GOOGLE CYBER SECURITY
Course 3: Connect and Protect: Networks and Network Security

SKILLS: Information Systems Security, Cybersecurity, Computer Networking, Encryption, Network Protocols, Firewall, Hardening, Virtual Private Networks (VPN), Network Infrastructure, Network Architecture, Intrusion Detection and Prevention, Network Security, Cloud Security, TCP/IP, Vulnerability Assessments, Cloud Computing

Course 3 Summary: Connect and Protect — Networks and Network Security

This course, led by Chris (CISO at Google Fiber), provides foundational knowledge and hands-on skills in network security. You'll explore how networks operate, what makes them vulnerable, and how to protect them using modern tools and techniques.

What You'll Learn:

1. **Network Architecture**
    - Understand how networks are structured
    - Learn how devices connect and communicate
    - Explore components like routers, switches, firewalls, and access points
2. **Network Tools**
    - Introduction to tools used for monitoring and securing networks
    - Examples: Wireshark, Nmap, Netcat
3. **Network Operations and Protocols**
    - Understand how data moves through a network
    - Study key protocols: IP, TCP/UDP, HTTP/HTTPS, DNS, DHCP
4. **Common Network Attacks**
    - Identify and understand threats such as DDoS, MITM, ARP spoofing, packet sniffing
    - Recognize early indicators and attacker tactics
5. **Intrusion Detection and Prevention**
    - Introduction to IDS and IPS
    - Learn how these systems monitor traffic and trigger alerts
6. **Security Hardening**
    - Techniques to improve network security
    - Includes disabling unused ports, enabling firewalls, segmentation, and regular patching

Course Modules:

- **Module 1: Network Architecture**
  - Network structure and security concepts
  - How attackers target networks
- **Module 2: Network Operations**
  - How networks communicate
  - Role of firewalls and related tools
- **Module 3: Secure Against Network Intrusions**
  - How to detect and respond to threats
  - Tools and strategies for securing networks
- **Module 4: Security Hardening**
  - Hardening systems and cloud infrastructure
  - Implementing best security practices

Module 1

Network Architecture

Introduction to Networks

Introduction to Network Security

Before you can secure a network, it's essential to understand its **basic design and functionality**.

In this section of the course, you will:

- Learn about **network structure** and **standard networking tools**
- Explore **cloud networks**
- Understand the **TCP/IP model**, which organizes communications across a network

Securing networks is a **key responsibility** of a security analyst. This section will prepare you to **protect your organization** against various threats, risks, and vulnerabilities.

What Is a Network?

1. Definition of a Network

A **network** is a group of **connected devices** that communicate with each other.

- **Home networks** may include: laptops, phones, smart appliances
- **Office networks** may include: workstations, printers, servers
- Devices connect via **cables or wireless** (Wi-Fi)

2. Device Identification

To communicate, devices need to **find each other** on the network using:

- **IP addresses** (Internet Protocol)
- **MAC addresses** (Media Access Control)

3. Types of Networks

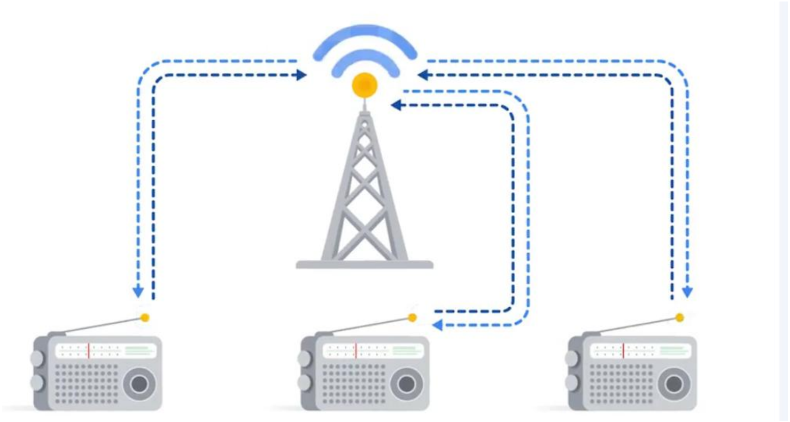
- **LAN (Local Area Network):**
  Covers a small area like a **home, school, or office**
  Example: Devices connected to home Wi-Fi
- **WAN (Wide Area Network):**
  Covers a large area like a **city, country, or the internet**
  Example: A company employee in the U.S. communicating with one in Ireland

Network Tools

Common network devices

1. Hub

- **What it does:** Sends data to **all devices** on a network.
- **Analogy:** Like a **radio tower** broadcasting to everyone tuned in.
- **Limitation:** Not secure or efficient — everyone gets the data, even if it's not meant for them.



2. Switch

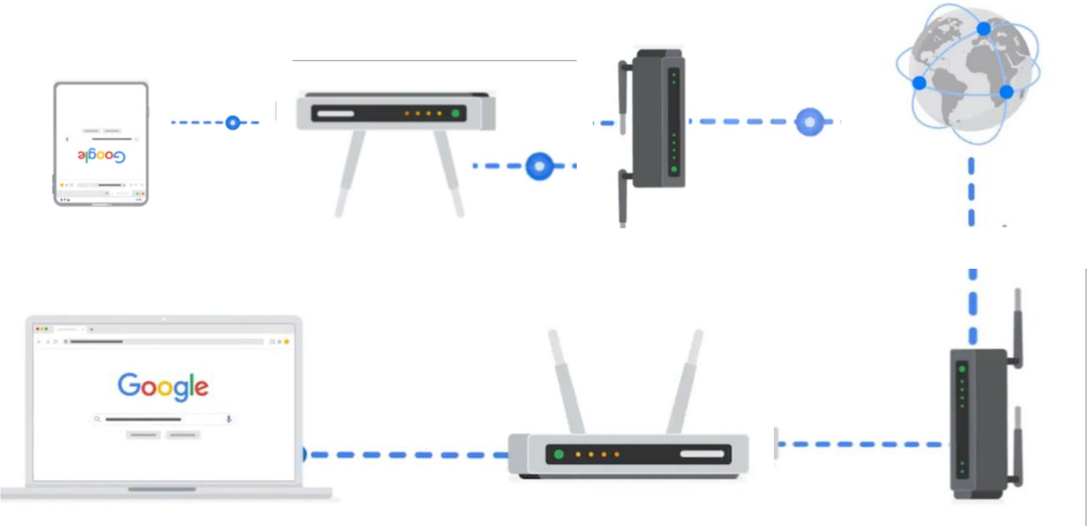
- **What it does:** Sends data **only to the specific destination device**.
- **More secure** and **smarter** than a hub.
- **Benefits:** Controls traffic flow, improves performance.

3. Router

- **What it does:** Connects **multiple networks** together.
- **Example:** If a computer in Network A wants to talk to a tablet in Network B:
  - Computer → Router A → Router B → Tablet

4. Modem

- **What it does:** Connects the **router to the internet**.
- **Example (long distance):**
  - Computer → Router → **Modem** → **Internet** → **Modem** → **Router** → Destination Device



5. Virtualization Tools

- **What they are:** Software alternatives to physical devices like hubs, switches, routers, and modems.
- **Offered by:** Cloud service providers.
- **Benefits:** Lower cost, more scalable, more flexible.

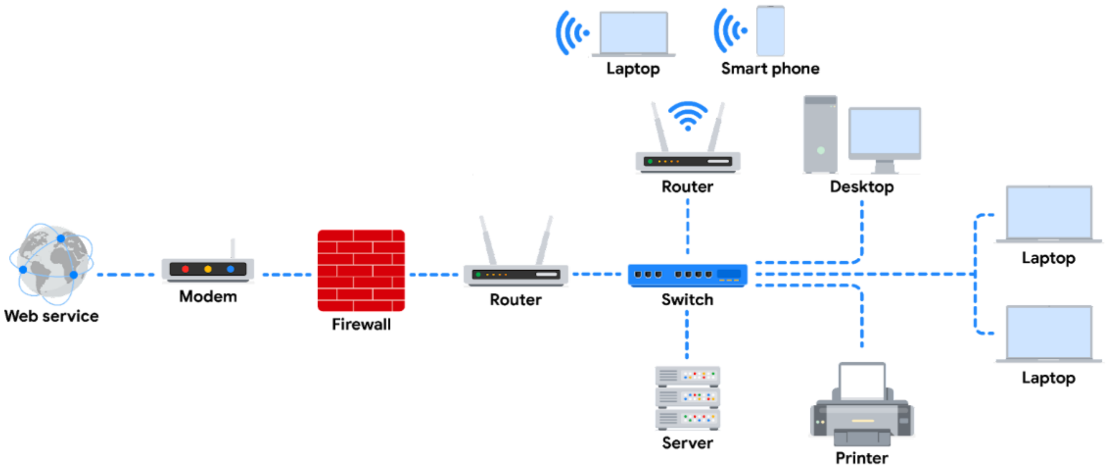
Network Components, Devices, and Diagrams

To work in cybersecurity, you need to understand how computer networks work. This includes knowing about devices like routers and switches, how they are connected, and how data moves across a network. Let's break this down.

Network Devices

Network devices help computers and other gadgets connect and talk to each other. These devices send data using wires or wirelessly (Wi-Fi). Data is sent in little packets that show where it's coming from and where it's going.

A **network** is the setup that connects everything. **Devices like routers and switches** control how data moves. Devices like computers and phones connect through these network devices.



In the example diagram:

- A **router** connects to the internet using a **modem** (provided by your internet company).
- A **firewall** is added to protect your network by checking traffic.
- The **router** sends traffic to home devices (like phones, computers, tablets).
- A **switch** is used to add more devices using Ethernet cables.
- Two **routers** are used for **load balancing** to make the network faster and more stable.
- A **server** in the diagram stores files, and all devices can use those files.

Devices & Desktop Computers

Most people use devices like laptops, desktops, or phones. Each has a **MAC address** and **IP address** (like an ID). These devices send and receive data using either cables or Wi-Fi.

Firewalls

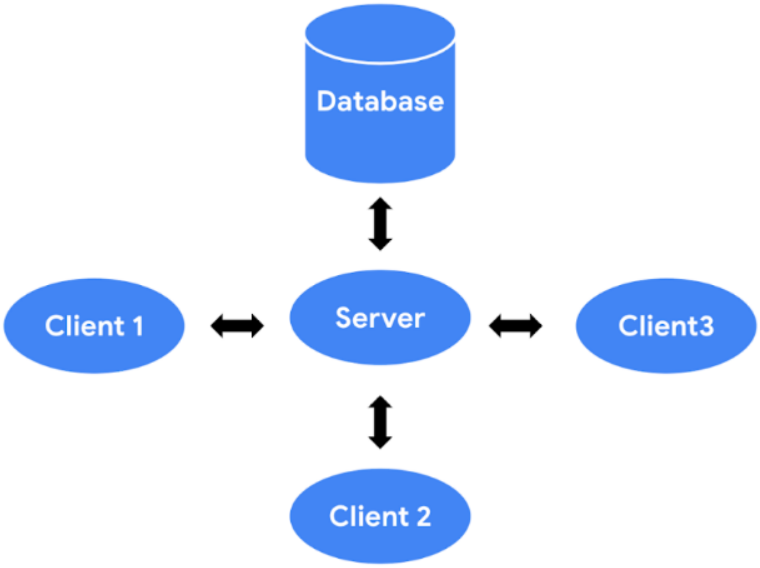
A **firewall** protects your network by checking what data is going in and out. It blocks bad or unwanted traffic based on rules. It's placed between your private network and the internet.

Servers

A **server** gives data and services to devices like phones or computers. These are called **clients**. In the **client-server model**, clients ask the server for things, and the server replies.

Examples:

- **DNS servers** help load websites.
- **File servers** store and share files.
- **Mail servers** manage emails.



Hubs and Switches

Both hubs and switches connect devices on the same local network.

- A **hub** sends data to **all** devices connected. This is not secure and can be slow.
- A **switch** is smarter. It sends data only to the **correct device**. It keeps a list (MAC table) of connected devices. Switches are more secure and fast, so they are used more today.

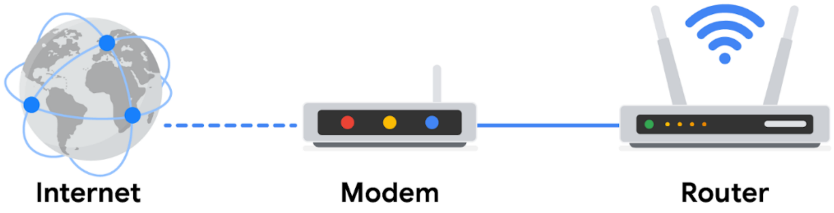
Routers

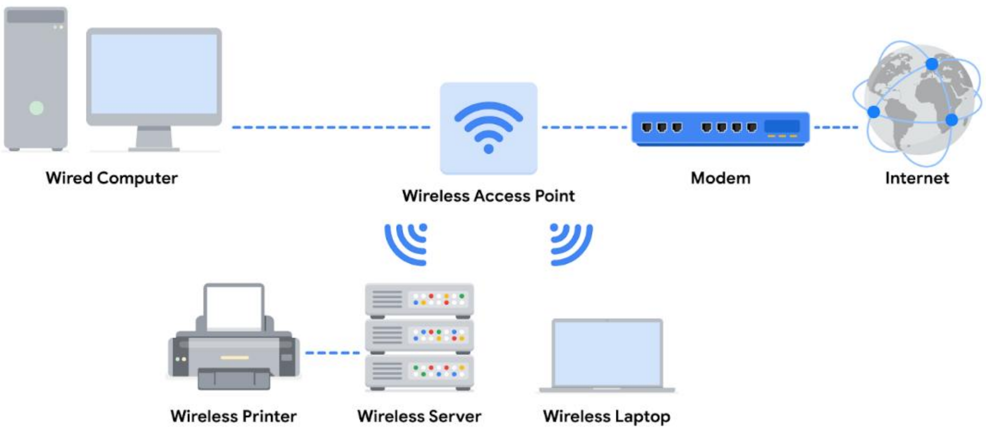
Routers connect different networks. They send data based on IP addresses. A router helps devices on different networks talk to each other. It can also act like a firewall and block bad data.

Modems and Wireless Access Points

- A **modem** connects your home to the internet from your internet company. It changes the signal into one that your devices can use.
- A **wireless access point (WAP)** lets your devices connect to Wi-Fi using radio waves. The data then travels through routers and switches to reach where it needs to go.

Big companies usually don't use modems—they use faster ways to connect to the internet.





Main Ideas

- **Client-server model:** Clients (like phones) ask servers (like file servers) for help or data.
- **Network devices:** Include routers, switches, hubs, servers, modems.
- **Network diagrams:** Help security people see how things are connected so they can protect it better.

Cloud Networks

- In the past, companies used to buy and keep all their network devices (like servers and routers) in their own offices.
- Now, many companies let third-party providers (like Amazon or Google) manage their networks for them.
- This helps companies save money and get more powerful resources without buying everything themselves.

What is Cloud Computing?

- Cloud computing means using servers, apps, and services that are on the internet, not on your own devices.
- These cloud servers are located in data centers, far away from your office, and can be accessed from anywhere using the internet.

Why Cloud is Popular

- More and more businesses use cloud computing every year because:
    - It’s cheaper
    - It’s easier to manage
    - You can access it from any location
- Instead of hosting their own web servers in their building, companies use remote cloud servers.

Cloud Services

- Cloud providers give services like:
    - On-demand storage (you only pay for what you use)
    - Processing power (for running apps)
    - Web analytics (to track visitors and sales)

Security in the Cloud

- As more companies move to the cloud, cloud security becomes very important.
- Security experts now have to:
    - Check where the traffic comes from
    - Confirm who is sending the traffic (identity-based security)
    -

Cloud Computing and Software-Defined Networks (SDNs)

You've learned how different devices like computers, servers, routers, and switches connect to make a network. Some networks are small (like inside an office), called **LAN (Local Area Network)**, and others are large (across cities or countries), called **WAN (Wide Area Network)**. You also learned about **cloud networks**—these are networks run using the internet, not just physical devices.

Now let's understand **cloud computing**, **hybrid networks**, and **software-defined networks (SDNs)** more simply.

What is Cloud Computing?

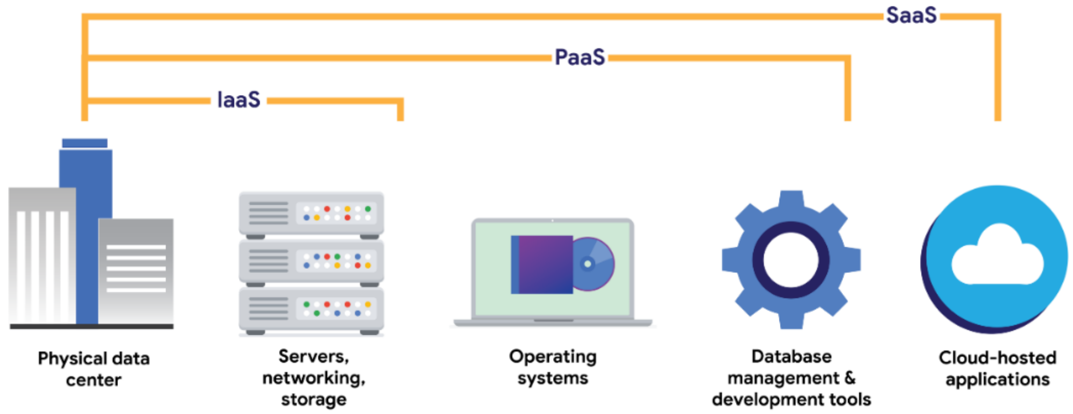
In older setups, companies had all their devices (computers, servers, etc.) in one physical place like an office. This is called an **on-premise network**.

