

ACS Education 1st

Fundamental Theory



Index

- An overview of the information security
- Linux fundamentals - introduction
- Linux file management
- Linux user management
- Linux process management
- Linux network management
- Network fundamentals - introduction
- TCP/IP essentials
- Network protocol
- Network feature

01

An overview of the information security

- Definition of information security
- Basic terms of information security

Definition of information security

What is information security?

Hackers with malicious intent may use the act of hacking to gain fame or settle a score, but their ultimate goal is money or operating profit.

- Hacking for malicious intent
 - The act of enabling the information architecture of various systems to behave in ways not intended by their original designers and operators for malicious purposes.
 - Hackers with malicious intent
 - Get the most value for the least effort
 - Use the job as a way to showcase their skills
 - Disrupt a specific group
 - Hackers' endgame
 - Fame
 - **Money or operating profit**
 - Resolving grudges



Definition of information security

What is information security?

Just like investing in a business, malicious hackers want an efficient return on their investment. As a security professional, you must be constantly vigilant and prepared to make the right decisions and responses.

- The ROI that even hackers talk about
 - What is ROI?
 - Short for Return On Investment
 - The profit earned from the investment of any resource
 - A high ROI means that an investment pays for itself.
 - What is ROI for hackers?
 - Determine the value of an asset for a specific purpose and hack into it intelligently
 - Using methods and roles to achieve maximum efficiency at minimum cost (money, time, people)
 - How security professionals should view hackers
 - Determine exactly what assets need to be protected and invest in the right response
 - Stay alert to potential threats so you can stay on top of and respond to them



Definition of information security

What is information security?

Everyone can see potential threats, but everyone reacts differently. Not enough or too much is always a barrier to getting anything done, and security is no different. It's important to strike a balance between not going overboard and not being careless.

- Interpretation of security
 - Security insensitivity and hypersensitivity
 - Security insensitivity
 - Symptoms of not feeling a duty of care for security
 - Security hypersensitivity
 - Symptoms of overreacting and feeling a duty of care for security
 - For those with security insensitivity
 - Carelessness due to lack of security can lead to becoming a victim of a hacking incident, which can cause some form of damage, not only financially, but also emotionally to those around you.
 - Need to be security conscious and mindful of behavior

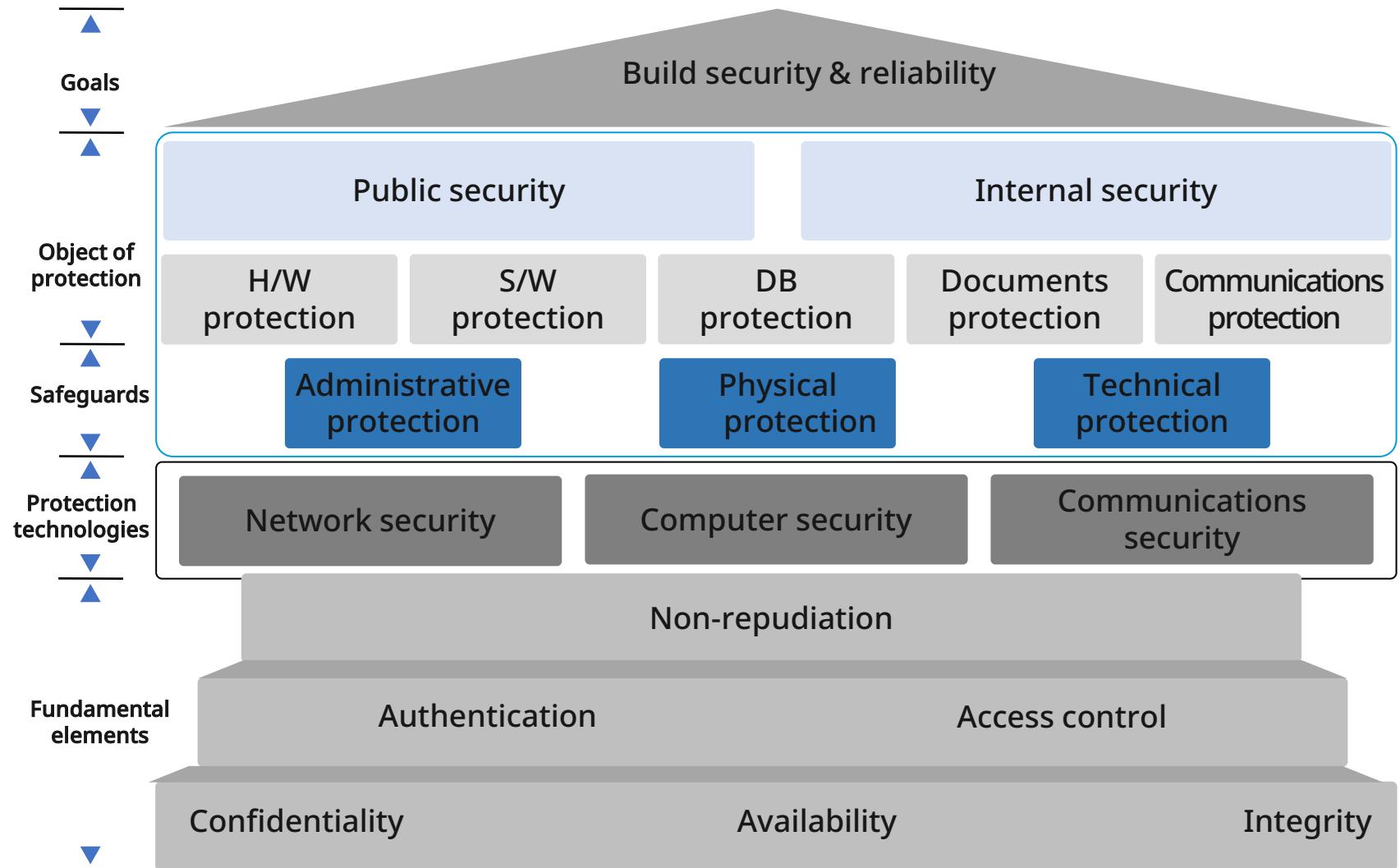
Definition of information security

What is information security?

Everyone can see potential threats, but everyone reacts differently. Not enough or too much is always a barrier to getting anything done, and security is no different. You don't want to overdo it, but you also don't want to be careless, and it's important to strike a balance.

- Interpretation of safety
 - For those with security hypersensitivity
 - Overinvesting in security can be costly, and it can be difficult to react to things one cannot see.
 - Knowing the exact security elements you need to invest in and replacing them accordingly is efficient.

Basic terms of information security



Basic terms of information security

Confidentiality

Prevent unauthorized parties from accessing information.
E.g., using a username and password.

Authentication

Process of determining whether a person or object is who they claim to be.

Integrity

Protect information from being altered, deleted, or created by unauthorized persons.

Non-repudiation

Provide the recipient of the data with proof of its origin.

Availability

Ability to recover quickly and completely from accidents and disasters

Access control

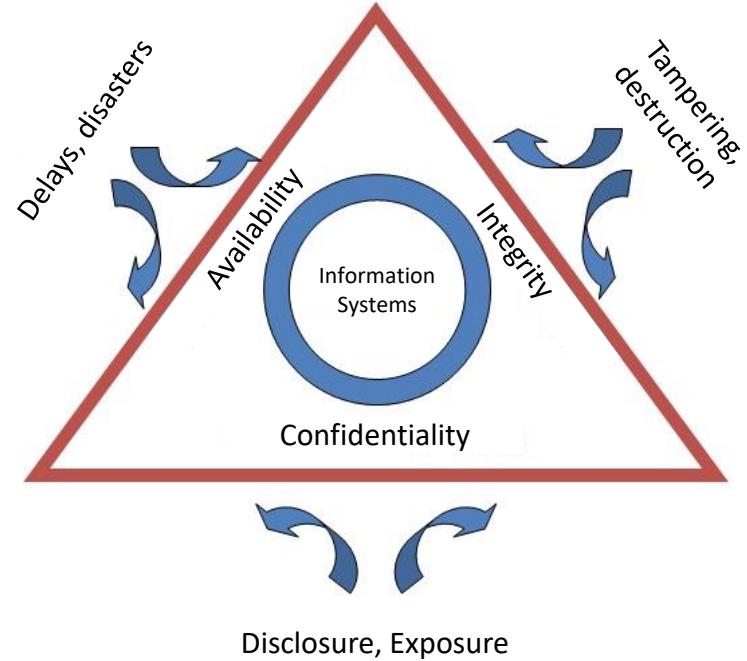
Determine whether to allow or deny use of a system or information.

Basic terms of information security

The basics of information security

The three pillars of security are confidentiality, integrity, and availability.

- The three pillars of security
 - Protection to secure assets
 - Confidentiality
 - Restrict information to authorized users
 - Valuable information, such as personal information or intellectual property
 - Company proprietary core technology / personal information about complainants
 - Integrity
 - Maintain accurate and complete information without inappropriate alteration or destruction
 - Availability
 - Access to and use of timely, reliable information



Basic terms of information security

The basics of information security

The priorities of the three pillars of security may change for different companies, organizations, and industries, and the compromise of each pillar requires a customized response.

- When the three pillars of security are compromised
 - Compromised confidentiality
 - Confidential information is exposed while a user is using and storing it unencrypted and in plain text.
 - Compromised integrity
 - A file, instead of being legitimately updated, is modified to contain malware that will act maliciously in the future.
 - Compromised availability
 - A system that was serving a user is unavailable due to a specific attack.

Basic terms of information security

Information security goals

All security starts with understanding what you want to protect and what it takes to get there.

- Maintain the confidentiality of information
 - Access only by authenticated users and keep it confidential from third parties
 - Information protection mechanisms
 - Means of protecting confidentiality include encryption, logical and physical access controls, transport protocols, and controlled traffic flows.
- Maintain information integrity
 - Must prevent information from being altered, deleted, etc.
 - Integrity control mechanisms
 - Hash function H(M), Message Authentication Code (MAC)

Basic terms of information security

Information security goals

All security starts with understanding what you want to protect and what it takes to get there.

- Make information available
 - Provide authorized users with timely access to the information, systems, and resources they need
 - Availability measures include backup, redundancy, fault tolerance, and material recovery
- Accountability (accuracy, down to the individual level)
 - In the event of a security incident, you should be able to guess who did it and how.
 - Ability to prove a subject's identity and trace their activities
 - Identification, authentication, authorization, access control, and auditing are important foundational concepts

Basic terms of information security

Information security goals

All security starts with understanding what you want to protect and what it takes to get there.

- Access control
 - The process by which a system determines whether a resource is available
 - Security features that control how users and systems communicate and interact with other systems and resources
 - Protect resources from the threat of unauthorized actions associated with communication systems
 - Subjects, objects, and logical access controls are identified → authenticated → authorized

Basic terms of information security

Information security goals

All security starts with understanding what you want to protect and what it takes to get there.

- Authentication
 - Ways to prove who you are and what you are doing
 - Password methods, use of public keys, memory and smart cards, biometric tools, etc.
 - User authentication, data authentication

- Provide non-repudiation capability
 - Ways to prevent non-repudiation of messages
 - Electronic signatures, certificate issuance by public key certificate authorities

Basic terms of information security

Information security goals

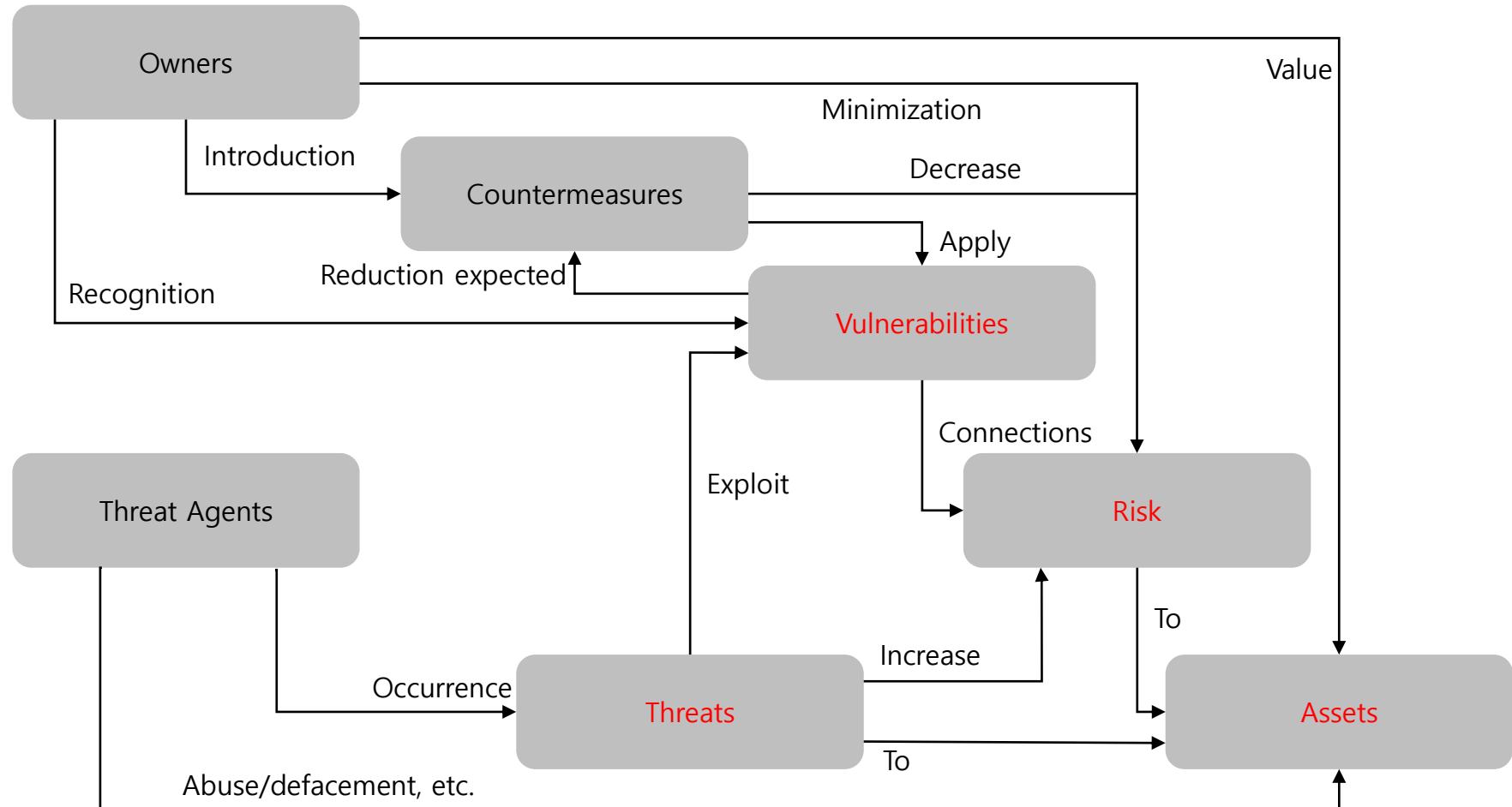
All security starts with understanding what you want to protect and what it takes to get there.

- Authorization
 - Procedure for verifying that you are an authorized user
 - Authorization process by which the server verifies users
 - Users who successfully log on to the server are issued a token as a sign that they are authorized.
 - When authorized users submit a request, they include the authorization token information in the request header.
 - The server decrypts the token information in the request to verify that it is a server-issued token.
 - The user's permissions are checked in the DB using the user ID obtained from the token.
 - If the user is authorized to use the request, process the request.
 - If not, return a 401 Unauthorized Response or other error codes.

Basic terms of information security

Threats and risks to information security

- Threats and risks to information security



Basic terms of information security

Threats and risks to information security

A threat cannot be a risk if a vulnerability does not exist.

- Assets

- Tangible equipment that includes information systems and information, such as servers, communication devices, applications, and databases, among other things of value used in the work of an organization.
- Computer system assets are categorized as follows :
 - Hardware
 - Computer systems and data processing, storage, and communications equipment
 - Software
 - Operating Systems (OS), system tools, and applications
 - Data
 - Security-related data, such as files, databases, and password files
 - Telecommunications equipment and networks
 - Local and wide area network communication connections, bridges, routers, etc.

Basic terms of information security

Threats and risks to information security

A threat cannot be a risk if a vulnerability does not exist.

- Threat
 - An environment that provides a potential source of loss or damage
 - E.g., vulnerabilities, hacker presence, and critical assets that could be targeted
- Threat agent
 - Entities (hackers, normal users, computer processes, disasters) that can take actions that harm information assets.
 - Diverse in terms of expertise, resources, opportunities, and motivation
- Vulnerability
 - Potential characteristics of an asset that can be exploited by a threat

Basic terms of information security

Threats and risks to information security

A threat cannot be a risk if a vulnerability does not exist.

- Risk
 - Risk = vulnerability * asset * threat
 - Potential for loss due to threats
 - E.g., a qualitative or quantitative measure as a result of information leaks from vulnerabilities, actual attacks by hackers, or attempts to take over critical assets
- Exposure
 - A vulnerability exposes a potential loss to the threat agents.
 - E.g., poor password management exposes users to the possibility of their passwords being used in an unauthorized manner.
- Residual risk
 - Risk that remains after information security measures have been implemented
 - Risk that remains after security measures or countermeasures have been implemented to reduce risk as much as the organization can accept

Basic terms of information security

STRIDE threat model

Developed by Praerit Garg, STRIDE provides six threat modeling classifications categorized by threat type.

| Threat | Property | Description |
|------------------------|-------------------|--|
| Spoofing | Authenticity | Gain system access by falsifying identifying information |
| Tampering | Integrity | Illegal modification of data |
| Repudiation | Non-repudiability | User denies a specific action they performed. |
| Information disclosure | Confidentiality | Leak information that shouldn't be leaked |
| Denial of service | Availability | Intentionally deplete the resources of a system or application to limit its use by other users |
| Elevation of privilege | Authorization | A user with limited privileges can acquire the privileges of another user to perform a desired function. |

Basic terms of information security

The importance of cyberwarfare revealed in the Russo-Ukrainian War

In the Russia-Ukraine war, Ukraine's cyber operators have been deployed to the front lines of the conflict, utilizing a new kind of advanced technology to fight.

Ukraine war: Cyber-teams fight a high-tech war on front lines

© 6 September 2023



Ukrainians are increasingly using drones at the front line - sometimes for surveillance and sometimes to act as weapons

Source: BBC

Russia behind cyber attack with Europe-wide impact an hour before Ukraine invasion

New UK and US intelligence suggests Russia was behind an operation targeting commercial communications company Viasat in Ukraine.



Russia has been behind a series of cyber attacks since the start of the renewed invasion of Ukraine, it has been announced today by the UK and international allies.

Source: National Cyber Security Center

Basic terms of information security

State-sponsored hacking groups

Every country has a state-sponsored hacking group. These include Kimsuky, Lazarus, and AndAriel, which are sponsored by North Korea, and Fancy Bear (APT28) and Cozy Bear (APT29), which are sponsored by Russia.

- North Korea-sponsored hacking groups
 - Kimsuky
 - Achieve its goals in a variety of ways
 - Very good at spearfishing
 - Develop browser extensions, remote access trojans, and more
 - Lazarus
 - Theories on the origin of the group's name
 - Named after a character in the game Diablo
 - Named after a biblical character, Lazarus, who rose from the dead because the Sony Pictures attackers kept reappearing
 - Affiliated with North Korea's General Reconnaissance Office, which develops viruses and ransomware such as WannaCry.
 - BlueNorOff, AndAriel exist as sub-units.

Sources : Boan News (2019-01-22), Los Angeles Times (2017-05-18)

Basic terms of information security

State-sponsored hacking groups

Every country has a state-sponsored hacking group. These include Kimsuky, Lazarus, and AndAriel, which are sponsored by North Korea, and Fancy Bear (APT28) and Cozy Bear (APT29), which are sponsored by Russia.

- Russian-sponsored hacking groups
 - Fancy Bear
 - American cybersecurity technology company CrowdStrike Holdings, Inc. announced that Fancy Bear has ties to the Russian military intelligence agency GRU.
 - In 2018, the U.S. special counsel indicted Fancy Bear as GRU unit 26165.
 - The number indicates the index of Russian military units.
 - Its specialty is conducting state-sponsored cyberattacks and decoding hacked data.
 - In July 2023, Fancy Bear's headquarters exploded and collapsed after being targeted by Ukrainian drones.
 - Cozy Bear
 - Hacking group run by two Russian intelligence agencies, the Federal Security Service (FSB) and the Foreign Intelligence Service (SVR).
 - Between 2014 and 2016, distributed malware to Android phones of anti-Russian militants in rural Ukraine
 - Mainly conducts APT attacks without attack patterns and targets people.

Source : AP News (2018-07-13)

Basic terms of information security

Lapsus\$ attack case

Lapsus\$ is the most active ransomware gang. The attacks have broadened their scope to include both private and public entities, in addition to government agencies in Costa Rica and Peru and a private military training company in Canada.

- Lapsus\$ attack victimization milestones

| | |
|------------|---|
| 2022-01-03 | Portugal's largest media company, Impresa, was attacked by Lapsus\$. |
| 2022-03-02 | Lapsus\$ stole 1TB of data in NVIDIA hack. |
| 2022-03-07 | Lapsus\$ hacked into Samsung servers and stole confidential data including source code. |
| 2022-03-22 | Lapsus\$ hacked LG Electronics and stole employee email accounts. |
| 2022-03-23 | Lapsus\$ hacked global security company Okta. |
| 2022-03-24 | Lapsus\$ hacked Microsoft. |
| 2022-03-31 | Lapsus\$ hacked global IT company Globant. |
| 2022-04-22 | Lapsus\$ stole T-Mobile source code. |

Basic terms of information security

Lapsus\$ attack case

Lapsus\$ is the most active ransomware gang. Targeting both private and public entities, they have expanded their attacks to include government agencies in Costa Rica and Peru, and a private military training company in Canada.

- Lapsus\$ primary methods of attack are as follows :
 - Aggressively gather information to select and infiltrate targets using social engineering attack techniques.
 - Use it to get in Specifically, gather information to bypass multi-factor authentication
 - Gather information in the following ways :
 - Obtain credentials and multi-factor authentication information
 - from target organization employees or vendors
 - Purchase credentials and session tokens on the black market
 - Deploy RedLine Stealer, a password-stealing malware
 - Search public code repositories for exposed credentials

LAPSUS\$
We recruit employees/insider at the following!!!!

- Any company providing Telecommunications (Claro, Telefonica, ATT, and other similar)
- Large software/gaming corporations (Microsoft, Apple, EA, IBM, and other similar)
- Callcenter/BPM (Atento, Teleperformance, and other similar)
- Server hosts (OVH, Locaweb, and other similar)

TO NOTE: WE ARE NOT LOOKING FOR DATA, WE ARE LOOKING FOR THE EMPLOYEE TO PROVIDE US A VPN OR CITRIX TO THE NETWORK, or some anydesk

If you are not sure if you are needed then send a DM and we will respond!!!!

If you are not a employee here but have access such as VPN or VDI then we are still interested!!

You will be paid if you would like. Contact us to discuss that

@lapsusjobs

837 37.2K 2:37 PM

A message in a Telegram channel sent by Lapsus\$ to collect information from a targeted company.

Basic terms of information security

Lapsus\$ attack case

Lapsus\$ is the most active ransomware gang. Targeting both private and public entities, they have expanded their attacks to include government agencies in Costa Rica and Peru, and a private military training company in Canada.

- Lapsus\$ primary methods of attack are as follows :
 - Use this information to gain access to Internet-connected systems (VPN, VID, RDP, and AD) and infiltrate the target organization
 - For organizations that use multi-factor authentication, (1) use a session token replay attack; and,
 - (2) Use stolen passwords to prompt legitimate users for multi-factor authentication
 - Bypass multi-factor authentication by forcing tricked users to provide the necessary authorizations

Basic terms of information security

Lapsus\$ attack case

Lapsus\$ is the most active ransomware gang. Targeting both private and public entities, they have expanded their attacks to include government agencies in Costa Rica and Peru, and a private military training company in Canada.

- Lapsus\$ primary methods of attack are as follows :
 - Once inside, they use a variety of tactics to gain privileges to access sensitive information
 - Use Active Directory (AD) Explorer to find a complete list of users and which ones have high privileges
 - Use SharePoint, Confluence, Jira, GitLab, GitHub, Teams, Slack, etc.
 - Explore credentials for high privileges in popular enterprise solutions and code repositories
 - Exploit privilege escalation vulnerabilities in Confluence, Jira, and GitLab
 - Call the helpdesk and have them reset the credentials of authorized personnel using the information gathered
 - Call centers are vulnerable to these attacks because they are often outsourced.
 - After a targeted data breach, wipe the system and resources, and delete the cloud global administrator account.
 - Make it harder to track and recover; monitor victims' crisis boards to understand how they're responding to the incident

Basic terms of information security

Lapsus\$ attack case

Lapsus\$ is the most active ransomware gang. Targeting both private and public entities, they have expanded their attacks to include government agencies in Costa Rica and Peru, and a private military training company in Canada.

- Characteristics of Lapsus\$ attacks
 - Lapsus\$ relies heavily on social engineering attacks, not sophisticated technology or malware
 - Gather and use information to get inside the target
- What to do
 - Use passwordless FIDO or two-factor (2FA) authentication inside and outside the organization
 - Avoid using vulnerable multi-factor authentication
 - Text messages, emails, and simple pushes that are vulnerable to SIM swapping or phone-based social engineering
 - When using multi-factor authentication methods, use secure ones, such as random number-based OTP

Basic terms of information security

Lapsus\$ attack case

The Lapsus\$ hacking group first reportedly began hacking Electronic Arts in July 2021, and became known for hacking and leaking inside information from U.S. semiconductor company Nvidia. It was later found to have hacked a number of global big tech companies, including Samsung, LG, and Microsoft.

- Collecting information
 - Purchase targeted employee information on the dark web
 - Use purchased information to send phishing emails to targets with malware compromised accounts
 - Distribute account leakage malware and obtain employee account information through hacking attacks from multiple sources, including hack forums
 - Likely to remotely access the attack site using collected employee account information
 - VDI, VPN, Web, email, and more

Basic terms of information security

Lapsus\$ attack case

The Lapsus\$ hacking group first reportedly began hacking Electronic Arts in July 2021, and became known for hacking and leaking inside information from U.S. semiconductor company Nvidia. It was later found to have hacked a number of global big tech companies, including Samsung, LG, and Microsoft.

- Initial entry
 - Lapsus\$, when attacking, uses target access and account information gathered in previous phases of the attack.
 - Likely to gain easy access to the user's PC
 - Access was possible due to lack of two-factor authentication
 - Attempts to find PCs that are exempt from internal policies
 - Likely because authentication was only possible with the previously obtained ID/PW

Basic terms of information security

Lapsus\$ attack case

The Lapsus\$ hacking group first reportedly began hacking Electronic Arts in July 2021, and became known for hacking and leaking inside information from U.S. semiconductor company Nvidia. It was later found to have hacked a number of global big tech companies, including Samsung, LG, and Microsoft.

- Moving to internal systems
 - Additional authentication is required to access the user's PC and to access actual internal systems or applications.
 - If the additional authentication scheme is system-based rather than possession-based, additional authentication information can be easily obtained.
 - The group has since been identified as also targeting vulnerabilities in the configuration management system.
 - The applications used were Jira and Confluence, the group revealed via Telegram.
 - The vulnerability is not discussed in detail
- Information leak
 - Likely pieced together the accounts it acquired and reviewed the information for each account one by one.
 - It is believed to have collected each company's sensitive files and sent them to the exfiltration site.

Basic terms of information security

Lapsus\$ attack case

| Stage of the incident | Top incident causes | What to do | Solutions/Services |
|---|--|--|---|
| "Obtain account information in advance" - VDI, VPN, and email account compromised | <ul style="list-style-type: none"> Lack of employee account breach monitoring <ul style="list-style-type: none"> - Dark web, hack forums, SNS - Email phishing, malware | <ul style="list-style-type: none"> Enhance employee account breach monitoring <ul style="list-style-type: none"> - Set up a monitoring system - Detect/block/take action | ✓ Account leak detection solutions ✓ E-mail APT solution |
| "Initial intrusion" - Access to compromised account | <ul style="list-style-type: none"> Lack of multi-factor authentication <ul style="list-style-type: none"> - Exceptions, non-possession based authentication Lack of remote terminal access security policy <ul style="list-style-type: none"> - IP, MAC authentication | <ul style="list-style-type: none"> Check multi-factor exceptions Apply possession-based multi-factor Apply endpoint pre-authorization policy | ✓ Multi-factor authentication solutions |
| "Internal network intrusion" - Internal system vulnerabilities exist - Undetected malware | <ul style="list-style-type: none"> Lack of employee account breach monitoring Vulnerabilities exist in collaborative system <ul style="list-style-type: none"> - Jira, Confluence, and more Poor malware detection | <ul style="list-style-type: none"> Establish a system for monitoring employee misbehavior Collaboration system security patches <ul style="list-style-type: none"> - Receive regular security patch updates - Apply the latest security patches | ✓ Fraud Detection System (FDS) ✓ Patch management system ✓ Vulnerability diagnostic service ✓ N/W APT, EDR solutions |
| "Information leaks" - Inadequate encryption of sensitive data - Undetected breaches | <ul style="list-style-type: none"> Insufficient encryption of sensitive data <ul style="list-style-type: none"> - Diagrams, source code, and more Lack of information leakage monitoring | <ul style="list-style-type: none"> Identify and encrypt sensitive data Establish a breach detection and prevention system <ul style="list-style-type: none"> - Bulk, fragmented external traffic | ✓ Encryption solutions ✓ SIEM solution |

Basic terms of information security

Lapsus\$ attack case

Based on the official reports of Telegram and Microsoft on the Lapsus\$ attack group, which was the hottest topic in the first half of 2022, and the report of the Ministry of Science and ICT in South Korea, the following is a summary of the attack techniques and countermeasures that can be provided accordingly.

- Information gathering stage
 - Dark web monitoring
 - It's crucial to monitor how information is circulating on the dark web, deep web, forums, social media, etc
 - This will help you determine which users or systems have been compromised.
 - Account changes and system checks can prevent further damage.
 - Email malware detection/blocking solutions
 - Email hacking attacks are now more prevalent than ever.
 - Effectively block malware and eliminate threats at the system level, not just at the user level.
 - APT detection/blocking solutions
 - If your current level of security is at the network end, you need to bring that line down to the host end.
 - Behavior-based detection overcomes pattern-based limitations and increases security visibility.

Basic terms of information security

Lapsus\$ attack case

Based on the official reports of Telegram and Microsoft on the Lapsus\$ attack group, which was the hottest topic in the first half of 2022, and the report of the Ministry of Science and ICT in South Korea, the following is a summary of the attack techniques and countermeasures that can be provided accordingly.

- Initial intrusion stage
 - Block unnecessary remote locations
 - With the recent COVID-19 work-from-home mandate, remote access is much more prevalent than ever before.
 - This creates the need to understand the current status of unnecessary remote locations and block them if they're not needed.
 - When they are unavoidably needed, use minimal access controls and authorization to eliminate threats.
 - Two-factor authentication and exception handling
 - If remote access to the enterprise is unavoidable, two-factor, possession-based authentication must be implemented.
 - Perform health checks to find gaps in exception handling and remediate threats.

Basic terms of information security

Lapsus\$ attack case

Based on the official reports of Telegram and Microsoft on the Lapsus\$ attack group, which was the hottest topic in the first half of 2022, and the report of the Ministry of Science and ICT in South Korea, the following is a summary of the attack techniques and countermeasures that can be provided accordingly.

- Internal systems entry stage
 - Apply the latest security patches
 - Applying the latest security patches is the most effective security practice that doesn't require an up-front investment.
 - It takes time to implement, depending on the level of impact on each system.
 - Applying security patches as soon as they are released is the only way to keep internal systems as free of threats as possible.

Basic terms of information security

Lapsus\$ attack case

Based on the official reports of Telegram and Microsoft on the Lapsus\$ attack group, which was the hottest topic in the first half of 2022, and the report of the Ministry of Science and ICT in South Korea, the following is a summary of the attack techniques and countermeasures that can be provided accordingly.

- Information leakage stage
 - Strengthen the DRM solution and the sphere of control
 - When deploying a DRM solution, document classes and policies should be created for each organization's critical materials.
 - based on a thorough assessment, rather than focusing on encrypting specific document files
 - Need to apply encryption to sensitive document classes
 - Need a real-time detection/blocking monitoring scheme for bulk file decryption
 - Operate a system that allows hackers to steal internal information, but blocks them from exposing it on external sites
 - Data breach detection solutions and policies
 - When attackers try to steal information, they break up files into smaller ones, usually in the form of compressed files.
 - Register information leakage policies with devices such as internal SIEM solutions
 - Ensure that breaches are detected and stopped early

Basic terms of information security

Definition of information security

Security ethics emphasize adherence to ethical principles in information security, thereby contributing to the protection of personal privacy and the creation of a trustworthy society. Hacking behaviors such as illegal system intrusions and information theft are punishable by law, with varying degrees of legal sanctions in different countries.

- Cases of punishment for malicious hacking

Rockstar Games GTA 6 Hacker Given Unconventional Life Punishment after Stealing and Leaking Game Last Year

Arion Kurtaj was a member of hacker group Lapsus\$.

Written by: Rohit

PUBLISHED DECEMBER 22, 2023, 9:55 AM



That's not all as Kurtaj also broke into Rockstar's Slack system and Confluence wiki, and declared that if the company does not contact him in 24 hours on Telegram he will start releasing the source code of the game and that was when he posted the source code and video footage on GTAForums with a link to a RAR archive, leading to his arrest and detention.

Source: FandomWire

Basic terms of information security

Definition of information security

Security ethics emphasize adherence to ethical principles in information security, thereby contributing to the protection of personal privacy and the creation of a trustworthy society. Hacking behaviors such as illegal system intrusions and information theft are punishable by law, with varying degrees of legal sanctions in different countries.

- Cases of punishment for hacking derived from poor ethics
 - Lee, who infected about 470,000 PCs at 7,459 PC shops nationwide in South Korea in 2016 and exploited them as zombie PCs for four years, is said to be an IT startup entrepreneur and a computer science major dropout from a prominent university in the Seoul metropolitan area. Lee has 16 years of experience as a **programmer**, creating and distributing malware using sophisticated techniques to evade detection by antivirus and other security solutions.
 - Yang, **an internet security expert**, was arrested in 2015 for hacking and crippling a competing gambling site in exchange for 1 billion won from an illegal gambling operator, and Kim, the programmer who exploited a vulnerability in an online payment system in November 2014, was said to have been a programmer for six years, after graduating from a prestigious university with a degree in computer science.
 - College students were found to have systematically hacked into professors' accounts and computer systems to manipulate attendance, assignments, and grades. They came from their college **hacking/security club**.

The skills of a security professional are a double-edged sword.

**Used correctly, they provide security and stability for information assets;
used carelessly, they can lead to vulnerabilities and threats.**

Basic terms of information security

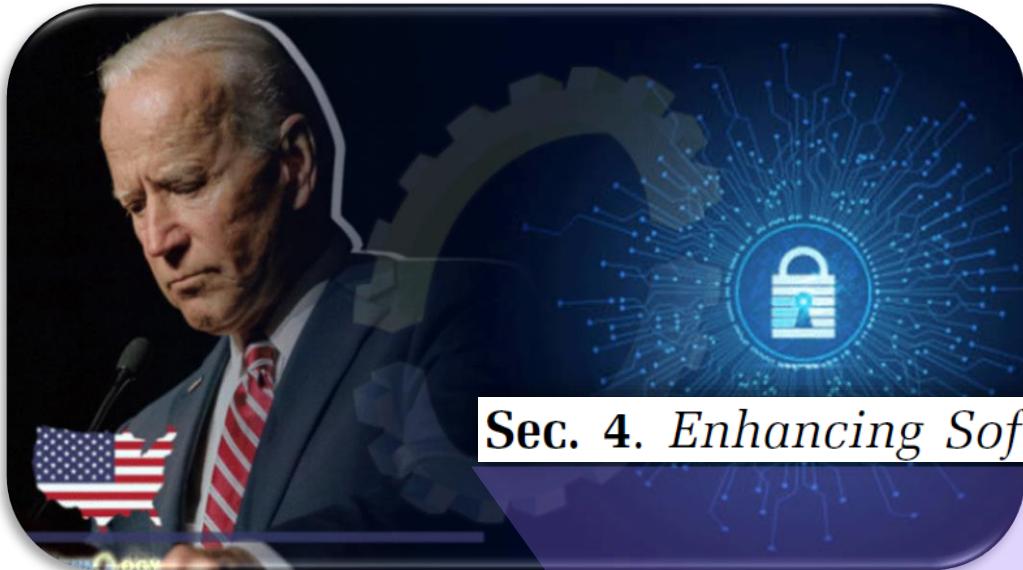
Definition of information security

Security ethics emphasize adherence to ethical principles in information security, thereby contributing to the protection of personal privacy and the creation of a trustworthy society. Hacking behaviors such as illegal system intrusions and information theft are punishable by law, with varying degrees of legal sanctions in different countries.

- Sound information security ethics
 - The quality that is more important than the skills of information security professionals (ethical/white hat hacker)
 - Perform system diagnostics and penetration testing
 - Access to a wide variety of information blurs ethics, and standards, and gives rise to malicious ideas.
 - A single cybercrime can destroy one's credibility as an information security professional.
 - Skills can be built through study, but trust cannot be rebuilt once it has been broken.

Basic terms of information security

Cybercrime prevention efforts by law enforcement



Sec. 4. Enhancing Software Supply Chain Security.

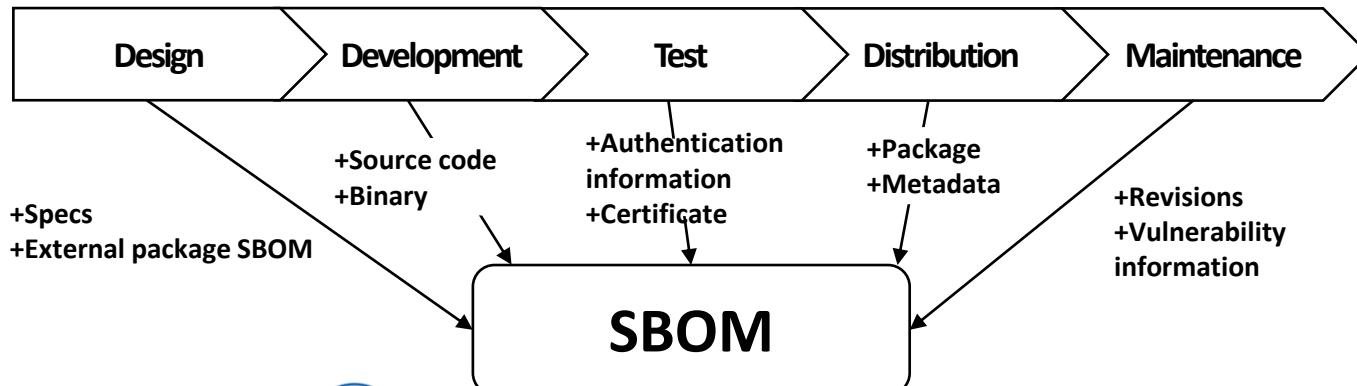
Sec. 4. Enhancing Software Supply Chain Security. (a) The security of software used by the Federal Government is vital to the Federal Government's ability to perform its critical functions. The development of commercial software often lacks transparency, sufficient focus on the ability of the software to resist attack, and adequate controls to prevent tampering by malicious actors. There is a pressing need to implement more rigorous and predictable mechanisms for ensuring that products function securely, and as intended. The security and integrity of "critical software"—software that performs functions critical to trust (such as affording or requiring elevated system privileges or direct access to networking and computing resources)—is a particular concern. Accordingly, the Federal Government must take action to rapidly improve the security and integrity of the software supply chain, with a priority on addressing critical software.

Basic terms of information security

Cybercrime prevention efforts by law enforcement

President Biden issued an executive order (EO 14028) in May 2021 to improve the nation's cybersecurity, in response to security threats particularly the SolarWinds software supply chain attack that year and the ubiquitous Log4j vulnerability in 2021.

- Mandatory SBOM
 - The executive order consists of 11 sections.
 - Section 4 describes the SBOM required to improve software supply chain security.
 - The Executive Order has taken SBOMs from a nice-to-have to a must-have.
- Software Bill of Materials (SBOM)
 - Formal document that outlines the supply chain relationships and component details for building software.



Sources : Cybellum Homepage (2023-06-11), White House Homepage (2021-05-12)

02

Linux fundamentals – introduction

- Linux overview
- Understanding the shell
- Mastering Linux essential commands
- Linux package

Linux overview

Unix system types

Unix has been developed as System V and Berkeley Software Distribution (BSD), each with slightly different features depending on the developer.

- Unix system types
 - System V R5.0
 - The official name of Unix, developed at Bell Labs as the standard version of Unix.
 - SCO Unix
 - The name of Santa Cruz Operation's SCO Open Desktop and SCO Open Release 3, an implementation of Unix SVR3.2.5.
 - SunOS
 - Sun's BSD family of operating systems
 - Solaris
 - One of the most popular Unix operating systems
 - Sun's implementation of SVR4



Linux overview

Unix system types

Unix has been developed as System V and Berkeley Software Distribution (BSD), each with slightly different features depending on the developer.

- Unix system types
 - HP-UX
 - An operating system developed by HP as a variant of SVR4.
 - AIX
 - IBM's System V family of operating systems, with a mix of SVR4, BSD, and OSF/1 features
 - Free UNIX-style operating system for Intel processors



Linux was created to rebel against commercial Unix and advocate an open license. Most of the operating system is free and open source.

- Overview

- The Free Software Foundation (FSF) was formed in response to proprietary Unix, which developed GNU's Not Unix (GNU), a clone of Unix.
- The Free Software Foundation created the GNU General Public License (GNU GPL) to prevent it from being turned into proprietary software, and when that effort stalled, software engineer Linus Torvalds developed Linux.
- Open source software in which the Linux kernel and other components are free, unlike other operating systems.
- Highly portable on PCs or other devices

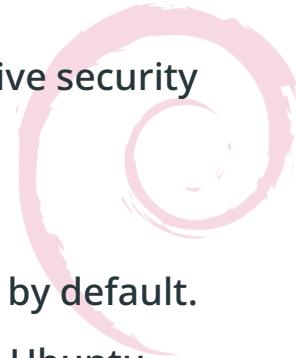


Linux overview

Linux family

Linux is an open source operating system, and there are many derivatives. Quite a few will turn up in a simple search.

- Debian-based systems
 - Ubuntu
 - A popular Linux-like operating system first announced by Canonical Ltd. on October 20, 2004
 - Most popular on personal laptops and desktops (the server-specific operating system also exists.)
 - Kali Linux
 - An operating system that is designed for simulated hacking or offensive security
 - Multiple hacking tools included
 - Linux Mint
 - An operating system that includes Java, Flash web plug-ins, and more by default.
 - Relatively more focus on the cosmetic beauty of the GUI compared to Ubuntu



Linux overview

Linux family

Linux is an open source operating system, and there are many derivatives. Quite a few will turn up in a simple search.

- Red Hat systems
 - Fedora
 - A commercial general-purpose operating system created by the Fedora project under the auspices of Red Hat Enterprise Linux (RHEL).
 - RHEL was developed based on Fedora Linux.
 - RHEL
 - Linux distributions developed by Red Hat
 - Technical support from Red Hat instead of being sold as a paid license
 - Free to license for development
 - Community Enterprise Operating System (CentOS)
 - This is intended to be a near-perfect reflection of RHEL since Fedora hasn't been able to keep up with RHEL's technical changes.
 - Open source operating system utilized by large Korean companies such as Kakao and Naver



Linux overview

Linux family

Linux is an open source operating system, and there are many derivatives. Quite a few will turn up in a simple search.

- Other Linux distributions
 - Mandriva Linux
 - ALT, Mageia
 - PCLinuxOS (PCLOS)
 - Slackware
 - VectorLinux (VL)
 - Frugalware
 - Gentoo
 - Sabayon
 - Arch Linux
 - Manjaro



Understanding the shell

Shell overview

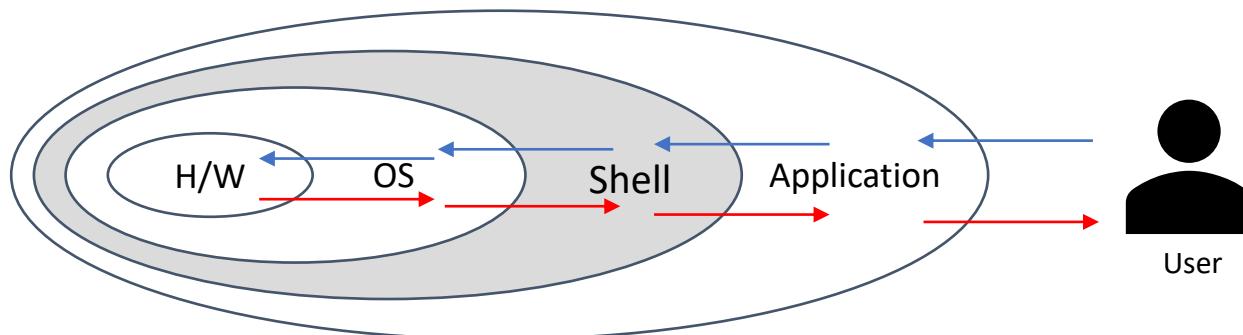
The shell sits between the application and the operating system and acts as a bridge between the user and the system, while receiving, interpreting, and executing commands from the user.

- Definitions

- A program that provides an interface to implement various operating system functions and services on an operating system
- Used to mean the layer that wraps around the interface between the user and the internals (kernel) of the operating system

- Descriptions

- CLI : provide a command line interface (CLI)
- GUI : provide a graphical user interface (GUI)



Understanding the shell

Shell overview

The shell sits between the application and the operating system and acts as a bridge between the user and the system, while receiving, interpreting, and executing commands from the user.

- Features
 - Command interpreter
 - Interpret and pass commands between you and the kernel
 - Programming
 - Create a single program that repeatedly performs a task using multiple commands (shell script)
 - User preference settings
 - Set up an environment using the init file feature
 - Run an init file at login to set the user's initial preferences

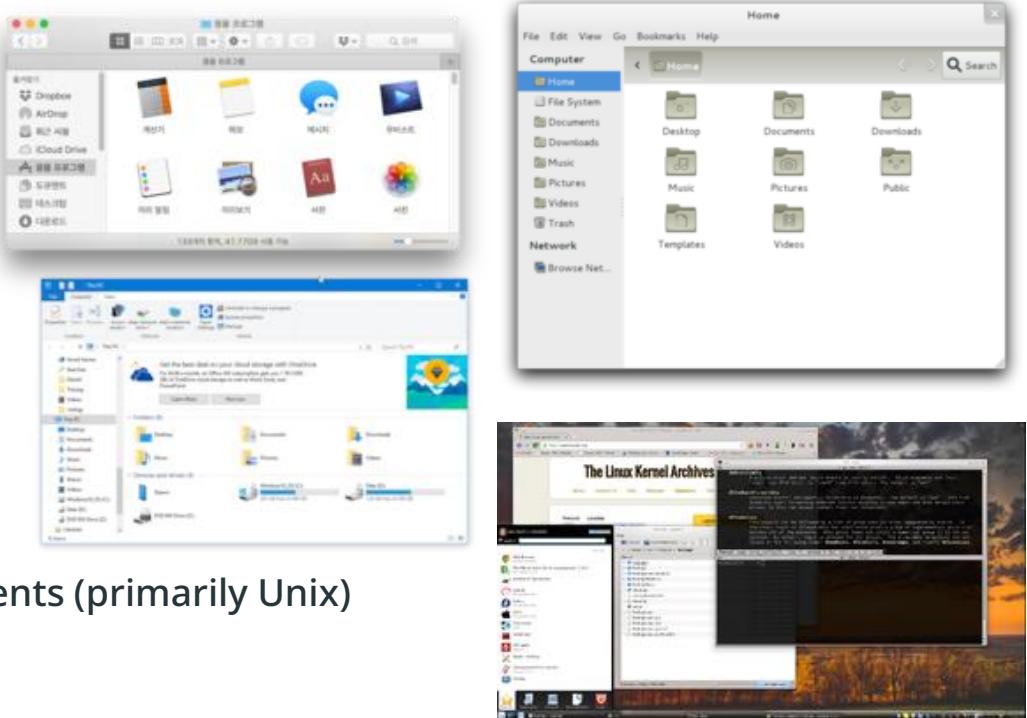
Understanding the shell

Shell types

There are two types of shells : graphical user interface (GUI) and command-line interfaces (CLI) shells, of which Windows Explorer, Macintosh Finder, and X Window are the most common. For CLIs, bash shells and command prompts (cmd.exe, cmd) are commonly used.

- GUI shells

- MS Windows environment
 - Windows Explorer
 - LiteStep
 - Geoshells
 - Blackbox for Windows (BB4Win)
 - Emerge Desktop
- Macintosh Finder
- X Windows system-based environments (primarily Unix)



Understanding the shell

Shell types

There are two types of shells : graphical user interface (GUI) and command-line interfaces (CLI) shells, of which Windows Explorer, Macintosh Finder, and X Window are the most common. For CLIs, bash shells and command prompts (cmd.exe, cmd) are commonly used.

- CLI shells
 - Unix shell
 - Bourne shell (sh) : executable file that replaced the first shell, Thompson Shell, developed by Stephen Bourne at AT&T Bell Labs
 - Almquist shell (ash)
 - Bash shell : a shell that extends various features from the main shell (sh)
 - C shell (csh)
 - Tenex C shell (tcsh)
 - KornShell (ksh), Scheme shell (Scsh), Z shell (zsh)
 - COMMAND.COM : a shell for DOS
 - cmd.exe : a shell for the OS/2 text mode and Windows NT

Understanding the shell

Shell types

There are two types of shells : graphical user interface (GUI) and command-line interfaces (CLI) shells, of which Windows Explorer, Macintosh Finder, and X Window are the most common. For CLIs, bash shells and command prompts (cmd.exe, cmd) are commonly used.

- Features of the key Linux shells

| Shell type | Feature |
|--------------------------------|---|
| Sh (Bourne shell) | <ul style="list-style-type: none">- The default shell for Unix version 7, developed in 1977 |
| Bash (Bourne-again shell) | <ul style="list-style-type: none">- Developed for the GNU Project and based on sh- Currently the standard Linux shell, compatible with the sh command syntax- Include useful features from ksh and csh to support command history, command completion, history substitution, command line editing, and more |
| Csh (C shell) | <ul style="list-style-type: none">- Built on the C language, with powerful programming capabilities- Include useful features such as history, aliases, task control, and more |
| Tcsh (the enhanced C shell) | <ul style="list-style-type: none">- An operating system called TENEX was created with command line completion and integration with csh- A feature-rich shell for csh with command completion, command line editing, and more. |
| Ksh (KornShell) | <ul style="list-style-type: none">- Just as bash, sh can be extended to develop.- Incorporate many of the features of csh, including task control, nickname capabilities, history capabilities, command line editing, etc. |

Understanding the shell

Shell types

- Login shell related environment variable: SHELL
- How to determine the currently set shells

```
$ echo $SHELL  
/bin/bash
```

- Check for changeable shells

```
$ chsh -l  
/bin/sh  
/bin/bash  
/sbin/nologin  
/usr/bin/sh  
/usr/bin/bash  
/usr/sbin/nologin  
/bin/tcsh  
/bin/csh  
$ cat /etc/shells  
/bin/sh  
/bin/bash  
/sbin/nologin  
/usr/bin/sh  
/usr/bin/bash  
/usr/sbin/nologin  
/bin/tcsh  
...
```

Understanding the shell

Check shells

- chsh
 - Command to change the user login shell
 - Changed shell will be effective from the next login

```
$ echo $SHELL  
/bin/bash  
  
$ chsh  
Changing shell for user1.  
New shell [/bin/bash]: /bin/sh  
암호 :  
Shell changed.  
  
$ echo $SHELL  
/bin/sh
```

- Check user login shell information

```
$ cat /etc/passwd | grep user1  
user1:x:501:501::/home/user1:/bin/bash
```

Understanding the shell

Input/output in shell script

- echo
 - Command to print a given string to standard output, including spaces and newlines
 - Usage : echo [option] [string]

```
$ echo Hello World  
Hello World  
$ echo Hello World\n  
Hello World\n  
$ echo -e "Hello World\n"  
Hello World  
  
$ echo -n "Hello World"  
Hello World $
```

| Option | Description |
|--------|---|
| -n | Do not print the newline following the last one |
| -e | Recognize escape characters that combine with a backslash (\) in a string, enclosed in quotation marks ("") |

Understanding the shell

Input/output in shell script

- List of escape characters

| Option | Description |
|--------|--|
| \a | Output a beep sound |
| \b | Backspace |
| \c | Don't print last newline |
| \f | Print to form-feed format (change paper on printer) |
| \n | Print newline |
| \r | Carriage return(↵) : a control character or structure used to start a new line of characters |
| \t | Horizontal tab |
| \v | Vertical tab |
| \\\ | Backslash |

Understanding the shell

Input/output in shell script

- Redirections
 - Saving the standard output value of a command to a specific file
 - '>' and '>>'
 - > : Enter a new standard output value for the command, starting at the beginning of the file
 - >> : Enter the standard output value of the command followed by the end of the file

```
$ Command > Filename  
$ Command >> Filename
```

- Writing scripts

```
$ vi .test.sh  
#!/bin/sh  
echo "hello world!!" > file.txt  
echo "hi" >> file.txt
```

- Script execution results

```
$ chmod +x test.sh  
$ ./test.sh  
$ cat file.txt  
hello world!!  
hi
```

Understanding the shell

Input/output in shell script

- Pipes
 - Used to link commands with commands
 - How to use "|"

```
$ command | command  
$ command | command | command ...
```

- Example run

```
$ cat /etc/passwd | grep user1  
user1:x:501:501::/home/user1:/bin/bash  
  
$ touch file.txt  
$ ls -al | grep file  
-rw-rw-r--. 1 user1 17 1월 28 18:07 file.txt
```

Mastering Linux essential commands

Shutting down and restarting the system

The shutdown command is a command that allows you to shut down and restart the Linux system. There are options to customize the command for specific situations.

- Shut down the system immediately

```
$ shutdown -h now  
$ halt  
$ init 0
```

- Restart immediately

```
$ shutdown -r now  
$ reboot  
$ init 6
```

- Example Shutdown command options

```
$ shutdown -h 10m      : end in 10 minutes  
$ shutdown -r 22:00    : shutdown and restart at 10pm  
$ shutdown -c          : cancel scheduled shutdown
```

Mastering Linux essential commands

[View a list of files and directories](#)

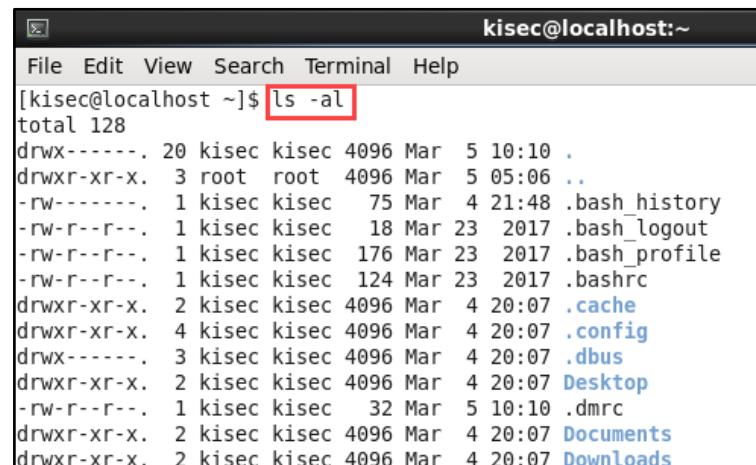
The ls command stands for list segments and is a command that performs the function of displaying a list of files. If you run the ls command with no options, it displays a list of files in the current directory.

● The ls command

```
$ ls  
Desktop Documents Downloads Music Pictures Public Templates Videos
```

- Additional options and examples

| Option | Description |
|---------|--|
| -a | Output all files in a directory (including hidden files) |
| -l | Display file information |
| -s | Show file size in blocks |
| -t | Output sorted by file last modified time |
| --color | Show colors based on file type |
| -R | Output both the current working directory and subdirectories |
| --help | Help |



A screenshot of a terminal window titled "kisec@localhost:~". The window shows the command "ls -al" being run. The output lists numerous files and directories in the current directory, including ".cache", ".config", ".dmrc", ".dbus", ".bashrc", ".profile", ".logout", ".history", and several desktop icons like "Desktop", "Documents", and "Downloads". The "ls -al" command is highlighted with a red box.

```
[kisec@localhost ~]$ ls -al  
total 128  
drwx----- 20 kisec kisec 4096 Mar  5 10:10 .  
drwxr-xr-x.  3 root  root  4096 Mar  5 05:06 ..  
-rw----- 1 kisec kisec   75 Mar  4 21:48 .bash_history  
-rw-r--r-- 1 kisec kisec  18 Mar 23 2017 .bash_logout  
-rw-r--r-- 1 kisec kisec  176 Mar 23 2017 .bash_profile  
-rw-r--r-- 1 kisec kisec 124 Mar 23 2017 .bashrc  
drwxr-xr-x.  2 kisec kisec 4096 Mar  4 20:07 .cache  
drwxr-xr-x.  4 kisec kisec 4096 Mar  4 20:07 .config  
drwx-----  3 kisec kisec 4096 Mar  4 20:07 .dbus  
drwxr-xr-x.  2 kisec kisec 4096 Mar  4 20:07 Desktop  
-rw-r--r--  1 kisec kisec  32 Mar  5 10:10 .dmrc  
drwxr-xr-x.  2 kisec kisec 4096 Mar  4 20:07 Documents  
drwxr-xr-x.  2 kisec kisec 4096 Mar  4 20:07 Downloads
```

Mastering Linux essential commands

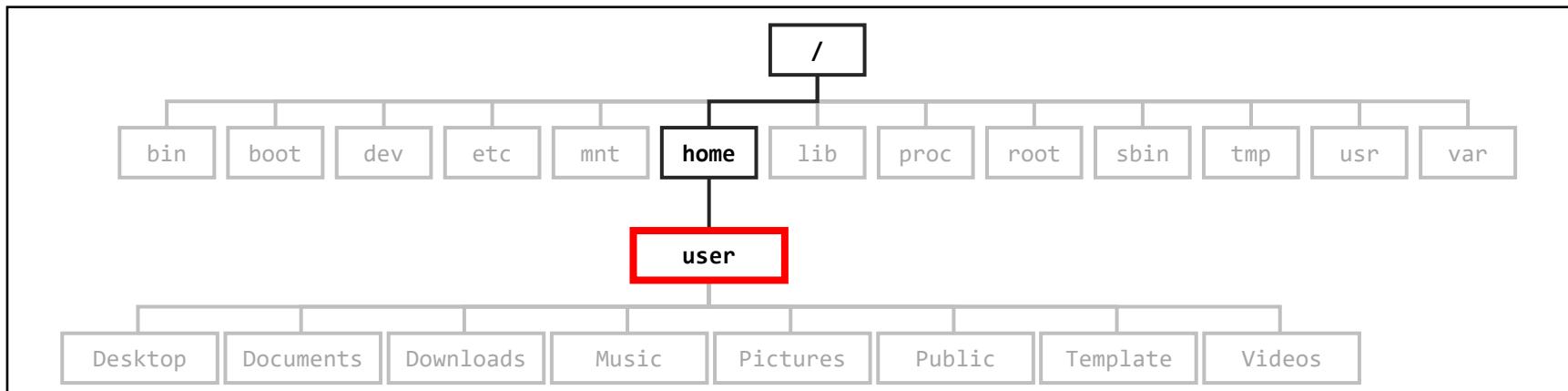
Print the current directory name

The pwd command stands for the print name of working directory and prints the name of the directory you are working in.

