

Communication and internet technologies

A2

Read online at pastopic.com/cambridge/computer-science-9618/notes/communication-and-internet-technologies

Last checked 30 September 2026

Read first: [Communication](#)

© 2026 Pastopic. Free for personal study. Please share the link, not the file.

What you need to know

This is the A Level topic (Paper 3). LANs, WANs, IP addresses and the router in a home network are the AS topic **Communication**. By the end of this topic you should be able to:

1. **Explain why computers need protocols** to communicate at all.
2. **Explain why protocols are organised as a stack** of layers, where each layer has its own job.
3. **Describe the TCP/IP protocol suite**: its four layers (Application, Transport, Internet, Link), the purpose and function of each, and how the stack is used when a message goes from one host to another across the internet.
4. **Describe the protocols HTTP, FTP, POP3, IMAP, SMTP and BitTorrent** and what each is for, including how BitTorrent gives peer-to-peer file sharing.
5. **Describe circuit switching**: how it works, its benefits and drawbacks, and where it is used.
6. **Describe packet switching**: how it works, its benefits and drawbacks, and where it is used.
7. **Describe the function of a router** in packet switching.
8. **Explain how packet switching carries a message across a network**, including the internet.

Understand it

Why protocols?

A **protocol** is a set of rules that both ends of a conversation agree to follow. For two computers to exchange data, they must agree on many things: how the data is laid out (which bits are the address, which are the message), how the receiver is identified, how errors are spotted and what happens next, how fast data is sent and in what order. If the sender and receiver follow different rules, the receiver cannot make sense of what arrives, so no communication takes place.

Because protocols are **standards**, they make communication **independent of the hardware and software** at each end. A phone from one manufacturer running one operating system can exchange data with a server from a different manufacturer running a different operating system, as long as both follow the same protocols.

A protocol is only rules. It does not "do" anything itself; the hardware and software that follow it do the work.

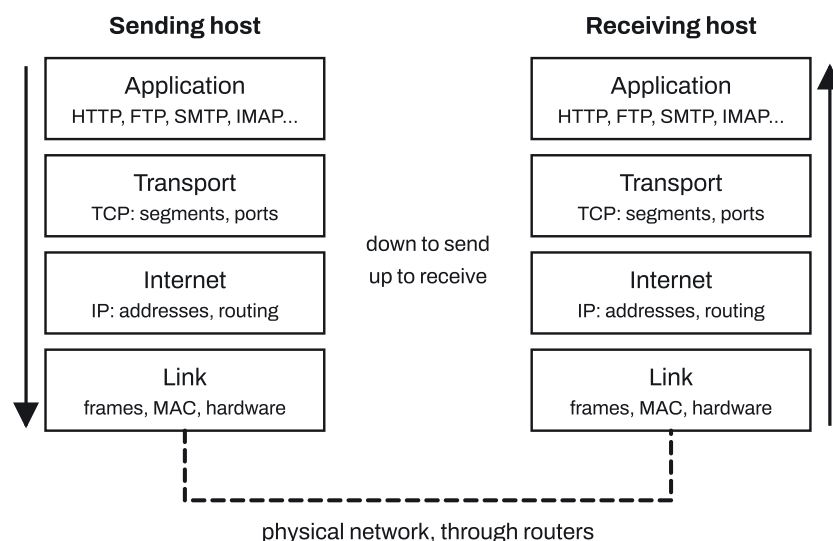
Why a stack of layers?

Getting a message from an app on one computer to an app on another involves many separate jobs: talking to the user's program, splitting the data up, addressing it, finding a route, and putting signals on a cable or radio link. Instead of one huge protocol, the jobs are split into **layers**, one protocol (or a few) per layer, stacked on top of each other.

- Each layer does **its own job** and relies on the layer below to do the rest.
- A layer only **passes data to and receives data from the layers directly above and below it**, through a fixed interface.
- Because the interfaces are fixed, one layer's protocol can be changed or replaced (for example Wi-Fi instead of a wired Ethernet link) without changing the others.
- The layers are carried out by **software** installed on each device (with the lowest layer driving the network hardware).