But **cloud computing** means using computers and services over the **internet**, not from your office. For example, instead of buying and setting up your own servers, you can just use a **Cloud Service Provider (CSP)** like Google Cloud, AWS, or Azure.

These CSPs have **huge buildings (data centers)** filled with millions of servers. They let companies "rent" computer power, storage, or services using the internet.

3 Types of Cloud Services

1. **SaaS (Software as a Service)**
   You use software (like Gmail, Google Docs) directly over the internet without installing it.
2. **IaaS (Infrastructure as a Service)**
   You get virtual computers, storage, and network devices from the cloud. You can use them to run your own software or websites.
3. **PaaS (Platform as a Service)**
   You get tools to **build and run custom apps** in the cloud (for developers).


SaaS
PaaS
IaaS
Physical data center
Servers, networking, storage
Operating systems
Database management & development tools
Cloud-hosted applications

What is a Hybrid Cloud?

A **hybrid cloud** is when a company uses both:

- its own devices (on-premise)
- and cloud services.

A **multi-cloud** is when a company uses services from more than one CSP (like using both AWS and Azure).

What are Software-Defined Networks (SDNs)?

SDNs are **virtual versions** of network devices like routers, switches, and firewalls. Instead of using actual hardware, these run as software in the cloud.

Modern devices also support this. For example, a router may use software to manage how data flows.

Why Businesses Like Cloud and SDNs

1. Reliability
Cloud is more stable. Employees and customers can always access what they need with fewer disruptions.

2. Lower Cost
Companies don’t need to buy expensive servers or network devices. They can use the cloud and only pay for what they use.

3. Scalability (Growing Easily)
If a company grows and needs more computer power or storage, it can get it instantly from the

cloud. And if the need drops, they can scale down. So they don't waste money on unused equipment.

Also, cloud services can quickly set up security tools (like firewalls and IDS/IPS) when needed.

Key Points to Remember

- **Cloud computing** means using services over the internet, not just from local computers.
- **CSPs** own big server centers and offer things like storage and computing to companies.
- **SDNs** are software-based versions of real network devices like routers and firewalls.
- Companies use cloud and SDNs because they are **cheaper, reliable, and easy to grow** with.

Network Communication

Networks help people and companies stay connected and share information. But this communication can also be risky. Hackers can try to attack weak devices or unprotected networks.

When devices talk to each other on a network, they send small pieces of data called **data packets**.

A **data packet** is like a small box of information. It travels from one device to another. Each packet has:

- Where it's going (destination)
- Where it's coming from (source)
- The actual message inside

Think of it like mailing a letter:

- The **envelope** has the address of your friend (destination) and your own address (source).
- Inside the envelope is the **message**.
- The end of the letter might have your **signature**, showing it's complete.

Similarly, a data packet has:

- A **header** – shows IP and MAC address (where to go and who it's from)
- A **protocol number** – tells what kind of data it is
- A **body** – the actual message
- A **footer** – shows the packet is finished

How fast data packets move shows how good the network is. This is measured using **bandwidth**.

- **Bandwidth** = how much data is received every second.
- **Speed** = how fast the data arrives.

If the bandwidth or speed is unusual, it might mean someone is attacking the network.

Packet sniffing means watching and checking the packets to see what's going on in the network.

Good communication over a network helps companies work smoothly. You'll learn next about the rules (protocols) that help make this happen.

TCP/IP Model

TCP/IP is a model used for communication over the internet. It stands for:

- **TCP (Transmission Control Protocol)**: This helps two devices (like two computers) connect and share data in a proper way. It also checks if the data reached the correct place.
- **IP (Internet Protocol)**: This is about the rules that help send the data to the right address. It uses **IP addresses** to make sure the data goes to the correct device.

What happens when data is sent?

- When data is sent across a network, it is broken into small parts called **data packets**.
- Each packet has a **header** (where it's going, where it came from), a **body** (the actual message), and a **footer** (which shows it's the end of the packet).

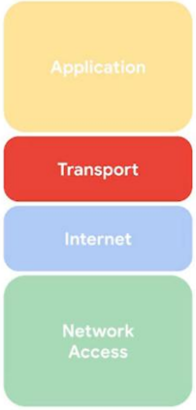
What are ports?

- A **port** is like a room number in a big building (computer system).
- Each port handles a different type of task or service. For example:
  - **Port 25** – for emails
  - **Port 443** – for secure websites
  - **Port 20** – for big file transfers

So, just like a mailman delivers letters to the right apartment in a building, computers use **port numbers** to send data to the correct place inside the device.

The four layers of the TCP/IP model

The TCP/IP model is like a guide that explains how data moves across a network. It has four main parts called *layers*. Each layer has its own job:



1. Network Access Layer

This is the bottom layer. It prepares the data so it can travel over the network. It also includes the devices and cables that physically send the data. Think of it like roads and vehicles carrying messages.

2. Internet Layer

This layer adds addresses (called IP addresses) to the data. These addresses show where the data is coming from and where it's going — like writing a home address on a letter. It also decides whether the data stays inside a small network (like a home or office) or goes out to a bigger network (like the internet).

3. Transport Layer

This layer makes sure data flows properly. It checks for errors and helps control traffic, like traffic lights managing cars on the road. It allows or blocks communication based on connection status.

4. Application Layer

This is the top layer. It decides how the data is used by the device. For example, it helps manage emails, downloads, or file sharing — like opening a letter and using what's inside.

Learn more about the TCP/IP model

This part will help you better understand the TCP/IP model. You'll also learn how it is different from the OSI model and how both are connected. Then we will look at each layer of the TCP/IP model and the common protocols used in each.

As someone working in cybersecurity, it's important to know the TCP/IP model because it explains how different network protocols work. The TCP/IP model is based on the TCP/IP protocol suite, which includes all the rules (protocols) that allow devices to send data across networks.

A *network protocol* is just a set of rules for how data is moved between devices on a network. This section will teach you which protocols work at which layers of the TCP/IP model.

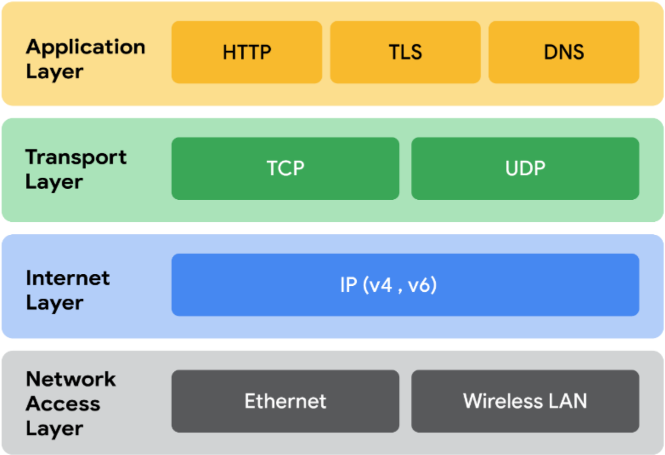
What is the TCP/IP model?

The TCP/IP model is a simple way to understand how data moves across a network. It helps network engineers and security experts figure out where problems or attacks might happen in a network.

It has four layers:

1. Network Access Layer
2. Internet Layer
3. Transport Layer
4. Application Layer

Security experts can look at each of these layers to find out where a problem or attack happened.



1. Network Access Layer

- Also called the *data link layer*.
- It creates and sends data packets over the network.
- This layer includes the physical parts like cables, hubs, and modems.
- ARP (Address Resolution Protocol) works in this layer. It connects IP addresses to MAC addresses so devices on the same network can talk to each other.

2. Internet Layer

- Also called the *network layer*.
- It makes sure data gets to the right device, even if it's on another network.
- It puts IP addresses on data packets to show where they're from and where they're going.
- Main protocols:
  - IP (Internet Protocol): Sends data to the right destination.
  - ICMP (Internet Control Message Protocol): Sends error messages if data is lost or a connection fails. It's useful for troubleshooting.

3. Transport Layer

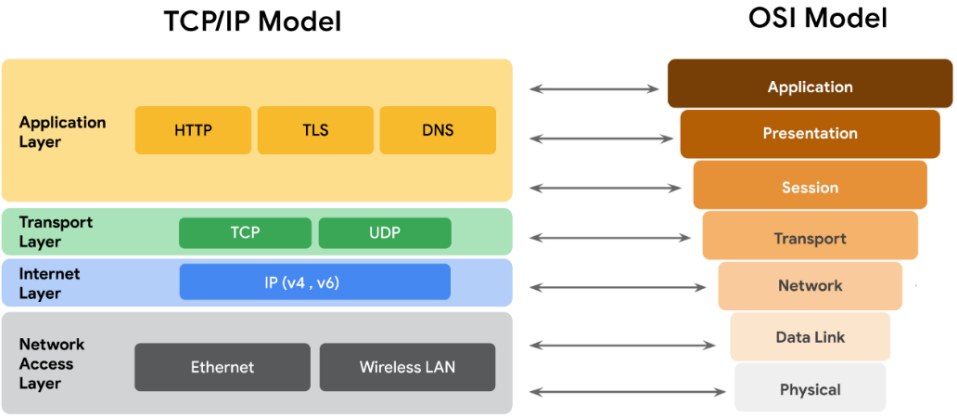
- Sends data between two devices.
- Controls how data flows across the network.
- Main protocols:
  - TCP (Transmission Control Protocol):
    - Makes a connection before sending data.
    - Checks if data arrives safely.
    - Adds the port number for the correct service.
  - UDP (User Datagram Protocol):
    - Does *not* make a connection first.
    - Doesn't check if data was received correctly.
    - Used for real-time things like video streaming where speed is more important than accuracy.

4. Application Layer

- This is where network requests are made or answered.
- It decides what services or apps a user can use.
- Depends on the lower layers to move data.
- Common protocols:
  - HTTP – for browsing websites.
  - SMTP – for sending emails.
  - SSH – for secure remote access.
  - FTP – for transferring files.
  - DNS – for changing domain names into IP addresses.

TCP/IP Model vs OSI Model

- The OSI model has 7 layers, and it helps people talk clearly about where issues happen in a network.
- The TCP/IP model has 4 layers, and it combines some of the OSI layers to make things simpler.
- Both models help professionals understand how data moves through a network.



Key Takeaways:

- The TCP/IP model has 4 layers.
- The OSI model has 7 layers.
- Both models help explain how devices send data across networks.
- The TCP/IP model is simpler and is used more often in real networks.

The OSI Model

In this part of your course, you’ve learned about how networks work, what devices are used in a network, and how communication happens between devices. You also learned about the **TCP/IP model**, which shows how data travels across the internet in layers.

All communication on a network happens through **protocols**. For example, **TCP** helps two devices create a connection, and **IP** helps send data to the right address. These are used in different layers of the TCP/IP model. The **TCP/IP model** has 4 layers, while the **OSI model** (which is more detailed) has 7 layers. In this reading, you’ll learn more deeply about the 7 layers of the OSI model. We’ll start from layer 7 (used by people) and go down to layer 1 (physical devices like cables).

TCP/IP Model vs. OSI Model

The **TCP/IP model** shows how data is sent and received over the internet. It helps professionals understand and talk about where a problem or security issue might have happened in the network.

- TCP/IP has **4 layers**:
    1. Network Access
    2. Internet
    3. Transport
    4. Application

The **OSI model** is a **standard 7-layer model** used to explain how computers send and receive data. Many network and security experts use this model to identify problems or security threats.

Both models are useful, and as a security analyst, you should know both.

The 7 Layers of the OSI Model

Layer 7 – Application Layer

This is the top layer where the **user interacts** with apps to connect to the internet. For example:

- When you open a **web browser**, it uses **HTTP/HTTPS** to talk to a website.
- Your email app uses **SMTP** to send emails.
- Your browser uses **DNS** to turn a website name (like google.com) into an IP address.

Layer 6 – Presentation Layer

This layer is about **formatting and protecting data** so both sides can understand it.

- It **encrypts** data (like **SSL/HTTPS** does).
- It may **compress** data to make it smaller.
- It changes data formats so apps can read it properly.

Layer 5 – Session Layer

This layer manages the **start, use, and end of a communication session** between two devices.

- It keeps the session open while data is being transferred.
- It handles **reconnection** if the session is interrupted.
- It can use **checkpoints** so data resumes from where it stopped.

Layer 4 – Transport Layer

This layer handles **moving data** between devices and **splitting data** into small pieces called **segments**.

- These segments are sent, and then put back together at the destination.
- It also controls **speed and flow** of data.
- Uses **TCP** (reliable) or **UDP** (faster, but less reliable).

Layer 3 – Network Layer

This layer decides how data travels from one **network to another**.

- It puts **IP addresses** in data packets.
- Routers use this layer to know where to send the data.

Layer 2 – Data Link Layer

This layer sends data within the **same network** (like inside your home or office).

- It uses devices like **switches** and **network cards**.
- It uses protocols like **HDLC**, **SDLC**, and **NCP**.

Layer 1 – Physical Layer

This layer is all about the **hardware**:

- **Cables, wires, modems, and hubs**.
- It changes data into **0s and 1s** (binary) and sends it through wires.

Final Points

- **Both TCP/IP and OSI models** help experts understand and explain how data travels across a network.
- The **OSI model has 7 layers**, and it helps to find problems or security threats.
- Security analysts and network engineers use these models to **find issues** and explain where they happen in the network.

Local and Wide Network Communication

IP addresses and network communication

What is an IP address?

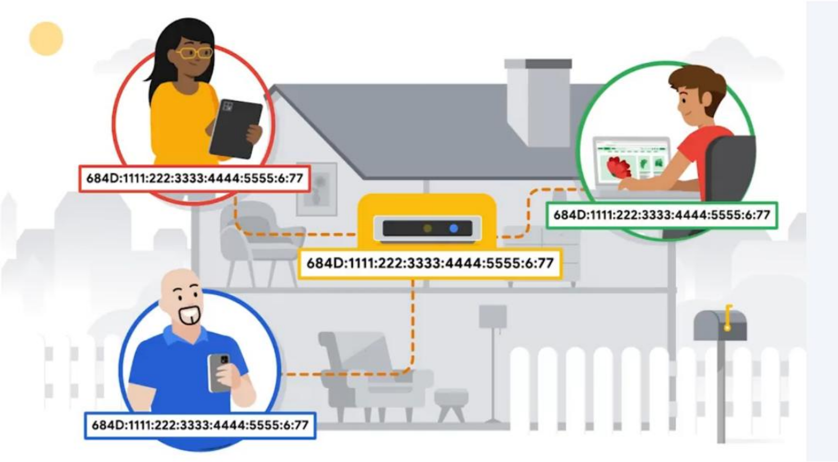
- **IP** means **Internet Protocol**.
- An **IP address** is a number that is used to **identify your device** on the internet.
- Just like every house has a unique address, **every device on the internet has a unique IP address**.

Types of IP addresses

1. **IPv4**:
   - Looks like this: 192.168.1.1
   - It uses **4 sets of numbers** separated by dots.
   - It's the older format but still used a lot.
2. **IPv6**:
   - Looks like: 2001:0db8:85a3:0000:0000:8a2e:0370:7334
   - It is **much longer** and allows **more devices** to connect to the internet.
   - Created because **IPv4 addresses were running out**.

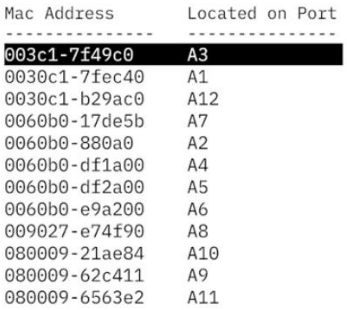
Public and Private IP addresses

- **Public IP**:
  - Given by your **internet provider**.
  - Can be **seen on the internet**.
  - All devices in your home share the **same public IP**.
- **Private IP**:
  - Used **inside your home network**.
  - Only seen by **devices in your home**.
  - Example: Your phone and laptop can talk to each other using private IPs.\


684D:1111:222:3333:4444:5555:6:77
684D:1111:222:3333:4444:5555:6:77
684D:1111:222:3333:4444:5555:6:77
684D:1111:222:3333:4444:5555:6:77

What is a MAC address?

