

Security A2

Read online at pastopic.com/cambridge/computer-science-9618/notes/security

Last checked 30 September 2026

Read first: [Security, privacy and data integrity](#)

© 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). Threats, firewalls, validation, verification and the outline of a digital signature are the AS topic **Security, privacy and data integrity**; read it first. By the end of this topic you should be able to:

1. **Explain how encryption works**, using the words plaintext, ciphertext, key, encryption and decryption.
2. **Explain symmetric key cryptography** (one shared secret key) and **asymmetric key cryptography** (a matching public key and private key), and the differences between them.
3. **Explain how the keys are used to send a private message** from anyone in the public to one person or organisation, so that only they can read it.
4. **Explain how the keys are used to send a verified message** from one person or organisation to the public, so that everyone can check who sent it.
5. **Describe how data is encrypted and decrypted** with symmetric and with asymmetric cryptography.
6. **Give the purpose, benefits and drawbacks of quantum cryptography.**
7. **Describe the purpose of SSL and TLS**, how they are used when a client and a server start to communicate, and situations where they should be used.
8. **Describe how a digital certificate is acquired** from a Certificate Authority, what it contains, and **how it is used to produce and check digital signatures.**

You don't need the mathematics inside the encryption algorithms (such as how RSA picks its keys): only what each key does and who holds it.

Understand it

Encryption: the words

Encryption uses an **algorithm** and a **key** to turn readable data, the **plaintext**, into scrambled data, the **ciphertext**. **Decryption** uses a key to turn the ciphertext back into the plaintext. The algorithm is usually public; the security comes from keeping the right key secret.

A very old example: shift every letter three places along the alphabet. The algorithm is "shift each letter", the key is 3, so the plaintext HELLO becomes the ciphertext KHOOR, and shifting back by 3 decrypts it. Modern algorithms are far more complex, and their keys are long binary numbers: a 128-bit key has $2^{128} \approx 3.4 \times 10^{38}$ possible values, far too many to try one by one.

Encryption does **not** stop data being intercepted. It makes intercepted data **meaningless** to anyone without the key. Cryptography is used to make sure that:

- **only the intended receiver** can understand the message (confidentiality);
- the message **came from the claimed sender** (authenticity);
- the message **has not been altered** on the way (integrity);
- the sender **cannot later deny** sending it (**non-repudiation**).

Symmetric key cryptography

Symmetric encryption uses **one secret key**. The sender encrypts with it and the receiver decrypts with **the same key**, so both must have a copy.

- **Benefit:** it is a **simple, fast** process, with **shorter keys** (typically 128 or 256 bits), so it suits large amounts of data.
- **Drawback:** the key itself has to be **shared**. If it is sent over the network and intercepted, whoever has it can read every message. Everyone who needs to talk securely needs a key, and each key must be kept secret by all who hold it.

Asymmetric key cryptography

Asymmetric encryption uses **two different keys that are a matching pair**, made together by a key-generating algorithm:

- the **public key** is given to **anyone** who asks (it can be published);
- the **private key** is **kept secret** by its owner and **never transmitted** anywhere.

Data encrypted with one key of the pair can **only** be decrypted with **the other key of the same pair**. You can't work out the private key from the public key. The **purpose** of asymmetric cryptography is **better security**: only the public key is ever shared, so there is no secret key to intercept. The price is that it is **slower and more complex**, with **longer keys** (typically 2048 bits).

Who encrypts with which key depends on what you want.