The TCP/IP suite

The internet uses the **TCP/IP** suite, named after its two best-known protocols, TCP (Transmission Control Protocol) and IP (Internet Protocol). It has four layers, from the top:



Layer	Its job in one line	Example protocols
Application	Gives programs and users access to the network	HTTP, HTTPS, FTP, SMTP, POP3, IMAP, BitTorrent

Layer	Its job in one line	Example protocols
Transport	End-to-end delivery between the two programs	TCP
Internet	Addressing with IP addresses and routing across networks	IP
Link	Moving the data over one physical link to the next device	Ethernet, Wi-Fi

Application layer. This is where the user's programs (a web browser, an email client, a file-transfer program) meet the network. It holds the protocols those programs use to exchange data, so it decides what the data means and how requests and replies are formed (for example "send me this web page"). It gives programs access to the services of the lower layers, and it can provide services such as encryption and authentication (HTTPS is HTTP made secure). It passes the data down to the Transport layer.

Transport layer. This layer makes sure the data gets from the program on one host to the right program on the other host, complete and in order.

- When sending, it **splits the data into segments** (small pieces that will travel as packets) and adds a header to each with a **sequence number** and **port numbers** that identify the programs at each end.
- It sets up an **end-to-end connection** between the two hosts and controls the flow of data, so the sender does not overwhelm the receiver.
- When receiving, it puts the pieces **back in the right order** using the sequence numbers, checks for errors, **acknowledges** pieces that arrive correctly, and has lost or damaged ones **sent again**. Then it hands the data to the right program.

Internet layer. This layer gets each packet across the networks that make up the internet.

- It adds a header with the **source and destination IP addresses** to each packet (the packet is now called a **datagram**). The IP address identifies the network and the host on it.
- It **routes** each packet independently, towards the best route available. Routers work at this layer.
- An IP address together with a port number is called a **socket**: it identifies one program on one host.

Link layer. This layer deals with the physical network between two directly connected devices, one hop at a time.

- It puts each datagram into a **frame** ready for the network hardware and sends it as signals over the cable or radio link.
- It works with **MAC addresses** (the physical addresses of network cards), so it maps the next device's IP address to its MAC address.
- It follows the rules of the local network (such as Ethernet or Wi-Fi) and checks frames for errors.

A message through the stack

Sending an email shows how the stack is used when a message goes from one host to another on the internet.

1. The user works with the email program, which uses an Application-layer protocol (SMTP) to prepare the message.
2. The message passes **down** the stack. Each layer adds its own header (**encapsulation**): the Transport layer splits it and numbers the pieces, the Internet layer adds IP addresses, and the Link layer frames it.
3. The Link layer puts the frames on the network. Routers along the way read the IP header and pass each packet on, link by link.
4. At the receiving host the data passes **up** the stack. Each layer reads and removes its own header: the Link layer takes out the datagram, the Internet layer checks the address, the Transport layer reorders and checks the pieces, and the Application layer gives the whole message to the mail program.

The user only ever deals with the top layer; only the bottom layer deals with the hardware.

Application-layer protocols

Protocol	What it is for
HTTP (HyperText Transfer Protocol)	Transferring web pages (hypertext) and the files in them between a web server and a browser. HTTPS is the secure version: the data is encrypted.
FTP (File Transfer Protocol)	Uploading, downloading and moving files between computers on a network, such as from a client to a server.
SMTP (Simple Mail Transfer Protocol)	Sending email: from the sender's device to their mail server, and between mail servers. It is a push protocol.
POP3 (Post Office Protocol 3)	Receiving email: it downloads messages from the mail server to one device, and usually deletes them from the server. A pull protocol.
IMAP (Internet Message Access Protocol)	Receiving email: messages stay on the mail server , so the user can read them from any device and every device shows the same, synchronised mailbox. A pull protocol.
BitTorrent	Peer-to-peer file sharing : many users share a file directly with each other, with no central server holding it.

Email uses two protocols: SMTP pushes the message out to the recipient's mail server, where it waits; the recipient's program then pulls it down with POP3 or IMAP.

BitTorrent

In **peer-to-peer** sharing, each user's computer (a **peer**) is both a client and a server: it downloads from other peers and uploads to them. BitTorrent lets thousands of peers across the internet share one file this way.

