Encryption & Security

A-Level Computer Science ▪ Topic 17

A-Level Computer Science

What we will cover

1. How encryption works
2. SSL / TLS
3. Digital certificates
4. Digital signatures
5. Recap

Symmetric and asymmetric

encrypt with the recipient's **public key** — only their matching **private key** can decrypt

security key

Worked example – who holds which key?

Alice sends Bob a secret. Whose key encrypts it, and whose decrypts it?

Encrypt with Bob's public key.

Decrypt with Bob's private key.

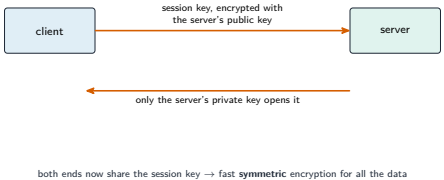
Encrypt with the recipient's public key – only they can read it. This is the reverse of signing.

SSL / TLS

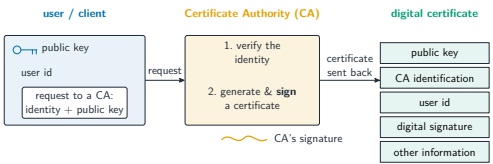
TLS 传输层安全 secures the web (the padlock, HTTPS):

- asymmetric keys exchange a shared session key
- then fast symmetric encryption for the data

> A hybrid: asymmetric for the handshake (safe key swap), symmetric for the bulk (speed).



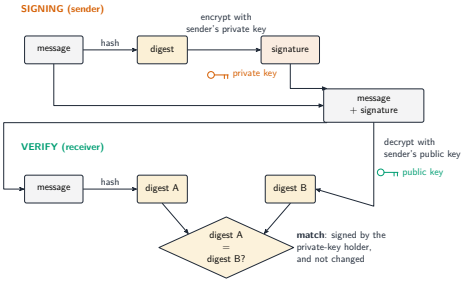
Digital certificates



A **digital certificate** 数字证书 proves a public key really belongs to a site. It is issued and signed by a trusted **certificate authority** 证书颁发机构.

> Without it you could be handed an impostor's public key – the certificate is what stops a man-in-the-middle.

Digital signatures



Hash 散列 the message, then encrypt the hash with the sender's private key:

- anyone verifies it with the public key
- proves authenticity (who) and integrity (unchanged)

> Signing is the reverse of encrypting: the private key does the locking, so only the owner could have signed.

Topic 17 – key takeaways

1 Keys

symmetric (fast) vs asymmetric (safe swap)

2 TLS

hybrid: asymmetric handshake, symmetric data

3 Certificates

a CA vouches for a public key

4 Signatures

hash + private key = authenticity + integrity

Check yourself

1. To send Bob a secret, whose key encrypts it?
2. Why is TLS a hybrid?
3. What does a certificate authority vouch for?
4. Which key signs a message?

Answers **1.** Bob's public key **2.** safe swap + speed **3.** a public key's owner **4.** the sender's private key