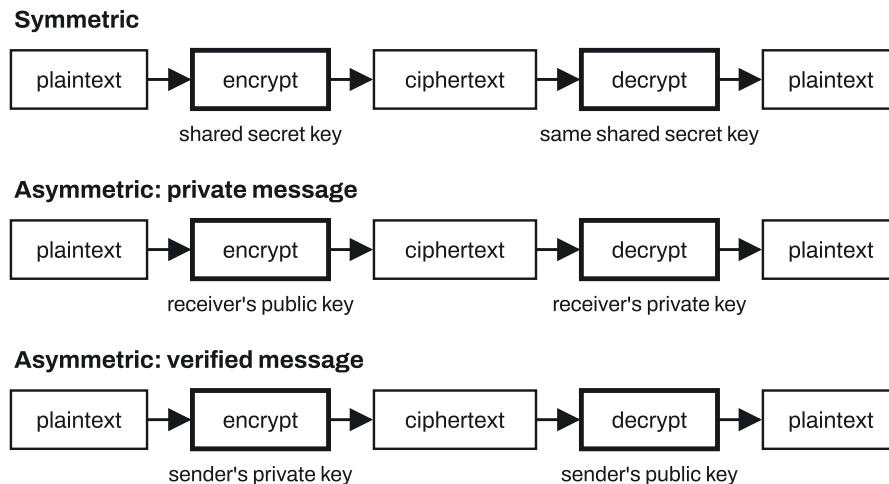
A private message, from the public to one organisation (confidentiality):

1. The organisation's computer **generates a matching pair** of keys, public and private.
2. The organisation makes its **public key** available; the sender **obtains the organisation's public key**.
3. The sender **encrypts the plaintext with the organisation's public key**, making ciphertext, and sends it.

4. The organisation **decrypts the ciphertext with its own private key**. Nobody else has that key, so nobody else can read the message; even the sender cannot decrypt it again with the public key.

A verified message, from one organisation to the public (authenticity):

1. The organisation **encrypts the message with its own private key** and sends or publishes it.
2. Anyone **decrypts it with the organisation's public key**.
3. If it decrypts correctly, it **must** have been encrypted with the matching private key, which only the organisation has, so everyone knows who sent it. It is not secret (anyone can decrypt it); it is **verified**.



A rule that covers both: **encrypt with the receiver's public key to keep it secret; encrypt with the sender's private key to prove who sent it**. In practice, because asymmetric encryption is slow, it is often used just to send a **symmetric key** safely, and the rest of the data is then encrypted symmetrically. SSL/TLS works this way.

Digital signatures

Encrypting a whole message with a private key is slow, so a **digital signature** signs only a short **digest** of it. A **hashing algorithm** turns the message into a fixed-length **message digest**; change one character of the message and the digest changes.

Producing the signature (sender):

1. **Hash** the message with the agreed hashing algorithm to produce a **message digest**.
2. **Encrypt the digest with the sender's private key**. The result is the **digital signature**.
3. Send the message and the signature together. For a confidential message, the message and signature can also be encrypted with the **receiver's public key**.

Checking the signature (receiver):

1. If it was encrypted for confidentiality, **decrypt the message and signature with the receiver's private key**.

2. **Decrypt the signature with the sender's public key** to recover the message digest that was sent.
3. **Hash the received message** with the **same** hashing algorithm to produce a second digest.
4. **Compare the two digests.** If they are the same, the message has not been altered and came from the holder of the private key. If they differ, it has been altered (or was not signed with that private key).

Digital certificates

A signature check is only as good as the public key used. If an attacker could hand you **their** public key and claim it was the bank's, they could sign fake messages that check out perfectly. A **digital certificate** solves this.

