

## Scope and prerequisites

This handout teaches Theme A2, Networks, for the course first assessed in 2027: network types and devices, protocols and layers, addressing, data transmission and network security. All of it is taught here as SL and HL content.

Prerequisites: bits and bytes from A1 (1 byte = 8 bits).

## Network fundamentals

### ■ What happens when you open a web page?

You type a web address on your phone at school. In under a second your request crosses the school Wi-Fi, a router, your internet provider and several countries, reaches a server, and the page comes back in hundreds of small pieces that are put together on your screen. Each step follows agreed rules. This handout explains them.

### ■ Types of network

A **network** is two or more devices joined to share data and resources.

| Type                                    | Covers                                          | Example                                 |
|-----------------------------------------|-------------------------------------------------|-----------------------------------------|
| <b>LAN</b><br>(local area network)      | one site                                        | a school or office                      |
| <b>WAN</b><br>(wide area network)       | a large area, joining LANs                      | the internet; a bank’s branches         |
| PAN (personal area network)             | one person’s devices                            | a phone and earbuds over Bluetooth      |
| <b>VPN</b><br>(virtual private network) | a private, encrypted link over a public network | working from home on the school network |

### ■ Network devices

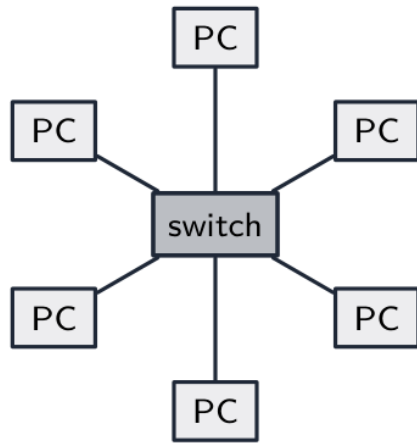
| Device                       | Job                                                                                                                                       |
|------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| network interface card (NIC) | connects a device to the network; has a unique <b>MAC address</b>                                                                         |
| <b>switch</b>                | joins devices in a LAN and forwards known unicast frames towards the destination port; broadcasts and unknown destinations may be flooded |
| <b>router</b>                | joins networks and forwards packets between them, choosing a route                                                                        |
| wireless access point (WAP)  | lets wireless devices join a wired LAN                                                                                                    |
| modem                        | converts digital data to a signal for the provider’s line, and back                                                                       |
| <b>firewall</b>              | checks traffic against rules and blocks what is not allowed                                                                               |
| server                       | provides a service to other devices: files, web pages, e-mail                                                                             |

### ■ Client–server and peer-to-peer

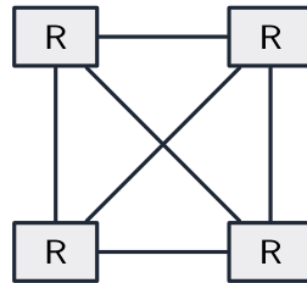
|          | Client–server                                    | - | Peer-to-peer                             |
|----------|--------------------------------------------------|---|------------------------------------------|
| roles    | servers provide services; clients request them   |   | every device is equal and can share      |
| control  | central: security, backups, updates in one place |   | none: each user manages their own device |
| weakness | the server is a single point of failure; costly  |   | hard to secure and back up               |
| example  | school log-ins, a web server                     |   | file sharing between two laptops         |

### ■ Topologies

A **topology** is the layout of the connections.



star



full mesh

*Star: everything through the centre. Mesh: many alternative paths*

- **Star:** easy to add devices; one broken cable affects one device; but if the central switch fails, the whole network fails.
- **Mesh:** many paths, so data can route round a failure; the internet's core is a partial mesh. Needs more cables and is costly.

Data travels over **wired** links (copper or fibre-optic cable: fast, reliable, secure) or **wireless** links (Wi-Fi, mobile: convenient, but slower, more interference and easier to intercept).

