Contents

Security requirements

Public key cryptography

- Key agreement/transport schemes
- Man-in-the-middle attack vulnerability

Encryption. digital signature, hash, certification

”Complete” security solutions

- SSL/TLS
- IPSec

Security requirements

The main security requirements of a secure system are:

Confidentiality - information is not made available to unauthorised entities

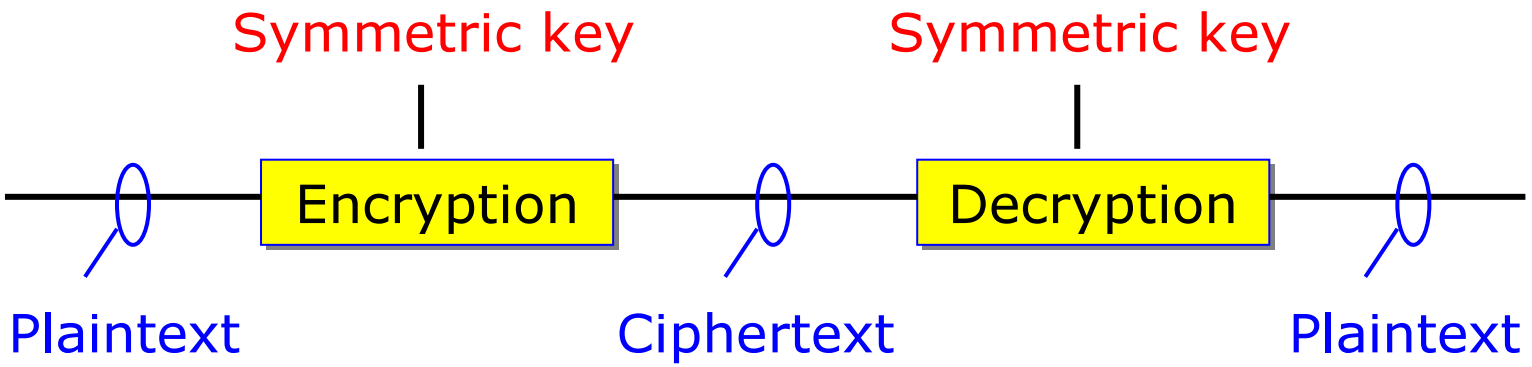
Integrity - information has not been altered during transmission in an unauthorised manner

Accountability – users must authenticate themselves before being able to access the system

Availability – in first hand this means prevention of Denial of Service (DoS) attacks.

Confidentiality

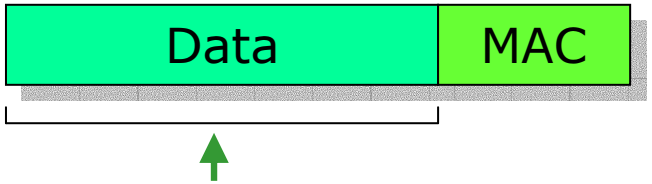
In packed-based transmission, confidentiality is achieved by encrypting the information (plaintext) before transmission and decrypting the ciphertext at the receiving end using the same (= symmetric) key.



Integrity protection

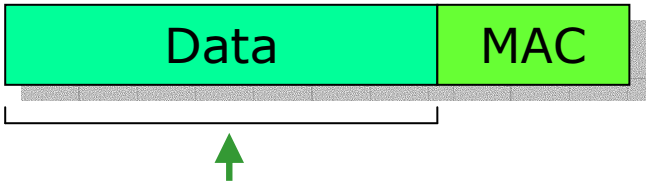
In packet-based transmission, integrity protection is ensured by using a message digest or hash algorithm to produce a Message Authentication Code (MAC) field that is appended to the data (usually before the encryption).

Transmitting end



Calculate MAC by applying hash algorithm to data

Receiving end



Calculate MAC again and check if = received MAC

Authentication

There are two widely used authentication methods :

- Shared key authentication: The authentication key is stored securely in the network and user equipment. The network sends a challenge to the user, who sends back a response encrypted with the authentication key. If the network can decrypt the response using the authentication key, the user has been authenticated.
- Digital signature: This is an authentication method, intended for packet-based transmission, using public key cryptography (see later slide).

Public key cryptography

The efficient usage of modern security mechanisms (e.g. SSL or SSH) would not be possible without a concept called public key cryptography.

Public key cryptography simultaneously makes use of both privat keys and public keys.