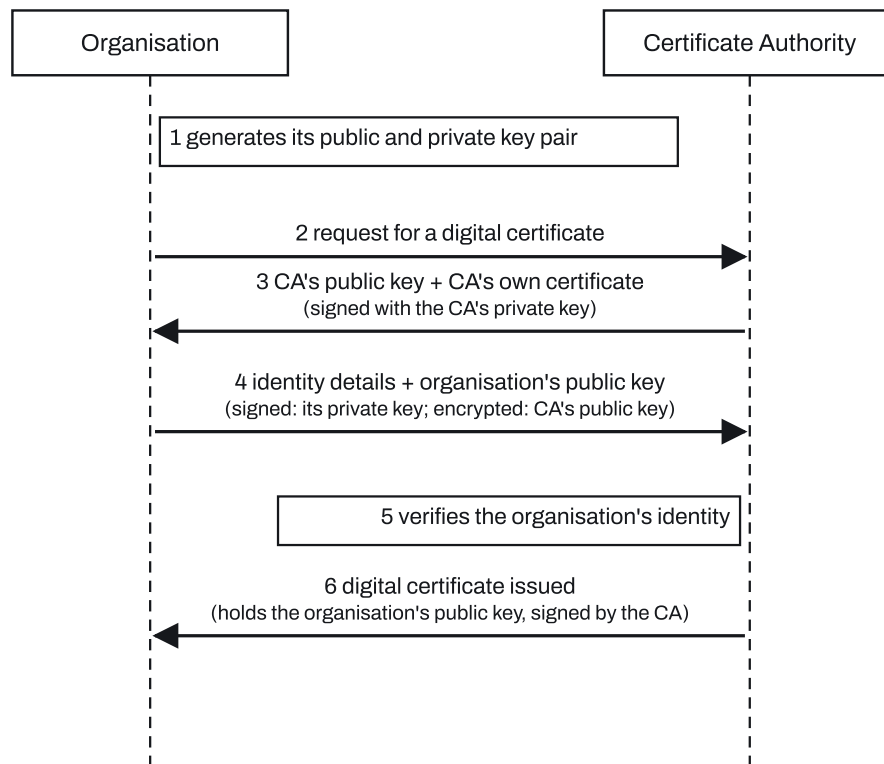
A **digital certificate** is an **electronic document**, issued by a trusted **Certificate Authority (CA)**, that **proves the online identity** of a website, organisation or person by **binding their public key to their identity**. It typically contains:

- the **name of the certificate holder** (the subject);
- the **holder's public key**;
- a **serial number** and a **version number**;
- the **validity dates** (valid from, expiry date);
- the **name of the issuer** (the CA);
- the **CA's digital signature**, made with the CA's private key.

Anyone can check the certificate with the CA's **public key** (browsers and operating systems already hold trusted CAs' public keys), so a certificate can't be forged or altered without the check failing.

How a certificate is acquired:

1. The organisation **generates its public and private key pair** on its own computer, using key-making software (some browsers and CA websites offer a tool for this). The private key stays with the organisation.
2. The organisation **requests a digital certificate from a CA**.
3. The CA **responds with its own public key and its own digital certificate**, signed with the **CA's private key**.
4. The organisation **gathers the information the CA needs to prove its identity**, including its **public key**, **signs it with the organisation's private key** and **encrypts it with the CA's public key**, then sends it.
5. The CA **verifies the organisation's identity** and details.
6. If they check out, the CA **generates and issues the certificate**, which **includes the organisation's public key** and is signed with the CA's private key.



Why a certificate is needed to validate a digital signature: the certificate **provides the sender's public key**, vouched for by the CA. Decrypting the signature with that key shows the signature was made with the **matching private key**, which belongs to the named holder. So the signature can't be **spoofed** (faked by someone claiming to be the sender), the receiver has evidence of **who signed** and that the document **was not altered**, and the sender can't deny signing it (**non-repudiation**).

Quantum cryptography

Quantum cryptography uses the **laws of quantum physics**, rather than hard mathematical problems, to exchange keys securely. The key is sent as a stream of **photons** (particles of light) along a **fibre optic cable**, each photon's **polarisation** carrying one bit. Measuring a photon **changes its state**, so if an eavesdropper reads the photons, the errors this causes show up and the sender and receiver know the key has been compromised. The exchanged key is then used to encrypt the data.

Its **purpose** is an encryption method that is, in practice, **impossible to break**, so data sent with it (usually along fibre optic cable) can't be read by an attacker.

Benefits

Security rests on the **laws of physics**, not on a mathematical algorithm that faster computers might one day break

Eavesdropping can be detected, because reading a photon changes its state

Drawbacks

Needs a **dedicated fibre optic line and specialist hardware**, which are **expensive** to buy and maintain

Limited range: it only works over relatively short distances

Benefits	Drawbacks
Almost impossible to hack; once the key has arrived safely it stays trustworthy, since an attacker can't store a copy to crack in future	Photons can have their polarisation disturbed inside the fibre, so a noticeable share of bits arrive wrong while the technology is still young
Longer, more secure keys can be exchanged	It only exchanges keys: extras that other methods offer, such as digital signatures , aren't part of it
Suits the most valuable government and industrial secrets	The same protection is open to people breaking the law , who could keep their messages hidden from investigators

Note that it **detects** eavesdropping; it does not **prevent** it.