1. A peer who has the whole file creates a small **torrent** file (a descriptor). It holds **metadata** about the file to be shared, such as its name, size and how it is split up, and the address of a tracker.
2. The file is broken into **equal-sized pieces**. At the start the whole file must be on at least one peer.

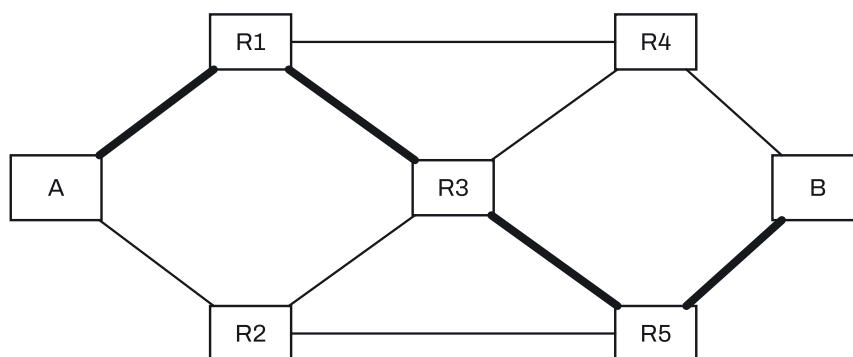
3. A peer who wants the file gets the torrent file and connects to the **tracker**, a server that keeps a list of all the peers sharing this file (the **swarm**). The tracker holds information about the peers, not the file itself.
4. The peer downloads **different pieces from many peers at the same time**.
5. As soon as a peer has a piece, it becomes a **source** of that piece and uploads it to others.
6. When a peer has the whole file and keeps sharing it, it is a **seed**.

Because every downloader is also an uploader, a popular file spreads faster, and no single server has to carry all the traffic.

Circuit switching

In **circuit switching** a **dedicated path** (a circuit) is set up between the sender and the receiver **before** any data is sent, like an old-fashioned telephone call.

1. The circuit is established through the network's switches from one end to the other.
2. All the data then flows along that one path as a **continuous stream**, using **the whole bandwidth** of the circuit. It arrives in the order it was sent. The circuit can carry data both ways.
3. At the end of the transmission the circuit is released.



heavy line: the dedicated circuit, set up first, held until the end

Benefits

- Once the circuit is up, **the whole bandwidth is available**, so data flows at a **steady, fast rate**.
- Data **arrives in the order it was sent**, so there is no reordering at the receiver.
- **No delays at switches** once the circuit is set up, so it suits **real-time** transmission.
- Data is unlikely to be lost, because it all follows the same dedicated path; it suits long, continuous transmissions.

Drawbacks

- **Setting up** the circuit takes time before any data can be sent.
- **No one else can use the circuit** while it is held, even when no data is flowing, so bandwidth is wasted.

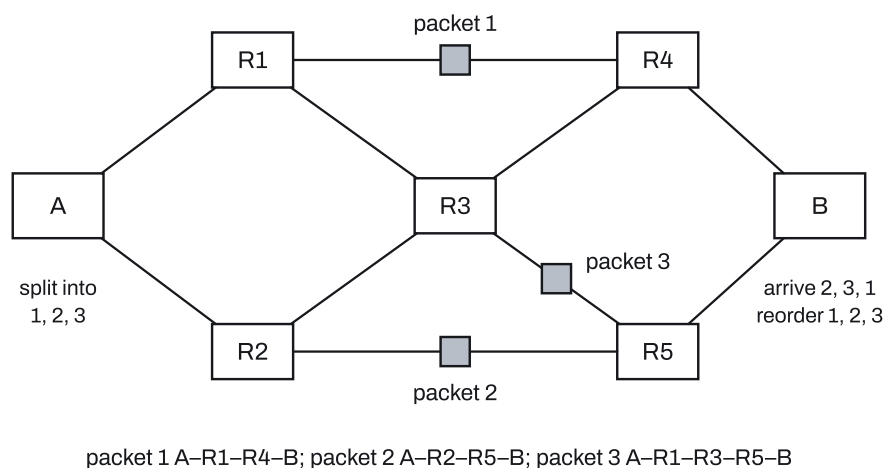
- There is **no alternative route**: if a link on the path fails, the transmission ends and a new circuit must be set up.
- All the data follows one path, so it is easier to intercept.
- A dedicated circuit for each connection is expensive, and the network is hard to scale.

