Network Programming

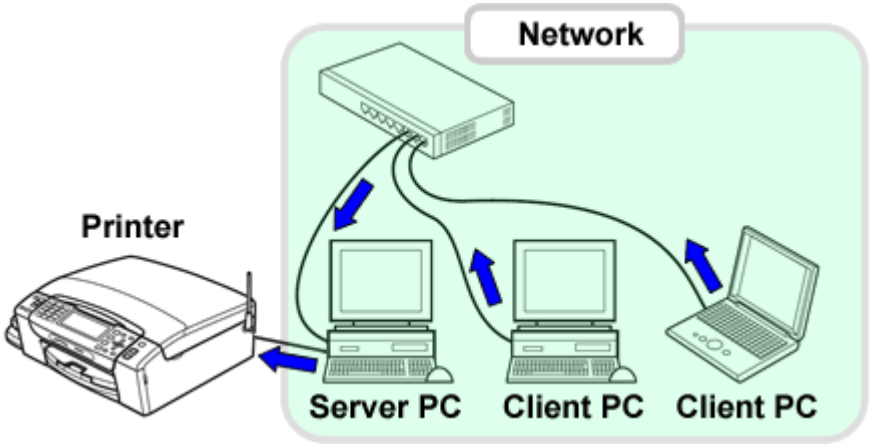
Ung Văn Giàu
Email: giau.ung@eiu.edu.vn

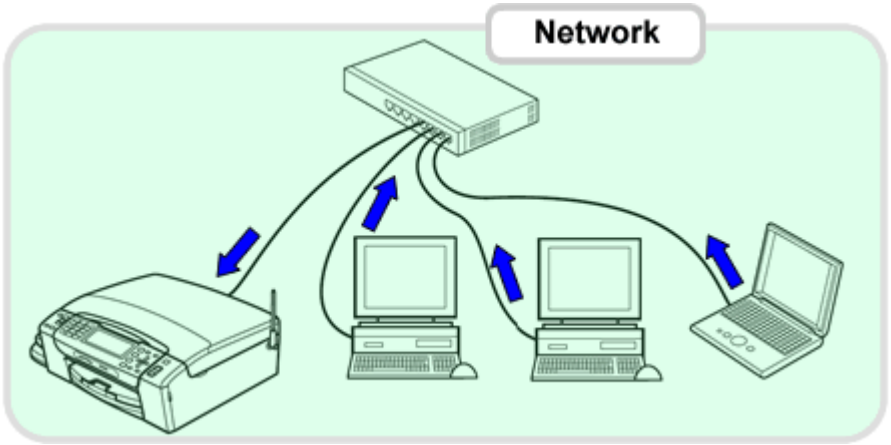
Contents

Introduction to network application and programming

What is a network program / application?