## Protocols and layers

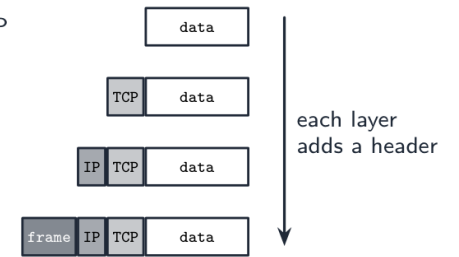
### ■ Why protocols?

A **protocol** is an agreed set of rules for communication: the format of the data, how errors are handled, and the order of messages. Devices from different makers can talk only because they follow the same protocols.

### ■ The TCP/IP model

Networking is split into **layers** ; each layer does one job and uses the layer below.

|                    |                        |
|--------------------|------------------------|
| <b>application</b> | HTTP, HTTPS, DNS, SMTP |
| <b>transport</b>   | TCP, UDP               |
| <b>internet</b>    | IP                     |
| <b>link</b>        | Ethernet, Wi-Fi        |



*Encapsulation: each layer wraps the data from the layer above in its own header*

| Layer       | Job                                                  | Protocols              |
|-------------|------------------------------------------------------|------------------------|
| application | the service the user sees                            | HTTP, HTTPS, DNS, SMTP |
| transport   | splits data into segments; checks delivery and order | TCP, UDP               |
| internet    | addresses packets and routes them between networks   | IP                     |
| link        | sends frames over the physical connection            | Ethernet, Wi-Fi        |

Layers let each part change on its own: a web browser works the same over Wi-Fi or cable, because only the link layer differs. Going down, protocols wrap data with control information such as headers and sometimes trailers (**encapsulation** ); the receiver removes the relevant information going up.

| Protocol     | Use                                                                                                                           |
|--------------|-------------------------------------------------------------------------------------------------------------------------------|
| HTTP / HTTPS | web pages; HTTPS is encrypted                                                                                                 |
| TCP          | reliable delivery: numbered segments, acknowledgements, resends                                                               |
| UDP          | datagrams without built-in delivery or ordering guarantees; checksums can detect corruption: video calls, games, live streams |
| IP           | addresses and routes each packet                                                                                              |
| DHCP         | gives a device an IP address when it joins                                                                                    |
| DNS          | turns a domain name into an IP address                                                                                        |
| SMTP / IMAP  | sending and fetching e-mail                                                                                                   |

TCP suits a web page or a file, where every byte must arrive. UDP suits a video call: a late packet is useless, so it is better to drop it than wait.

## Addressing

### ■ IP and MAC addresses

|            | IP address                            | MAC address                                             |
|------------|---------------------------------------|---------------------------------------------------------|
| identifies | a device's position on a network      | the network card itself                                 |
| set by     | the network (often DHCP)              | often the maker; software may configure or randomise it |
| changes?   | may change with network configuration | may be changed or randomised                            |
| example    | 192.168.1.24                          | 3C:22:FB:9A:10:07                                       |

- **IPv4** uses 32 bits, about 4.3 billion addresses: too few for every device. **IPv6** uses 128 bits, written in hex such as 2001:db8::1.
- A **private** address (such as 192.168.x.x) works only inside a LAN; the router shares one **public** address with the whole LAN.
- A **static** address never changes (servers need one so they can be found); a **dynamic** address is lent by DHCP.