Where it is used: traditional voice telephone calls, live video calls, and dedicated private lines, where a sustained, steady, real-time connection matters more than sharing the line.

Packet switching

In **packet switching** there is no dedicated path. The message is cut into **packets** that travel independently and share the network's links with everyone else's packets.

1. The message is split into **packets**, usually of equal size.
2. Each packet gets a **header** containing the **source and destination IP addresses**, the packet's **sequence number**, the total number of packets, a **hop count** and error-checking data, followed by its share of the data.
3. Each packet is sent **independently**. At every router it passes, the router decides the next hop, so packets from one message can take **different routes** (whichever is best at that moment, avoiding congested or failed links).
4. Packets can therefore **arrive out of order**. The receiver puts them back in order using the sequence numbers and **reassembles** the message.
5. The receiver checks each packet for errors and for gaps in the sequence numbers. It sends a **request to resend** any packet that is missing or damaged.
6. The hop count is reduced by 1 at each router. If it reaches 0, the packet is thrown away, so lost packets cannot circle the network forever; the missing packet is then asked for again.



Benefits

- **No line is tied up**: many users' packets share the same links, so bandwidth is used efficiently.

- Packets can be **re-routed** around failed, faulty or busy links, so data is more likely to get through.
- Only a **lost or damaged packet is resent**, not the whole message.
- Packets take different routes, so it is harder to intercept the whole message.
- A high rate of data transmission is possible, and users are only charged for the data they actually send.

Drawbacks

- **Delays**: packets must be reordered and reassembled at the destination, lost packets must be resent, and links are shared with other traffic. The delay varies, which is poor for real-time use.
- Packets can be **lost** on the way.
- It needs **complex protocols** and routing algorithms, and the receiver needs plenty of memory to hold packets while it reassembles large messages.
- Every packet carries a header, which is extra data to send.

Where it is used: the internet and other data networks: web pages, email, text messages, sending files and documents, and voice over IP (VoIP); anywhere large volumes of data are sent that don't need a live dedicated line.

The router in packet switching

A **router** receives packets and passes each one on towards its destination.

1. It **reads the destination IP address** in the packet's header.
2. It looks this up in its **routing table**, which holds information such as the networks it can reach, the neighbouring routers (the possible next hops) and their state, for example how busy they are.
3. It chooses the **next hop** on the best route available and sends the packet on. If a link is congested or down, it sends the packet another way.
4. It **reduces the hop count by 1**, and discards the packet if the count reaches 0.

Routers share information with each other to keep their routing tables up to date.

Circuit switching and packet switching side by side

Circuit switching	Packet switching
A dedicated path is set up before any data is sent	Data is sent straight away; no dedicated path
The message is sent as one continuous stream	The message is split into packets
All data follows the same route	Packets can take different routes
Data arrives in the order it was sent	Packets can arrive out of order and are reordered
Uses the whole bandwidth of the circuit	Shares the bandwidth with other users

Circuit switching

Each unit follows the path fixed at set-up

A failure on the path ends the transmission

Packet switching

Each packet only knows its destination; routers decide the path

Packets are re-routed; only lost packets are resent

Key facts and formulas

There is no formula list in the 9618 exams, so every fact here is **Learn it**.

Formula	Status
Protocol = an agreed set of rules for communication, so different hardware and software can exchange data	Learn it
TCP/IP layers, top to bottom: Application, Transport, Internet, Link	Learn it
Each layer only passes data to the layer directly above or below it; sending goes down the stack, receiving goes up	Learn it
Transport: segments, sequence numbers, ports, end-to-end delivery, flow control, resending	Learn it
Internet: IP addresses in the header (datagrams), routing	Learn it
Link: frames, MAC addresses, the physical link to the next device	Learn it
HTTP(S): web pages · FTP: files · SMTP: send email (push) · POP3 and IMAP: receive email (pull) · BitTorrent: peer-to-peer file sharing	Learn it
POP3 downloads to one device; IMAP keeps mail on the server and synchronises every device	Learn it
BitTorrent: torrent file (metadata) → tracker → swarm of peers → pieces → seeds	Learn it
Circuit switching: dedicated path set up first, whole bandwidth, same route, in order	Learn it
Packet switching: packets with headers, independent routes, reordered by sequence number, lost packets resent	Learn it
Hop count: reduced by 1 at each router; the packet is discarded at 0	Learn it