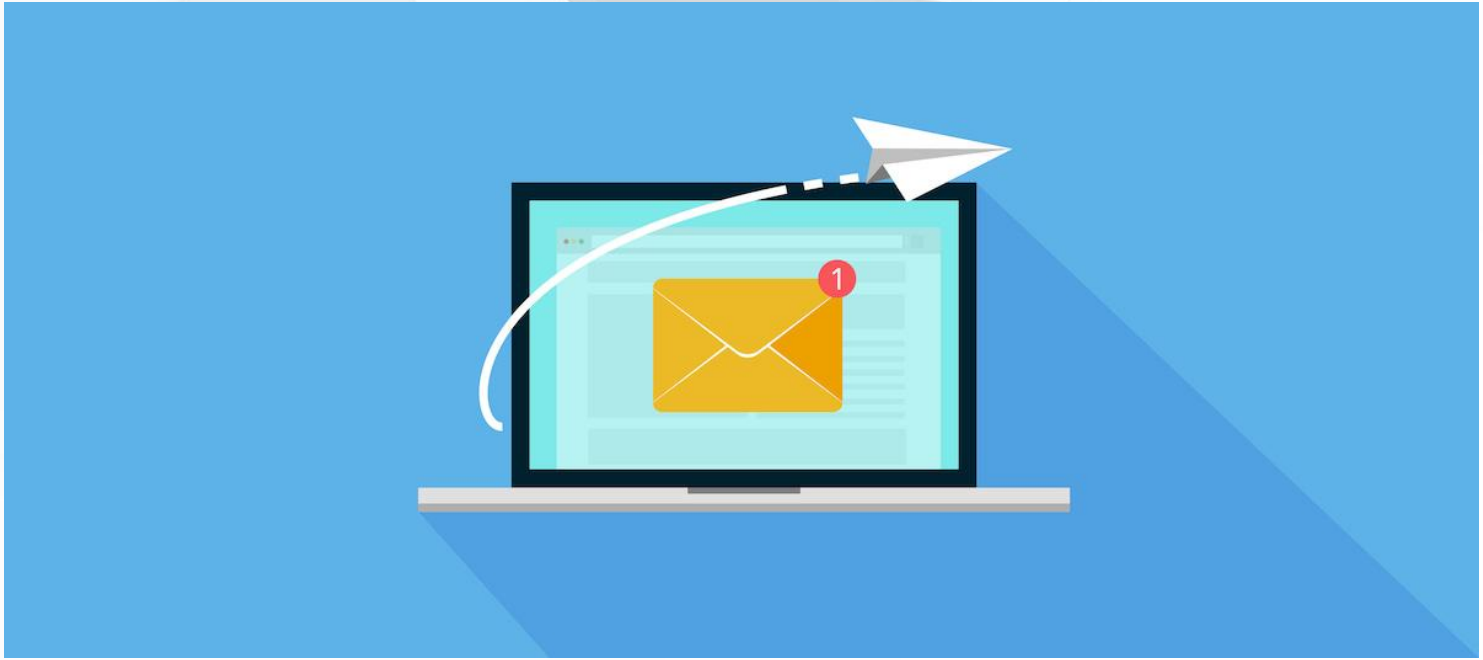
Example 1: Network Shared Printing





What is a network program / application?

Example 2: Email Delivery



What is a network program / application?

- A **network program** is any application that **uses a computer network to transfer information** to and from other applications

- Example:
  - Web browsers: Chrome, Firefox,…
  - Email programs: MS Outlook, Mozilla Thunderbird,…
  - Social Network Applications: Facebook, Zalo,…

What is a network program / application?

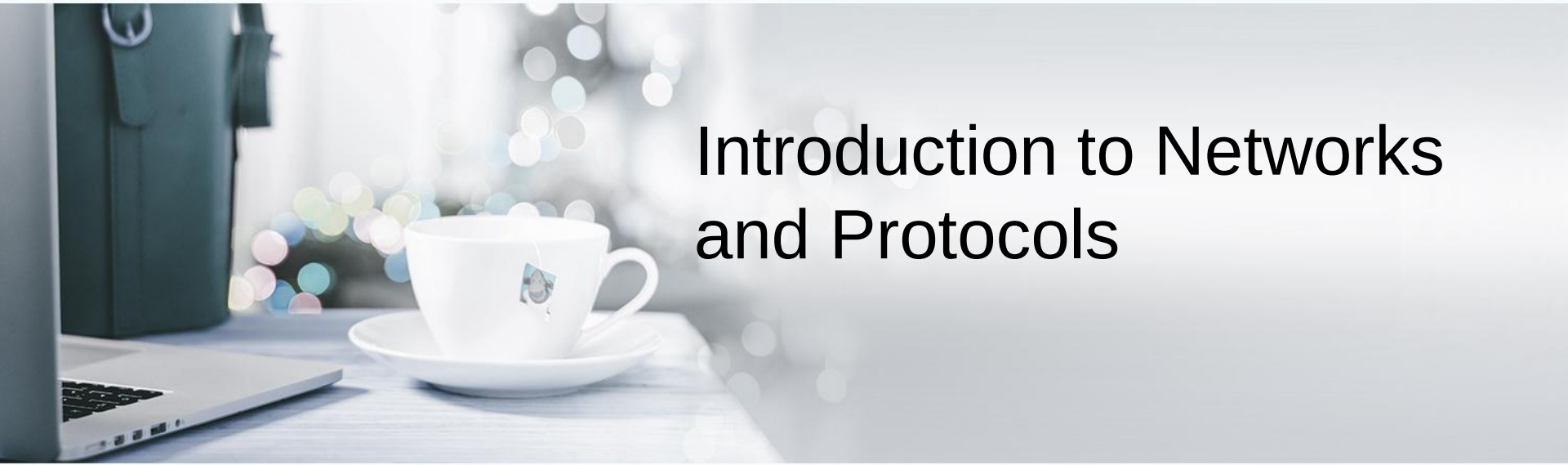
- The apps are classified broadly into 2 kinds:
  - **Client applications** request information
  - **Server applications** respond to information requests from clients

- The client app makes a request by identifying the requested Internet resource and the communication protocol
- If necessary, the client also provides any additional data required:
  - proxy location
  - authentication information

Network Programming

- Network Programming involves **writing programs** that **communicate** with other programs **across** a computer **network.**

- Design network application:
  - Security
  - High performance
  - Scalability



Introduction to Networks and Protocols

- Millions of desktops, laptops, routers, and servers are connected to the internet.
- Billions of additional devices are now connected as well—mobile phones, tablets, gaming systems, vehicles, refrigerators, television sets, industrial machinery, surveillance systems, doorbells, and even light bulbs.