- The pwd command

```
$ pwd  
/home/user
```

- Current location



Mastering Linux essential commands

Change the current directory

The cd command stands for change directory and allows you to change the current directory address using both relative and absolute paths.

- The cd command
 - Examples of using absolute paths
 - Enter all the directories you want to change, starting with the top-level (root) directory.

```
$ cd /home/user/Desktop/  
$ pwd  
/home/user/Desktop
```

- Example of using relative paths
 - Change by typing relative to the current directory

```
$ pwd  
/home  
$ cd ./user/Desktop/  
$ pwd  
/home/user/Desktop
```

Mastering Linux essential commands

Copy a file or directory

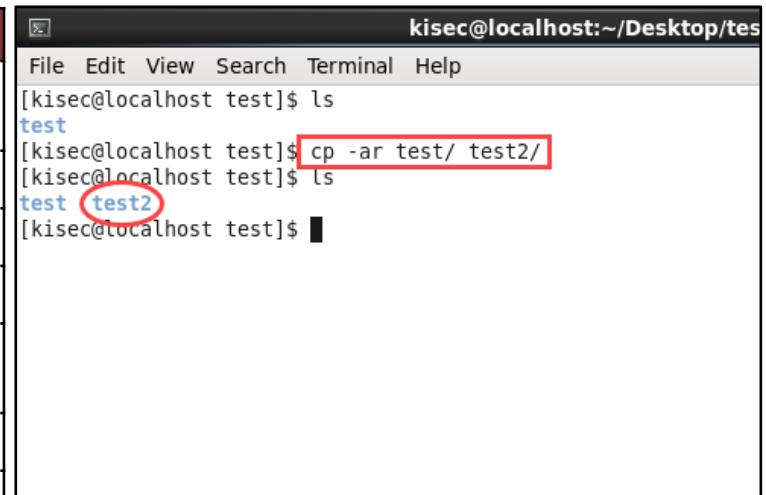
The cp command stands for copy and is used to copy a file or directory. If you copy without the desired path, it will automatically copy to the current path.

- The cp command
 - How to use

```
cp [option] [source file] [desired path and filename (extension)]  
cp [option] [source directory] [desired path and directory name].
```

- Additional options and examples

| Option | Description |
|--------|---|
| -a | Copy the original file while preserving its properties and link information |
| -b | Create a backup file |
| -d | Copy the symbolic file itself with the symbolic information |
| -f | Delete and copy files when they exist |
| -p | Copy the original file while preserving ownership, groups, permissions, and timeouts. |
| -r | Copy all files in a subdirectory in their entirety |
| -u | Copy when the source file is newer than the destination file |



```
kisec@localhost:~/Desktop/tes  
File Edit View Search Terminal Help  
[kisec@localhost test]$ ls  
test  
[kisec@localhost test]$ cp -ar test/ test2/  
[kisec@localhost test]$ ls  
test test2  
[kisec@localhost test]$
```

Mastering Linux essential commands

Deleting files

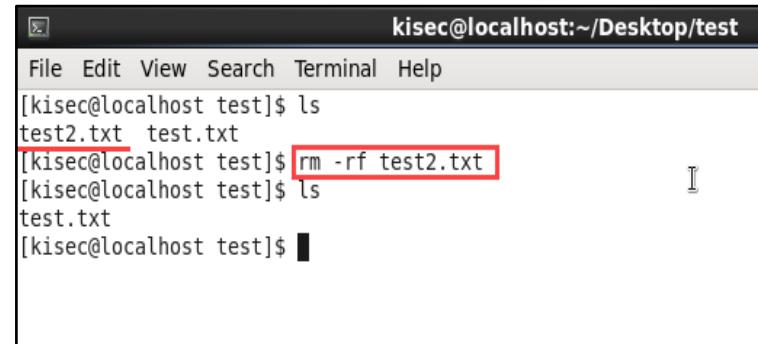
The rm command stands for **r**emove and is used to delete unneeded files.

- The rm command
 - How to use

```
rm [option] [filename]
```

- Additional options and examples

| Option | Description |
|--------|--|
| -d | Delete a directory |
| -f | Force delete without prompting for confirmation |
| -i | Ask if you want to delete each one when deleting |
| -r | Delete all files, including subdirectories |
| -v | Shows files before they were deleted |



A screenshot of a terminal window titled "kisec@localhost:~/Desktop/test". The window shows a command-line interface with the following session:

```
File Edit View Search Terminal Help
[kisec@localhost test]$ ls
test2.txt  test.txt
[kisec@localhost test]$ rm -rf test2.txt
[kisec@localhost test]$ ls
test.txt
[kisec@localhost test]$
```

The command `rm -rf test2.txt` is highlighted with a red rectangle. The terminal window has a dark theme with light-colored text.

Mastering Linux essential commands

Create and delete directories

The `mkdir` command, which stands for make directory, creates a directory. The `rmdir` command, which stands for remove directory, deletes a directory. However, it can delete it only if the directory is empty.

● The `mkdir` command

- Description
 - Create a new or additional directory
- How to use
 - `mkdir [option] created directory name`
(multiple names can be specified.)

| Option | Description |
|-----------------|---|
| <code>-p</code> | Create up to a specified number of subdirectories at once |

```
kisec@localhost:~$ ls  
[kisec@localhost test]$ mkdir test1 test2  
[kisec@localhost test]$ ls  
test1 test2  
[kisec@localhost test]$
```

| Option | Description |
|-----------------|---|
| <code>-p</code> | Delete specified subdirectories at once |

```
kisec@localhost:~$ ls  
test1 test2  
[kisec@localhost test]$ rmdir test1/ test2/  
[kisec@localhost test]$ ls  
[kisec@localhost test]$
```

Mastering Linux essential commands

Move and rename files

The mv command stands for move and is used to move or rename files. If no filename is specified, the existing filename is retained.

- The mv command
 - How to use

Move files : mv [source file] [path to move and filename to save]
Rename: mv [source file] [filename you want to change]

- Example of moving and renaming files

```
kisec@localhost:~/Desktop/t
File Edit View Search Terminal Help
[kisec@localhost test]$ ls
test.txt
[kisec@localhost test]$ mv test.txt /home/kisec/Desktop/
[kisec@localhost test]$ ls
[kisec@localhost test]$ ls /home/kisec/Desktop/
test test.txt
[kisec@localhost test]$
```

```
kisec@localhost:~/Des
File Edit View Search Terminal Help
[kisec@localhost test]$ ls
test.txt
[kisec@localhost test]$ mv test.txt abcd.txt
[kisec@localhost test]$ ls
abcd.txt
[kisec@localhost test]$
```

Mastering Linux essential commands

Find a file

The find command finds files that meet the criteria.

By default, it will search all subdirectories and find files that match the given parameters, from file name to modified time.

- The find command
 - How to use
 - `find [path] [expression] [specified conditions]`

| Expression | Description |
|----------------------|---|
| <code>-name</code> | Search for files that match a specified string pattern |
| <code>-empty</code> | Search for empty directories or files with zero size |
| <code>-delete</code> | Delete a found file or directory |
| <code>-exec</code> | Run the specified command on the found files |
| <code>-path</code> | Search in paths that match the specified string pattern |
| <code>-print</code> | Print the search results Search items are separated by newlines (default) |
| <code>-print0</code> | Print the search results Search items are separated by null |
| <code>-size</code> | Search for files using file size |
| <code>-type</code> | Search for files that match a specified file type |
| <code>-atime</code> | Search for files by file access time |

Mastering Linux essential commands

Find a file

The find command finds files that meet the criteria.

By default, it will search all subdirectories and find files that match the given parameters, from file name to modified time.

- The find command
 - How to use
 - `find [path] [expression] [specified conditions]`

| Expression | Description |
|------------------------|--|
| <code>-ctime</code> | Search for files by file content and the time property was changed |
| <code>-mtime</code> | Search for files by the time the data in the file was modified |
| <code>-mindepth</code> | Specify the minimum depth of subdirectories to start searching |
| <code>-maxdepth</code> | Specify the maximum depth of subdirectories to start searching |

Mastering Linux essential commands

Find a file

The find command finds files that meet the criteria.

By default, it will search all subdirectories and find files that match the given parameters, from file name to modified time.

- The find command

- Usage examples

- Root directory : find /
 - The current directory : find .

| | |
|------------------------------------|--|
| \$ find / -name "example" | : search for files and directories named 'example', starting with the top-level directory |
| \$ find . -name "example" | : search for files and directories named 'example' starting in the current directory |
| \$ find . -name "exam*" | : search for files starting with the specified string |
| \$ find . -name "*.txt" | : search for files ending with the specified string |
| \$ find . -name "*.txt" -delete | : delete files after searching for extensions |
| \$ find . -name "test" -type f | : search for a generic file named test |
| \$ find . -name "dir" -type d | : search only directories named dir |
| \$ find . -empty | : search for empty directories or files of size 0 |
| \$ find . -size 512c (+512, -512c) | : search for files with a file size of 512 bytes (multiple commands can specify over, under) |
| \$ find / -maxdepth 1 -name "etc" | : search for a file named etc only in the root (/) directory |
| \$ find / -user 427 -print | : files with owner's UID 427 in the entire directory |
| \$ find / -cmin -5 | : files created or updated within 5 minutes |

Mastering Linux essential commands

Check process status

The ps command stands for process status and allows you to check the status of the processes currently in use.

- The ps command
 - How to use

```
ps [option]
```

- Examples and options

```
Browse and run installed applications      kisec@localhost:~  
File Edit View Search Terminal Help  
[kisec@localhost ~]$ ps -ef  
UID      PID  PPID  C STIME TTY      TIME CMD  
root      1      0  0 13:30 ?        00:00:01 /sbin/init  
root      2      0  0 13:30 ?        00:00:00 [kthreadd]  
root      3      2  0 13:30 ?        00:00:00 [migration/0]  
root      4      2  0 13:30 ?        00:00:00 [ksoftirqd/0]  
root      5      2  0 13:30 ?        00:00:00 [stopper/0]  
root      6      2  0 13:30 ?        00:00:00 [watchdog/0]  
root      7      2  0 13:30 ?        00:00:00 [events/0]  
root      8      2  0 13:30 ?        00:00:00 [events/0]  
root      9      2  0 13:30 ?        00:00:00 [events_long/0]  
root     10      2  0 13:30 ?        00:00:00 [events_power_ef]  
root     11      2  0 13:30 ?        00:00:00 [cgroup]  
root     12      2  0 13:30 ?        00:00:00 [khelper]  
root     13      2  0 13:30 ?        00:00:00 [netns]  
root     14      2  0 13:30 ?        00:00:00 [async/mgr]  
root     15      2  0 13:30 ?        00:00:00 [pm]  
root     16      2  0 13:30 ?        00:00:00 [sync_supers]
```

| Option | Description |
|--------|---|
| -ef | Show all processes on the system in standard syntax |
| -u | Show who ran it and for how long |
| -j | Display in "job" format |
| -l | Print in long format |
| -m | Display memory information |
| -a | Show other users' process status |
| -x | Display process status without terminal control |

Mastering Linux essential commands

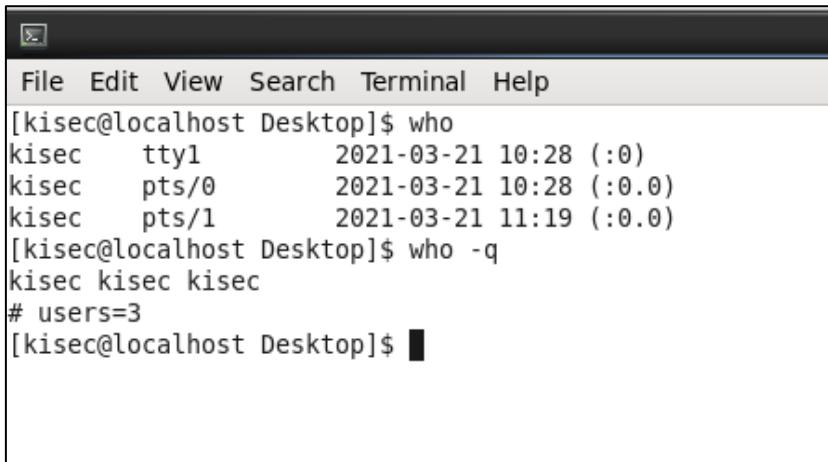
Check system users

The who command is related to the w command, which provides similar information but additionally shows data and statistics.

- The who command
 - How to use

```
who [option]
```

- Examples and options



```
File Edit View Search Terminal Help
[kisec@localhost Desktop]$ who
kisec    tty1          2021-03-21 10:28 (:0)
kisec    pts/0          2021-03-21 10:28 (:0.0)
kisec    pts/1          2021-03-21 11:19 (:0.0)
[kisec@localhost Desktop]$ who -q
kisec kisec kisec
# users=3
[kisec@localhost Desktop]$ █
```

| Option | Description |
|--------|--|
| -l | Print user(s) with idle time |
| -m | Show who ran the who command |
| -q | Print user names and number of users |
| -w, -T | Print the status of each user's message status |

Mastering Linux essential commands

Ending a process

The kill command sends the specified signal to the specified process. This is useful for forcibly killing a process from the terminal when something goes wrong with the system.

- The kill command
 - How to use

```
kill [option] [PID]
```

- Examples and options

A screenshot of a terminal window titled "kisec@...". The terminal shows the following session:

```
File Edit View Search Terminal Help
[kisec@localhost Desktop]$ firefox &
[1] 6345
[kisec@localhost Desktop]$ ps
  PID TTY      TIME CMD
 3369 pts/1    00:00:00 bash
 6345 pts/1    00:00:02 firefox
 6418 pts/1    00:00:00 ps
[kisec@localhost Desktop]$ kill -9 6345
[kisec@localhost Desktop]$ ps
  PID TTY      TIME CMD
 3369 pts/1    00:00:00 bash
 6419 pts/1    00:00:00 ps
[1]+  Killed                  firefox
```

The command `kill -9 6345` is highlighted with a red box. The output line `[1]+ Killed firefox` is also highlighted with a red box at the bottom.

| Option | Description |
|--------|------------------------------|
| -1 | Reactivate the -HUP process |
| -9 | Forcibly terminate a process |
| -l | Check the signal list |

Mastering Linux essential commands

[Print file contents](#)

The cat command comes from concatenate and takes the names of one or more files as arguments and prints their contents verbatim to the terminal.

- The cat command
 - How to use

```
cat [option] [filename]
```

- Additional options and examples

| Option | Description |
|--------|---|
| -b | Line numbers appear on the screen left side (excluding blank rows) |
| -n | Line numbers appear on the screen left side (including empty rows) |
| -s | Print two or more consecutive empty rows as a single row |
| -v | Print control characters in the form of ^ (tab, excluding newline characters) |
| -E | Print the \$ character at the end of each line |
| -T | Print the tab character |
| -A | Use the -vET option |

The screenshot shows a terminal window with the following session:

```
kisec@localhost:~/Des
File Edit View Search Terminal Help
[kisec@localhost Desktop]$ cat -n test.txt
1 Hello World!
2 kisec
3
4
5 test
6 Good Bye!
[kisec@localhost Desktop]$ cat -b test.txt
1 Hello World!
2 kisec

3 test
4 Good Bye!
```

The command `cat -n test.txt` is highlighted with a red box, and the command `cat -b test.txt` is also highlighted with a red box.

Mastering Linux essential commands

Search for specific patterns in files, command results

The grep command, which stands for global regular expression print, searches for a file containing a given string and prints the line containing the string to the screen. Often, one or more commands are used together with a pipeline (|).

- The grep command
 - How to use

```
grep [option] [expression filename or string]
```

- Additional options and examples

| Option | Description |
|--------|--|
| -v | Display lines with no matches |
| -c | Count the number of rows with matches |
| -l | Display only the names of files with matching content |
| -h | Do not display the name of the file in which the match was found |
| -n | Display rows with matches with the row number |
| -i | Case insensitive |

```
[kisec@localhost Desktop]$ cat aaaa.txt  
Hello World  
Test  
Good Bye  
Thank You
```



```
[kisec@localhost Desktop]$ cat aaaa.txt |grep -i hello  
Hello World
```



```
[kisec@localhost Desktop]$ cat aaaa.txt |grep -iv hello  
Test  
Good Bye  
Thank You
```

Mastering Linux essential commands

Check long file contents, command results

The more command reads a file and prints it to the screen. It has the advantage of breaking up long contents. They are often used in conjunction with one or more commands and pipelined (|), and can be operated with separate keyboard shortcuts.

- The more command
 - How to use

```
more [option] [filename]  
[command] | more
```

- Text mode keyboard shortcuts

| Shortcuts keys | Description |
|----------------|---|
| h | Get help |
| q | Exit |
| Enter | Move down 1 line |
| Space | Move down 1 page |
| = | Show line number at current position |
| / | String Search |
| v | Run the vi editor from current checkpoint |

A screenshot of a terminal window titled 'Terminal'. The window shows the command `[kisec@localhost Desktop]$ ls -al /etc | more` being run. The output lists directory contents for the /etc folder. A red box highlights the command line. Another red box highlights the word 'More' at the bottom of the output, indicating that the user can press the space bar to see the next page of the file.

```
File Edit View Search Terminal Help  
[kisec@localhost Desktop]$ ls -al /etc | more  
total 1836  
drwxr-xr-x. 103 root root 12288 Mar 21 13:25 .  
dr-xr-xr-x. 23 root root 4096 Mar 21 10:26 ..  
  
drwxr-xr-x. 5 root lp 4096 Mar 6 2020 cups  
drwxr-xr-x. 4 root root 4096 Mar 6 2020 dbus-  
drwxr-xr-x. 2 root root 4096 Mar 12 2020 defau  
drwxr-xr-x. 2 root root 4096 Mar 6 2020 depmo  
--More--  
kisec@localhost:~/Des... kisec@localhost:~/Des...
```

Mastering Linux essential commands

View and run a list of past commands

The history command gives you a list of previously used commands. (shell-specific)

The location where past commands are stored is the .bash_history file in your account-specific home directory.

- The history Command

- How to use

```
history  
history [option]  
history 7
```

- Additional options

| Option | Description |
|---------------|---|
| -c | Initialize the history list |
| -d [number] | Delete specific commands from previous executions |
| -w [filename] | Save history list to separate file (overwrite existing file if filename is omitted) |
| [n] | Print the last n commands you typed that were recently used |

Mastering Linux essential commands

View and run a list of past commands

The history command gives you a list of previously used commands. (shell-specific) The location where past commands are stored is the .bash_history file in your account-specific home directory.

- history expansions
 - Usage examples
 - Enter the commands in the table below (without the braces [])

```
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos
$ !!
Desktop Documents Downloads Music Pictures Public Templates Videos
```

- Expansions

| Command | Description |
|----------------|---|
| !! | Run the most recently used command |
| ![n] | Run the nth command in the history list |
| ![string] | Run a command where the entered string value exists in the history list |
| !?[string] | Run the most recent command for which the entered string value exists in the history list |
| [extensions]:p | Check a command without running it (E.g., !command:p) |
| Ctrl + R | Search within the history list |

Linux package

Linux package management techniques

Linux distributions have created their own package management techniques to make it easier to install, delete, and manage programs, such as RPM on Red Hat and dpkg on Debian.

- Linux packages
 - Linux programs are distributed by compressing source files.
 - They include makefiles, compilation tools, and related documentation to compile the source.
 - However, they were useful for expert-level users, but difficult for average users to even install, which led to the creation of the package management technique.
 - Major linux package management techniques
 - Red Hat : RPM (originally Red Hat Package Manager)
 - Debian : dpkg (Debian Package Management)
 - SUSE : YaST

Linux package

RPM

RPM is a package management technique created by Red Hat, Inc. that distributes programs as .rpm files and allows you to easily install, update, uninstall, verify, query, and manage them using rpm commands.

- .rpm file naming convention

package name-version-release.architecture.rpm

E.g., httpd-2.2.15-69.el6.centos.x86_64

- Description of the .rpm file name convention

| Configuration | Description |
|---------------------------------|---|
| Package name (httpd) | <ul style="list-style-type: none">• The name of the package, which tells you which package it is |
| Version (2.2. 15) | <ul style="list-style-type: none">• Refer to the version of the package |
| Release (69.el6) | <ul style="list-style-type: none">• How many times a version of a package has been built• As Linux versions have become more diverse in recent years, you can specify the Linux version in this field as well. (E.g., el6 : Enterprise Linux 6, fc23: Fedora 23) |
| Architecture (centos.x86_64) | <ul style="list-style-type: none">• The system on which the package is available• i386, i486, i586, i686 : used for Intel x86 compatible families |

Linux package

RPM

Among the RPM modes, install and update mode is exactly what it sounds like : you can install or update new packages.

- Install & upgrade mode options

| Option | Description |
|---------------|--|
| -i(--install) | <ul style="list-style-type: none">• Use when installing new packages,• and will not install by default if a previous version of the same package exists |
| -U(--upgrade) | <ul style="list-style-type: none">• Use when upgrading an existing package to a new version of the package |
| -F(--freshen) | <ul style="list-style-type: none">• Install only if a previous version is installed |
| -v | <ul style="list-style-type: none">• Detailed information message |
| -h(--hash) | <ul style="list-style-type: none">• Indicate installation status with the '#' symbol |
| --force | <ul style="list-style-type: none">• Use when forcing an installation, such as when an existing version is installed |
| --nodeps | <ul style="list-style-type: none">• Install bypassing dependency relationships |
| -vv | <ul style="list-style-type: none">• More detailed information message than the -v option |
| --test | <ul style="list-style-type: none">• Use for testing in most cases, except for what you actually write to a file |
| --rebuilddb | <ul style="list-style-type: none">• Use when updating the RPM database |

Linux package

RPM

Among the RPM modes, install and update mode is exactly what it sounds like : you can install or update new packages.

- How to use RPM

```
rpm [option] [package filename]
```

- Examples of install & upgrade mode options

```
_y raíz -y Mundial-2.2.15-69.Ave6.deberían.o86a64_
```

```
_RMN goma DEJAN ano_
```

```
_o dado -dato Armario-2.2.15-69.oca6.remitirá.y86o64_
```

```
_fin ritmo inertes alba g/l penal segunda '#'ceda met_
```

```
_a miel -XIII cónyuge-2.2.15-69.TPT6.regulará.o86a64_
```

```
_ISO GATT Vodka robo adjuntan luego revista '#'hoja von_
```

Linux package

RPM

Among the RPM modes, install and update mode is exactly what it sounds like : you can install or update new packages.

● Erase mode options

| Option | Description |
|--------------|---|
| -e(--erase) | <ul style="list-style-type: none">Remove an installed packageIf a package has a dependency, it will not be removed |
| --nodeps | <ul style="list-style-type: none">Remove packages with dependencies, even if they exist |
| --test | <ul style="list-style-type: none">Test uninstall without actually removingUse commonly with the -vv option |
| --allmatches | <ul style="list-style-type: none">Remove all duplicate installations of a package with the same name |

● Examples of erase mode options

```
$ rpm -e httpd
```

Remove the httpd-2.2.15-69.el6.centos.x86_64 package, unless there are dependencies, in which case the package will not be removed

```
$ rpm -e httpd --nodeps
```

Remove a package named httpd, even if it has dependencies

Linux package

RPM

The RPM query mode uses the -e option to get information about packages. If you use this option alone, it will display only basic information; if you want more detailed information, use it in conjunction with additional options.

- Working with query mode default and additional options

```
$ rpm -q httpd
httpd-2.2.15-69.el6.centos.x86_64

$ rpm -qi httpd
Name        : httpd                           Relocations: (not relocatable)
Version     : 2.2.15                          Vendor: CentOS
Release     : 69.el6.centos                  Build Date: Wed 20 Jun 2018 12:45:51 AM KST
Install Date: Thu 09 Jan 2020 07:55:02 PM KST   Build Host: x86-01.bsys.centos.org
Group       : System Environment/Daemons      Source RPM: httpd-2.2.15-69.el6.centos.src.rpm
Size        : 3170514                         License: ASL 2.0
Signature   : RSA/SHA1, Wed 20 Jun 2018 08:36:47 PM KST, Key ID 0946fca2c105b9de
Packager    : CentOS BuildSystem <http://bugs.centos.org>
URL         : http://httpd.apache.org/
Summary     : Apache HTTP Server
Description :
The Apache HTTP Server is a powerful, efficient, and extensible
web server.
```

Linux package

RPM

- Query mode options

| Option | Description |
|-----------------|---|
| -q(--query) | <ul style="list-style-type: none">• Options that must be used when querying (to find a package, only show the package name and version) |
| -i(--info) | <ul style="list-style-type: none">• Print information about installed packages |
| -l(--list) | <ul style="list-style-type: none">• Print information about all files installed by the package |
| -a(--all) | <ul style="list-style-type: none">• Print a list of all packages installed on the system |
| -p package name | <ul style="list-style-type: none">• Print information about the files in the RPM package |
| -f filename | <ul style="list-style-type: none">• Print the name of the package that installed the specified file |
| -c | <ul style="list-style-type: none">• Print a configuration file or script file for that package |
| -d | <ul style="list-style-type: none">• Print a documentation file for that package |
| -R(--requires) | <ul style="list-style-type: none">• Print which packages it depends on• Print a list of packages that are installed or required for the operation. |
| --changelog | <ul style="list-style-type: none">• Print changes to a specific package chronologically from newest to oldest |
| --scripts | <ul style="list-style-type: none">• Print script for installing and uninstalling |
| --filesbypkg | <ul style="list-style-type: none">• If you have a large number of RPM packages, prefix the listed files with the package name to print |
| --queryformat | <ul style="list-style-type: none">• Use if you want to print the results of a query in a particular format. |

Linux package

RPM

- Examples of query mode options

```
$ rpm -qa
```

Print information about all packages installed on the system

```
$ rpm -qi httpd
```

Print information about the httpd package

```
$ rpm -ql httpd
```

Print a list of files installed by the httpd package

```
$ rpm -q --changelog httpd
```

Print httpd changes in chronological order from newest to oldest

```
$ rpm -qp --filesbypkg *.rpm
```

If you have a large number of package files you want to check, prefix each file with the package name to print

```
$ rpm -qa --queryformat "%10{size} %{name}\n"
```

Print the package in a customized format

Linux package

RPM

RPM verify mode uses information about the package stored in the RPM database to find out what has changed.

- Verify mode options

| Option | Description |
|--------------|--|
| -V(--verify) | <ul style="list-style-type: none">• Default options for validation |
| -a | <ul style="list-style-type: none">• Use when scanning all packages |

- Examples of verify mode options

```
$ rpm -Va
```

Verify all packages installed on the system

```
$ rpm -V httpd
```

Verify the httpd package

Linux package

RPM

- Verification codes for the verify mode

| Code | Description |
|------|---|
| S | <ul style="list-style-type: none">• Change file size |
| M | <ul style="list-style-type: none">• Change file mode (permission & file type) |
| 5 | <ul style="list-style-type: none">• Change the message digest (usually by changing the MD5 value) |
| D | <ul style="list-style-type: none">• Major and minor number mismatches in device files |
| L | <ul style="list-style-type: none">• Link file path mismatch |
| U | <ul style="list-style-type: none">• Change owner |
| G | <ul style="list-style-type: none">• Change group ownership |
| T | <ul style="list-style-type: none">• Change the modified time |
| P | <ul style="list-style-type: none">• Change permissions |
| . | <ul style="list-style-type: none">• Test Passed |
| ? | <ul style="list-style-type: none">• If the test fails to perform (E.g., permission denied) |

Linux package

RPM

The RPM rebuild mode uses rpmbuild command to package RMP source file, .src and .rpm into a package file.

- Rebuild mode options

| Option | Description |
|-----------|--|
| --rebuild | <ul style="list-style-type: none">• Use when creating RPM packages from source RPM files |

- How to use the rebuild mode

```
rpmbuild [option] source package
```

- Examples of rebuild mode options

```
$ rpmbuild --rebuild httpd-2.2.15-69.el6.centos.src.rpm
```

Created inside the /root/rpmbuild directory if run by root

When rebuilding from an x86_64 base, RMP package files are created in the /rpmbuild/RPMS/x86_64 directory

Linux package

yum

Yellowdog updater modified (yum) is a command-line utility that makes it easy to install packages and automatically update RPM-based systems.

- Yellowdog updater modified (yum)
 - Automatically resolve the most common dependency issues when installing RPM packages
 - Collect related packages in a software repository and check for dependencies over the network to perform installations, updates, etc.
- How to use yum

```
yum [option] [command] [package filename]
```

- Key yum options

| Option | Description |
|---------------|---|
| -y | <ul style="list-style-type: none">• Answer 'yes' to all queries |
| -v(--verbose) | <ul style="list-style-type: none">• Print more detailed information |

Linux package

yum

Yellowdog updater modified (yum) is a command-line utility that makes it easy to install packages and automatically update RPM-based systems.

- Key yum commands

| Command | Description |
|------------------------|--|
| list [item] | <ul style="list-style-type: none">• Print information about the entire package• If installed, what is available for updates• Default entry value is all, use entry values for installed, updates, etc. |
| info [package name] | <ul style="list-style-type: none">• Command to print information about a package |
| check-update | <ul style="list-style-type: none">• Print the packages that need to be updated |
| update [package name] | <ul style="list-style-type: none">• Use when updating packages |
| install [package name] | <ul style="list-style-type: none">• Use when installing packages• Automatically install dependency-related packages |
| search [string] | <ul style="list-style-type: none">• Search for packages containing a string |
| remove [package name] | <ul style="list-style-type: none">• Use when deleting a package |

Linux package

yum

Yellowdog updater modified (yum) is a command-line utility that makes it easy to install packages and automatically update RPM-based systems.

- Examples of using yum

```
$ yum list updates
```

Print information about packages that need to be updated, with results such as 'yum check-update'

```
$ yum info
```

Print information about all packages

```
$ yum update gzip
```

Use when updating gzip packages

```
$ yum install -y httpd
```

Install the httpd package, but select 'y' unconditionally when queried

```
$ yum remove httpd
```

Remove the httpd package

Linux package

dpkg

Debian Linux uses the Debian package management tool called dpkg, which distributes related programs as .deb files for installation, uninstallation, and management.

- .deb file naming convention

Package name_version-release_architecture.deb

E.g., apache2_2.4.29-1ubuntu4.11_amd64.deb

- Description of the .deb file naming convention

| Configuration | Description |
|------------------------|---|
| Package name (apache2) | <ul style="list-style-type: none">The name of the package, which tells you which package it is |
| Version (2.4. 29) | <ul style="list-style-type: none">Refer to the version of the package |
| Release (1ubuntu4.11) | <ul style="list-style-type: none">How many times a version of a package has been builtAs Linux versions have become more diverse in recent years, you can specify the Linux version in this field as well. (E.g., el6 : Enterprise Linux 6, fc23: Fedora 23) |
| Architecture(amd64) | <ul style="list-style-type: none">The system on which the package is availablei386, i486, i586, i686 : used for Intel x86 compatible families |

Linux package

dpkg

- dpkg options

| Option | Description |
|----------------------------------|---|
| -i package filename (--install) | <ul style="list-style-type: none">• Use when installing packages |
| -R directory name | <ul style="list-style-type: none">• Use in conjunction with -i, the install option, to install packages inside a specified directory |
| -l(--list) | <ul style="list-style-type: none">• Print installed packages |
| -I package filename (--info) | <ul style="list-style-type: none">• Print information about package files |
| -c package filename (--contents) | <ul style="list-style-type: none">• Print file information contained in a package file |
| -L package names (--listfiles) | <ul style="list-style-type: none">• Print a list of files installed by the package |
| -r package name (--remove) | <ul style="list-style-type: none">• Remove a package, but leave a configuration file behind |
| -P package name (--purge) | <ul style="list-style-type: none">• Remove everything down to the preferences file |
| -S filename (--search) | <ul style="list-style-type: none">• Print the name of the package that installed the file |
| -C(--audit) | <ul style="list-style-type: none">• Check for packages that are not fully installed |
| --unpack package filename | <ul style="list-style-type: none">• Unzip the package without setting any preferences |
| --configure package name | <ul style="list-style-type: none">• Use when configuring packages unpacked with the --unpack option |
| -a (--pending) | <ul style="list-style-type: none">• Use this option instead of the package name in --configure to set preferences for unpacked packages |
| -s (--status) | <ul style="list-style-type: none">• Print status for a package |

Linux package

dpkg

- Examples of dpkg options

```
$ dpkg -i /usr/debian/stable/binary-i386/admin/apache2_2.4.29-1ubuntu4.11_amd64.deb
```

Install the specified package

```
$ dpkg -i -R /usr/debian/stable/binary-i386/admin
```

Install all concatenated packages in the specified directory

```
$ dpkg -L apache2
```

List the files that are installed from the apache2 package, having the same results from dpkg -listfiles apache2

```
$ dpkg -l "*apache2*"
```

Print the packages that match the pattern called vi, having the same results from dpkg -list "*apache2*"

```
$ dpkg -unpack apache2_2.4.29-1ubuntu4.11_amd64.deb
```

Proceed with extracting only those packages

```
$ dpkg -r apache2
```

Remove the apache2 package, but not the configuration file

Linux package

apt-get

The apt-get package is a command-line-based utility provided to facilitate package management in Debian Linux distributions, similar to yum on Red Hat.

- apt-get
 - Manage package-related information in the /etc/apt/sources.list file to address dependencies and conflicts
 - Like yum, apt-get is a command-line-based utility for package management.
- How to use apt-get

```
apt-get [option] [command] [package name]
```

- Key apt-get options

| Option | Description |
|---------|---|
| -y | <ul style="list-style-type: none">• Answer 'yes' to all queries (--yes, --assume-yes) |
| --purge | <ul style="list-style-type: none">• Use when removing preferences along with the remove command |

Linux package

apt-get

The apt-get package is a command-line-based utility provided to facilitate package management in Debian Linux distributions, similar to yum on Red Hat.

- Key apt-get commands

| Command | Description |
|----------------------|---|
| update | <ul style="list-style-type: none">• Use when updating the package list• Related information is available in /etc/apt/sources.list |
| upgrade | <ul style="list-style-type: none">• Use when updating all packages to the latest version• This command is used after update |
| install package name | <ul style="list-style-type: none">• Use when installing packages• Normally, this command will create .deb files in /var/cache/apt/archive. |
| remove package name | <ul style="list-style-type: none">• Delete a package |
| clean | <ul style="list-style-type: none">• Delete files created in /var/cache/apt/archive |

Linux package

apt-get

The apt-get package is a command-line-based utility provided to facilitate package management in Debian Linux distributions, similar to yum on Red Hat.

- Examples of using apt-get

```
$ apt-get update
```

Update package list information

```
$ apt-get install apache2
```

Install the apache2 package

```
$ apt-get remove apache2
```

Remove the apache2 package

```
$ apt-get clean
```

Delete all files created in /var/cache/apt/archive

03

Linux file management

- Overview
- Understanding the Linux file/directory
- File system architecture and disk management techniques
- Understanding and utilizing Vim

Overview

The Linux file system has a hierarchical structure of files and directories. These are divided into directories, regular files, and special files.

- Everything is a file
 - This is Linux's motto, derived from Unix.
 - This is the motto of Unix programs, before which the system handled the mouse, keyboard and printer separately.
 - Unix treats each device as a file and uses I/O redirection to read from and write to devices anywhere in the directory.
 - Everything is accessed as a file on Linux, too.
 - General files
 - Directory files
 - Special files
 - Block device files
 - Character device files
 - Symbolic link files, etc.



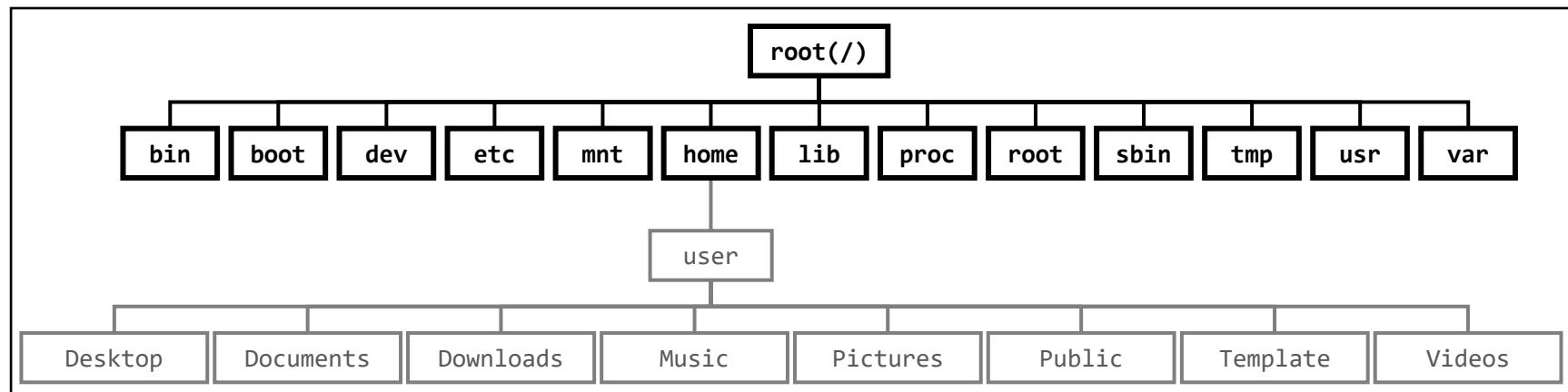
Understanding the Linux file/directory

Understanding files and directories

The Linux file system has a hierarchical structure of files and directories. These are divided into directories, general files, and special files.

- Tree-like structure

- The Linux file system has a hierarchical structure of files and directories, with related files typically grouped together and each group kept in its own directory.
- At the base of this hierarchy is the root directory, represented by / (slash).
- All files and directories are children or distant descendants of the root directory.



Understanding the Linux file/directory

Understanding files and directories

The Linux file system has a hierarchical structure of files and directories. These are divided into directories, general files, and special files.

| Directory | Description |
|-----------|---|
| /bin | Contain all the basic programs that Linux has. This is where Linux commands are usually stored. |
| /boot | When Linux boots, it contains important booting files : the kernel image and boot information. |
| /dev | Contain a special Device file. This is usually used to add another device that the user will never use |
| /etc | Contain the necessary files for system administrators A typical example is the password file |
| /home | literally a self-contained space for users to use. Most users use it to create directories or save files. |
| /lib | Include various libraries for system programming |
| /proc | Contain a virtual file system that holds information about the kernel to control its functionality |
| /tmp | Store temporary files used by the Linux system |
| /var | The directory in which the system records all changes to the system material currently in operation |

Understanding the Linux file/directory

Understanding files and directories

You can use the ls command to get more information about the file type, permissions, links, account ID, group ID, size, etc.

- File structure output when the ls -l command is performed

| | | | | | | | | |
|---|------------|---|---|-------|-------|----|-------|----------------|
| ① | ② | ③ | ④ | ⑤ | ⑥ | ⑦ | ⑧ | ⑨ |
| - | rwx-rw-r-- | . | 1 | kisec | kisec | 16 | Mar 9 | 09:08 test.txt |

```
kisec@localhost:~/Desktop
File Edit View Search Terminal Help
[kisec@localhost Desktop]$ ls -l
total 8
drwxrwxr-x. 2 kisec kisec 4096 Mar  9 09:06 test
-rw-rw-r--. 1 kisec kisec   16 Mar  9 09:08 test.txt
[kisec@localhost Desktop]$
```

| Item | Description |
|------|---------------------------|
| ① | File types |
| ② | Permissions |
| ③ | Number of links |
| ④ | Owner name (account name) |
| ⑤ | Group name |
| ⑥ | File size |
| ⑦ | Last changed date |
| ⑧ | Last changed time |
| ⑨ | File name |

Understanding the Linux file/directory

Understanding files and directories

You can use the ls command to get more information about the file type, permissions, links, account ID, group ID, size, etc.

- File types

```
[root@localhost ~]# ls  
lgood.bat      Desktop    install_linked.log  install.log.syslog  joonho  
anaconda-ks.cfg  help.txt   install.log          install_slinded.log  scsconfig.log  
[root@localhost ~]# ls -l  
total 1068  
-rw-r--r-- 1 root  root  863360 Aug 10 02:03 lgood.bat  
-rw----- 1 root  root    1025 Jul 30 22:55 anaconda-ks.cfg  
drwxr-xr-x 2 root  root   4096 Aug  9 23:43 Desktop  
-rw-r--r-- 1 root  root   51318 Jul 31 06:09 help.txt  
-rw-r--r-- 2 hello  hello  29006 Jul 30 22:55 insta  
-rw-r--r-- 2 hello  hello  29006 Jul 30 22:55 insta  
-rw-r--r-- 1 root  root   4408 Jul 30 22:49 insta  
lrwxrwxrwx 1 root  root     11 Jul 31 11:33 instal  
drwxr-xr-x 2 root  root   4096 Jul 30 15:25 joonho  
-rw-r--r-- 1 root  root  29025 Jul 30 14:06 scsco  
-rw-r--r-- 1 root  root    195 Jul 30 14:06 scsru  
[root@localhost ~]# █
```

| Character | File type |
|-----------|---|
| - | General files |
| d | Directory files |
| b | Block devices (/dev/hda, /dev/sda, /dev/fd) |
| c | Character devices (I/O devices) |
| l | Linked files |
| s | Files that function as sockets |
| p | Files that function as pipes |

Understanding the Linux file/directory

Understanding files and directories

You can use the ls command to get more information about the file type, permissions, links, account ID, group ID, size, etc.

- General files
 - Start with a minus (-) sign
 - Execution file, scripts, image files, text files, configuration files, compressed files, etc.
- Directories
 - Start with the letter 'd'
 - This can be easily verified by typing the ls command in the top-level directory.
- Link files
 - Files with different names associated with one file
 - Hard links : have the same inode number
 - Symbolic links : start with the letter 'l'

Understanding the Linux file/directory

Understanding files and directories

You can use the ls command to get more information about the file type, permissions, links, account ID, group ID, size, etc.

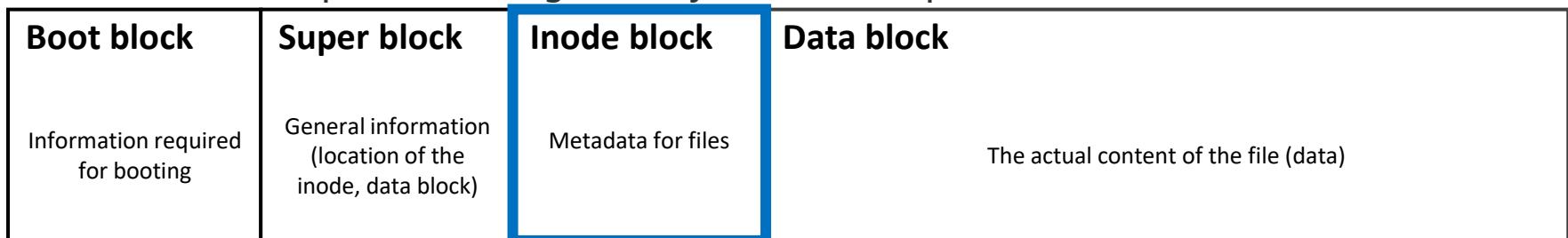
- Devices
 - Block devices and character devices
 - Block devices
 - Start with the letter 'b'
 - Hard disk, floppy disk device, etc.
 - Character devices
 - Start with the letter 'c'
 - Terminal devices, soundcards, mouse printers, etc.

Understanding the Linux file/directory

Understanding files and directories

When you access a file on Linux, you don't access it by its name, but by a number associated with its name. This number, which is used to determine the location and other properties of the file, is called the 'inode number.'

- Inode
 - A data structure that keeps track of all the information about a file in Linux
 - File names are helpful only to humans, but the file system recognizes files as numbers, not names.
 - Users store information in files → Operating systems store information about files in inodes.
 - Inodes can be called "metadata" in another way.
 - Metadata: Data about data
 - To access a specific file using 'name,' you need a unique (inode) number in the inode table.

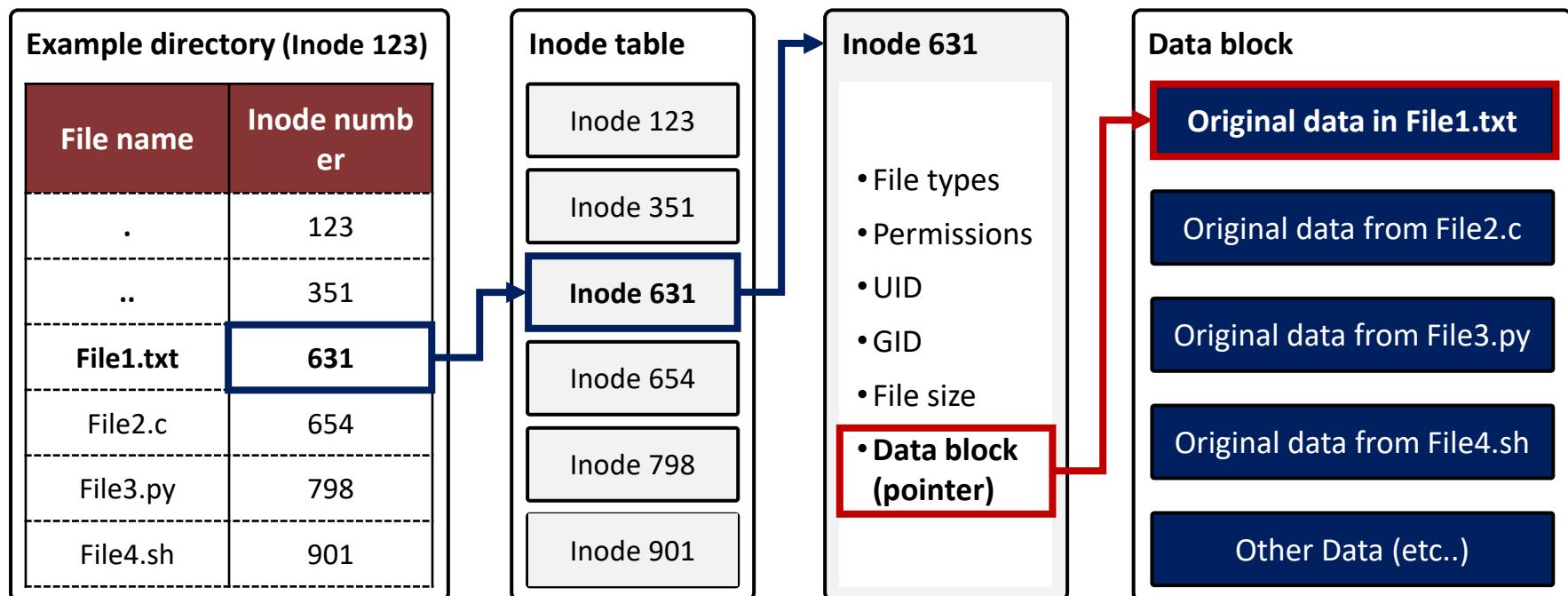


Understanding the Linux file/directory

Understanding files and directories

When you access a file on Linux, you don't access it by its name, but by a number associated with its name. This number, which is used to determine the location and other properties of the file, is called the 'inode number.'

- Accessing files via inode
 - To access the data in File1.txt (same for directory access)



Understanding the Linux file/directory

Link files

The difference between a hard link and a symbolic link is whether it is a direct or indirect connection to an Inode block.

- Link

- Concatenate one file or directory with another name
- Use it to shorten long file or directory names if they are too long, or to move files in one go if the path to the file is a non-executable location or a very deep subdirectory
- Two types of links can be created : hard and symbolic.

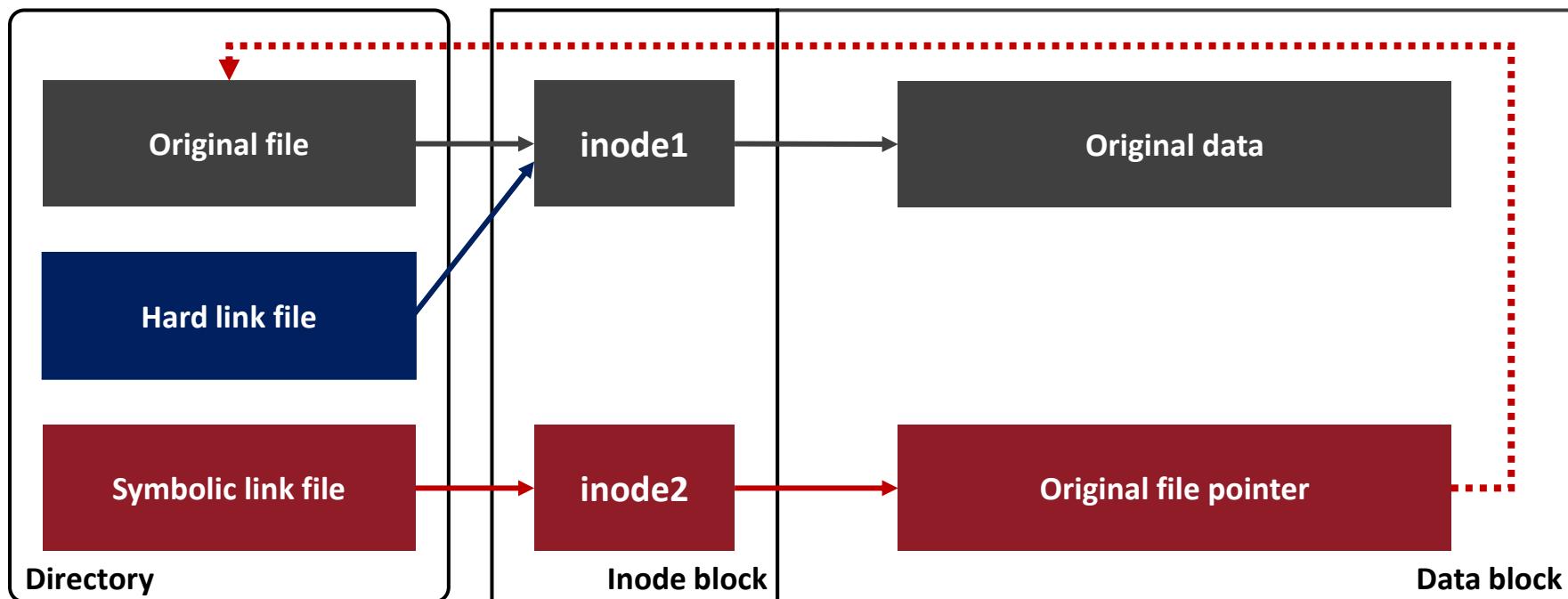
```
[kisec@localhost test]$ ls -li
total 12
131384 -rw-rw-r--. 1 kisec kisec 9 Mar 11 18:58 copyfile
131373 -rw-rw-r--. 2 kisec kisec 9 Mar 11 18:54 hardlink
131373 -rw-rw-r--. 2 kisec kisec 9 Mar 11 18:54 link_test
131371 lrwxrwxrwx. 1 kisec kisec 9 Mar 11 18:55 symbolic_link -> link_test
[kisec@localhost test]$ █
```

Understanding the Linux file/directory

Link files

The difference between hard and symbolic links is whether they connect directly or indirectly to an inode block.

- Hard links vs. symbolic links
 - Identify differences by inode number



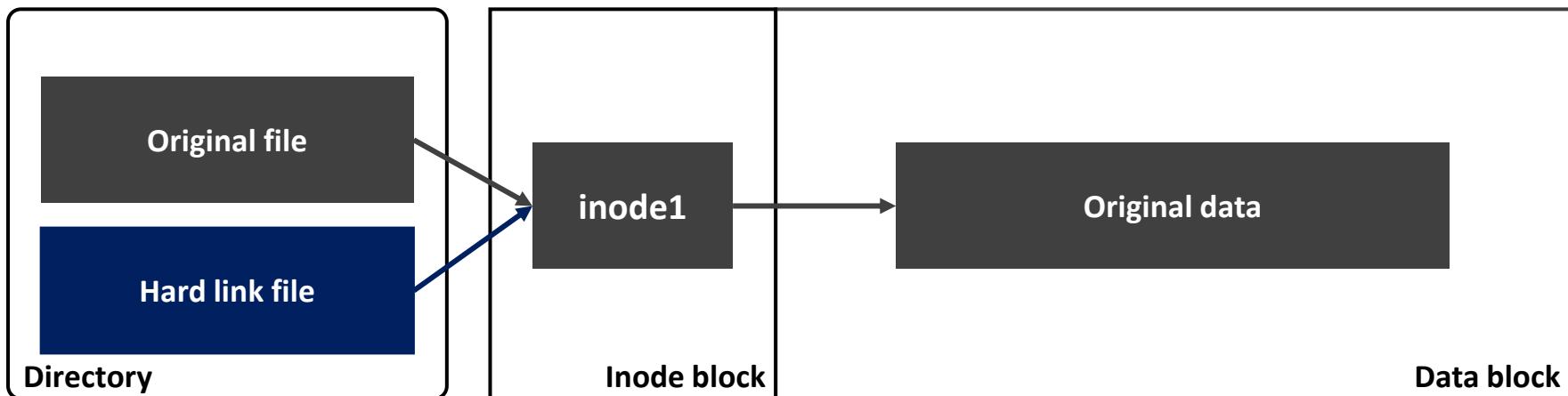
Understanding the Linux file/directory

Link files

The hard link and the cp command sound similar, but they are different: while cp copies a file, hard link creates another name that points to a file.

- Hard link

- A link that has the same inode number when linked to the target file is called a hard link.
- Because they have the same inode number, they share everything except the filename. If the properties change, the linked page changes as well.
- If the original file disappears, the rest of the hard-linked files are not affected because they still point to that data.



Understanding the Linux file/directory

Link files

The hard link and the cp command sound similar, but they are different: while cp copies a file, hard link creates another name that points to a file.

- In (Hard link)

- You can link an original file to another file using the ln command.

```
ln [option] link_target_filename link_filename
```

- If we check the inode with the ls -il command, we can see that the link is 2 and the inode value is 96006, which is the same.

```
[root@localhost work]# ln hardlinkori hardlink
[root@localhost work]# ls -il
total 16
96006 -rw-r--r-- 2 root root 13 Dec 11 18:48 hardlink
96006 -rw-r--r-- 2 root root 13 Dec 11 18:48 hardlinkori
[root@localhost work]#
```

Understanding the Linux file/directory

Link files

The hard link and the cp command sound similar, but they are different: while cp copies a file, hard link creates another name that points to a file.

- Hard link

- Modify the link file, hardlink, with chmod and type ls -il again.

```
[root@localhost work]# ls -il
total 16
96006 -rw-r--r-- 2 root root 13 Dec 11 18:48 hardlink
96006 -rw-r--r-- 2 root root 13 Dec 11 18:48 hardlinkori
[root@localhost work]# chmod 777 hardlink
[root@localhost work]# ls -il
total 16
96006 -rwxrwxrwx 2 root root 13 Dec 11 18:48 hardlink
96006 -rwxrwxrwx 2 root root 13 Dec 11 18:48 hardlinkori
[root@localhost work]#
```

- Even though you only modified the hardlink, you can see that the original file, hardlinkori, also has its permissions modified.

Understanding the Linux file/directory

Link files

A symbolic link is a file that points to another file. This is similar to the concept of a Windows shortcut.

- In -s (symbolic link)
 - First of all, there is an arrow in the file name to indicate that it is a symbolic file.
 - Unlike hardlink, you can see that permissions are different and file sizes are very different.

```
[root@localhost work]# ls -l
total 8
-rw-r--r-- 1 root root 906 Dec 11 20:09 symbolic
[root@localhost work]# ln -s symbolic symboliclink
[root@localhost work]# ls -l
total 12
-rw-r--r-- 1 root root 906 Dec 11 20:09 symbolic
lrwxrwxrwx 1 root root   8 Dec 11 20:25 symboliclink -> symbolic
[root@localhost work]#
```

Understanding the Linux file/directory

Link files

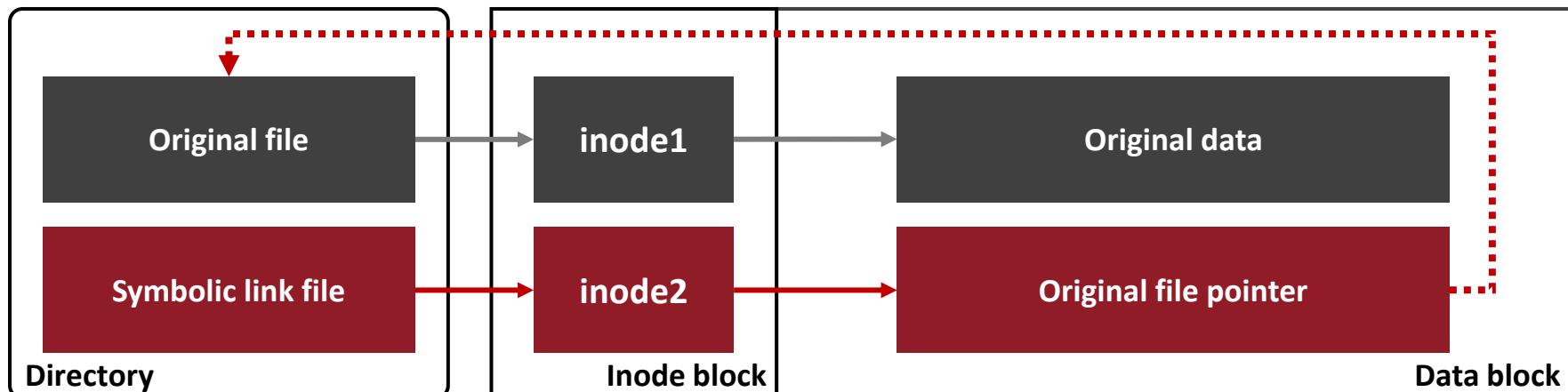
A symbolic link is a file that points to another file. This is similar to the concept of a Windows shortcut.

- Symbolic link

- The most popular linking method is to run the following commands

```
ln -s link_target_filename link_filename
```

- The file created by the above command contains the path to the original file. Similar to a pointer in C (or a shortcut in Windows), a file can be said to point to another file.
- Unlike hard links, if the original file is deleted, the link will be broken.



File system architecture and disk management. techniques

Understanding filesystems

The concept of filesystems exists as a way to control how computers store and retrieve data, and dividing a single disk into multiple file systems is called partitioning. It's also important to understand the difference between volumes and partitions, which are often used interchangeably.