How to do it

In the Paper 3 questions we have tagged for this topic (35 questions, 2021–2025, counting a question each time it was set, as several were set on two or three papers), the types came up this often. 19 of the 35 questions mix types, so the counts add up to more than 35.

Question type	Questions
Application-layer protocols	12
Purpose of a TCP/IP layer	11
How packet switching passes a message	7
The protocol stack: layer order and interaction	7
Circuit switching: description, benefits and drawbacks	7
Packet switching: benefits, drawbacks and uses	7
Why protocols are essential	3
The router in packet switching	3
Differences between circuit and packet switching	2
How BitTorrent shares files	2

Every question is written: **one mark per correct point**, usually worth 2 to 10 marks. The facts are in "Understand it"; below is how to use them.

1. Application-layer protocols (12 questions)

1. **Name** the protocol exactly: HTTP (or HTTPS), FTP, SMTP, POP3, IMAP or BitTorrent. Most questions give one mark for the name.
2. **Describe** it with its **purpose**, precise enough that it fits no other protocol: "transfers web pages between a web server and a browser", "sends email from a device to a mail server and between mail servers".
3. For two or three protocols, pick ones with clearly different jobs (for example HTTP, FTP and SMTP) so each description earns its own mark.

Variations you will meet:

- **"Two protocols used when sending or receiving email"**: SMTP for sending; POP3 or IMAP for receiving. If POP3 is already given in the question, use SMTP and IMAP.
- **"State the purpose of IMAP"**: the mail stays on the server, so it can be read on any device, and all devices stay in sync. IMAP is **not** for sending.
- **Matching protocols to uses**: pair each protocol with its one best use; there may be a spare use that matches none (for example mapping addresses, which is a Link-layer job).
- **"Outline one protocol of the Application layer"**: the name plus one clear sentence of purpose.

2. Purpose of a TCP/IP layer (11 questions)

1. Check which layer is asked for, and whether it says "purpose" or "function". Both want what the layer does and why.
2. Give **several separate points** for that layer, each specific, from its list in "Understand it". Typical mark schemes allow up to 3 per layer.
3. Use the technical words: segments, sequence numbers, ports, IP addresses in the header, datagrams, routing, frames, MAC addresses.

Variations you will meet:

- **Two layers in one question** (Internet and Link, Application and Transport, Transport and Internet): write under separate headings and don't let points drift into the wrong layer. Addresses and routing are Internet; frames and MAC addresses are Link; splitting and sequencing are Transport.
- **"Identify two other layers and describe each"**: after the Application layer is given, name any two of Transport, Internet and Link, then describe each.
- **"Explain the purpose and function of the Application layer"**: access to the network for programs such as browsers and email clients, the user interacts with it, it holds the protocols programs use to exchange data, and it passes data to and from the Transport layer.

3. How packet switching passes a message (7 questions)

Write the steps in order, one point each:

1. The message is split into (equal-sized) **packets**.
2. Each packet gets a **header** with the **source and destination IP addresses** and a **sequence number**.
3. Each packet is sent **independently**, and packets can take **different routes**.
4. **Routers** send each packet along the best route available, using routing tables; a packet can be re-routed if a link fails or is busy.
5. Packets may **arrive out of order**; they are **reassembled in the right order** at the destination using the sequence numbers.
6. Missing or damaged packets are **resent**.

Variations you will meet:

- **"Do not include checking for completeness and resending"**: stop at step 5; resending earns nothing in that question.
- **"Two ways packet switching ensures the complete message is received"** (2 marks each): give a way and then its consequence. Packets are checked on arrival, **so** a damaged or missing packet leads to a resend request. Packets are sent separately, **so** if a route is blocked the packet is sent another way. The hop count stops a lost packet circulating, **so** when it is discarded a resend is requested.