- A **MAC address** is a special code given to each device's **network card** (like Wi-Fi or Ethernet).
- It's **unique to every device** and doesn't change.
- **Switches** use MAC addresses to know **which port to send data to**.



```
Mac Address        Located on Port
---------------    ---------------
003c1-7f49c0       A3
0030c1-7fec40      A1
0030c1-b29ac0      A12
0060b0-17de5b      A7
0060b0-880a0       A2
0060b0-df1a00      A4
0060b0-df2a00      A5
0060b0-e9a200      A6
009027-e74f90      A8
080009-21ae84      A10
080009-62c411      A9
080009-6563e2      A11
```

Recap:

- IP addresses help devices talk on the internet.
- IPv4 and IPv6 are two formats.
- Public IP = visible on internet, Private IP = used inside home.
- MAC address = helps local devices connect correctly.

Components of network layer communication

What happens at the network layer (Layer 3 of OSI)?

- The network layer helps **send data from one device to another**.
- It figures out the **best path** for data to travel using **IP addresses**.
- Routers use this information to **move the data from one place to another**.

What is in a data packet?

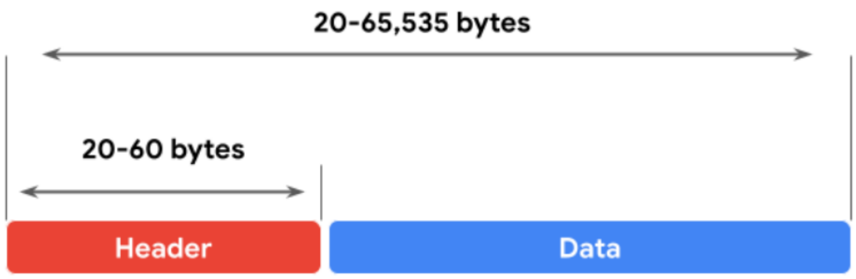
- A **data packet** is a small chunk of information being sent over a network.
- It has two parts:
  - **Header**: Tells where it's going, where it came from, and how to handle it.
  - **Data**: The actual message, like a web page or email.
- If the packet is for **TCP**, it's called an **IP packet**.
- If it's for **UDP**, it's called a **datagram**.

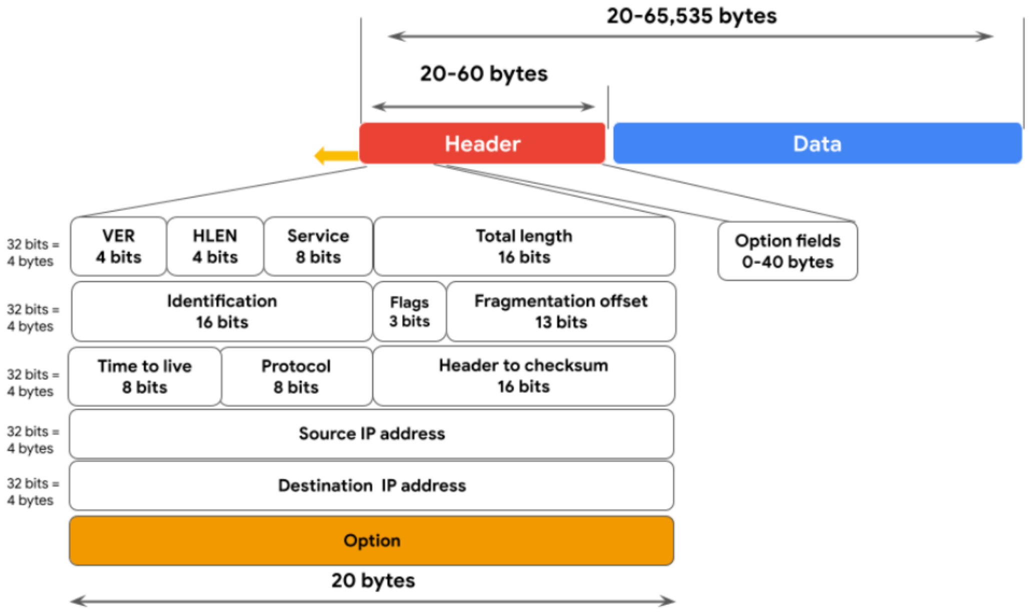
What's inside the IPv4 packet header?

There are **13 parts** in the header:

1. **Version (VER)**: Shows which IP version (IPv4 or IPv6) is used.
2. **Header Length (HLEN)**: Tells where the header ends and data begins.
3. **Type of Service (ToS)**: Helps routers know how important the packet is.
4. **Total Length**: Size of the whole packet. Max is 65,535 bytes.
5. **Identification**: If a large packet is split into smaller ones, this keeps track of them.
6. **Flags**: Shows if the packet is part of a bigger one.
7. **Fragment Offset**: Helps put split packets back in the right order.
8. **Time to Live (TTL)**: Stops packets from going in circles forever. Each router lowers the number.
9. **Protocol**: Tells which protocol (like TCP or UDP) the data is using.
10. **Header Checksum**: Checks if the header is damaged.
11. **Source IP**: IP address of the sender.
12. **Destination IP**: IP address of the receiver.
13. **Options**: Extra info (not always used).

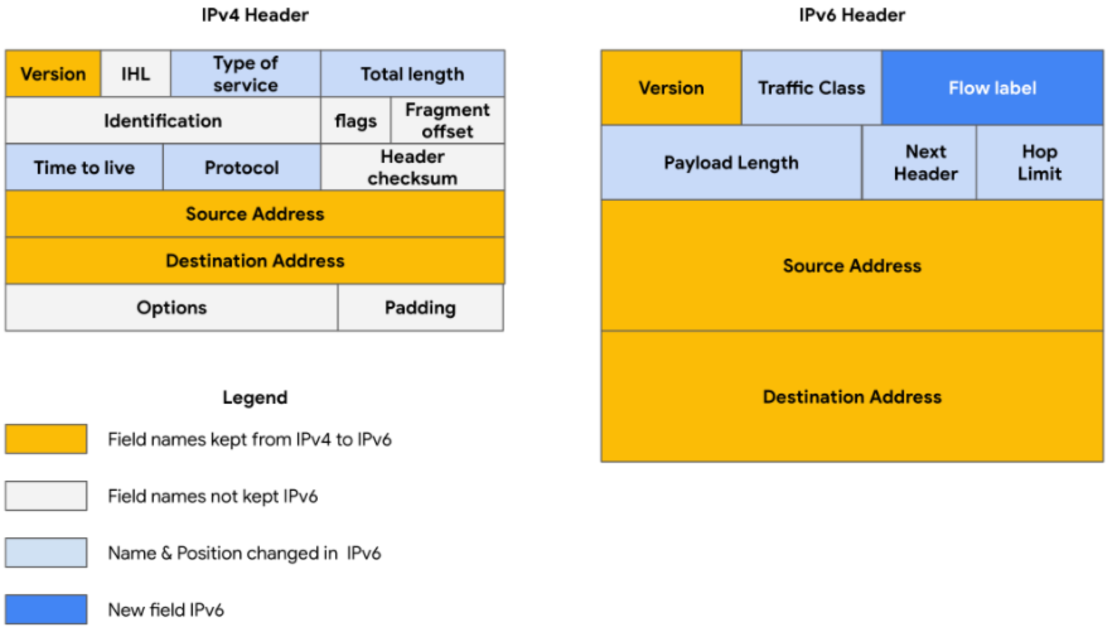
Format of an IPv4 packet





IPv4 vs. IPv6

Feature	IPv4	IPv6
Address format	Four numbers (0–255) with dots (e.g., 192.168.1.1)	Eight groups with colons (e.g., 2002:db8::1234)
Size	4 bytes (up to 4.3 billion addresses)	16 bytes (up to 340 undecillion addresses)
Header	More fields, more complex	Simpler, faster
Routing	Less efficient	More efficient and secure
Collisions	Can happen (private IPs may conflict)	No collisions with public IPs



Why this matters for security

- By reading packet headers, security experts can find out:
  - Where the packet came from.
  - Where it's going.
  - What kind of data it carries.
- This helps decide if a packet is **safe or a threat**.

Course wrap-up

• You learned about how networks are built, like **WANs (big networks)** and **LANs (small local networks)**.

• You studied tools used in networks like **hubs, switches, routers, and modems**.

• You were introduced to **cloud networks** and their **advantages**.

• You also learned about the **TCP/IP model**, which helps people in tech talk about where problems happen in a network.

Glossary terms from module 1

Terms and definitions from Course 3, Module 1

Bandwidth: The maximum data transmission capacity over a network, measured by bits per second

Cloud computing: The practice of using remote servers, application, and network services that are hosted on the internet instead of on local physical devices

Cloud network: A collection of servers or computers that stores resources and data in remote data centers that can be accessed via the internet

Data packet: A basic unit of information that travels from one device to another within a network

Hub: A network device that broadcasts information to every device on the network

Internet Protocol (IP): A set of standards used for routing and addressing data packets as they travel between devices on a network

Internet Protocol (IP) address: A unique string of characters that identifies the location of a device on the internet

Local Area Network (LAN): A network that spans small areas like an office building, a school, or a home

Media Access Control (MAC) address: A unique alphanumeric identifier that is assigned to each physical device on a network

Modem: A device that connects your router to the internet and brings internet access to the LAN

Network: A group of connected devices

Open systems interconnection (OSI) model: A standardized concept that describes the seven layers computers use to communicate and send data over the network

Packet sniffing: The practice of capturing and inspecting data packets across a network

Port: A software-based location that organizes the sending and receiving of data between devices on a network

Router: A network device that connects multiple networks together

Speed: The rate at which a device sends and receives data, measured by bits per second

Switch: A device that makes connections between specific devices on a network by sending and receiving data between them

TCP/IP model: A framework used to visualize how data is organized and transmitted across a network

Transmission Control Protocol (TCP): An internet communication protocol that allows two devices to form a connection and stream data

User Datagram Protocol (UDP): A connectionless protocol that does not establish a connection between devices before transmissions

Wide Area Network (WAN): A network that spans a large geographic area like a city, state, or country

Module 2

Network Operations

What you'll learn

- Network protocols
- Virtual private networks (VPNs)
- Firewalls, security zones, and proxy servers

Network Protocols

Network Protocols

Networks need **rules** so devices can talk to each other properly.
These rules are called **network protocols**.
They help decide **how data is sent**, **what order it's in**, and **how it's shaped** so that it reaches the right place.

Example: Visiting a Recipe Website

Let's say you want to visit a recipe site: www.yummyrecipesforme.org.

1. TCP (Transmission Control Protocol)

Your device first connects with the website's server using **TCP**.
TCP helps **set up a connection** and **stream data** between your computer and the website.
It also **checks** that both devices are real and ready to talk. This is called a **handshake**.

2. ARP (Address Resolution Protocol)

Once the handshake is done, your device starts **sending data packets** across the network.
These packets pass through devices like **routers**.
To know where to go next, your device uses **ARP** to find the **MAC address** of the next stop.

3. HTTPS (Hypertext Transfer Protocol Secure)

Now, to safely talk to the website, your browser uses **HTTPS**.
HTTPS **secures the data** you send and receive (like your login info).
It makes sure hackers can't read or steal the information.

4. DNS (Domain Name System)

Before you reach the site, your computer needs the site's IP address.
DNS takes the website name (www.yummyrecipesforme.org) and **finds its IP address**.
This IP address helps data packets reach the correct server.

Why Do Protocols Matter for Security?

Some protocols like **HTTPS** are **secure**.
HTTPS uses **SSL/TLS** to **encrypt** the data, meaning it turns it into a secret code.
This keeps your info safe from hackers while it travels across the internet.

Summary of Protocols Used

When you visit a website, your device uses **multiple protocols**:

- **TCP**: Makes the connection
- **ARP**: Finds the next device to send data to
- **HTTPS**: Secures your communication
- **DNS**: Finds the website's IP address

Common Network Protocols

What Are Network Protocols?

- **Definition**: A set of rules that define how data is structured and transmitted between devices.
- **Function**: Tell devices how to handle data in packets.
- **Security Note**: Some protocols have vulnerabilities that attackers can exploit.

3 Categories of Network Protocols

1. Communication Protocols

These handle the exchange of data between devices.

Protocol	Function	Layer	Port	Notes

TCP	Reliable connection using 3-way handshake (SYN, SYN/ACK, ACK)	Transport	—	Ensures all data is received
UDP	Fast, connectionless transmission	Transport	—	Less reliable, used for DNS, video calls
HTTP	Transfers web data (insecure)	Application	80	Being replaced by HTTPS
DNS	Resolves domain names to IPs	Application	53 (UDP), TCP if needed	Can be abused for redirection attacks

2. Management Protocols

Used to monitor and manage network performance.

Protocol	Function	Layer	Notes
SNMP	Monitors and configures network devices	Application	Can reset passwords or monitor bandwidth
ICMP	Reports errors, used in ping command	Internet	Helps in troubleshooting connectivity

3. Security Protocols

Protect data during transmission.

Protocol	Function	Layer	Port	Notes
HTTPS	Secure version of HTTP with encryption (SSL/TLS)	Application	443	Encrypts communication between browser and server
SFTP	Secure file transfer using SSH	Application	22	Uses encryption like AES; used in cloud storage

🔐 *Note*: These protocols **don't hide the IP addresses**, so attackers can still see source and destination.

Key Takeaways

- Network protocols allow devices to communicate and define how data flows.
- Security analysts should understand **common protocols** to:
  - Identify vulnerabilities
  - Detect abnormal use or misuse (e.g., DNS hijacking)
  - Strengthen defenses and troubleshoot issues

Additional Network Protocols Overview

These protocols are essential for network communication, device management, remote access, and email handling. Knowing their **functions**, **port numbers**, and **TCP/IP model layers** is critical for a cybersecurity analyst.

Core Protocols & Details

Protocol	Function	Port(s)	Layer	Notes
NAT (Network Address Translation)	Converts private IPs to a public IP for internet access	N/A	Internet (L2) & Transport (L3)	Used by routers/firewalls
DHCP	Assigns IPs & config info to devices	UDP 67 (server), UDP 68 (client)	Application	Used during device boot-up
ARP	Resolves IP → MAC addresses	None	Network Access (Layer 2)	Uses ARP cache
Telnet	Remote login to systems (insecure)	TCP 23	Application	Sends data in plain text
SSH	Secure remote login and command execution	TCP 22	Application	Encrypted; replaces Telnet
POP3	Downloads email (local storage)	TCP/UDP 110 (unencrypted), 995 (SSL/TLS)	Application	Does **not** sync across devices
IMAP	Syncs email across devices	TCP 143 (unencrypted), 993 (SSL/TLS)	Application	Keeps emails on the server
SMTP	Sends and routes outgoing emails	TCP/UDP 25 (unencrypted), 587 (TLS)	Application	Used with MTA; filters spam

🛡Security & Usage Notes

- **SSH vs. Telnet**: SSH is secure (encrypted), Telnet is not.
- **IMAP vs. POP3**: IMAP supports **multi-device sync**; POP3 does not.
- **NAT**: Helps preserve IPv4 addresses; required for private IPs to access the internet.
- **Port-based filtering**: Firewalls often block or allow traffic based on these port numbers.

Key TCP/IP Layers Recap

Layer	What It Covers
Application (Layer 4)	User-facing services (e.g., HTTP, DNS, SMTP)
Transport (Layer 3)	End-to-end communication (TCP, UDP)
Internet (Layer 2)	Logical addressing (IP, NAT)
Network Access (Layer 1)	Physical/MAC-level communication (ARP)

Memorization Tip

Group protocols by **function**:

- **Remote access**: SSH, Telnet
- **Email**: POP3, IMAP, SMTP
- **IP config**: DHCP, NAT
- **Address resolution**: ARP

Wireless Protocols

What is IEEE 802.11?

- **IEEE 802.11** = Standard for **Wi-Fi** (Wireless LANs)
- Created by **IEEE** (Institute of Electrical and Electronics Engineers)
- Defines how wireless devices communicate over short distances

Wi-Fi Security Protocols

Protocol	Year Introduced	Security Level	Notes
WEP (Wired Equivalent Privacy)	~1997	● Weak	Deprecated; easily cracked
WPA (Wi-Fi Protected Access)	2004	□ Improved	Temporary fix for WEP flaws
WPA2	2006	□ Strong	Industry standard for years (uses AES)

WPA3	2018	□🔒 Strongest	Better encryption, protection against brute-force attacks

Important Concepts for Analysts

- **WPA2 and WPA3** use **strong encryption algorithms (like AES)**
- **Always recommend using WPA3** when available
- Security analysts must ensure:
  - Proper encryption settings on routers/access points
  - No outdated protocols (like WEP or WPA) in use
  - Guest networks are isolated
  - Strong Wi-Fi passwords are enforced

Real-World Task Example

“As a security analyst, you might audit the organization's wireless network and enforce WPA3 encryption on all access points to reduce the risk of wireless attacks.”

The evolution of wireless security protocols

In the beginning, the internet only worked through cables. But in the mid-1980s, the US allowed some radio wave frequencies to be used without needing a license. This helped the internet grow without cables.

In the late 1990s and early 2000s, new technology came that allowed data to be sent and received using radio waves. Now, people use Wi-Fi on laptops, phones, tablets, and even smart devices like thermostats and security cameras.

What is Wi-Fi?