→ **Internet of Things** (IoT) trend

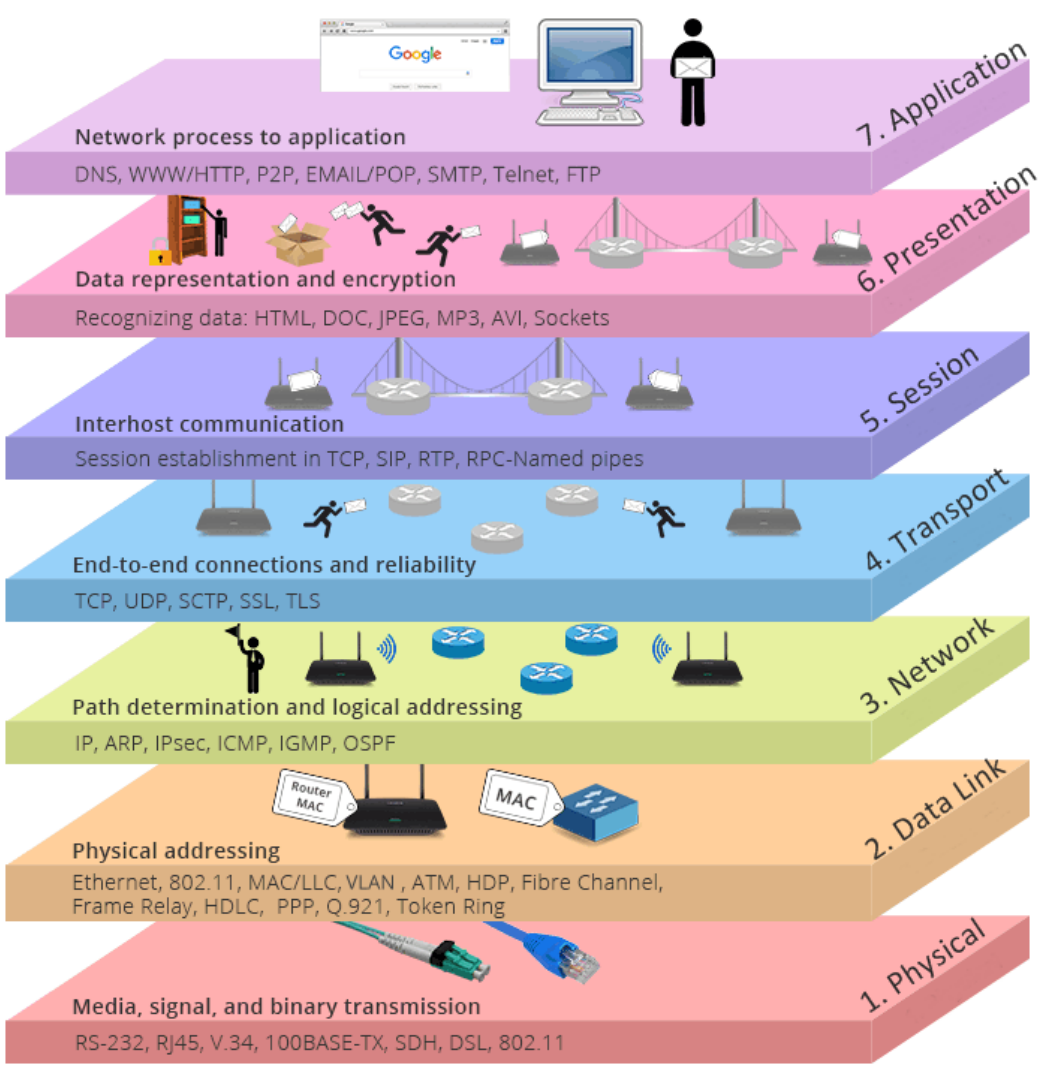
- These devices use a wide variety of hardware:
  Ethernet connection, Wi-Fi, cellular, phone line, fiber optics,…

