

Communication and internet technologies

A-Level Computer Science 9618 · Topic 14 revision notes

Everything from the 3 lessons of this topic on a few pages: the key idea of each lesson, the diagram to remember, the points to learn and the traps examiners set. Read it, cover it, test yourself.

1. The TCP/IP protocol suite

2. HTTP, FTP, POP3, IMAP, SMTP and BitTorrent

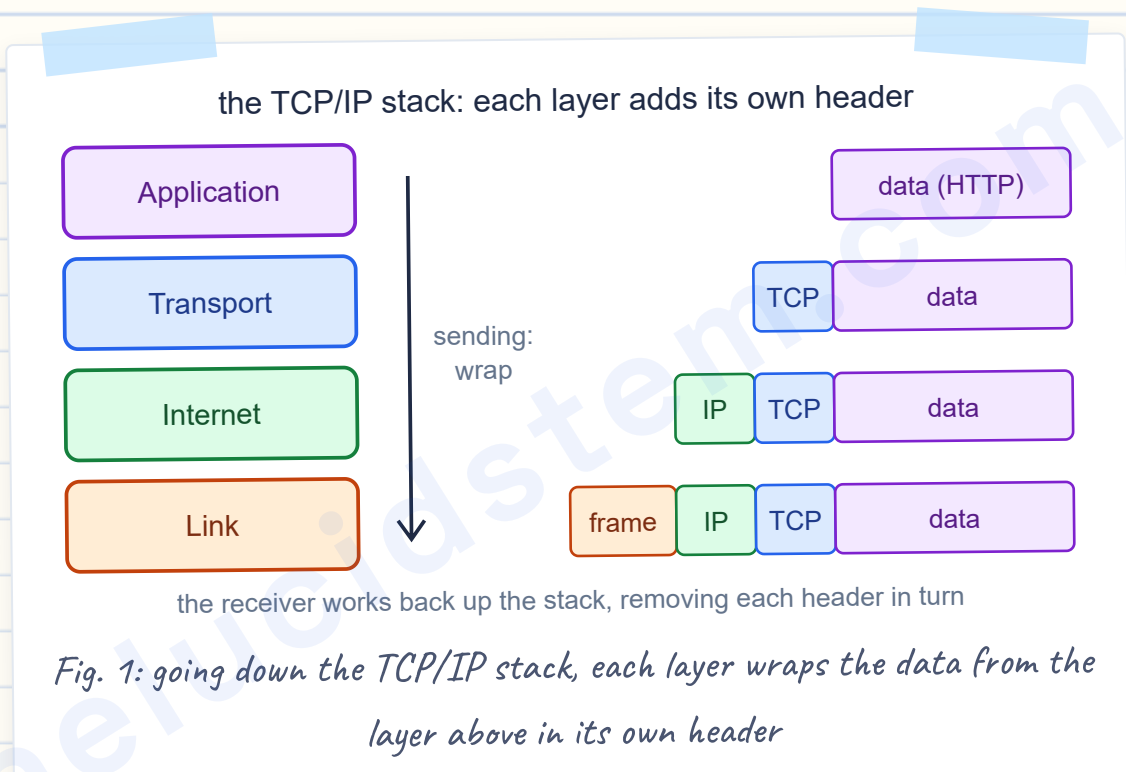
3. Circuit switching and packet switching.

1 The TCP/IP protocol suite

Syllabus 14.1

A **protocol** is a set of rules that governs communication between devices: the format of the data, how it is addressed, how errors are detected and what happens when they are. Without agreed protocols, devices from different manufacturers could not understand each other.

TCP/IP is a **protocol suite** arranged as a **stack** of four layers: **Application**, **Transport**, **Internet** and **Link**. Each layer has its own job, uses the services of the layer below and provides a service to the layer above. On the way down the stack each layer adds a **header** (encapsulation); on the way up the receiver's layers remove them.



Worked example: How many segments?

A browser sends 4500 bytes of data. The Transport layer can put at most 1460 bytes of data in each segment. How many segments are sent, and how many bytes does the last one carry?

1. Divide: $4500 / 1460 = 3.08\dots$, so 3 full segments are not enough.
2. 3 full segments carry $3 \times 1460 = 4380$ bytes.
3. Remaining: $4500 - 4380 = 120$ bytes go in a 4th segment.
4. Each segment gets a sequence number so the receiver can rebuild the data in the right order.