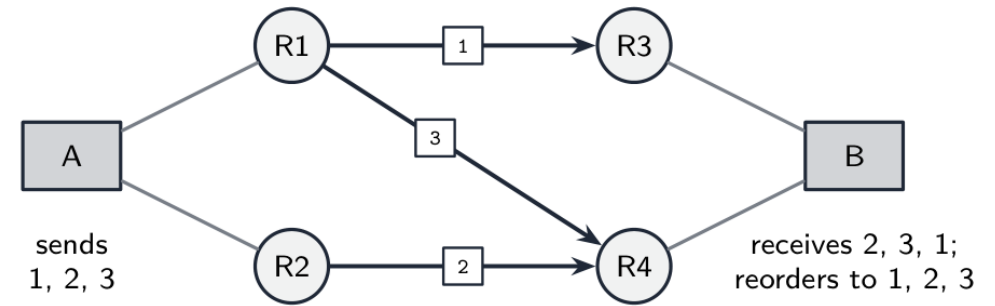
### ■ DNS: finding a server

1. You type `www.example.org`.
2. Your device asks a DNS server for its IP address.
3. If that server does not know, it asks other DNS servers in turn.
4. The IP address comes back, and your browser connects to it. The answer is stored (cached) to save time next visit.

## Data transmission

### ■ Packet switching

The internet uses **packet switching** : a message is split into **packets**, each sent separately.



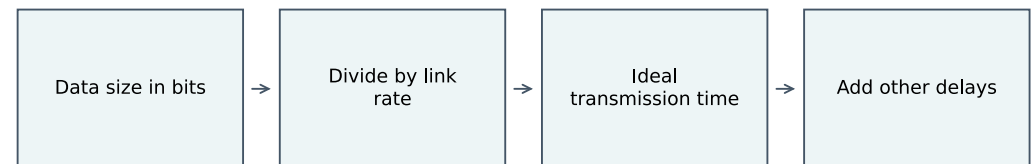
*Packets can take different routes and arrive out of order*

1. The data is split into packets. Each has a header with the source and destination IP addresses and a sequence number.
2. Each router reads the destination and forwards the packet along the best route it knows at that moment.
3. Packets may arrive out of order; TCP uses the sequence numbers to put them back, and asks again for any that are missing.

Advantages: lines are shared by many users, and packets route round a failed or busy link.

### ■ Speed: bandwidth, latency, throughput

- **Bandwidth** : the most data a link can carry per second, in bits per second.
- **Latency** : the delay before data starts to arrive, in milliseconds.
- **Throughput** : the data actually delivered per second; always at or below the bandwidth, because of other users, errors and resends.



*The ideal transmission time; real transfers take longer*

$$\text{time} = \frac{\text{data size in bits}}{\text{rate in bits per second}}$$

$$\text{time} = \frac{12 \times 10^6 \times 8 \text{ bits}}{8 \times 10^6 \text{ bits/s}} = 12 \text{ s}$$

for a 12 MB file on an 8 Mbit/s link, before latency and overhead are added. Note the units: file sizes are usually in **bytes**, link speeds in **bits** per second.

## Network security

### Threats

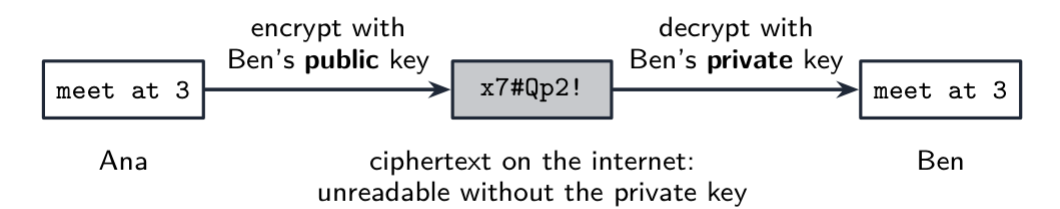
| Threat              | What happens                                                           |
|---------------------|------------------------------------------------------------------------|
| malware             | harmful software, such as a virus or ransomware, damages or locks data |
| phishing            | a fake message tricks a user into giving away a password               |
| interception        | data is read as it travels, for example on open Wi-Fi                  |
| unauthorised access | an attacker uses a weak or stolen password                             |
| denial of service   | a server is flooded with requests until real users cannot reach it     |

### Countermeasures

- Firewalls** check each packet against rules: allowed addresses, ports and protocols.
- Authentication** checks who a user is: something they know (password), have (phone) or are (fingerprint). **Multi-factor authentication** uses two or more. **Authorization** then decides what that user may do.
- Software updates close known weaknesses; backups allow recovery from ransomware; user training reduces phishing.
- Encryption** makes intercepted data unreadable without the key.