Private keys must be securely stored in the end user equipment, whereas public keys can be sent in unencrypted form over the network without compromising the security of the system.

Diffie-Hellman vs. RSA

Public key cryptography is generally used in two ways:

1. by generating a shared secret at both ends of the communications link (key agreement)

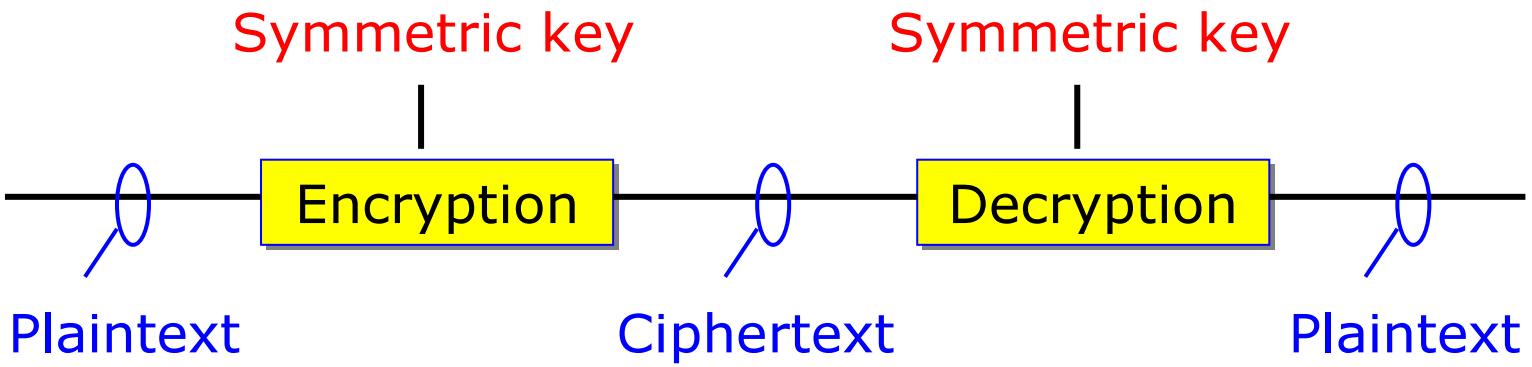
   Diffie-Hellman key agreement scheme

2. by sending a secret to the other end of the communications link (key transport)

   RSA (Rivest, Shamir, Adleman) scheme

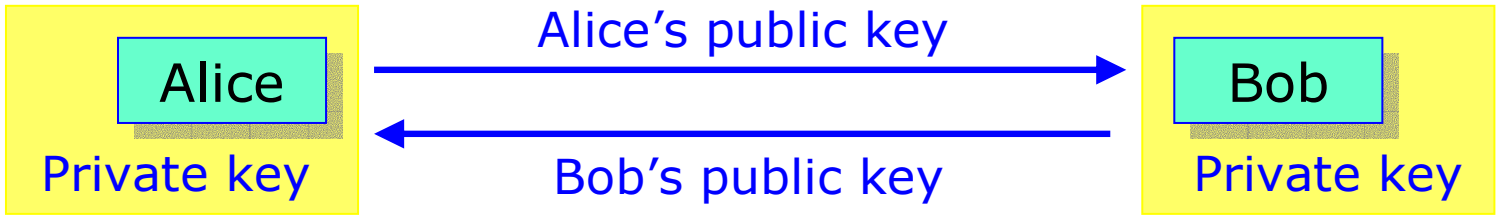
Symmetric keys vs. private/public keys

The word “key” in “key agreement” and “key transport” refers to the actual symmetric keys used in encryption and decryption, not the privat or public keys used in the public key cryptography scheme.



