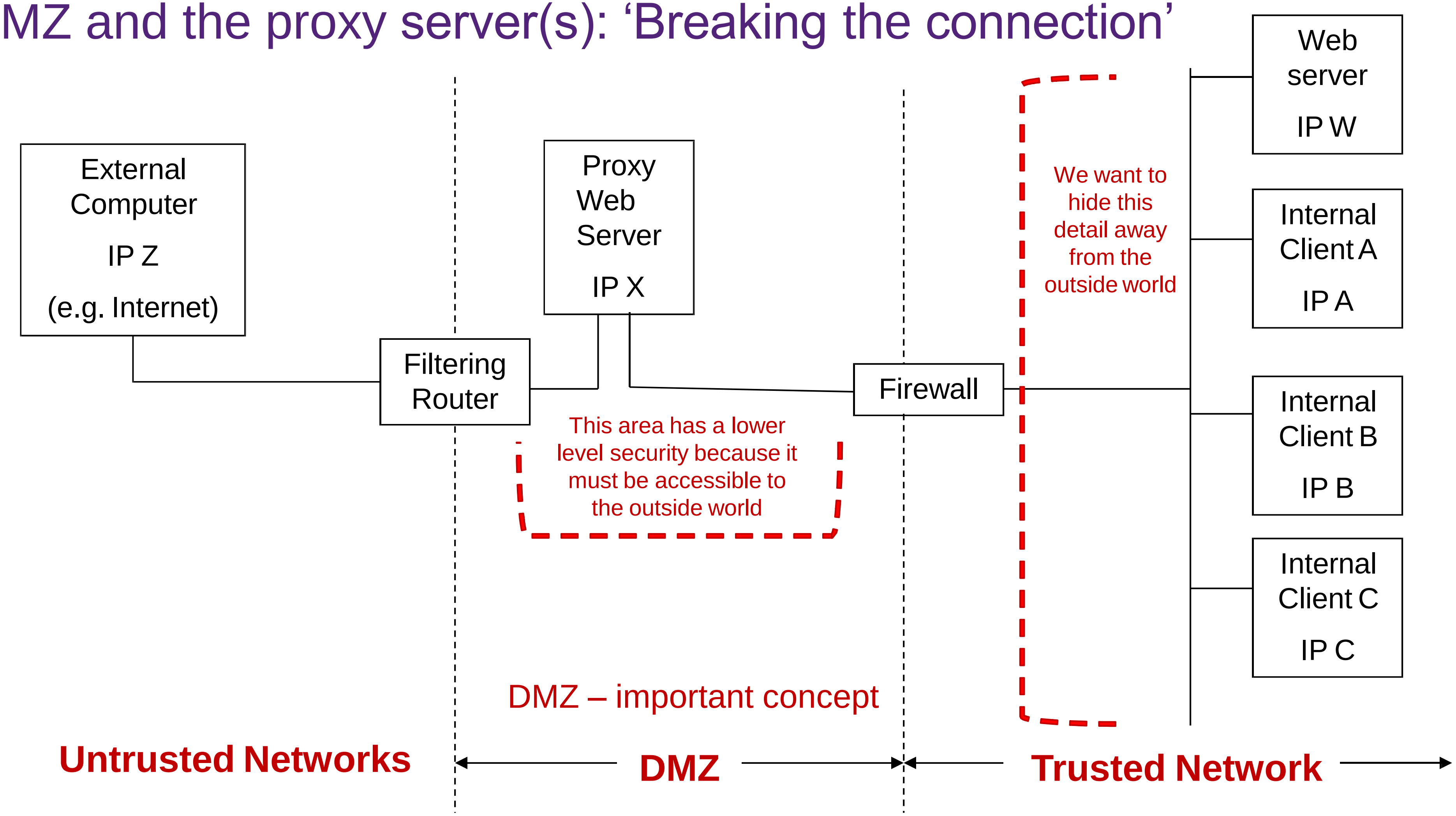
- **Screened subnet firewall** is the dominant architecture used today
- Commonly consists of two or more internal proxies (aka application gateways OR bastion hosts) behind packet filtering router, with each host protecting trusted network:

- Let's firstly discuss the proxy server concept and also the design approach known as a **screened subnet** or **DMZ** (demilitarized zone)

Firewall processing modes (continued)

- **Proxies**/**application gateways**
  - Frequently installed on a dedicated computer, also known as a proxy (server). We (almost) always have a web proxy, an email proxy – and others
  - Proxies only look at their traffic (not like packet filtering)
  - Since proxy server is often placed in unsecured area of the network because it must be accessible to outside world – it is exposed to higher levels of risk from less trusted networks
  - Additional filtering routers/firewalls can be implemented behind the proxy server, further protecting internal systems (REMEMBER: DEFENCE IN DEPTH)
  - When we place proxy servers between two firewalls/routers – we create a DMZ (or also called a screened subnet)