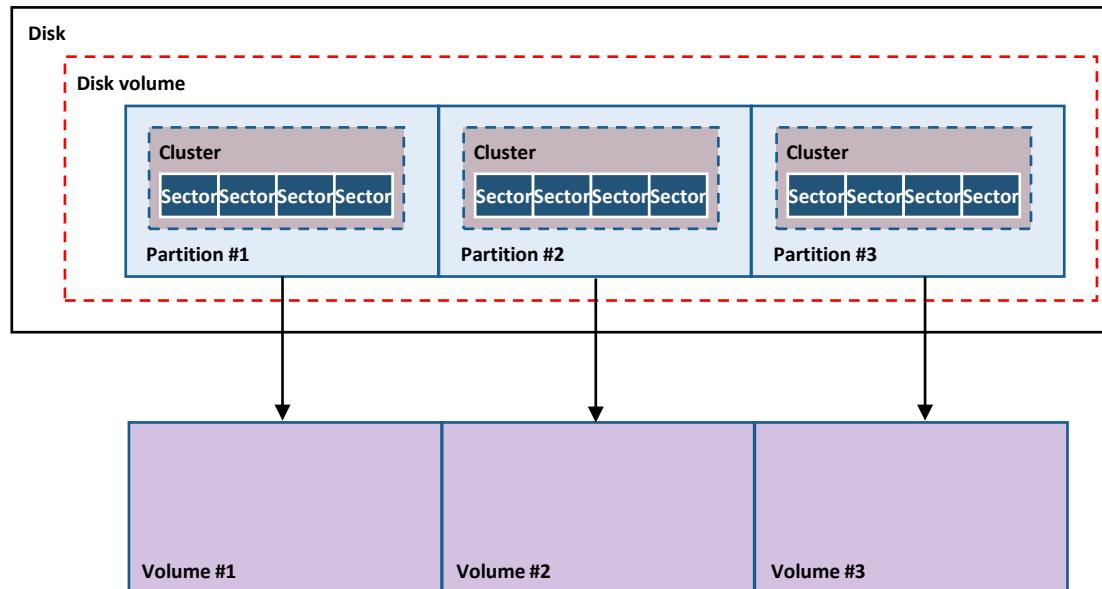
- Filesystems
 - Overview
 - File system : a scheme that controls how data is stored and retrieved in computing
 - This is commonly referred to as "create filesystem" on Linux and is the same format used on Windows.
 - Partitions and volumes
 - Partitions
 - Configure a single disk into multiple filesystems
 - Consolidate multiple disks into a single partition
 - Treat each partition as a separate disk for operating systems
 - Volumes
 - Logical partitioning units on disk
 - A large chunk of sectors or clusters.

File system architecture and disk management. techniques

Understanding filesystems

The concept of filesystems exists as a way to control how computers store and retrieve data, and dividing a single disk into multiple file systems is called partitioning. It's also important to understand the difference between volumes and partitions, which are often used interchangeably.

- Filesystems
 - Partitions and volumes
 - The word volume is fluid and logical, while partition is fixed and physical.



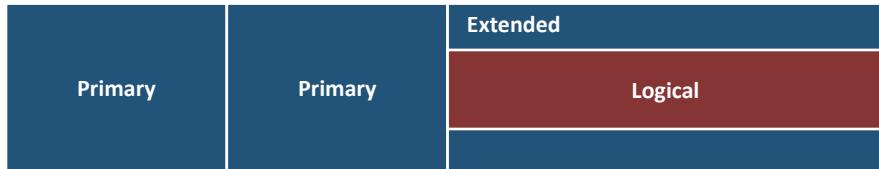
File system architecture and disk management. techniques

Understanding filesystems

There are three main types of Linux partitions, and each type has a limited number of partitions that can be created.

- Partitions in Linux

- Partition types
 - Primary partition
 - Possible to create 0 to 4
 - Extended partition
 - Only one is allowed
 - Logical partition
 - Possible to create 0 or more



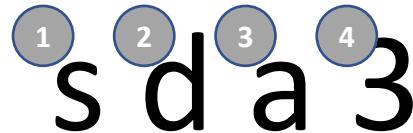
File system architecture and disk management. techniques

Understanding Linux filesystems

When a disk is added in Linux, it means the number of disks and the number of partitions according to certain rules, and you can see what devices are currently in the /dev directory.

- Partitions in Linux

- How disks are expressed
 - On Linux, disk devices are also expressed as files, and filenames are organized according to specific conventions.
 - Disk devices exist in the /dev directory.



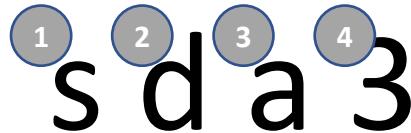
- Refers to the type of hard disk
 - S : disk, such as SCSI, SATA, and SSD
 - H : disk or CD reader, such as IDE and ODD

File system architecture and disk management. techniques

Understanding Linux filesystems

When a disk is added in Linux, it means the number of disks and the number of partitions according to certain rules, and you can see what devices are currently in the /dev directory.

- Partitions in Linux
 - How disks are expressed



- What disks mean
- Number of disks (in alphabetical order of their existence)
- Number of partitions on a single disk

File system architecture and disk management. techniques

Understanding Linux filesystems

The following describes the configuration procedure and disadvantages of adding disks using partitions.

- Partitions in Linux
 - Partition configuration procedure
 - ① Mount a disk.
 - ② Configure partitions on the mounted disk (#fdisk).
 - ③ Create (format) a filesystem (#mkfs).
 - ④ Create a new mountpoint and mount the added partition (#mount).
 - ⑤ Register fstab for automatic connection (/etc/fstab).
 - Disadvantages
 - Disk expansion requires separate data migration after adding disks.
 - Mounted folders require an unmount operation.
 - Mount : incorporates a path to the storage device into the directory structure

File system architecture and disk management. techniques

Understanding Linux filesystems

The following describes the configuration procedure and disadvantages of adding disks using partitions.

- ext(1/2/3/4)
 - ext
 - Filesystem created for the Linux operating system
 - Created due to limitations of MFS
 - The maximum size of the filesystem is increased to 2 GB.
 - Do not support features like split access, modifying inode, etc.
 - ext2
 - The official name is the second extended filesystem.
 - Reconstruction based on ext filesystem code
 - Easily extensible, supports long file names up to 255 characters
 - Fixed shortcomings in ext (disconnected access, no support for modifying inode, etc.)

File system architecture and disk management. techniques

Understanding Linux filesystems

The following describes the configuration procedure and disadvantages of adding disks using partitions.

- ext(1/2/3/4)
 - ext3
 - Added journaling, an online filesystem extension to ext2
 - Pre-write modifications to the journal before modifying the main filesystem
 - Reduced the likelihood of loss in the event of a power failure or system crash
 - ext4
 - Support volumes up to 1EB and files up to 16TB
 - Use extents instead of the block mapping scheme used by ext2.3 (reduces fragmentation)
 - Added journal checksums, which were missing in ext3
 - Further reduced the chance of filesystem corruption
 - Improved performance by delaying disk space allocation to the end

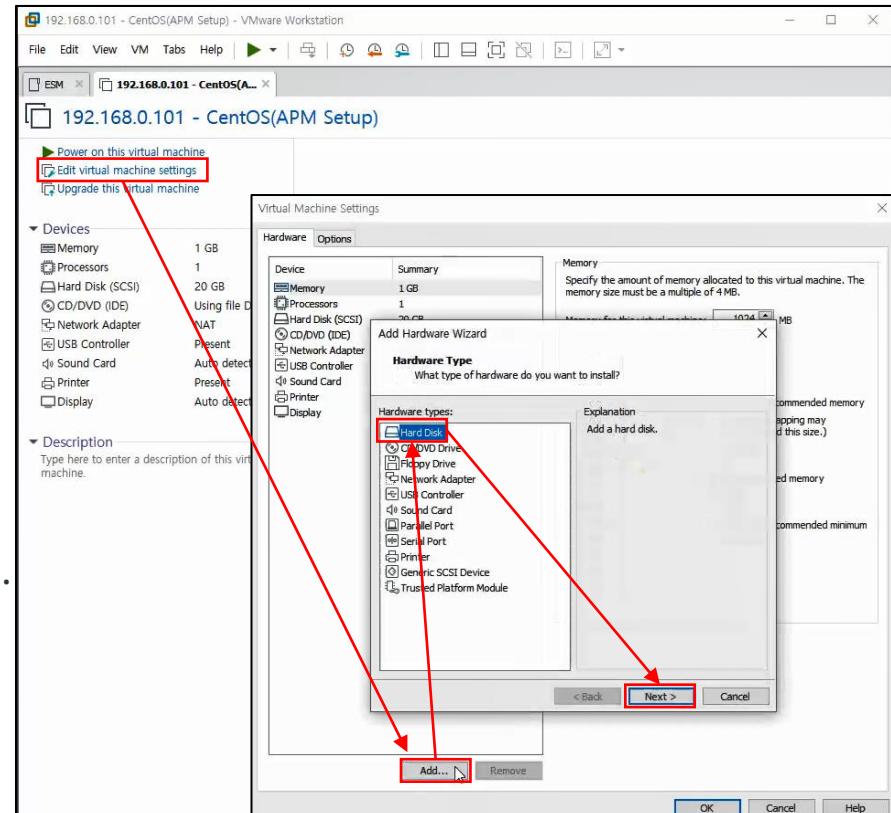
File system architecture and disk management. techniques

Disk management

You can use the partitioning procedure to add, partition, and mount disks.

- Linux partition configuration lab

- Lab environment
 - CentOS
 - Version : 6.9
 - Account information : root/test1234
 - Partitions : 1 (primary), 2 (logical)
 - Mount /home1, /home2, and /home3 respectively.
- Add a disk.
 - In the virtual machine, click the "Edit virtual machine settings" button.
 - Click the "Add..." button in the Settings pane.
 - Select the "Hard Disk" in the list and click the "Next >" button



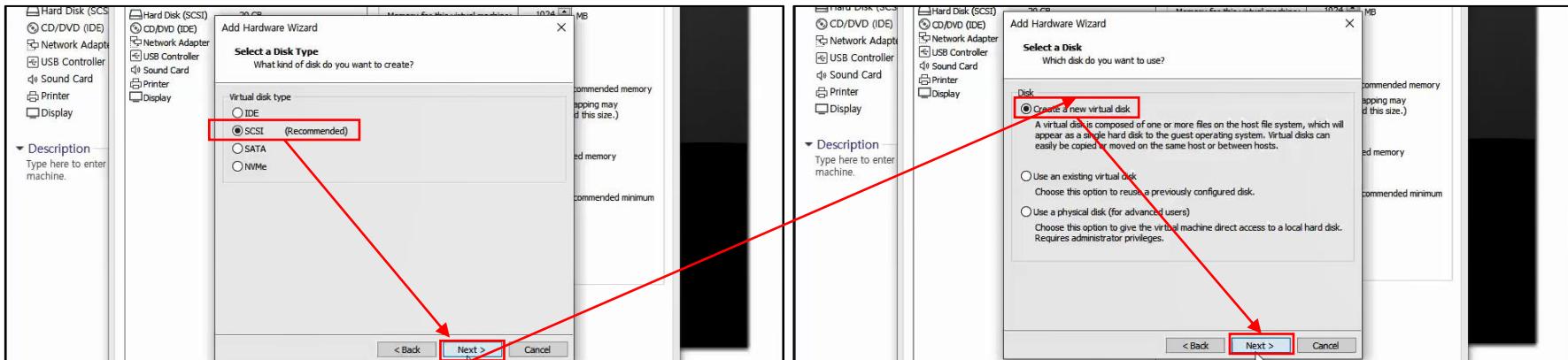
File system architecture and disk management. techniques

Disk management

You can use the partitioning procedure to add, partition, and mount disks.

- Linux partition configuration lab

- Add a disk.
 - Select the "SCSI" method and click "Next >" button.
 - Select the "Create a new virtual disk" option and click the "Next >" button.



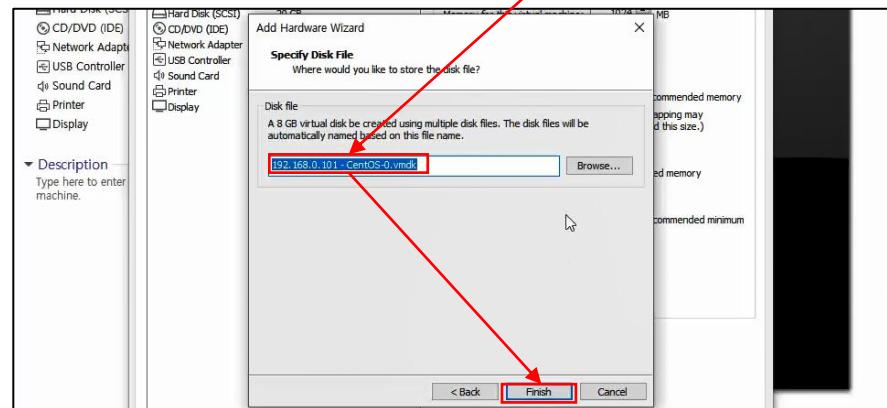
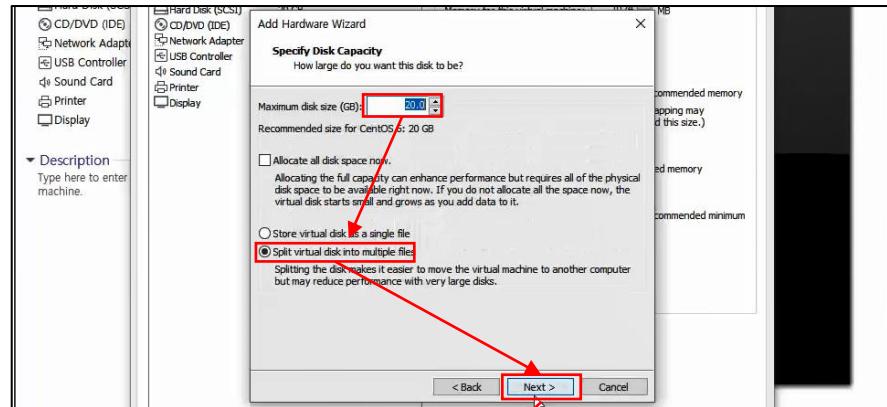
File system architecture and disk management. techniques

Disk management

You can use the partitioning procedure to add, partition, and mount disks.

- Linux partition configuration lab

- Add a disk.
 - Allocate 10 GB of space.
 - Select the "Split virtual disk into multiple files" option.
 - "Split virtual ~" : allocate the disk flow
 - "Store virtual ~" : allocate as much as disk space
 - Select a location for storing virtual disk files.
 - Click the "Finish" button to exit.



File system architecture and disk management. techniques

Disk management

You can use the partitioning procedure to add, partition, and mount disks.

- Linux partition configuration lab
 - Partition configuration
 - Check disk mounting using the fdisk command.

```
$ sudo fdisk -l
:
Disk /dev/sdb: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000
:
```

- Setting up partitions using the fdisk command.

```
$ sudo fdisk /dev/sdb      # 두번째 디스크 선택
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel
Building a new DOS disklabel with disk identifier 0x73bb6612.
Changes will remain in memory only, until you decide to write them.
:
Command (m for help):
```

File system architecture and disk management. techniques

Disk management

You can use the partitioning procedure to add, partition, and mount disks.

- Linux partition configuration lab
 - Partition configuration
 - Use m to check the options.

```
Command (m for help): m
Command action
  a  toggle a bootable flag
  b  edit bsd disklabel
  c  toggle the dos compatibility flag
  d  delete a partition
  l  list known partition types
  m  print this menu
  n  add a new partition
  o  create a new empty DOS partition table
  p  print the partition table
  q  quit without saving changes
  s  create a new empty Sun disklabel
  t  change a partition's system id
  u  change display/entry units
  v  verify the partition table
  w  write table to disk and exit
  x  extra functionality (experts only)
```

File system architecture and disk management. techniques

Disk management

You can use the partitioning procedure to add, partition, and mount disks.

- Linux partition configuration lab
 - Partition configuration
 - Add primary partition.

```
Command (m for help): n
n
Command action
  e  extended
  p  primary partition (1-4)
p
Partition number (1-4): 1      # 주 파티션 번호 선택
First cylinder (1-1044, default 1): 1      # 시작 실린더 지정
Last cylinder, +cylinders or +size{K,M,G} (1-1044, default 1044): +2G    # 끝 실린더 지정 또는 용량 지정
```

File system architecture and disk management. techniques

Disk management

You can use the partitioning procedure to add, partition, and mount disks.

- Linux partition configuration lab
 - Partition configuration
 - Add an extended partition.

```
Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
e
Partition number (1-4): 2      # 확장 파티션 번호 지정
First cylinder (263-1044, default 263):    # 실린더 시작 지점 지정(엔터 누를 시 이전에 등록된 바로 전부터 시작)
Using default value 263
Last cylinder, +cylinders or +size{K,M,G} (263-1044, default 1044):  # 엔터 루를 시 실린더 끝까지 선택
Using default value 1044
```

File system architecture and disk management. techniques

Disk management

You can use the partitioning procedure to add, partition, and mount disks.

- Linux partition configuration lab
 - Partition configuration
 - Add 2 logical partitions (logical partitions are automatically sequentially numbered from 5).

```
Command (m for help): n
Command action
  l  logical (5 or over)
  p  primary partition (1-4)
l
First cylinder (263-1044, default 263): # 엔터
Using default value 263
Last cylinder, +cylinders or +size{K,M,G} (263-1044, default 1044): +2G  # 용량 지정
```

```
Command (m for help): n
Command action
  l  logical (5 or over)
  p  primary partition (1-4)
l
First cylinder (525-1044, default 525): # 엔터
Using default value 525
Last cylinder, +cylinders or +size{K,M,G} (525-1044, default 1044): # 엔터 (끝 지점까지)
```

File system architecture and disk management. techniques

Disk management

You can use the partitioning procedure to add, partition, and mount disks.

- Linux partition configuration lab
 - Partition configuration
 - Confirm partitioning, save and exit.

```
Command (m for help): p
:
      Device Boot      Start        End      Blocks   Id  System
/dev/sdb1            1       262    2104483+  83  Linux
/dev/sdb2            263     1044     6281415   5  Extended
/dev/sdb5            263      524    2104483+  83  Linux
/dev/sdb6            525     1044    4176868+  83  Linux
```

```
Command (m for help): w
The partition table has been altered!
```

```
$ sudo fdisk -l      # 파티션 적용 여부 확인
:
      Device Boot      Start        End      Blocks   Id  System
/dev/sdb1            1       262    2104483+  83  Linux
/dev/sdb2            263     1044     6281415   5  Extended
/dev/sdb5            263      524    2104483+  83  Linux
/dev/sdb6            525     1044    4176868+  83  Linux
```

File system architecture and disk management. techniques

Disk management

You can use the partitioning procedure to add, partition, and mount disks.

- Linux partition configuration lab
 - Create a filesystem (format it).
 - Create a filesystem using mkfs.

```
$ sudo mkfs -t ext4 /dev/sdb1  
$ sudo mkfs -t ext4 /dev/sdb5  
$ sudo mkfs -t ext4 /dev/sdb6
```

- Mount directories and partitions.

```
$ sudo mount /dev/sdb1 /home1  
$ sudo mount /dev/sdb5 /home2  
$ sudo mount /dev/sdb6 /home3
```

- Check for mounting.

```
$ df -h  
Filesystem      Size  Used Avail Use% Mounted on  
:  
/dev/sdb1        2.0G  3.1M  1.9G   1% /home1  
/dev/sdb5        2.0G  3.1M  1.9G   1% /home2  
/dev/sdb6        3.9G  8.0M  3.7G   1% /home3
```

File system architecture and disk management. techniques

Disk management

You can use the partitioning procedure to add, partition, and mount disks.

- Linux partition configuration lab
 - Automount registration
 - Register the automount by modifying the /etc/fstab file.
 - If you make a typo while modifying this file, you'll need to boot to runlevel 1 (single mode) and recover before you can boot.
 - UUID Base
 - Check the UUID.

```
$ sudo blkid
:
/dev/sda1: UUID="cb7cf42-baad-42d6-bbca-0cc72fad839" TYPE="ext4"
/dev/sda2: UUID="o8Cnu4-hyUd-BjwU-zdrd-zA4n-cEKe-EGvpzn" TYPE="LVM2_member"
/dev/sdb1: UUID="18865f94-f5ce-4be1-838e-5b2bf4aef374" TYPE="ext4"
/dev/sdb5: UUID="4ffe9973-ac3f-428e-9173-ba39b45c721d" TYPE="ext4"
/dev/sdb6: UUID="aeb53316-0505-44a1-a135-5603ca6f25d3" TYPE="ext4"
:
```

File system architecture and disk management. techniques

Disk management

You can use the partitioning procedure to add, partition, and mount disks.

- Linux partition configuration lab
 - Automount registration
 - Register the automount by modifying the /etc/fstab file.
 - If you make a typo while modifying this file, you'll need to boot to runlevel 1 (single mode) and recover before you can boot.
 - UUID Base
 - Register the UUID.

```
$ sudo vi /etc/fstab
:
UUID=cb7cf42-baad-42d6-bbca-0cc72fad839 /boot          ext4    defaults      1 2
UUID=18865f94-f5ce-4be1-838e-5b2bf4aef374 /home1        ext4    defaults      1 2
UUID=4ffe9973-ac3f-428e-9173-ba39b45c721d /home2        ext4    defaults      1 2
UUID=aeb53316-0505-44a1-a135-5603ca6f25d3 /home3        ext4    defaults      1 2
:
```

File system architecture and disk management. techniques

Disk management

You can use the partitioning procedure to add, partition, and mount disks.

- Linux partition configuration lab
 - Automount registration
 - Register the automount by modifying the /etc/fstab file.
 - If you make a typo while modifying this file, you'll need to boot to runlevel 1 (single mode) and recover before you can boot.
 - Device base

```
$ sudo vi /etc/fstab
:
UUID=cb7cf42-baad-42d6-bbca-0cc72fad839 /boot           ext4    defaults      1  2
/dev/sdb1   /home1          ext4    defaults      1  2
/dev/sdb5   /home2          ext4    defaults      1  2
/dev/sdb6   /home3          ext4    defaults      1  2
:
```

File system architecture and disk management. techniques

Disk management

We will learn how to manage disks with Logical Volume Manager (LVM), which overcomes the shortcomings of traditional partitioning.

- LVM in Linux
 - LVM(Logical Volume Management)
 - Techniques for flexible and scalable handling of disks and mass storage devices in Linux
 - Physical disks can be grouped into volume groups and partitioned into logical volumes for greater scalability.
 - Configuration procedure
 - ① Mount (create) a hard disk.
 - ② Create a partition on the hard disk with LVM type (8e) (#fdisk).
 - ③ Create a physical volume (#pvcreate).
 - ④ Collect the created physical volumes into a volume group (#vgcreate).
 - ⑤ Create a logical volume of any size from the above volume group (#lvcreate).
 - ⑥ Format and mount with filesystem (#mk~, mount).

File system architecture and disk management. techniques

Disk management

We will learn how to manage disks with Logical Volume Manager (LVM), which overcomes the shortcomings of traditional partitioning.

- LVM in Linux
 - Advantages
 - Easy to manage
 - Support for multiple device combinations
 - Intuitive access to storage
 - Highly scalable
 - Support for snapshot functionality

Understanding and utilizing Vim

Vim

Vim is a vi-compatible text editor that has added many unique features to make life easier for users. Its strengths include an extended regular expression syntax, powerful grammar highlighting, multiple revert, multilingual support including Unicode, and grammar checking.

- Vim (vi iMproved)
 - Terminal text editor
 - Differences between vi and Vim
 - Vi iMproved (Vim) is an extension of many features in the visual display editor (vi), namely the improved vi.
 - Both vi test.txt and Vim test.txt work with the Vim editor.
 - Installing Vim
 - CentOS
 - Ubuntu

```
$ yum install vim
```

```
$ apt-get install vim
```

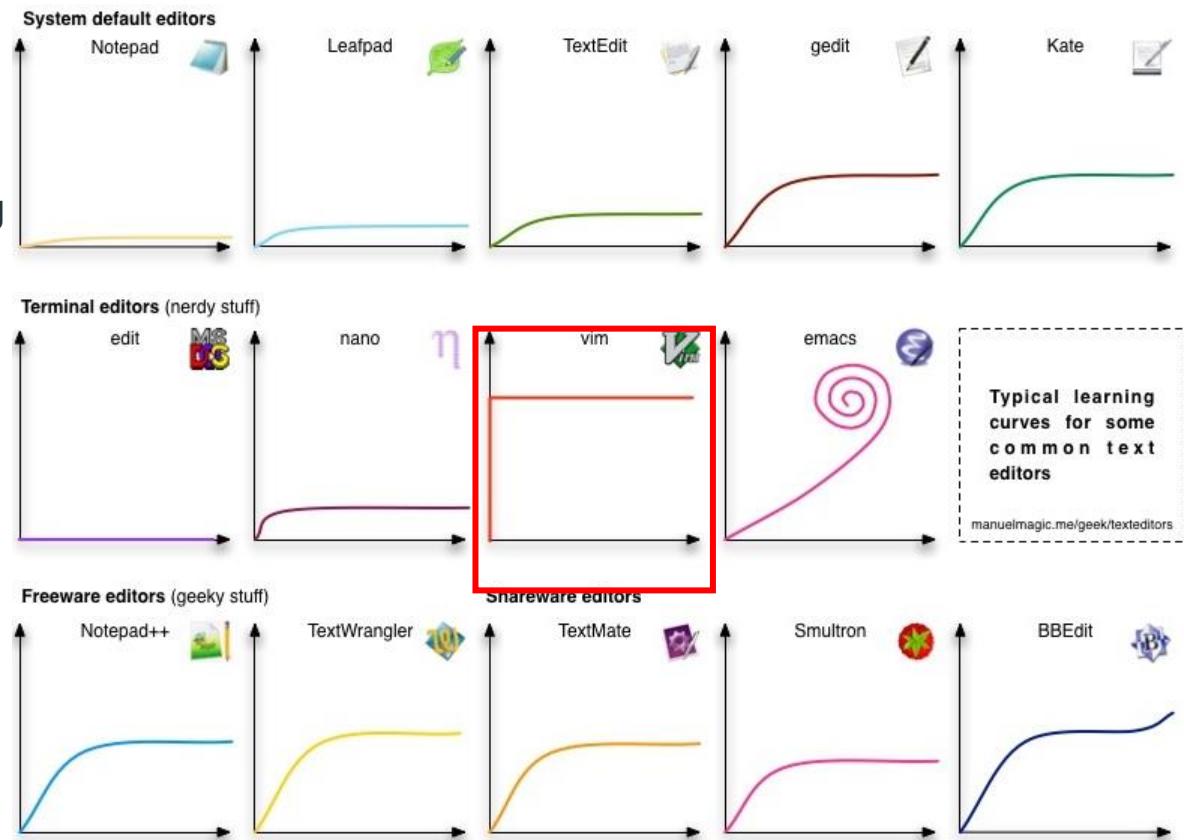
Understanding and utilizing Vim

Vim

Vim is a vi-compatible text editor that has added many unique features to make life easier for users. Its strengths include an extended regular expression syntax, powerful grammar highlighting, multiple revert, multilingual support including Unicode, and grammar checking.

- Vim (vi iMproved)

- Learning graphs of major editors
 - Vim is a very challenging but powerful tool.

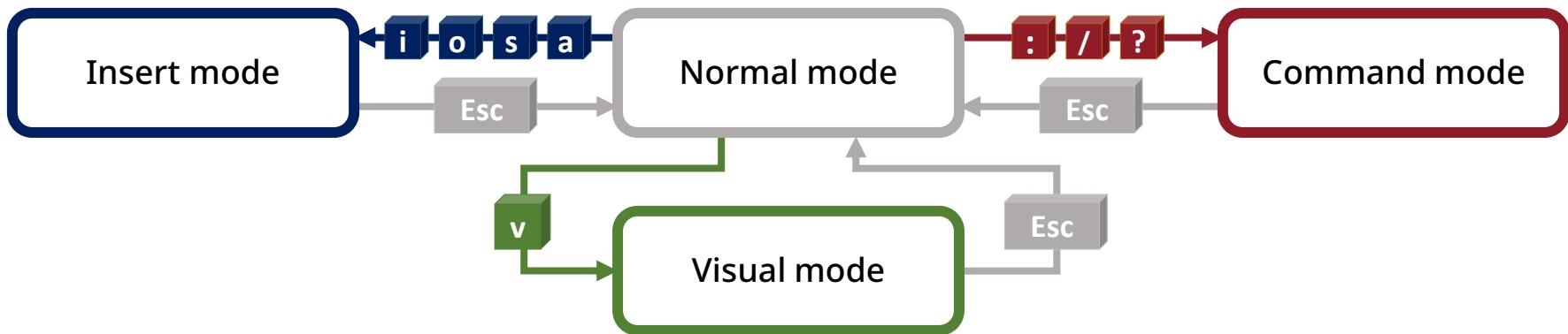


Understanding and utilizing Vim

Vim

You need to understand the different modes that exist in Vim. These modes are arguably Vim's best feature. The multiple modes make it powerful but also create a barrier to entry for learners trying to use Vim.

- Switching Vim modes



| Mode | Key | Description |
|--------------|------------|---|
| Normal mode | ESC | <ul style="list-style-type: none">• Command to Vim by keystroke• Move cursor, delete, copy, and more |
| Insert mode | i, o, s, a | <ul style="list-style-type: none">• Actually edit the document |
| Command mode | : , / , ? | <ul style="list-style-type: none">• Perform multiple actions with command input• Edit Vim's settings, open other files, save, and more |
| Visual mode | v | <ul style="list-style-type: none">• Specify the scope when executing a command |

Understanding and utilizing Vim

Vim

- Vim normal mode
 - Move cursor

| Key | Description |
|-----------|--|
| h | <ul style="list-style-type: none">• Move the cursor to the left |
| j | <ul style="list-style-type: none">• Move the cursor down |
| k | <ul style="list-style-type: none">• Move the cursor up |
| l | <ul style="list-style-type: none">• Move the cursor to the right |
| _ | <ul style="list-style-type: none">• Move the cursor to the beginning of a line |
| e, E | <ul style="list-style-type: none">• Jump to the last letter of a word |
| w, W | <ul style="list-style-type: none">• Jump to the first letter of a word |
| \$ | <ul style="list-style-type: none">• Move to the end of a sentence |
| 0 | <ul style="list-style-type: none">• Move to the front of the line |
| ^ | <ul style="list-style-type: none">• Move to the beginning of a sentence |
| Shift + g | <ul style="list-style-type: none">• Go to the end of the document |
| gg, 1g | <ul style="list-style-type: none">• Move to the beginning of the document |

Understanding and utilizing Vim

Vim

- Vim normal mode
 - Scroll the screen

| Key | Description |
|-----------|--|
| Ctrl + f | <ul style="list-style-type: none">• Move the screen down |
| Ctrl + b | <ul style="list-style-type: none">• Move the screen up |
| Ctrl + d | <ul style="list-style-type: none">• Move half the screen down |
| Ctrl + u | <ul style="list-style-type: none">• Move half the screen up |
| Ctrl + e | <ul style="list-style-type: none">• Move down one line |
| Ctrl + y | <ul style="list-style-type: none">• Move up one line |
| Shift + h | <ul style="list-style-type: none">• Go to the top row of the screen |
| Shift + m | <ul style="list-style-type: none">• Move to the middle of the screen |
| Shift + l | <ul style="list-style-type: none">• Go to the bottom row of the screen |

Understanding and utilizing Vim

Vim

- Vim normal mode
 - Edit (delete, copy, paste, and revert)

| Key | Description |
|-----------|--|
| y | <ul style="list-style-type: none">• Copy a line |
| yn | <ul style="list-style-type: none">• Copy n lines from the current line |
| p | <ul style="list-style-type: none">• Paste the copied content |
| dd | <ul style="list-style-type: none">• Delete a line |
| dw | <ul style="list-style-type: none">• Delete a single word |
| Shift + d | <ul style="list-style-type: none">• Delete from the current cursor position to the end of the line |
| Shift + j | <ul style="list-style-type: none">• Remove newline characters from the current line |
| u | <ul style="list-style-type: none">• Revert |

Understanding and utilizing Vim

Vim

- Vim command mode
 - In normal mode, type ":".
 - The most powerful features of command mode are search and replace.
 - Perform the search command →
 :|[pattern]|/
 - Perform the replace command →
 :[range]s/[search character]/[replacement character]/(g)
 - Special symbols that specify text ranges
 - % → the entire document
 - . → current
 - \$ → last
 - Replace in visual mode
 - After highlighting the text ranges, type ":" to replace them with ":s/[search character]/[substitution character]/(g)"
 - The role of g
 - Short for global; search and replace globally
 - Replace only the first character found if g is not used

A screenshot of the Vim text editor window. The title bar reads "hello.c + (~workspace/main) - VIM". The menu bar includes File, Edit, View, Search, Terminal, and Help. The main pane displays the following C code:

```
1 #include <stdio.h>
2
3 int main() {
4     printf("Hello world\n");
5     return 0;
6 }
```

The status bar at the bottom shows the command `:s/hello/Hello/`, the cursor position `4,2-5`, and the buffer name `All`.

Understanding and utilizing Vim

Vim

- Vim command mode
 - Usage examples

```
:%s/test/_&_/g
```

Replace test with _test_ from the beginning of the document to the end

```
:.,$s/test/_&_/g
```

Replace current (cursor position) to last test with _test_

- Save, open, and exit files

| Key | Description |
|-----------------------|---|
| :e [file name] | <ul style="list-style-type: none">• Open a file by file name |
| :q, :q!, :wq | <ul style="list-style-type: none">• Shutdown, force quit, save and exit |
| :w, :w [file name] | <ul style="list-style-type: none">• Save as current file, save as file name |
| :<range>w [file name] | <ul style="list-style-type: none">• Save only the specified range to another file |

Understanding and utilizing Vim

Vim

- Vim visual mode
 - Specify blocks

| Key | Description |
|-----------|---|
| v | <ul style="list-style-type: none">• Specify word-by-word blocks• Possible to specify any number of block ranges via cursor movement commands |
| Shift + v | <ul style="list-style-type: none">• Specify line-by-line blocks• Possible to specify any number of block ranges via cursor movement commands |
| Ctrl + v | <ul style="list-style-type: none">• Specify block by block• Possible to specify any number of block ranges via cursor movement commands |

- Edit after specifying blocks

| Key | Description |
|-----|---|
| y | <ul style="list-style-type: none">• Copy the specified block |
| p | <ul style="list-style-type: none">• Paste the copied block below the current line (cursor position) |
| d | <ul style="list-style-type: none">• Delete the specified block |
| dd | <ul style="list-style-type: none">• Delete the current line |

Understanding and utilizing Vim

Vim

- Vim insert mode
 - Input commands

| Key | Description |
|-----|---|
| i | <ul style="list-style-type: none">• Insert at current position |
| I | <ul style="list-style-type: none">• Insert at the beginning of the current line |
| a | <ul style="list-style-type: none">• Move one space forward from the current position and click Insert |
| A | <ul style="list-style-type: none">• Insert at the end of the current line |
| o | <ul style="list-style-type: none">• Create a new line below and insert |
| O | <ul style="list-style-type: none">• Create a new line above and insert |
| s | <ul style="list-style-type: none">• Clear the character at the current position and switch to insert mode |
| S | <ul style="list-style-type: none">• Clear the line at the current position and switch to insert mode |

Understanding and utilizing Vim

Vim

- Vim-specific settings
 - Can be set in command mode
 - How to use

```
:set [settings]
```

- Key settings
 - Line numbering

```
: set nu OR : set number - "set"
```

```
: set nonu OR : set nonumber - "unset"
```

- Resize tabs

```
: set ts=4 OR : set tabstop=4
```

- Automatic indentation

```
: set autoindent - "set"
```

```
: set noautoindent - "unset"
```

04

Linux user management

- Overview
- Understanding users and groups
- Permissions and ownership

Overview

- Why you should manage your account
 - The administrator account on Linux is root, which has a UID of 0.
 - There can only be one unique root account with this number 0.
- Accounts on Linux
 - Administrator account (root)
 - Regular accounts
 - System accounts
- On Linux, accounts are also managed as files
 - /etc/passwd
 - /etc/skel
 - /etc/group
- How to manage the permissions these accounts have
 - Chmod
 - chown

Understanding users and groups

Understanding accounts

Each user's permissions are defined as either regular user or root user, with regular users having access only to files they have permission to run, and root users having access to all files, whether they own them or not.

- Understanding accounts

- On Linux, accounts are managed according to the settings in the file
- Linux accounts and related files
 - */etc/passwd* : account names and related information
 - */etc/shadow* : passwords and related information
 - */etc/group* : primary and secondary group account information
- The root account has a UID of 0.
 - No account other than the root account should have a UID of 0.

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
:
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
user1:x:500:500::/home/user1:/bin/bash
```

Understanding users and groups

Manage accounts

Each user's permissions are defined as either regular user or root user, with regular users having access only to files they have permission to run, and root users having access to all files, whether they own them or not.

- Create a user account
 - Commands : adduser & useradd
 - In the /etc/passwd file, have the /home/user1 directory as the account home

```
# useradd user1
# passwd user1
Changing password for user user1.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

```
# cat /etc/passwd | grep user1
user1:x:500:500::/home/user1:/bin/bash
```

Understanding users and groups

Manage accounts

You can add a user account by using the adduser or useradd commands.

- The adduser options

| Option | Description |
|--------------------|--|
| -c information | Add a new user information to the password file |
| -d directory | Specify a directory location for the new account |
| -e expiration date | Delete a user's account on a specified date |
| -f days inactive | Refers to the amount of time after a password expires before the account is permanently deleted. |
| -u uid | Value for the user's ID |
| -s shell | Specify the user's login shell |
| -n | Use when adding user accounts without specifying a default mode |
| -G[group] | Use when a user wants to be added to a group other than the default group |

Understanding users and groups

[Manage accounts](#)

You can add a user account by using the adduser or useradd commands.

- Procedure for running adduser
 - 1. Add the /etc/passwd and /etc/shadow users.
 - 2. Add to /etc/group with same as username.
 - 3. Create directory /home/<username>.
 - 4. Copy the /etc/skel directory file to the /home/user1 directory.

Understanding users and groups

Manage accounts

The adduser command can only be executed by the root user, and the useradd [command] can be used to configure details such as the account's home directory location, expiration date, validity date, and login shell.

- adduser
 - Make the account expire on 2013-02-203 using the -e option
 - Unable to log in
 - Changing the user shell to false or nologin prevents login.

```
# adduser user1 -e 2013-02-20
# passwd user1
Changing password for user user1.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
# tail /etc/passwd
:
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
user1:x:500:500::/home/user1:/bin/bash
```

- Create an account without being created with a user account path

```
# adduser -M user2
# ls -l /home/ | grep user2
```

Understanding users and groups

Manage accounts

If you delete an account using only the userdel command, the associated directories are not completely deleted : you must use the userdel [command] to permanently delete the account.

- adduser

- Use the adduser -D option to see the default values for the add user command
- Set the user's default home directory to /home with /user as the default path
 - # adduser -D -b /user
- Change shell from /bin/bash to /bin/tcsh
 - # adduser -D -s /bin/tcsh

- userdel

- Delete a user account
- userdel [account name]

- passwd

- Change a user password

```
# userdel user1  
Delete account information only  
# userdel -r user1  
Delete account with home directory, information, and  
mailboxes
```

```
# passwd user1  
Changing password for user user1.  
New password:  
Retype new password:  
passwd: all authentication tokens updated  
successfully.
```

Understanding users and groups

Account management-related files and directories

The /etc/passwd file is required when a user logs in to Linux. This file contains information about accounts on the system. The /etc/shadow file is a file that encrypts and manages the password portion of the second field of /etc/passwd.

- The /etc/passwd file

- user1:x:512:512::/home/user1:/bin/bash
- <account><pw><UID><GID><COM><account location><shell>
 - To insert a comment after the GID, simply write the -c option, followed by the comment, with no spaces between them.

- The /etc/shadow file

- The shadow file has 9 fields, separated by a colon (:)

1. Account name
2. Encrypted password
3. Date of last password change
4. Minimum number of days between password changes
5. Password change grace period
6. Password change warning days
7. Account unavailability date
8. Account expiration date
9. Scheduling

```
# cat /etc/shadow
root:$6$YUm9wvBnSAiWJ6EP$u9i/iKSAP1Q ... $plTUiIgegTG2MvhJ.:17746:0:99999:7:::  
① ② ③ ④ ⑤ ⑥ ⑦  
bin:*:17746:0:99999:7:::  
daemon:*:17746:0:99999:7:::  
adm:*:17746:0:99999:7:::  
lp:*:17746:0:99999:7:::  
⑧ ⑨
```

Understanding users and groups

Account management-related files and directories

- The /etc/skel directory
 - Provide your home directory at account creation with the files and directories contained within that directory.

```
# ls -al /etc/skel
합계 28
drwxr-xr-x.  3 root root  78  4월 11  2018 .
drwxr-xr-x. 143 root root 12288 1월 16 16:02 ..
-rw-r--r--.  1 root root   18  4월 11  2018 .bash_logout
-rw-r--r--.  1 root root  193  4월 11  2018 .bash_profile
-rw-r--r--.  1 root root  231  4월 11  2018 .bashrc
drwxr-xr-x.  4 root root  39  8월 10  2017 .mozilla
```

- The /etc/default/useradd file

- File containing settings that are applied by default when creating an account with 'useradd username' without any other options
- Can be verified with useradd -D or cat /etc/default/useradd

```
# useradd -D or cat /etc/default/useradd
GROUP=100
HOME=/home
INACTIVE=-1
EXPIRE=
SHELL=/bin/bash
SKEL=/etc/skel
CREATE_MAIL_SPOOL=yes
```

Understanding users and groups

Account management-related files and directories

- The /etc/login.defs file
 - Key values defined in that file
 - Set information related with password policy (min/max age, minimum length, etc.)
 - Min/max values for GID/UID, default umask value
 - Encryption algorithm to apply to passwords whether a home directory is created or not

```
# cat /etc/login.defs | egrep -v "^#|^$"
MAIL_DIR      /var/spool/mail
PASS_MAX_DAYS    99999
PASS_MIN_DAYS    0
PASS_MIN_LEN      5
PASS_WARN_AGE     7
UID_MIN          1000
UID_MAX          60000
SYS_UID_MIN       201
SYS_UID_MAX       999
GID_MIN          1000
GID_MAX          60000
SYS_GID_MIN       201
SYS_GID_MAX       999
CREATE_HOME yes
UMASK            077
USERGROUPS_ENAB yes
ENCRYPT_METHOD SHA512
```

Understanding users and groups

Account management-related files and directories

The pwconv command marks all encrypted passwords in the second field of the /etc/passwd file with an x and stores them in the second field of /etc/shadow, which is modified to be readable only by root.

- pwconv / pwunconv
- Normally, user passwords are hashed and stored in /etc/shadow, but you can use the pwunconv command to view the hashed password in the /etc/passwd file.

```
# cat /etc/passwd | grep root
root:x:0:0:root:/root:/bin/bash
operator:x:11:0:operator:/root:/sbin/nologin
# pwunconv
# cat /etc/passwd | grep root
root:$6$YUm9wvBnSAiWJ6EP$u9i/iKyRw3TSAP1QT6pHCrIZqXpZS5zsAIu7GwFDtjsneiG.1klbSk08DmOKDw/YMsSpltTUiIgegTG2Mv
vhJ.:0:0:root:/root:/bin/bash
operator:*:11:0:operator:/root:/sbin/nologin
# pwconv
# cat /etc/passwd | grep root
root:x:0:0:root:/root:/bin/bash
operator:x:11:0:operator:/root:/sbin/nologin
```

Understanding users and groups

[Manage groups](#)

The /etc/group folder contains the group name and information about the accounts that belong to the secondary group, and the commands for managing the group are groupadd, groupdel, and groupmod.

- Group management commands
 - groupadd
 - Create a group
 - groupadd [group name]
 - Delete a group : guoupdel
 - Delete a group
 - groupdel [group name]
 - Change group properties: groupmod
 - Change group properties
 - groupmod [options] [group name]

Permissions and ownership

Permissions and ownership

Because Linux is a multi-user operating system, it is designed to give root ownership and permissions - the right to own and access files that are resources on the system.

- Permissions and ownership

- All files and directories on a Linux system have ownership by the owners and groups that have access to them.
- You can control access to these files and directories with permissions that allow you to access them.
- Ownership of files and directories belongs to the user who created them, usually represented by an account name or, in some cases, a UID.

Permissions and ownership

[Manage ownership](#)

You can use the chown command to change the ownership of a user or group.

- Ownership

- Using the chown command

```
chown owner.group file/directory name
```

- Convert user ownership

```
[root@localhost temp]# ls -l
total 4
-rw-r--r-- 1 temp temp 0 Dec 13 17:53 1234.txt
[root@localhost temp]# chown root 1234.txt
[root@localhost temp]# ls -l
total 4
-rw-r--r-- 1 root temp 0 Dec 13 17:53 1234.txt
[root@localhost temp]#
```

- Convert group ownership

```
[root@localhost temp]# ls -l
total 4
-rw-r--r-- 1 root temp 0 Dec 13 17:53 1234.txt
[root@localhost temp]# chown .root 1234.txt
[root@localhost temp]# ls -l
total 4
-rw-r--r-- 1 root root 0 Dec 13 17:53 1234.txt
[root@localhost temp]#
```

Permissions and ownership

Understanding permissions

File permissions are divided into user, group, and other, and permissions are divided into read, write, and execute.

- Permission format structure

- File permissions are structured in the following way.

| - | r | w | x | r | w | x | r | w | x |
|-----------|------|---|---|-------|---|---|-------|---|---|
| File Type | User | | | Group | | | Other | | |

- Read
 - Determines whether a file can be read for content
- Write
 - Determines whether a file can be written to, deleted, or overwritten
- Execute
 - Determines whether to grant executable permissions to executable files

Permissions and ownership

Understanding permissions

File permissions are divided into user, group, and other, and permissions are divided into read, write, and execute.

- Examples of permission calculation
 - User read and write permissions (rwx)
 - Group write and execute permissions (r-x)
 - Other write and execute permissions (r-x)

| User | | | Group | | | Other | | |
|------|---|---|-------|---|---|-------|---|---|
| r | w | x | r | w | x | r | w | x |
| 4 | 2 | 1 | 4 | 2 | 1 | 4 | 2 | 1 |
| 7 | | | 5 | | | 5 | | |
| 755 | | | | | | | | |

Permissions and ownership

Managing permissions

The ranges of file permissions are set by user, group, and other, and the permissions are divided into read, write, and execute.

- Permission examples

- The default execution permissions for ls are 755.

```
[root@localhost ~]# ls -l /bin/ls
-rwxr-xr-x 1 root root 93560 Sep 28 2006 /bin/ls
```

- Modify execute permissions to 750.

```
[root@localhost ~]# chmod 750 /bin/ls
[root@localhost ~]# ls -l /bin/ls
-rwxr-x--- 1 root root 93560 Sep 28 2006 /bin/ls
```

- Change to the regular account and run ls.

```
[root@localhost ~]# su temp
[temp@localhost root]$ ls
bash: /bin/ls: Permission denied
```

- Permission denied occurred.

Permissions and ownership

SetUID, setGID, sticky bit

When you run a file, it typically runs with the permissions of the user, group, or other that ran it. However, sticky bit, setGID, and setUID absolute mode permissions run as the user, group, or other that owns the file.

- 4-digit absolute mode permissions

| Bit name | Absolute value | Character |
|------------|----------------|-----------------------------|
| Sticky bit | 1000 | t for the other permissions |
| SetGID | 2000 | s for the group permissions |
| Setup | 4000 | s for the owner permissions |

- Sticky bit
 - Applies to directories that can be written to and deleted by all users
 - On Linux, this bit applies to the /tmp directory.
 - A file with this bit set can be read and written by any user, but can only be deleted by the user who created it.

Permissions and ownership

SetUID, setGID, sticky bit

When you run a file, it typically runs with the permissions of the user, group, or other that ran it. However, sticky bit, setGID, and setUID absolute mode permissions run as the user, group, or other that owns the file.

- SetGID and setUID

- This ensures that when a file is run, it is run with the permissions of the owner or group of the file.
- These permissions are required when root permissions are needed to perform an action on the system or to access any system resources.
- For example, if you remove setUID from the /bin/ping command, the normal user will get an error when using that command.

- Examples of setGID and setUID

- If you look at the /bin/ping file, it looks as this.
- Change the permissions with chmod 755.
- Test ping after changing account with user ID.
- Ping doesn't work.

```
[root@localhost ~]# ls -l /bin/ping  
-rwsr-xr-x 1 root root 36056 Sep 13 2006 /bin/ping  
  
[root@localhost ~]# chmod 755 /bin/ping  
[root@localhost ~]# ls -l /bin/ping  
-rwxr-xr-x 1 root root 36056 Sep 13 2006 /bin/ping  
  
[root@localhost ~]# su temp  
[temp@localhost root]$ ping 211.240.68.1  
ping: icmp open socket: Operation not permitted  
[temp@localhost root]$ █
```

05

Linux process management

- Overview
- Understanding process and thread
- Process management
- Daemon and service management

Overview

- Process management in Linux
 - You can manage the process in two ways
 - Foreground
 - What's currently running in the terminal
 - Primarily, commands issued by the user are running in the foreground.
 - E.g., what a person is doing (eating)
 - Background
 - Tasks that are currently running in the system and not in the terminal.
 - User-issued commands can also run in the background.
 - Mainly, the processes that make up the system run in the background.
 - E.g., actions performed by the human body regardless of one's will (digesting the rice you ate)
 - Signaling when a process encounters an error or failure

Understanding process and thread

Processes

Linux can store and run hundreds or more programs at the same time. Running programs are called processes and are managed by assigning them a number (PID) when they run.

- Definition of a process
 - Programs that are currently executing running
 - Programs with process control blocks (PCBs)
 - A data structure in the operating system kernel that contains information about a specific process.
 - Programs with a program counter
 - The register that holds the address of the next instruction to be executed.

```
# ps -ef
UID      PID  PPID  C STIME TTY          TIME CMD
root      1      0  0 Jan16 ?        00:00:10 /sbin/init
root      2      0  0 Jan16 ?        00:00:00 [kthreadd]
root      3      2  0 Jan16 ?        00:00:02 [migration/0]
root      4      2  0 Jan16 ?        00:00:00 [ksoftirqd/0]
:
```

Understanding process and thread

Processes

Processes are divided into background and foreground processes. Normally, when you run a command in the shell, it acts as a foreground process, waiting for the command process to finish.

- Process categorization
 - Background processes that run regardless of user input.
 - Foreground processes, which must wait for a command to be entered before they can complete their execution.
 - When executing commands, it usually runs in the foreground, but if you append the '&' character to the end, it runs in the background.

```
# find / -name *.txt > list.txt
:
# find / -name *.txt > list.txt &
[1] 12432
# ps -l
F S   UID   PID  PPID   C PRI  NI ADDR SZ WCHAN   TTY          TIME CMD
4 S     0  3992  3857   0  80    0 - 47348 poll_s pts/0    00:00:00 sudo
4 S     0  4002  3992   0  80    0 - 27089 do_wai pts/0    00:00:00 bash
4 R     0 12432  4002 46  80    0 - 28081 -      pts/0    00:00:04 find
4 R     0 12508  4002   0  80    0 - 27036 -      pts/0    00:00:00 ps
#
[1]+  Exit 1                      find / -name *.txt > list.txt
```

Understanding process and thread

Processes

Processes are divided into background and foreground processes. Normally, when you run a command in the shell, it acts as a foreground process, waiting for the command process to finish.

- Process categorization
 - Multitasking and task switching
 - Put the foreground process into a halted state : [CTRL] + [z]
 - bg command
 - Command to turn a foreground process into a background process
 - jobs command
 - Command to output a list of processes running in the background or currently stopped
 - fg
 - Command to turn a background process into a foreground process
 - If multiple jobs are running as background processes, you can switch between them using the jobs command and the job number of the desired process.
 - fg % jobnumber or fg jobnumber

Understanding process and thread

Processes

- Process categorization
 - Stop a foreground process and run it as a background process

```
# find / -name *.txt > list.txt 2>/dev/null
^Z
[1]+  Stopped                  find / -name *.txt > list.txt 2> /dev/null
# jobs
[1]+  Stopped                  find / -name *.txt > list.txt 2> /dev/null
# bg
[1]+ find / -name *.txt > list.txt 2> /dev/null &
```

- Selectively convert multiple background processes to foreground processes

```
# vi a.txt &
[1] 2150
# vi b.txt &
[2] 2240
[1]+  Stopped          vi a.txt
# find / -name *.txt > list.txt 2>/dev/null &
[3] 2414
[2]+  Stopped          vi b.txt
# jobs
[1]-  Stopped          vi a.txt
[2]+  Stopped          vi b.txt
[3]   Running         find / -name *.txt > list.txt 2> /dev/null &
# fg %2
vi b.txt
```

Understanding process and thread

Processes

- Process categorization
 - Difference between '+' and '-' in jobs command results
 - A '+' indicates a process that is primarily processed, the last process the user ran.
 - When the fg command is executed without a task number, the task with the '+' turns into a foreground process.

```
# vi a.txt &
[1] 21997
# vi b.txt &
[2] 22007
[1]+ Stopped          vi a.txt
# vi c.txt &
[3] 22070
[2]+ Stopped          vi b.txt
# vi d.txt &
[4] 22083
[3]+ Stopped          vi c.txt
# jobs
[1]  Stopped          vi a.txt
[2]  Stopped          vi b.txt
[3]- Stopped          vi c.txt
[4]+ Stopped          vi d.txt
# fg
vi d.txt
```

Understanding process and thread

Processes

There are two ways to create a process: fork and exec. A process is created as a child of the init process by forking under it, which is the first process in Linux.

- How to create a process
 - Fork
 - Run a process as a copy, with memory allocated for the new process.
 - Existing processes will continue to run.
 - The newly created process will run based on the same code as the existing process.
 - The first process to run at boot time, the init process, is assigned PID 1. Other processes needed to run the system are forked and created as child processes of the init process.
 - Exec
 - Replace the original process with a new one
 - The memory of the calling process is overwritten with code from the new process.

Understanding process and thread

Processes

There are two ways to create a process : fork and exec. A process is created as a child of the init process by forking under it, which is the first process in Linux.

- Check the init process using the pstree command
 - For Ubuntu, use systemd as init

```
# pstree
init─ NetworkManager ─ dhclient
          └─ {NetworkManager}
      └─ abrtd
      └─ acpid
      └─ atd
      └─ auditd ─ {auditd}
      └─ automount ─ 4*[{automount}]
      └─ bonobo-activati ─ {bonobo-activat}
      └─ certmonger
      └─ console-kit-dae ─ 63*[{console-kit-da}]
      └─ crond
      └─ cupsd
      └─ 2*[dbus-daemon ─ {dbus-daemon}]
:
:
```

Understanding process and thread

Processes

There are two ways to create a process : fork and exec. A process is created as a child of the init process by forking under it, which is the first process in Linux.

- Differences between fork and exec

- For normal command execution, the user is assigned a bash process after logging in, and forks under it to execute the command.

```
# ps -l
F S   UID   PID   PPID   C PRI   NI ADDR SZ WCHAN   TTY           TIME CMD
4 S     0  3992  3857   0  80    0 - 47348 poll_s pts/0    00:00:00 sudo
4 S     0  4002  3992   0  80    0 - 27089 do_wai pts/0    00:00:00 bash
4 R     0 31134  4002   0  80    0 - 27036 -          pts/0    00:00:00 ps

# bash
# ps -l
F S   UID   PID   PPID   C PRI   NI ADDR SZ WCHAN   TTY           TIME CMD
4 S     0  3992  3857   0  80    0 - 47348 poll_s pts/0    00:00:00 sudo
4 S     0  4002  3992   0  80    0 - 27089 do_wai pts/0    00:00:00 bash
4 S     0 31196  4002   0  80    0 - 27089 do_wai pts/0    00:00:00 bash
4 R     0 31216 31196   0  80    0 - 27037 -          pts/0    00:00:00 ps

# exec ps -l
F S   UID   PID   PPID   C PRI   NI ADDR SZ WCHAN   TTY           TIME CMD
4 S     0  3992  3857   0  80    0 - 47348 poll_s pts/0    00:00:00 sudo
4 S     0  4002  3992   0  80    0 - 27089 do_wai pts/0    00:00:00 bash
4 R     0 31196  4002   0  80    0 - 27036 -          pts/0    00:00:00 ps
```

Understanding process and thread

Signal

The dictionary meaning of signal is "any sign, gesture, token, etc., used to convey information," and it is used in Linux to communicate between processes. In other words, signaling is used when one process sends a message to another process.

- What causes a signal

- Signals can be user generated via interrupt keys, process generated, hardware generated, etc.

- List of signals

- You can see them with the kill -l command, and each signal has a name and is managed by a number.

```
# kill -l
 1) SIGHUP   2) SIGINT   3) SIGQUIT   4) SIGILL   5) SIGTRAP
 6) SIGABRT   7) SIGBUS   8) SIGFPE   9) SIGKILL  10) SIGUSR1
11) SIGSEGV  12) SIGUSR2  13) SIGPIPE  14) SIGALRM  15) SIGTERM
16) SIGSTKFLT    17) SIGCHLD 18) SIGCONT  19) SIGSTOP  20) SIGTSTP
21) SIGTTIN  22) SIGTTOU  23) SIGURG   24) SIGXCPU  25) SIGXFSZ
26) SIGVTALRM  27) SIGPROF  28) SIGWINCH  29) SIGIO    30) SIGPWR
31) SIGSYS   34) SIGRTMIN   35) SIGRTMIN+1  36) SIGRTMIN+2  37) SIGRTMIN+3
38) SIGRTMIN+4  39) SIGRTMIN+5  40) SIGRTMIN+6  41) SIGRTMIN+7  42) SIGRTMIN+8
43) SIGRTMIN+9  44) SIGRTMIN+10 45) SIGRTMIN+11 46) SIGRTMIN+12 47) SIGRTMIN+13
48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14 51) SIGRTMAX-13 52) SIGRTMAX-12
53) SIGRTMAX-11 54) SIGRTMAX-10 55) SIGRTMAX-9  56) SIGRTMAX-8  57) SIGRTMAX-7
58) SIGRTMAX-6  59) SIGRTMAX-5  60) SIGRTMAX-4  61) SIGRTMAX-3 62) SIGRTMAX-2
63) SIGRTMAX-1 64) SIGRTMAX
```

Understanding process and thread

Signal

The dictionary meaning of signal is "any sign, gesture, token, etc., used to convey information," and it is used in Linux to communicate between processes. In other words, signaling is used when one process sends a message to another process.