SSL and TLS

Secure Socket Layer (SSL) and its newer replacement **Transport Layer Security (TLS)** are protocols that keep data **secure while it travels across a network**, most often the internet. Their **purpose** (or functions):

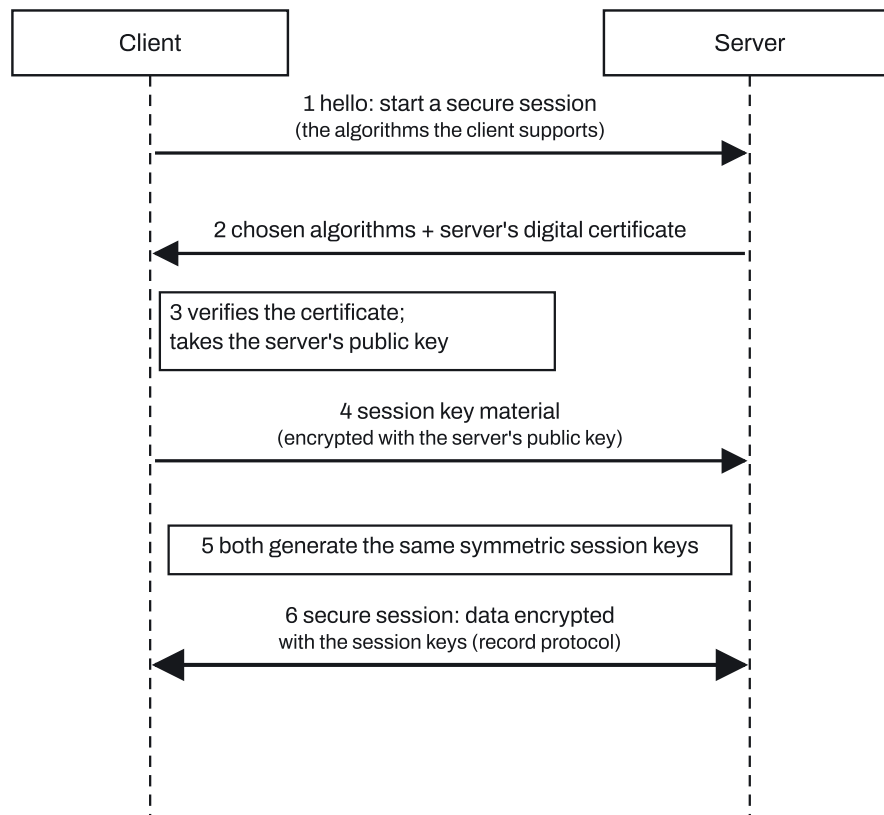
- **encrypt** the data sent between the two parties, so it is private;
- let each side **prove who it is** to the other (at least the server, using its digital certificate), so each knows it is talking to the genuine party;
- make sure the data keeps its **integrity** (it is not altered on the way).

TLS sits between the application layer (for example HTTP, which becomes **HTTPS**) and the transport layer. It is made of two main protocols:

- the **handshake protocol**, which **sets up a secure connection**: the server and client **authenticate** each other and **agree the encryption algorithms and keys**;
- the **record protocol**, which then **encrypts and authenticates the data exchanged** between client and server, securing the application's data and its integrity, and deals with the format of the data sent.

Using SSL/TLS when client-server communication is initiated:

1. An **application** (such as a browser) **starts the connection** and becomes the **client**; the application that receives the connection becomes the **server**.
2. Before any data moves, the two sides carry out a **handshake** to set up the session.
3. The client **requests the server's digital certificate**, and the server **sends it**.
4. The client **verifies the certificate** (with the CA's public key) and so **obtains the server's public key**.
5. The **encryption algorithms are agreed**, and **symmetric session keys are generated**, with the key material sent under the server's public key so only the server can read it.
6. A **secure session is established**: from now on the data is encrypted with the symmetric session keys (the record protocol).