Wi-Fi is the name used for wireless internet. It follows certain rules (called standards) for how wireless internet should work. These rules are part of something called **IEEE 802.11**.

To keep Wi-Fi connections safe, security protocols are used. Over time, these protocols improved because older ones had problems. The main ones are **WEP, WPA, WPA2, and WPA3**.

WEP (Wired Equivalent Privacy) – 1999

- WEP was the first wireless security protocol.
- It was meant to give the same privacy as wired internet.
- WEP is no longer safe and is rarely used now.

- Hackers can easily break WEP, so it's considered high risk.

WPA (Wi-Fi Protected Access) – 2003

- WPA was made to fix the problems in WEP.
- It used a stronger method called **TKIP** to make guessing the password harder.
- It also had a system to check if messages were changed during transfer.
- But hackers found a new way to attack it using something called a **KRACK attack**, so WPA became outdated.

WPA2 – 2004

- WPA2 replaced WPA and is still widely used today.
- It uses stronger encryption called **AES**.
- It also uses **CCMP** to make messages safe and unchanged.
- But WPA2 is still weak against KRACK attacks.

WPA2 Personal (for home):

- Easy to set up.
- All devices use the same Wi-Fi password.
- Good for homes, but not suitable for businesses.

WPA2 Enterprise (for business):

- More secure and professional.
- Each user has their own login.
- Admins can allow or block users at any time.
- Users never see the Wi-Fi encryption keys.

WPA3 – 2018

- WPA3 is the latest and safest Wi-Fi protocol.
- Fixes the KRACK problem using **SAE** (a safer way to share passwords).
- Makes it harder for hackers to decode saved Wi-Fi data.
- Uses **128-bit encryption** for stronger passwords (192-bit for enterprise).

Final points:

As a security analyst, it’s important to know how these protocols changed over time. You need to know their weaknesses and always use the latest, most secure protocol (WPA3 if possible) to protect wireless networks.

System Identification

Firewalls and network security measures

What is a Firewall?

A **firewall** is like a **security guard** for your network. It **watches the internet traffic** coming in and going out of your computer or company network.

- It decides what is **allowed in or out** based on some **rules**.
- These rules are made by your company’s **security team**.
- Example: It may only allow websites that use **port 443 (HTTPS)** and **block** others.

Types of Firewalls

1. **Hardware Firewall**
    - A **physical device**.
    - It checks every **data packet** (piece of data) before it can enter the network.
    - It's like a **fence with a gatekeeper**.
2. **Software Firewall**
    - A **program** installed on your **computer or server**.
    - It checks the traffic coming **to your computer or server**.
    - It’s cheaper and doesn’t take space, but it uses **some of your computer’s power**.
3. **Cloud-Based Firewall**
    - Not on your device – it lives in the **cloud (internet)**.
    - You can create rules from the **cloud provider's website**.
    - It protects both your **local network** and any **services you use in the cloud**.

How Firewalls Think: Stateful vs Stateless

1. **Stateful Firewall**:
    - **Smart firewall**.
    - It **remembers** data from before and checks for **weird or dangerous behavior**.
    - It can stop attacks because it understands what's happening.
2. **Stateless Firewall**:
    - **Basic firewall**.
    - It just follows **simple rules** (e.g., block port 80).

  - It **doesn’t remember anything** or notice patterns.
  - Less secure than stateful.

Next Generation Firewall (NGFW)

- It’s a **super-smart firewall**.
- It includes:
  - **Stateful inspection** (remembers traffic).
  - **Deep packet inspection** (looks inside the data).
  - **Intrusion protection** (stops hacking attempts).
  - Connects to **cloud intelligence** to learn about **new threats quickly**.

Summary

- **Firewalls** protect networks by **blocking or allowing traffic**.
- Types: **Hardware, Software, Cloud-based**.
- They can be **Stateless** (basic) or **Stateful** (smart).
- **Next Generation Firewalls** offer the **strongest protection**.

Virtual Private Networks (VPNs)

What is a VPN?

A **VPN (Virtual Private Network)** is a tool that **protects your privacy** when you're using the internet.

When you connect to a website, your device sends **requests** through your **Internet Service Provider (ISP)**. These requests include your:

- **IP address**
- **Location**
- Sometimes **sensitive info** like bank logins or credit card numbers

Without protection, someone (like a hacker) could **see or steal** this data if they intercept your traffic.

What Does a VPN Do?

A VPN does **two major things** to protect you:

1. Changes Your IP Address

- Hides your **real location**
- Makes it **harder to track you**

2. Encrypts Your Data

- Scrambles your data so it's **unreadable** to anyone trying to spy
- Even if someone intercepts it, they **can't understand it**

What is Encapsulation?

VPN uses a process called **encapsulation** to protect your data.

- Normally, data packets have **headers and footers** that show your IP and MAC address.
- That's a problem because hackers can **see where you're located**.
- If you just encrypt the data, **routers can't read it**, so your request won't go through.

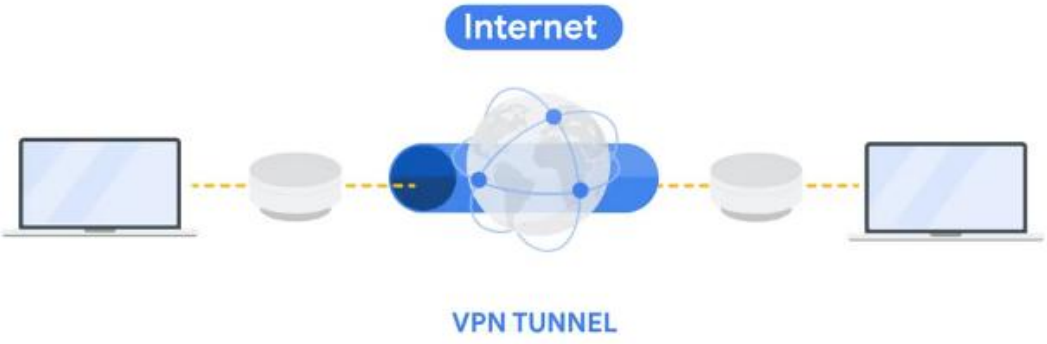
Encapsulation fixes this:

- Your **original data is encrypted** (so no one can read it)
- Then it is **wrapped inside a new packet** with readable header info
- So routers can still **send it to the right place**, but your actual info stays **hidden and safe**

What is a VPN Tunnel?

A **VPN tunnel** is a **secure and encrypted path** between your device and the VPN server.

- Your data travels through this tunnel
- It's **encrypted** using cryptographic keys
- No one can see what's inside the tunnel — not your ISP, not a hacker, not even the government (without the key)



Why Use a VPN?

- **Stay private** online
- **Hide your location and IP**
- **Protect your sensitive data**
- **Secure your internet connection**, especially on **public Wi-Fi**

Simple Flow of How VPN Works:

1. You open your browser and visit a site
2. Your VPN:
    - Hides your real IP
    - Encrypts your data
    - Wraps (encapsulates) it
    - Sends it through a **secure tunnel**
3. The VPN server sends your request to the website
4. The website replies to the VPN server
5. The VPN server sends the reply **back to you**, encrypted and protected

Security Zones

Security zones are segments of a network that are separated to **protect sensitive areas** and **control who can access what**. This concept is part of **network segmentation**, which helps in **containing threats** and **improving security**.

Why Use Security Zones?

- **Limit access** to sensitive resources.
- **Prevent malware** or attacks from spreading across the entire network.
- **Separate public from private systems** (e.g., hotel guests vs. hotel staff).
- Help security teams apply **different rules to different areas** of the network.

Network Segmentation Example

Imagine a hotel:

- **Guest Wi-Fi** = open, public.
- **Staff network** = secure, private.
  These are **segmented**, so if someone hacks the guest network, they **can't access staff data**.

Subnetting Example in an Organization

In a **university**:

- **Faculty subnet** for professors and admin.
- **Student subnet** for students.
  If students' devices are infected, the admin can **contain it** without affecting faculty.

Types of Security Zones

There are **two main categories**:

1. Uncontrolled Zone

- Anything **outside** the organization’s control.
- Example: the **internet**.

2. Controlled Zone

- Internal segments protected from the uncontrolled zone.

There are **3 layers** within the Controlled Zone:

Demilitarized Zone (DMZ)

- Also called the **perimeter network**.
- Public-facing services go here:
  - **Web servers**
  - **Proxy servers**
  - **DNS servers**
  - **Email & file servers** handling external communication
- Connects to the **internet** but is separated from the internal network.
- Acts as the **first line of defense**.

Example: A web server hosting the company’s public website lives in the DMZ.

Internal Network

- Where the **organization's private servers and data** reside.
- Only accessible to internal users.
- Protected behind a **firewall** that separates it from the DMZ.

Example: HR system, file storage for employees.

Restricted Zone

- For **highly confidential data**.
- Only **authorized personnel** with special privileges can access.
- Protected by **another firewall**, even stricter than the one guarding the internal network.

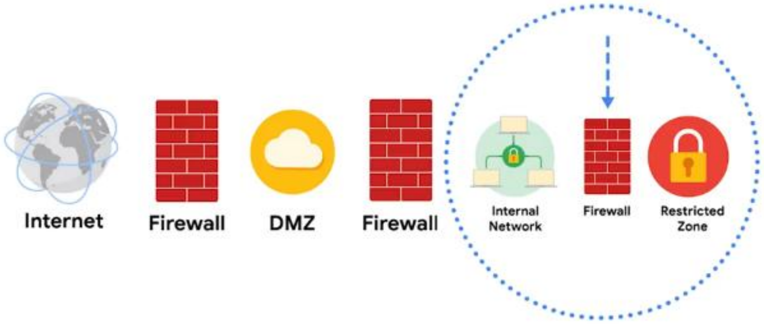
Example: Financial records, top-secret product data, employee PII.

How Firewalls Protect Zones

- **Firewall 1**: Between Internet ↔ DMZ (filters incoming traffic).
- **Firewall 2**: Between DMZ ↔ Internal Network (extra layer of filtering).
- **Firewall 3 (optional)**: Between Internal ↔ Restricted Zone.

This setup provides:

- **Multiple layers of defense**
- Attackers must pass **several firewalls** to reach critical data.



As a **Security Analyst**, you may:

- Create and enforce **firewall rules**.
- Control **access policies** to:
  - Allow only safe protocols (e.g., HTTPS).
  - Block unnecessary ports.
  - Whitelist specific IPs.
- Ensure **only trusted users** reach internal or restricted areas.

Summary

Term	Description
Security Zone	A part of a network with its own security rules.
Network Segmentation	Dividing a network to isolate parts and control access.
DMZ (Demilitarized Zone)	Public-facing area of the network (e.g., web servers).
Internal Network	Organization’s private systems.
Restricted Zone	Sensitive data, very limited access.
Firewall	Filters traffic between zones.
Uncontrolled Zone	External (e.g., internet), not trusted.

Subnetting and CIDR

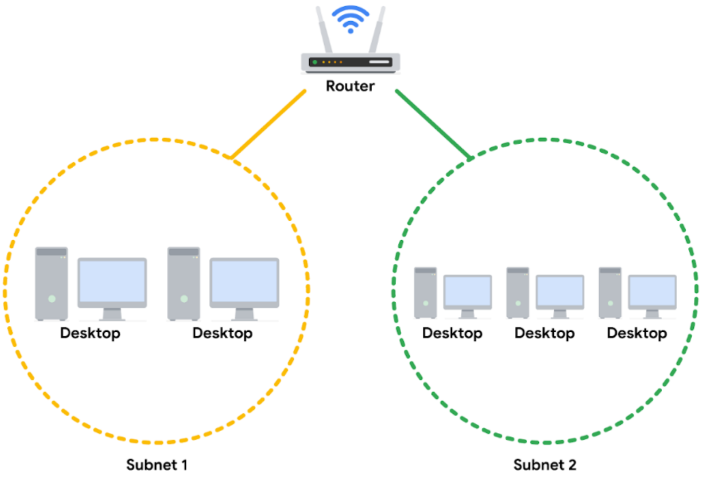
Subnetting and CIDR

Earlier in this course, you learned about network segmentation, a security technique that divides networks into sections. A private network can be segmented to protect portions of the network from the internet, which is an unsecured global network.

For example, you learned about the uncontrolled zone, the controlled zone, the demilitarized zone, and the restricted zone. Creating security zones is one example of a networking strategy called **subnetting**.

Overview of Subnetting

- **Subnetting** is the subdivision of a network into logical groups called **subnets**.
- It works like a network inside a network.
- Subnetting divides up a network address range into smaller subnets within the network.
- These smaller subnets form based on the IP addresses and network mask of the devices on the network.
- Subnetting creates a network of devices to function as their own network. This makes the network more efficient and can also be used to create security zones.
- If devices on the same subnet communicate with each other, the switch changes the transmissions to stay on the same subnet, improving speed and efficiency of the communications.



Classless Inter-Domain Routing (CIDR) Notation for Subnetting

- **Classless Inter-Domain Routing (CIDR)** is a method of assigning subnet masks to IP addresses to create a subnet. **Classless addressing** replaces **classful addressing**.
- Classful addressing was used in the 1980s as a system of grouping IP addresses into classes (Class A to Class E).
- Each class included a limited number of IP addresses, which were depleted as the number of devices connecting to the internet outgrew the classful range in the 1990s.
- Classless CIDR addressing expanded the number of available IPv4 addresses.

CIDR allows cybersecurity professionals to segment classful networks into smaller chunks. CIDR IP addresses are formatted like IPv4 addresses, but they include a **slash ("/") followed by a number** at the end of the address. This extra number is called the **IP network prefix**.

For example:

- A regular IPv4 address: `198.51.100.0`
- A CIDR IP address: `198.51.100.0/24`

This CIDR address encompasses all IP addresses between `198.51.100.0` and `198.51.100.255`.

The system of CIDR addressing:

- Reduces the number of entries in routing tables
- Provides more available IP addresses within networks

You can try converting CIDR to IPv4 addresses and vice versa through an online conversion tool like **IPAddressGuide** for practice and to better understand this concept.

Note: You may learn more about CIDR during your career, but it won't be covered in any additional depth in this certificate program. For now, you only need a basic understanding of this concept.

Security Benefits of Subnetting

Subnetting allows network professionals and analysts to create a network within their own network **without requesting another network IP address** from their internet service provider. This process uses **network bandwidth more efficiently** and improves **network performance**. Subnetting is one component of creating isolated subnetworks through:

- Physical isolation
- Routing configuration
- Firewalls

Key Takeaways

- **Subnetting** is a common security strategy used by organizations.
- Subnetting allows organizations to **create smaller networks within their private network**.
- This improves the **efficiency** of the network and can be used to **create security zones**.

Proxy Servers

What is a Proxy Server?

A **proxy server** is a system that helps keep a network safe. It sits **between the internet and your private network**. When someone from outside tries to connect, the proxy checks if the request is safe.

It uses a **public IP address** that's different from the rest of your internal (private) network. This **hides the real IP addresses** inside your organization, making it harder for hackers to reach your internal devices.

How It Works (With Example)

Imagine a person tries to access your company's website. Instead of seeing the **real IP address** of your server, they'll see the **proxy server's IP** or nothing at all. This protects your actual server.

Proxy servers can also **block bad websites** that employees shouldn't visit.

They use **temporary memory (called cache)** to store commonly requested data. This means it doesn't have to always go to your internal server to get it. It's **faster** and **safer**, since fewer direct connections are made to the internal server.

Types of Proxy Servers

1. **Forward Proxy Server**
    - Controls what people inside the network can access on the internet.
    - It **hides the user's IP address**.
    - It **checks and approves outgoing requests** from users before allowing access to websites.
2. **Reverse Proxy Server**
    - Protects the **internal servers** from people on the internet.
    - It **hides the real IP of the internal server**.
    - It **checks incoming traffic** from the internet before passing it to internal web servers.
    - Useful when internal servers hold **sensitive information**.
3. **Email Proxy Server**
    - Helps to **filter spam** and block fake or dangerous emails.
    - It checks if an email sender is real or pretending to be someone else.
    - Helps **prevent phishing attacks** that try to trick employees.

Real-Life Example
At a big internet company in the US, they used an email proxy server to **scan emails** before they reached users. About **95% of emails were spam**, and the proxy helped **block them before delivery**. This also kept the actual email system fast and clean.

Why Proxy Servers Are Important

Proxy servers help in:

- **Filtering traffic**
- **Hiding IP addresses**
- **Blocking spam and dangerous websites**
- **Protecting internal systems**
- **Reducing direct contact with the internet**

They are a **key part of network security** and help stop hackers and bad traffic from getting into your organization's private network.

Virtual Networks and Privacy

This part of the course explained **how networks work and how to keep them safe.** Keeping a network secure means making sure only trusted people can access it and that private information stays safe.

we'll go over some important security tools and ideas you've already seen: **VPNs (Virtual Private Networks), proxy servers, firewalls, and security zones**. You'll learn how they work and how they're connected.

Common Network Protocols

Network protocols are like rules that help computers send data to the right place. These rules help computers understand how to talk to each other over a network.