4. The protocol stack: layer order and interaction (7 questions)

1. **Order:** Application, Transport, Internet, Link, from top to bottom. A diagram question usually gives two layers and asks for the missing two in their places.
2. **"How do the layers interact?"** Give these points:
 - the suite is a **stack** of layers, each with its own function;
 - each layer only accepts data from the **layer directly above or below** it, through an interface between adjacent layers;
 - as data passes down, each layer **adds its header**;
 - the **user** interacts only with the top (Application) layer, through its protocols;
 - the bottom (Link) layer is the only one with **direct access to the hardware** and the network;
 - the layers are carried out by software installed on the device.
3. **"How is the stack applied when a message is sent from one host to another?"** The sender works at the Application layer; the message passes **down** from Application to Link; the Link layer sends it over the network; at the receiver it passes **up** from Link to Application. If the question says "don't describe the individual layers", describe only this movement.

Variation: a fill-in question on the stack: the answer words are *stack*, *layered* and *TCP/IP*. See [9618/32/O/N/23 Q4](#).

5. Circuit switching: description, benefits and drawbacks (7 questions)

1. **Describe:** a dedicated circuit is set up **before** transmission; it lasts for the whole transmission and is released at the end; data travels as a continuous stream along the same route; it can carry data both ways.
2. **Benefits and drawbacks:** pick them from the lists in "Understand it". Give **different** points: "steady rate" and "fast rate" may count as the same benefit.
3. **Where appropriate, with a reason:** voice calls or live video calls, **because** a dedicated path must be kept for the whole call, the whole bandwidth is needed, and the communication is real time.

Write only about circuit switching. Comparing it with packet switching earns nothing unless the question asks for differences.

6. Packet switching: benefits, drawbacks and uses (7 questions)

1. **Benefits:** bandwidth shared; re-routing around failures; only lost packets resent; harder to intercept; high data rates possible.
2. **Drawbacks:** delay from reordering, resending and sharing; packets can be lost; complex algorithms; memory needed to reassemble; header overhead.
3. **Uses:** the internet, email, text messages, sending documents and large files that don't need to be live, VoIP. For "where appropriate", add the reason (re-routing around faulty lines, not needing the whole bandwidth, high volumes of data).

State a real benefit or drawback, not just a feature: "packets take different routes" is a feature; "packets can be re-routed if a link fails, so they are more likely to arrive" is a benefit.

7. Why protocols are essential (3 questions)

Give two or three separate points: protocols are a **standard set of rules** for communication; they allow devices from **different manufacturers and platforms** to communicate, independent of their hardware and software; without the same protocol, two devices could not understand each other's messages. Add what the rules cover (format, addressing, error checking) to expand a point.

8. The router in packet switching (3 questions)

In order: the router **reads the destination IP address** from the header; looks it up in its **routing table** (neighbouring routers, next hops, their state); decides the **next hop** on the best route and sends the packet on; **reduces the hop count** by 1. Avoid "it connects to the internet" or "it helps packets find their way": examiners say these are not specific enough.

9. Differences between circuit and packet switching (2 questions)

1. Each difference must mention **both** methods in one sentence: "In circuit switching all data follows the same route, but in packet switching packets can take different routes."
2. Use a different row of the comparison table for each difference; two wordings of the same idea score once.
3. "**One benefit of each**": one from each method's own list.

10. How BitTorrent shares files (2 questions)

Give the steps as points: peer-to-peer network; a peer creates a **torrent file** holding **metadata**; the whole file starts on at least one peer; the file is split into **equal-sized pieces**; a new peer gets the torrent file and connects to the **tracker**, which keeps track of the peers in the **swarm**; it downloads pieces from other peers until it has the whole file; each piece it has makes it a **source**; a peer with the whole file that keeps sharing is a **seed**.

"**Explain: BitTorrent provides peer-to-peer file sharing**": users share files **directly** with each other; the computers act as **peers** of equal status, with **no central server** holding the file; thousands of users across the internet can share one file.

Worked examples

The bold codes show where marks are earned, as in Cambridge mark schemes. The number is how many marks.