Answer: 4 segments; the last one carries 120 bytes

One job per layer.

- ✓ **Application:** the program's protocol (HTTP, FTP, SMTP, POP3, IMAP, BitTorrent).
- ✓ **Transport:** segments, port numbers, sequence numbers, acknowledgements (TCP).
- ✓ **Internet:** IP addresses and routing of packets between networks (IP).
- ✓ **Link:** frames, MAC addresses and signals on one physical link (Ethernet, WiFi).

Watch out!

"A protocol lets computers communicate" is too vague for the mark. Say it is a *set of rules* and say *why rules are needed*: so that devices from *different manufacturers or platforms* interpret the data in the same way.

Do not mix up the addresses. Port numbers belong to the **Transport** layer, IP addresses to the **Internet** layer and MAC addresses to the **Link** layer.

Sequence numbers are added by the **Transport** layer, not by IP, and it is TCP at the receiver that reorders the data and requests lost segments.

2 HTTP, FTP, POP3, IMAP, SMTP and BitTorrent

Syllabus 14.1

HTTP transfers web pages and their resources between a web server and a browser (**HTTPS** is HTTP made secure by encryption). **FTP** transfers files between a client and a server. Email uses three protocols: **SMTP** pushes an email from the sender's client to their mail server and on to the recipient's mail server; **POP3** and **IMAP** let the recipient pull it from their server. POP3 downloads the email (usually deleting the server copy); IMAP keeps it on the server so every device stays in sync. **BitTorrent** is a **peer-to-peer** protocol: a file is split into **pieces** and each peer in a **swarm** downloads different pieces from many other peers at once while uploading the pieces it already has.

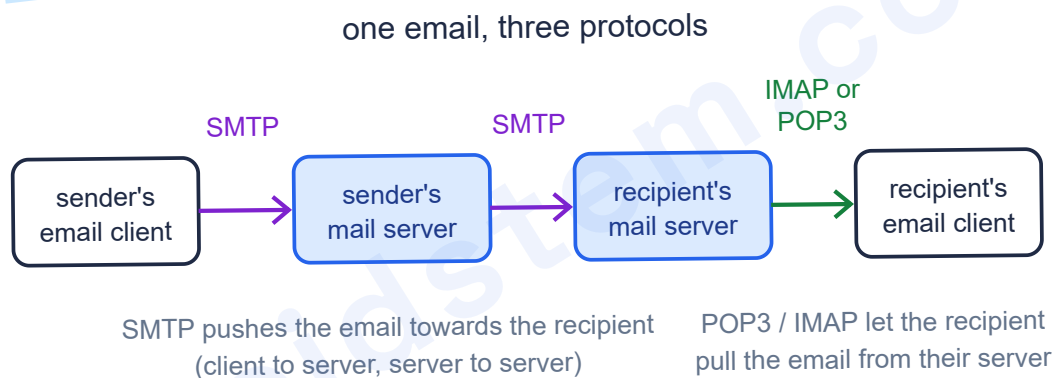


Fig. 1: SMTP pushes an email to the recipient's mail server; POP3 or IMAP pulls it to the recipient's device

Worked example: Choose the retrieval protocol

A sales manager reads email on a phone, a tablet and an office PC, and wants an email read or moved to a folder on one device to show as read or moved on all of them. POP3 or IMAP?

1. With POP3 the email is downloaded to one device and usually deleted from the server, so the other devices never see it.
2. With IMAP the email stays on the server; each device views the server copy.
3. Flags (read/unread) and folders are stored on the server, so every device stays in sync.

Answer: IMAP, because the email and its folders stay on the server and are synchronised across all devices

Push, pull or share.

- ✓ **HTTP/HTTPS:** request and transfer web resources (HTTPS encrypted).
- ✓ **FTP:** transfer files between client and server.
- ✓ **SMTP push; POP3 / IMAP pull:** POP3 downloads and deletes, IMAP keeps on the server and syncs.
- ✓ **BitTorrent:** torrent file, tracker, swarm, seeds and leechers, pieces from many peers at once.

Watch out!

HTTP is not "the internet" and not "how web pages are displayed". It is the set of rules for requesting and transferring web resources between a client and a web server; the browser renders the page.