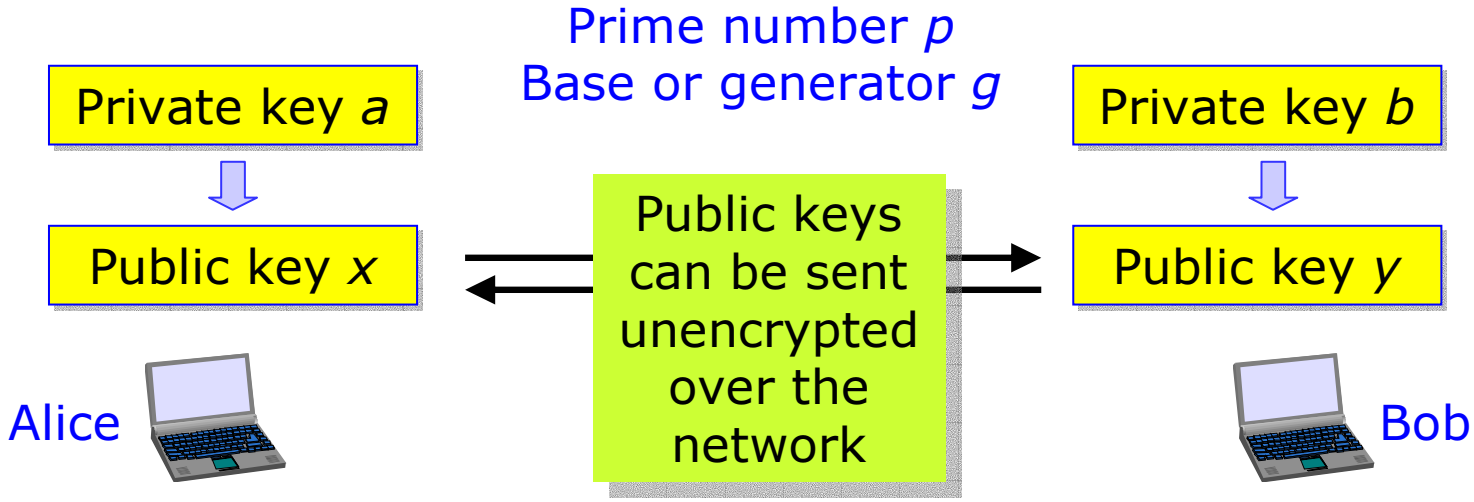
Key agreement scheme

In a key agreement scheme, two users, Alice and Bob, collectively generate a "shared secret" (for example, the symmetric key used in encryption and decryption) that only these two users know. To compute the shared secret, Alice combines her private key with Bob's public key. At the other end, Bob combines his private key with Alice's public key. In both cases, the result is the shared secret.



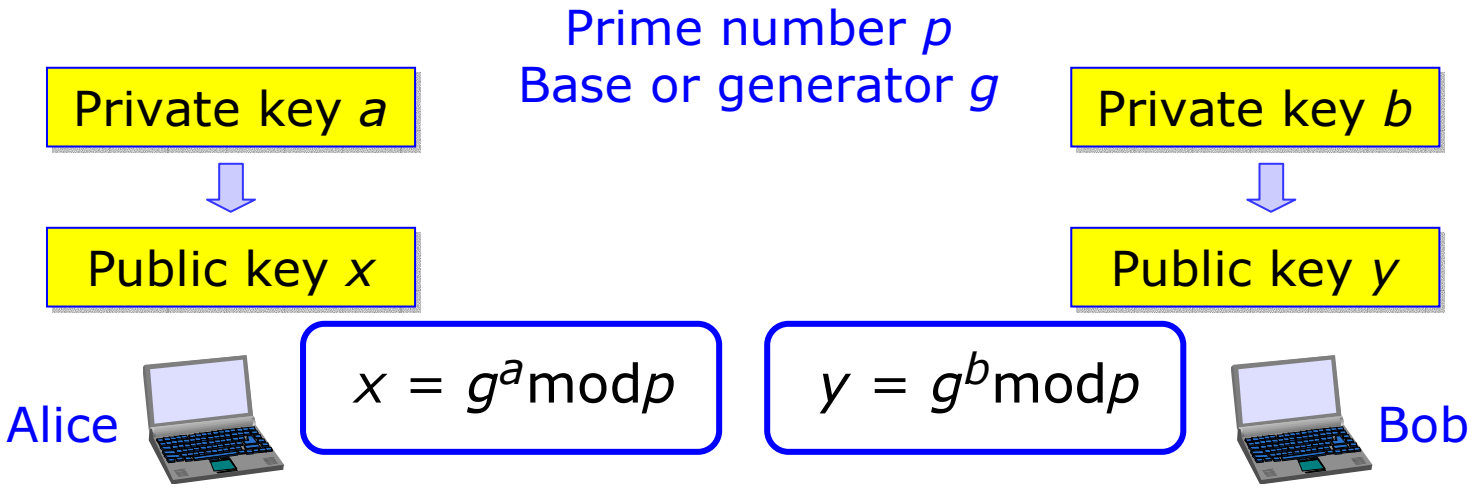
Diffie-Hellman key agreement scheme (1)