- **B1** mark for a correct result on its own, with no method needed.

Example 1

Ana reads and sends email on her phone and her laptop.

- (a) Explain why protocols are essential for communication between Ana's devices and the mail server. **[2]**
- (b) Identify the protocol her email program uses to send a message, and describe its purpose. **[2]**
- (c) Ana wants to see the same inbox, with the same messages read and deleted, on both devices. Identify the protocol she should use to receive email and explain why it is better here than POP3. **[3]**
- (a)** Protocols are an agreed, standard set of rules for communication **B1**, so her devices and the server, which come from different manufacturers and run different software, can exchange data that each can understand **B1**.
- (b)** SMTP **B1**. It pushes the message from her device to her mail server, which sends it on to the recipient's mail server **B1**.
- (c)** IMAP **B1**. With IMAP the messages stay on the mail server and every device reads them from there **B1**, so both devices are kept synchronised. POP3 downloads the messages to one device and usually removes them from the server, so the other device would not see them **B1**.

Example 2

The TCP/IP protocol suite is used when a browser requests a web page.

- (a) Put the four layers of the TCP/IP suite in order from top to bottom. **[2]**
- (b) Describe how the layers of the stack work together when the browser sends its request to the web server. **[4]**
- (c) Describe the function of the Transport layer. **[3]**
- (a)** Application, Transport, Internet, Link. **B2** for all four in order, **B1** for two in the right place.
- (b)** Any four:
- The browser uses HTTP at the top, the Application layer, which is where the user interacts **B1**.
 - The request passes **down** the stack from the Application layer to the Link layer **B1**.
 - Each layer only passes data to the layer directly below it (or above it), and each adds its own header **B1**.
 - The Link layer sends the request over the network hardware **B1**.
 - At the web server the request passes **up** from the Link layer to the Application layer, where the server software reads it **B1**.
- (c)** Any three: it splits the data into segments (packets) for sending **B1**; it adds a header with a sequence number and port numbers **B1**; it sets up end-to-end delivery between the two hosts and controls the flow of data **B1**; at the receiver it reassembles the segments in order and has lost or damaged ones resent **B1**.

Example 3

A design company sends large drawing files to clients over the internet, and its staff hold live video calls with each other.

(a) Describe how packet switching is used to send a drawing file to a client. [4]

(b) Describe the function of a router in this process. [3]

(c) The company considers a dedicated circuit-switched line for its video calls. State one benefit and one drawback of circuit switching for the calls. [2]

(d) Give **two** differences between circuit switching and packet switching. [2]

(a) Any four: the file is split into equal-sized packets **B1**; each packet has a header with the source and destination IP addresses and a sequence number **B1**; each packet is sent independently and may take a different route **B1**, the best route at the time **B1**; the client's computer reorders the packets using the sequence numbers and reassembles the file **B1**; any missing or damaged packets are resent **B1**.

(b) Any three: the router reads the destination IP address in the packet's header **B1**; it uses its routing table, which holds details of neighbouring routers and the next hops **B1**; it decides the next hop on the best route and forwards the packet **B1**; it reduces the packet's hop count by 1 **B1**.

(c) Benefit: the whole bandwidth is available with no delays at switches once the circuit is set up, so the video arrives at a steady rate in real time **B1**. Drawback: the line is tied up for the whole call and no other data can use it, even during silences **B1**.

(d) In circuit switching a dedicated path is set up before sending, but in packet switching data is sent straight away **B1**. In circuit switching the data arrives in the order sent, but in packet switching the packets may arrive out of order and must be reordered **B1**.

Example 4

A student wants to download a large open-source software file using the BitTorrent protocol.

(a) Explain what is meant by "BitTorrent is a peer-to-peer protocol". [2]

(b) Describe how the student's computer obtains the file, and how it helps other users while doing so. [4]

(a) The users' computers share the file directly with each other **B1**; each acts as a peer of equal status, both downloading and uploading, with no central server holding the file **B1**.

(b) Any four: the student first gets the torrent file, which holds metadata about the software file **B1**; the program connects to the tracker, which keeps a list of the peers in the swarm **B1**; the file is split into equal-sized pieces, and the computer downloads different pieces from several peers at once **B1**; as soon as it has a piece it uploads that piece to other peers, becoming a source **B1**; when it has the whole file and keeps sharing it, it is a seed **B1**.