This is asymmetric and symmetric cryptography working together: asymmetric keys solve the key-sharing problem once, and fast symmetric encryption carries the data.

Situations where SSL/TLS is appropriate: whenever data crosses a network and must stay private or unaltered, for example:

- sending **authentication data**: passwords, login details, session cookies;
- sending data whose **content must arrive unchanged**, in either direction, such as answers typed into a web form or a page of exam results;
- sending **non-public data**: online banking and shopping (card details), email, medical or personal records.

Key facts and formulas

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

Formula	Status
Plaintext → (encrypt with a key) → ciphertext → (decrypt with a key) → plaintext	Learn it
Symmetric: one shared secret key encrypts and decrypts; fast, shorter keys (128/256 bits), but the key must be shared	Learn it
Asymmetric: a matching public/private key pair; one encrypts, only the other decrypts; slower, longer keys (2048 bits), only the public key is shared	Learn it

Formula	Status
Private message: encrypt with the receiver's public key , decrypt with the receiver's private key	Learn it
Verified message: encrypt with the sender's private key , decrypt with the sender's public key	Learn it
Digital signature = message digest (hash of the message) encrypted with the sender's private key	Learn it
Check: decrypt signature with sender's public key → digest 1; hash received message → digest 2; equal means unaltered	Learn it
Digital certificate: issued by a CA; holder's name and public key, serial and version numbers, validity dates, issuer, CA's signature	Learn it
Quantum cryptography: photons over fibre; eavesdropping detected because measuring changes the state	Learn it
SSL/TLS: encryption, authentication, integrity; handshake protocol + record protocol	Learn it
TLS handshake: certificate sent → verified → server's public key → algorithms agreed → symmetric session keys → secure session	Learn it

How to do it

In the Paper 3 questions we have tagged for this topic (20 questions, 2021–2025, repeats counted on each paper), the types came up this often. 13 of the 20 questions mix types, so the counts add up to more than 20.

Question type	Questions
Asymmetric encryption: how the keys are used	6
Digital signatures and the certificate's role	6
Digital certificates: what they are and how one is acquired	6
SSL/TLS: purpose, handshake and uses	5
Symmetric and asymmetric cryptography compared	5
Quantum cryptography	5

1. Asymmetric encryption: how the keys are used (6 questions)

For "explain how X uses asymmetric encryption to send a document securely" (4 marks):

1. The **receiver's** computer generates a **matching pair** of keys, public and private.
2. The receiver's **public key** is sent to (or obtained by) the **sender**.

3. The sender **encrypts the plaintext with the receiver's public key**, producing **ciphertext**, and sends it.
4. The receiver **decrypts the ciphertext with the receiver's own private key**; only it has that key.

Always say **whose** key: "the public key" alone is not enough.

Variations you will meet:

- **"State what is meant by a private key"**: the key that is **kept secret** (never published or transmitted), which has a **matching public key**, and which **decrypts data encrypted with that public key**.
- **"Describe the purpose of asymmetric key cryptography"**: give the **purpose**, not the steps: to provide **better security** by using **two different keys**, a public and a private key, where **one encrypts** and **only the matching key decrypts**.
- **"Describe the process of asymmetric encryption"** (2 marks): encrypt with the recipient's public key, decrypt with the recipient's private key. (Encrypting with the sender's private key and decrypting with the sender's public key also earns the marks: that is the verified message.)
- **Verified message to the public**: encrypt with the sender's private key; anyone decrypts with the sender's public key, which proves who sent it.

2. Digital signatures and the certificate's role (6 questions)

Producing a signature (3 marks): hash the message with the agreed hashing algorithm → a **message digest** → **encrypt the digest with the sender's private key**; that is the signature.

Checking it on receipt (4 marks):

1. The message and signature are **decrypted with the receiver's private key** (if they were sent encrypted).
2. The signature is **decrypted with the sender's public key** to recover the digest that was sent.
3. The received message is **hashed with the same algorithm** to make a new digest.
4. The two digests are **compared**: the same means the message was **not altered**; different means it **was**.