The Diffie-Hellman key agreement scheme is based on six numbers (*p*, *g*, *a*, *b*, *x*, and *y*)



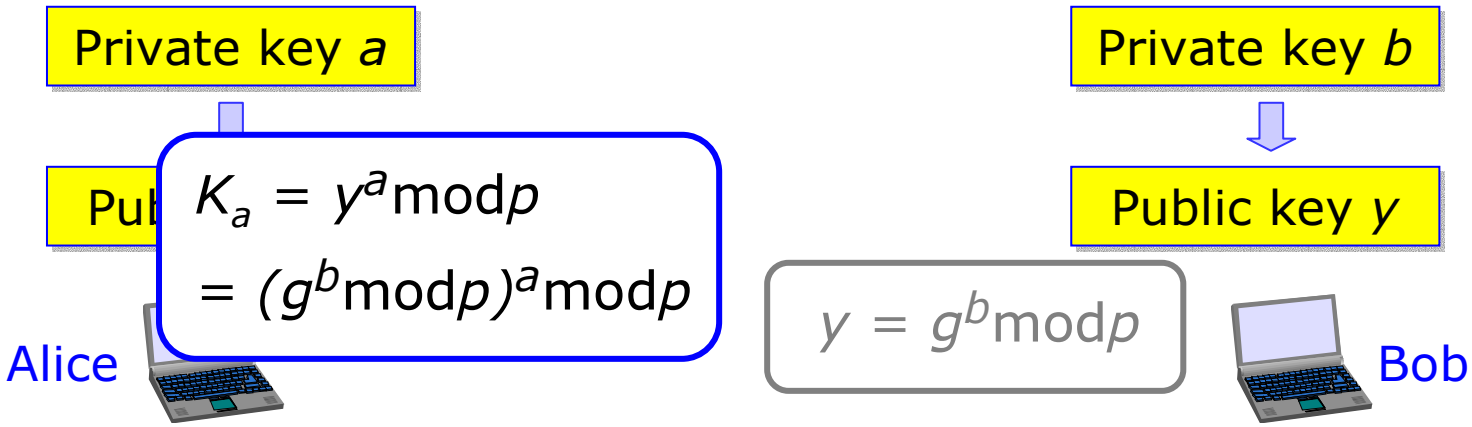
Diffie-Hellman key agreement scheme (2)

The public keys x and y are calculated using p, g, a, and b.



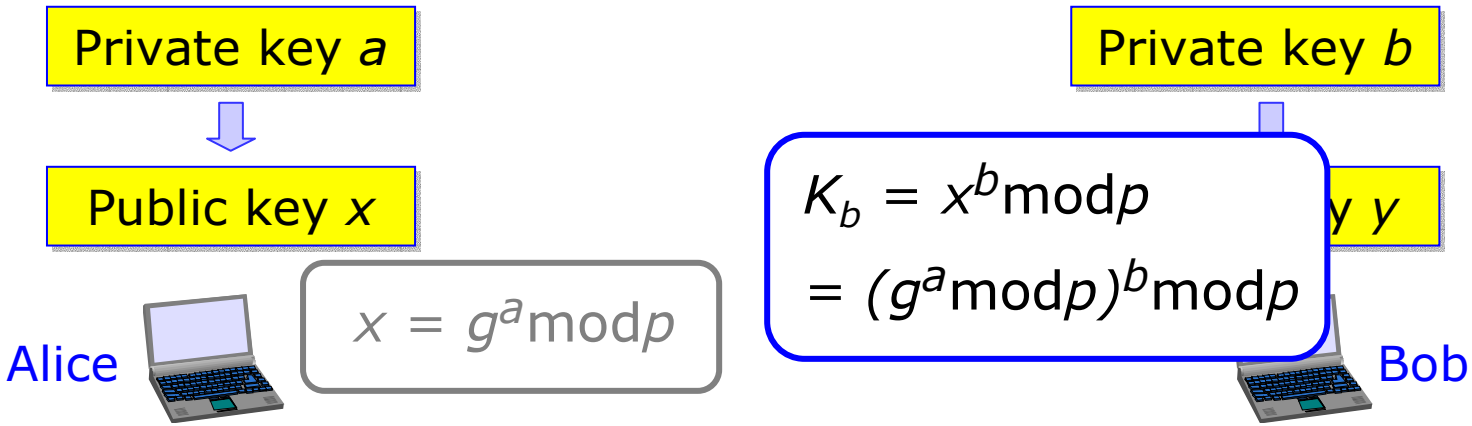
Diffie-Hellman key agreement scheme (3)

Alice calculates the shared secret (K_a) using Alice's private key and Bob's public key.