There are **three main types** of protocols:

- **Communication protocols:** Help devices connect and talk to each other.
  Examples:
  - **TCP** (Transmission Control Protocol)
  - **UDP** (User Datagram Protocol)
  - **SMTP** (Simple Mail Transfer Protocol) – used to send emails
- **Management protocols:** Help find and fix network problems.
  Example:
  - **ICMP** (Internet Control Message Protocol) – used to send error messages like "host not reachable"
- **Security protocols:** Help protect data while it's being sent.
  Examples:
  - **IPSec** (Internet Protocol Security)
  - **SSL/TLS** (Secure Sockets Layer / Transport Layer Security)

Other commonly used protocols:

- **HTTP (HyperText Transfer Protocol):** Helps web browsers and websites communicate.
- **DNS (Domain Name System):** Converts website names (like google.com) into IP addresses.
- **ARP (Address Resolution Protocol):** Helps find the physical (MAC) address of a device on a local network using its IP address.

Wi-Fi

You also learned about **wireless security protocols**, including:

- **WEP** (old and not very secure)
- **WPA**
- **WPA2**
- **WPA3** (most secure)

WPA3 uses **AES encryption**, which scrambles the data so it can't be read by others. WPA2 and WPA3 have two types:

- **Personal mode** – for home networks
- **Enterprise mode** – for businesses and large organizations

Network Security Tools and Practices

Firewalls

A **firewall** is a device or software that watches and controls traffic going in and out of your private network. It checks if the traffic is allowed based on rules set by the administrator.

There are **two main types** of firewalls:

- **Stateless firewall:** Uses fixed rules. It doesn't remember anything about past data packets.
- **Stateful firewall:** Remembers past traffic and only needs one rule (because it knows the connection). It uses a **state table** to track ongoing connections.

Next-Generation Firewalls (NGFWs) offer more advanced protection. They:

- Can inspect the contents of data packets (called **deep packet inspection**)
- Can detect attacks and alert security teams (**intrusion prevention**)
- Can look at **application-level traffic**, not just IPs and ports
- May include tools like:
  - **Malware sandboxing** (test files in a safe space)
  - **Network antivirus**
  - **URL and DNS filtering**

Proxy Servers

A **proxy server** adds an extra layer of security between your internal network and the internet. It hides internal systems and controls what traffic goes in or out using **NAT (Network Address Translation)**.

There are two types:

- **Forward proxy:** Helps internal users access the internet.
- **Reverse proxy:** Helps external users connect to services inside your network (like a company website).

Proxies can also block dangerous websites by setting **filters**, like a firewall.

Virtual Private Networks (VPN)

A **VPN** hides your IP address and encrypts your data while it travels over the internet. It does this using **encapsulation**, which means wrapping your data in a secure packet.

VPNs are used by:

- **Companies** – to protect employee access to company servers and apps
- **People** – to protect their privacy online

A **good VPN** will:

- Hide your identity from outside websites
- Encrypt your internet traffic
- Not track your activity

Many companies also use **SD-WAN (Software-Defined Wide Area Network)** along with VPNs. SD-WAN helps connect users to apps across many places (even far apart) in a secure way.

Key Takeaways

- **Three main types of protocols**: communication, management, and security
- You learned about **firewalls, proxy servers, and VPNs**
- Many organizations use **cloud-based security** by combining VPN and SD-WAN features

VPN Protocols: WireGuard and IPSec

- A **VPN** (Virtual Private Network) is a service that hides your real IP address and online location.
- It helps you stay private and safe when using public networks like the internet.
- A VPN works through a **server** that acts like a **middle point** between your computer and the internet.
- This server creates a **virtual tunnel** that hides your computer's IP address and **encrypts** the data (so no one can read it) as it travels to the internet.
- The main goal of a VPN is to create a **safe and private connection** between your device and the network. It also lets you make trusted connections even when using untrusted networks (like public Wi-Fi).

VPN protocols are the set of rules that decide **how this secure tunnel works**. Different VPN services use different VPN protocols.

Remote Access and Site-to-Site VPNs

- **Remote Access VPNs** are used by individual users to connect their own device (like a laptop or phone) to a VPN server. These VPNs **encrypt** the data sent or received on that device. The connection is made through the **internet**.
- **Site-to-Site VPNs** are used by **companies** to connect one network to another in different locations. For example, if a company has offices in multiple cities or countries, this VPN helps them all stay connected. **IPSec** is often used to create the encrypted tunnel between the main office and remote offices. The downside of site-to-site VPNs is that they are **more difficult to set up and manage** than remote access VPNs.

WireGuard VPN vs. IPSec VPN

Both **WireGuard** and **IPSec** are VPN protocols. They are used to **protect data** that is sent through the secure VPN tunnel. Different VPN services offer these options depending on what you need.

WireGuard VPN

- WireGuard is a **fast VPN protocol** with strong encryption.
- It is **simple to set up** and use.
- It can be used for both **remote access** and **site-to-site** connections.
- WireGuard is **newer** than IPSec.
- It uses **less code**, which makes it **faster to download** and easier to fix problems.
- It is also **open source**, meaning anyone can look at or improve its code.
- WireGuard is good for things like **watching videos or downloading large files** quickly.

IPSec VPN

- IPSec is an **older** VPN protocol.
- It is used by many VPN services to **encrypt and verify** data.
- Most operating systems **already support** IPSec.
- IPSec is more **complex** than WireGuard.
- Some people prefer IPSec because it has been around longer, has gone through **more security testing**, and is widely trusted.
- Others prefer WireGuard because it's **faster and easier** to set up.

Key Takeaways

- A **VPN protocol** is like a **network protocol**: It's a set of rules that decides how data moves between devices on a network.
- There are **two main types of VPNs**:
  - **Remote access VPNs**: connect a personal device to a VPN server and encrypt its data.
  - **Site-to-site VPNs**: connect entire networks together, often used by companies with offices in different places.
- **IPSec** is commonly used in **site-to-site VPNs**.

- **WireGuard** can be used for both **site-to-site** and **remote access VPNs**.

Course wrap-up

1. Common Network Protocols

We covered some basic but essential protocols:

- **TCP (Transmission Control Protocol)**: Helps send data reliably between computers.
- **ARP (Address Resolution Protocol)**: Maps IP addresses to MAC (physical) addresses inside a local network.
- **HTTPS (Hypertext Transfer Protocol Secure)**: Secures web communication by encrypting data between the browser and website.
- **DNS (Domain Name System)**: Translates website names (like google.com) into IP addresses computers understand.

2. Virtual Private Networks (VPNs)

- **VPNs** help **hide your location and IP address**, especially when using public Wi-Fi.
- They create a **secure "tunnel"** for your internet data.
- **Two main VPN types**:
  - **Remote Access VPN**: Used by individuals to connect their personal device to a VPN server.
  - **Site-to-Site VPN**: Used by companies to securely connect different office networks.
- **Two common VPN protocols**:
  - **WireGuard**: Fast, modern, and simple to use.
  - **IPSec**: Older, widely used, and very secure.

3. Firewalls and Security Zones

- **Firewalls**: Act like security guards, allowing or blocking traffic based on set rules.
  - **Stateless firewalls** do **not** remember previous traffic.
  - **Stateful firewalls** keep track of active connections and are smarter at filtering.
- **Security Zones**:
  - **Restricted Zone**: Highest level of protection. Only trusted employees can access.
  - **Management Zone**: For internal control systems.
  - **DMZ (Demilitarized Zone)**: Where public-facing services (like websites) are placed.
  - **Uncontrolled Zone**: Open to the public (like the internet).

4. Proxy Servers

- **Reverse Proxy Server**: Sits in front of internal servers and **filters incoming traffic from the internet**. Only safe traffic reaches the internal server.
- **Forward Proxy Server**: Sits between a user and the internet. Hides user identity and helps enforce access rules.

Final Note

Network operations involve many tools and techniques to ensure everything runs **smoothly and securely**. These concepts are crucial for any role in cybersecurity, especially as a **security analyst**.

Glossary terms from module 2

Terms and definitions from Course 3, Module 2

Address Resolution Protocol (ARP): A network protocol used to determine the MAC address of the next router or device on the path

Cloud-based firewalls: Software firewalls that are hosted by the cloud service provider

Controlled zone: A subnet that protects the internal network from the uncontrolled zone

Domain Name System (DNS): A networking protocol that translates internet domain names into IP addresses

Encapsulation: A process performed by a VPN service that protects your data by wrapping sensitive data in other data packets

Firewall: A network security device that monitors traffic to or from your network

Forward proxy server: A server that regulates and restricts a person's access to the internet

Hypertext Transfer Protocol (HTTP): An application layer protocol that provides a method of communication between clients and website servers

Hypertext Transfer Protocol Secure (HTTPS): A network protocol that provides a secure method of communication between clients and servers

IEEE 802.11 (Wi-Fi): A set of standards that define communication for wireless LANs

Network protocols: A set of rules used by two or more devices on a network to describe the order of delivery of data and the structure of data

Network segmentation: A security technique that divides the network into segments

Port filtering: A firewall function that blocks or allows certain port numbers to limit unwanted communication

Proxy server: A server that fulfills the requests of its clients by forwarding them to other servers

Reverse proxy server: A server that regulates and restricts the internet's access to an internal server

Secure File Transfer Protocol (SFTP): A secure protocol used to transfer files from one device to another over a network

Secure shell (SSH): A security protocol used to create a shell with a remote system

Security zone: A segment of a company's network that protects the internal network from the internet

Simple Network Management Protocol (SNMP): A network protocol used for monitoring and managing devices on a network

Stateful: A class of firewall that keeps track of information passing through it and proactively filters out threats

Stateless: A class of firewall that operates based on predefined rules and does not keep track of information from data packets

Subnetting: The subdivision of a network into logical groups called subnets

Transmission Control Protocol (TCP): An internet communication protocol that allows two devices to form a connection and stream data

Uncontrolled zone: The portion of the network outside the organization

Virtual private network (VPN): A network security service that changes your public IP address and masks your virtual location so that you can keep your data private when you are using a public network like the internet

Wi-Fi Protected Access (WPA): A wireless security protocol for devices to connect to the internet

Module 3

Secure against Network Intrusions

Introduction to Network Intrusion tactics

You're now going to learn how to **secure networks** so that **important and private information stays safe**.

In this section, you'll understand:

- **What kind of attacks** can happen on a network (called *network intrusion tactics*).
- **How a security analyst** can protect the network from those attacks.

The Case for Securing Networks

Why do we need to secure networks?
Networks are always at risk of being attacked by hackers. Attackers can break into networks using various methods, including:

- **Malware** – harmful software that damages or steals data.
- **Spoofing** – pretending to be a trusted source to gain access.
- **Packet sniffing** – secretly capturing data as it travels over the network.
- **Packet flooding** – sending large volumes of data to crash the system or disrupt operations.

Why is protecting the network important?
Even one successful attack can cause serious harm to an organization. These impacts include:

- Leakage of confidential or valuable information.
- Damage to the organization's reputation.
- Loss of customer trust and loyalty.
- Financial loss and wasted time due to recovery efforts.

Real-world example of a cyberattack
In 2014, the American home-improvement chain **Home Depot** was targeted by hackers. The attackers infected Home Depot's servers with malware. Before the IT team could stop the attack, the hackers had stolen credit and debit card information of over **56 million customers**.

How Intrusions Compromise Your System

Every network has weaknesses
All networks have some weaknesses (called vulnerabilities), and attackers can use them to harm the system. These attacks can happen for different reasons. The attacker might:

- Want to steal money.
- Have personal or political reasons.
- Be a former or current angry employee.
- Be an activist who disagrees with the organization.

So, **any network can be attacked**. Security analysts must always be alert, find weaknesses, and act fast to fix them.

Network Interception Attacks

These attacks happen when a hacker **intercepts (catches)** the data that is moving through the network. They either **steal the data** or **change it**.

For example:

- Attackers can use tools (hardware or software) to watch data as it travels. This is called **packet sniffing**.
- They can **see private information** or **change it** to cause problems.
- An attacker might **change the details** of a bank transfer, so the money goes to their own account.

Later in this course, you'll learn about more types of network interception attacks like:

- **On-path attacks** (also known as Man-in-the-Middle)
- **Replay attacks**

Backdoor Attacks

A **backdoor attack** is when someone **gains access to a system without using the regular method**.

Example:
Imagine a building with many security measures (like cameras and biometric scanners). But someone finds a secret door (backdoor) that's not watched and uses it to get in or out without being seen.

In cybersecurity:

- A **backdoor** is a secret way into a system.
- Sometimes programmers or system admins add these on purpose to fix problems easily.
- But hackers can also **install backdoors** after they break in, so they can **come back anytime**.

Once inside through a backdoor, the hacker can:

- Install **malware** (harmful software),
- Launch a **DoS (Denial of Service) attack**, which floods the system with traffic and shuts it down,
- **Steal private data**,
- Or **change security settings** to make the system even weaker.

Possible Impacts on an Organization

1. Financial Loss
When a system is under attack (like with a DoS), the company can't perform normal tasks, which **stops them from making money**.

- Some companies can lose **millions of dollars**.
- They may need to **spend a lot** to fix the system and pay if data is stolen.
- If customer data is leaked, the company might face **lawsuits** or **legal settlements**.

2. Damage to Reputation
If people find out the company was attacked, they may:

- Stop trusting the company,
- Be scared to give their personal info,
- And choose to **go to competitors** instead.

3. Public Safety Risks
If attackers hit **government systems**, it can affect **citizen safety**.
Example impacts:

- Power grids,
- Water systems,
- Military communication systems.

If these are attacked, people could face **real-world harm**.

Key Takeaways

- Attackers are always **looking for new weaknesses** to exploit.

- They use **network interception** and **backdoor attacks** to get into systems.
- Once in, they can cause **financial damage**, **ruin reputations**, and even **threaten public safety**.
- It’s important for **security analysts** to keep learning so they can **protect the network** and reduce risks.

Securing networks has never been more important.

Secure Networks against DOS attacks

Denial of Service (DoS) Attacks

A **Denial of Service (DoS)** attack is when someone tries to make a network or server stop working by sending it **too much traffic**. The goal is to **overload** the system so that **normal users can’t use it**. This stops business activities, which leads to **loss of money, time, and reputation**. Also, when a network crashes, it can be **open to more attacks**.

Distributed Denial of Service (DDoS) Attacks

A **Distributed Denial of Service (DDoS)** attack is a type of DoS attack, but instead of one device, it uses **many devices** from different places to send traffic. Since traffic comes from multiple sources, it's **harder to block** and **easier to overwhelm** the server.

For example: An attacker sends a carefully designed packet that **confuses a router**, making it take extra time to process. Even though there’s not much traffic, the **router gets stuck** because of the tricky packet.

Network-Level DoS Attacks

These attacks target **network bandwidth**—the internet “space” a network has to send and receive data. Here are three common types:

1. SYN Flood Attack

This attack targets the **TCP three-way handshake**, which is used to make a connection between a device and a server.

- Step 1: Device sends a **SYN (synchronize)** request.
- Step 2: Server replies with **SYN/ACK** and opens a port.
- Step 3: Device sends back **ACK**, and connection is made.

In a **SYN Flood Attack**, the attacker sends **many SYN requests** but never completes the handshake. If too many ports are left waiting for the last step, the server **runs out of resources** and **crashes**.

2. ICMP Flood Attack

ICMP (Internet Control Message Protocol) is used to check if devices on the network are working properly.

In an **ICMP Flood Attack**:

- The attacker sends **lots of ICMP packets** to a server.
- The server tries to respond to each one, using up **all its bandwidth**.
- Eventually, the server **crashes** due to overload.

3. Ping of Death Attack

Normally, an ICMP (ping) packet should be **less than 64 KB** in size.

In a **Ping of Death Attack**:

- The attacker sends a **very large ping packet** (bigger than 64 KB).
- The system receiving it can't handle it and **crashes**.

Example: Imagine an ant colony where each ant can carry food. If someone drops a huge rock on the colony, it **destroys everything**. The ants can't work anymore. That's what happens to a system hit with a Ping of Death.

Summary

- **DoS Attacks**: Send too much traffic to a server to make it stop working.
- **DDoS Attacks**: Same goal, but using **multiple devices**.
- **SYN Flood**: Tricks the server by **starting connections but never finishing them**.
- **ICMP Flood**: Sends many status requests to **use up network bandwidth**.
- **Ping of Death**: Sends **oversized packets** that cause the system to **crash**.

tcpdump logs

A **network protocol analyzer**, also known as a **packet sniffer** or **packet analyzer**, is a tool used to **capture and analyze data traffic** on a network. Cybersecurity professionals use these tools to **monitor activity and detect suspicious behavior**.

There are many protocol analyzers available. Some common ones include:

- **SolarWinds NetFlow Traffic Analyzer**
- **ManageEngine OpManager**
- **Azure Network Watcher**
- **Wireshark**
- **tcpdump**

This reading focuses on **tcpdump**, but what you learn here can apply to other analyzers as well.