Variations you will meet:

- **"Explain why a digital certificate is required to validate a digital signature"** or **"the role of a digital certificate in creating a digital signature"**: the certificate **provides the (trusted) public key**, which is used to **validate the private key** that made the signature. Add: it makes the signature virtually impossible to **spoof**, gives evidence of the signer's identity and that the document was not altered, and gives **non-repudiation**.
- Name every key exactly: "sender's private key", "sender's public key", "receiver's private key". The examiners asked for the keys to be identified exactly.

3. Digital certificates: what they are and how one is acquired (6 questions)

"Describe what is meant by a digital certificate" (3 marks): an **electronic document** → used to **prove the online identity** of a website, organisation or individual → **issued by a Certificate Authority** → containing the holder's identifying information and **public key**.

"Identify two items in a digital certificate": any two of the holder's name, the holder's public key, serial number, version number, validity dates (start or expiry), the issuer (CA), the CA's digital signature.

"Outline the process of acquiring a digital certificate" (3–4 marks), in order:

1. The organisation **generates its public/private key pair** locally.
2. It **requests a certificate from a Certificate Authority (CA)**.
3. The CA replies with **its public key and its digital certificate**, signed with the CA's private key.
4. The organisation **gathers the information needed to prove its identity**, including **its public key**, and sends it **signed with its private key and encrypted with the CA's public key**.
5. The CA **verifies the organisation's identity**.
6. The CA **generates and issues the certificate**, including the **organisation's public key**.

Don't describe the SSL/TLS handshake here: examiners saw that swap often.

4. SSL/TLS: purpose, handshake and uses (5 questions)

- **Purpose or functions** (2 marks): communications **security (privacy) over the internet**; **encryption** of the data; **identification and authentication** of client and server; confidentiality and **integrity** of the data.
- **Two protocols of TLS** (4 marks): **handshake protocol**, which establishes a secure connection, authenticates the two sides and agrees the algorithms; **record protocol**, which **encrypts and authenticates the data exchanged** and keeps its integrity.
- **"Explain how SSL/TLS is used when client-server communication is initiated"** (4 marks), **in order**:
 1. an application starts the connection and becomes the client; the one receiving it becomes the server;
 2. the session begins with a **handshake**;
 3. the client **requests the server's digital certificate** (the server sends it);
 4. the client **verifies the certificate** and **obtains the server's public key**;
 5. the **encryption algorithms are agreed** and **symmetric session keys are generated**;
 6. a **secure session is established**.
- **Situations where it is appropriate** (2 marks): sending **passwords or session cookies**; data whose content must **arrive unchanged** in either direction (form answers, results); **non-public data** such as bank or card details.

5. Symmetric and asymmetric cryptography compared (5 questions)

A difference must mention **both** types. Use "..., whereas ...":

- symmetric uses **one key** to encrypt and decrypt, whereas asymmetric uses **a pair**, public and private;
- the symmetric key is **shared** by everyone who uses it, whereas in asymmetric only the **public key** is shared and the private key is kept secret;
- so the risk of the key being compromised is **higher** with symmetric, whereas asymmetric is **more secure**;
- symmetric is **simple and fast**, whereas asymmetric is **more complex and slower**;

- symmetric keys are usually **shorter** (128 or 256 bits) than asymmetric keys (2048 bits).

Variations you will meet:

- "Identify one other type of cryptography" (given asymmetric): **symmetric** or **quantum**.
- "Give two methods of key cryptography": **symmetric** and **asymmetric**.
- "Give two reasons for using key cryptography": so **only the intended receiver can understand** the message; so the message is **authentic** (from a trusted source); so it **has not been altered** in transmission; **non-repudiation**. "It stops hacking" is too vague.

6. Quantum cryptography (5 questions)

- **Purpose** (2 marks): an encryption method that is **practically impossible to break** → using the **laws of quantum mechanics / properties of photons**; or: to **detect eavesdropping** → because **the photons' properties change** when they are read. It also protects data over **fibre optic cables** and allows **longer keys**.
- **Benefits**: security based on **physics, not mathematical algorithms**; **eavesdropping can be detected**; almost impossible to hack; a key that arrives safely stays trustworthy; **longer keys**.
- **Drawbacks**: **dedicated fibre and specialist hardware**, and their **high cost**; **limited range**; photons' **polarisation can be disturbed** in the fibre, so many bits arrive wrong; no built-in extras such as **digital signatures**; people breaking the law could use it too.