### Encryption

|         | Symmetric encryption          | Asymmetric encryption                                              |
|---------|-------------------------------|--------------------------------------------------------------------|
| keys    | one shared secret key         | a public key and a private key                                     |
| speed   | fast                          | slower                                                             |
| problem | the key must be shared safely | the public key must be authenticated and the private key protected |



*Anyone can lock with the public key; only the private key unlocks*

HTTPS uses both: asymmetric encryption to agree a secret key safely, then fast symmetric encryption for the rest of the session. A **digital certificate**, signed by a trusted authority, binds a domain or identity to a public key; the browser checks the trusted signature chain, domain and validity. This authenticates the connection but does not guarantee that a site is honest.

## Practice

- State the difference between a LAN and a WAN, with an example of each.
- State the job of (a) a switch; (b) a router; (c) a DHCP server.
- Give **one** advantage and **one** disadvantage of a star topology.
- Explain why a school uses a client–server network rather than peer-to-peer.
- Explain why networking is organised in layers.
- State whether TCP or UDP suits (a) downloading an app; (b) an online multiplayer game. Explain (b).
- Describe what DNS does when you visit a website.
- Explain why IPv6 was introduced.
- Describe how a file is sent across the internet by packet switching.
- A 30 MB video is sent over a 20 Mbit/s link. Calculate the ideal transmission time, and state why the real time is longer.
- Explain the difference between symmetric and asymmetric encryption.

12. A user receives an e-mail asking them to “confirm your password” at a link. Name the threat and give **two** ways to reduce it.

### Answers.

1. A LAN covers one site, such as a school; a WAN covers a large area and joins LANs, such as the internet.
2. (a) joins devices in a LAN and sends data only to the device it is for; (b) forwards packets between networks along a route; (c) gives each device an IP address when it joins.
3. Advantage: e.g. one cable failure affects only one device; easy to add devices. Disadvantage: if the central switch fails, the whole network fails.
4. Security, backups and software updates are managed centrally on the server; users can log in from any computer and find their files.
5. Each layer does one job with its own protocols, so one layer can change, such as cable to Wi-Fi, without changing the others; makers can build to one layer’s standard.
6. (a) TCP; (b) UDP: it is fast with no resends, and a late game update is useless, so it is better to drop it than wait.
7. The browser asks a DNS server for the IP address of the domain name; if needed, other DNS servers are asked; the IP address is returned and the browser connects to it.
8. IPv4’s 32 bits give only about 4.3 billion addresses, too few for every device; IPv6’s 128 bits give far more.
9. The file is split into packets with headers (source, destination, sequence number); routers forward each packet by the best route; packets may arrive out of order and are reassembled by sequence number; missing ones are resent.
10.  $30 \times 10^6 \times 8 = 240 \times 10^6$  bits;  $240 \times 10^6 \div 20 \times 10^6 = 12$  s. Latency, other traffic, headers and resends lower the throughput.
11. Symmetric uses one shared key to encrypt and decrypt, which must be passed on safely; asymmetric encrypts with a public key and decrypts with a different, private key.
12. Phishing. E.g. train users to check links and senders; spam filters; multi-factor authentication so a stolen password alone is not enough.

### Exam tips

---

- Name each device by its **job**, not just its name: “a router forwards packets between networks”.
- In TCP/UDP answers, link the choice to the scenario: reliability for files, speed for live data.
- Describe packet switching in order: split, header, route, reassemble, resend missing.

- Keep bandwidth (the maximum), throughput (what you actually get) and latency (the delay) apart.
- Convert bytes to bits ( $\times 8$ ) before dividing by a speed in bits per second.
- For encryption, state which key encrypts and which decrypts.