The DMZ and the proxy server(s): ‘Breaking the connection’

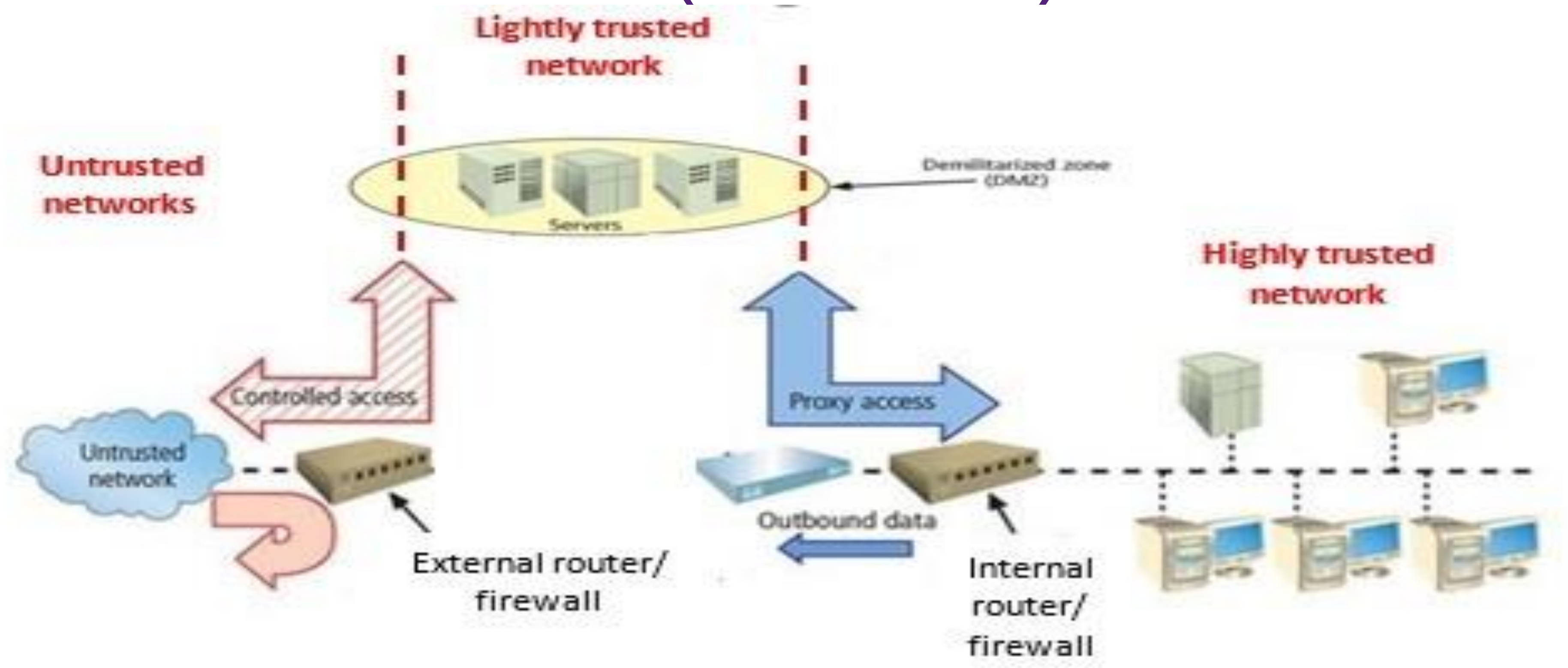


The request from Client A to the External Server is ‘***proxied***’ – the Proxy acts as the agent – this **breaks the connection** – a very important security benefit – the **details of the trusted network are hidden away – defence in depth** also

Firewall architectures (continued)

- ***Screened subnet firewall*** is the ***dominant architecture*** used today
- Commonly consists of two or more internal **proxies** (bastion hosts) behind packet filtering router, with each host protecting trusted network:
  - Connections from outside (untrusted network) routed through external filtering router to separate network segment known as DMZ
  - Connections into trusted internal network allowed only from DMZ proxy servers
- Screened subnet performs two functions:
  - Creates and **protects DMZ systems and information from outside threats**
  - **Protects the internal networks** by limiting how external connections can gain access to internal systems

Firewall architectures (continued)



Operational logic:

- All application traffic (email, Web, etc.) gets routed to the proxies
- All application traffic from trusted network (going to Internet) gets routed to proxies
- All other traffic (incoming/outgoing) blocked
- This would mean all application servers in internal network – proxies only in DMZ. Thus no site is directly reachable from the Internet (and vice-versa) – this '**breaks the connection**' and **hides internal network details**