Give the benefit or drawback itself; don't compare with other methods or list features.

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

Kiri runs an online tutoring company. Her assistant suggests this plan for receiving parents' files: "We make one secret key, email it to every parent, and they encrypt their files with it before uploading."

- (a) Explain **one** weakness of the assistant's plan. **[2]**
- (b) Kiri decides to use asymmetric keys instead. For each stage, say **whose** key is used and **which** key: (i) the parent encrypts a file; (ii) Kiri's computer decrypts it. **[2]**
- (c) Explain why a parent who intercepts another parent's encrypted file still can't read it. **[2]**

(a) It is symmetric encryption, so the single key is both the encryption and the decryption key **B1**. Email can be intercepted, and every parent holds the same key, so anyone who gets it (an interceptor or any parent) can decrypt every file uploaded **B1**.

(b) (i) The parent encrypts with **Kiri's public key B1**. (ii) Kiri's computer decrypts with **Kiri's private key B1**.

(c) The file was encrypted with Kiri's public key, and only the matching private key can decrypt it; the public key itself can't **B1**. Kiri's private key is kept secret and never sent, so no parent has it **B1**.

Example 2

A council publishes a notice on its website. It wants residents to be able to check that the notice came from the council and has not been changed.

(a) Explain how the council produces a digital signature for the notice. [3]

(b) Explain how a resident's computer uses the signature to check that the notice has not been altered. [3]

(c) Explain why the resident's computer needs the council's digital certificate to trust the signature. [2]

(a) The notice is put through the agreed hashing algorithm **B1** to produce a message digest **B1**. The digest is encrypted with the **council's private key**, and this is the digital signature, sent with the notice **B1**.

(b) The signature is decrypted with the **council's public key**, giving the digest that was sent **B1**. The received notice is hashed with the same hashing algorithm to give a new digest **B1**. The two digests are compared: if they are the same, the notice has not been altered; if they are different, it has **B1**.

(c) The certificate, issued by a Certificate Authority, provides the council's public key **B1**, which is used to show that the private key that made the signature belongs to the council, so the signature cannot be spoofed by someone pretending to be the council **B1**.

Example 3

An online shop uses TLS on its website.

(a) Give **two** situations on the shop's website where TLS is needed. [2]

(b) Explain how TLS is used when a customer's browser starts to communicate with the shop's web server. [4]

(c) Before the shop can use TLS, it needs a digital certificate. Describe how the shop acquires one. [4]

(a) When the customer logs in, so the password (and session cookie) are protected **B1**. When the customer enters card details at checkout, which are non-public and must not be altered **B1**.

(b) The browser starts the connection, so it is the client, and the web server is the server; the session begins with a handshake **B1**. The browser requests the server's digital certificate and the server sends it **B1**. The browser verifies the certificate and obtains the server's public key from it **B1**. The encryption algorithms are agreed and symmetric session keys are generated, and a secure session is established **B1**.

(c) The shop generates its own public and private key pair **B1**. It requests a certificate from a Certificate Authority, which sends its own public key and certificate **B1**. The shop sends the information that proves its identity, with its public key, signed with its private key and encrypted with the CA's public key **B1**. The CA verifies the shop's identity and then issues the certificate containing the shop's public key **B1**.