1. OSI layer model

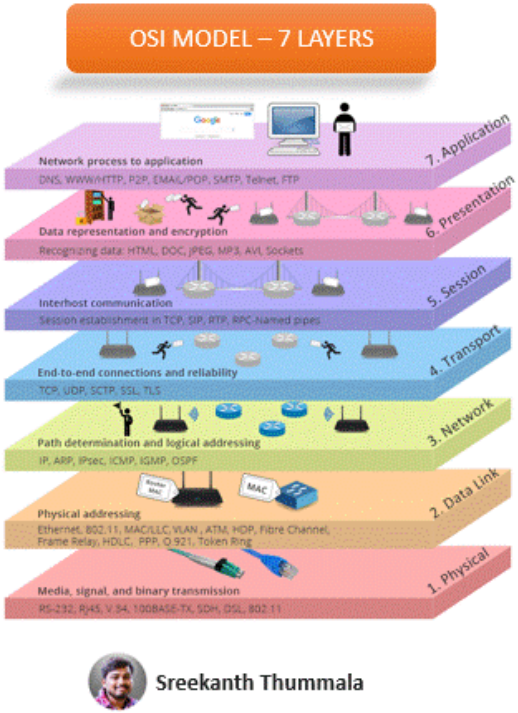
- All of the disparate devices composing the internet are going to communicate seamlessly → there must be **agreed-upon standards** that define their communications.
- These standards are called **protocols**.

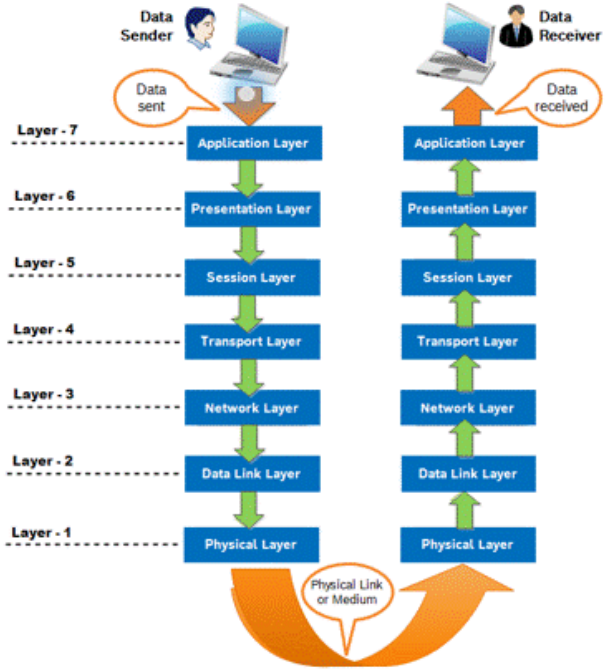
1. OSI layer model

- The most popular layer system for networking is called the **Open Systems Interconnection** model (OSI model).
- It was standardized in 1977 and is published as ISO 7498.
- It has 7 layers.



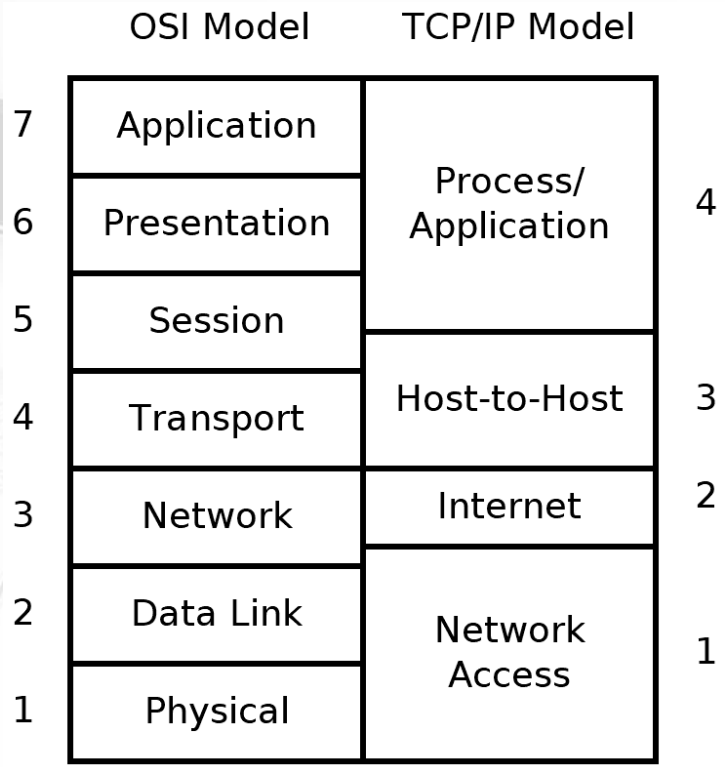
1. OSI layer model



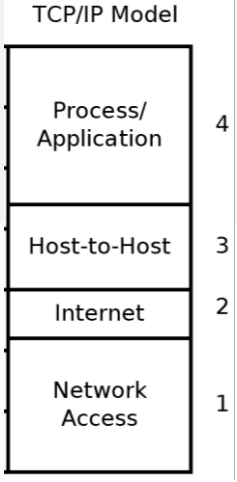


2. TCP/IP layer model

- The TCP/IP protocol suite is the most common network communication model in use.
- It has only four layers.



