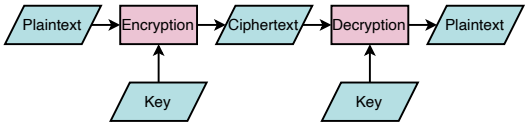
ALevel CS C21 Security(1)

Plaintext: data before encryption
Ciphertext: the result of applying an encryption algorithm to data



Security concerns relating to a transmission:
1. **Confidentiality**: Only the intended recipient should be able to decrypt the ciphertext.
2. **Authenticity**: The receiver must be certain who sent the ciphertext.
3. **Integrity**: The ciphertext must not be modified during transmission.
4. **Non-repudiation**: Neither sender nor receiver should be able to deny involvement in the transmission.
5. **Availability**: Nothing should happen to prevent the receiver from receiving the transmission.

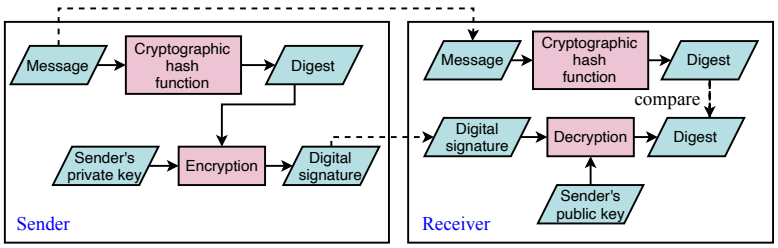
Symmetric key encryption: **one private key** is held by both sender and receiver and is used for both encryption and decryption
1. The issue with symmetric key encryption is delivery of the secret key.

Asymmetric key encryption: there is a **public key** and a **private key** one of which is used for encryption and the other for decryption
public key: sent to anyone who is going to partake in an encrypted communication.
private key: never sent to anyone.
The private and public keys are a matched pair.

Plaintext: data before encryption
Ciphertext: the result of applying an encryption algorithm to data
Asymmetric encryption:
An individual can encrypt a message with a **private key** and send this to a recipient who has the corresponding **public key** and who can then use this to decrypt the received ciphertext. **it could be used if it was important to verify who the sender was.** if the recipient finds that the decryption is successful, the message has in effect been received with a digital signature identifying the sender.
disadvantage: it is associated with an encryption of the whole of a message.
benefits:
1. Increased message security as one key is private
2. Allow message authentication
3. Allows non-repudiation
3. detects tempering

similarities between public key and private key:
1. both used in asymmetric
2. both use hashing algorithms
3. as a pair of key is required

differences between public key and private key:
1. private key only known to owner of the key pair, public key can be distributed to anyone
2. message encrypt with owner's public key, and only can decrypted by the owner's private key
3. digest are encrypted with the private key of the sender. and are encrypted with the public key of the receiver.



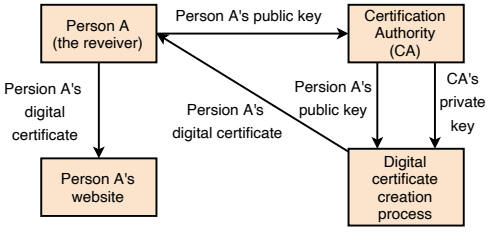
public cryptographic one-way hash function:
sender:
1. creates a number that is uniquely defined for the particular message, called a '**digest**'.
2. The private key is used to **encrypt the digest**.
3. The encrypted digest is the **digital signature**.
4. The message can be transmitted as **plaintext together with the encrypted digest as a separate file**.

receiver:
1. The same public one-way hash function is used to create a digest from the received message.
2. Then the encrypted version of the original digest is **decrypted using the public key.**
3. If the two digests are identical the receiver can be confident that the message is authentic and has been transmitted unaltered.

feature: the digital signature is different each time

CA digital certificate:
The receiver wants to be able to **receive secure messages from other individuals**, and these individuals want to be confident about the identity of the receiver. The public key must be made available in a way that ensures authentication
process of receiver to obtain a digital certificate to allow safe public key delivery:
1. An individual (person A) who is a would-be receiver and has a public –private key pair contacts a local CA.
2. The CA confirms the identity of person A.
3. Person A's public key is given to the CA.
4. The CA creates a public-key certificate (a digital certificate) and writes person A's public key into this document.
5. The CA uses encryption with the CA's private key to add a digital signature to this document. 6 The digital certificate is given to person A.
6. Person A posts the digital certificate on a website.