POP3 and IMAP are not used to send email. A common lost mark is writing "IMAP sends the email to the recipient": the recipient's client retrieves it.

The tracker does *not* store or send the file. It only coordinates the swarm by telling peers about each other; the pieces travel directly between peers.

3 Circuit switching and packet switching

Syllabus 14.2

In **circuit switching** a **dedicated path** (circuit) between sender and receiver is set up before any data is sent; all the data follows that path, in order, and the bandwidth is reserved until the circuit is released. In **packet switching** the message is split into **packets**, each with a header holding the source and destination **IP addresses** and a **sequence number**. Each packet is routed **independently**; every **router** reads the destination address and uses its **routing table** to forward the packet to the next hop, so packets may take different routes and arrive out of order. The receiver **reassembles** them using the sequence numbers. Links are shared by everyone, so packet switching uses the network far more efficiently.

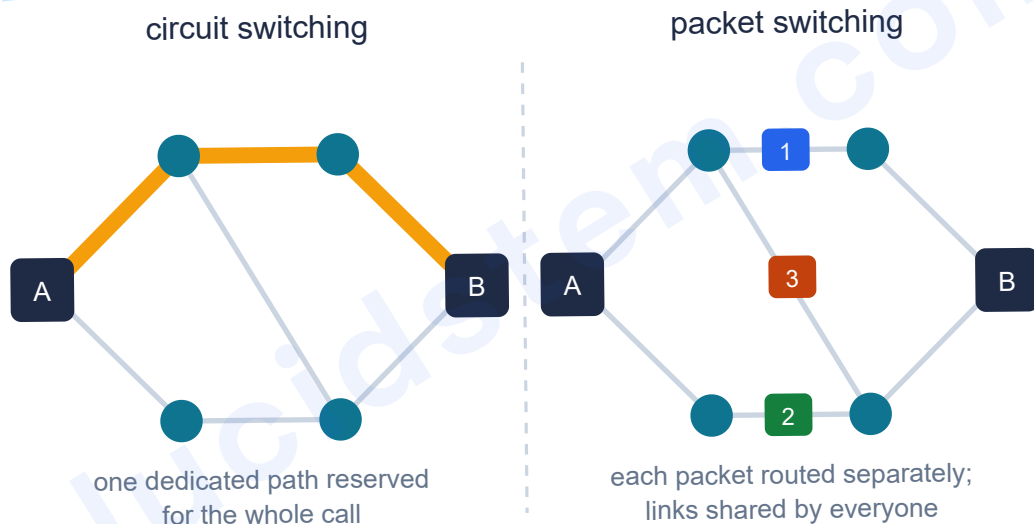


Fig. 1: circuit switching reserves one path (orange) for the whole call; in packet switching each numbered packet may take a different route

Worked example: Put the packets back together

A message is split into packets numbered 1 to 5. They arrive in the order 2, 5, 1, 4. What does the receiving computer do?

1. It reads the sequence number in each packet header and holds the packets in a buffer.
2. The header says there are 5 packets, but packet 3 has not arrived.
3. After waiting, the receiver requests packet 3 to be sent again.
4. When packet 3 arrives, the packets are reassembled in the order 1, 2, 3, 4, 5 to rebuild the message.

Answer: Buffer and reorder the packets by sequence number, request the missing packet 3, then reassemble 1 to 5

Path first, or packets first.

- ✓ **Circuit:** set up a dedicated path, send in order at a constant rate, release; bandwidth wasted when idle.
- ✓ **Packet:** split, header (IPs, sequence number, hop limit), route independently, reassemble.
- ✓ **Router:** read destination IP, look up routing table, forward to next hop.
- ✓ **Justify:** real-time and constant rate favours circuit; shared, bursty, resilient favours packet.

Watch out!

"Circuit switching is faster" is not enough. Say *why*: once set up, the bandwidth is **dedicated**, so the data arrives at a constant rate, in order, without queuing at routers.

A router does not read the whole message, and it does not reassemble the packets. It looks only at the **destination IP address** of each packet; reassembly happens at the **destination** computer.

Modern internet phone calls (VoIP) use **packet switching**, not circuit switching. They cope with variable delay by buffering, and quality drops when the network is congested.