2. TCP/IP layer model



The four layers of the TCP/IP model are as follows:

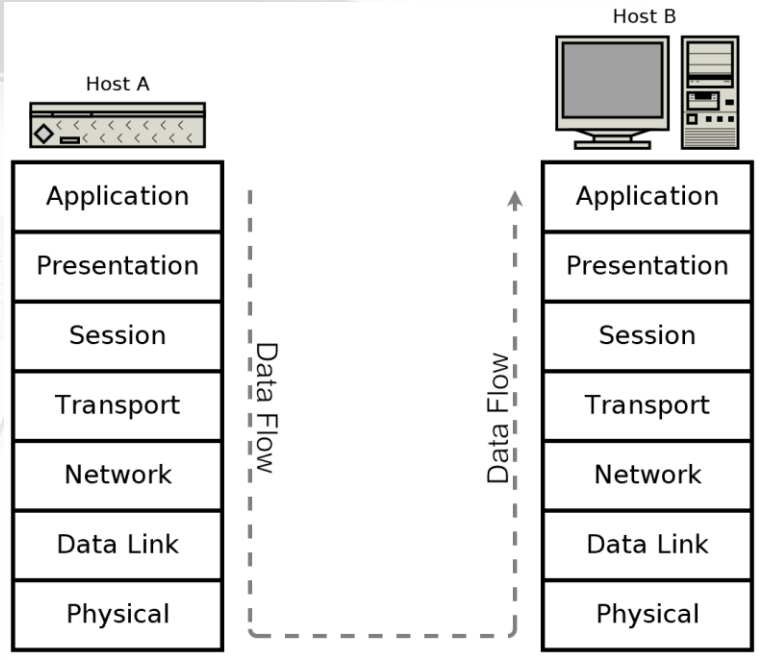
- **Network Access layer (1):** physical connections and data framing happen.
- **Internet layer (2):**
  - deals with the concerns of addressing packets and routing them over multiple interconnection networks.
  - An IP address is defined.
- **Host-to-Host layer (3):** provides two protocols, TCP and UDP. These protocols address concerns such as data order, data segmentation, network congestion, and error correction.
- **Process/Application layer (4):** protocols such as HTTP, SMTP, and FTP are implemented.

3. Data encapsulation

- Lower levels are responsible for handling data for the higher levels → these lower-level data structures must encapsulate data from the higher levels.

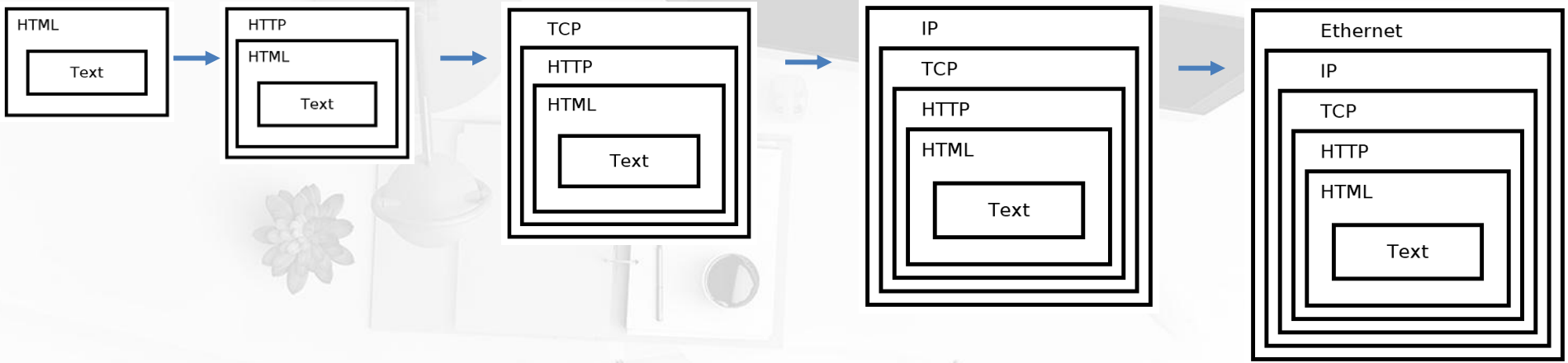
→ when programming an application, only need to consider the highest-level protocol.

- When communicating over a network, data must be processed down through the layers at the sender and up again through the layers at the receiver.



3. Data encapsulation

- For example:



4. Internet Protocol

- One protocol is overwhelmingly common.
- It comes in two versions—IPv4 and IPv6
- IPv4:
  - IPv4 is completely ubiquitous and deployed everywhere.
  - IPv4 uses 32-bit addresses, which limits it to addressing no more than 2^{32} or 4,294,967,296 systems.
- IPv6:
  - IPv6 was designed to replace IPv4
  - It uses a 128-bit address, $2^{128} \approx 3.4 \times 10^{38}$ addresses.