What is tcpdump?

tcpdump is a **command-line network analyzer**. It is:

- **Lightweight** (uses little memory and CPU)
- Uses the **open-source libpcap library**
- **Text-based**, so it runs entirely in the terminal
- Pre-installed on many **Linux** systems, and also works on **macOS**

tcpdump gives you a **quick overview of network traffic**. It shows information like:

- Source IP address
- Destination IP address
- Source and destination **port numbers**

How tcpdump shows information

When you run a tcpdump command, it shows the **packets** it captures right in your terminal or saves them to a **log file**. Each line in the output represents a packet and shows important network details.

Here’s what each part means:

- **Timestamp**: When the packet was captured (in hours, minutes, seconds, fractions).
- **Source IP**: Where the packet came from.
- **Source port**: The port on the sender’s side.
- **Destination IP**: Where the packet is going.
- **Destination port**: The port on the receiving side.

Note: By default, **tcpdump will try to convert IP addresses to hostnames**, and **port numbers to common service names** (like port 80 to HTTP).

What is tcpdump used for?

Network protocol analyzers like tcpdump are useful for:

- **Monitoring traffic** and viewing communications between devices
- **Troubleshooting** performance issues
- **Detecting unusual activity** on the network
- **Creating alerts** for potential threats
- **Finding unauthorized apps** like instant messengers or hidden Wi-Fi access points

But attackers can also use tcpdump to capture sensitive data such as usernames and passwords. So it's important to understand both its **benefits** and **risks**.

Key takeaways

- A **network protocol analyzer** like **tcpdump** helps monitor and investigate network traffic.
- **tcpdump** runs on **Linux/Unix** and **macOS**, and it works in the terminal (command line).
- The output includes: **timestamp**, **source IP and port**, **destination IP and port**.
- While **helpful for defenders**, these tools can also be used by **attackers** to steal sensitive information.

Real-Life DDoS Attack

Previously, you were introduced to **Denial of Service (DoS)** attacks. You also learned that **volumetric distributed DoS (DDoS)** attacks overwhelm a network by sending unwanted data packets in such large quantities that the servers become unable to service normal users. This can be **detrimental to an organization**. When systems fail, organizations cannot meet their customers' needs. They often **lose money**, and in some cases, incur **other losses**. An organization’s **reputation may also suffer** if news of a successful DDoS attack reaches consumers, who then question the security of the organization.

In this reading, you’ll learn about a **2016 DDoS attack against DNS servers** that caused major outages at multiple organizations that have millions of daily users.

A DDoS Targeting a Widely Used DNS Server

In previous videos, you learned about the **function of a DNS server**. As a review, **DNS servers** translate website domain names into the **IP address** of the system that contains the information for the website.

For instance, if a user were to type in a website URL, a DNS server would translate that into a numeric IP address that **directs network traffic** to the location of the website’s server.

On the day of the DDoS attack we are studying, **many large companies were using a DNS service provider**. The service provider was **hosting the DNS system** for these companies. This meant that when internet users typed in the URL of the website they wanted to access, their devices would be directed to the right place.

On **October 21, 2016**, the service provider was the victim of a **DDoS attack**.

Leading Up to the Attack

Before the attack on the service provider, a group of **university students created a botnet** with the intention to attack various **gaming servers and networks**.

A **botnet** is a collection of **computers infected by malware** that are under the control of a single threat actor, known as the "**bot-herder**." Each computer in the botnet can be **remotely controlled** to send a data packet to a target system. In a botnet attack, cyber criminals **instruct all the bots** on the botnet to send data packets to the target system at the same time, resulting in a **DDoS attack**.

The group of students **posted the code for the botnet online** so that it would be accessible to thousands of internet users and authorities wouldn’t be able to trace the botnet back to the students.

In doing so, they made it possible for other **malicious actors** to learn the code to the botnet and control it remotely. This included the **cyber criminals who attacked the DNS service provider**.

The Day of Attack

At **7:00 a.m.** on the day of the attack, the **botnet sent tens of millions of DNS requests** to the service provider. This **overwhelmed the system** and the DNS service shut down.

This meant that all of the websites that used the service provider **could not be reached**. When users tried to access various websites that used the service provider, they were not directed to the website they typed in their browser.

Outages occurred all over North America and Europe for each affected web service.

The service provider's systems were **restored after only two hours of downtime**. Although the cyber criminals sent **subsequent waves of botnet attacks**, the DNS company was prepared and able to **mitigate the impact**.

Key Takeaways

As demonstrated in the above example, **DDoS attacks can be very damaging** to an organization. As a **security analyst**, it's important to acknowledge the seriousness of such an attack so that you're aware of opportunities to protect the network from them.

If your network has **important operations distributed across hosts** that can be dynamically scaled, then **operations can continue** if the baseline host infrastructure goes offline.

DDoS attacks are damaging, but there are **concrete actions** that security analysts can take to help protect their organizations.

Keep going through this course and you will learn about **common mitigation strategies** to protect against **DDoS attacks**.

Network Tactics and Defense

Malicious Packet Sniffing

Introduction to Packet Sniffing

Packet sniffing is the practice of using software tools to observe data as it moves across a network.

- Data is transmitted in the form of **packets**, which include:
  - A **header** containing the sender's and receiver's IP addresses.
  - A **body** that may contain sensitive data such as:
    - Full names
    - Date of birth
    - Personal messages
    - Financial details
    - Credit card numbers

As a **security analyst**, packet sniffing tools are used to:

- **Capture** and **analyze** packets for:
  - Debugging network issues
  - Investigating security incidents

However, **threat actors** also use packet sniffing to **intercept and view data** not meant for them—this is considered **unauthorized and malicious**.

How Malicious Actors Use Packet Sniffing

- **Threat actors** can insert themselves into the communication channel between two authorized devices.
- Once in position, they use packet sniffing tools to:
  - **Spy on every packet** passing through their device
  - **Extract sensitive data** from packet bodies
  - **Modify** packet contents (e.g., altering bank account numbers)

They may use either:

- **Software applications** designed for sniffing
- **Hardware devices** configured for interception and monitoring

Types of Packet Sniffing Attacks

1. Passive Packet Sniffing

- The attacker **listens** to network traffic but does **not alter** it.
- All traffic on a shared network hub is **visible** to any connected device.
- Example analogy:
  - Like a **postal worker** reading private mail while still delivering it to the correct recipient.
- The attacker does not interfere but violates privacy.

2. Active Packet Sniffing

- The attacker **manipulates** packets while they're in transit.
- They may:
  - **Inject protocols** to reroute packets
  - **Modify** data within the packet
- Example analogy:
  - Like a **neighbor** offering to deliver your mail, reading or changing its content, and then delivering it.

Both forms are **unauthorized and malicious**, but active sniffing is **more invasive** as it alters data.

Preventing Malicious Packet Sniffing

Network security professionals use the following **defensive measures**:

1. Use of VPN (Virtual Private Network)

- VPNs **encrypt data** as it moves across the network.
- Even if a hacker captures traffic, **they can't read the encrypted data**.
- VPNs provide a **secure tunnel** for all communications.

2. HTTPS Instead of HTTP

- Websites that begin with **HTTPS** use **SSL/TLS encryption**.
- This prevents attackers from reading or altering the data during transmission.
- Always verify that the website uses **HTTPS**, especially when entering personal or financial information.

3. Avoid Unprotected WiFi

- Public WiFi (e.g., in **cafes, airports, restaurants**) is usually **unencrypted**.
- Anyone connected to the same network can **monitor traffic**.
- **Precaution**: Never use public WiFi without a **VPN**.

Summary

- **Packet sniffing** allows for analysis of network data, but can be misused by attackers.
- **Passive sniffing** reads data; **active sniffing** alters it.
- Threat actors use sniffers to **steal and manipulate sensitive information**.
- Defenses include:
  - **VPNs**
  - **HTTPS websites**
  - **Avoiding unprotected public WiFi**

Understanding how **malicious packet sniffing** works helps cybersecurity professionals implement stronger **network security practices** and **protect sensitive data**

IP Spoofing

What is IP Spoofing?

- IP spoofing is a **network attack** where an attacker **changes the source IP address** of a data packet to impersonate an authorized system and gain access to a network.
- The goal is to **bypass firewall rules** and security systems that rely on IP-based authentication, by **pretending to be a trusted device**.

Types of IP Spoofing Attacks

1. On-Path Attack (a.k.a. Man-in-the-Middle Attack)

- The attacker places themselves between two authorized devices (like a web browser and a web server).
- They **intercept or alter** the data being transmitted.
- The attacker uses packet sniffing to learn the **IP and MAC addresses** of both devices.
- After gathering this information, they **impersonate** one or both devices.

2. Replay Attack

- The attacker **intercepts a data packet** and either **delays it** or **replays it later**.
- This can disrupt communications between devices.
- For example, a previously valid login request can be replayed to gain unauthorized access.

3. Smurf Attack

- This combines a **DDoS attack** and **IP spoofing**.
- The attacker spoofs the IP address of a target and then **floods the network with packets**.
- These packets bounce off other devices (amplifiers) and return to the spoofed IP, overwhelming the target.
- It can bring down a **single device**, **server**, or even an **entire network**.

How to Protect Against IP Spoofing

1. Use Encryption

- Encrypt data in transit using **SSL/TLS** to prevent unauthorized reading or tampering.

2. Configure Firewalls Properly

- Firewalls can be set up to **detect and block spoofed packets**.
- If a packet arrives from the internet with a **source IP matching the private network**, the firewall **blocks it**.

- This is because **internal IPs shouldn't appear as senders from the outside**.

3. Implement Packet Filtering Rules

- Use **inbound and outbound filters** to validate IP addresses.
- Create rules to **reject all incoming traffic** that appears to be from the **internal IP address range**.

Summary

- **IP Spoofing** is a deceptive method to **impersonate trusted devices**.
- It enables attacks like:
  - **On-Path Attacks**
  - **Replay Attacks**
  - **Smurf Attacks**
- **Proper firewall rules and encryption** are crucial defenses against IP spoofing.

Overview of Interception Tactics

In the earlier lessons, you learned how **packet sniffing** and **IP spoofing** are used in attacks. These types of attacks are called **interception attacks** because they involve capturing data as it travels across a network.

This reading will explain some **specific attacks** that use these methods. You'll also learn how hackers use these tricks and how security analysts can protect against them.

Packet Sniffing

Packet sniffing means **capturing and inspecting data packets** that are moving through a network.

Normally, on a private network, data is sent only to the correct device. Each device has a **Network Interface Card (NIC)**, which is hardware that connects it to the network. The NIC checks if the packet is meant for it by looking at the **MAC address**. If yes, it accepts it.

However, if a NIC is set to **promiscuous mode**, it will **accept all data packets**, even the ones not meant for it. Hackers use this by running tools like **Wireshark** to capture all this information.

Once they collect personal data, hackers can use it for their own benefit or to **pretend to be a trusted user** by using the stolen IP and MAC addresses. This leads to IP spoofing.

IP Spoofing – A Closer Look

After a hacker has sniffed packets and stolen addresses, they can **pretend to be another user or device** on the network. This is called **IP spoofing**.

Firewalls can help stop this by refusing to allow **unauthorized or suspicious IP packets**.

Let's now look at a few common IP spoofing attacks that you should know about.

1. On-Path Attack (Meddler-in-the-Middle)

In this attack, a hacker **sits between two devices** that trust each other and **intercepts their communication**. They can steal sensitive info like usernames and passwords.

They may also intercept **DNS lookups**, which convert website names into IP addresses. The hacker can then **redirect users** to a fake or malicious site.

How to protect: Use **encryption**, such as **TLS**, to protect your data during transfer.

2. Smurf Attack

This is when a hacker uses a **stolen IP address** and sends **floods of packets** to a **broadcast address** (which sends it to everyone on the network).

Often, they send **ICMP ping messages**. ICMP is used to test network connections. But too many ICMP replies can **overload the servers** and make them shut down – a **Denial of Service (DoS)** attack.

How to protect: Use a **next-generation firewall (NGFW)** that can detect and block strange or high-volume traffic.

3. DoS Attack (Denial of Service)

In a DoS attack, a hacker **uses a stolen identity** to send huge numbers of fake packets. The target server becomes so overwhelmed that it **cannot respond to real users**.

Even though the IP address in the packet looks real, the hacker **does not want a reply** – just to make the system crash by overloading it.

How to protect: Follow the **defense-in-depth** strategy – this means using **multiple layers of security**. One strong method is to use **encryption** along with firewalls and monitoring tools.

Key Takeaways

- **Packet sniffing** lets attackers capture network traffic and steal data.
- **IP spoofing** allows attackers to impersonate trusted devices or users.
- Common attacks include:
  - **On-path attacks** (stealing data between trusted devices)
  - **Smurf attacks** (flooding a network with fake traffic)
  - **DoS attacks** (making a system stop working by overloading it)
- **Defense strategies** include encryption, firewalls, and monitoring tools.

Security analysts must always be ready to use the right protection methods to **limit risks** and **stop attackers** from breaking into the system.

Module wrap-up

What You Learned:

- **Securing Networks:**
  You studied methods to secure a network from unauthorized access and threats.
- **Network Intrusion Tactics:**
  You learned about how attackers use **malicious packet sniffing** to monitor or capture network traffic, and how **IP spoofing** is used to impersonate trusted devices.
- **DoS and DDoS Attacks:**
  These attacks aim to overwhelm systems and make services unavailable. Key examples include:
  - **ICMP Flooding:** Sending a large number of ICMP packets to a server.
  - **SYN Flood Attacks:** Abusing the TCP handshake to exhaust server resources.
  - **Ping of Death:** Sending oversized ICMP packets that crash or freeze systems.

Why This Matters:

All of this knowledge is **essential** for working as a **security analyst**, helping you detect, understand, and defend against various types of network attacks.

Glossary terms from module 3

Active packet sniffing: A type of attack where data packets are manipulated in transit

Botnet: A collection of computers infected by malware that are under the control of a single threat actor, known as the “bot-herder"

Denial of service (DoS) attack: An attack that targets a network or server and floods it with network traffic

Distributed denial of service (DDoS) attack: A type of denial of service attack that uses multiple devices or servers located in different locations to flood the target network with unwanted traffic

Internet Control Message Protocol (ICMP): An internet protocol used by devices to tell each other about data transmission errors across the network

Internet Control Message Protocol (ICMP) flood: A type of DoS attack performed by an attacker repeatedly sending ICMP request packets to a network server

IP spoofing: A network attack performed when an attacker changes the source IP of a data packet to impersonate an authorized system and gain access to a network

On-path attack: An attack where a malicious actor places themselves in the middle of an authorized connection and intercepts or alters the data in transit

Packet sniffing: The practice of capturing and inspecting data packets across a network

Passive packet sniffing: A type of attack where a malicious actor connects to a network hub and looks at all traffic on the network

Ping of death: A type of DoS attack caused when a hacker pings a system by sending it an oversized ICMP packet that is bigger than 64KB

Replay attack: A network attack performed when a malicious actor intercepts a data packet in transit and delays it or repeats it at another time

Smurf attack: A network attack performed when an attacker sniffs an authorized user's IP address and floods it with ICMP packets

Synchronize (SYN) flood attack: A type of DoS attack that simulates a TCP/IP connection and floods a server with SYN packets

Module 4

Security Hardening

Introduction to Security Hardening

Now, Let's Talk About Security Hardening

Security hardening means **making systems more secure** so attackers can't easily get in. This can be done on:

- **Devices** (like computers or phones)
- **Networks** (how devices communicate)
- **Applications** (the software you use)
- **Cloud platforms** (online servers that store data)

As a **security analyst**, you'll often do things like:

- Installing **security updates** (patching)
- Creating **backups** to protect against data loss

Why Hardening Is Important for You

As a future cybersecurity professional, **hardening will be a key part of your daily job**. That's why this topic is very important.

You'll soon learn:

- **Operating System (OS) Hardening**
- **Network Hardening Techniques**
- **Cloud Security Hardening**

We'll go over each of these step by step in this course.

Security Hardening

Security analysts and the organizations they work for must be **proactive** when it comes to protecting systems from cyberattacks. This is where **security hardening** becomes important.

What is Security Hardening?

Security hardening is the process of making systems **stronger and more secure** by reducing their **vulnerabilities** and **attack surface**.

- The **attack surface** includes all the ways an attacker could enter or compromise a system.
- Fewer vulnerabilities = smaller attack surface = safer systems.

Real-Life Analogy: A House

Think of a network like a **house**:

- The **attack surface** is like all the **doors and windows** a robber could use to break in.
- **Security hardening** is like **putting locks** on every door and window to prevent break-ins.

Similarly, in cybersecurity:

- We secure all possible entry points.
- We reduce unnecessary access to keep systems safer.

Where Is Security Hardening Applied?

Security hardening can be performed on:

- **Hardware** (physical devices)
- **Operating Systems (OS)**
- **Applications**
- **Computer Networks**
- **Databases**
- **Physical Security** (e.g., CCTV cameras, guards)

It ensures systems stay secure and function properly over time.