Common mistakes

- **Not saying whose key.** "It is encrypted with the public key" loses the mark; write "the **receiver's** public key" or "the **sender's** private key". Examiners stress that every key must be identified exactly.
- **Getting the keys the wrong way round.** For secrecy, the **receiver's public** key encrypts; for a signature, the **sender's private** key encrypts. A message encrypted with a public key can't be decrypted with the same public key.
- **Mixing up digital signature and digital certificate.** The signature is the encrypted digest attached to one message; the certificate is the CA-issued document that proves whose public key it is.
- **Describing SSL/TLS when asked how a certificate is acquired**, and the other way round. Acquisition is between the organisation and a CA; the handshake is between a client and a server.
- **Putting the handshake steps in the wrong order.** The certificate is sent and verified **before** the session keys are made.
- **Describing the process when asked for the purpose.** "The purpose of asymmetric cryptography" or "of quantum cryptography" wants what it is **for** (better security using two keys; a system that can't practically be broken), not how it works.
- **A "difference" that mentions only one side.** Write about both symmetric and asymmetric in each point.
- **"Quantum cryptography prevents eavesdropping."** It **detects** it, because reading the photons changes their state.
- **Vague answers.** "Too expensive" needs the reason (dedicated fibre and specialist hardware); "stops hacking" or "others can't read it" needs to say that only the intended receiver can understand the message.
- **Comparing when not asked.** Benefits and drawbacks of quantum cryptography are about quantum cryptography only.

How to get the marks

- **One mark per point.** A 4-mark "explain" needs four separate, correct points; in a process, give them **in order**, one step per sentence.
- **"State" and "Identify"** want a short answer: "Symmetric", "Handshake protocol".
- **"Describe the purpose"** wants what it achieves and why, not a list of features.
- **"Explain the differences"** wants both sides in every point, and several different points, not one point in detail.
- **"Explain how"** a key is used: name the key, whose it is, and whether it encrypts or decrypts.

- **Use the key words** the mark schemes reward: plaintext, ciphertext, matching pair, message digest, hashing algorithm, Certificate Authority, handshake, session keys, photons, eavesdropping, non-repudiation.
- **Answer in context.** If the question names people or a company, write "Sheila's public key", "the company's private key".

Quick check

1. A message is encrypted with a Caesar shift of 3 (each letter moves three places along the alphabet). Encrypt the plaintext SAFE. Is this symmetric or asymmetric encryption?
2. Priya wants anyone in her club to be able to confirm that an announcement came from her. Which key does she encrypt it with, and which key do members use?
3. A browser opens a certificate claiming to belong to a bank. Which **three** details inside it would the browser use to decide whether to trust it, and why?
4. These TLS handshake steps are jumbled. Put them in order: A. symmetric session keys are generated; B. the client verifies the certificate with the CA's public key; C. the server sends its digital certificate; D. the client asks for a secure connection; E. encrypted data is exchanged.
5. A government lab and a bank both consider quantum key exchange. The lab's two sites are 20 km apart; the bank's branches are spread across a continent. Which is the better fit, and give **one** reason for and **one** reason against for the lab.

Answers

1. $S \rightarrow V$, $A \rightarrow D$, $F \rightarrow I$, $E \rightarrow H$, so the ciphertext is VDIH. It is **symmetric**: the same key (3) encrypts and decrypts (shift back by 3).
2. She encrypts it with **her own private key**; members decrypt it with **Priya's public key**. Only her private key could have produced something her public key decrypts, so it must be from her.
3. For example: the **holder's name** (it must match the bank's site); the **validity dates** (it must not have expired); the **CA's digital signature** (checked with the CA's public key, so it proves a trusted CA issued it and nothing was altered). The issuer's name is also a fair answer.
4. **D, C, B, A, E**: the client asks, the server sends its certificate, the client verifies it (and gets the server's public key), session keys are generated, then data is exchanged under them.
5. The **lab**: quantum key exchange only works over short distances on dedicated fibre, so it can't reach across a continent. For the lab: any eavesdropping on the key is detected, because reading the photons changes their state (or: security rests on physics, not on maths a future computer might break). Against: the dedicated fibre and specialist hardware are expensive (or: disturbed photons give a high error rate).

Past questions to try

- [9618/32/O/N/25 Q7](#): other types of cryptography, and how an organisation uses its two asymmetric keys.
- [9618/32/M/J/22 Q7](#): obtaining a certificate, then producing and checking a digital signature.
- [9618/32/M/J/25 Q10](#): the two protocols of TLS and their purposes.
- [9618/33/M/J/22 Q8](#): reasons for cryptography, then the benefits and drawbacks of quantum cryptography.