5. What is an address?

- All Internet Protocol traffic routes to an address.
- IPv4 addresses:
  - are 32 bits long
  - are commonly divided into four 8-bit sections.

    Each section is displayed as a decimal number between 0 and 255 inclusive and is delineated by a period.

5. What is an address?

- Examples of IPv4 addresses:
  - 0.0.0.0
  - 127.0.0.1: loopback address (establish a connection to myself)
  - 10.0.0.0 → 10.255.255.255: for private use (local ip)
  - 172.16.0.0 → 172.31.255.255: for private use (local ip)
  - 192.168.0.0 → 192.168.255.255: for private use (local ip)
  - 192.168.50.1
  - 255.255.255.255

5. What is an address?

- IPv6 addresses:
  - are 128 bits long.
  - are written as eight groups of four hexadecimal characters delineated by colons.

    A hexadecimal character can be from 0-9 or from a-f.
- Examples:
  - 0000:0000:0000:0000:0000:0000:0000:0001
  - 2001:0db8:0000:0000:0000:ff00:0042:8329
  - fe80:0000:0000:0000:75f4:ac69:5fa7:67f9
  - ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

6. Domain names

- The Internet Protocol can only route packets to an IP address.
- If you try to connect to a website, your system must **first resolve that domain name into an IP address** for the server that hosts that website.
- This is done by connecting to a **Domain Name System** (DNS) server. The IP address for a domain name server is usually assigned by your ISP.

6. Domain names

Free and public DNS servers

DNS Provider	IPv4 Addresses	IPv6 Addresses
Cloudflare 1.1.1.1	1.1.1.1	2606:4700:4700::1111
	1.0.0.1	2606:4700:4700::1001
FreeDNS	37.235.1.174	
	37.235.1.177	
Google Public DNS	8.8.8.8	2001:4860:4860::8888
	8.8.4.4	2001:4860:4860::8844
OpenDNS	208.67.222.222	2620:0:ccc::2
	208.67.220.220	2620:0:ccd::2

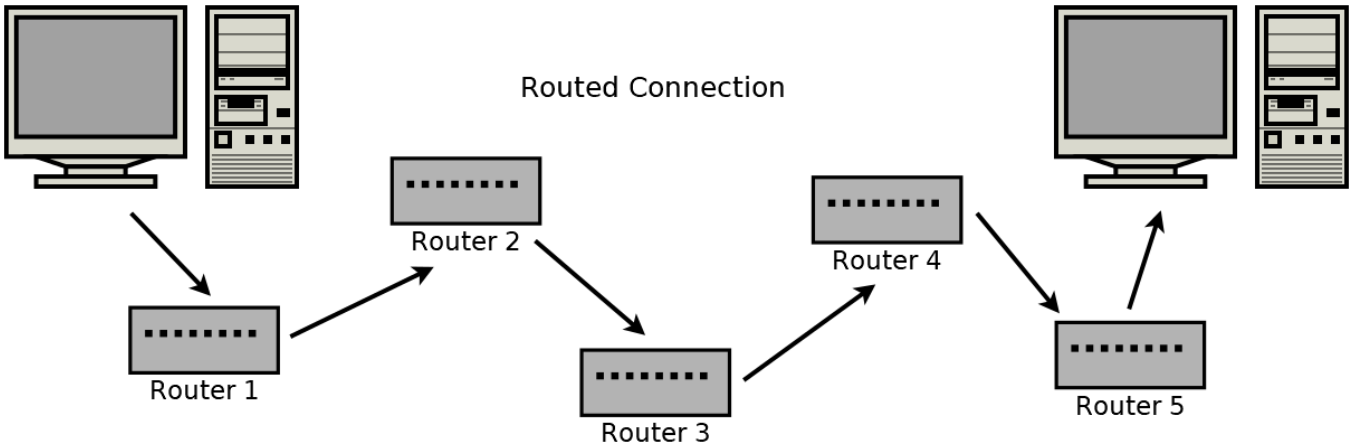
6. Domain names

To resolve a hostname:

- Your computer sends a UDP message to domain name server and asks it for an **AAAA-type record** for the domain you're trying to resolve.
- If this record exists, an IPv6 address is returned.
- You can then connect to a server at that address to load the website.
- If no AAAA record exists, then your computer queries the server again, but asks for an **A record**.
- If this record exists, you will receive an IPv4 address for the server.