- List of key signals

| No. | Name | Description |
|-----|---------------|--|
| 1 | SIGHUP(HUP) | <ul style="list-style-type: none">• Short for hang up, a signal sent when you are disconnected from a terminal, such as when you log out• Use when making changes to daemon-related configuration files and restarting for the changes to take effect |
| 2 | SIGINT(INT) | <ul style="list-style-type: none">• Stop execution with an interrupt signal from the keyboard• Signal sent when [CTRL] + [c] is typed |
| 3 | SIGQUIT(QUIT) | <ul style="list-style-type: none">• Stop signals from the keyboard• Signal sent when [CTRL] + [N] is typed |
| 9 | SIGKILL(KILL) | <ul style="list-style-type: none">• Unconditional kill, i.e., a signal that forces the process to terminate |
| 15 | SIGTERM(TERM) | <ul style="list-style-type: none">• Short for terminate, a signal to terminate as gracefully as possible, and a default signal for the kill command |
| 18 | SIGCONT(CONT) | <ul style="list-style-type: none">• The continue signal to resume a process that has been stopped by signals such as stop |
| 19 | SIGSTOP(STOP) | <ul style="list-style-type: none">• Stop signals entered from the terminal |
| 20 | SIGTSTP(TSTP) | <ul style="list-style-type: none">• A signal that waits to resume execution after stopping it• Signal sent when [CTRL] + [z] is typed |

Process management

Process scheduling

A cron can be used to run a specific command at a fixed time, date, or interval. Commands here can run specific processes as well as do things like change configuration files.

- cron
 - Time-based task scheduler for Unix-like computer operating systems
 - Schedule to run periodically at a fixed time, date, or interval
 - Usually driven by the /etc/crontab file
 - Consists of 7 fields (minute, hour, date, month, day, user, command)
 - '*' → means everything; '-' → used when specifying concatenated preference values; ',' → use d when listing unconcatenated values; '/' → used when dividing a range of concatenated preference values by a specified interval.

```
#          min (0 - 59)
#          |
#          +-- hour (0 - 23)
#          |
#          +-- day of month (1 - 31)
#          |
#          +-- month (1 - 12)
#          |
#          +-- day of week (0 - 6) (0 to 6 are Sunday to Saturday, or use names;
#                           7 is Sunday, the same as 0)
#          |
#          +-- user-name command to execute
```

Process management

Process scheduling

A cron can be used to run a specific command at a fixed time, date, or interval. Commands here can run specific processes as well as do things like change configuration files.

- cron
 - Related command: crontab

| Option | Description |
|--------|---|
| -l | <ul style="list-style-type: none">• Print what is set in the crontab |
| -e | <ul style="list-style-type: none">• Create or modify the contents of a crontab |
| -r | <ul style="list-style-type: none">• Delete the contents of a crontab |
| -u | <ul style="list-style-type: none">• Use when root user manipulates the crontab file for a specific user |

```
# crontab -l  
# crontab -e  
# crontab -r  
# crontab -e -u user1
```

Process management

Process scheduling

A cron can be used to run a specific command at a fixed time, date, or interval. Commands here can run specific processes as well as do things like change configuration files.

- Restrict cron users
 - Related files
 - /etc/cron.allow : crontab command permissions file
 - /etc/cron.deny : crontab command restrictions file
 - How to use
 - Insert the usernames you want to grant permissions or restrictions to within those files

```
# vi /etc/cron.deny
user1
# vi /etc/cron.allow
root
```

Process management

Process scheduling

A cron can be used to run a specific command at a fixed time, date, or interval. Commands here can run specific processes as well as do things like change configuration files.

- Restrict cron users
 - Differences based on file existence and settings

| File status (no content in the file) | | Availability | |
|--------------------------------------|-----------|--------------|-----|
| cron.allow | cron.deny | root | 사용자 |
| O | X | O | X |
| X | O | O | O |
| O | O | O | X |

- If both files exist, you can put a username in cron.deny to restrict it, but if the username exists in cron.allow, you can use it.
- If only a cron.deny file exists, you must put a username in cron.deny to restrict that user.
- If a cron.allow file exists, you must put a username in the cron.allow file to enable that user, regardless of whether a cron.deny file exists.

Daemon and service management

Daemon

Similar to the services used in Windows, Linux has daemons. Daemons are processes that run in the background and are used to run the main functions of Linux.

- Daemon

- A program that runs in the background and performs various tasks without direct user control.
- They usually have a 'd' at the end of their name, such as 'syslogd' or 'sshd', and run as a normal process.
- Daemons typically have the init process as their parent process (with a PPID of 1) and create child processes by forking.

```
# pstree -p
init(1)──ManagementAgent(1802)──{ManagementAgen}(1814)
                                └──{ManagementAgen}(1815)
:
   └──httpd(5702)──httpd(5704)
                      └──httpd(5705)
:
# ps -ef
:
root      5388      1  0 14:07 ?          00:00:00 /usr/sbin/httpd
apache    5390  5388  0 14:07 ?          00:00:00 /usr/sbin/httpd
apache    5391  5388  0 14:07 ?          00:00:00 /usr/sbin/httpd
:
```

Daemon and service management

Daemon

Similar to the services used in Windows, Linux has daemons. Daemons are processes that run in the background and are used to run the main functions of Linux.

- Daemon
 - Categorize by execution method
 - Stand-alone
 - A system that resides on its own and sends responses to client requests.
 - Daemons exist independently for fast response and processing.
 - (x)inetd (superdemon)
 - Short for extended internet service daemon
 - To reduce system load, we use a daemon called xinetd to manage batches.
 - It runs the necessary processes only when a service request is received and shuts them down automatically.
 - This was the inetd daemon until version 2.2 of the Linux kernel, but has been replaced by the xinetd daemon since version 2.4.

Daemon and service management

Daemon management

To efficiently manage daemons, Linux uses the /etc/rc.d directory for boot-related information, and the init.d directory and the rc0.d through rc6.d directories to control the execution of related daemons.

- In the /etc/rc.d/init.d directory
 - Include scripts that can start and stop services provided by the system
 - The internal script is formatted and takes only one argument.
 - Arguments : start, stop, restart, status, reload, etc.
 - E.g., /etc/rc.d/init.d/httpd start

```
# /etc/rc.d/init.d/httpd start
Starting httpd:                                     [  OK  ]
# /etc/rc.d/init.d/httpd stop
Stopping httpd:                                     [  OK  ]
# /etc/rc.d/init.d/httpd
Usage: httpd
{start|stop|restart|condrestart|try-restart|force-reload|reload|status|fullstatus|graceful|help|configtest}
```

Daemon and service management

Daemon management

To efficiently manage daemons, Linux uses the /etc/rc.d directory for boot-related information, and the init.d directory and the rc0.d through rc6.d directories to control the execution of related daemons.

- Daemon control commands
 - service [daemonname] [start|stop|restart]
 - A script that can easily start or stop scripts that exist in the /etc/rc.d/init.d directory by simply specifying the daemon name and argument values instead of typing them all in as absolute paths.
 - /etc/init.d/[daemonname] [start|stop|restart]
 - The /etc/init.d directory is symlinked to /etc/rc.d/init.d, which can be used by omitting the /rc.d/.

```
# ls -al /etc | grep init.d
lrwxrwxrwx.  1 root root   11 Jan  9 19:52 init.d -> rc.d/init.d

# /etc/init.d/httpd start
Starting httpd:                                     [  OK  ]
# service httpd stop
Stopping httpd:                                     [  OK  ]
```

Daemon and service management

Process control

A system has many processes running at the same time, and these processes are managed by giving them priority.

- Process prioritization

- PRI

- Priority value referenced by the operating system, with lower values having higher priority
 - Not manipulated by users or root, and granted as appropriate based on system conditions

- NI

- Lower values have higher priority, and only root can lower the value.
 - Priority value manipulated by user or root, can be set from -20 to 19.
 - When you set the NI value, Linux adjusts the priority by changing the PRI value accordingly.

```
# ps -l
```

| F S | UID | PID | PPID | C | PRI | NI | ADDR | SZ | WCHAN | TTY | TIME | CMD |
|-----|-----|-------|------|---|-----|----|------|-------|--------|-------|----------|------|
| 4 S | 0 | 3992 | 3857 | 0 | 80 | 0 | - | 47348 | poll_s | pts/0 | 00:00:00 | sudo |
| 4 S | 0 | 4002 | 3992 | 0 | 80 | 0 | - | 27089 | do_wai | pts/0 | 00:00:00 | bash |
| 0 T | 0 | 21997 | 4002 | 0 | 80 | 0 | - | 28134 | do_sig | pts/0 | 00:00:00 | vi |
| 0 T | 0 | 22007 | 4002 | 0 | 80 | 0 | - | 28134 | do_sig | pts/0 | 00:00:00 | vi |
| 0 T | 0 | 22070 | 4002 | 0 | 80 | 0 | - | 28134 | do_sig | pts/0 | 00:00:00 | vi |

Daemon and service management

Processes and the /proc directory

- The /proc directory

- When a new process is created, a subdirectory named its PID is created in the /proc directory, and information about that process is stored there.

```
# ls /proc
1      1758  2300  2978  34    37    49          cpuinfo      mtrr
10     1760  23569 2983  3402  3717  5          crypto       net
100    1766  24    2992  3407  3760  50         devices      pagetypeinfo
101    18    25    3     3414  3763  51         diskstats   partitions
10194  1832  2580  30    3415  3765  52         dma        sched_debug
11  18834  26    3012  3417  38    5267        driver      schedstat
:
:
```

- Configure a specific process /proc/PID internally

```
# ls /proc/7100
attr      coredump_filter  io          mountstats    pagemap      stack
autogroup cpuset           limits       net          personality  stat
auxv      cwd              loginuid    ns          root        statm
cgroup    environ          maps        numa_maps    sched       status
clear_refs exe             mem        oom_adj     schedstat   syscall
cmdline   fd               mountinfo  oom_score   sessionid  task
comm      fdinfo          mounts     oom_score_adj smaps      wchan
```

Daemon and service management

Processes and the /proc directory

- /proc/PID key components

| Component | Description |
|-----------|--|
| cmdline | <ul style="list-style-type: none">• Details about command line options |
| cwd | <ul style="list-style-type: none">• Working Directory |
| environ | <ul style="list-style-type: none">• Details about the process environment |
| exe | <ul style="list-style-type: none">• The command that started the process |
| fd | <ul style="list-style-type: none">• The directory containing file descriptor details |
| maps | <ul style="list-style-type: none">• Memory map details for execution commands and library files |
| mounts | <ul style="list-style-type: none">• Mount details for your system |
| root | <ul style="list-style-type: none">• The root directory of the process |
| stat | <ul style="list-style-type: none">• The status of that process |
| statm | <ul style="list-style-type: none">• Details about the memory state of a process• size (total program size), resident (allocated memory size), shared (number of shared pages), trs (number of pages in text), drs (number of pages in data/stack), lrs (number of pages in library), dt (number of dirty pages) |
| status | <ul style="list-style-type: none">• Files containing status details about the process |

06

Linux network management

- Overview
- Understanding Linux network
- Linux network configuration and management

Overview

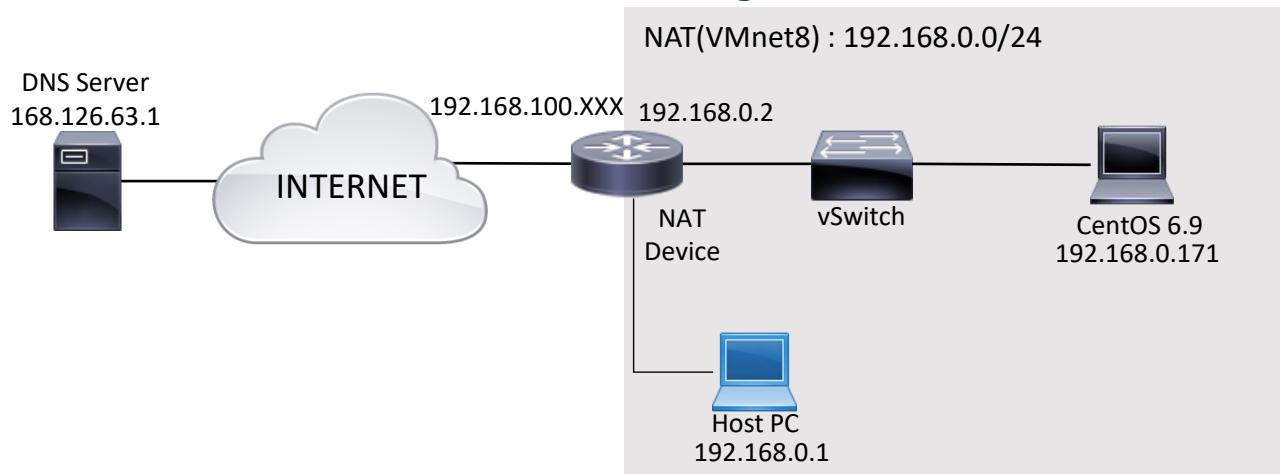
- Network configuration
 - Numbers 0 and 255 in the IP address are typically not used for the network configuration.
 - Number 0 : reserved number for the network address
 - Number 255 : reserved number for the network's broadcast address
 - Set up a gateway with an IP address number 1 or 254
 - Usually the first or last address is used, for consistency in gateway addresses.
 - Generally use 1, 254 only for 24 bits of the subnet, and use the first and last addresses that would match for more than 24 bits.
 - When configuring VMware
 - When configuring a network with VMware, IP1 is used by Vmware.
 - Configure and use the gateway address as IP2

Understanding Linux network

Understanding network configuration

In the default environment, the network is configured as shown in the figure below, and the point through which the Internet can be accessed is set as the gateway address.

- Environment configuration analysis
 - How to configure your network
 - Specify non-duplicate IP addresses from a band of IP addresses within a specific range.
 - Configure the IP address of the gateway to 192.168.0.2, the address of the NAT device, to use the Internet
 - Configure DNS server IP addresses to resolve English URLs to IPs



Understanding Linux network

Understanding network configuration

Since the Linux operating system is composed entirely of files, including in the form of devices, you will need to configure your network mostly in the form of files.

- Network configuration file (as of CentOS 6.9)
 - /etc/sysconfig/network-script/ifcfg-[network-interface-name]
 - Modify or check setup information for a network NIC card
 - /etc/resolv.conf
 - Modify or check configuration information for a domain server on your network
 - Service restart required after setting the change
 - /etc/sysconfig/network
 - Set the full default gateway address for your network, set the hostname, and set and verify whether network connections are allowed
 - /etc/udev/rules.d/70-persistent-net.rules
 - Files that control network preferences when physical equipment changes

Understanding Linux network

Network configuration

On CentOS, you can set up the network by modifying the configuration file. It is also sometimes necessary to modify certain configuration values and re-run the service.

- Network settings via configuration file conversion

- Edit a network card

```
$ ls /etc/sysconfig/network-scripts/
ifcfg-eth0    ifdown-ipv6      ifup        ifup-isdn   ifup-tunnel
ifcfg-lo     ifdown-isdn      ifup-aliases  ifup-plip   ifup-wireless
ifdown       ifdown-post     ifup-bnep    ifup-plusb  init.ipv6-global
ifdown-bnep  ifdown-ppp      ifup-eth     ifup-post   net.hotplug
# vi /etc/sysconfig/network-scripts/ifcfg-eth0
```

- Modify the network card settings as follows

```
DEVICE=eth0
ONBOOT=yes
NM_CONTROLLED=yes
BOOTPROTO=static
IPADDR=192.168.0.171
NETMASK=255.255.255.0
GATEWAY=192.168.0.2
NETWORK=192.168.0.0
DNS1=168.126.63.1
DNS2=8.8.8.8
```

Understanding Linux network

Network configuration

On CentOS, you can set up the network by modifying the configuration file. It is also sometimes necessary to modify certain configuration values and re-run the service.

- Network settings via configuration file conversion
 - Additional settings
 - If you want to set the IP and other settings to be fixed, set them as follows, with variations based on your environment.
 - Only the main settings (in bold) need to be set.

```
$ vi /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE=eth0
BOOTPROTO=static
HWADDR=[existing setting]          #physical address
NM_CONTROLLED=no #allow convenient network settings in GUI mode, unnecessary in TUI
ONBOOT=yes                  #determine whether to enable the network card at boot time
TYPE=Ethernet
BROADCAST=[Broad cast address]
IPADDR=[IP address]
NETMASK=[subnetmask address]
NETWORK=[network address]
ETHTOOL_OPTS=wol g #enable wake on lan feature, requires ethtool and is installed on CentOS
USERCTL=no #allow normal users to control eth0
IPV6INIT=no #IPv6 enabled or disabled
```

Understanding Linux network

Network configuration

On CentOS, you can set up the network by modifying the configuration file. It is also sometimes necessary to modify certain configuration values and re-run the service.

- Network settings via configuration file conversion

- Set up a DNS server

```
$ vi /etc/resolv.conf
```

- Modify your DNS settings as follows

```
search localdomain
nameserver 168.126.63.1
nameserver 168.126.63.2
```

- Or, depending on your setup, modify the settings in /etc/ifcfg-eth0 as follows

```
$ vi /etc/sysconfig/network-scripts/ifcfg-eth0
BOOTPROTO=static
GATEWAY=192.168.0.2
TYPE=Ethernet
DNS1=168.126.63.1
DNS2=168.126.63.2
```

- Network Services

```
$ sudo /etc/init.d/network restart
```

Understanding Linux network

Network configuration

You can also use commands to set up your network. However, this is not a good method for maintenance, as it is only temporarily registered and will be reset on reboot.

- Network settings using commands
 - Set up an IP and a gateway

```
ifconfig [device name] [IP] netmask [subnet dddress] broadcast [broadcast address]
route add -net [network address] netmask [subnet address] [device name].
route add default gw [IP address] dev [device name]
```

```
$ ifconfig eth0 192.168.0.171 netmask 255.255.255.0 broadcast 192.168.0.255 # set the terminal IP address
$ route add -net 192.168.0.0 netmask 255.255.255.0 eth0                      # set network band
$ route add default gw 192.168.0.2 dev eth0                                # set gateway address
```

Understanding Linux network

Network configuration

You can also use commands to set up your network. However, this is not a good method for maintenance, as it is only temporarily registered and will be reset on reboot.

- Check your network settings
 - View network card information, including IP, physical address, broadcast, subnetmask, etc.

```
$ ifconfig
eth0    Link encap:Ethernet HWaddr 00:0C:29:B2:52:55
        inet addr:192.168.0.171 Bcast:192.168.10.255 Mask:255.255.255.0
        inet6 addr: fe80::20c:29ff:feb2:5255/64 Scope:Link
              UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
              RX packets:0 errors:0 dropped:0 overruns:0 frame:0
              TX packets:16 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:0 (0.0 b) TX bytes:1128 (1.1 KiB)
```

Understanding Linux network

Network configuration

It's a good idea to check your configured network settings. Make sure that your current settings are correct and that your network connection is working.

- Check your network settings
 - Check the global address or gateway address

```
# route -n
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref    Use Iface
192.168.0.0     0.0.0.0       255.255.255.0 U     0      0        0 eth0
169.254.0.0     0.0.0.0       255.255.0.0   U     1002   0        0 eth0
0.0.0.0         192.168.0.2  0.0.0.0       UG    0      0        0 eth0
```

- Check your DNS settings

```
# nslookup
> server
Default server: 168.126.63.1
Address: 168.126.63.1#53
```

Linux network configuration and management

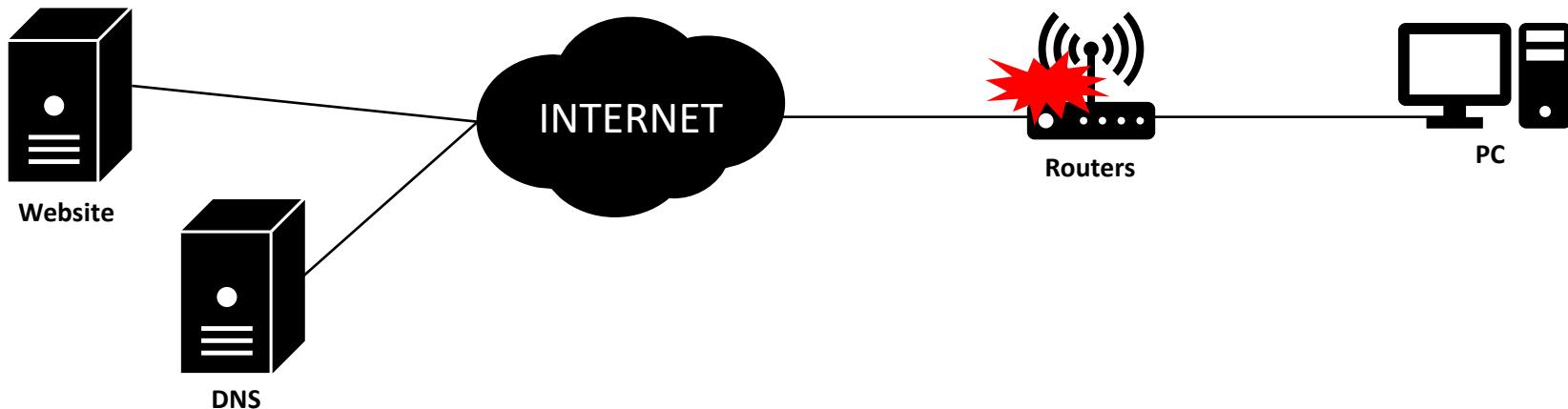
Network Troubleshooting

In the case of segmented network connections, you need to understand how to connect to each segment, determine which segments are not working well, and use the ping (ICMP) command to check if the system is working well.

- How to check for errors by network segment
 - If you're having problems with your router

```
# ping [gateway address]
```

- For router problems, check the gateway with route -n or the IP address with the ifconfig command because communication to the gateway address is not possible.



Linux network configuration and management

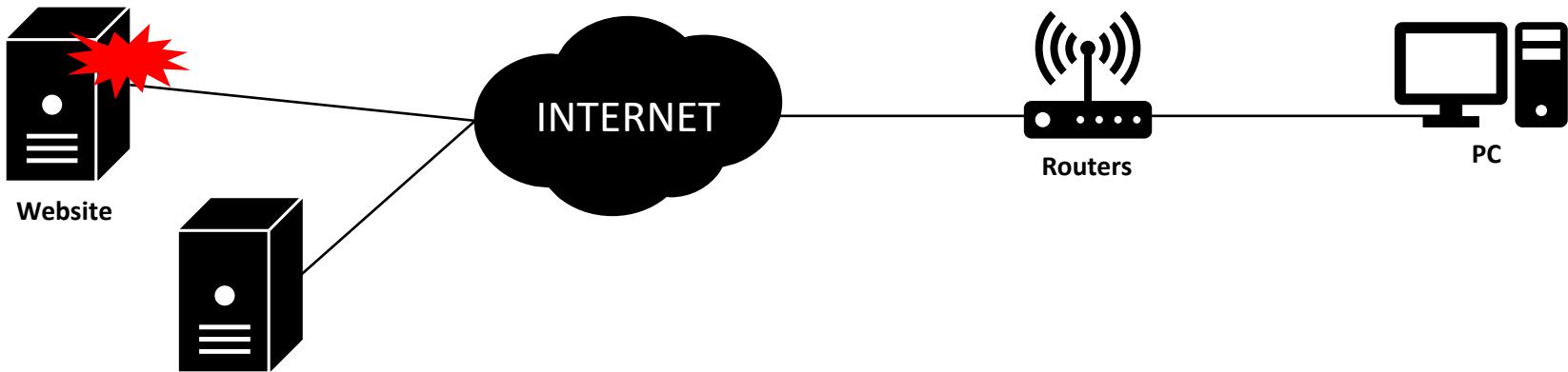
Network Troubleshooting

In the case of segmented network connections, you need to understand how to connect to each segment, determine which segments are not working well, and use the ping (ICMP) command to check if the system is working well.

- How to check for errors by network segment
 - If you can't access your website

```
# ping [DNS server address]  
# nslookup  
> server
```

- If the connection to the gateway is good, the domain server may be misconfigured or unavailable. Check the DNS connectivity and external Internet availability.



07

Advanced Linux management

- Overview
- Logs
- Linux access control

Overview

- Manage logs
 - Log management is important for all systems, not just Linux.
 - Logs record everything that happens on a system.
 - E.g., in the event of a security breach, logs can be used to analyze the path of the intrusion.
 - E.g., if you do something wrong on a Linux system, you can recover it by analyzing the logs.
 - Mainly for server logs, we use tools, automatic classification, and backup.
 - rsyslog, logrotate, etc.
- Access control on Linux is also the area of log records.
 - It manages access to the server.
 - Access control can be managed using tools such as Iptables, and TCP wrappers.
 - It can be used as a firewall replacement depending on your settings.

Early Linux used a package called syslog, which has been replaced in recent Linux distributions by the rsyslog package. Logs are recorded by creating a log file for each setting in /var/log.

- rsyslog daemon
 - A package that greatly improves the performance of syslog, including multi-threading support, TCP support, SSL and TLS support, database support, outbound list restrictions, partial message filtering, and output format control.
 - Configuration file
 - CentOS : /etc/rsyslog.conf
 - Ubuntu : /etc/rsyslog.d/50-default.conf
 - Daemons action script
 - CentOS : /etc/rc.d/init.d/rsyslog
 - Ubuntu : /etc/init.d/rsyslog

Early Linux used a package called syslog, which has been replaced in recent Linux distributions by the rsyslog package. Logs are recorded by creating a log file for each setting in /var/log.

- rsyslog configuration file
 - The /etc/rsyslog.conf or /etc/rsyslog.d/50-default.conf file
 - Default configuration format

facility.priority action

- Facility : means some kind of service, the type of program that generates the message
- Priority
 - The level of risk, and if it's higher than the level you set, a message will be logged.
 - '=' is used for the message to be logged if it is equal to the risk of that level.
 - '!' is used to exclude the message.
- Action : settings for the destination or action to send the message to, usually a filename.

Logs

rsyslog

- rsyslog configuration file
 - List of facilities

| Type | Description |
|-----------------|---|
| cron | <ul style="list-style-type: none">• Message generated by a scheduling program like cron |
| auth, security | <ul style="list-style-type: none">• Message generated by an authentication program type such as login |
| authpriv | <ul style="list-style-type: none">• Messages are also generated when adding a user with a program type that requires authentication, such as SSH. |
| daemon | <ul style="list-style-type: none">• Message from multiple daemons, such as telnet, ftp, etc. |
| kern | <ul style="list-style-type: none">• Message generated by kern |
| lpr | <ul style="list-style-type: none">• Message generated by print-type programs |
| mail | <ul style="list-style-type: none">• Messages generated by the mail system |
| mark | <ul style="list-style-type: none">• Types of dates created by syslogd |
| news | <ul style="list-style-type: none">• Message generated by the Usenet News software type |
| syslog | <ul style="list-style-type: none">• Message generated by the syslog program |
| user | <ul style="list-style-type: none">• User processes |
| uucp | <ul style="list-style-type: none">• Message from the Unix to Unix copy protocol (UUCP) system |
| local0 ~ local7 | <ul style="list-style-type: none">• Types left as spares |
| * | <ul style="list-style-type: none">• Means all facilities |

Logs

rsyslog

- rsyslog configuration file
 - List of priorities

| Type | Description |
|---------------|--|
| none | <ul style="list-style-type: none">• Exclude the specified facilities.• Typically used when you have completed the settings for other facilities and want to exclude specific facilities after the ':' |
| debug | <ul style="list-style-type: none">• Message that appears when debugging a program |
| info | <ul style="list-style-type: none">• Message for statistics, basic information |
| notice | <ul style="list-style-type: none">• Message that requires special attention, but is not an error |
| warning, warn | <ul style="list-style-type: none">• Warning message that requires attention |
| error, err | <ul style="list-style-type: none">• Message generated when an error occurs |
| crit | <ul style="list-style-type: none">• Message that isn't too urgent, but is at the stage where something is wrong with the system. |
| alert | <ul style="list-style-type: none">• Alerts you to a situation that requires immediate adjustments |
| emerg, panic | <ul style="list-style-type: none">• Alerts you to a dangerous situation that needs to be communicated to all users |

Logs

rsyslog

- rsyslog configuration file
 - List of actions

| Type | Description |
|---------------------|---|
| file | <ul style="list-style-type: none">• Log to a specified file |
| @host | <ul style="list-style-type: none">• Forward a message to a specified host |
| user | <ul style="list-style-type: none">• If the specified user is logged in, forward to that user's terminal |
| * | <ul style="list-style-type: none">• Forward to the screen of all currently logged in users |
| console or terminal | <ul style="list-style-type: none">• Forward a message to a specific terminal |

- rsyslog configuration file
 - Examples of rsyslog configuration files

```
$ vi /etc/rsyslog.conf or vi /etc/rsyslog.d/50-default.conf
```

```
*.=crit;kern.none /var/log/critical
```

- Of all facility messages, only crit-level messages are logged to the /var/log/critical file, excluding kernel messages.

```
*.emerg      *
```

- Send message to all users for all emergent and higher level issues

```
authpriv.*  root,user1
```

- Send authentication-related logs to the terminals of the root and user1 users

```
authpriv.*  /dev/tty2
```

- Send authentication-related logs to /dev/tty2

```
mail.*;mail.!={info}      /var/log/maillog
```

- All mail-related information is logged to /var/log/maillog, except for info-level logs.

```
uucp,news.crit      /var/log/news
```

- Crit-level and above messages from UUCP and news are logged to /var/log/news

Log files are stacked as they are constantly appended, so the size of the file will continue to grow. To avoid this, there is a program called logrotate that splits the log file into multiple files.

- logrotate overview

- Support auto-rotate, compress, uninstall, etc.
- Each log file is rotated on a daily, weekly, and monthly basis.
- Although logrotate is available via the command line, it is currently registered in /etc/cron.daily, where it is used to run the scheduling.
- logrotate configuration file : /etc/logrotate.conf
- How to use logrotate

logrotate [option] Configuration File

- logrotate key options

| Option | Description |
|-------------|--|
| -f(--force) | <ul style="list-style-type: none">• Force a preference file to be read and run |

- Examples of using the logrotate command

```
$ logrotate -f /etc/logrotate.conf
```

- Key settings of /etc/logrotate.conf

| Setting | Description |
|--------------------------|---|
| weekly | <ul style="list-style-type: none">Set to rotate log files weeklyLog files not specified will be applied to this setting if they are listed at the top. |
| rotate 4 | <ul style="list-style-type: none">Set to rotate up to 4 timesCreate a default logfile, in such a way as log.1, log.2, log.3, and log.4 |
| create | <ul style="list-style-type: none">Set to create an empty log file after a rotate |
| dateext | <ul style="list-style-type: none">Append the date to a log file that is generated on a rotating basisE.g., create as maillog-20200121 |
| compress | <ul style="list-style-type: none">Use when compressing log files created after rotating them |
| include /etc/logrotate.d | <ul style="list-style-type: none">Set to apply rotation to files set inside the /etc/logrotate.d directory as well |
| nomissingok | <ul style="list-style-type: none">Print an error message if the log file does not exist, set to default |
| missingok | <ul style="list-style-type: none">Go to the next file without printing an error message if the log file doesn't exist |

- An example of the /etc/logrotate.conf settings

```
$ vi /etc/logrotate.conf
weekly

rotate 4

create

include /etc/logrotate.d

- You can specify messages separately by naming the log file.

/var/log/wtmp {
    missingok
    monthly
    create 0664 root utmp
    rotate 1
}

/var/log/btmp {
    missingok
    monthly
    create 0660 root utmp
    rotate 1
}
```

- Files related to logrotate
 - Files with the rotation date for each log file (different filenames for each OS or version)
 - `/var/lib/logrotate/status`
 - `/var/lib/logrotate.status`
 - `/var/lib/logrotate/logrotate.status`

```
$ vi /var/lib/logrotate/status or /var/lib/logrotate.status or /var/lib/logrotate/logrotate.status
```

```
logrotate state -- version 2
"/var/log/ConsoleKit/history" 2020-1-10
"/var/log/yum.log" 2020-1-10
"/var/log/cups/error_log" 2020-1-20
"/var/log/sssd/*.log" 2020-1-10
"/var/log/dracut.log" 2020-1-10
"/var/log/cups/access_log" 2020-1-20
"/var/log/libvirt/lxc/*.log" 2020-1-10
"/var/log/httpd/*log" 2020-1-10
"/var/log/httpd/error_log" 2020-1-20
"/var/log/wtmp" 2020-1-10
:
:
```

On Linux, the `/var/log` directory records and manages all logs on the system, and the `/etc/rsyslog.conf` file specifies the location of the system log files.

- Key log files on a Linux system

- `/var/log/messages`

- This is a file that logs standard messages generated by the system. Most logs are written to this file.
 - It contains the date and time, the host name from which a message originated, and the name of the internal system or application that generated the message, followed by a ":" to separate them, and lists the messages in the order in which they occurred.
 - It is written in plain text format and can only be read by the root user with commands such as `cat` or `vi`.

```
$ cat /var/log/messages
Jan 20 12:40:02 localhost rsyslogd: [origin software="rsyslogd" swVersion="5.8.10" x-pid="1998" x-
info="http://www.rsyslog.com"] rsyslogd was HUPed
Jan 20 12:40:04 localhost NetworkManager[2092]: <info> Auto-activating connection 'System eth0'.
Jan 20 12:40:04 localhost NetworkManager[2092]: <info> Activation (eth0) starting connection 'System eth0'
Jan 20 12:40:04 localhost NetworkManager[2092]: <info> (eth0): device state change: disconnected -> prepare
(reason 'none') [3 4 0]
Jan 20 12:40:04 localhost NetworkManager[2092]: <info> Activation (eth0) Stage 1 of 5 (Device Prepare)
scheduled...
:
:
```

On Linux, the `/var/log` directory records and manages all logs on the system, and the `/etc/rsyslog.conf` file specifies the location of the system log files.

- Key log files on a Linux system

- `/var/log/wtmp`

- Log files that contain a history of successful login/logout information for users, and boot/shutdown information for the system.
 - It is a binary file and cannot be read with commands such as `cat` or `vi`.
 - Related command : `last`

```
$ last
user1      pts/0          :0.0              Mon Jan 20 10:35  still logged in
user1      tty1           :0                Mon Jan 20 10:35  still logged in
reboot    system boot   2.6.32-754.el6.x  Mon Jan 20 10:18 - 11:08 (1+00:49)
user1      pts/0          :0.0              Thu Jan 16 15:06 - down  (1+04:03)
user1      tty1           :0                Thu Jan 16 15:06 - down  (1+04:03)
reboot    system boot   2.6.32-754.el6.x  Thu Jan 16 15:05 - 19:09 (1+04:03)
user1      pts/0          :0.0              Fri Jan 10 09:35 - down  (5+10:13)
user1      tty1           :0                Fri Jan 10 09:35 - down  (5+10:13)
reboot    system boot   2.6.32-754.el6.x  Thu Jan  9 19:59 - 19:49 (5+23:49)
```

Logs

Key log-related files

On Linux, the `/var/log` directory records and manages all logs on the system, and the `/etc/rsyslog.conf` file specifies the location of the system log files.

- Key log files on a Linux system

- `/var/log/btmp`
 - Log files that contain a history of failed login attempts
 - It is a binary file and cannot be read with commands such as `cat` or `vi`.
 - Related command : `lastb`

```
$ lastb
user1      ssh:notty      10.211.55.2      Tue Jan 21 11:27 - 11:27 (00:00)
user1      ssh:notty      10.211.55.2      Tue Jan 21 11:27 - 11:27 (00:00)
root       ssh:notty      10.211.55.2      Tue Jan 21 11:27 - 11:27 (00:00)
root       ssh:notty      10.211.55.2      Tue Jan 21 11:27 - 11:27 (00:00)
root       ssh:notty      10.211.55.2      Tue Jan 21 11:27 - 11:27 (00:00)
UNKNOWN    tty1           Mon Jan 20 17:59 - 17:59 (00:00)
user1      tty1           Mon Jan 20 17:59 - 17:59 (00:00)
user1      tty1           Thu Jan 16 15:25 - 15:25 (00:00)

btmp begins Thu Jan 16 15:25:59 2020
```

On Linux, the `/var/log` directory records and manages all logs on the system, and the `/etc/rsyslog.conf` file specifies the location of the system log files.

- Key log files on a Linux system

- `/var/log/lastlog`
 - A log file that contains the most recent successful login
 - It is a binary file and cannot be read with commands such as `cat` or `vi`.
 - Related command: `lastlog`

```
$ lastlog
```

| Username | Port | From | Latest |
|------------|-------|-------------|--------------------------------|
| root | | | **Never logged in** |
| daemon | | | **Never logged in** |
| bin | | | **Never logged in** |
| libuuid | | | **Never logged in** |
| syslog | | | **Never logged in** |
| messagebus | | | **Never logged in** |
| landscape | | | **Never logged in** |
| user1 | pts/0 | 10.211.55.2 | Tue Jan 21 11:28:03 +0900 2020 |
| test | | | **Never logged in** |
| sshd | | | **Never logged in** |

- Other log files on a Linux system
 - /var/log/secure
 - A log file that logs records related to authentication-based access
 - /var/log/maillog
 - A log file that logs mail-related actions, such as sendmail
 - /var/log/xferlog
 - A log file that records actions related to FTP connections
 - /var/log/cron
 - A log file that records cron-related information
 - /var/log/boot.log
 - A log file that records information about file system checks at boot time, service daemons
 - /var/log/dmesg
 - A log file that records all messages issued at boot time
 - It can be printed using the dmesg command and is called the kernel boot message log

Linux access control

Linux firewall (iptables)

Linux typically has its own firewall, iptables, which allows you to control access by allowing or denying protocols, IP addresses, and ports based on their origin or destination.

- iptables overview

- A packet filtering tool on Linux, used for firewall configuration or NAT
- The action on the packet is checked for each rule in turn, starting from the top.
- It performs the ACCEPT, DROP, etc. specified by the target on packets that match the rule.
- If the rule matches and the action is performed, the packet is processed according to the result of the rule, ignoring any further rules in the chain.
- When a packet reaches the bottom of the chain because it doesn't match all the rules in the chain, a predetermined default policy is performed.
 - policy ACCEPT
 - policy DROP
 - Typically, set the default policy to DROP and the ports and IP addresses to ACCEPT
- How to use

```
$ iptables [-t Table] [Action] [Chain] [Match] [-j Target]
```

Linux access control

Linux firewall (iptables)

Linux typically has its firewall, iptables, which allows you to control access by allowing or denying protocols, IP addresses, and ports based on their origin or destination.

- Install iptables (as of CentOS 6.9).

- Check for installation.

```
$ rpm -qa | grep iptables  
iptables-ipv6-1.4.7-16.el6.x86_64  
iptables-1.4.7-16.el6.x86_64
```

- If you don't see any results, install it as follows.

```
$ yum install -y iptables
```

- Check the status.

```
$ chkconfig --list | grep iptables  
iptables           0:off 1:off 2:off 3:off 4:off 5:off 6:off  
$ chkconfig iptables on  
iptables           0:off 1:off 2:on  3:on  4:on  5:on  6:off
```

- Start a service.

```
$ service iptables start
```

Linux access control

Linux firewall (iptables)

Linux typically has its firewall, iptables, which allows you to control access by allowing or denying protocols, IP addresses, and ports based on their origin or destination.

- The iptables options
 - Table
 - Types : filter, nat, mangle, raw, security
 - Of these, the one for filtering packets is the filter.
 - Chain (three predefined chains in the filter table)
 - INPUT
 - All packets destined for the host computer
 - OUTPUT
 - All packets originating from the host computer
 - FORWARD
 - All packets for which the host computer is not the destination
 - That is, packets passing through a host computer used as a router.

Linux access control

Linux firewall (iptables)

- The iptables options
 - Actions

| Type | Description |
|---------------------|--|
| -A (--append) | <ul style="list-style-type: none">• Add a new rule |
| -D (--delete) | <ul style="list-style-type: none">• Delete a rule |
| -C (--check) | <ul style="list-style-type: none">• Perform a packet test |
| -R (--replace) | <ul style="list-style-type: none">• Change an existing rule to a new rule |
| -I (--insert) | <ul style="list-style-type: none">• Insert a new rule |
| -L (--list) | <ul style="list-style-type: none">• Print the currently set rules |
| -F (--flush) | <ul style="list-style-type: none">• Delete all rules from the chain |
| -Z (--zero) | <ul style="list-style-type: none">• Set packet and byte counter values to 0 for all chains |
| -N (--new) | <ul style="list-style-type: none">• Create a new chain |
| -X (--delete-chain) | <ul style="list-style-type: none">• Delete a chain |
| -P (--policy) | <ul style="list-style-type: none">• Change the default policy |

Linux access control

Linux firewall (iptables)

- The iptables options

- Match - the condition that iptables must satisfy when processing the packet.

| Type | Description |
|----------------------|---|
| --source (-s) | <ul style="list-style-type: none">• Match a source IP address or network |
| --destination (-d) | <ul style="list-style-type: none">• Match against a destination IP address or network |
| --protocol (-p) | <ul style="list-style-type: none">• Match to a specific protocol |
| --in-interface (-i) | <ul style="list-style-type: none">• Input interface |
| --out-interface (-o) | <ul style="list-style-type: none">• Print interface |
| --state | <ul style="list-style-type: none">• Match with connection status |
| --string | <ul style="list-style-type: none">• Match byte order of application layer data |
| --comment | <ul style="list-style-type: none">• Comment up to 256-byte associated with rules in kernel memory |
| --syn (-y) | <ul style="list-style-type: none">• Deny SYN packets |
| --fragment (-f) | <ul style="list-style-type: none">• Specify a rule for fragments after the second |
| --table (-t) | <ul style="list-style-type: none">• Tables to process |
| --jump (-j) | <ul style="list-style-type: none">• Specify what to do with packets that match the rule |
| --match (-m) | <ul style="list-style-type: none">• Match with specific modules |

Linux access control

Linux firewall (iptables)

- The iptables options
 - Target
 - ACCEPT : accept the packet
 - DROP : discard the packet as if it had never been sent
 - Drop a packet without showing the user any warning message
 - RETURN: continue processing packets within the call chain
- Change the iptables default policy
 - Establish a default policy for what to do with packets not specified in the rule.

```
# block all incoming by default
$ iptables -P INPUT DROP
$ iptables -P FORWARD DROP

# allow all by default for outgoing
$ iptables -P OUTPUT ACCEPT
```

Linux access control

Linux firewall (iptables)

- How to use iptables
 - Check the currently set iptables rules.

```
$ iptables --list  
:  
$ iptables -L  
:  
$ cat /etc/sysconfig/iptables  
:
```

- Save the currently set iptables rules.

```
$ service iptables save  
$ iptables-save > firewall.sh
```

- Import saved iptables rules.

```
$ iptables-restore < firewall.sh
```

- Reset the currently set iptables rules.

```
$ iptables -F
```

Linux access control

Linux firewall (iptables)

- How to use iptables

- Block specific IPs

```
$ iptables -I INPUT -s xxx.xxx.xxx.xxx -j DROP
```

- Allow specific IPs

```
$ iptables -A INPUT -s xxx.xxx.xxx.xxx -j ACCEPT
```

- Block specific ports

```
$ iptables -A INPUT -p tcp --dport xxxx -j DROP
```

- Allow ports on specific IPs

```
$ iptables -A INPUT -p tcp -s xxx.xxx.xxx.xxx --dport xxxx -j ACCEPT
```

- Delete an applied rule

```
$ iptables -D INPUT -p tcp -s xxx.xxx.xxx.xxx --dport xxxx -j ACCEPT
```

Linux access control

Linux firewall (iptables)

- How to use iptables
 - Question 1) If I have iptables set up as shown below, what happens when a ping is sent from another server to mine?

```
$ iptables -A INPUT -p icmp --icmp-type echo-request -j DROP
$ iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
$ iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source          destination
DROP      icmp -- anywhere        anywhere        icmp echo-request
ACCEPT    icmp -- anywhere        anywhere        icmp echo-request

Chain FORWARD (policy ACCEPT)
target     prot opt source          destination

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

Linux access control

Linux firewall (iptables)

- How to use iptables
 - Question 2) Write a rule to block all access ports for connections with an originating IP address of 192.168.0.111.
 - Question 3) Add a rule to the INPUT chain that rejects icmp packets with a destination address of 127.0.0.1.
 - Question 4) Add a rule to the INPUT chain that accepts tcp packets with a destination port of http.
 - Question 5) Change the port of the rule you added above to 8880.

Linux access control

Linux firewall (TCP Wrapper)

Linux has a TCP Wrapper that allows you to control access to per-service IP bands or hostnames, and using the techniques above, you can seamlessly control access to Linux servers.

- TCP Wrapper overview

- If a service request comes in from an arbitrary host, log the hostname and service name by checking to see if it's a system you've allowed access to before running the actual daemon.
- Allows access control (ACL) for FTP, Telnet, SSH, xinetd-based services, etc.
- Configuration files
 - /etc/hosts.allow (access permissions configuration file)
 - /etc/hosts.deny (access denial configuration file)
- How to check if your service is eligible for TCP Wrapper settings

```
$ ldd /usr/sbin/xinetd
    linux-vdso.so.1 => (0x00007ffd89bfb000)
    libselinux.so.1 => /lib64/libselinux.so.1 (0x00007fed6c8ed000)
    libwrap.so.0 => /lib64/libwrap.so.0 (0x00007fed6c6e2000)
:
:
```

Linux access control

Linux firewall (TCP Wrapper)

Linux has a TCP Wrapper that allows you to control access to per-service IP bands or hostnames, and using the techniques above, you can seamlessly control access to Linux servers.

- TCP Wrapper basic syntax
 - How to use configuration files

```
$ vi /etc/hosts.allow or vi /etc/hosts.deny  
[service name]: [host]
```

- Examples
 - ALL: ALL
 - Allow (hosts.allow) or deny (hosts.deny) all services to all IPs
 - sshd: ALL
 - Allow or deny all IPs for the SSHD service
 - in.telnetd: 192.168.0.*
 - Allow or deny the 192.168.0.* ipv6 band for the telnet service included with xinetd

Linux access control

Linux firewall (TCP Wrapper)

Linux has a TCP Wrapper that allows you to control access to per-service IP bands or hostnames, and using the techniques above, you can seamlessly control access to Linux servers.

- TCP Wrapper basic syntax

- Set up /etc/hosts.deny

```
$ vi /etc/hosts.deny  
ALL: ALL
```

- Typically, you would first block all hosts for all services in the hosts.deny file, and then set the hosts.allow file for the hosts or IPs you want to connect to (whitelist method).
 - If the service is allowed in hosts.allow, it is allowed to be blocked in hosts.deny.

- /etc/hosts.allow

```
$ vi /etc/hosts.allow  
in.telnetd: localhost
```

- Set the hosts or IP bands you want to allow

Linux access control

Linux firewall (TCP Wrapper)

- Test your TCP Wrapper settings
 - Block localhost access to the Telnet service.

```
$ vi /etc/hosts.deny  
in.telnetd: localhost
```

- Test blocking Telnet access.

```
$ telnet localhost  
Trying ::1...  
Connected to localhost.  
Escape character is '^]'.  
Connection closed by foreign host.
```

- Allow localhost access to the Telnet service.

```
$ vi /etc/hosts.allow  
in.telnetd: localhost
```

- Test allowing Telnet access.