Common Security Hardening Techniques

1. **Patching and Updates**
    - Installing **software updates** (patches) fixes known security bugs.
    - Keeps systems up to date and protected from the latest threats.
2. **Configuration Changes**
    - Enforcing **strong password policies** (e.g., longer passwords, regular changes).
    - Updating **encryption standards** for stored data to prevent unauthorized access.
3. **Removing Unused Components**
    - **Disabling unused apps, services, and ports**.
    - **Removing unnecessary permissions**.
    - Reduces the number of possible entry points attackers can use.
4. **Access Control Adjustments**
    - Giving users **only the access they need**.
    - Helps in **monitoring** devices more efficiently.
    - Keeps sensitive data safe from unauthorized users.

Penetration Testing (Pen Testing)

Another important method of hardening is **penetration testing** (also called a **pen test**):

- A **simulated cyberattack** conducted by experts to find weaknesses in:
    - Systems
    - Networks
    - Applications
    - Websites
    - Processes
- Testers then create a **report** listing the vulnerabilities found.
- Security teams review the report and **fix the problems** based on where the system failed.

Why Security Hardening Matters

Security hardening is **foundational** to cybersecurity because:

- It **reduces the chances of successful cyberattacks**.
- It helps **identify and fix vulnerabilities** early.
- It ensures systems run securely and **meet security compliance requirements**.

As a future security analyst, understanding and applying hardening techniques will be a **core part of your job**.

Security Hardening

Security analysts and the organizations they work for must be **proactive** when it comes to protecting systems from cyberattacks. This is where **security hardening** becomes important.

What is Security Hardening?

Security hardening is the process of making systems **stronger and more secure** by reducing their **vulnerabilities** and **attack surface**.

- The **attack surface** includes all the ways an attacker could enter or compromise a system.
- Fewer vulnerabilities = smaller attack surface = safer systems.

Real-Life Analogy: A House

Think of a network like a **house**:

- The **attack surface** is like all the **doors and windows** a robber could use to break in.
- **Security hardening** is like **putting locks** on every door and window to prevent break-ins.

Similarly, in cybersecurity:

- We secure all possible entry points.
- We reduce unnecessary access to keep systems safer.

Where Is Security Hardening Applied?

Security hardening can be performed on:

- **Hardware** (physical devices)
- **Operating Systems (OS)**
- **Applications**
- **Computer Networks**
- **Databases**
- **Physical Security** (e.g., CCTV cameras, guards)

It ensures systems stay secure and function properly over time.

Common Security Hardening Techniques

1. **Patching and Updates**
    - Installing **software updates** (patches) fixes known security bugs.
    - Keeps systems up to date and protected from the latest threats.
2. **Configuration Changes**
    - Enforcing **strong password policies** (e.g., longer passwords, regular changes).
    - Updating **encryption standards** for stored data to prevent unauthorized access.
3. **Removing Unused Components**
    - **Disabling unused apps, services, and ports**.
    - **Removing unnecessary permissions**.
    - Reduces the number of possible entry points attackers can use.
4. **Access Control Adjustments**
    - Giving users **only the access they need**.
    - Helps in **monitoring** devices more efficiently.
    - Keeps sensitive data safe from unauthorized users.

Penetration Testing (Pen Testing)

Another important method of hardening is **penetration testing** (also called a **pen test**):

- A **simulated cyberattack** conducted by experts to find weaknesses in:
    - Systems
    - Networks
    - Applications
    - Websites
    - Processes
- Testers then create a **report** listing the vulnerabilities found.
- Security teams review the report and **fix the problems** based on where the system failed.

Why Security Hardening Matters

Security hardening is **foundational** to cybersecurity because:

- It **reduces the chances of successful cyberattacks**.
- It helps **identify and fix vulnerabilities** early.
- It ensures systems run securely and **meet security compliance requirements**.

As a future security analyst, understanding and applying hardening techniques will be a **core part of your job**.

OS Hardening

Operating System (OS) Hardening

What is OS Hardening?

Operating System (OS) hardening means applying security measures to make an operating system more secure. It helps protect a single device, and more importantly, the entire network. One weak or unprotected OS can give attackers access to the whole network.

Why Is OS Hardening Important?

The OS acts as a bridge between:

- **Hardware** (the physical parts of the computer),
- **Users** (the people using the device), and
- **Software Applications** (programs running on the computer).

If the OS is not secured, attackers can take control of the system, access files, install malware, and even move through the network to attack other devices.

Types of OS Hardening Tasks

There are two main types of OS hardening tasks:

1. Regular Tasks (Repeated Over Time)

These tasks are done frequently to keep the OS safe and up-to-date.

- **Patch Updates (Patch Installation):**
  - These are software updates released by the OS vendor to fix bugs or security holes.
  - It's important to apply these updates quickly. Once a patch is released, hackers know what the vulnerability is and can attack systems that haven't updated yet.
  - Example: Your team may need to urgently update servers if a vulnerability is found in a commonly used programming library.
  - The latest patched OS version should be saved as a **Baseline Configuration**.
- **Baseline Configuration:**
  - A documented set of settings that shows how a secure system should be set up.
  - It includes things like which network ports should be open or blocked.
  - If something seems wrong later, you can compare the current settings to the baseline to detect changes or attacks.

- **Hardware and Software Disposal:**
    - Old or unused hardware should be **securely wiped** and **properly disposed of**.
    - Unused software should be deleted. Some old programs or languages may have known vulnerabilities.

2. One-Time Tasks (Set Up Once)

These tasks are usually done during the initial setup of the OS.

- **Encryption Configuration:**
    - The device should be set to use **secure encryption standards** to protect data.

Strong Password Policies

To stop unauthorized access, organizations should create strong password rules, for example:

- At least **8 characters**
- At least one **capital letter**
- At least one **number**
- At least one **symbol** (e.g., !, @, #)
- Lock the account after a set number of wrong password attempts

Multi-Factor Authentication (MFA):

MFA is an extra layer of security. It asks users to verify their identity in more than one way, such as:

- **Something you know** – a password or PIN
- **Something you have** – an ID card or phone
- **Something you are** – a fingerprint or facial scan

Summary

- **OS Hardening** means improving the security of an operating system.
- It involves **patch updates**, **strong passwords**, **removal of unused software**, **MFA**, and keeping a **baseline configuration**.
- These practices reduce the chance of attacks and help protect the whole network.

Brute Force Attacks and OS Hardening

Brute Force Attacks

A brute force attack is a trial-and-error method used to discover private information such as login credentials. There are different types:

- **Simple brute force attacks**: Attackers manually or programmatically try different combinations of usernames and passwords until successful.
- **Dictionary attacks**: Attackers use lists of common passwords or previously leaked credentials. Originally, these lists were based on dictionary words—hence the name.

Tools

Attackers often use automated tools to speed up brute force attacks.

Assessing Vulnerabilities

Before an attack occurs, analysts can assess vulnerabilities using **virtual machines** and **sandbox environments**.

Virtual Machines (VMs)

- VMs are software versions of physical computers.
- They provide isolated environments to safely run and test suspicious code.
- Benefits:
  - Malware can’t easily affect the host machine.
  - Easy to revert to a clean state.
- Caution: Advanced malware might escape the VM and affect the host system.

Sandbox Environments

- A sandbox is a safe testing space separate from the main network.
- Used to:
  - Test patches and updates
  - Detect bugs or vulnerabilities
  - Simulate attack scenarios
- Types:
  - Standalone physical machines
  - Virtual or cloud-based sandboxes
- Note: Some malware can detect when it's running in a sandbox and act harmless to avoid detection.

Prevention Measures

Organizations can use several techniques to prevent brute force attacks:

- **Salting and Hashing**
    - Hashing turns data into a one-way string (cannot be reversed).
    - Salting adds random characters to passwords before hashing to increase security.
- **Multi-Factor Authentication (MFA) & Two-Factor Authentication (2FA)**
    - MFA requires two or more verification methods (e.g., password + fingerprint or OTP).
    - 2FA is a type of MFA using just two factors.
- **CAPTCHA and reCAPTCHA**
    - CAPTCHA: A test to differentiate between humans and bots.
    - reCAPTCHA: A Google service that blocks bots from brute forcing websites.
- **Password Policies**
    - Define rules for password complexity, expiration, reuse, and lockouts.
    - Common elements include:
        - Minimum character length
        - Use of uppercase, lowercase, numbers, and symbols
        - Limiting login attempts
        - Regular password updates

Key Takeaways

- Brute force attacks guess passwords through trial and error.
- Simple and dictionary attacks are the most common forms.
- Analysts can use VMs and sandboxes to test and simulate attacks safely.
- Preventative measures include:
    - Hashing and salting
    - MFA/2FA
    - CAPTCHA and reCAPTCHA
    - Strong password policies

Network Hardening

Network Hardening Practices

Earlier, you learned that **OS hardening** focuses on device security by using things like patch updates, secure settings, and controlling account access.

Now, let's focus on **Network Hardening**. This is all about protecting the **network** using different techniques such as **port filtering**, **access control**, and **encryption**. Some tasks are done **regularly**, while others are set up **once** and only updated when needed.

Regular Network Hardening Tasks

These tasks are performed regularly to keep the network secure:

- **Firewall Rule Maintenance:** Updating rules that control traffic flow into and out of the network.
- **Network Log Analysis:** Checking logs (records of network activity) to find anything suspicious.
- **Patch Updates:** Fixing security bugs by updating network devices and software.
- **Server Backups:** Creating regular backups to protect data in case of a security breach or failure.

Network Log Analysis Tools

- **Log Analyzer or SIEM (Security Information and Event Management) Tools** are used.
- A **SIEM tool**:
  - Collects security data from across the network.
  - Shows everything on a single dashboard (called a **single pane of glass**).
  - Helps analysts detect and respond to threats.
  - Prioritizes vulnerabilities (high, medium, low) so that **high-risk** issues are fixed quickly.

One-Time Network Hardening Tasks

These are usually done during the initial setup, then only updated when necessary:

Port Filtering

- A **firewall** can allow or block specific **port numbers**.
- Only **needed ports** should be open; unused ones should be **blocked**.

- This reduces the risk of **port-based attacks**.

Network Access Privileges

- Only authorized users should be able to access certain parts of the network.
- Helps limit the spread of threats or sensitive data leaks.

Encryption for Network Communication

- All communication should use **strong encryption** to protect data.
- Especially important in **restricted zones** (with sensitive data), where **stronger encryption** is required.

Other Important Practices

Use of Modern Wireless Protocols

- Networks should use the **latest and most secure wireless protocols**.
- Older, less secure protocols should be **disabled**.

Network Segmentation

- Dividing the network into **subnets** (smaller networks) based on departments or roles.
- Example: One subnet for marketing, another for finance.
- Limits damage if one subnet is attacked.
- Also separates **security zones**, like keeping confidential areas apart from general ones.

Summary

You now understand the **most common network hardening practices**. These include both regular maintenance tasks and one-time setup tasks. These methods are crucial in keeping a network secure and are an important part of a **security analyst's daily job**.

This knowledge will help you not just in this course, but also in your future cybersecurity career.

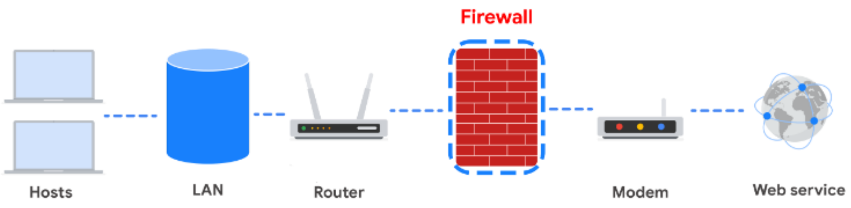
Network Security Applications

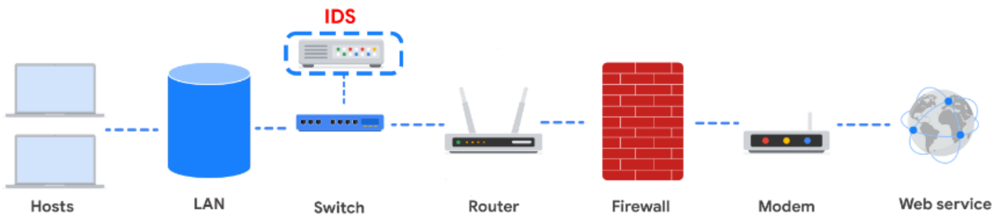
This section focuses on **network hardening and monitoring**. Every device, tool, or strategy used by security analysts adds another layer of protection. This layered approach to securing a network is called **defense in depth**.

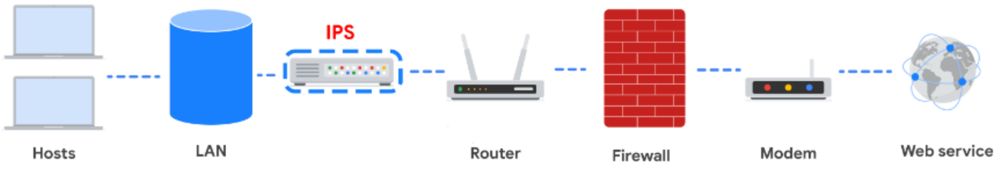
In this reading, you'll learn about **four key tools** used to protect a network:

- Firewalls
- Intrusion Detection Systems (IDS)
- Intrusion Prevention Systems (IPS)
- Security Information and Event Management (SIEM) tools

Organizations choose one or more of these tools based on the level of security they want to achieve. Each tool adds a layer of defense to help harden the network.







Firewall

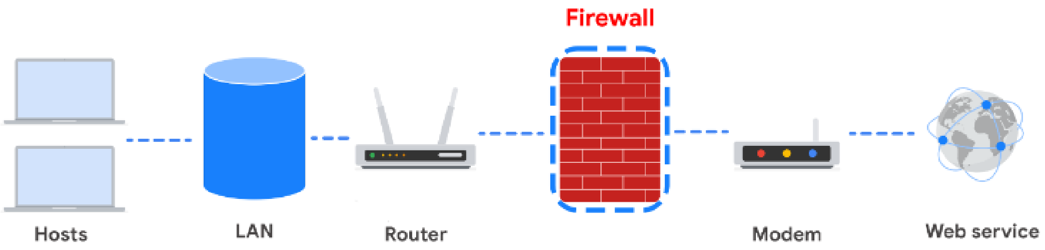
You've already learned about different types of firewalls:

- **Stateless firewalls**
- **Stateful firewalls**
- **Next-Generation Firewalls (NGFWs)**

Purpose of a Firewall:

- Firewalls **allow or block network traffic** based on a set of rules.
- They inspect **packet headers** to decide whether traffic should be allowed.
- **NGFWs** go further by also inspecting the **packet payload** (the content).

Every system, not just the network, should have its own firewall for added protection.



Intrusion Detection System (IDS)

An **IDS** is a tool that monitors system activity and **alerts administrators** when suspicious activity is detected.

Key Features:

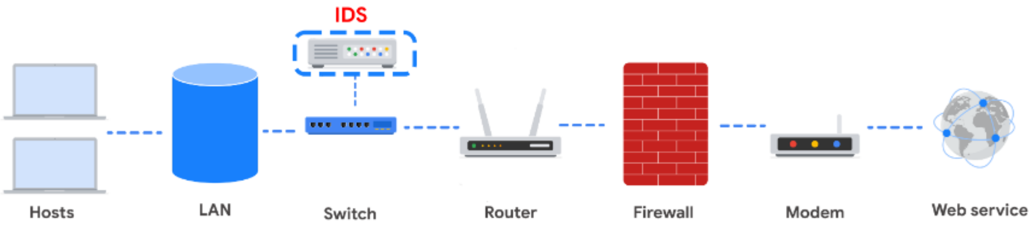
- Detects **known attack signatures**.
- Monitors **network traffic** and **packet data**.
- Alerts are triggered based on **known patterns** or **anomalies**.
- Helps analysts investigate possible threats.

Limitations:

- Only detects **known attacks** or **obvious anomalies**.
- It does **not block** malicious traffic—it just reports it.
- New or advanced attacks may **go undetected**.

Placement in the Network:

- An IDS is placed **behind the firewall** and **before the internal network**.
- This reduces **false positives** by filtering out known bad traffic before the IDS processes it.



Intrusion Prevention System (IPS)

An **IPS** is similar to an IDS, but it **actively blocks threats** rather than just alerting about them.

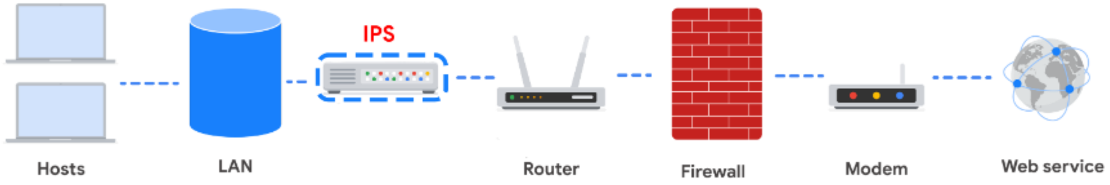
Key Features:

- Scans traffic for **known attack signatures** and **anomalies**.
- **Automatically blocks** or **drops** suspicious packets or sources.
- Reports events to analysts for review.