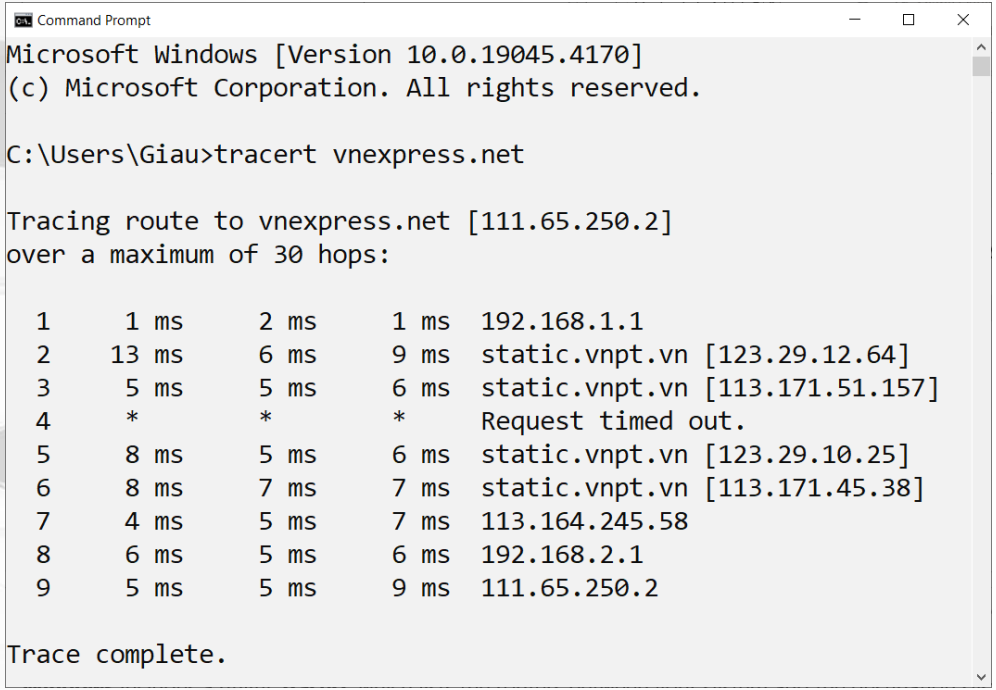
7. Internet routing

- When you make a connection over the internet, your **data first transmits to** your **local router**.
- From there, **it is transmitted to another router**, which is connected to another router, and so on.
- Eventually, your data reaches a router that is connected to the receiving device, at which point, the data has **reached its destination**.



7. Internet routing

Windows includes a utility, **tracert**, which lists the routers between your system and the destination system.

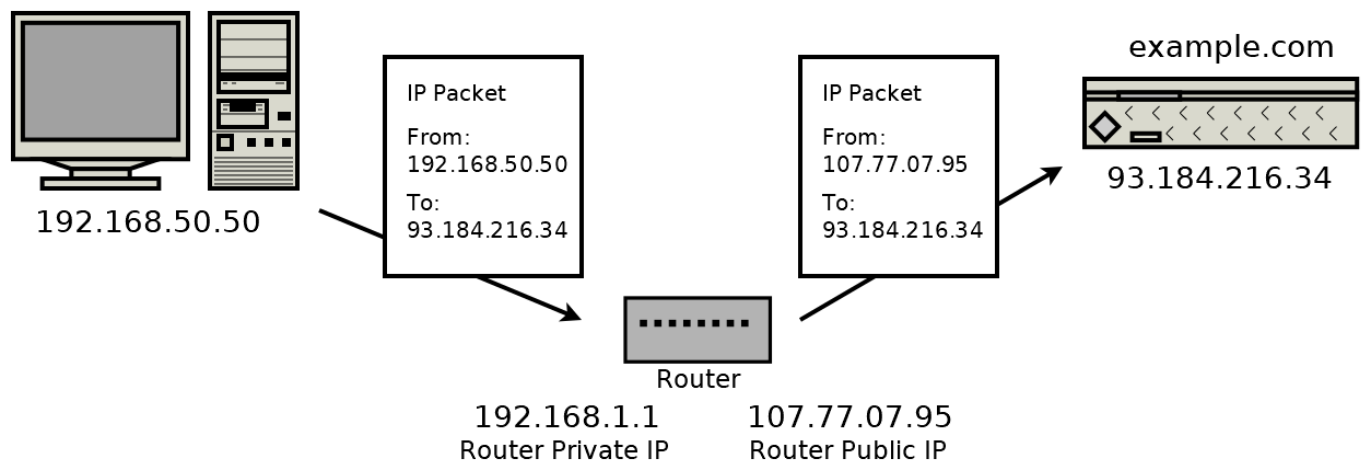


8. Local networks and address translation

- It’s common for households and organizations to have small **Local Area Networks** (LANs). There are IPv4 addresses ranges reserved for use in these small local networks.
  - 10.0.0.0 to 10.255.255.255
  - 172.16.0.0 to 172.31.255.255
  - 192.168.0.0 to 192.168.255.255
- The devices on the same LAN can directly address one another by their local address.