Receiver to obtain a digital certificate to allow safe public key delivery.



Digital certificate contains:
1. a hashing algorithm
2. a public key
3. serial number
4. dates valid
5. Name of subject
6. Name of issuer

Purpose of digital signature:
1. To ensure a document is authentic// came from a trusted source
2. To ensure a document has not been altered during transmission
3. Non repudiation

Digital signature step:
1. The message is hashed with the agreed hashing algorithm
2. ...to produce a message digest
3. The message digest is encrypted with the sender's private key
4. ... so the digital signature can be decrypted with sender's public key

similarities between digital certificate and digital signature:
1. both used for authentication
2. both are unique to the owner/subject
3. both use owner's public key
4. both make use of hash algorithm

differences between digital certificate and digital signature:
1. certificate obtained from issuing authority, signature created from a message
2. certificate provides authentication of owner, signature used to authenticate message that are sent by the owner
3. certificate remains unchanged whilst it is valid. new signature created for every message
4. only certificate provides extra information, only signature makes use of a private key

ALevel CS C21 Security(2)

RSA key generation algorithm:
1. Two very large prime numbers p and q are chosen and their product n is calculated.
2. The product (p-1)(q-1) is calculated.
3. A prime number e less than (p-1)(q-1) and not a factor of it is chosen (65537 is the usual choice).
4. Another number d is found which satisfies the condition that the product of d times e when divided by (p-1)(q-1) gives a remainder of
5. The public key becomes the pair (n,e).
6 The private key becomes the pair (n,d).

TSL:
1. A protocol with two layers:
   1. **Record protocol**: can be used with or without encryption
   2. **Handshake protocol**: permits the website and the client to authenticate each other and to make use of encryption algorithm(a **secure session** between client and website is established)
2. A **TLS/digital/public key certificate** is used for authentication
3. Handshake use asymmetric cryptography
4. establish a **shared session key**
5. TLS can make use of **session caching** which improves the overall performance
6. one the session has been established. the client and server can agree which encryption algorithm are to be used and can define the values for the session keys that are to be used.

TSL usages:
1. Browsers accessing secure websites
2. VPNs
3. Email
4. VOIP

TSL purpose:
1. provide **secure communication**
2. maintain **data integrity**
3. additional **layer of security**

TSL applications:
1. online banking
2. private email
3. online shopping
4. online messages

handshake process:
1. the client sending some communication data to the server
2. the client asking the server to identify itself
3. the server sending its digital certificate including the public key.
4. The client validates (the server's) TLS Certificate
5. The client sends its digital certificate (to the server if requested)
6. Client sends an encrypted message to the server using the server's public key
7. The server can use its private key to decrypt the message • and get data needed for generating symmetric key
8. Both server and client compute symmetric key (to be used for encrypting messages) // session key established
9. The client sends back a digitally signed acknowledgement to start an encrypted session
10. The server sends back a digitally signed acknowledgement to start an encrypted session

Two concerns access a website:
1. whether or not the **website is genuine**
2. whether we can **transfer sensitive personal data to the website**

When the SSL protocol is in place, the application protocol HTTP becomes HTTPS.

SSL facts:
1. Although SSL is referred to as a protocol, it is in fact **a protocol suite.**
2. There is a **Record Protocol** that deals with the format for data transmission.
3. There is also a **Handshake Protocol responsible for security**.
4. The operation of SSL happens without any action from the user.
5. The **starting point for SSL** implementation is a connection between the client and the server being established by TCP.
6. The client browser then invokes the Handshake Protocol from the SSP suite.
7. The Handshake Protocol requests from the server its **SSL certificate** which is a digital certificate confirming its identity.
8. The server sends this SSL certificate plus its public key.
9. The browser uses this public key to encrypt a key which is to be used as a one-off session key for symmetric key encryption to be used for the data transfer during the session.
10. There may also be a need at this time to agree which encryption algorithms are to be used.

Security protocols: Secure Sockets Layer(SSL), Transport Layer Security(TSL)

SSL step:
1. the user's web browser sends a message so that it can connect with the required website which is secured by SSL
2. The web browser then requests that the web server identifies itself
3. The web server responds by sending a copy of its SSL certificate to the user's web browser
4. if the web browser can authenticate this certificate, it sends a message back to the web server to allow communication to begin.
5. Once this message is received, the web server acknowledges the web browser, and the SSL-encrypted two-way data transfer begins.