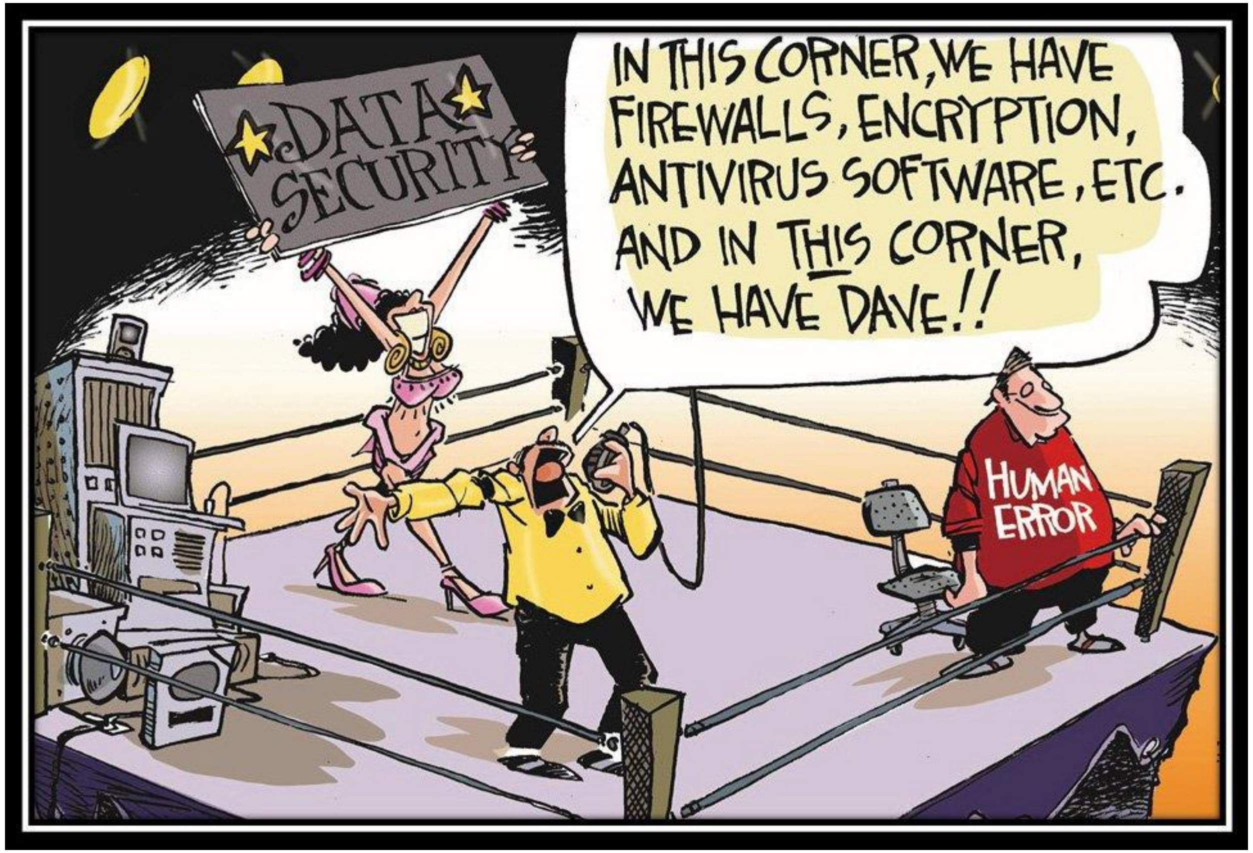
Based upon 'defence in depth' AND 'breaking the connection' concepts

Summary

Firewalls

- Technology from packet filtering to dynamic stateful inspection
- Architectures vary with the needs of the network
- Proxies
- Building the DMZ – advantages
- Building defence in depth

https://www.cisco.com/site/us/en/products/security/firewalls/firepower-9300-series/index.html#tabs-ca9b217826-item-1b113ceb83-tab CISCO Firewall 8 min video (optional).


DATA SECURITY
IN THIS CORNER, WE HAVE FIREWALLS, ENCRYPTION, ANTIVIRUS SOFTWARE, ETC. AND IN THIS CORNER, WE HAVE DAVE!!
HUMAN ERROR

```
                          func, std::vect
        end;

    write(Endtext);
end.
CREATE TABLE product(
    class MultinomialNB(object):
    def __init__(self):
2})
        self.X = None
        self.y = None
    def __loading(self):
        self.list_labels = cl.Counter(se
int acc(std::function<int(int, int)> fun
    auto it = operands.begin();
    int result = func(*it, *(++it));
    if (operands.size() > 2) {
        for (++it; it!=operands.end(); +
            result = func(result, *(it);
        }
    }
    return result;     CDog& operator=(c
```

Thank you