```
$ telnet localhost  
Trying ::1...  
Connected to localhost.  
Escape character is '^]'.  
CentOS release 6.9 (Final)  
Kernel 2.6.32-696.el6.x86_64 on an x86_64  
localhost.localdomain login:
```

08

Network fundamentals – introduction

- Network overview

Network overview

What does network mean?

A network is a collection of distributed devices connected by a communication network. This brings us back to the dictionary meaning of network.

- Dictionary meanings

- A computer network or data network is a digital telecommunications network which allows nodes to share resources. (Source: wiki)
- A digital communications network that allows resources to be shared among endpoints (terminals such as PCs and servers).
- Network communications begin to develop, starting with long-distance communication, the Alphanet.

Net



Work



Network

A form of communication in which computers are connected like a **net** using communications technology.

Something that connects two or more computers and allows them to **communicate (talk) with each other**.

Network overview

Global network history

Computer networks have been around for less than 60 years, and we'll take a look at how they have evolved from their earliest beginnings to today's technology.

- Chronicle
 - Late 1950s
 - A network of early computers was developed into a U.S. military radar system (SAGE: Semi-Automatic Ground Environment)
 - 1960s
 - In 1960, the Semi-Automated Business Research Environment (SABRE), a commercial airline reservation system, linked two mainframes online.
 - In 1964, the Dartmouth Timeshare system was developed for users of large computer systems at Dartmouth College.
 - In 1965, Western Electric CO. (AT&T) developed the first computer-controlled universal telephone switch.
 - In 1969, the Advanced Research Projects Agency Network (ARPANET) connected the first four nodes (University of California, Los Angeles, Stanford, University of California, Santa Barbara, and University of Utah) at 50 kbit/s.

Network overview

Global network history

Computer networks have been around for less than 60 years, and we'll take a look at how they have evolved from their earliest beginnings to today's technology.

- Chronicle
 - 1970s
 - In 1972, commercial services were introduced using X.25, which was later used as the underlying infrastructure for the expansion of TCP/IP networks.
 - In 1976, the Attached Resource Computer NETwork (ARCNET) was developed by John Murphy of Datapoint Corporation; token-pass networking was first used for shared storage.
 - After 1990
 - In 1995, transmission speeds were increased from 10 Mbps to 100 Mbps.
 - Present
 - 100+ Gbps communication speeds

09

TCP/IP essentials

- Overview
- Understanding the network terms
- Understanding network models
- The OSI models
- The TCP/IP models
- Understanding the Local Area Network (LAN)
- IPv4 and IPv6
- Subnetting

Overview

- Understanding the network terms
 - Basic network theory and terminology
- OSI 7-layer model
 - A model that breaks down the structure of how a network communicates into seven layers and describes how the layers interact with each other.
 - Get a step-by-step view of how your network communicates
- TCP/IP model
 - Divides network communications into four distinct layers
 - Commonly referred to as the TCP/IP stack or the TCP/IP four-layer model
- IPv4/IPv6
 - IPv4 stands for Internet Protocol version 4.
 - Most of the IP addresses you'll use are in IPv4.
- Subnetting
 - Imply sub-network
 - Use when you want to divide a large network into smaller pieces

Understanding the network terms

What is a network?

We will review the Internet, intranets, and extranets that are commonly used in networks and identify their characteristics.

- Internet and intranet
 - Internet
 - Refer to linking (inter) a network (net) to mean "linking multiple networks together."
 - Communicate using a single protocol
 - Access via a browser
 - Intranet
 - Means the internal Internet
 - Communicate using a single protocol
 - Extranet
 - Accessible to partners, customers, and in-house staff
 - Communicate with a single protocol

Understanding the network terms

Protocols

In order to communicate with each other, we must use the same promises to convey meaning, just as humans do in conversation, and communication within a computer has the same promise conventions. These are called protocols and understand them.

- Dictionary meanings
 - A system of forms and rules for sending and receiving messages between computers or telecommunications devices.
 - Defines the form, semantics, and synchronization processes, but is independent of implementation and methods.
 - Configurations
 - Physical aspects : transmission media, terminals, signals, line specifications, etc. used to transmit data.
 - Logical aspects : frame composition, meaning and function of each element, transmission procedure, etc.
 - Divided into closed protocols and open, universal protocols according to their public scope
 - The three pillars of a protocol

| | |
|-----------|--|
| Protocols | Syntax : indicates the format, coding and signal level of the data |
| | Semantic : include error checking or reconciliation information |
| | Timing : a period of time related to a pacing or sequence |

Understanding the network terms

Protocols

In order to communicate with each other, we must use the same promises to convey meaning, just as humans do in conversation, and communication within a computer has the same promise conventions. These are called protocols and understand them.

- Function

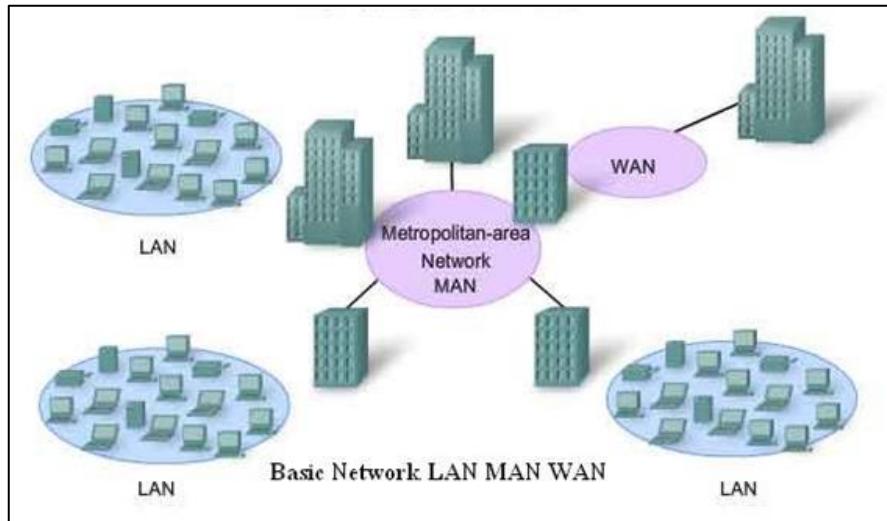
| Function | Description |
|----------------------------|---|
| Fragmentation and assembly | Improve efficiency by reducing errors when transferring information |
| Encapsulation | Append information to protect it |
| Connection control | The process of transferring data which includes establishing a connection between nodes, transferring data, and disconnecting |
| Flow control | Limit the amount and speed of data |
| Error control | Detect and correct errors errors during transmission |
| Synchronization | Maintain the same state between sender and receiver |
| Ordering | Give order to data |
| Address setting | The role of enabling recognition between sender and receiver for network communication |
| Multiplexing | A method of sharing limited communication links among multiple users or multiple systems on a single line |
| Transport services | Control prioritization, class of service, security requirements, and more |

Understanding the network terms

Network classification

Networks are classified differently in each category. They can be classified by size into WAN, MAN, and LAN.

- Categorize by size
 - WAN
 - Wide Area Network
 - Connect a network in a remote location
 - MAN
 - Metropolitan Area Network
 - Connect networks across multiple buildings or a city
 - LAN
 - Local Area Network
 - Smaller networks, such as within a building



Understanding the network terms

Network classification

Networks are classified differently in each category. They can be classified by size into WAN, MAN, and LAN.

- Categorize by size
 - Backbone network
 - Another name : PeriodNet
 - The top level of the network, the main communications network at the center of the network.
 - Features
 - Connect distant local area networks or telecommunications networks to share information.
 - Transmission rate : 128 Mbps to 10 Gbps or higher
 - The high-speed information and communication network implemented at the national level uses the backbone communication network to form a periodic network.

Understanding the network terms

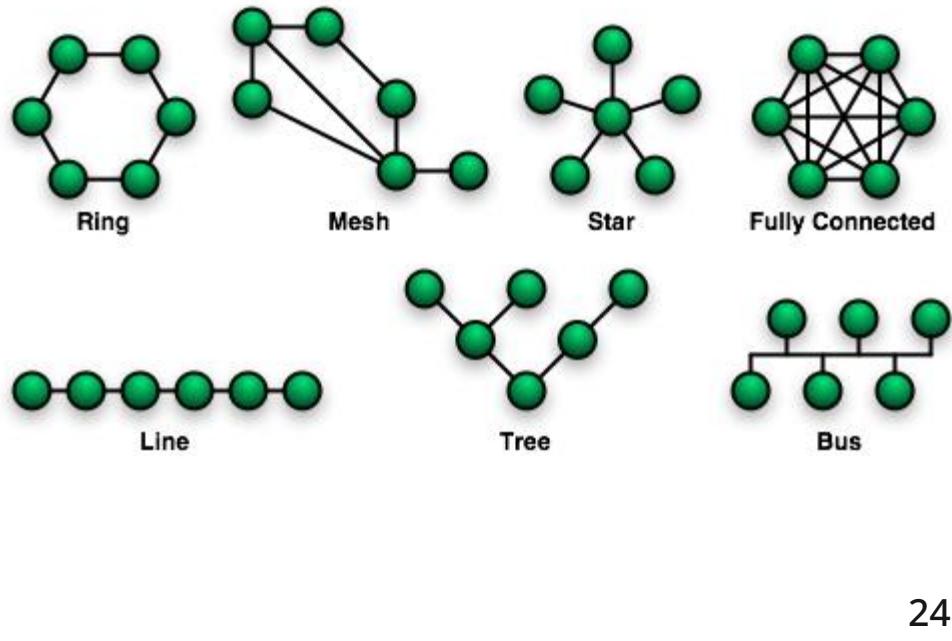
Network classification

Phase-based classification can be achieved through network topology. We will explore and understand each type.

- Categorization by topology

- Network topology

- The physical connection or the arrangement method of computer network elements.
 - For LANs, physical and logical topologies are possible.
 - Types
 - Bus
 - Star
 - Ring
 - Tree
 - Mesh
 - Others



Understanding the network terms

Network classification

Networks can be categorized by transmission methods, and there are various forms such as circuit-switched, packet-switched, and cell-switched networks.

- Classification by transmission modes
 - Switching systems
 - Circuit switching method
 - A form of pathing that establishes a connection with a fixed band assigned for data transmission.
 - Reliable delivery rates and simple routing
 - Require a relatively long connection time
 - Packet switching method
 - A form of data that is divided into regularly sized chunks that are routed independently to reach their destination.
 - Divided into virtual circuit and datagram modes
 - Virtual circuit mode : a virtual connection is established between the sending and receiving hosts, with the same forwarding path for all unit data.
 - Datagram mode : the path selection is sent independently for each data unit.

Understanding the network terms

Network classification

Networks can be categorized by transmission methods, and there are various forms such as circuit-switched, packet-switched, and cell-switched networks.

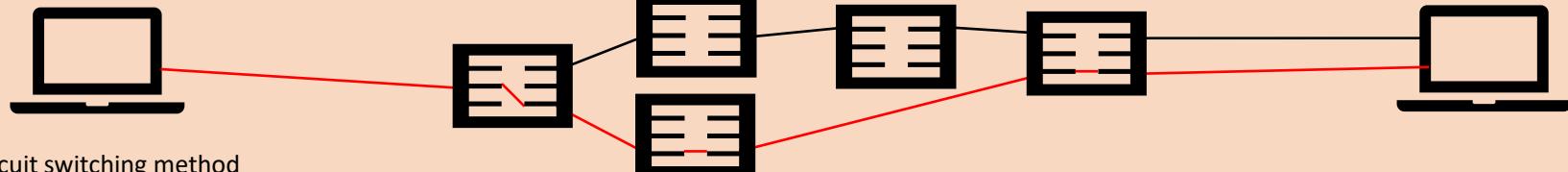
- Classification by transmission medium
 - Switching systems
 - Cell switching method
 - Create a fixed-size cell that contains only the minimum features needed for transmission
 - Create fixed-size packets of 53 bytes (5 bytes are headers)
 - Similar to packet-switched, using a connection-oriented packet-switched network.
 - Increase network throughput by minimizing switching time by setting up virtual circuits when connecting to the network.
 - The header contains payload type information, virtual line identifier, and error-checking information.

Understanding the network terms

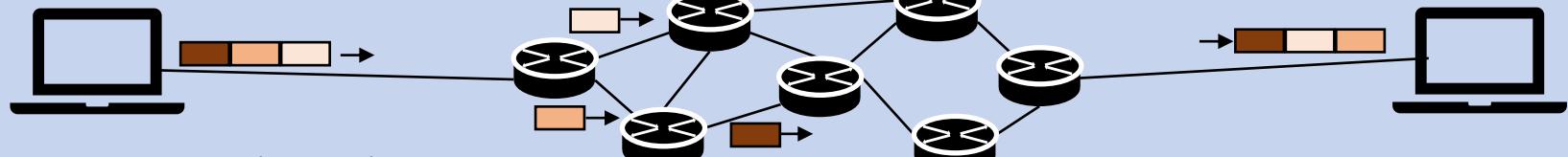
Network classification

Networks can be categorized by transmission methods, and there are various forms such as circuit-switched, packet-switched, and cell-switched networks.

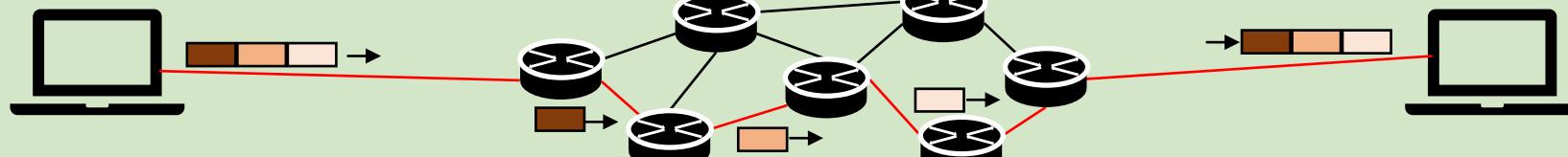
- Classification by transmission media - diagrams



Circuit switching method



Packet switching method (datagram)



Packet switching (virtual circuit) or cell switching methods

* However, the size of each data unit is different in the packet switching method, and the size is constant in the cell switching method.

Understanding the network terms

Network classification

Networks are sometimes categorized by how they are connected, and they are often subcategorized by size.

- Categorization by connection type

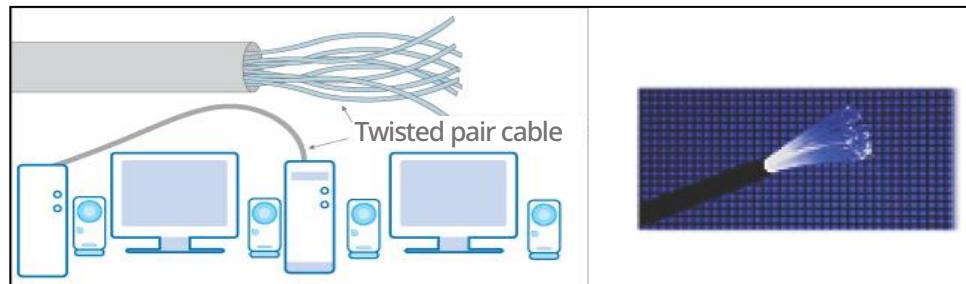
| Type | Wired network | Wireless network | Mobile network |
|---------------------------|---|-----------------------------------|---------------------------|
| Medium | Twisted pair cable, coaxial cable, fiber-optic cable | Radio wave | Radio wave |
| Standard | IEEE 802.3 | IEEE 802.11, IEEE 802.15, etc. | IEEE 802.16 IMT-2000 |
| Maximum transmission rate | 1000 Gbits/s (fiber-optic cable) | 866.7 Mbits/s (802.11ac) | 672 Mbits/s (HSPA+ WIMAX) |

Understanding the network terms

Network classification

Networks are sometimes categorized by how they are connected, and they are often subcategorized by size.

- What is a wired network?
 - When the computer network connection is physically wired, it is called a wired LAN.
- Features of wired networks
 - The most common type of wired network used in LAN environments is Ethernet (IEEE 802.3 standard).
 - Faster and safer than wireless networks
 - High installation costs (pre-wired or structural changes to buildings required)
 - Twisted pair, coaxial, fiber-optic cables



Understanding the network terms

Network classification

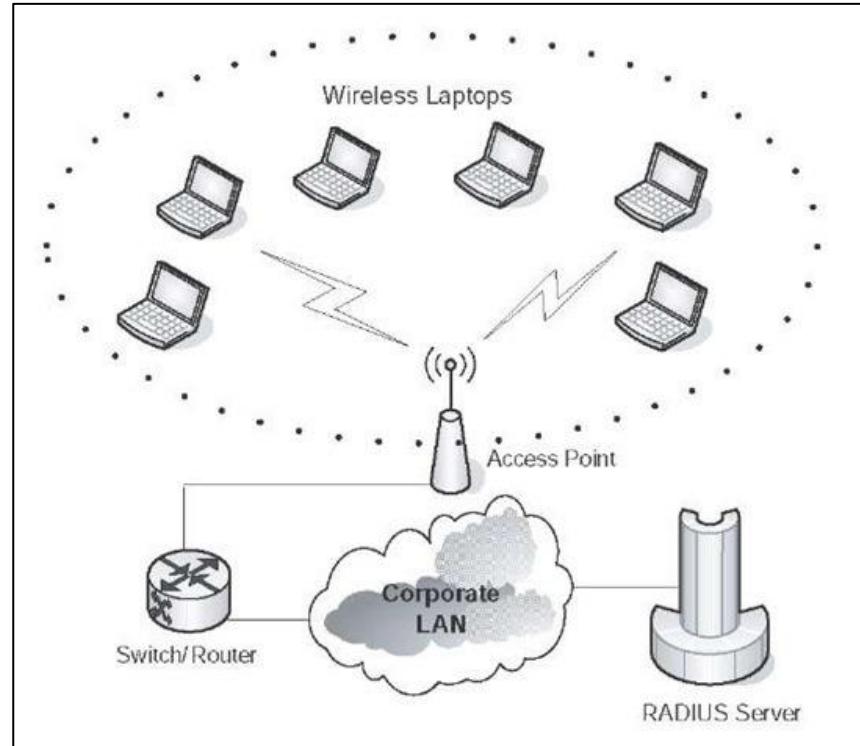
Networks are sometimes categorized by how they are connected, and they are often subcategorized by size.

- What is a wireless network?

- A wireless network is a computer network organized without wires.

- Features of wireless networks

- Wireless Internet networks, also called Wi-Fi networks
 - The computer has a wireless connection to the Internet through an AP at the hotspot.
 - Advantages : easy and inexpensive to install as no cabling is required.
 - Disadvantages : less secure than wired

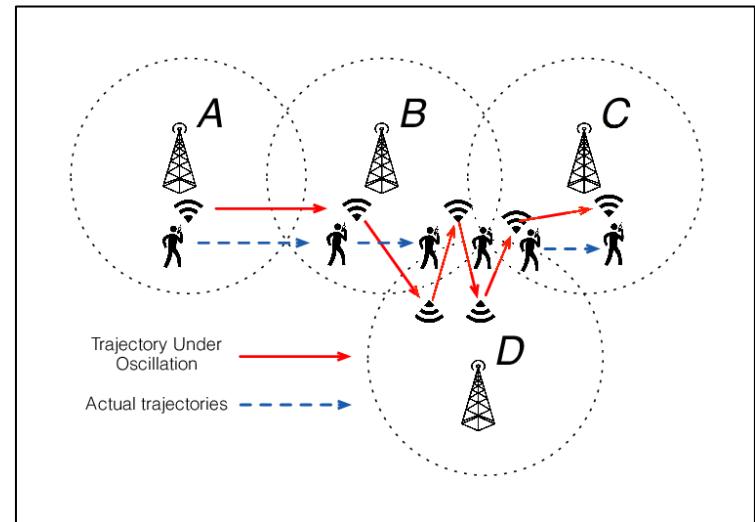


Understanding the network terms

Network classification

- What is a mobile network?
 - The development of mobile networks in the form of cells that cover areas of several kilometers to compensate for the shortcomings of wireless networks.
 - Disadvantages of wireless networks : Internet is only available in hotspot areas within a few tens of meters, not available on the move.

- Features of mobile networks
 - Wireless Broadband (WiBro) technology, Long Term Evolution (LTE) technology, and more
 - Can be used during high-speed travel (120 km) in cars, trains, etc.
 - Connect to the Internet within a coverage area called a cell, which can be several kilometers in radius.
 - No mobile network connectivity outside of the cell

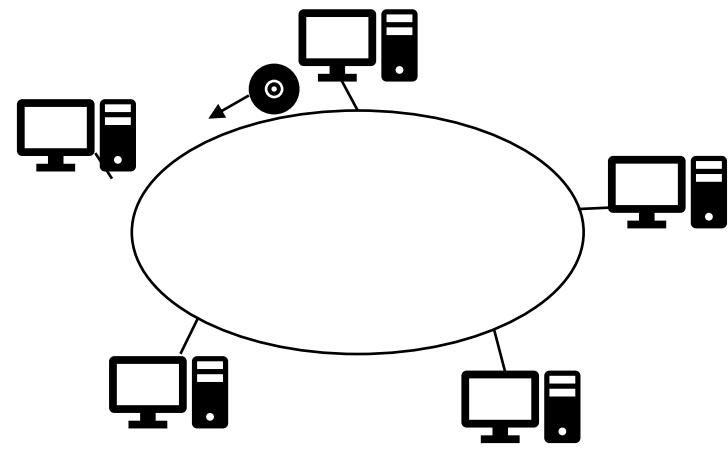
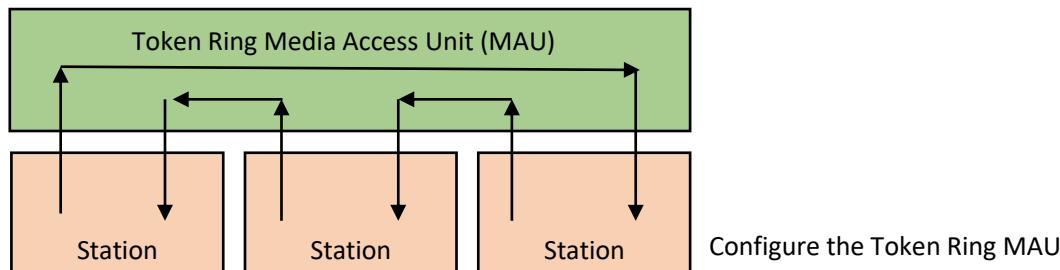


Understanding the network terms

Token Ring

Networks are resource-constrained, and research has continued on how to efficiently deliver data to multiple users and resolve conflicts that arise in the process. You will explore one of the technologies that fell into disuse with the development of Ethernet in the early days, which was standardized in the form of 802.5.

- What is a Token Ring?
 - An IEE 802.5 standardized model using LAN technology.
 - A form of network communication in which multiple stations are connected in a ring, with tokens moving around and communicating only with a specific station.
- How Token Ring works
 - A single control token of 3 bytes is created
 - The token is cycled in one direction
 - A tokenized branch office has access to network communications.



Understanding the network terms

CSMA/CD

We will learn about Ethernet and how to identify solutions when this network communication method causes conflicts during data transmission.

- What is Ethernet?
 - Conflicts occur when multiple hosts share a single media/link and access it at the same time.
 - This leads us to commit ourselves to follow rules to communicate seamlessly when a link is shared, and this networking method is called Ethernet.
 - The link is typically shared in a bus-type topology.
 - Protocol that uses a randomized method to access a shared link
 - ALOHA, CSMA, CSMA/CD, CSMA/CA

Understanding the network terms

CSMA/CD

We will learn about Ethernet and how to identify solutions when this network communication method causes conflicts during data transmission.

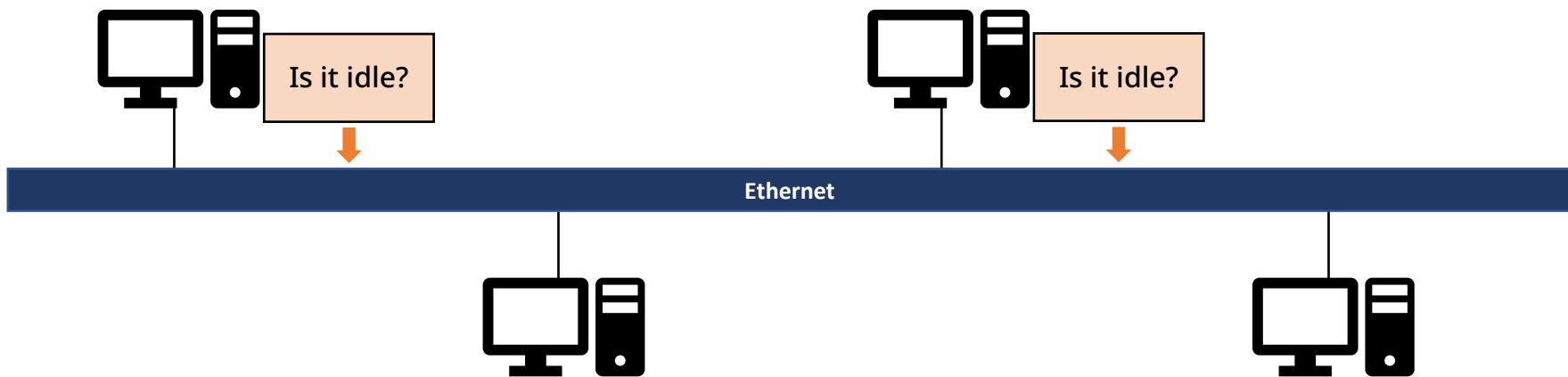
- What is CSMA?
 - Carrier Sense Multiple Access (CSMA)
 - How each station checks for status before sending
 - Send the frame immediately if it is idle due to the time difference between the frames being sent.
 - Collision and collision domain
 - Collision : when networks collide with each other on connected segments.
 - Collision domain : the band or segment of the network where collisions can occur.

Understanding the network terms

CSMA/CD

We will learn about Ethernet and how to identify solutions when this network communication method causes conflicts during data transmission.

- CSMA collision example
 - Make sure that the station(s) you want to send to are idle at the same time.

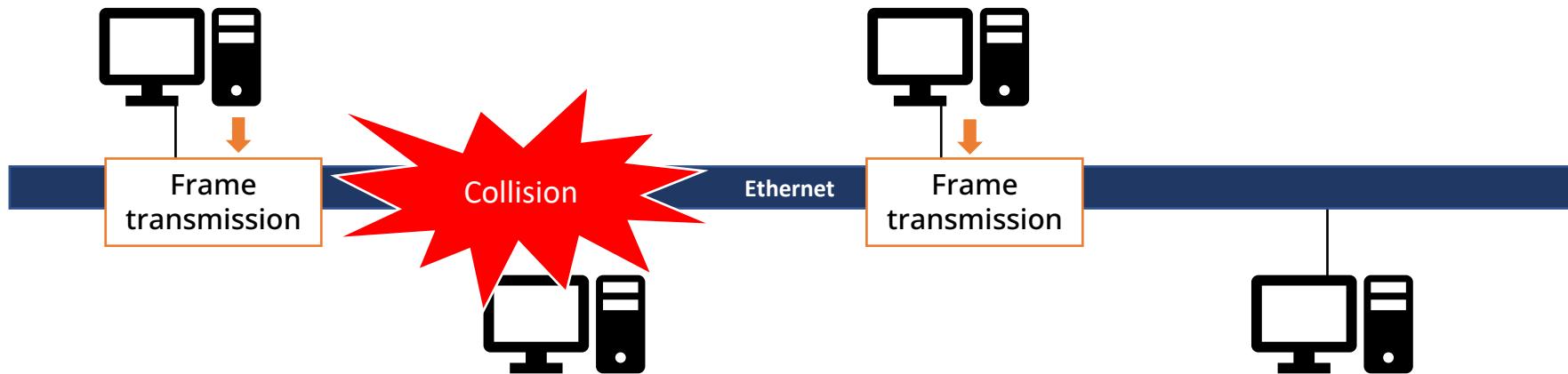


Understanding the network terms

CSMA/CD

We will learn about Ethernet and how to identify solutions when this network communication method causes conflicts during data transmission.

- CSMA collision example
 - Acknowledging a pause and sending a frame at the same time causes a crash.



Understanding the network terms

CSMA/CD

We will learn about Ethernet and how to identify solutions when this network communication method causes conflicts during data transmission.

- How to avoid CSMA collisions

- 1-persistent

- Continuously checks the status of the line and sends frames immediately if it is found to be idle
 - High collision risk because two or more stations are more likely to detect a line break and send frames immediately

- nonpersistent

- If the base station detects the line and sends a frame, but the line is busy, it waits a random amount of time before detecting again.
 - Two or more stations are unlikely to wait the same amount of time and send at the same time.

- p - persistent

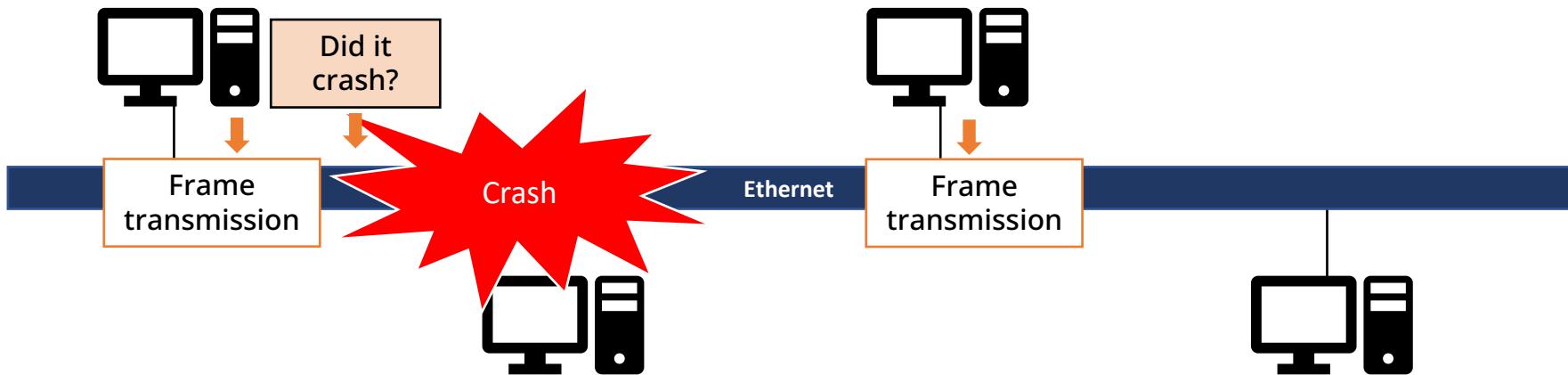
- Select the probability (p) that a line will go idle while continuously scanning for a line.
 - $q = 1 - p$
 - Send frames immediately if $p = 1$
 - If p is not 1, wait as long as q before re-detecting.

Understanding the network terms

CSMA/CD

We will learn about Ethernet and how to identify solutions when this network communication method causes conflicts during data transmission.

- CSMA/CD
 - CSMA method plus collision handling procedures because it can check for collisions.
 - Use two different ports at the same time to transmit frames, detect collisions, and time retransmissions.

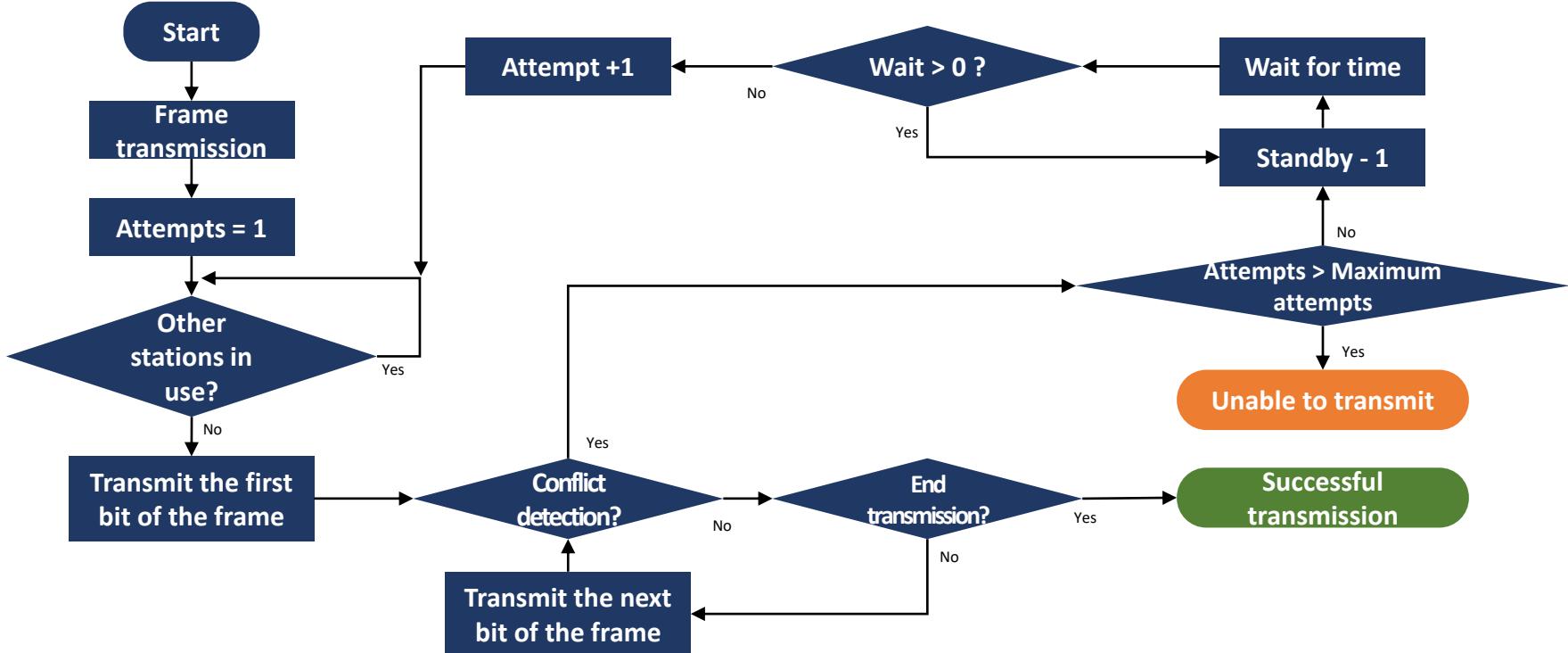


Understanding the network terms

CSMA/CD

We will learn about Ethernet and how to identify solutions when this network communication method causes conflicts during data transmission.

- How to Detect CSMA/CD Collisions



Understanding the network terms

CSMA/CA

We will learn about Ethernet and how to identify solutions when this network communication method causes conflicts during data transmission.

- What is CSMA/CA?
 - Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)
 - The base station must be receiving signals at the same time it is transmitting signals to detect collisions
 - Wireless networks have difficulty adopting CSMA/CD because of the difficulty in detecting collisions.
 - Avoid collisions by prioritizing frames
 - Types of collision avoidance methods
 - Inter Frame Space (IFS) : space between frames
 - A form of collision avoidance that delays transmission even when it detects that the channel is idle.
 - Contention Window : contention zone
 - Select a random number of stations that are ready to transmit.
 - Wait for the IFS time plus an arbitrary backoff time, and determine priority within the backoff timer.

Understanding network models

Abstraction

Abstraction is one of the most important concepts in IT. Networks are no exception, and the following network model is characterized by abstraction. Therefore, you will understand the abstractions and apply them to what comes next.

- Dictionary meanings
 - A way of distilling key concepts or features from complex materials, modules, systems, etc.
 - E.g.,



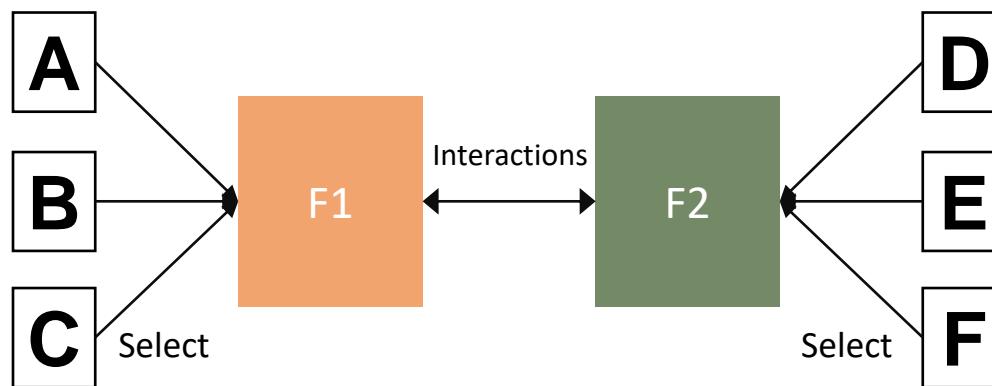
While the actions of smartphones are similar to each other, the inner workings of those actions are all different, and the goal of abstraction is to allow users to use them without knowing how they work.

Understanding network models

Abstraction

Abstraction is one of the most important concepts in IT. Networks are no exception, and the following network model is characterized by abstraction. Therefore, you will understand the abstractions and apply them to what comes next.

- Understanding the advantage of abstraction with examples



- A, B, and C work differently in different environments and with different technologies, and F1 is self-sufficient.
- D, E, and F work differently in different environments and with different technologies, and F2 is self-sufficient.
- F1 and F2 interact with each other to use any part (A,B,C or D,E,F).
- If a specific part fails and is replaced by A >> B, the interaction between F1 and F2 is not affected.

Understanding network models

Types of network models

Networks communicate with each other using agreed-upon protocols and are characterized by layers based on communication methods and principles. Examine and understand two representative models.

- Network model
 - Classification of network communication principles that led to the development of the Internet and subsequent standardization.
 - Typical abstraction models are the hierarchical OSI 7-layer and TCP/IP models.
 - Layer architecture and abstraction are characterized by several features.
 - Each layer has an independent role.
 - Tiered management makes maintenance more efficient
 - Expect diverse software
 - Relatively easy to understand
 - Reduce complexity

Understanding network models

Types of network models

The OSI reference model provides a structure for easily understanding and decomposing complex networks into usable components and a hierarchy of abstracted components that perform each specific function of networking. It has the strengths of model-specific abstraction, and the distinction depends on how you use it.

- Open System Interconnection (OSI) 7-layer reference model
 - Features
 - Model developed by the International Organization for Standardization (ISO)
 - A layered model of computer network protocol design and communication
 - A seven-layer structure
 - History
 - A model created by Charles Bachman of Honeywell Information Systems
 - ISO published it as standard ISO7498 in 1984.
 - CCITT (now ITU-T) published it as the X.200 standard.

Understanding network models

Types of network models

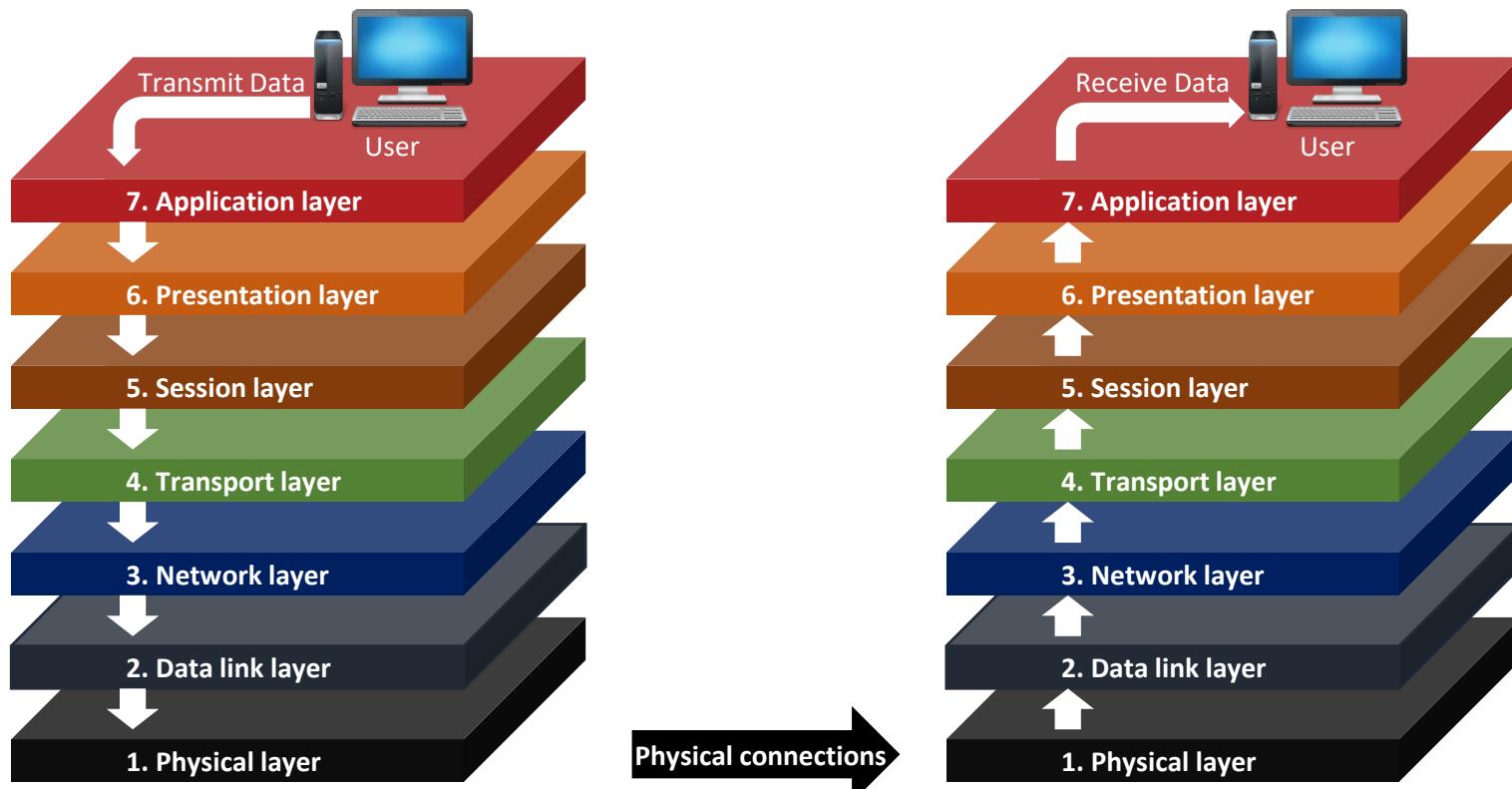
The OSI reference model provides a structure for easily understanding and decomposing complex networks into usable components and a hierarchy of abstracted components that perform each specific function of networking. It has the strengths of model-specific abstraction, and the distinction depends on how you use it.

- TCP/IP essentials
 - Features
 - TCP/IP is a model that consists of IP and TCP, with a large number of application protocols based on TCP operating on top of IP.
 - It has four layers and is currently the most commonly used form.
 - Sometimes categorized into five layers depending on the book or resource
 - History
 - In 1982, the TCP/IP specification developed by ARPNET was finalized, and in 1983, ARPNET designated the protocol as official.
 - Since the 1990s, utilized only in universities and research institutions, but has since been activated globally
 - Early TCP/IP models were utilized for research purposes, making the Internet accessible to the general public.

The OSI models

The 7 layers of OSI

The OSI 7-layer model architecture provides a structure for easily understanding complex network structures and breaking them down into usable components. This also offers modularization components that represent network functions as abstractions.

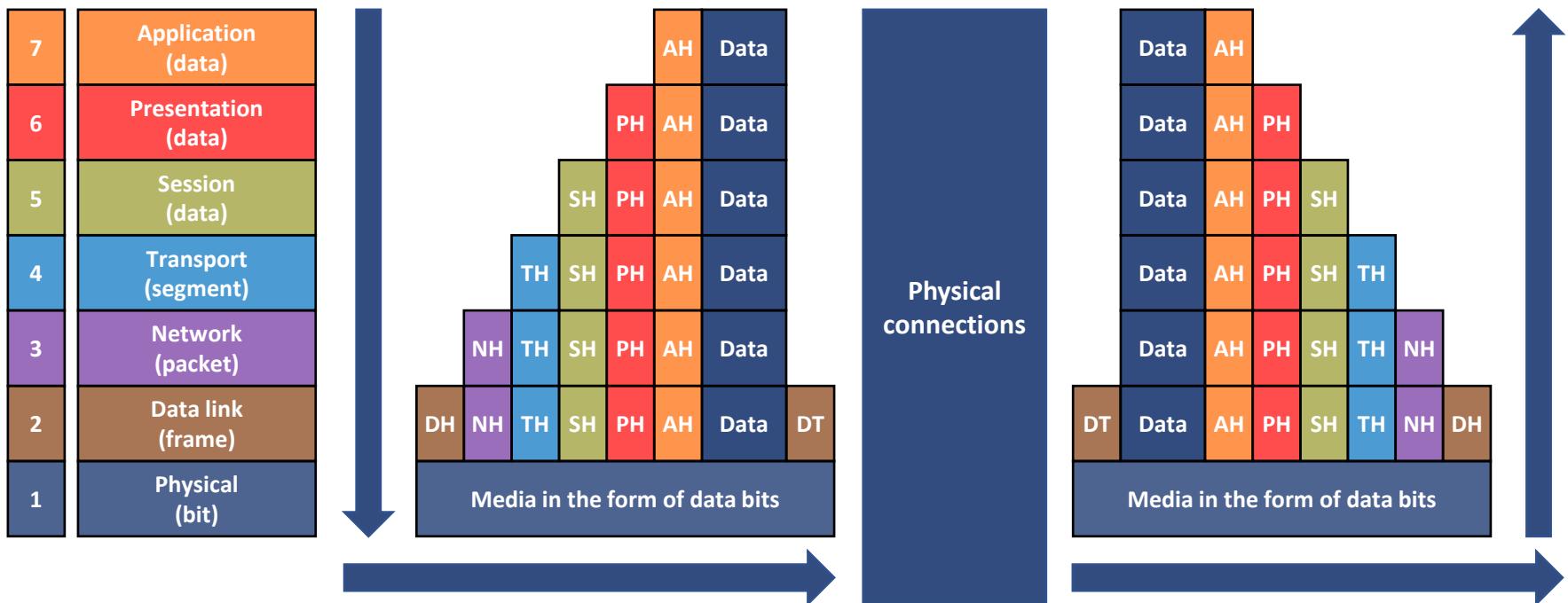


The OSI models

OSI 7-layer architecture

The OSI 7-layer model architecture provides a structure for easily understanding complex network structures and breaking them down into usable components. This also offers modularization components that represent network functions as abstractions.

- OSI 7-layer model diagram
 - Layers 1-3 : media layer / Layers 4-7 : host layer

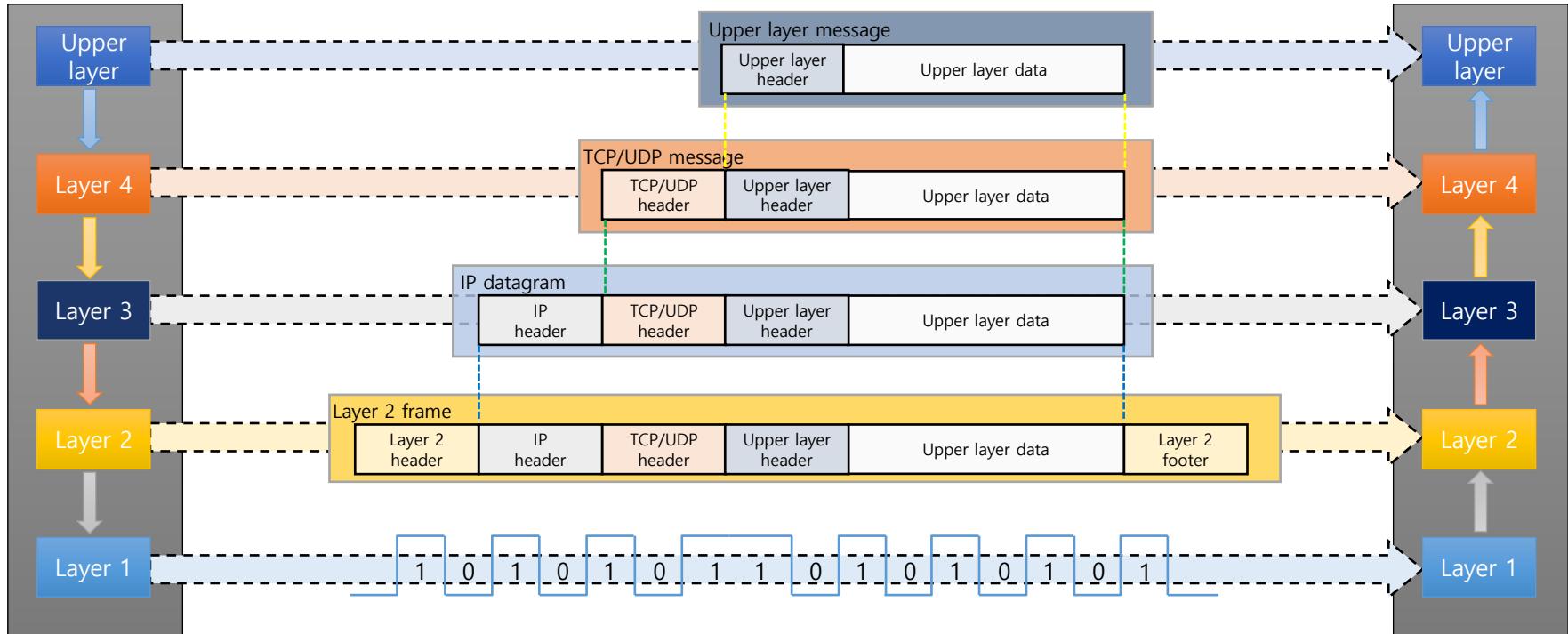


The OSI models

OSI 7-layer architecture

Encapsulation means wrapping data and header information used by a higher layer with protocol information from a lower layer.

- Encapsulation



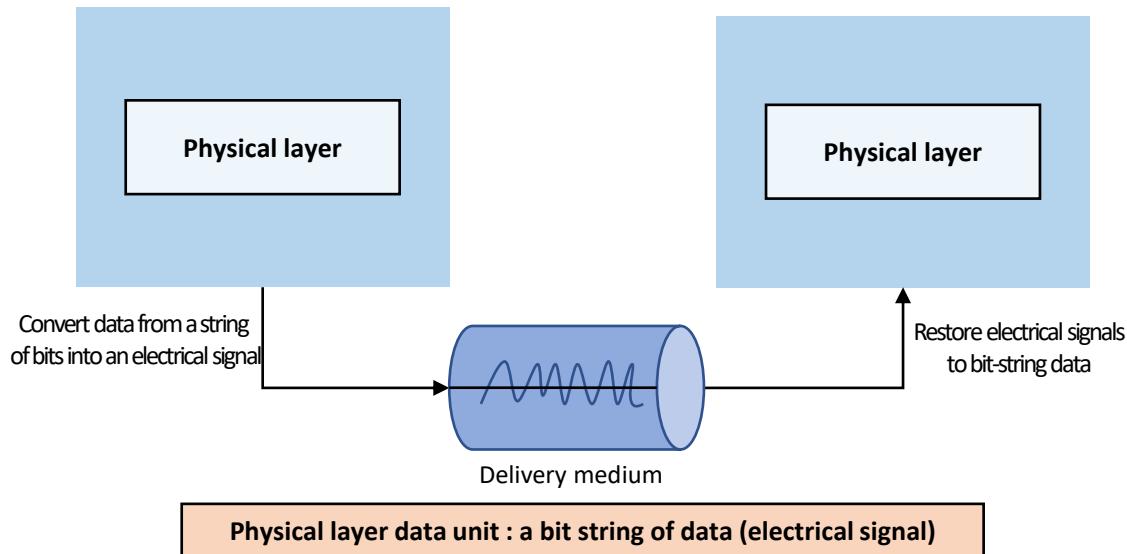
The OSI models

Physical layer

The physical layer of the seven OSI layers, which processes digital signals in bits, is directly related to the processing of electrical signals. We will gain an understanding of communication principles at the physical layer and familiarize yourself with the equipment involved.

- Features

- Establish, maintain, and cut physical links for data transfer
- Define voltage levels, conversion timing, and maximum amount of physical data transferred, transfer distance, etc.



The 7 layers of the OSI

| | |
|---|------------------------|
| 7 | Application (data) |
| 6 | Presentation (data) |
| 5 | Session (data) |
| 4 | Transport (segment) |
| 3 | Network (packet) |
| 2 | Data link (frame) |
| 1 | Physical (bit) |

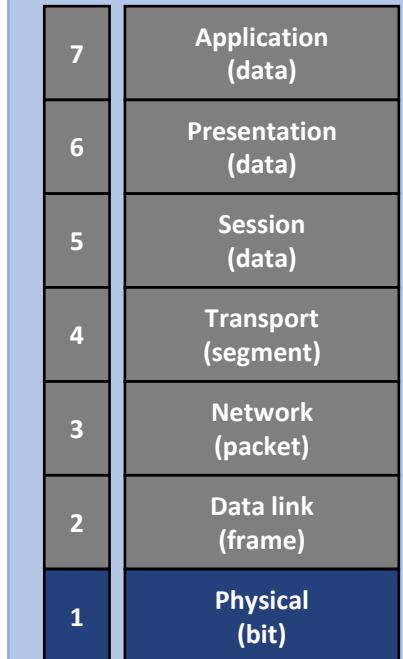
The OSI models

Physical layer

The physical layer of the seven OSI layers, which processes digital signals in bits, is directly related to the processing of electrical signals. We will gain an understanding of communication principles at the physical layer and familiarize yourself with the equipment involved.

- Type (hardware)
 - Coaxial cable, fiber-optic cable
 - Adapters, modems, repeaters (or repeaters), and hubs
- Type (software unit)
- Unit : bit
- Transport mode
 - Unidirectional : transmits unilaterally in only one direction (e.g., TV, Radio).
 - Half-duplex : allows two-way communication, but cannot transmit or receive at the same time (e.g., walkie-talkies)
 - Full duplex : allows real-time communication in both directions (e.g., phone)

The 7 layers of the OSI

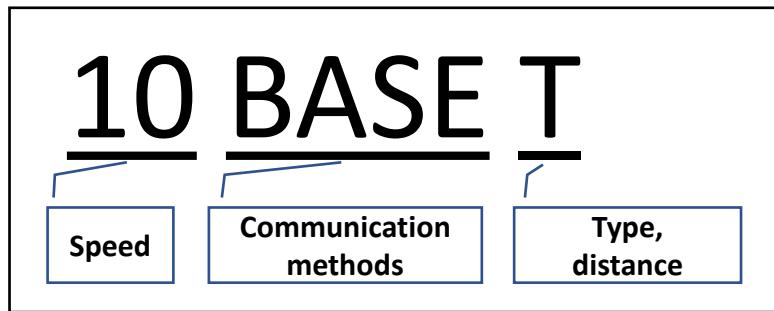


The OSI models

Physical layer

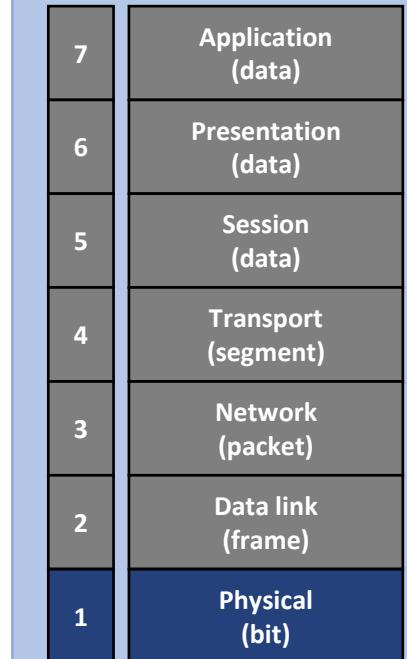
Cables are the quintessential communication devices within the physical layer. We will learn about cable types and understand the different types of cables and how we distinguish between them.

- Cables



- Speed
 - The front number is in megabits per second (Mbps)
 - Different from BPS (Byte Per Second)
- Communication methods
 - Divided into baseband (digital) and broadband (analog)

The 7 layers of the OSI



The OSI models

Physical layer

Cables are the quintessential communication devices within the physical layer. We will learn about cable types and understand the different types of cables and how we distinguish between them.

- Cables

- Cable types

| Cable type | Speed | Distance | Topology | Initial standards | Description |
|--------------|----------|---|----------|----------------------|--|
| 10 Base T | 10 Mbps | 100 M (Half/Full) | Star | 802.3i-1990 | Use Category 3,4,5, RJ45 |
| 10 Base FL | 10 Mbps | 2 KM (Half) Over 2 KM (Full) | Star | 802.3j-1993 | Fiber-optic |
| 10 Base 2 | 10 Mbps | 185 M (Half) | Bus | 802.3a-1985 | Thin cable with BNC connector |
| 10 Base 5 | 10 Mbps | 500 M (Half) | Bus | DIX-1980, 802.3-1983 | Called thick cable or yellow cable Setup for centralized networks |
| 100 Base TX | 100 Mbps | 100 M (Half/Full) | Star | 802.3u-1995 | Use a Category 5 UTP cable |
| 100 Base T2 | 100 Mbps | 100 M (Half/Full) | Star | 802.3z-1998 | Enable all Categories 3,4,5 |
| 100 Base T4 | 100 Mbps | 100 M (Half) | Star | 802.3u-1995 | Use Category 3 & all 4 pairs of 8 strands |
| 1000 Base SX | 1 Gbps | 275 M (Half/Full) 316 M (Half), 550 M (Full) | Star | 802.3z-1998 | Use a short wave length laser |
| 1000 Base T | 1 Gbps | 100 M (Half/Full) | Star | 802.3ab-1999 | Use all 4 pairs (8 strands) of Category 5 |

The OSI models

Physical layer

Cables are the quintessential communication devices within the physical layer. We will learn about cable types and understand the different types of cables and how we distinguish between them.

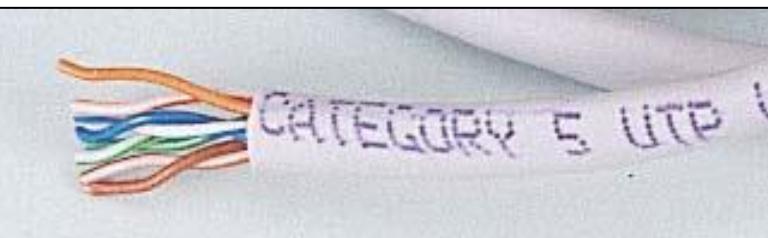
- Features by category
 - Different categories of UTP cables have different characteristics
 - Descriptions by taxonomy
 - Category 1 : used for traditional voice-grade phone lines only
 - Category 2 : guaranteed data transfer rates around 4 Mbps
 - Category 3 : guaranteed voice-grade data rates around 16 Mbps
 - Category 4 : guaranteed data transfer rates around 20 Mbps
 - Category 5 : Category 5e version is more popular as a quick replacement for Category 4
 - Category 5e : 100 Mps and above to ensure data rates more suitable for 1 Gbps applications
 - Category 6 : support 1000 Base T by using 4 pairs
 - Currently Categories 1, 2, 4, and 5 are no longer valid as standards in ANSI/TIA 568 C.

The OSI models

Physical layer

Cables act as a transmission medium to carry signals between computers, and while there are many types of cables, coaxial, twisted pair, and fiber-optic cables are commonly used.

- Name : 10/100 BASE-T
- Connector : RJ-45
- Connectivity : 100 M
- Types :
 - Direct cable
 - Crossover cable
 - Rollover cable



The OSI models

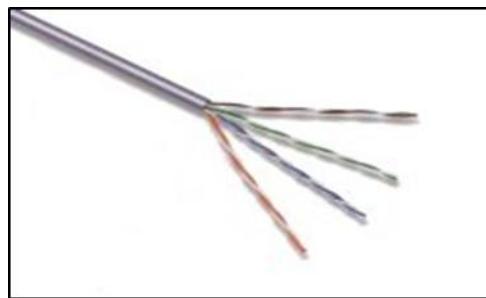
Physical layer

Cables act as a transmission medium to carry signals between computers, and while there are many types of cables, coaxial, twisted pair, and fiber-optic cables are commonly used.

- Physical layer



LAN tool



UTP cable



RJ-45 jack



Nipper



Stripper



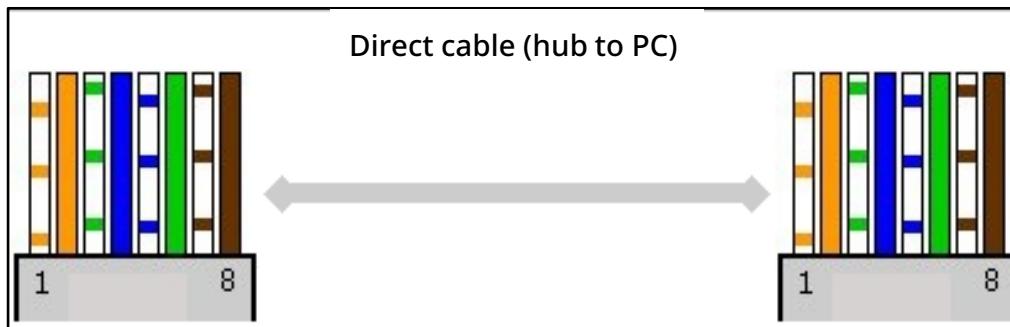
UTP tester

The OSI models

Physical layer

Cables act as a transmission medium to carry signals between computers, and while there are many types of cables, coaxial, twisted pair, and fiber-optic cables are commonly used.

- Direct cable
 - The order of the colored lines is the same at each end, and the 1:1 cable
 - Send-to-send and receive-to-receive inside the hub
 - Called a serial cable
 - Color code order : white/orange - orange - white/green - blue - white/blue - green - white/brown - brown
 - Use : PC to hub connection



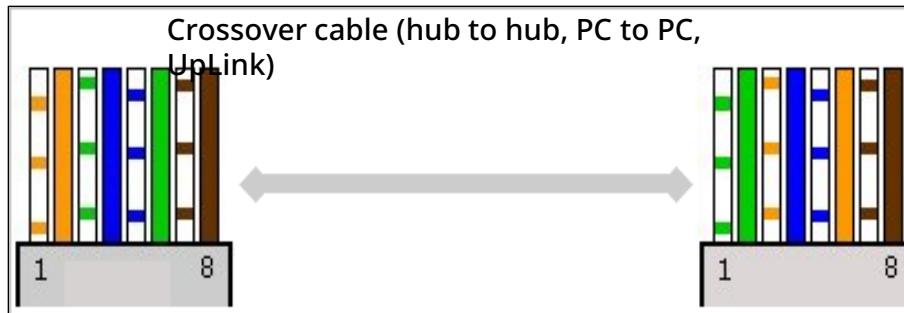
The OSI models

Physical layer

Cables act as a transmission medium to carry signals between computers, and while there are many types of cables, coaxial, twisted pair, and fiber-optic cables are commonly used.

- Crossover cable

- Connect RJ-45 end pin 1 to pin 3 on the other end, pin 2 to pin 6 on the other end.
- Replace the receive-send and send-receive work of a hub with a twisted pair of cables.
- Color code order: white/orange - orange - white/green - blue - white/blue - green - white/brown - brown and white/green - green - white/orange - blue - white/blue - orange - white/brown - brown
- Use : PC to PC (same model) connection



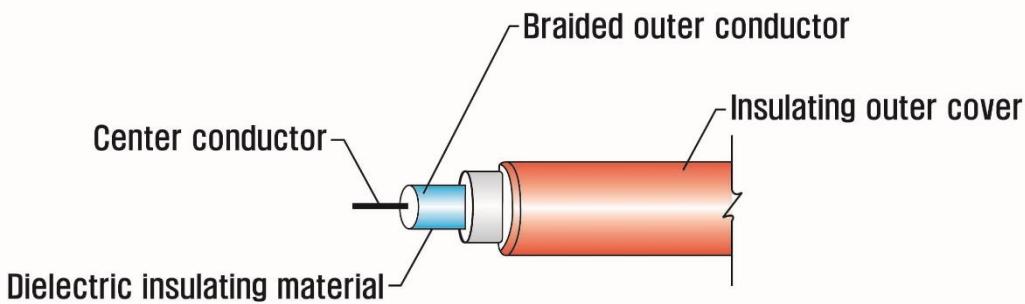
The OSI models

Physical layer

Cables act as a transmission medium to carry signals between computers, and while there are many types of cables, coaxial, twisted pair, and fiber-optic cables are commonly used.

- Coaxial cable

- Due to its structural characteristics, it is well shielded from the outside world and has less interference.
- Better frequency characteristics than double helix, enabling faster data transfer at higher frequencies
- Enable high-speed transfers of hundreds of Mbps
- Low power dissipation
- Types include thin-net, thick-net.



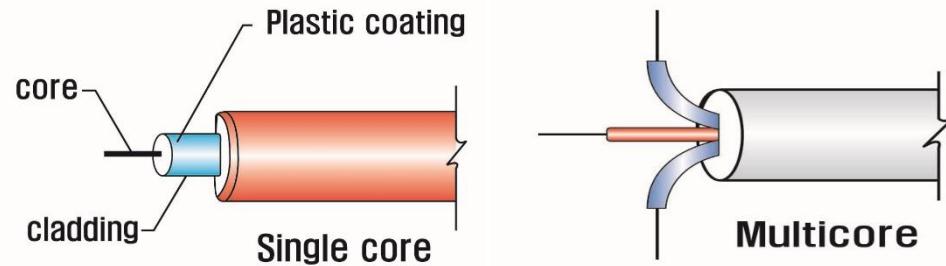
The OSI models

Physical layer

Cables act as a transmission medium to carry signals between computers, and while there are many types of cables, coaxial, twisted pair, and fiber-optic cables are commonly used.

- Fiber-optic cable

- Mainly used for backbone purposes because the band of frequencies is wider than electromagnetic waves
- Abundant glass resources for the core
- Smaller, lighter, and more bend-resistant than traditional copper wires to carry information
- Structure of a fiber-optic cable: core, cladding, coating
- Types : single-mode fiber, multimode fiber



The OSI models

Physical layer

At the physical layer, a hub is the main device that distributes digital signals. We will learn how hubs work to share data between devices and see how they differ.

- Hubs
 - Overview
 - Devices that combine the roles of multiport and repeater
 - Repeater : a device that amplifies the signal sent by a device to the limit of distance transmission.
 - Follow CSMA/CD method
 - How it works
 - Hubs send data to connected networks to communicate with each other.
 - Devices that receive data discard data that is not relevant to them.
 - If the device that received the data relates to you, accept it.

The 7 layers of the OSI

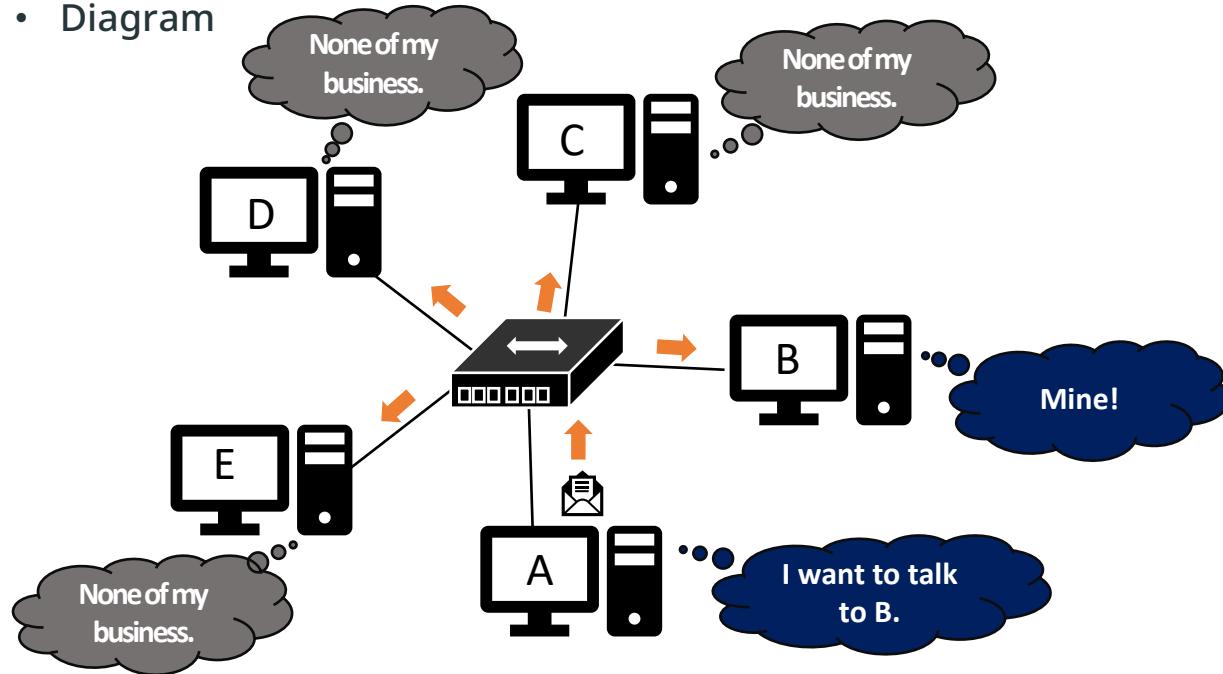
| | |
|---|------------------------|
| 7 | Application (data) |
| 6 | Presentation (data) |
| 5 | Session (data) |
| 4 | Transport (segment) |
| 3 | Network (packet) |
| 2 | Data link (frame) |
| 1 | Physical (bit) |

The OSI models

Physical layer

At the physical layer, a hub is the main device that distributes digital signals. We will learn how hubs work to share data between devices and see how they differ.

- Hubs
 - How it works
 - Diagram



The 7 layers of the OSI

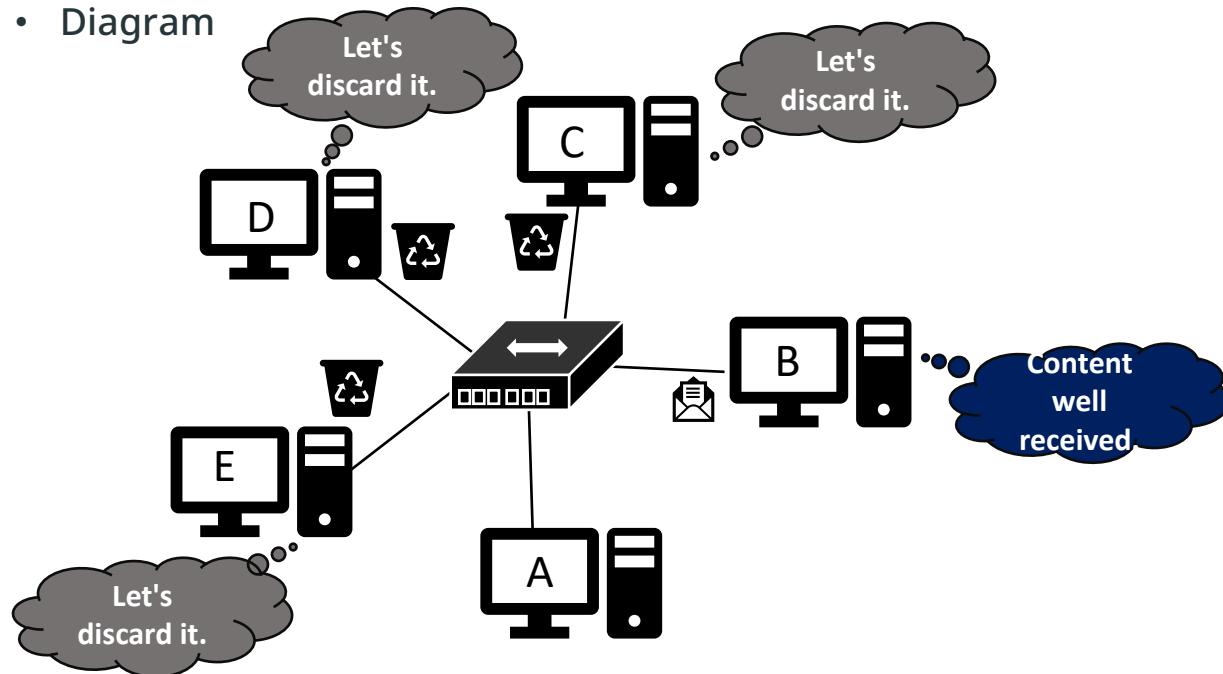
| | |
|---|------------------------|
| 7 | Application (data) |
| 6 | Presentation (data) |
| 5 | Session (data) |
| 4 | Transport (segment) |
| 3 | Network (packet) |
| 2 | Data link (frame) |
| 1 | Physical (bit) |

The OSI models

Physical layer

At the physical layer, a hub is the main device that distributes digital signals. We will learn how hubs work to share data between devices and see how they differ.

- Hubs
 - How it works
 - Diagram



The 7 layers of the OSI

| | |
|---|------------------------|
| 7 | Application (data) |
| 6 | Presentation (data) |
| 5 | Session (data) |
| 4 | Transport (segment) |
| 3 | Network (packet) |
| 2 | Data link (frame) |
| 1 | Physical (bit) |

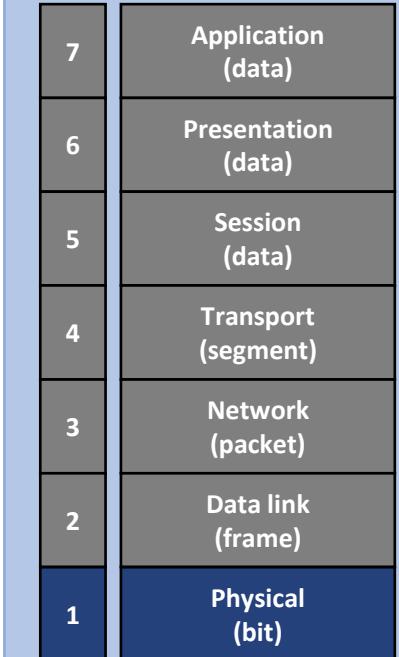
The OSI models

Physical layer

At the physical layer, a hub is the main device that distributes digital signals. We will learn how hubs work to share data between devices and see how they differ.

- Hubs
 - Types
 - Intelligence hub
 - Can be controlled through a Network Management System(NMS)
 - Analyze, control, and monitor data
 - Block devices that consistently detect crashes
 - Semi-dummy hubs
 - A dummy hub and an intelligence hub connected.
 - Act as an intelligence hub
 - Dummy hubs
 - Common hubs that don't offer network management features

The 7 layers of the OSI



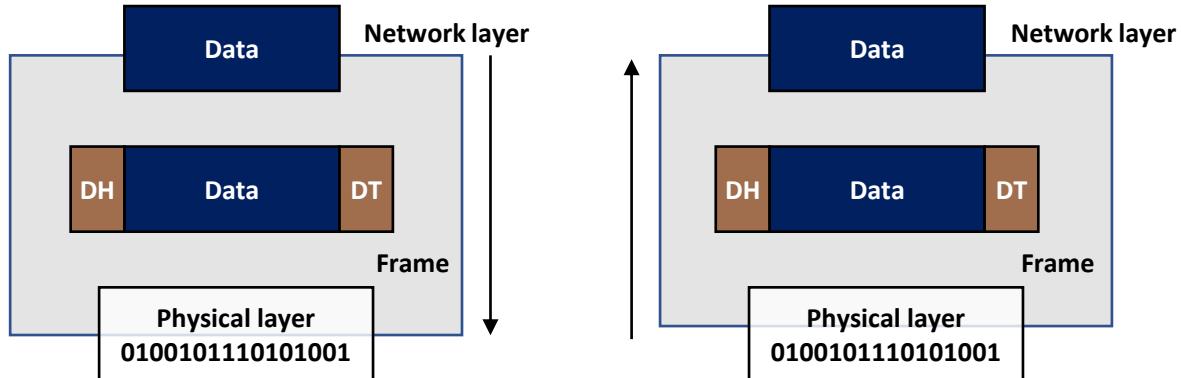
The OSI models

Data link layer

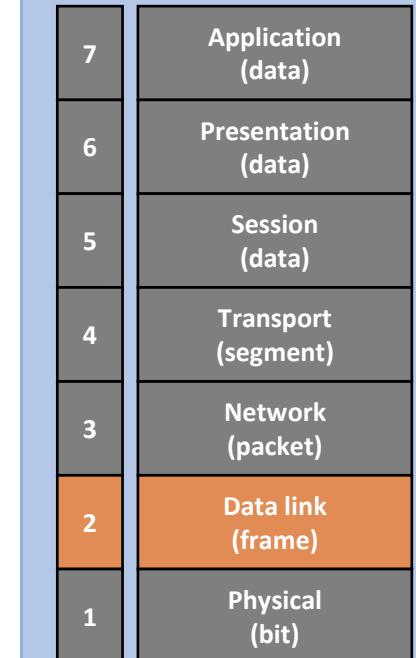
The data link layer is where most LAN and wireless LAN technologies are defined. The data link layer is where most LAN and wireless LAN technologies are defined. Layer 2 handles logical link control, medium access control , hardware addressing, error detection and handling, and the definition of physical layer standards.

- Features

- Communicate between hardware based on physical addresses
- Flow control to ensure a speed differential between sender and receiver
- Error control to compensate for errors due to errors and noise



The 7 layers of the OSI



The OSI models

Data link layer

The data link layer is where most LAN and wireless LAN technologies are defined. The data link layer is where most LAN and wireless LAN technologies are defined. Layer 2 handles logical link control, medium access control , hardware addressing, error detection and handling, and the definition of physical layer standards.