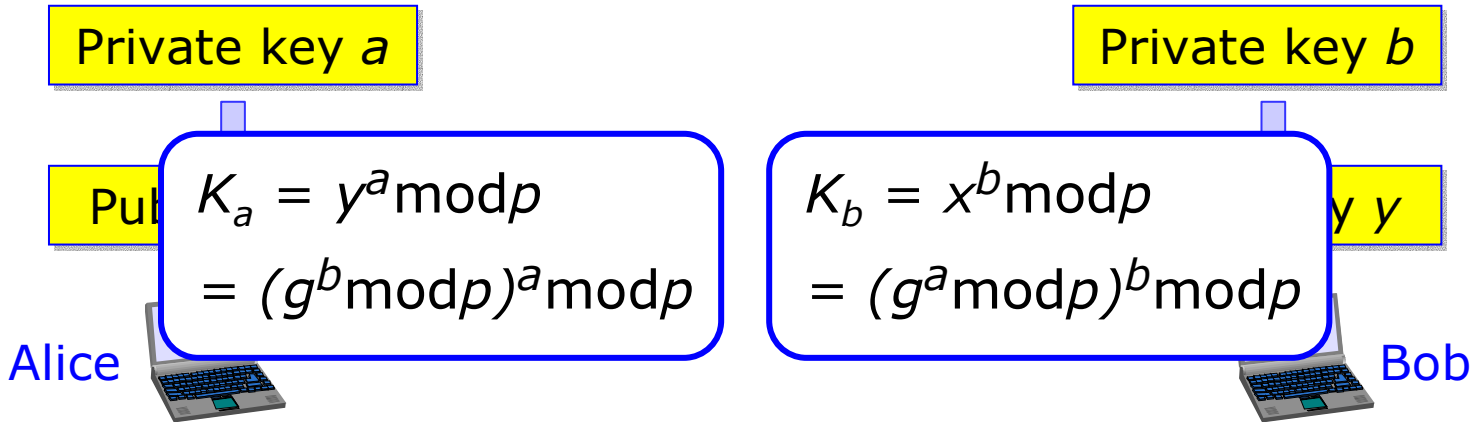
Diffie-Hellman key agreement scheme (4)

Bob calculates the shared secret (K_b) using Bob's private key and Alice's public key.



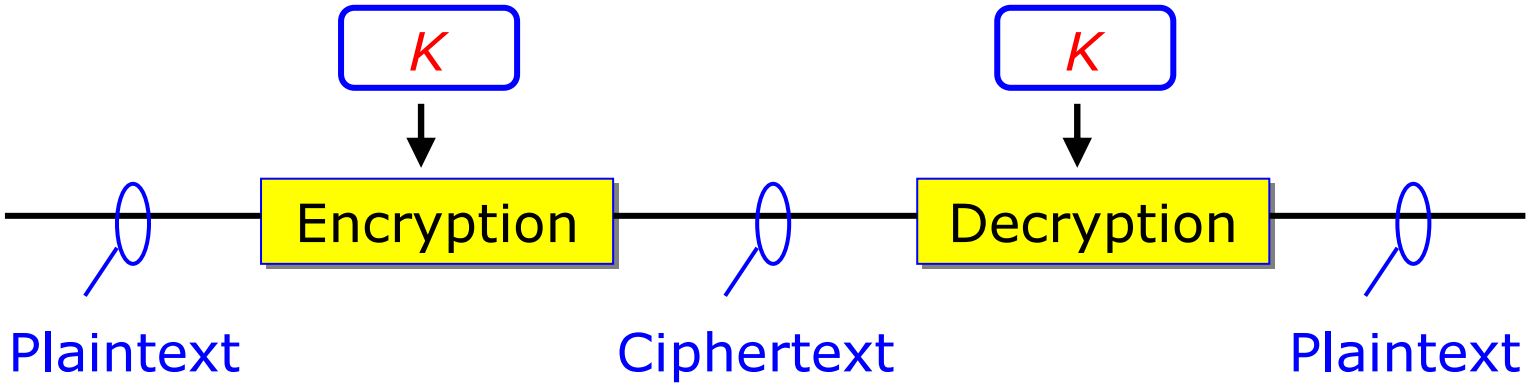
Diffie-Hellman key agreement scheme (5)

It turns out that $K_a = K_b$. From the attacker point of view, it is virtually impossible to find out the value of K by using x and/or y, provided the numbers are sufficiently large.