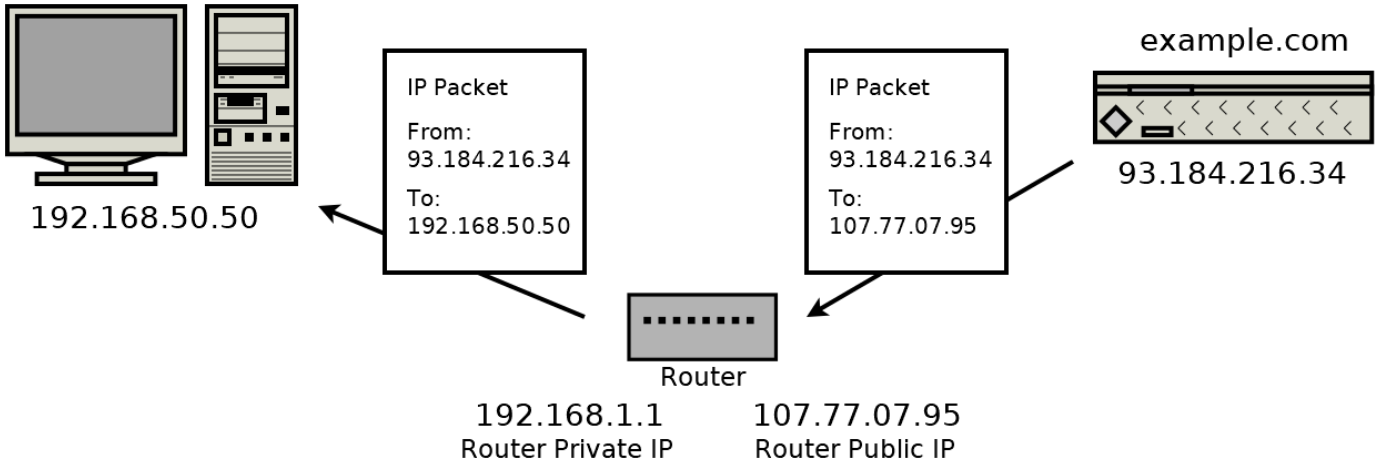
8. Local networks and address translation

- Any traffic communicated to the internet must undergo address translation (**Network Address Translation** (NAT)) by the router. The router does this by modifying the source IP address from the original private LAN IP address to its public internet IP address.



8. Local networks and address translation

- When the router receives the return communication, it must modify the destination address from its public IP to the private IP of the original sender.



9. Multicast, broadcast, and anycast

- When a packet is routed from one sender to one receiver, it uses **unicast** addressing.
- **Broadcast** addressing allows **a single sender** to address a packet to **all recipients** simultaneously.
- **Multicast** is a one-to-many communication. Multicast involves some group management.
- **Anycast** addressed packets are used to deliver a message to one recipient when you don’t care who that recipient is.

10. Port numbers

- An IP address gets a packet routed to a specific system, but a port number is used to **route the packet to a specific application** on that system.

- Port numbers are stored as unsigned 16-bit integers (0 – 65,535)

10. Port numbers

Some port numbers for common protocols

Port Number		Protocol
20, 21	TCP	File Transfer Protocol (FTP)
22	TCP	Secure Shell (SSH)
25	TCP	Simple Mail Transfer Protocol (SMTP)
53	UDP	Domain Name System (DNS)
80	TCP	Hypertext Transfer Protocol (HTTP)
110	TCP	Post Office Protocol, Version 3 (POP3)
143	TCP	Internet Message Access Protocol (IMAP)
443	TCP	HTTP over TLS/SSL (HTTPS)

11. Clients and servers

- The client-server model is a common paradigm in networking.
  - A server listens for connections
  - The client, knowing the address and port number that the server is listening on, establishes the connection by sending the first packet.

- For example, the web server at example.com listens on port 80 (HTTP) and port 443 (HTTPS).

12. Socket

- A **socket** is **one end-point of a communication link** between systems.

  It’s an abstraction in which your application can send and receive data over the network.
- An open socket is uniquely defined by a 5-tuple consisting of:
  - Local IP address
  - Local port
  - Remote IP address
  - Remote port
  - Protocol (UDP or TCP)

13. What’s your address?

- Local IP address:
  - Windows: ipconfig
  - Unix-based systems: ifconfig or ip addr
- Public IP address:
  - http://api.ipify.org/
  - http://helloacm.com/api/what-is-my-ip-address/
  - http://icanhazip.com/
  - http://ifconfig.me/ip


Q&A