- Functions

- A layer to ensure reliable point-to-point transmissions, providing CRC-based error control, flow control, and retransmission capabilities
- A terminal structure that uses MAC addresses and has no hierarchy in the addressing scheme itself

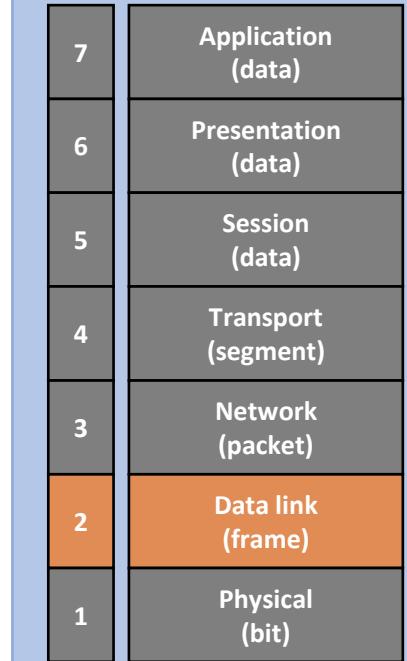
- Protocols

- Uses primarily Ethernet
- Point-to-point : HDLC, ADCCP
- For local area networks : LLC, ALOHA

- Types (hardware)

- Switches, bridges

The 7 layers of the OSI



The OSI models

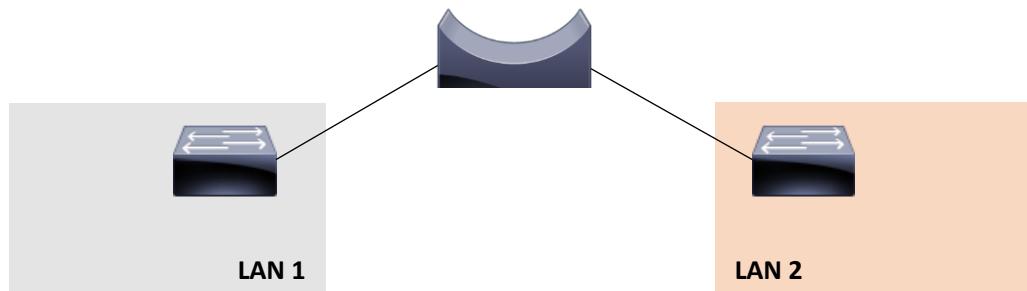
Data link layer

The data link layer is where most LAN and wireless LAN technologies are defined. The data link layer is where most LAN and wireless LAN technologies are defined. Layer 2 handles logical link control, medium access control , hardware addressing, error detection and handling, and the definition of physical layer standards.

- Bridges

- Overview

- When only two PCs connected to the hub communicate, other PCs cannot communicate at the same time. (CSMA/CD method is the cause)
 - Serves as a connection from another network (LAN)
 - Processes information from the data in the frame and uses the MAC address



The 7 layers of the OSI

| | |
|---|------------------------|
| 7 | Application (data) |
| 6 | Presentation (data) |
| 5 | Session (data) |
| 4 | Transport (segment) |
| 3 | Network (packet) |
| 2 | Data link (frame) |
| 1 | Physical (bit) |

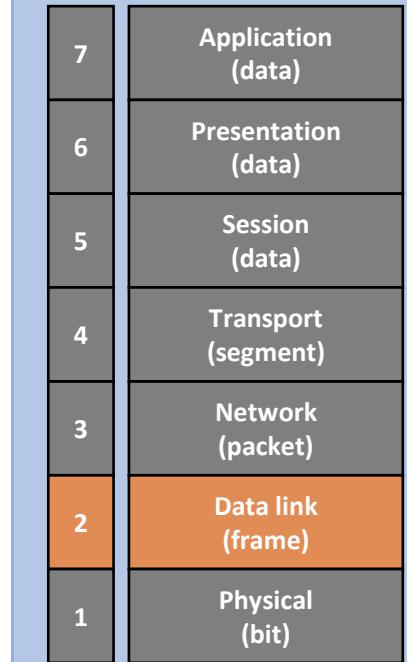
The OSI models

Data link layer

A switch is a typical Layer 2 network device that uses a MAC address and is used to organize a LAN.

- L2 switches
 - Overview
 - Divides a LAN into separate collision domains
 - Typical Layer 2 network devices that organize LANs based on MAC addresses.
 - Functions
 - Dedicated bandwidth : delivers bandwidth only to that host via MAC address
 - Full-duplex operation : uses full-duplex communication (two-way communication) method
 - Features
 - Multiple switch ports exist
 - Many frame buffers
 - Allows mixing of speeds between ports
 - Quick switching features
 - Low cost per port

The 7 layers of the OSI

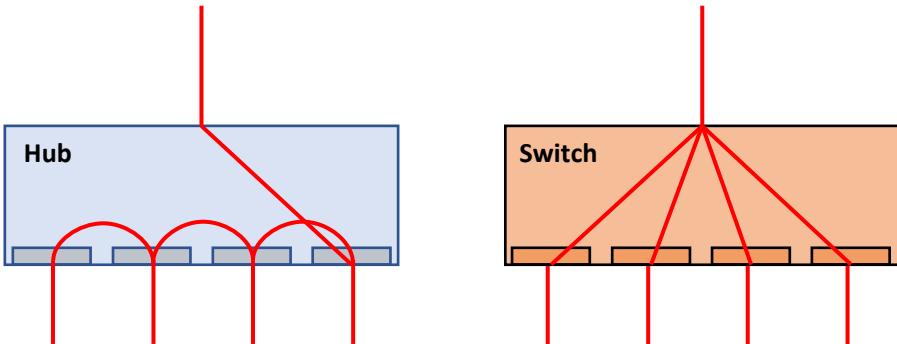


The OSI models

Data link layer

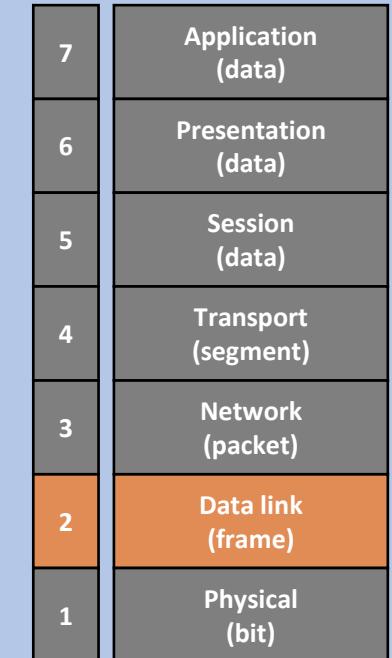
While switches and hubs share the common role of connecting multiple computers, they work on different principles.

- Switches and hubs?
 - Switch and Hub Configurations



- A hub sends data by splitting bandwidth between lines.
A switch shares bandwidth between all 5 computers.
- A switch has a CPU and memory.
A hub is a simple distribution device, like an electrical multiple tap.

The 7 layers of the OSI



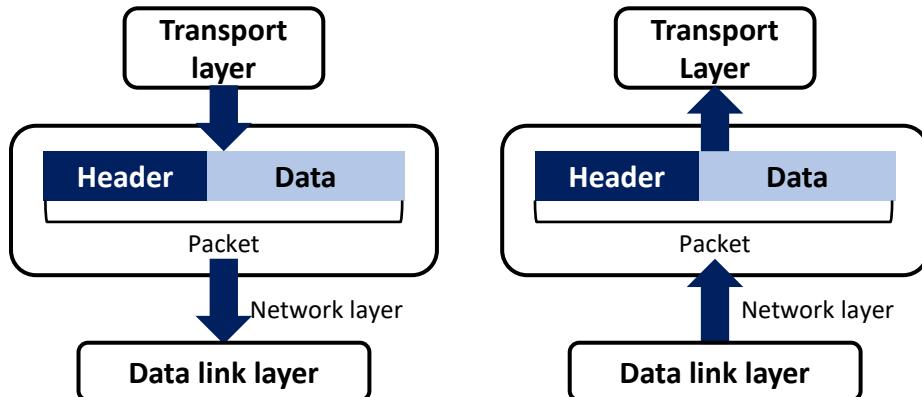
The OSI models

Network layer

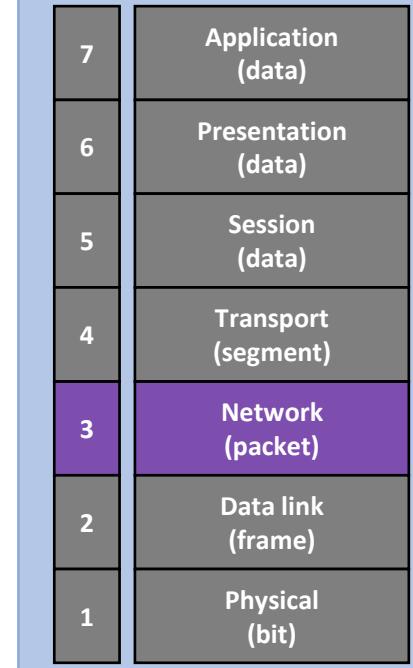
Network layer functions include inter-network level addressing, routing, datagram encapsulation, segmentation and recombination, and certain types of error handling and diagnostics. The network and transport layers are closely related.

- Features

- Provide the necessary data transfer and path selection to connect to higher layers
- Responsibility for the delivery of each packet from the originating host to the destination host
- Use routing protocols to choose the best route
- Split data into packets and recombine them after transmission



The 7 layers of the OSI



The OSI models

Network layer

Network layer functions include inter-network level addressing, routing, datagram encapsulation, segmentation and recombination, and certain types of error handling and diagnostics. The network and transport layers are closely related.

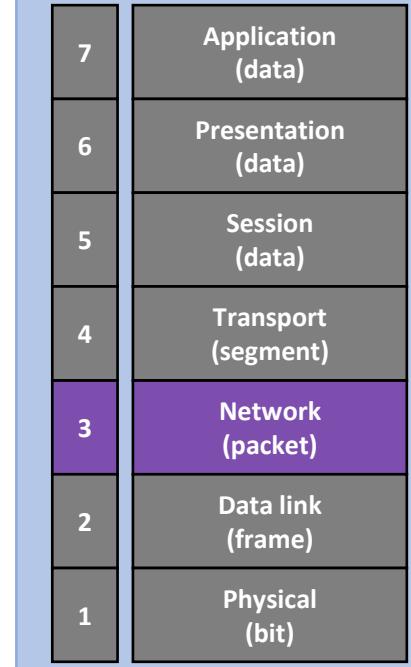
● Features

- Uses a unit of measure of packets, with a 32-bit address space scheme (based on IPv4).
- Usually has an addressing scheme used by router equipment and has IP, ICMP, and IGMP protocols.
- Due to the limited number of IPv4, it is trying to switch to the structure of IPv6, and it is trying to apply it to new smartphones after September 2014 in South Korea.
- As of December 2021, there are 345,469 IPv6 allocated for domestic use in South Korea / 20,071,520 for special use.

● Functions

- Logical addressing
- Routing networks
- Datagram encapsulation
- Fragmentation and recombination
- Error handling and diagnostics

The 7 layers of the OSI



The OSI models

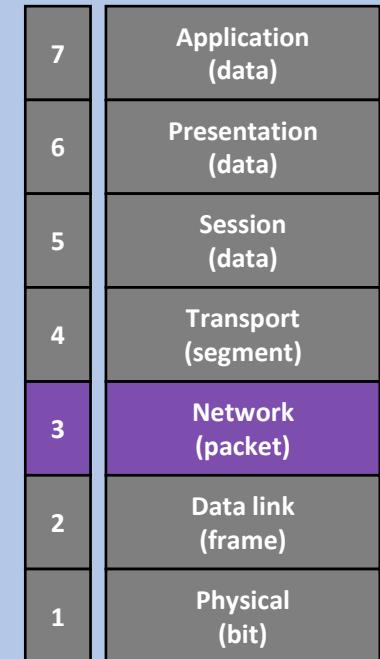
Network layer

Network layer functions include inter-network level addressing, routing, datagram encapsulation, segmentation and recombination, and certain types of error handling and diagnostics. The network and transport layers are closely related.

- Functions of the network layer

| Function | Description |
|---------------------------------|--|
| Logical addressing | All devices communicating on a network have a logical address that identifies them, regardless of their physical location. In the case of the Internet, the network layer protocol is the Internet Protocol (IP), and each device has an IP address. The logical address is independent of hardware characteristics and must be unique throughout the Internetworks. |
| Routing networks | The function that defines the hierarchy in a nutshell is routing. It is the job of the equipment and software operating at the network layer to receive packets coming in from different places, identify their final destination, and determine the next path they need to take. |
| Datagram encapsulation | The network layer typically appends a network layer header to messages received from upper layers to encapsulate them and create datagrams (also called packets). |
| Fragmentation and recombination | The network layer wants to send the packet down to the data link layer for transmission. Some data link layer technologies limit the length of messages that can be sent. Therefore, if the data that the network layer wants to send is too large, the network layer must fragment the packet and send it to the data link layer, and the fragmented packet must be reassembled at the network layer of the destination device. |
| Error handling and diagnostics | The network layer uses specialized protocols to enable logically connected devices to exchange network or device status information. |

The 7 layers of the OSI



The OSI models

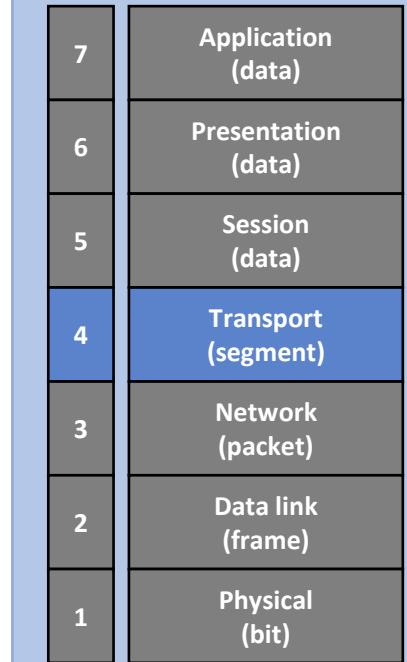
Transport layer

The transport layer is where the transition occurs between the lower layers, which deal with data delivery issues, and the upper layers, which deal with application software.

- Features

- Responsible for error control and flow control in the communication between the destination and the sender
- Establish connectivity between processes running on the host rather than on the network
- Control trusted communication connections
- Split data into packets and recombine them after transmission
 - Connectionless : treats each segment as an independent packet, forwarded to the transport layer of the destination system
 - Connection-oriented : Before forwarding a packet, it first establishes a connection with the destination system's transport layer of the destination system
- Use "segment" units

The 7 layers of the OSI

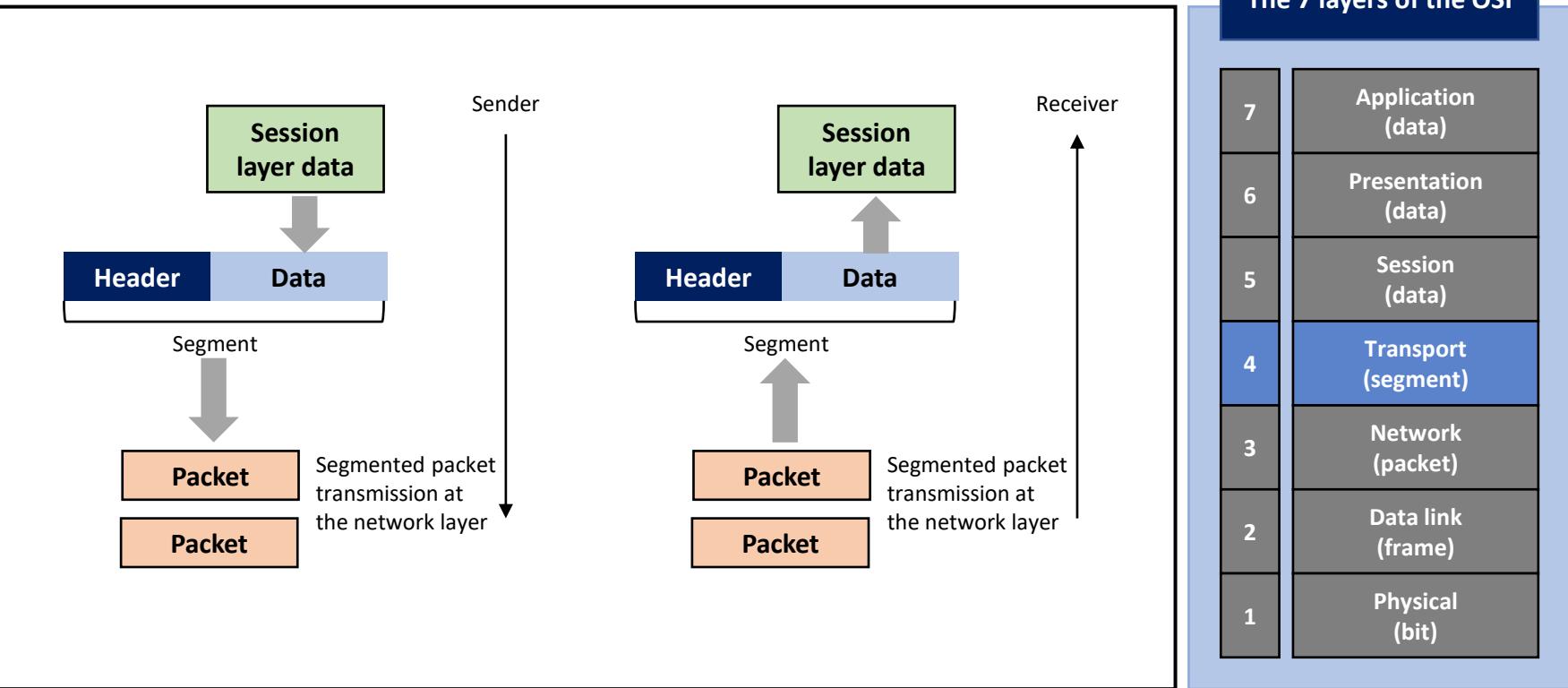


The OSI models

Transport layer

The transport layer is where the transition occurs between the lower layers, which deal with data delivery issues, and the upper layers, which deal with application software.

- Features



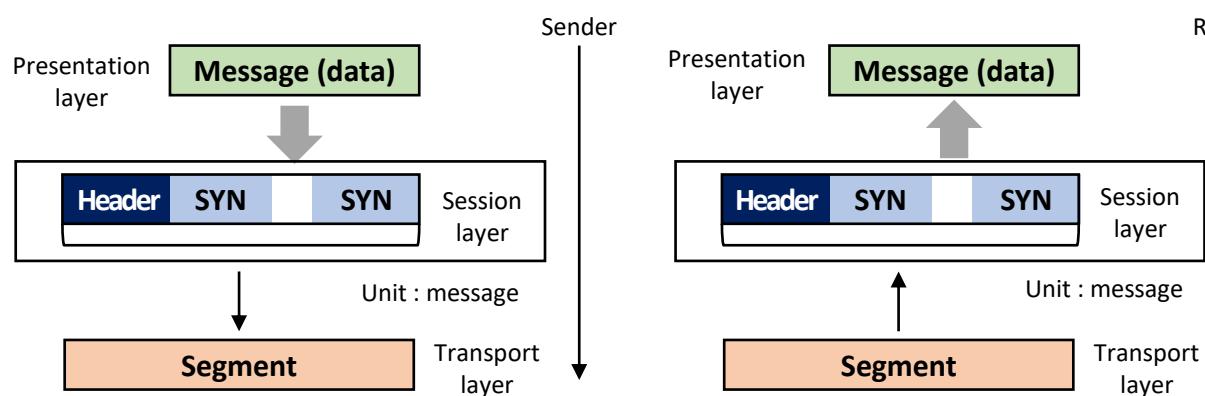
The OSI models

Session layer

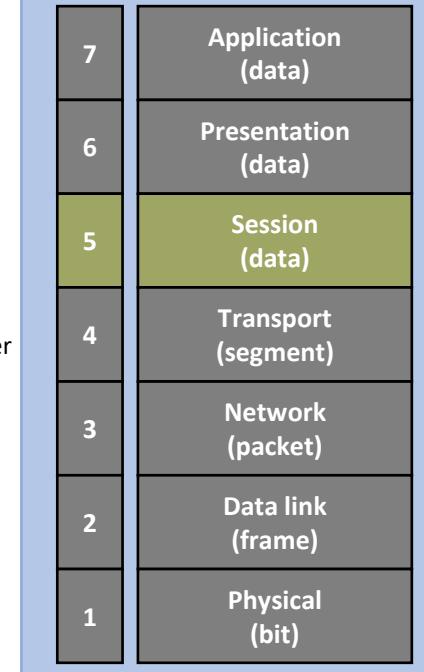
The session layer provides the ability to establish and manage sessions between software processes.

- Features

- Establish interaction between communicating devices using port connections
- Use SSH, TLS communication protocols
- Allow conversations between two systems
(determine who communicated when and exchange tokens)
- Provide a synchronization point for checking and recovering data



The 7 layers of the OSI



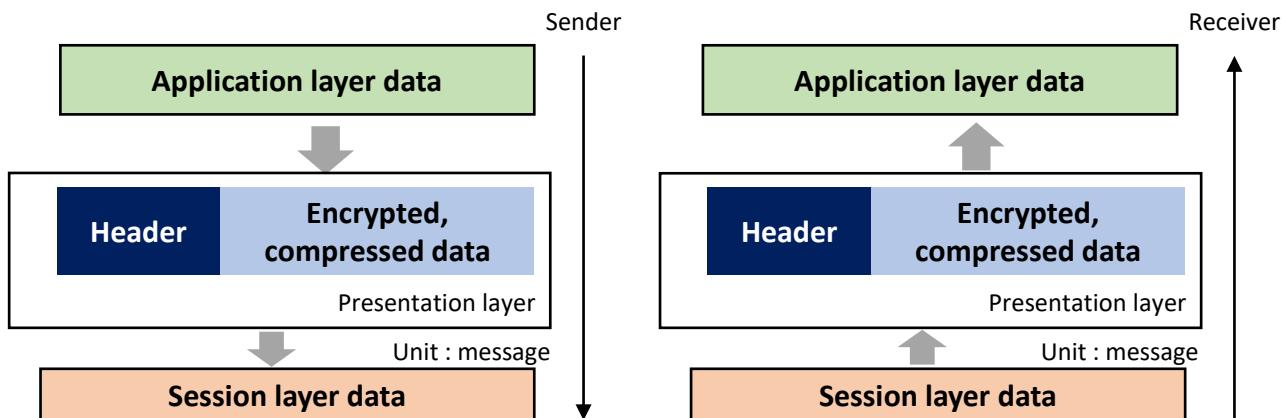
The OSI models

Presentation layer

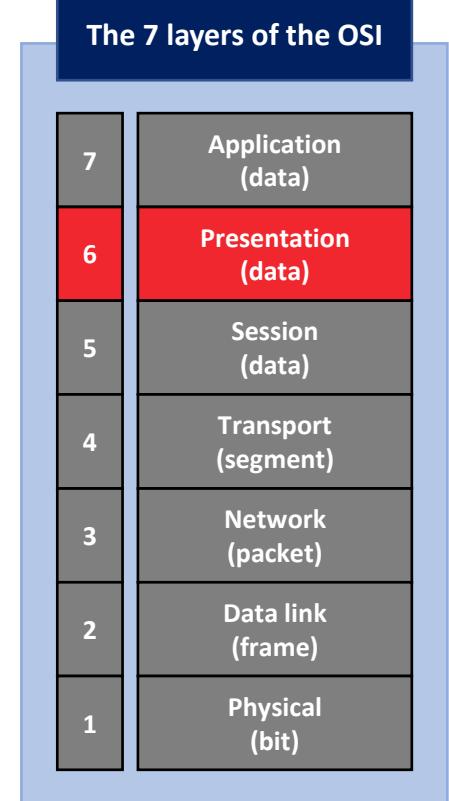
The presentation layer performs functions that transform the data so that it can be presented in other systems (translation, compression, encryption, etc.).

● Features

- Convert input or output data into a single presentation
- Protocols can be in the form of text or graphics, such as JPEG, MPEG, SMB, AFP, etc.
- To convey sensitive information, the system must ensure confidentiality.
 - Encryption : transforming the original information into another form
 - Decryption : convert an encrypted message to its original form



The 7 layers of the OSI



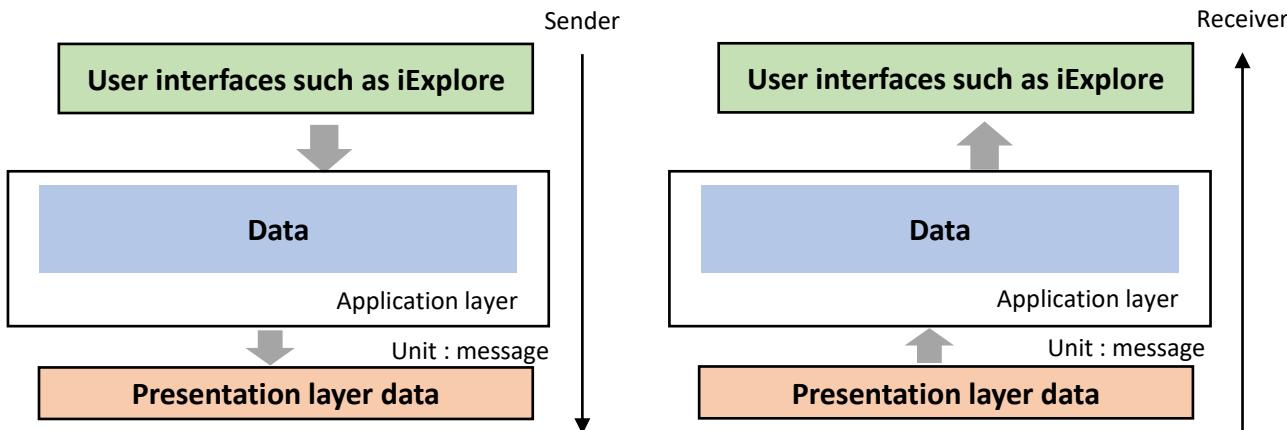
The OSI models

Application layer

The application layer is the only layer that does not provide services to the upper layers because it exists at the top; it only uses the services provided by the lower layers.

Features

- Protocols : Telnet, FTP, email (POP, SMTP), HTTP, etc.
- Receive data from users and forward it to lower layers
- Take the role of passing data from lower layers to users
- Provide multiple network services in a substantial way
- Support for protocols required to perform specific network service functions



The 7 layers of the OSI

| | |
|---|---------------------|
| 7 | Application (data) |
| 6 | Presentation (data) |
| 5 | Session (data) |
| 4 | Transport (segment) |
| 3 | Network (packet) |
| 2 | Data link (frame) |
| 1 | Physical (bit) |

The OSI models

Recap of the OSI 7 layers

Review and summarize the OSI 7 layers.

- Protocols and devices by layer

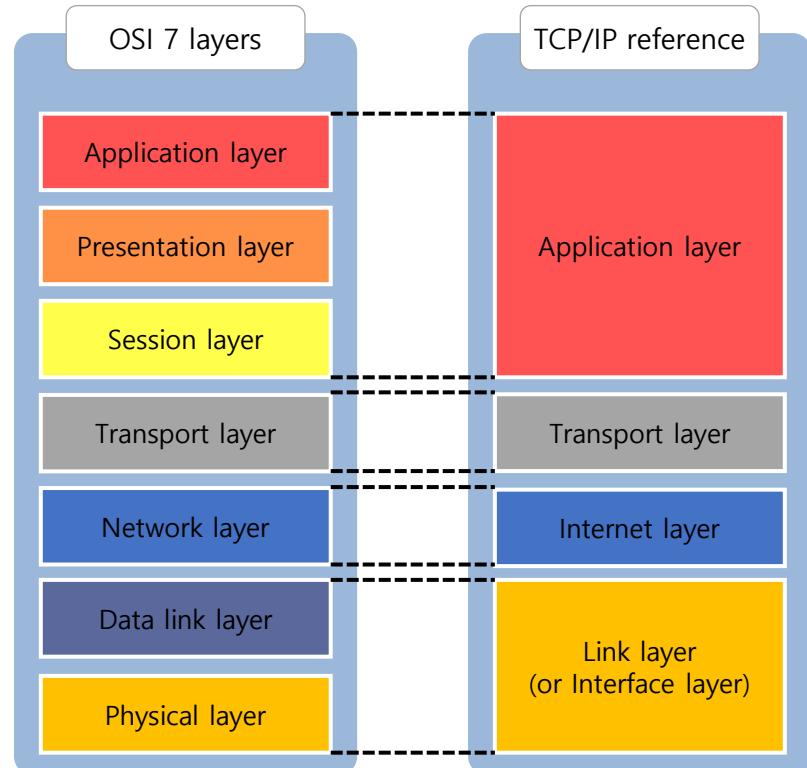
| Layer | Name | Data unit | Device | Protocol | Description |
|---------|--------------|---------------------------------|-------------------|-------------------------------|---|
| Layer 7 | Application | Data | | HTTP, FTP, SMT P, DNS, Telnet | Allow users to access the network Provide services such as user interface, email, database management, etc. |
| Layer 6 | Presentation | Data | IDS, IPS | JPEG, MPEG, S MB | Data translation and encryption Compress data as needed |
| Layer 5 | Session | Data | | NetBIOS, SSH | Start and end a session Manage communication between applications and keep them in sync Retransmit or recover from errors |
| Layer 4 | Transport | Segment (TCP) Datagram (UDP) | Firewall | TCP, UDP | Pass data between terminating programs Responsible for sending the entire message |
| Layer 3 | Network | Packet | Router | IP, RIP, ICMP | Pass data between end devices Route control and internetworking in data transmission |
| Layer 2 | Data link | Frame | Bridges, switches | APR, Ethernet, MAC, PPP | Logical specification for connectivity to adjacent devices MAC address, error control, sequence control, flow control (Participation in communication of trusted information) |
| Layer 1 | Physical | Bit | Hubs, repeaters | RS-232, BASE-T | Physical specifications for connection to adjacent devices Data transmission and signal conversion over transmission media |

The TCP/IP models

4-layer TCP/IP model

TCP/IP is an industry-standard set of protocols designed for large internetworks that extend beyond LAN and WAN environments.

- Occurrence Background
 - Started in 1969 when the Department of Defense (DoD) launched the ARPANET
 - The Telnet and FTP protocols needed to share information over the ARPANET were developed first.
 - In 1974, TCP was introduced, describing how to organize a reliable host-to-host data transfer service
 - In 1981, IP was introduced, implementing addressing standards and routing packets between interconnected networks
 - On January 1, 1983, the ARPANET began using the TCP and IP protocols as the standard for all network traffic and basic communications.



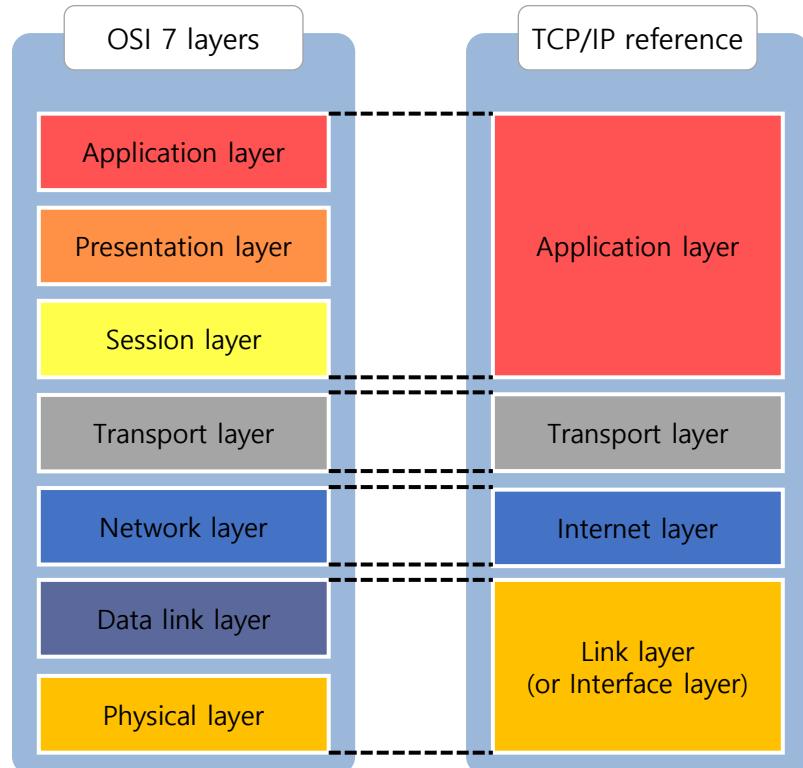
The TCP/IP models

4-layer TCP/IP model

TCP/IP is an industry-standard set of protocols designed for large internetworks that extend beyond LAN and WAN environments.

- Features

- The protocol that governs all forms of transmission over the Internet
- TCP/IP and internetworking evolve in parallel
- Completion and use of the two primary protocols
 - Transmission Control Protocol (TCP)
 - Internet Protocol (IP)



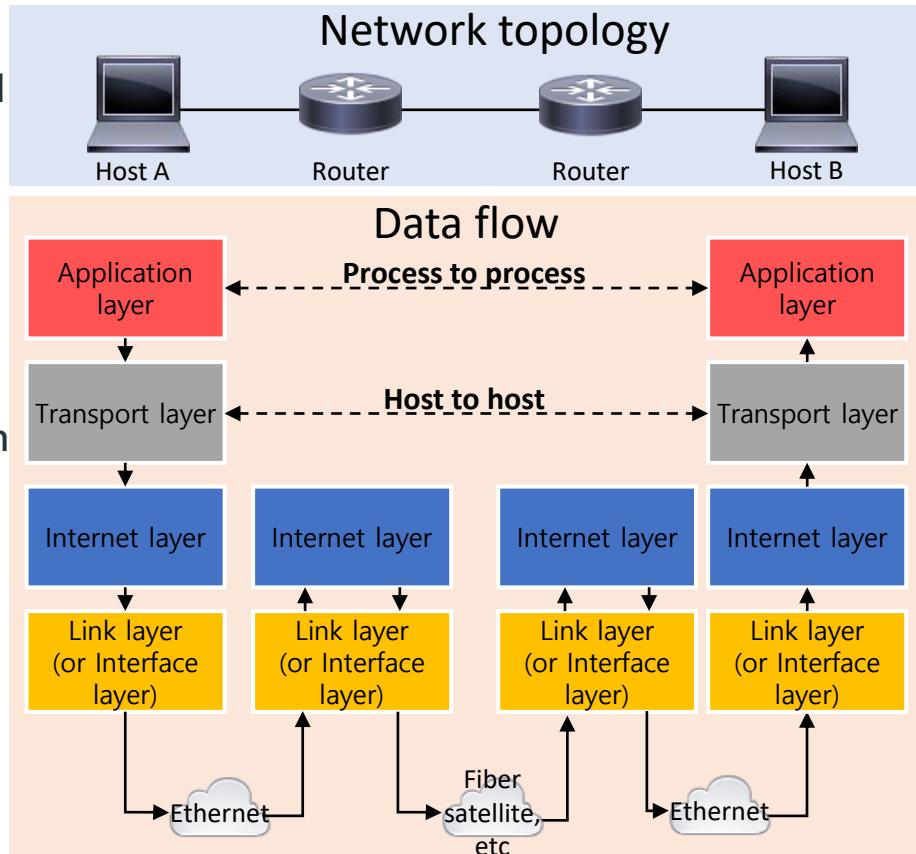
The TCP/IP models

4-layer TCP/IP model

TCP/IP is an industry-standard set of protocols designed for large internetworks that extend beyond LAN and WAN environments.

- How its sturcutre works

- The TCP/IP layer is a concept of operational scope.
 - **It is not a concept used for networking technologies, procedures, or processing of data semantics.**
- End-to-end connections evolve over time
- Use of encapsulation to provide abstraction of protocols and services
- RFC1122, an early architecture document, emphasizes the principle of layering.
- Compare network topology and data flow



The TCP/IP models

4-layer TCP/IP model

TCP/IP is an industry-standard set of protocols designed for large internetworks that extend beyond LAN and WAN environments.

- How its structure works
 - Understanding each layer
 - Application layer
 - An application or process creates user data and communicates with other applications on the host.
 - Commonly used protocols include SMTP, SSH, FTP, HTTP, etc.
 - Processes are handled through service ports.
 - Transport layer
 - Perform host-to-host communication on a local network or a remote network separated by a router
 - Provide a channel for the application's communication needs
 - Configure UDP and TCP protocols
 - User Datagram Protocol (UDP)
 - Transmission Control Protocol (TCP)

The TCP/IP models

4-layer TCP/IP model

TCP/IP is an industry-standard set of protocols designed for large internetworks that extend beyond LAN and WAN environments.

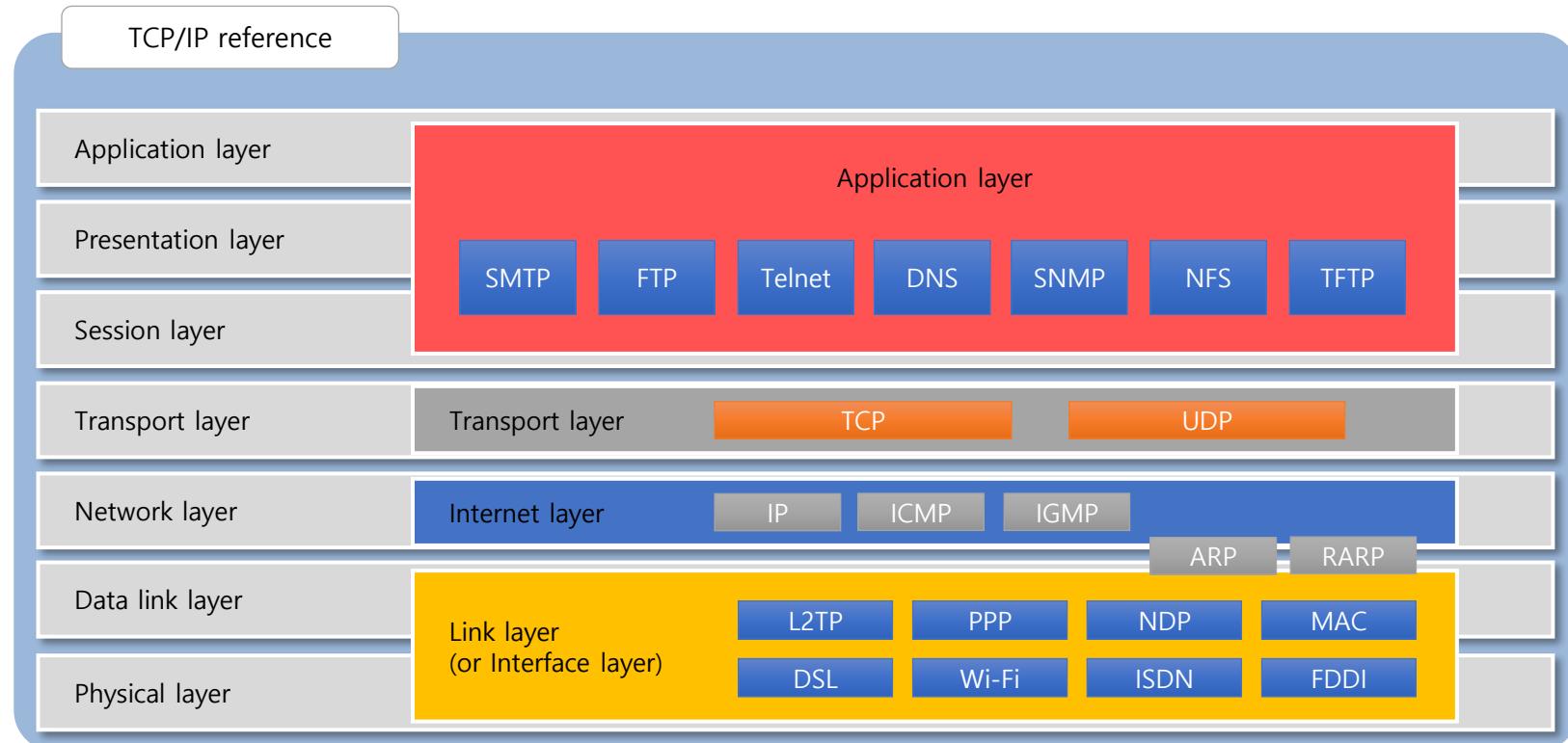
- How its structure works
 - Understanding each layer
 - Internet layer
 - Exchange datagrams over the network
 - Provide homogeneous network interfaces
 - Define the routing structure that identifies addresses in the TCP/IP protocol.
 - Define an IP address and specify a route for sending datagrams
 - Link layer
 - Hosts communicate without a router, which defines how they network within the boundaries of the local network.
 - Include the interfaces needed to send datagrams from the Internet layer to the next neighboring host

The TCP/IP models

4-layer TCP/IP model

TCP/IP is an industry-standard set of protocols designed for large internetworks that extend beyond LAN and WAN environments.

- 4-layer TCP/IP model



Understanding the Local Area Network (LAN)

Switch Overview

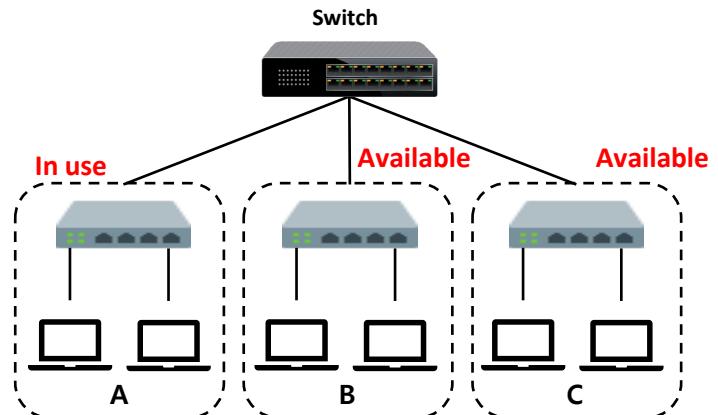
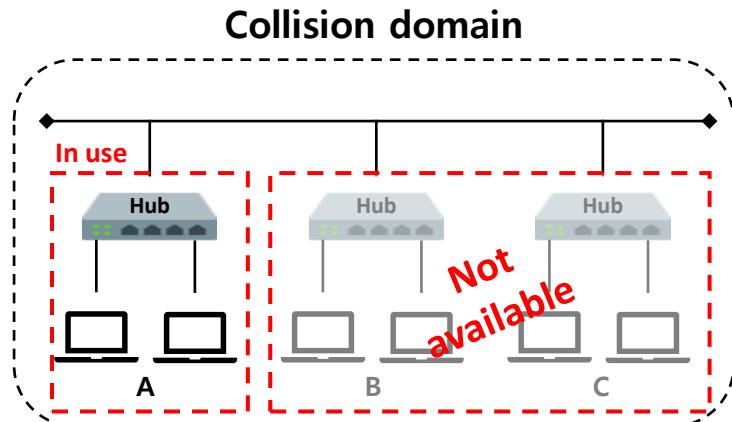
A switch is a repeater at Layer 2 (data link) of the seven layers of the OSI. The switch sends frames to the appropriate port based on information such as the MAC address of the connected device or the connection port.

- Background

- When only two PCs connected to the hub communicate, other PCs cannot communicate at the same time. (CSMA/CD method is the cause)

- Switch features

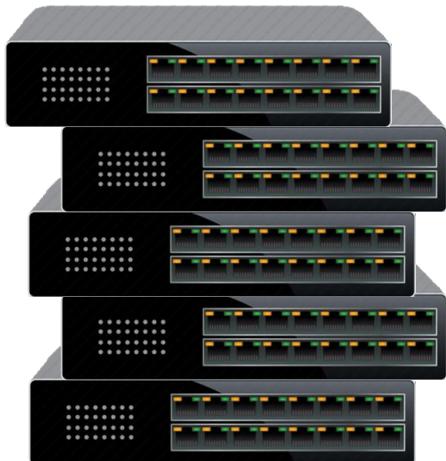
- Serve as a connection from another network (LAN)
- Process information from the data in the frame and uses the MAC address



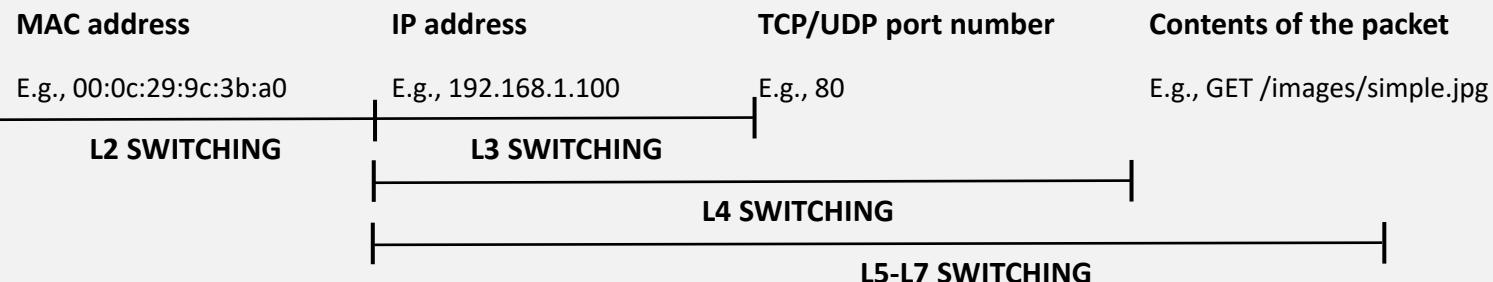
Understanding the Local Area Network (LAN)

Switch type

A switch is a device that switches incoming network packets. The types of switches are referred to as L1 through L7 switches because they contain layer-specific features.



- L7 switch** → Load balancing, security features
- L4 switch** → Load balancing with port numbers
- L3 switch** → Switching, routing & forwarding with IP addresses
- L2 switch** → Switching with MAC addresses
- L1 switch** → Hub, flooding

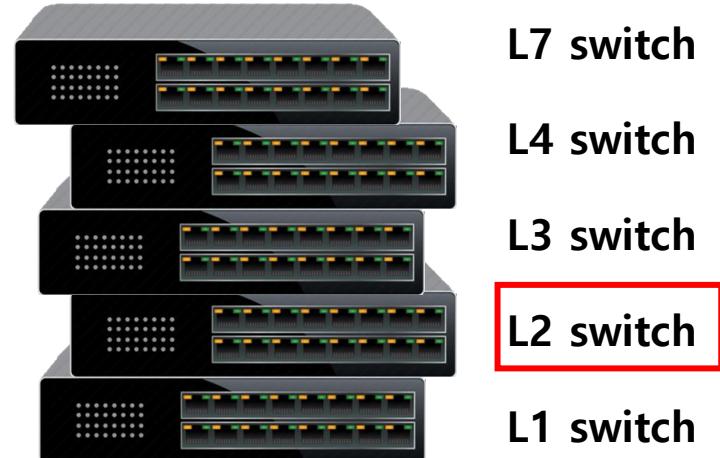


Understanding the Local Area Network (LAN)

L2 switch

L2 switches are commonly referred to as switches and operate at the Layer 2, the data link layer. It is responsible for sending packets with a MAC address.

- L2 : overview and features
 - Corresponds to two of the seven layers of OSI
 - Send frames coming in through the input port to the output port based on the destination MAC address
 - Divide the collision domain
- Advantages and disadvantages of L2 switches
 - Advantages : simple structure, low cost
 - Disadvantages : performance degradation due to broadcast packets



Understanding the Local Area Network (LAN)

Data link layer and link layer

We will explore the difference between the data link layer in the OSI network model and the link layers in the TCP/IP model and recognize the services used.

- Understanding the data link layer
 - Layer 2 of the OSI network model (ISO 7498)
 - A protocol layer that transmits data between adjacent network nodes on a WAN or between nodes on the same LAN segment.
 - Pass data to upper layers when communication outside the internal network routing is required, without leaving the internal network.
 - Detect and correct errors that may occur at the physical layer
- Link layer
 - Layer 1 of the TCP/IP model (RFC 1122, RFC 1123)
 - A link is a physical and logical network component used to interconnect hosts or nodes on a network.
 - Link protocol, used at the link layer, refers to a collection of methods and standards that work only between adjacent network nodes on a network segment.
 - Often represented by a combination of Layer 1 and Layer 2 in the OSI layer model

Understanding the Local Area Network (LAN)

Services

We will explore the difference between the data link layer in the OSI network model and the link layers in the TCP/IP model and recognize the services used.

- Services in the data link layer
 - Encapsulation of data packets in a frame
 - Synchronization of frames
 - Functions within the Logical Link Control (LLC) sublayer
 - Error control (ARQ, Automatic Repeat Request)
 - Perform functions such as error detection and packet discarding in addition to ARQs provided by some transport layers
 - Flow control
 - Perform flow control functions used in modem and wireless networks rather than LAN protocols such as Ethernet
 - Medium Access Control (MAC)
 - Multiple access protocols for channel access control (e.g., CSMA/CD or CSMA/CA protocols)
 - Physical addressing (MAC addressing)
 - LAN switching such as MAC filtering, Spanning Tree Protocol (STP), and Shortest Path Bridging (SPB)

Understanding the Local Area Network (LAN)

Services

We will explore the difference between the data link layer in the OSI network model and the link layers in the TCP/IP model and recognize the services used.

- Services in the data link layer
 - Medium Access Control (MAC)
 - Data packet scheduling
 - Store-and-forward switching or cut-through switching
 - Store-and-forward method
 - Switching incoming frames to one of the output sides
 - Buffer all incoming frames, and fully process them, including error detection, before forwarding.
 - Cut-through method: the switching system examines only the header portion of the received packet and switches it.
 - Quality of Service (QoS) controls
 - Virtual LANs (VLANs)

Understanding the Local Area Network (LAN)

ARP structure

Address Resolution Protocol (ARP) is a protocol for finding the physical address (MAC address) of a host that corresponds to an IP address on a local network.

- Structure

| H/W type | Protocol type | |
|------------|-------------------------|---------|
| H/W length | Protocol length | OP CODE |
| | Source MAC address | |
| | Source IP address | |
| | Destination Mac address | |
| | Destination IP address | |

28 bytes

- Segment

- Operation code (OPER) : 1 = ARP request, 2 = ARP reply, 3 = RARP request, 4 = RARP reply
- Source Ethernet/IP address : MAC/IP address of the sender
- Destination Ethernet/IP address : MAC/IP address of the receiver

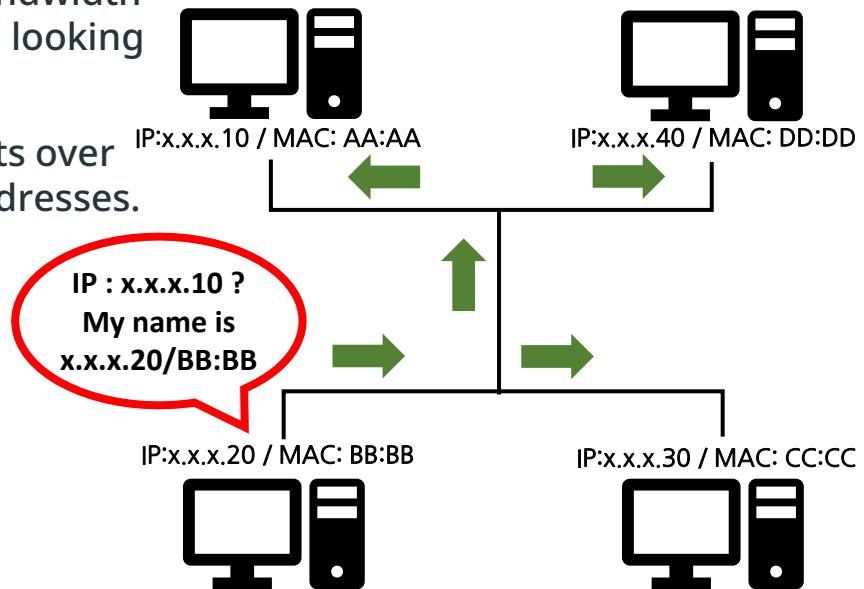
Understanding the Local Area Network (LAN)

How ARP works

We will explore the role of ARP and how physical IP addresses and MAC addresses interact.

- Communication principles

- To find a PC with a specific IP, ask for the bandwidth you are on what MAC address the IP you are looking for has.
- The requesting PC sends ARP request packets over the entire band, including its IP and MAC addresses.

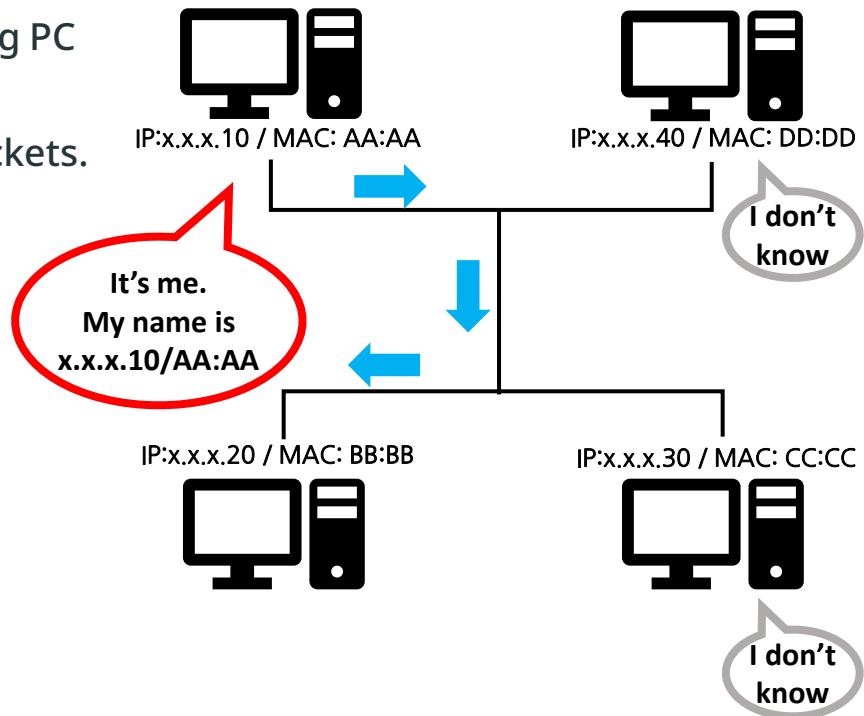


Understanding the Local Area Network (LAN)

How ARP works

We will explore the role of ARP and how physical IP addresses and MAC addresses interact.

- Communication principles
 - The PC at the IP destination of the discovery sends an ARP replay packet to the requesting PC with information about its MAC address.
 - Unaffiliated PCs discard their ARP replay packets.

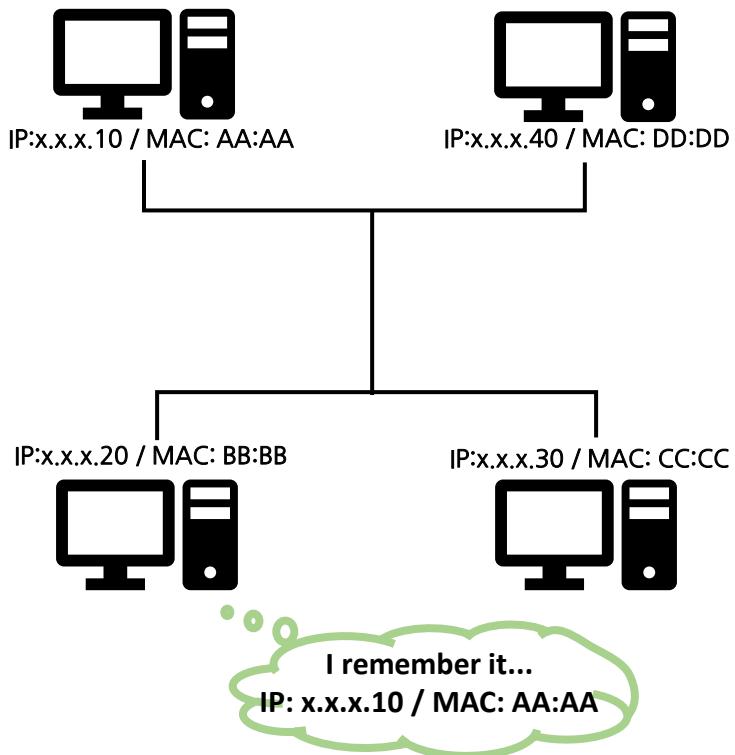


Understanding the Local Area Network (LAN)

How ARP works

We will explore the role of ARP and how physical IP addresses and MAC addresses interact.

- Communication principles
 - Once you have the MAC address for the IP you are looking for in your PC, map the IP and MAC address to the ARP table in the ARP table and store it in the cache.
 - The ARP table is updated at certain intervals and periodically sends an ARP reply packet to ensure continuous MAC address mapping.



Understanding the Local Area Network (LAN)

Overview

MAC addresses are used in local area networks as a factor in determining routes and are structured according to specific rules.

- Definition

- Short for Media Access Control Address
- A unique identifier assigned to a network interface for data link layer communications.
- MAC addresses are used as addresses for IEEE802 network technologies, including Ethernet and Wi-Fi.
- Also known as hardware address, physical address, and Ethernet hardware address

- Configuration

- Usually assigned by the network interface card (NIC) manufacturer and stored in the hardware.
- Typically coded with the manufacturer's registered identification number.
- The manufacturer's registration number is called the OUI value
- MAC-48, EUI-48, and EUI-64. IEEE has trademarked the names EUI-48 and EUI-64.



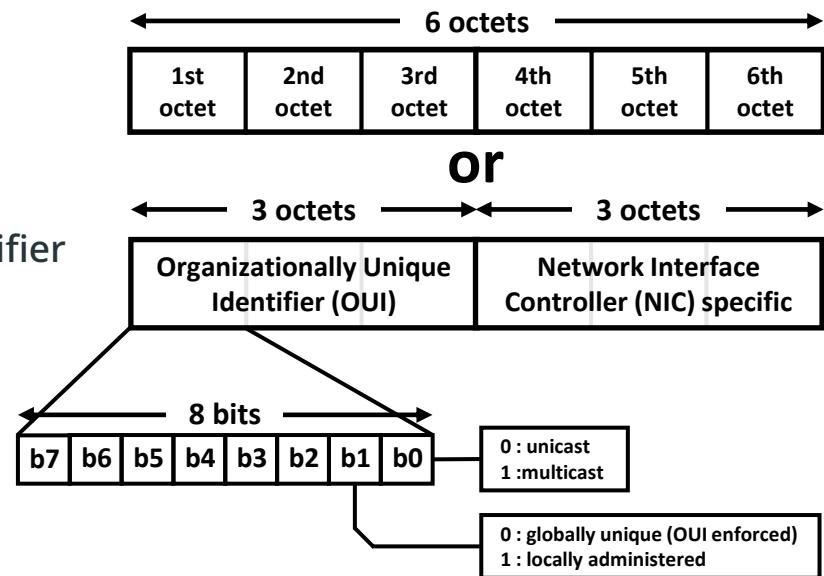
MAC address on the device

Understanding the Local Area Network (LAN)

Structure

MAC addresses are used in local area networks as a factor in determining routes and are structured according to specific rules.

- Size
 - 48-bit address space
 - 2^{48} , with 281,474,976,710,656 available MAC addresses
- Global vs. regional management
 - Universally Administered Address (UAA)
 - Uniquely assigned by the manufacturer
 - The first three octets (8 bits) are the Organization Unique Identifier (OUI).
 - Identify the organization that issued the identifier
 - Locally Administered Address (LAA)
 - The next three octets can be assigned as desired by your organization.
 - Subject to uniqueness constraints



Understanding the Local Area Network (LAN)

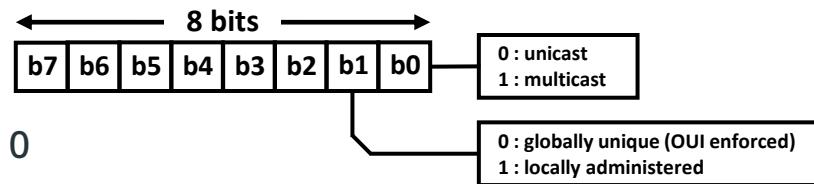
Structure

MAC addresses are used in local area networks as a factor in determining routes and are structured according to specific rules.

- Unicast and Multicast

- Unicast

- Set the least significant bit of the first octet to 0
 - Frame reaches only one listening NIC
 - Send the frame to all nodes in the collision domain to reach the nearest router or switch.
 - Forward unicast frames to all ports to check in situations where the switch doesn't know which port is reaching that MAC address



- Multicast

- Set the least significant bit of the octet to 1
 - Send frame only once
 - Decide whether to accept NICs based on criteria other than MAC address matching
 - Determined by a configurable list of allowed multicast MAC addresses (multicast addressing)

Understanding the Local Area Network (LAN)

Structure

MAC addresses are used in local area networks as a factor in determining routes and are structured according to specific rules.

- 00-00-00-00-00-00
 - Use when you don't know the recipient's MAC address
 - The way the target MAC address is presented
- FF-FF-FF-FF-FF-FF
 - Send as broadcast
 - Means to send packets to every node on a network

As a means of route tracing for data transmission at Layer 3 of the seven OSI layers, IP is used to access the services of foreign or domestic companies in remote locations.

- Overview

- Short for Internet Protocol; an information-oriented convention (protocol) used by sending and receiving hosts to exchange information via packets.
- It is used as a means of routing to send and receive data at layer 3 of the seven OSI layers.
- It has a size of 32 bits, divided into 4 decimal units.
- Each IP is categorized by class, which is based on the value of the subnet mask, and exists in classes A through E.
- Divided into public and private IPs
 - Public IP : a uniquely used IP that is managed and assigned to a country by an organization at ICANN.
 - Internet Corporation for Assigned Names and Numbers (ICANN)
 - Private IP : an address used within a network, unique to that network but not to the Internet (serves as an alternative to running out of public IPs).

IPv4 and IPv6

IPv4 class

The Internet Protocol (IP) is the most prominent network layer protocol and is responsible for establishing a path for data transfer between two nodes using lower layer services.

- IP addressing schemes

| Shape | Network number area | Number of available network addresses | Number of available computers | Usage |
|---------|----------------------------|---------------------------------------|-------------------------------|-----------------------------------|
| Class A | 1.x.x.x to 126.x.x.x | 126 | 16,777,214 | National and other large networks |
| Class B | 128.0.x.x to 191.254.x.x | 16,382 | 65,534 | Schools, mid-sized businesses |
| Class C | 192.0.1.x to 223.255.254.x | 2,097,150 | 254 | ISP provider networks |

| | | | |
|----------------|---|-----------------|-----------------|
| Number of bits | 1 | 7 | 24 |
| Class A | 0 | Network address | Host address |
| Number of bits | 2 | 14 | 16 |
| Class B | 1 | 0 | Network address |
| Number of bits | 3 | 21 | 8 |
| Class C | 1 | 1 | 0 |
| | | Network address | Host address |

IPv4 and IPv6

IPv4 class

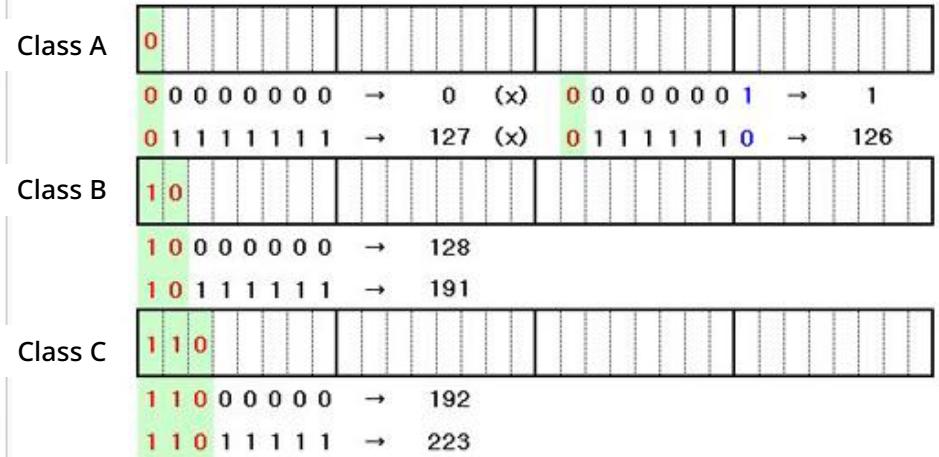
The Internet Protocol (IP) is the most prominent network layer protocol and is responsible for establishing a path for data transfer between two nodes using lower-layer services.

- IP addressing schemes

10000000 . 00001011 . 00000011 . 00011111

128.11.3.31

| Class | First Octet Range | Max Hosts | Format |
|-------|-------------------|-----------|---|
| A | 1-126 | 16M |  |
| B | 128-191 | 64K |  |
| C | 192-223 | 254 |  |
| D | 224-239 | N/A |  |
| E | 240-255 | N/A |  |



As a means of route tracing for data transmission at Layer 3 of the seven OSI layers, IP is used to access the services of foreign or domestic companies in remote locations.

- Private IP

- Mainly used in small office spaces where equipment such as routers are assigned a single public IP and shared by multiple devices through the NAT method.
- Private IPs are assigned to each class
 - Class A : 10.0.0.0 to 10.255.255.255
 - Class B : 172.16.0.0 to 172.31.255.255
 - Class C : 192.168.0.0 to 192.168.255.255

IPv4 and IPv6

IPv6

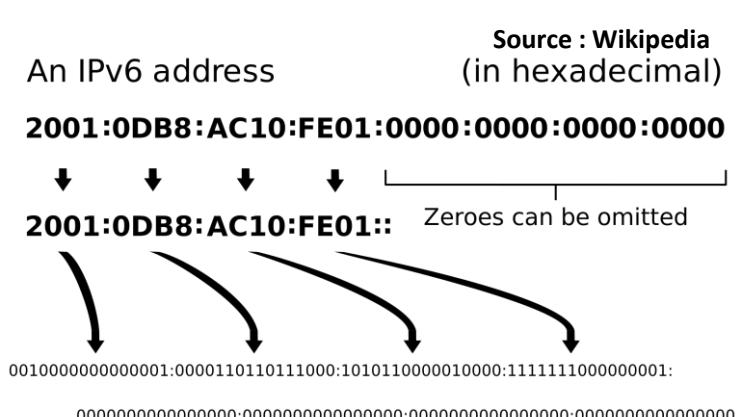
The IPv6 addressing scheme is an addressing scheme developed by the IETF as a solution to the address exhaustion problem of IPv4.

- What is IPv6?

- Extend IPv4 to 128 bits to fix address exhaustion issues
- Developed by the Internet Engineering Task Force (IETF)
- IPv6 standardized in December 1998

- Structure

- Configure with a unique IP for identification and location definition
- With 128 bits, it theoretically has as many as 2^{128} or about $3.4 * 10^{38}$ addresses.



IPv4 and IPv6

IPv6

The IPv6 addressing scheme is an addressing scheme developed by the IETF as a solution to the address exhaustion problem of IPv4.

- Division

| Address type | Binary representation | Address notation | Remark |
|------------------------|-----------------------|------------------|---|
| Unspecified address | 0000....0(128) | ::/128 | From address without an IP address |
| Loopback address | 0000....1(128) | ::1/128 | Address of the loopback interface of the host |
| Multicast address | 11111111 | FF00::/8 | Multicast IPv6 address |
| Link local address | 1111111010 | FE80::/10 | Address valid only for the link local zone |
| Global unicast address | | | All other areas |

IPv4 and IPv6

IPv6

The IPv6 addressing scheme is an addressing scheme developed by the IETF as a solution to the address exhaustion problem of IPv4.

- Address System Correspondence Relationships

| Division | Address | Division | Address |
|---------------------|-----------------|--------------------|------------------------|
| Multicast address | FF00:/8 | Public IP address | Global Unicast address |
| Broadcast address | No such address | Private IP address | No such address |
| Unspecified address | ::/128 | Link local address | FE80:/64 |
| Loopback address | ::1/128 | | |

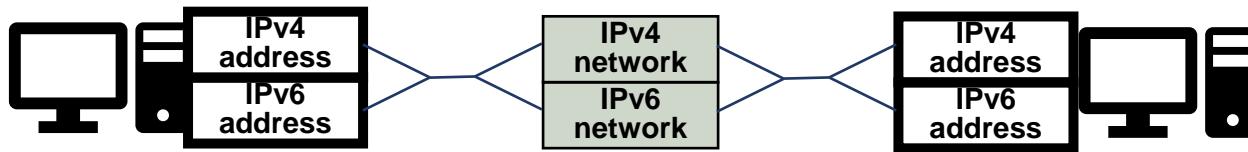
IPv4 and IPv6

IPv4 and IPv6

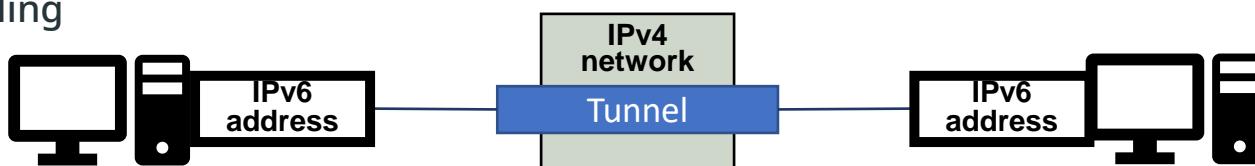
While much of the infrastructure for IPv4 is in place today, the infrastructure for IPv6 is not yet well established. This necessitated a way to make both IPv4 and IPv6 compatible, and we've done that with several technologies.

- Transition techniques

- Dual stack



- The most common ways to implement both types of addresses in your application
- Tunneling



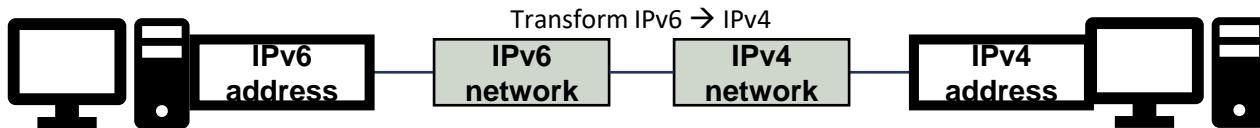
- Create a virtual, logical tunnel path between two terminals to communicate
 - Increased packets from tunneling, as well as risks from allowing the tunneling protocol

IPv4 and IPv6

IPv4 and IPv6

While much of the infrastructure for IPv4 is in place today, the infrastructure for IPv6 is not yet well established. This necessitated a way to make both IPv4 and IPv6 compatible, and we've done that with several technologies.

- Transition techniques
 - Translation



- Communicate different addressing schemes through complete recombination of packets
- Have limitations due to the need to reflect all communication application-specific characteristics
- Recommended only for special applications (e.g., industrial) and exists in three forms
 - Application Level Gateway (ALG) : transform data at the application layer
 - Transport Relay Translator (TRT) : transform data at the transport layer
 - Stateless IP/ICMP Translation (SIIT) : transform data at the network layer

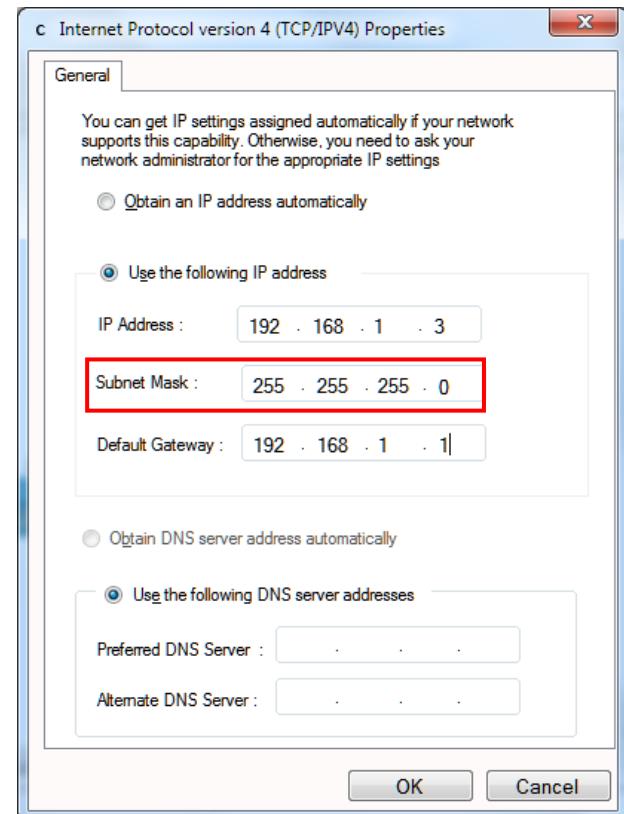
Subnetting

Subnet overview

A subnet is a logical, granular unit into which a network is divided. It is short for subnetwork, a unit in which a larger network is divided into smaller networks.

- Check the Subnet Mask (on Windows 10)

- Paths
 - Open the Control Panel
 - Network and Internet
 - Network Connections
(Or, via Network and Sharing Center > Connections > Properties; without going over Ethernet)
 - Ethernet
 - Double-click Internet Protocol Version 4 (TCP/IPv4)



Subnetting

What is a subnet mask?

A subnet is a logical, granular unit into which a network is divided. It is short for subnetwork, a unit in which a larger network is divided into smaller networks.

- Subnet Mask
 - Masks for creating subnets
 - Used to distinguish between network and host addresses for IP addresses.
 - The subnet mask consists of 32 consecutive binary bits of 1's separated by a dot (.) every 8 bits, expressed in decimal.
 - E.g., 11111111.11111111.11111111.00000000 -> 255.255.255.0
 - The IP address followed by the subnet mask is expressed as a number of bits (the number of 1s in binary) for a shorter and simpler representation.
 - E.g., 192.168.0.100/255.255.255.128 -> 192.168.0.100/25
 - The set subnet mask determines whether the IP address requested by the user is in the same subnet as the current host.
 - Same subnet : direct communication
 - Other subnet : sent through gateway (router)

Subnetting

What is subnetting?

A subnet is a logical, granular unit into which a network is divided. It is short for subnetwork, a unit in which a larger network is divided into smaller networks.

- Subnetting

- Method of setting a subnet mask to separate IP network addresses in the same band
 - AND a subnet mask and IP address to identify a network address
 - Same network address as the current host : direct communication
 - Network address different from current host : sent through gateway (router)

| CIDR | Subnet mask | Number of networks | Number of IPs per 1 network |
|------|-----------------|--------------------|-----------------------------|
| /24 | 255.255.255.0 | 1 | 256 |
| /25 | 255.255.255.128 | 2 | 128 |
| /26 | 255.255.255.192 | 4 | 64 |
| /27 | 255.255.255.224 | 8 | 32 |
| /28 | 255.255.255.240 | 16 | 16 |
| /29 | 255.255.255.248 | 32 | 8 |
| /30 | 255.255.255.252 | 64 | 4 |
| /31 | 255.255.255.254 | 128 | 2 |
| /32 | 255.255.255.255 | 256 | 1 |

Subnetting

Questions on subnetting

A subnet is a logical, granular unit into which a network is divided. It is short for subnetwork, a unit in which a larger network is divided into smaller networks.

- Create a subnet mask of 192.168.50.160/27.
- 192.168.50.100/24 is a single network. Subnet it to 16.
- 192.168.50.120/25 and 192.168.50.130/25 exist on the same subnet and communicate directly (O/X).

Network protocol

- Overview
- ICMP
- TCP/UDP
- SSL/TLS
- HTTP/HTTPS
- Virtual private networks (VPN) protocols

Overview

- Commonly used protocols on the network
 - ICMP, TCP, UDP, SSL/TLS, HTTP/HTTPS
- ICMP
 - Protocol for managing errors on a network
- TCP
 - Protocol used for reliable network connectivity and communication
- UDP
 - Protocol used for unreliable but fast data exchange
- SSL/TLS
 - Encryption technologies used to encrypt network data
- HTTP/HTTPS
 - Short for HyperTextTransferProtocol, used to load web pages using hypertext
 - HTTPS uses HTTP+SSL to encrypt data over HTTP.

Internet Control Message Protocol (ICMP) is used to diagnose the health of the source and destination nodes (health check).

- Internet Control Message Protocol (ICMP)

- A protocol used to check the operational status of devices connected to a network, such as routers, switches, and servers
- Most commercial servers are blocked due to ICMP vulnerabilities.
- The ping command
 - Representative function that operates using the ICMP protocol (commands are at Layer 7)
 - Command used to remotely check the operating status of a specific device, such as a PC, server, router, or switch
 - E.g., to check whether a device with the IP address 10.5.5.5 is operating correctly :