Limitations:

- An IPS is **inline**, meaning it sits **between** the firewall and the internal network. If it fails, the connection to the network may also fail.
- May produce **false positives**, potentially blocking **legitimate traffic**.

and blocks a specific sender or drops network packets that seem suspect.



Full Packet Capture Devices

These devices record **all network traffic**. They are useful for:

- Investigating **alerts generated by an IDS**.
- Reviewing detailed packet data to understand threats.

Security Information and Event Management (SIEM)

A **SIEM tool** collects and analyzes **log data** from multiple sources to monitor critical network activity.

Key Features:

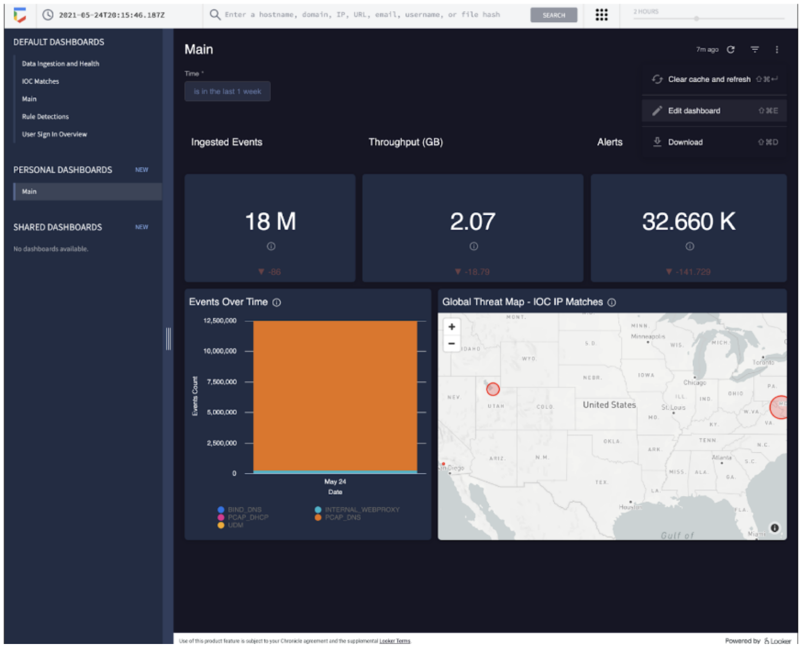
- **Aggregates logs** from firewalls, IDS, IPS, VPNs, DNS servers, etc.
- Displays data on a **central dashboard** (called a **single pane of glass**).
- Operates in **real time** to highlight suspicious activity.
- Helps analysts monitor and react to incidents more effectively.

Examples of SIEM Tools:

- **Google Chronicle** – a cloud-native SIEM that stores, analyzes, and searches security data.
- **Splunk Enterprise / Splunk Cloud** – widely used tools with customizable dashboards.

While a SIEM helps monitor network security, it **does not take action** on threats. That responsibility still lies with the **security analyst**.

Security analysts typically work in a **Security Operations Center (SOC)**, where they monitor and investigate network activity using tools like SIEMs.



Use of this product feature is subject to your Chronicle agreement and the supplemental Looker Terms.

Key Takeaways

Device / Tool	Advantages	Disadvantages
Firewall	Blocks or allows traffic based on rules.	Only inspects packet headers; limited visibility.
Intrusion Detection System (IDS)	Alerts administrators to suspicious traffic.	Doesn't block traffic; limited to known threats.
Intrusion Prevention System (IPS)	Blocks threats in real time.	If it fails, it may disconnect the network; can block good traffic.
SIEM	Collects logs and presents data in one place.	Doesn't block threats—only reports them.

Final Thoughts

All of these tools contribute to **network hardening**. Choosing the right tools depends on the **organization's risk level and budget**. Some tools require **extra staff** to monitor and respond to alerts, like with a SIEM.

Later in the course, you'll learn more about how organizations decide which security tools and strategies to use.

Cloud Hardening

Network Security in the Cloud

In recent years, many organizations are using **network services in the cloud**. So, in addition to securing on-premises networks, a **security analyst** will also need to secure **cloud networks**.

In a previous video, you learned that a **cloud network** is a collection of servers or computers that stores **resources and data in a remote data center**, which can be accessed via the internet. These cloud servers can host company **data and applications** using **cloud computing** to provide **on-demand storage, processing power, and data analytics**.

Just like regular web servers, **cloud servers also require proper maintenance**, which is done through **various security hardening procedures**.

Although cloud servers are **hosted by a cloud service provider**, these providers **cannot prevent all intrusions** in the cloud—especially from **malicious actors**, whether they are **internal or external** to an organization.

Key Difference Between Cloud and Traditional Network Hardening

One main difference in **cloud network hardening** compared to **traditional network hardening** is the **use of a server baseline image** for all server instances stored in the cloud.

- A **baseline image** allows you to **compare the current state of the cloud server** to a clean, known-good version to ensure **no unauthorized or unverified changes** have been made.
- An **unverified change** might be a sign of an **intrusion** or unauthorized activity in the cloud.

Separation of Applications and Data

Similar to **Operating System (OS) hardening**, in cloud environments:

- **Data and applications are kept separate** depending on their **service category**.
- For example:
    - **Older applications** should be kept separate from **newer applications**.
    - **Internal function software** should be separate from **front-end applications** that users interact with.

Shared Responsibility Model

Even though the **cloud service provider** is responsible for managing the infrastructure, the **organization using the cloud** also has its own security responsibilities.

- This is known as the **shared responsibility model**.
- The **organization must implement additional security measures** to protect their cloud resources.

Final Thoughts

Just like traditional networks, **cloud operations must also be secured**. As a **security analyst**, understanding how to secure cloud networks is an essential skill.

You're doing great. Meet you in the next video.

Secure the Cloud

Earlier in this course, you were introduced to **cloud computing**. Cloud computing is a model for allowing convenient and on-demand network access to a shared pool of configurable computing resources. These resources can be configured and released with minimal management effort or interaction with the service provider.

Just like any other IT infrastructure, a cloud infrastructure needs to be secured. This reading will address some main security considerations that are unique to the cloud and introduce you to the **shared responsibility model** used for security in the cloud. Many organizations that use cloud resources and infrastructure express concerns about the privacy of their data and resources. This concern is addressed through cryptography and other additional security measures, which will be discussed later in this course.

Cloud Security Considerations

Many organizations choose to use cloud services because of the ease of deployment, speed of deployment, cost savings, and scalability of these options. Cloud computing presents unique security challenges that cybersecurity analysts need to be aware of.

Identity Access Management

Identity access management (IAM) is a collection of processes and technologies that helps organizations manage digital identities in their environment. This service also authorizes how users can use different cloud resources. A common problem that organizations face when using the cloud is the **loose configuration of cloud user roles**. An improperly configured user role increases risk by allowing unauthorized users to have access to critical cloud operations.

Configuration

The expanding cloud ecosystem introduces significant complexity to network management. Each cloud service necessitates **precise configuration** to uphold security and compliance standards. This challenge intensifies during **cloud migrations**, where ensuring accurate configuration for every migrated process is critical. Neglect in this area can expose the network to vulnerabilities. **Misconfigured cloud services** are a frequent source of security breaches, underscoring the importance of meticulous attention to detail by network administrators and architects during the migration and ongoing management of cloud services.

Attack Surface

Cloud service providers (CSPs) offer numerous applications and services for organizations at a low cost.

Every service or application on a network carries its own set of risks and vulnerabilities and **increases an organization's overall attack surface**. An increased attack surface must be compensated for with increased security measures.

Cloud networks that utilize many services introduce lots of **entry points** into an organization's network. However, if the network is designed correctly, utilizing several services does not introduce more entry points into an organization's network design. These entry points can be used to introduce **malware** onto the network and pose other security vulnerabilities. It is important to note that CSPs often defer to more secure options, and have undergone more scrutiny than a traditional on-premises network.

Zero-Day Attacks

Zero-day attacks are an important security consideration for organizations using cloud or traditional on-premise network solutions. A zero-day attack is an exploit that was previously unknown. CSPs are more likely to know about a zero-day attack occurring before a traditional IT organization does. CSPs have ways of **patching hypervisors** and **migrating workloads** to other virtual machines. These methods ensure the customers are not impacted by the attack. There are also several tools available for **patching at the operating system level** that organizations can use.

Visibility and Tracking

Network administrators have access to every **data packet** crossing the network with both on-premise and cloud networks. They can **sniff and inspect data packets** to learn about network performance or to check for possible threats and attacks.

This kind of visibility is also offered in the cloud through **flow logs** and tools, such as **packet mirroring**. CSPs take responsibility for security in the cloud, but they do not allow the organizations that use their infrastructure to monitor traffic on the CSP's servers. Many CSPs offer strong security measures to protect their infrastructure. Still, this situation might be a concern for organizations that are accustomed to having full access to their network and operations. CSPs **pay for third-party audits** to verify how secure a cloud network is and identify potential vulnerabilities. The audits can help organizations identify whether any vulnerabilities originate from on-premise infrastructure and if there are any **compliance lapses** from their CSP.

Things Change Fast in the Cloud

CSPs are large organizations that work hard to stay up-to-date with technology advancements. For organizations that are used to being in control of any adjustments made to their network, this can be a potential challenge to keep up with. **Cloud service updates** can affect security considerations for the organizations using them. For example, **connection configurations** might need to be changed based on the CSP's updates.

Organizations that use CSPs usually have to update their **IT processes**. It is possible for organizations to continue following established best practices for changes, configurations, and other security considerations. However, an organization might have to adopt a **different approach** in a way that aligns with changes made by the CSP.

Cloud networking offers various options that might appear attractive to a small company—options that they could never afford to build on their own premises. However, it is important to consider that each service adds **complexity to the security profile** of the organization, and they will need **security personnel** to monitor all of the cloud services.

Shared Responsibility Model

A commonly accepted cloud security principle is the **shared responsibility model**. The shared responsibility model states that the **CSP must take responsibility** for security involving the cloud infrastructure, including **physical data centers**, **hypervisors**, and **host operating systems**. The **company using the cloud service** is responsible for the **assets and processes** that they store or operate in the cloud.

The shared responsibility model ensures that both the **CSP and the users agree** about where their responsibility for security begins and ends. A problem occurs when organizations assume that the CSP is taking care of security that they have not taken responsibility for. One example of this is **cloud applications and configurations**. The CSP takes responsibility for securing the cloud, but it is the organization's responsibility to ensure that services are **configured properly** according to the security requirements of their organization.

Key Takeaways

- It is essential to understand the **security considerations unique to the cloud**.
- Understand the **shared responsibility model** for cloud security.
- Organizations are responsible for **correctly configuring and maintaining** best security practices for their cloud services.
- The shared responsibility model clarifies what the **CSP** is responsible for and what the **organization** must secure on their end.

Cryptography and Cloud Security

Earlier in this course, you were introduced to the concepts of the **shared responsibility model** and **identity and access management (IAM)**. Similar to on-premise networks, cloud networks also need to be secured through a mixture of **security hardening practices** and **cryptography**.

This reading will address common **cloud security hardening practices**, what to consider when implementing cloud security measures, and the fundamentals of **cryptography**. Since cloud infrastructure is becoming increasingly common, it's important to understand how cloud networks operate and how to secure them.

Cloud Security Hardening

There are various techniques and tools that can be used to secure cloud network infrastructure and resources. Some common cloud security hardening techniques include incorporating:

- **Identity Access Management (IAM)**
- **Hypervisors**
- **Baselining**
- **Cryptography**
- **Cryptographic Erasure**

Identity Access Management (IAM)

Identity Access Management (IAM) is a collection of processes and technologies that helps organizations manage digital identities in their environment. This service also authorizes how users can leverage different cloud resources.

Hypervisors

A **hypervisor** abstracts the host's hardware from the operating software environment. There are two types of hypervisors:

- **Type One Hypervisors**: Run directly on the hardware of the host computer.
  Example: *VMware ESXi*
- **Type Two Hypervisors**: Operate on top of the host's operating system.
  Example: *VirtualBox*

Cloud Service Providers (CSPs) commonly use **Type One Hypervisors** and are responsible for managing them and applying regular patches and updates.

Vulnerability Note: Misconfigurations or flaws in hypervisors can lead to **Virtual Machine (VM) Escapes**, where an attacker may gain access to the host system or other VMs.

As a cloud customer, you will rarely deal directly with hypervisors, as CSPs manage them.

Baselining

Baselining involves establishing a fixed reference point for a cloud environment's configuration and setup. It helps in identifying unwanted or unexpected changes.

Examples of baselining tasks:

- Restricting access to the admin portal
- Enabling password management
- Enabling file encryption
- Enabling threat detection for SQL databases

Cryptography in the Cloud

Cryptography secures data stored and processed in the cloud using **encryption** and **secure key management**. It provides **data confidentiality and integrity**.

- **Encryption**: Converts data into unreadable **ciphertext**. Only someone with the **encryption key** can read it.
- Earlier encryption was done manually; modern encryption focuses on **keeping the key secret**, not the algorithm.

Encryption is critical for:

- Securing sensitive data
- Protecting cloud-stored data from unauthorized access
- Ensuring data confidentiality

Cryptographic Erasure

Cryptographic Erasure (also called **crypto-shredding**) involves **destroying the encryption keys** used to secure data. Without the key, encrypted data becomes **unreadable** and **irretrievable**.

This method is used in cloud environments where traditional data destruction methods are not effective.

Important: All copies of the key must be destroyed to ensure no future access to the data.

Key Management

Modern encryption security depends on how well **encryption keys** are protected.

Key protection tools include:

- **Trusted Platform Module (TPM)**: A hardware chip that securely stores passwords, certificates, and keys.
- **Cloud Hardware Security Module (CloudHSM)**: A cloud-based device for safely storing cryptographic keys and handling encryption/decryption tasks.

Customer and Cloud Service Provider (CSP) Responsibilities

- Customers generally **don’t have access** to the encryption keys used by CSPs.
- Many CSPs **allow customers to bring their own keys**.
- If a customer’s key is compromised, the CSP **cannot help recover the data**.
- Customers should review the CSP’s **audit reports** and **security controls**.
- **FEDRAMP** offers a list of verified CSPs for federal contractors.

Key Takeaways

- **Cloud security hardening** is essential for protecting data and systems in cloud environments.
- Techniques like **IAM**, **baselining**, **hypervisor management**, **cryptography**, and **cryptographic erasure** help secure cloud infrastructure.
- Encryption protects sensitive data and must be paired with strong **key management practices**.
- Customers and CSPs share responsibility, but customers must **protect their own keys** and **monitor the CSP’s security practices**.

Module wrap-up

You've just completed learning about **security hardening** and its critical role in strengthening an organization's infrastructure. Here's a summary of what you covered:

1. What is Security Hardening?

Security hardening involves making systems and networks more secure by reducing vulnerabilities and minimizing the attack surface. Its goal is to lower the chances of a successful cyberattack.

2. Operating System (OS) Hardening

You explored techniques to protect operating systems, including:

- **Applying patch updates** to fix known vulnerabilities
- **Setting baseline configurations** to define a secure system state
- **Proper disposal of outdated hardware and software** to prevent data leakage

3. Network Hardening

You learned about securing physical and virtual networks using:

- **Network log analysis** to monitor unusual activities
- **Firewall rule maintenance** to control what traffic can enter or leave the network

4. Cloud Network Hardening

You discovered how to secure cloud environments by understanding:

- The **shared responsibility model** between the organization and the cloud provider
- How to **compare server states to baseline images** to detect unauthorized changes
- The need for **separating applications and data** based on function and sensitivity

Glossary terms from module 4

Baseline configuration (baseline image): A documented set of specifications within a system that is used as a basis for future builds, releases, and updates

Hardware: The physical components of a computer

Multi-factor authentication (MFA): A security measure which requires a user to verify their identity in two or more ways to access a system or network

Network log analysis: The process of examining network logs to identify events of interest

Operating system (OS): The interface between computer hardware and the user

Patch update: A software and operating system update that addresses security vulnerabilities within a program or product

Penetration testing (pen test): A simulated attack that helps identify vulnerabilities in systems, networks, websites, applications, and processes

Security hardening: The process of strengthening a system to reduce its vulnerabilities and attack surface

Security information and event management (SIEM): An application that collects and analyzes log data to monitor critical activities for an organization

World-writable file: A file that can be altered by anyone in the world

Course Review – What You Learned

1. **Network Basics:**
   - You learned **how networks are built** (their structure and design).
   - This helps you find **weak spots (vulnerabilities)** in a network.
2. **Network Operations:**
   - You studied **how data moves** over a network using **protocols**.
   - You also learned about **common attacks** like:
     - **DoS (Denial of Service)**
     - **Packet sniffing** (watching data traffic)
     - **IP spoofing** (pretending to be someone else on the network)
   - Security tools like **firewalls** help protect the network from these attacks.
3. **Security Hardening:**
   - This means **reducing the chances of attacks** by making the network stronger.
   - Hardening can happen at:
     - **Hardware level** (physical security)
     - **Software level** (secure OS and apps)
     - **Network level** (firewalls, segmentation)
   - The goal is to **reduce the "attack surface"**, so hackers have fewer ways to get in.