Common mistakes

- **Saying the protocol does the work.** A protocol is a set of rules; the devices and software follow them. Write "the router forwards the packet", not "the protocol forwards it".
- **Vague layer answers.** "The Internet layer uses IP addresses" scores nothing; "the Internet layer adds the source and destination IP addresses to the packet header" scores. Examiners give credit only for specific actions.
- **Putting jobs in the wrong layer.** Splitting data and sequence numbers are the Transport layer; IP addresses and routing are the Internet layer; frames and MAC addresses are the Link layer.
- **IMAP for sending.** IMAP and POP3 receive email; only SMTP sends it.
- **Answers that barely expand.** "Protocols are rules" is only half a point. Say what the rules make possible: devices from different manufacturers can communicate.
- **Comparing when not asked.** When the question asks for benefits and drawbacks of circuit switching, write only about circuit switching; a comparison with packet switching earns nothing.
- **Features instead of benefits.** "The packets take different routes" is a feature. The benefit is "if a link fails the packets can be re-routed, so they still arrive".
- **Giving the same point twice.** Two benefits that say the same thing in different words, or two differences that are one idea, score once.
- **One-sided differences.** "Packet switching splits data into packets" isn't a difference until you add "whereas circuit switching sends it as one continuous stream".
- **Generic router answers.** "The router connects to the internet" or "it helps the packet get there" is not enough. Say it reads the destination IP address, uses its routing table and chooses the next hop.
- **Writing "packages" or "data".** Use the correct word: **packets**.

How to get the marks

- **One mark, one point.** Mark schemes list separate points. Write each point as its own short sentence or bullet, and give one more than the marks if you can.
- **"Identify" or "State":** a short, precise answer, such as the name of a protocol or layer. **"Describe":** say what happens or what something does, with detail. **"Explain":** give reasons, with "because" or "so that". **"Outline":** the main points briefly.
- **Protocol questions** usually give one mark for the name and one for a description that fits only that protocol.
- **Layer questions** often cap the marks per layer (for example 3 for each of two layers in a 5-mark part), so spread your points across both layers.
- **Process questions** (packet switching, BitTorrent, the stack): give the steps in order, each with its technical term.
- **Follow the limits in the question.** "Do not include resending" or "do not describe the individual layers" means those points score nothing there.
- **Difference questions:** each difference names both methods.

- **"With a reason"**: the use alone earns one mark; the reason earns the other.

Quick check

1. Which email protocol is a push protocol, and which two are pull protocols?
2. A message is received by a host. List the TCP/IP layers in the order the message passes through them.
3. A packet leaves its sender with a hop count of 8. It reaches its destination after passing through 5 routers.
What is its hop count when it arrives? What would have happened if the route had needed 9 routers?
4. A message was sent as 5 packets. The receiver gets packets 4, 1, 5 and 2. Describe what the receiver does next.
5. Give **two** drawbacks of packet switching.

Answers

1. SMTP is the push protocol (it sends email). POP3 and IMAP are pull protocols (they fetch email from the mail server).
2. Link, Internet, Transport, Application: received data goes **up** the stack.
3. $8 - 5 = 3$. With 9 routers the count would reach 0 at the 8th router, which would discard the packet; the receiver would then find it missing and request it again.
4. It puts the packets in order by their sequence numbers (1, 2, 4, 5), sees that packet 3 is missing, and sends a request for packet 3 to be resent. When it arrives, the receiver reassembles the whole message.
5. Any two: delays while packets are reordered and reassembled or while lost packets are resent; packets can be lost; the delay varies, which is poor for real-time use; it needs complex protocols and plenty of memory to reassemble large messages.

Past questions to try

- [9618/33/O/N/25 Q3](#): how packet switching passes a message, without the resending.
- [9618/31/M/J/25 Q3](#): the Application and Transport layers, then packet switching.
- [9618/33/O/N/24 Q4](#): how the layers of the TCP/IP stack interact.
- [9618/33/M/J/23 Q5](#): where circuit switching is used, and its benefits and drawbacks.