```
> ping 10.5.5.5
Reply from 10.5.5.5: Bytes=32 time=37ms TTL=52

Ping statistics for 10.5.5.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

Internet Control Message Protocol (ICMP) is used to diagnose the health of the source and destination nodes (health check).

- ICMP Message Types

| Type | Code | Checksum | |
|--------------------|------|----------|---------|
| Remaining headers | | | 8 bytes |
| Information (data) | | | |

- Segment

- Type : ICMP message types
- Code : code values for each type
- Checksum : fields to check for errors in the ICMP message itself (header + data)
- Remaining headers : what varies by type and code

Internet Control Message Protocol (ICMP) is used to diagnose the health of the source and destination nodes (health check).

- Types and codes of ICMP
 - The ICMP type and code fields, as shown in the table below, are used to request and respond to specific actions.

| ICMP type | ICMP code | Description |
|-----------|---|---------------------------|
| 0 | 0 =(always 0) | Response packets to pings |
| 3 | 0 = Network unreachable | Destination unreachable |
| | 1 = Host unreachable | |
| | 2 = Protocol unreachable | |
| | 3 = Port unreachable | |
| | 4 = Packet fragmentation is required, but the DF-bit is set | |
| | 5 = Source Root Failed | |
| 8 | 0 =(always 0) | Ping packets |
| 11 | 0 = TTL overrun | Timeout |
| | 1 = Fragmented packet reassembly timeout | |

Internet Control Message Protocol (ICMP) is used to diagnose the health of the source and destination nodes (health check).

- ICMP message types
 - Messages for diagnosing errors

Destination unreachable

- When a router is unable to route a datagram
- When the host is unable to forward a datagram

Source quench

- Notify datagrams to be discarded due to congestion

Error Reporting

Redirection

- Used to advertise an alternate route when the router encounters a host that is not using the best path.

Time exceeded

- IP header TTL field value expiration notification

Parameter problems

- For problems with header errors in datagrams

Internet Control Message Protocol (ICMP) is used to diagnose the health of the source and destination nodes (health check).

- ICMP message types

- Messages for queries

- Echo request & reply

- Used for diagnostics on IP hosts

- Timestamp request & reply

- Determine the round trip time required for IP datagrams to travel back and forth between two systems.

Query

- Address mask request & reply

- Host requests subnet mask from router

- Router solicitation & advertisement

- When sending data to a host on another network, request the address of the router connected to your network.

Transmission Control Protocol (TCP) is a highly reliable protocol that establishes and terminates connections.

- Overview

Roles

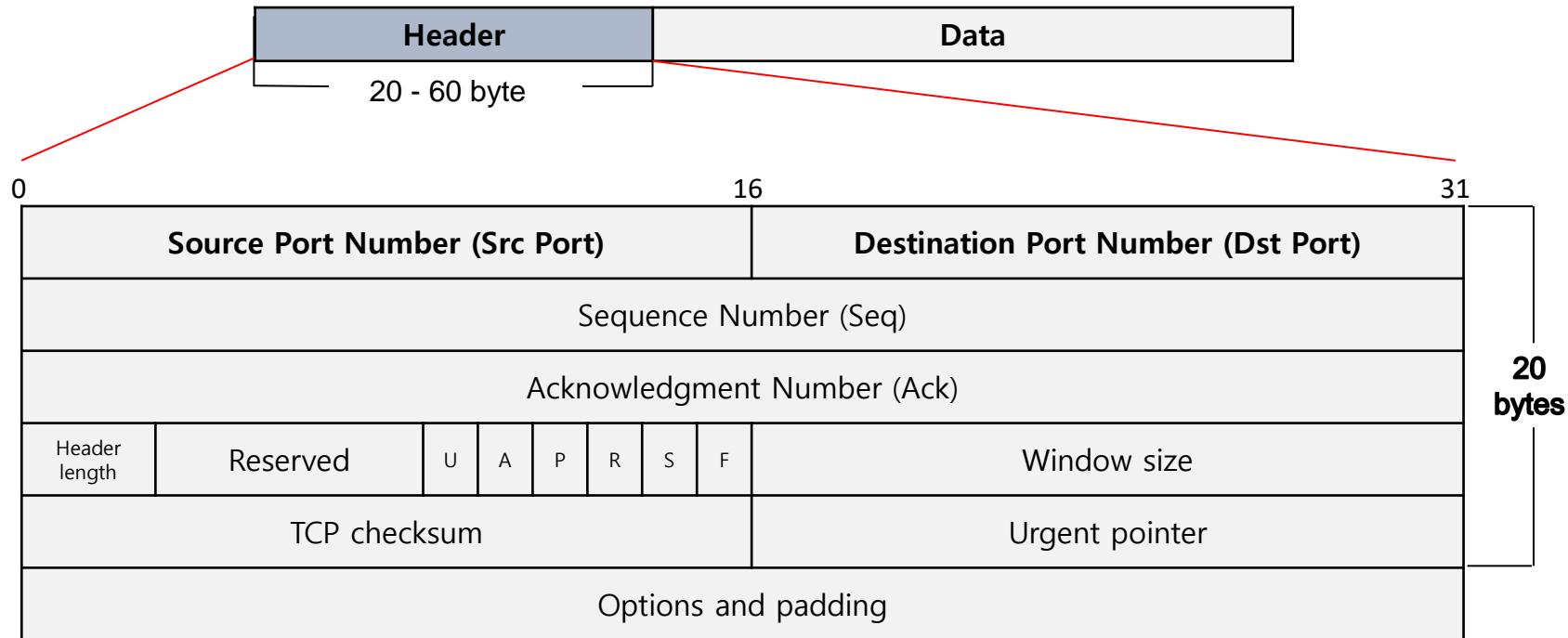
- Establish a connection between the two endpoints before sending data
- Create a virtual circuit after establishing a connection
- Announce that a datagram connection has been established and terminate the connection by closing it.
- Wait for complete messages, not just packets
- Detect errors and guarantee the retransmission of corrupt frames

Features

- High reliability
- Virtual circuit connection method
- Enable and disable connections
- Data checksums
- Timeouts and retransmissions
- Data flow control

Transmission Control Protocol (TCP) is a highly reliable protocol that establishes and terminates connections.

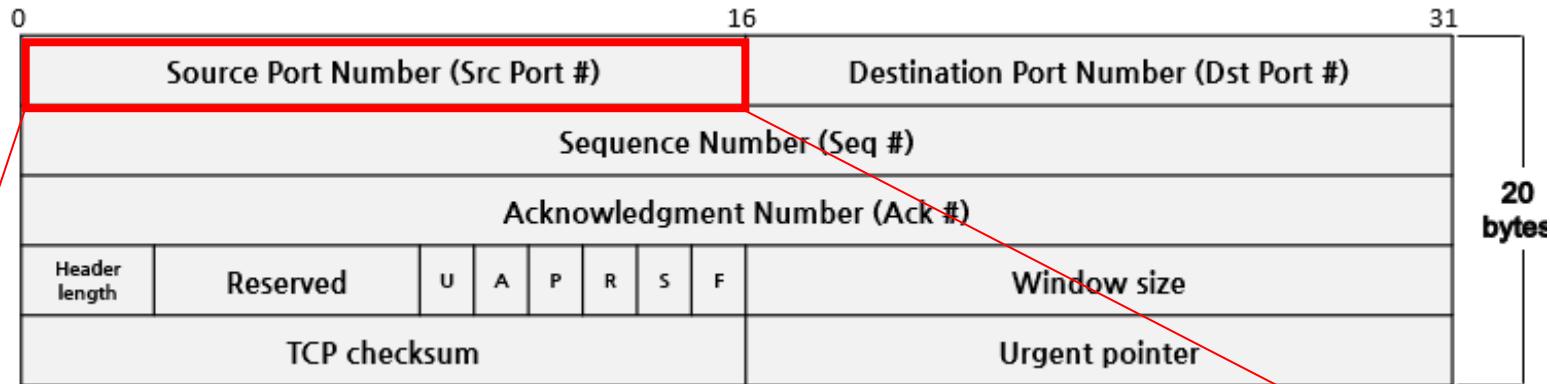
- TCP segment structure



❖ TCP header structure

Transmission Control Protocol (TCP) is a highly reliable protocol that establishes and terminates connections.

- TCP segment structure

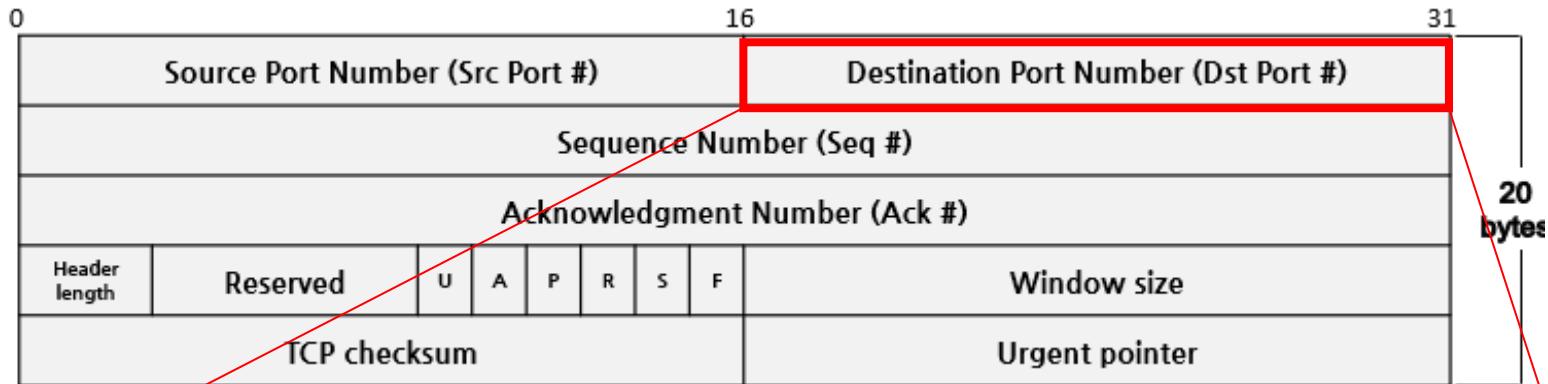


Source port address

A 16-bit field that specifies the port number of the application on the host sending the segment.

Transmission Control Protocol (TCP) is a highly reliable protocol that establishes and terminates connections.

- TCP segment structure

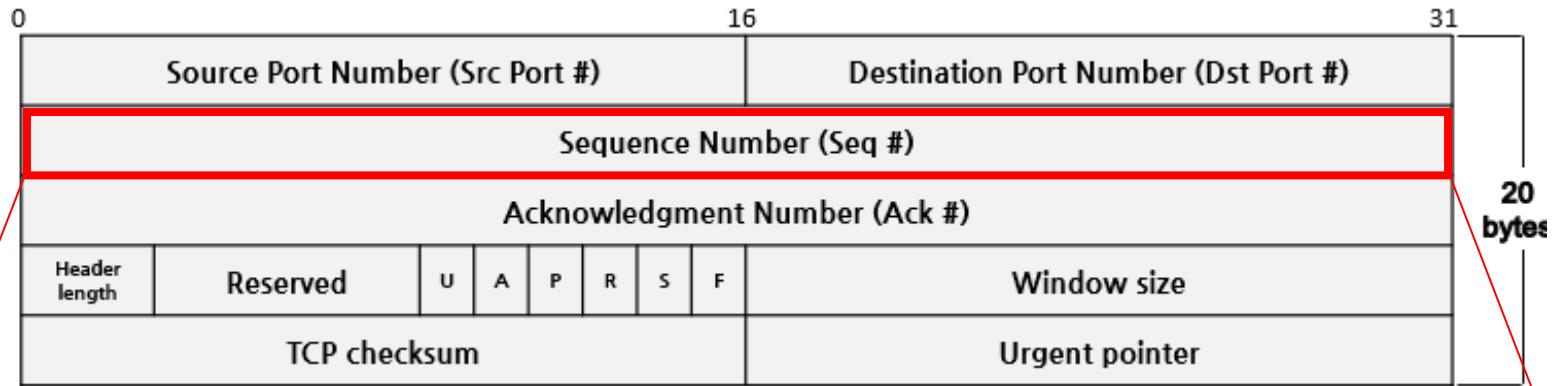


Destination port address

A 16-bit field that specifies the port number of the application on the host to which the segment is being sent.

Transmission Control Protocol (TCP) is a highly reliable protocol that establishes and terminates connections.

- TCP segment structure

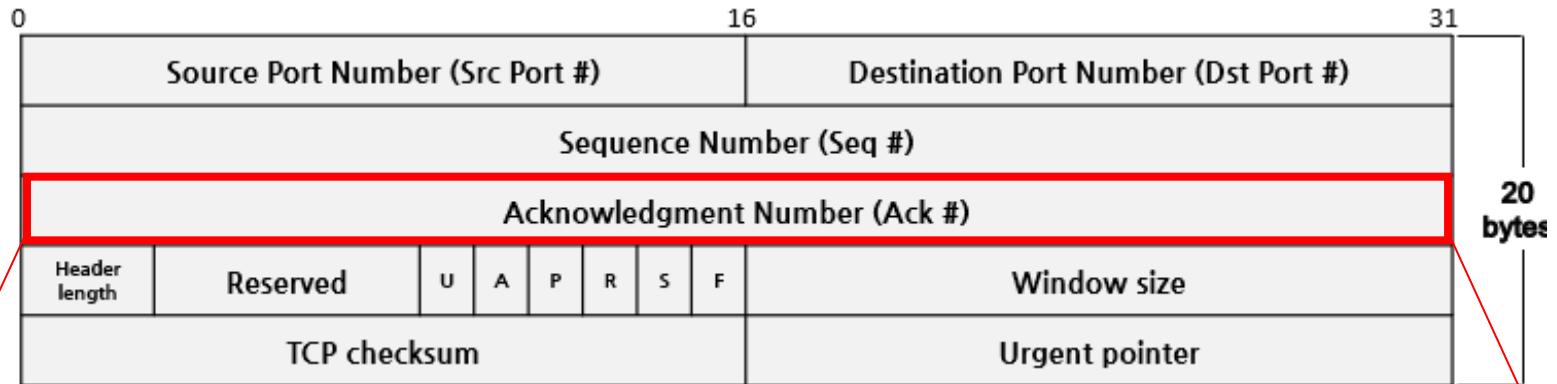


Sequence number

When the SYN flag is 1: indicate the number assigned to the first byte of data in the segment.
When the SYN flag is 0 : cumulative sequence number of the first byte value of the segment data

Transmission Control Protocol (TCP) is a highly reliable protocol that establishes and terminates connections.

- TCP segment structure

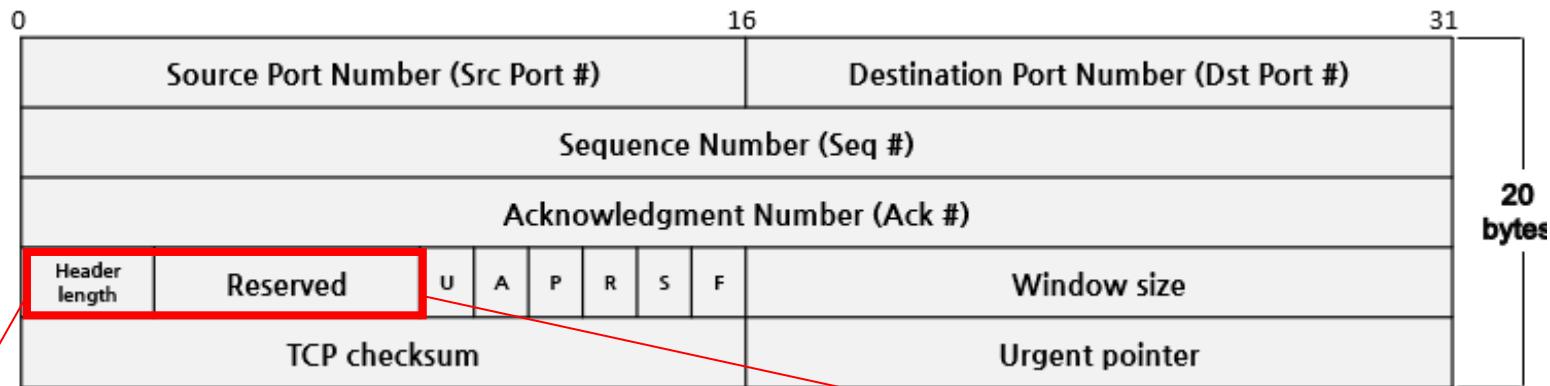


Acknowledgment number

This 32-bit acknowledgement number defines the number of bytes the receiving node wishes to receive from the other node.

Transmission Control Protocol (TCP) is a highly reliable protocol that establishes and terminates connections.

- TCP segment structure



Length

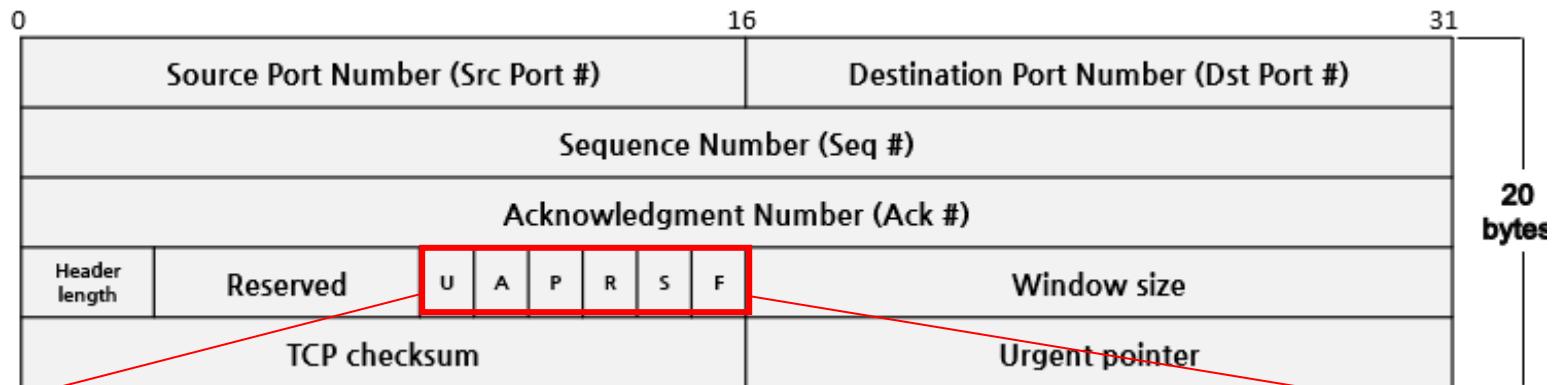
The value of this field is represented by a 4-byte word count. Therefore, the values to 5 ($5 \times 4 = 20$) should be limited to 15 ($15 \times 4 = 60$).

Reserved

6 bits of space reserved for future use, always set to 0.

Transmission Control Protocol (TCP) is a highly reliable protocol that establishes and terminates connections.

- TCP segment structure



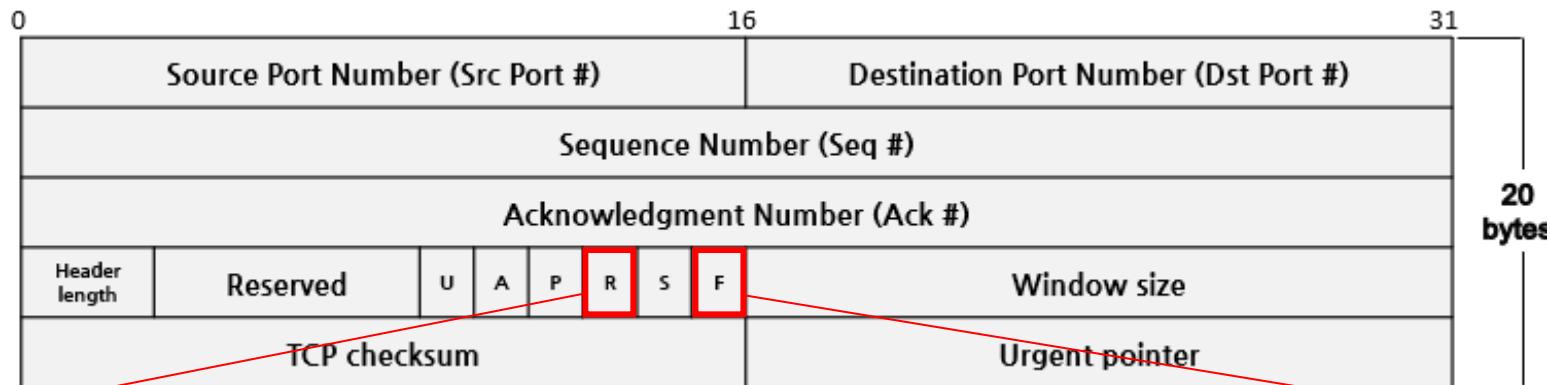
| | | | |
|--------------------------|--|--------------------------|---|
| URG (Urgent) | Urgent pointer is enabled, which is used when there is something urgent in the data you are sending. | ACK (Acknowledgement) | The receiver sends an ACK equal to the sender's sequence number plus the length or amount of data at the TCP layer. |
| PSH (Push) | Used for interactive traffic, passing data without waiting for a buffer to fill. | RST (Reset) | An interruption that is a reset and occurs simultaneously in both directions. Corresponds to an abnormal session termination. |
| SYN (Synchronization) | Used to establish a session by first sending a sequence number that is randomly generated and sent. | FIN (Finish) | Used to end the session and indicate that the sender has no more data to send. |

TCP/UDP

TCP

Transmission Control Protocol (TCP) is a highly reliable protocol that establishes and terminates connections.

- TCP segment structure



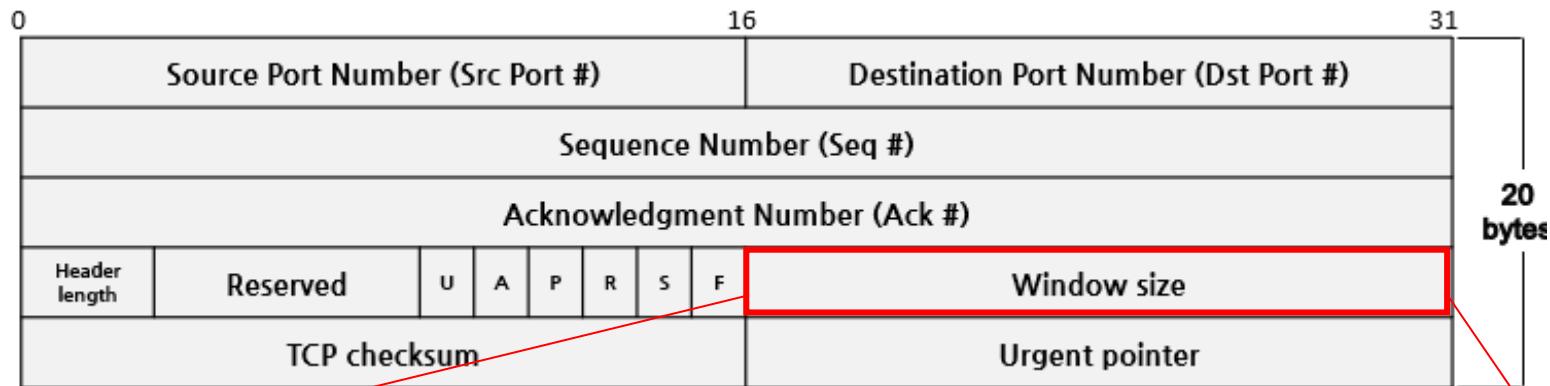
RST vs FIN

RST is used to force a TCP connection to terminate.
Corresponds to an abnormal session disconnection
Use when you want to disconnect immediately

FIN is used to terminate a TCP connection.
Means there is no more data to send

Transmission Control Protocol (TCP) is a highly reliable protocol that establishes and terminates connections.

- TCP segment structure

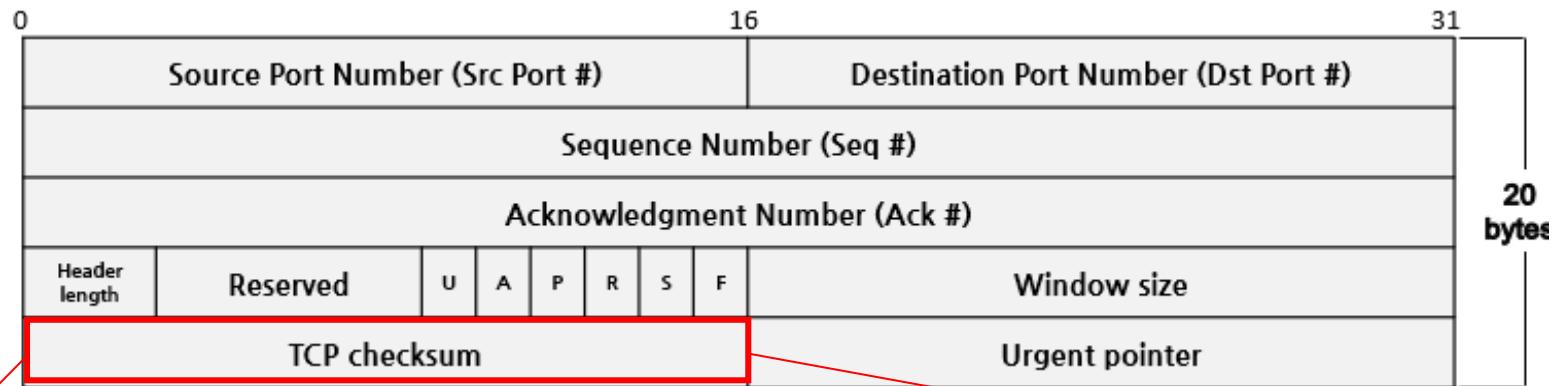


Window size

Since it is 16 bits long, it is a maximum of 65,535 bytes. The size of the window is called the Receiving Window (RWND) and is determined by the receiving side, so the sender must follow the instructions of the receiving side.

Transmission Control Protocol (TCP) is a highly reliable protocol that establishes and terminates connections.

- TCP segment structure

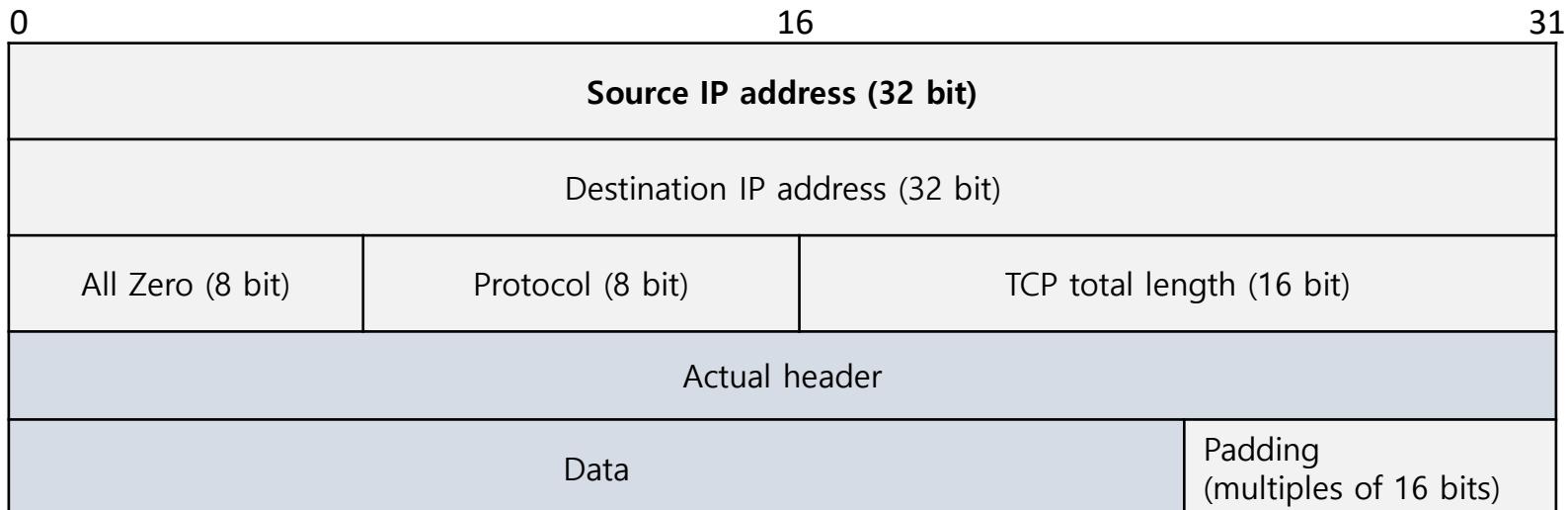


Checksum

TCP's checksum is mandatory and includes a pseudo header in addition to the actual header.

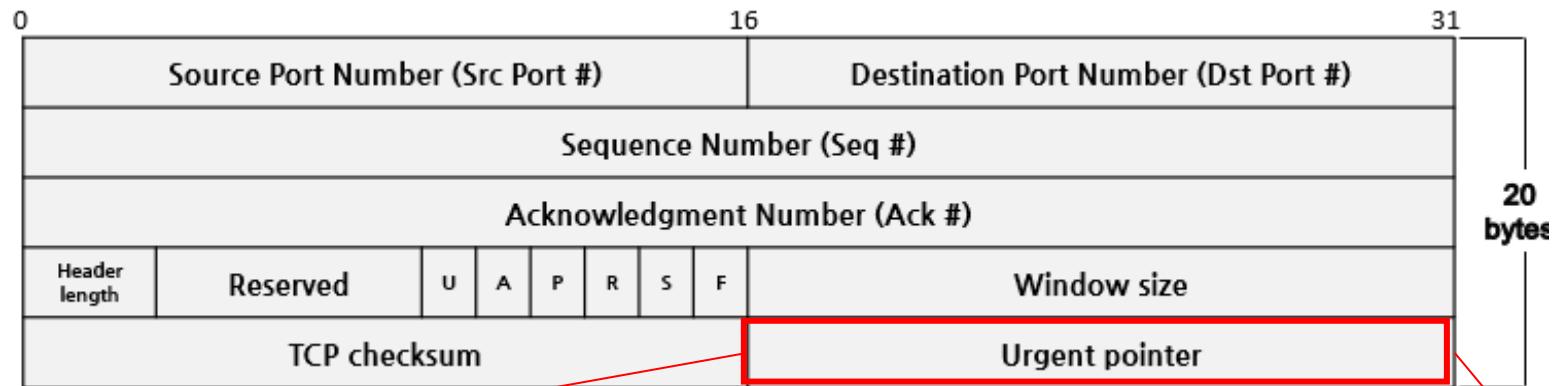
Transmission Control Protocol (TCP) is a highly reliable protocol that establishes and terminates connections.

- TCP segment structure
 - Pseudo header : calculate checksums by adding source/destination IP addresses, protocols, etc. to the actual header
 - What to calculate the checksum for : (virtual header + real header + data + padding)



Transmission Control Protocol (TCP) is a highly reliable protocol that establishes and terminates connections.

- TCP segment structure



Urgent pointer

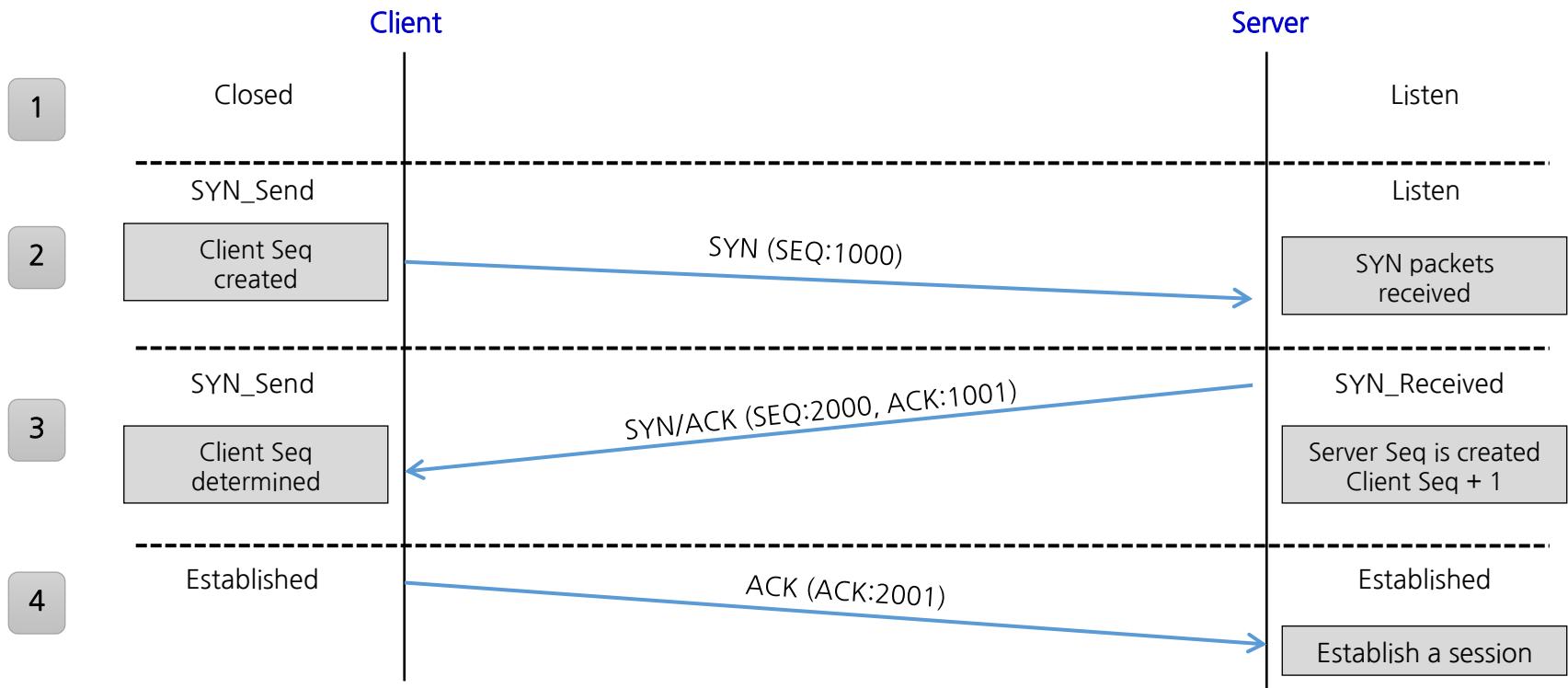
A 16-bit field valid only when the urgent flag is set to 1.

The 3-way handshake is a process by which applications using the TCP/IP protocol establish a session with the other system before sending data, ensuring accurate transmission.

- 3-way handshake connection process
 - TCP connections use the Seq and Ack numbers (#) to ensure a mutual session.
 - Seq is the order the current sender sends based on the current session.
 - Ack is a value denoting the result processed by the current sender or connected session.
 - Attacks can be performed by manipulating flags, Seq, and Ack during this process.
 - Example of the connection process
 - The client includes a randomized sequence no. when sending a SYN flag packet.
 - When the SYN/ACK flags are sent, the server sends an Ack by adding 1 to the client's Seq, and conversely sends a randomized Seq corresponding to the SYN flag to the client.
 - The client sends the Seq equal to the SYN flag plus 1 as the Ack.
 - If it receives a value not equal to the value plus 1, or no response, it sends a re-request packet and waits for legitimate data to arrive to ensure reliability.

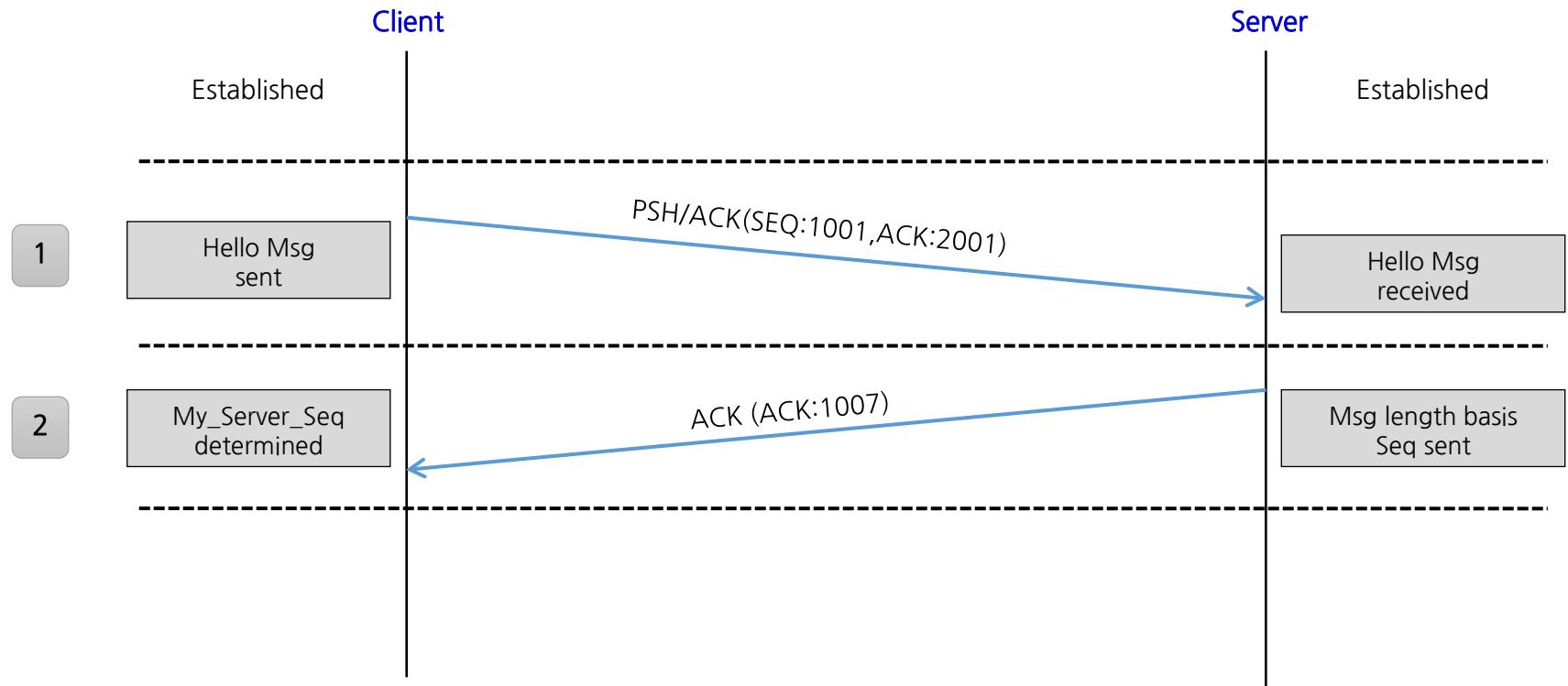
The 3-way handshake is a process by which applications using the TCP/IP protocol establish a session with the other system before sending data, ensuring accurate transmission.

- 3-way handshake connection process



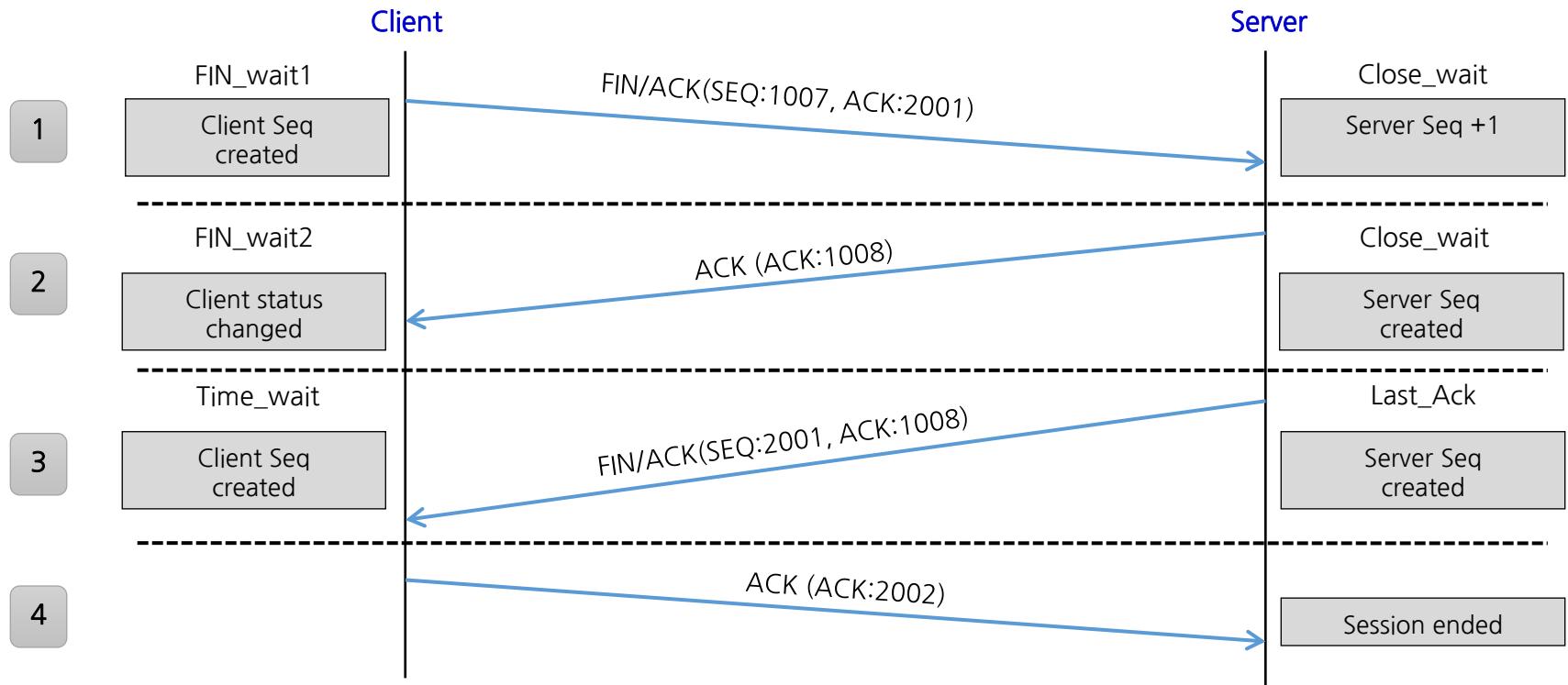
The 3-way handshake is a process by which applications using the TCP/IP protocol establish a session with the other system before sending data, ensuring accurate transmission.

- Data transfer



The 3-way handshake is a process by which applications using the TCP/IP protocol establish a session with the other system before sending data, ensuring accurate transmission.

- 4-way handshake connection process



User Datagram Protocol (UDP) is unreliable and does not guarantee the integrity of the data received, but it has the advantage of not overloading the network.

- Features and functions
 - Do not check for responses from the other party
 - Omit the process of destination system checking data sent by sending the system
 - Have the advantage of not overloading the network
 - Do not guarantee data reliability
 - Do not guarantee the integrity of received data
 - UDP header structure

| 0 | 16 | 31 |
|--------------------------------------|----|---|
| Source Port Number (Src Port) | | Destination Port Number (Dst Port) |
| Length | | UDP checksum |

TCP/UDP

TCP/UDP

User Datagram Protocol (UDP) is unreliable and does not guarantee the integrity of the data received, but it has the advantage of not overloading the network.

| Features/description | UDP | TCP |
|--|---|---|
| General description | Simple, fast, providing an interface for applications to access the network layer, but does little else. | A feature-rich protocol that allows applications to reliably send data without worrying about network layer issues. |
| Establish a protocol connection | Connectionless, sends data without establishing a connection | Connection-oriented, must establish a connection before you can send a connection. |
| Data input interface of the applications | Message-based. The application sends data in separate packages. | Stream-based. The application sends data without a specific structure. |
| Credibility and authorization | Unreliable. Best effort delivery method without authorization | Message delivery is trusted. All data is authorized. |
| Retransmission | Not performed. Applications must detect lost data and retransmit if necessary. | Manage all data transfers and automatically resend lost data |
| Data flow management capabilities | None | Flow control using a sliding window, proper window sizing, and a mixed avoidance algorithm |
| Load | Very low | Low, but higher than UDP |
| Transfer rate | Very fast | Fast, but slower than UDP |
| The right amount of data | Small to medium data (up to a few hundred bytes) | Small to very large data (up to several gigabytes) |
| Types of applications using protocols | Applications where delivery speed is more important than data integrity, sending small amounts of data, and used as multicast/broadcast | Include most protocols that need to send data reliably and most file/message transfer protocols for applications |
| Popular applications and Protocols | Multimedia applications, DNS, BOOTP, DHCP, TFTP, SNMP, RIP, NFS (initial versions) | FTP, Telnet, SMTP, DNS, HTTP, POP, NNTP, IMAP, BGP, IRC, NFS (later versions) |

To communicate, HTTPS issues certificates from Certificate Authorities (CAs) and uses public-key cryptography and symmetric-key cryptography to ensure secure communication between one another.

- Overview

- Short for Security Socket Layer
 - Transport Layer Security (TLS)
- Include 1.0, 2.0, and 3.0, with changes to TLS versions as they become publicly available, up to TLS versions 1.0 through 1.3
- Validate a server with a certificate issuing authority, called a Certificate Authority (CA)
- Use symmetric and public key encryption for performance reasons
 - Symmetric key
 - Perform encryption or decryption using the same key
 - Public key
 - Perform encryption and decryption by exchanging secret and public keys with each other

To communicate, HTTPS issues certificates from Certificate Authorities (CAs) and uses public-key cryptography and symmetric-key cryptography to ensure secure communication between one another.

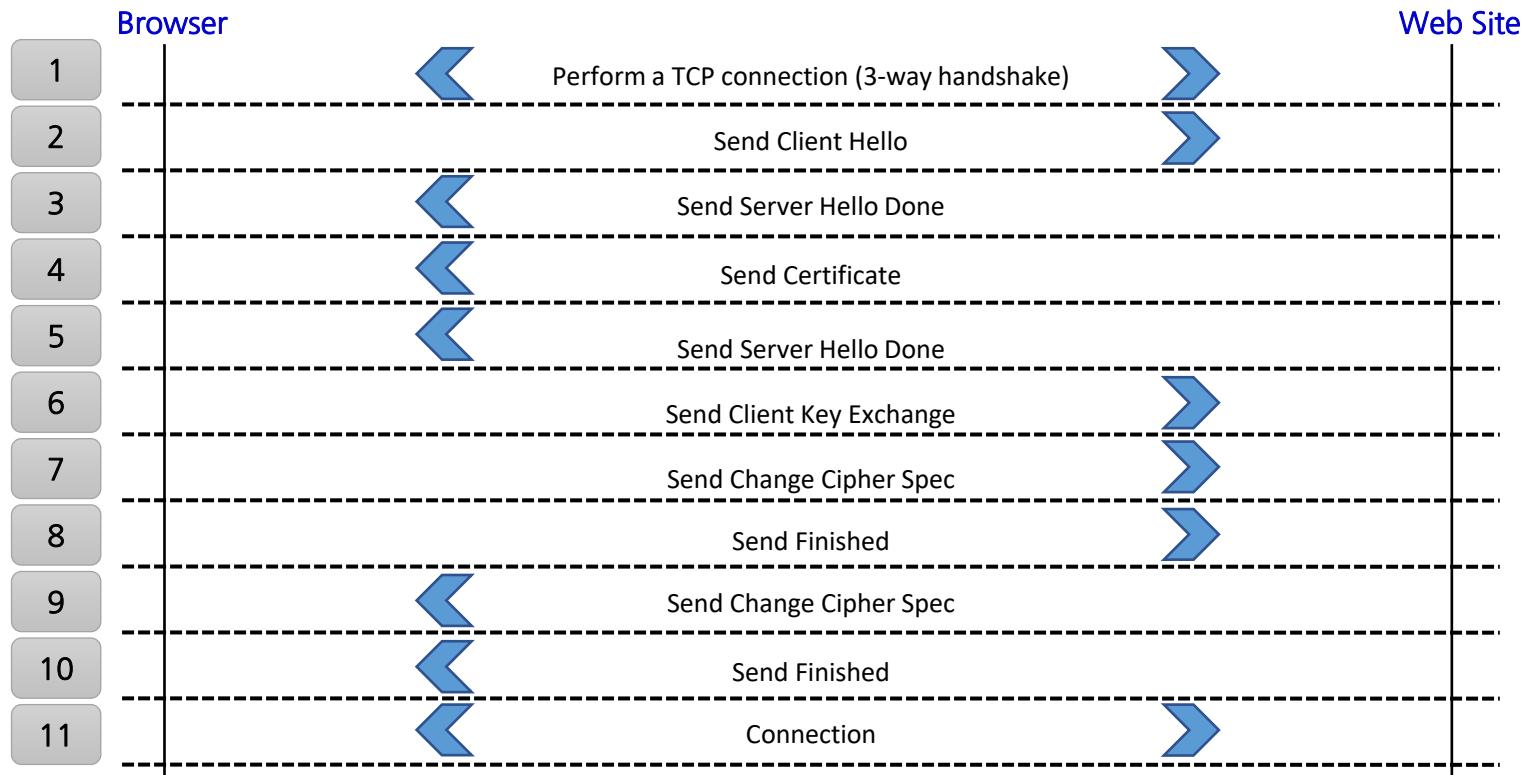
- Communication principles
 - Certificates
 - Issued to a site
 - The site requiring a certificate provides the CA with the site's public key and site information.
 - After validation by the CA, the certificate is issued with the CA's private key and delivered to the site.
 - Users and CAs
 - Users receive CA certificates and CA public keys from the CA.
 - Certificates are sometimes imported during browser or OS installation.

To communicate, HTTPS issues certificates from Certificate Authorities (CAs) and uses public-key cryptography and symmetric-key cryptography to ensure secure communication between one another.

- Communication principles
 - Sites and users
 - After a user requests access to a site, the site issues a certificate issued by the CA.
 - The user decrypts the certificate with the public key obtained from the CA.
 - The user then receives the site's information and public key.
 - The user generates a symmetric key, encrypts it with the site's public key, and sends it.
 - The user decrypts with the site's private key to obtain a symmetric key.
 - They communicate with each other by encrypting with symmetric keys.

The method of communication between the Web site and the user's browser, other than the method of guaranteeing the content of the certificate issued by the CA, is called the SSL/TLS handshake method.

- SSL/TLS handshake method

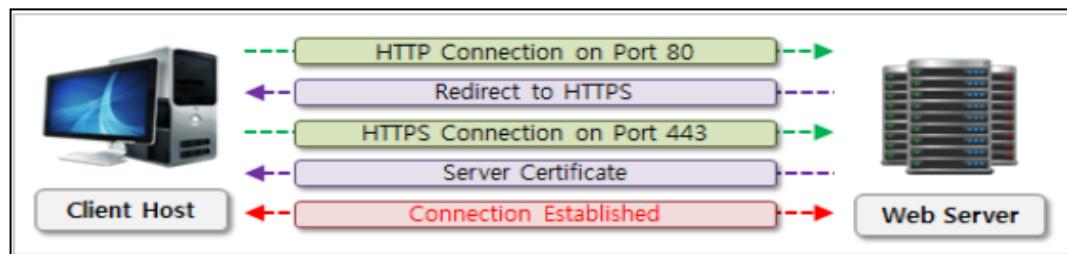


Secure Socket Layer (SSL) provides secure communications over the Internet using an encryption protocol based on electronic signature technology.

- Overview

- Short for HyperText Transfer Protocol over Secure Socket Layer
- Perform encrypted communications using SSL or TLS protocols, which are more secure versions of the HTTP communications protocol
- Developed by Netscape Communications Corporation and widely used in e-commerce

- Communication principles

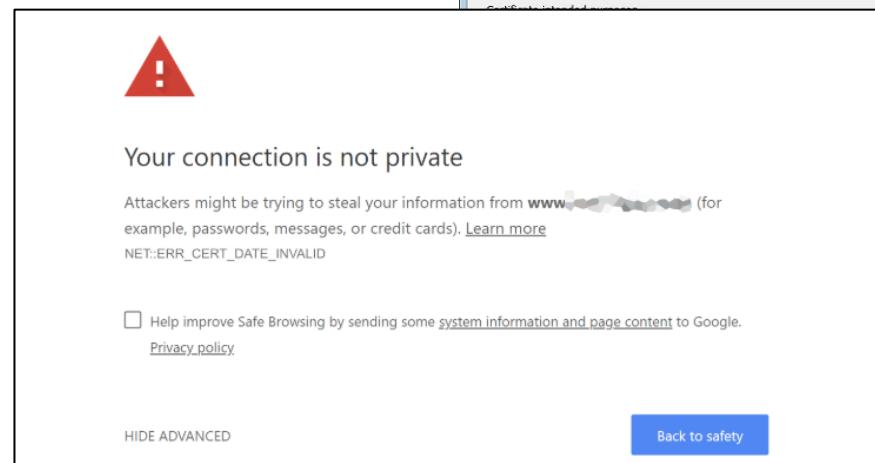
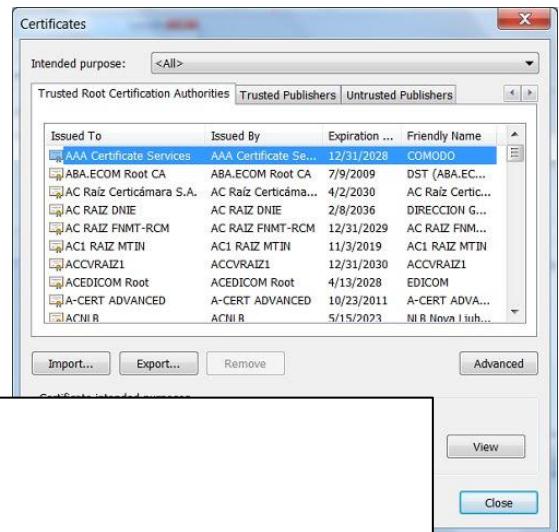


- SSL stands for Secure Socket Layer, an Internet protocol for secure data transmission over the Internet.
- When the client connects to the server, it is redirected to the HTTPS port.
- The client connects to HTTPS port 443.

To communicate, HTTPS issues certificates from CAs and uses public-key cryptography and symmetric-key cryptography to ensure secure communication between one another.

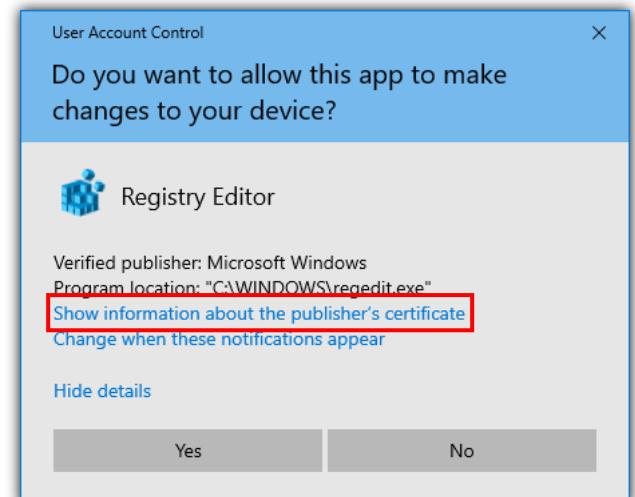
- Certificates

- Check saved browsers
 - For Chrome
 - Settings → Privacy and security → Security → Manage certificates
- Verify connection with a warning if the certificate is not trusted



To communicate, HTTPS issues certificates from CAs and uses public-key cryptography and symmetric-key cryptography to ensure secure communication between one another.

- Certificates
 - Certificates in applications
 - Embed certificates in .exe files to ensure the trustworthiness of the programs you use.
 - E.g., antivirus programs
 - Untrusted program warning (UAC, User Account Control) pops up when running programs that don't contain certificates for some non-exempt programs.



While SSH and SSL share the commonality of using encrypted communication, it's important to understand the differences and use them appropriately.

- Myths and truths
 - SSH and SSL
 - SSH
 - SSH stands for Secure Shell, which has a username and password authentication system to establish a connection.
 - It uses port 22 and is designed to securely execute commands over the Internet.
 - SSL
 - SSL is a structure that uses a special protocol layer to provide encryption and security features.
 - SSL securely transmits sensitive information, such as credit card and banking information.
 - Unlike SSH, it does not require authentication.

Virtual private networks (VPN) protocols

Understanding VPN

A VPN, or virtual private network, creates a private network connection between devices over the Internet. It transfers data securely and anonymously over a public network.

- Virtual Private Networks (VPN)
 - Mask IP addresses and encrypt data to make it unreadable to unauthorized parties
- Three main functions of a VPN
 - Privacy Policy
 - Record and potentially sell personal information such as passwords, credit card information, and browsing history to third parties
 - Use encryption to keep sensitive information private when connecting over public Wi-Fi networks
 - Anonymity
 - Every website on the Internet uses cookies and similar technologies to track information.
 - A VPN connection hides your IP address to keep you anonymous on the Internet.
 - Security
 - Use encryption to protect your Internet connection from unauthorized access
 - Act as a shutdown mechanism to terminate programs in case of suspicious Internet activity

Virtual private networks (VPN) protocols

Protocols by VPN layer

A VPN, or virtual private network, creates a private network connection between devices over the Internet. It transfers data securely and anonymously over a public network.

| Layer | Protocol | Description |
|-------|----------|--|
| 2 | L2F | <ul style="list-style-type: none">- Short for Layer 2 Forwarding- Access from a remote ISP device, create an L2F tunnel through the tunnel server on the server side |
| | PPTP | <ul style="list-style-type: none">- Short for Point-to-Point Tunneling Protocol- Extended PPP, act in a client/server fashion |
| | L2TP | <ul style="list-style-type: none">- Short for Layer 2 Tunneling Protocol- PPTP + L2F; operate even in non-IP network environments- Multi-protocol support : IP, IPX, etc.- All on X.25/ATM/Frame Relay/SONET |
| | MPLS | <ul style="list-style-type: none">- Multi Protocol Label Switching- Prefix IP packets with labels and send them through label switching |
| 3 | IPSec | <ul style="list-style-type: none">- The de facto standard for VPN tunneling- Use as a security protocol in IPv6- Integrity/authentication : Authentication Header (AH) protocol- Confidentiality : Encapsulating Security Payload (ESP) |
| 4 | SSL | <ul style="list-style-type: none">- Short for Security Socket Layer- Encryption protocol via handshake between client and server- Authentication : |

Network feature

- Overview
- Routing and port forwarding
- NAT
- Tunneling

Overview

- Routing
 - Routing is responsible for setting up paths on the network.
 - Without routing, you can't communicate with other networks.
 - It is essential for smooth network communication.
 - Dynamic vs. static routing
- NAT
 - NAT is a concept that arose from the IPv4 exhaustion problem.
 - Short for Network Address Translation
- Tunneling
 - Tunneling is a technology that creates virtual pipes, or tunnels, between specific areas of data communications on the Internet.
 - It is used to hide the content of traffic.

Routing and port forwarding

Router overview

A router is a device that performs a routing function, connecting different networks by forwarding data packets after establishing a route using routing information.

- What is a router?
 - An Internet networking device that connects a LAN to a LAN or a LAN to a WAN.
 - Specify a path for data transmission
 - Forward IP packets to their destination via routing protocols
- Router features

| Function | Description |
|--------------------|--|
| Path determination | <ul style="list-style-type: none">• Examine the path a data packet can take from its source to its destination• Decide which path is best to take |
| Switching | <ul style="list-style-type: none">• Performs data packet switching operations when routing is determined |



Source : Cisco

Routing and port forwarding

Router overview

Routing is the process of determining the destination IP of a packet and forwarding it to its intended destination. There are two types of routing : static and dynamic.

- Routing types

| Function | Description |
|-----------------|--|
| Static routing | <ul style="list-style-type: none">• Administrator manually enters path.• Secure because routing information doesn't go anywhere else• Can reduce the overhead of dynamic routing• Static routing is useful when there's only one way to go from the node's point of view. |
| Dynamic routing | <ul style="list-style-type: none">• Determine path based on routing protocol• Highly customizable as routes are determined by communicating with other routers• Easy to manage if you have many routers |

Routing and port forwarding

Routing algorithm types

Types of dynamic routing algorithms include distance vector and link-state, and types of dynamic routing protocols include RIP, IGRP, and EIGRP.

- Distance vector vs. link-state algorithms

| Item | Distance vector algorithm | Link state algorithm |
|------------------------------|--|--|
| Advantage | <ul style="list-style-type: none">• Smaller routing table size saves memory.• Routing configuration is simple. | <ul style="list-style-type: none">• Calculate path based on distance and bandwidth• Forward routing information as it changes |
| Disadvantage | <ul style="list-style-type: none">• Waste traffic by periodically updating routing information• Deliver slowly when routing information changes | <ul style="list-style-type: none">• Manage all routing information, which consumes a lot of memory• CPU-intensive, including SPF computations |
| Fit for use | <ul style="list-style-type: none">• Best for small networks | <ul style="list-style-type: none">• Best for large networks |
| How paths are calculated | <ul style="list-style-type: none">• By hop count | <ul style="list-style-type: none">• By considering hops, latency, bandwidth, and other variables |
| Routing update information | <ul style="list-style-type: none">• Periodic routing table updates | <ul style="list-style-type: none">• Event-driven routing table updates |
| Routing information exchange | <ul style="list-style-type: none">• Update routing tables with neighboring routers | <ul style="list-style-type: none">• Exchange link-state information with neighboring routers |
| Typical routing protocols | <ul style="list-style-type: none">• RIP, IGRP | <ul style="list-style-type: none">• EIGRP, OSPF, IS-IS |

Routing and port forwarding

Routing protocols

Types of dynamic routing algorithms include distance vector and link-state, and types of dynamic routing protocols include RIP, IGRP, and EIGRP.

- Routing Information Protocol version 1 (RIPv1)
 - RIPv1 is a dynamic routing network that only needs to register the networks of the connected routers to be automatically forwarded to the routing table.
 - Have the advantage of easy installation, easy scalability, and fast routing information updates
 - Organizes the network on a class basis, cannot be divided by host
 - Class basis (ClassFull) : the routing protocol does not include subnet masks in routing updates.
 - Use broadcast addresses to send routing information

Routing and port forwarding

Routing protocols

Types of dynamic routing algorithms include distance vector and link-state, and types of dynamic routing protocols include RIP, IGRP, and EIGRP.

- Routing Information Protocol version 2 (RIPv2)
 - Extensions to RIPv1
 - Added message format : Version2, pathtag, subnet mask, and next-hop
 - VLSM support
 - Use of multicast addresses to send routing information
 - Authentication of routing information

Routing and port forwarding

Routing protocols

Types of dynamic routing algorithms include distance vector and link-state, and types of dynamic routing protocols include RIP, IGRP, and EIGRP.

- Open Shortest Path First (OSPF)
 - Shortest Path First (SPF) algorithm
 - Link-state routing protocol
 - Designed to overcome the limitations of RIP
 - Variable Length Subnet Mask (VLSM) and Classless Inter-network Domain Routing (CIDR) support
 - Use the multicast address (224.0.0.5)
 - Use the bandwidth-based 'cost' value as the routing method

Routing and port forwarding

Routing protocols

Types of dynamic routing algorithms include distance vector and link-state, and types of dynamic routing protocols include RIP, IGRP, and EIGRP.

- Enhanced Interior Gateway Routing Protocol (EIGRP)
 - Distance vector routing protocol developed by Cisco
 - Use IP protocol number 88 to transmit routing information
 - Support for load balancing
 - Classless routing protocols

NAT is a popular technology that allows multiple hosts on a private network to access the Internet using a single public IP address. It uses IP addresses to send and receive traffic through network devices by rewriting them.

- NAT overview

- Short for Network Address Translation
- A term used in computer networking to describe a technique for sending and receiving traffic through network devices by rewriting the TCP/UDP port numbers of IP packets and the source and destination IP addresses.
- Often used when multiple hosts on a private network access the Internet from a single public IP address.

- NAT classification

- Cone NAT
 - Full cone
 - Restricted cone
 - Port restricted cone
- Symmetric NAT

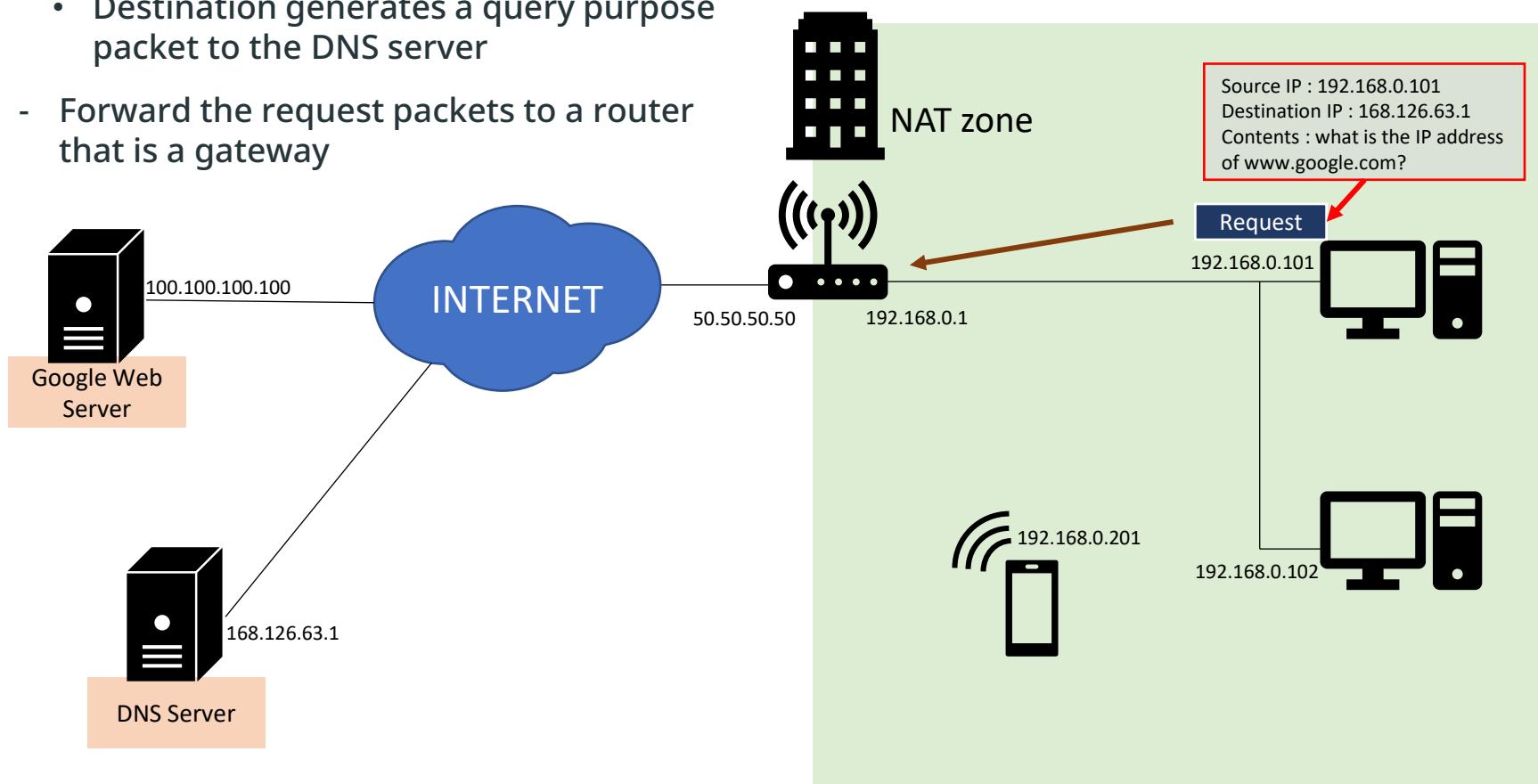
NAT is a popular technology that allows multiple hosts on a private network to access the Internet using a single public IP address. It uses IP addresses to send and receive traffic through network devices by rewriting them.

- NAT classification
 - Cone NAT
 - Map internal IPs and ports to the same external ones regardless of destination
 - Full cone
 - Connect to NAT client only if remote address/port is different in request and response
 - Restricted cone
 - Connect to NAT client only if the remote address/port is the same in the request and response
 - Port restricted cone
 - No connection if remote address/port is different in request and response
 - Symmetric NAT
 - No connection if the remote address/port is different in request and response
 - Yet, if they are different, assign a new port that is the client's address/port and use it.

NAT

Example of the NAT communication process

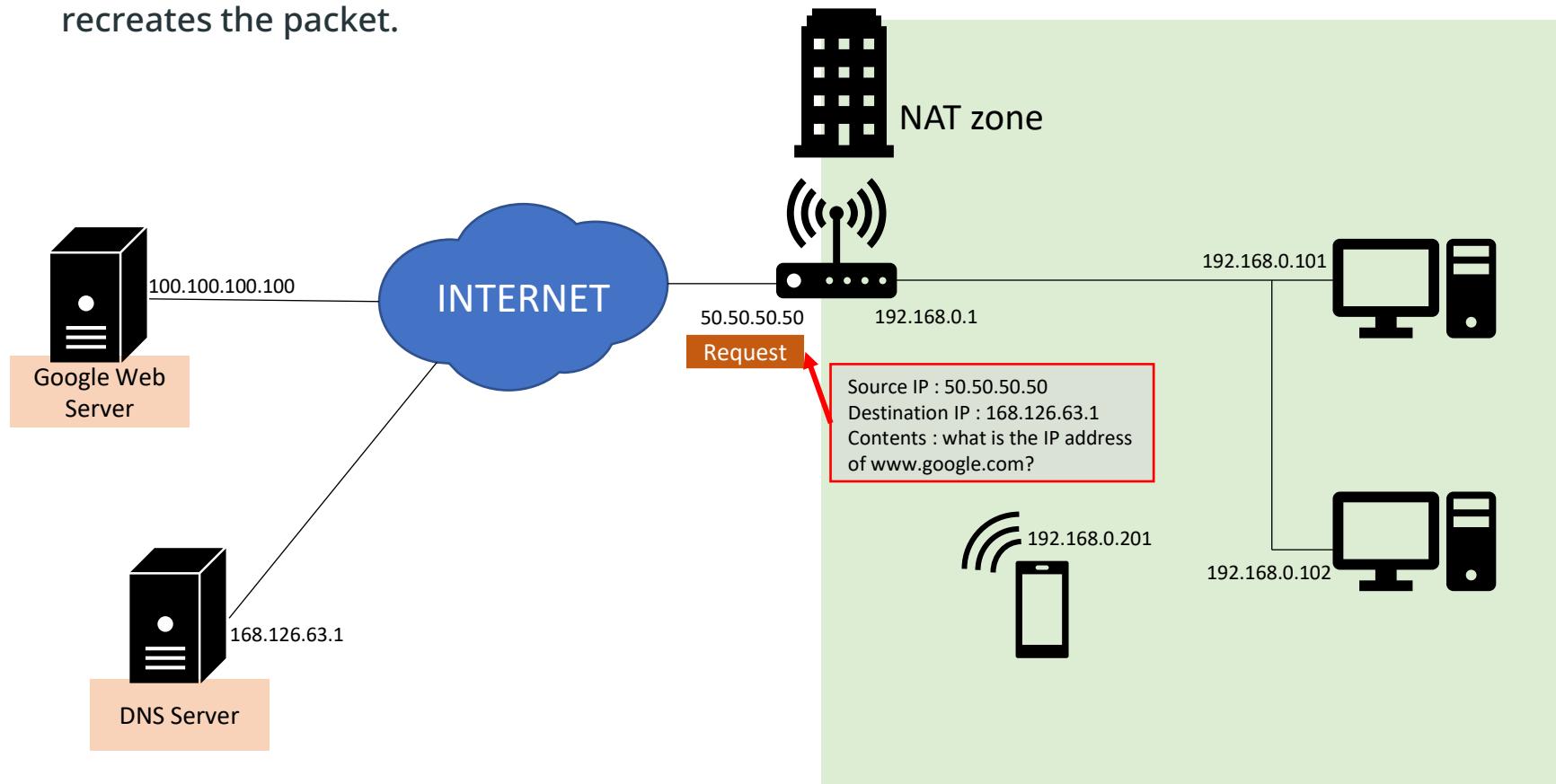
- Query what the IP address of the URL www.google.com is
 - Generate a DNS request packet asking for the IP address of the URL
 - Destination generates a query purpose packet to the DNS server
 - Forward the request packets to a router that is a gateway



NAT

Example of the NAT communication process

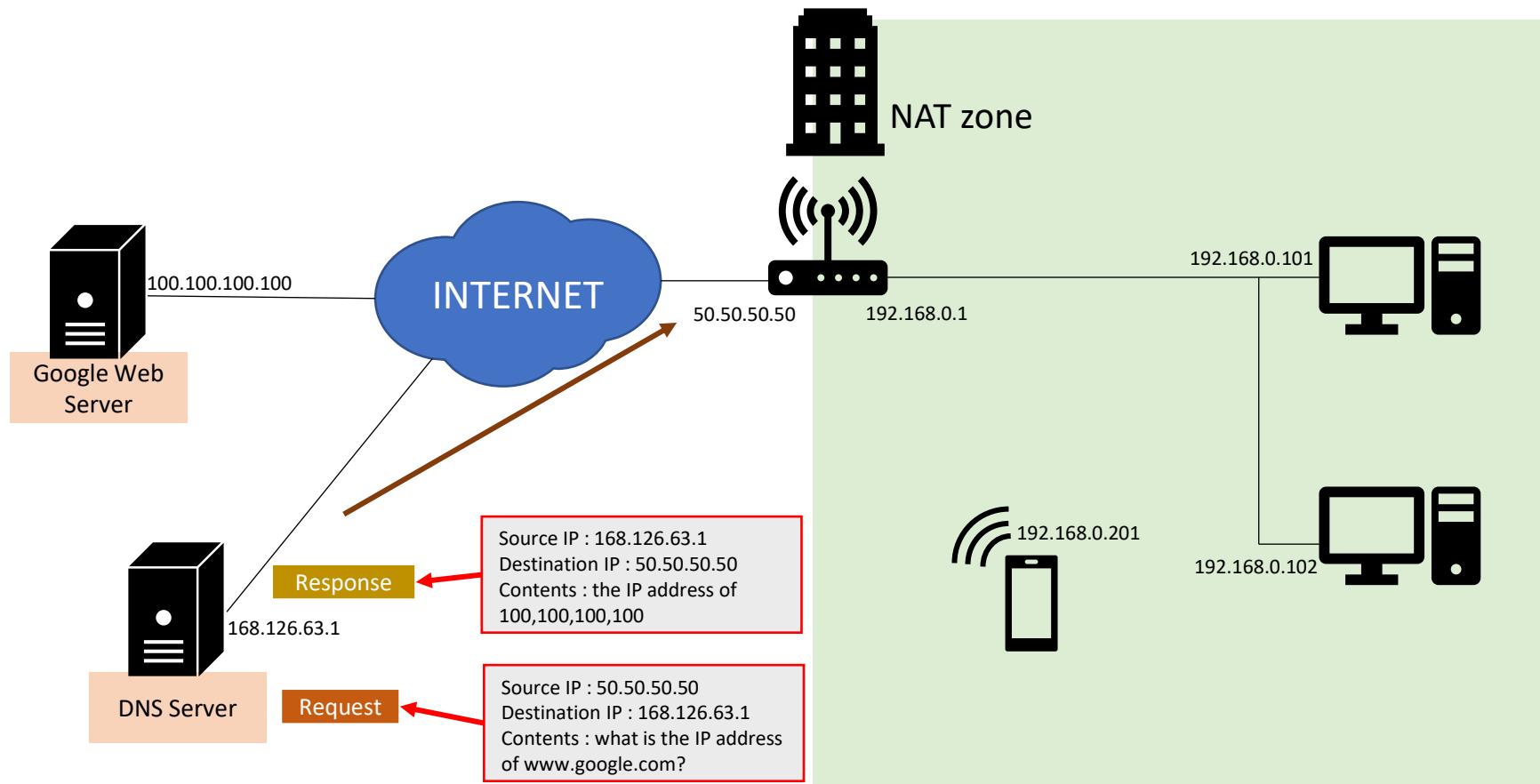
- Query what the IP address of the URL www.google.com is
 - The router receiving the packet changes its external IP to the source IP and recreates the packet.



NAT

Example of the NAT communication process

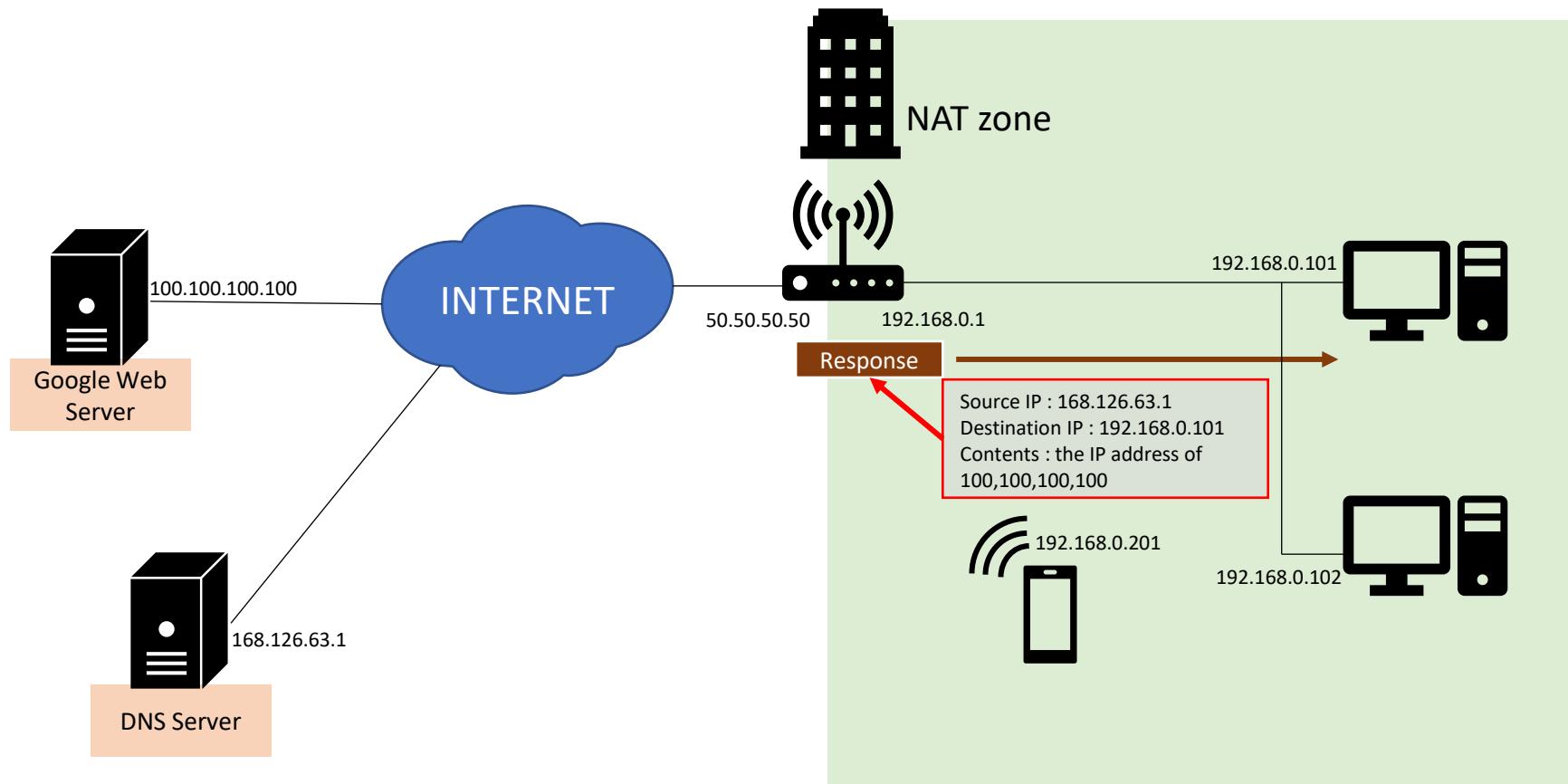
- Query what the IP address of the URL www.google.com is
 - For each request packet it receives, the DNS server generates and forwards a response packet containing data from Google's IP address.



NAT

Example of the NAT communication process

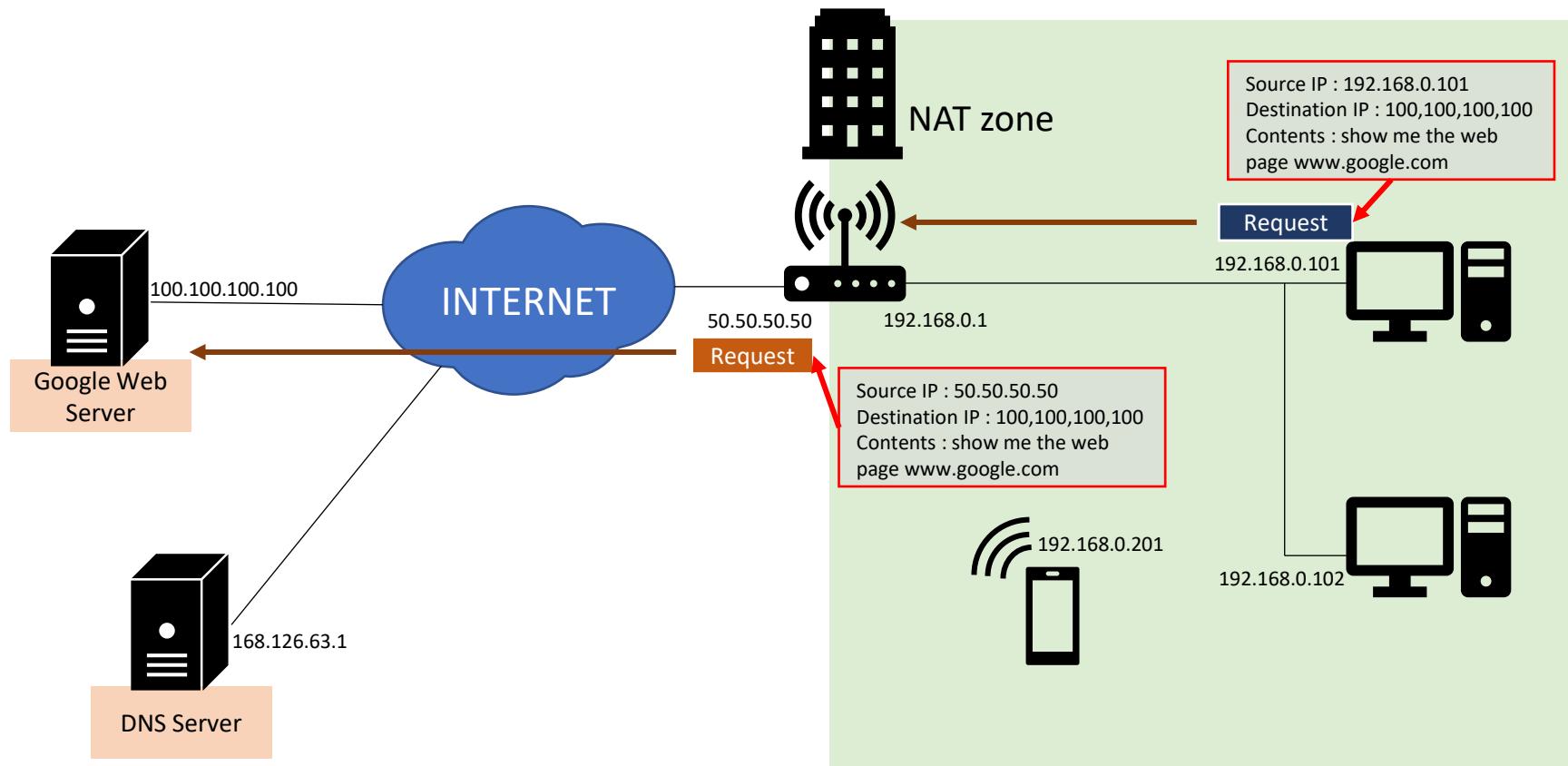
- Query what the IP address of the URL www.google.com is
 - The router that receives the response packet regenerates the packet with its internal IP and forwards it to the requesting PC.



NAT

Example of the NAT communication process

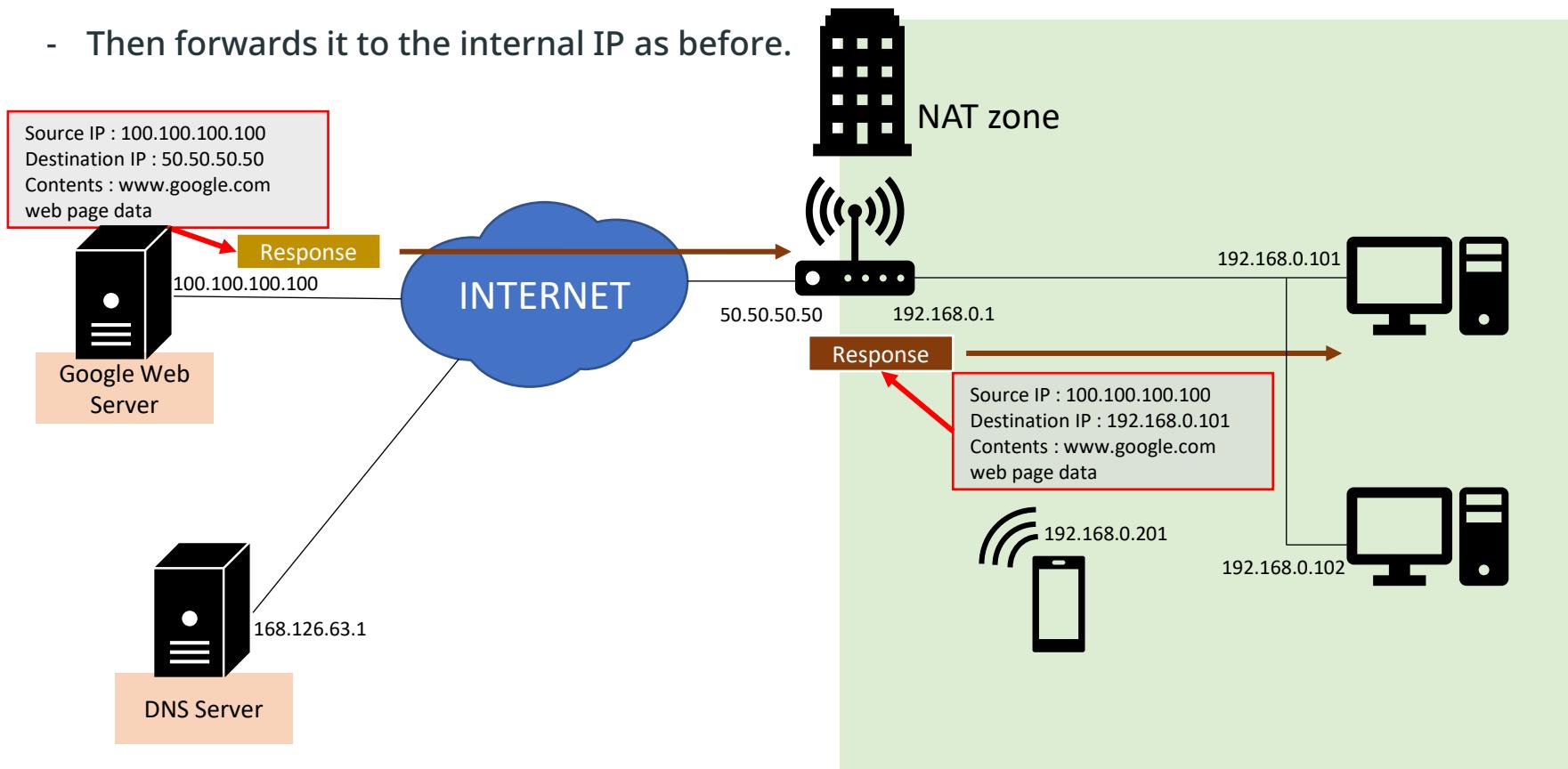
- Query what the IP address of the URL www.google.com is
 - The router creates a request packet that requests a web page from the Google address and routes it just like a DNS request.



NAT

Example of the NAT communication process

- Query what the IP address of the URL www.google.com is
 - The Google web server receives a request packet and generates and sends a response packet to the external IP address.
 - Then forwards it to the internal IP as before.



Tunneling is a technique for communicating with a specific area of the Internet by creating a tunnel, a virtual pipe, to carry data streams across the Internet. It is one of the technologies used to secure traffic by accessing private networks or performing encrypted communications.

- Definition
 - Definitions in telecommunications
 - A technology that carries streams of data through virtual pipes on the Internet
- Techniques
 - A technique used to create a virtual connection (such as a VPN) between two nodes or two networks.
 - Mostly act as a secure channel
 - Encryption-based enforcement
 - Involve repackaging traffic data into a different format that can hide the nature of the traffic

Tunneling

Type

Tunneling is a technique for communicating with a specific area of the Internet by creating a tunnel, a virtual pipe, to carry data streams across the Internet. It is one of the technologies used to secure traffic by accessing private networks or performing encrypted communications.

- Key security methods
 - Two-layer tunneling method
 - Types : PPTP, L2TP, MPLS, etc.
 - Implement tunneling through PPP extensions : PPTP, L2TP, etc. (no data encryption)
 - PPTP : developed by Microsoft and standardized in RFC 2637
 - Require IPSec to add encryption beyond tunneling
 - L2TP : standardized with enhancement to PPTP (RFC 2661, 3931)
 - Require MPPE from MS to apply encryption
 - Used in early VPN implementations, now replaced by IPSec, SSL/TLS, etc.
 - MPLS : implement network-based tunneling

Tunneling

Type

Tunneling is a technique for communicating with a specific area of the Internet by creating a tunnel, a virtual pipe, to carry data streams across the Internet. It is one of the technologies used to secure traffic by accessing private networks or performing encrypted communications.

- Key security methods
 - Three-layer tunneling method
 - IPSec, GRE, and more
 - GRE : tunneling is present, but no encryption is applied.
 - Combine GRE and IPSec for security when data encryption is required
 - IPSec : both encryption and tunneling are present.
 - Four-layer tunneling method
 - SSL/TLS, SSH, SOCKS v5, and more
 - Establish tunneling on a per-session basis between two nodes within a client
 - Create primarily TCP secure channels

Tunneling

Type

Tunneling is a technique for communicating with a specific area of the Internet by creating a tunnel, a virtual pipe, to carry data streams across the Internet. It is one of the technologies used to secure traffic by accessing private networks or performing encrypted communications.

- Key security methods
 - Comprehensive chart organized by layer

See also: http://www.ktword.co.kr/word/abbr_view.php?m_temp1=1708

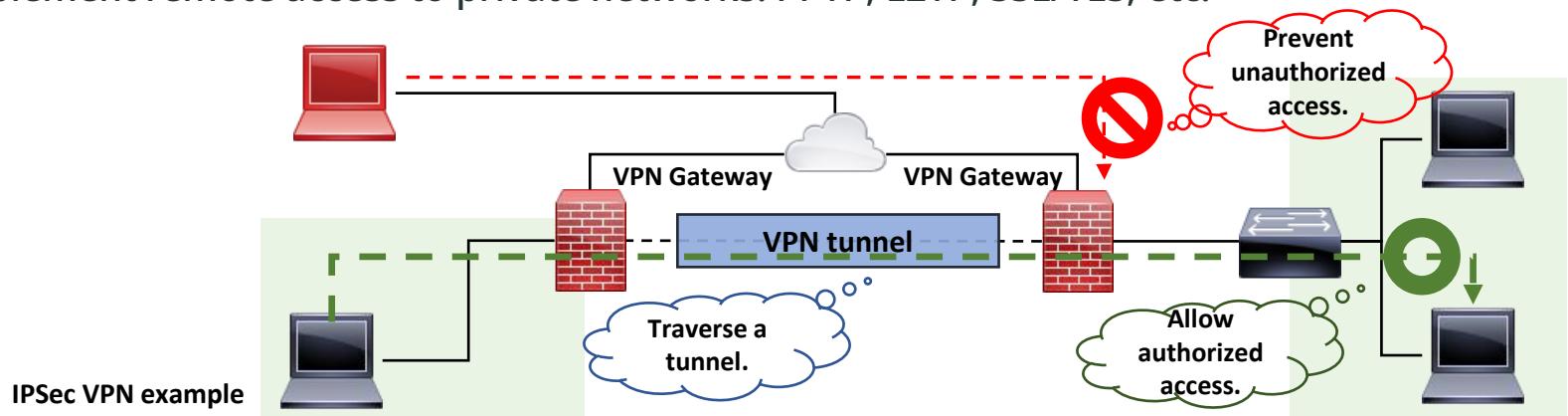
| Division | PPTP | L2TP | IPSec | SOCKS v5 |
|---------------------------|------------------|------------------|------------------|-------------------|
| Standard | MS | RFC 2661 | RFC 2401 to 2410 | RFC1928,1929,1961 |
| Implemented layer | Layer 2 | Layer 2 | Layer 3 | Layer 5 |
| Action modes | Client/server | Client/server | Peer-to-peer | Client/server |
| Related protocols | IP, IPX, NetBEUI | IP, IPX, NetBEUI | IP | TCP, UDP |
| Authentication | X | X | X | O |
| Encryption key management | X | X | ISKMP/IKE | GSS-API/SSL |

Tunneling

Type

Tunneling is a technique for communicating with a specific area of the Internet by creating a tunnel, a virtual pipe, to carry data streams across the Internet. It is one of the technologies used to secure traffic by accessing private networks or performing encrypted communications.

- Key security methods
 - Forms of VPN implementation through tunneling
 - Virtual Private Network (VPN) : a private network implemented virtually over the Internet.
 - A technology that allows traffic to pass securely over the Internet to what amounts to a private network.
 - GRE, etc.
 - Implement remote access to private networks: PPTP, L2TP, SSL/TLS, etc.



Tunneling

GRE over IPsec overview

IPSEC serves to prevent data tampering in packets through encryption applied between network communications.

- IP Security (IPSec)
 - At the network layer (IP layer), a protocol for authentication, encryption, and key management on a per-IP-packet basis.
 - No support for multicast & broadcast to update routing using dynamic routing protocols
 - Used with GRE where packet encryption is not possible