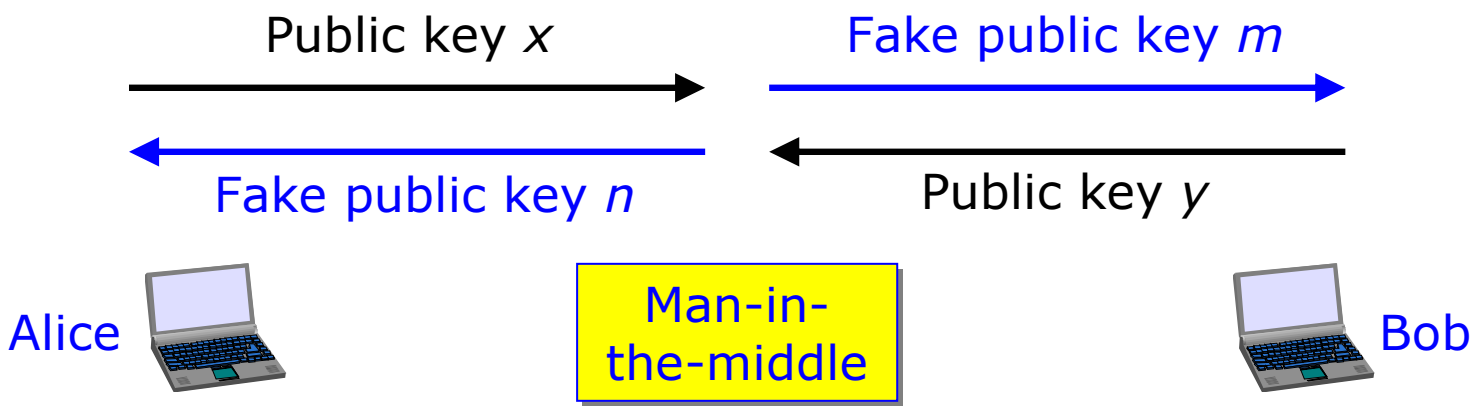
Diffie-Hellman key agreement scheme (6)

Computation of *K* is quite computationally intensive, so that public keys cannot be used for encrypting "real time data" (running at a high bit rate) directly, but rather for first generating a symmetric key *K* which is then used for encrypting and decrypting the data.



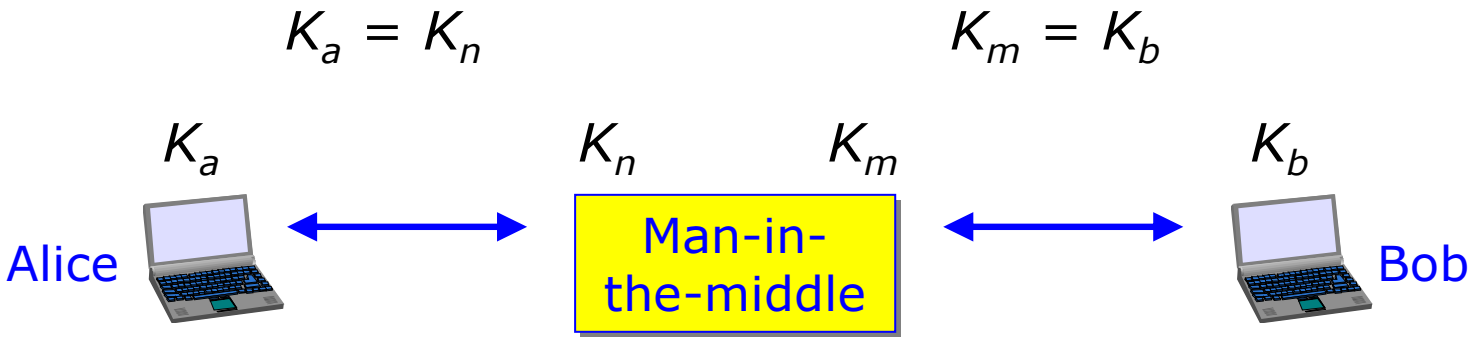
Man-in-the-middle attack vulnerability

Key agreement schemes are vulnerable to man-in-the-middle attacks and the public key in at least one direction should be sent in a signed certificate.



Man-in-the-middle attack vulnerability

After a successful man-in-the-middle attack, the man in the middle can decrypt the information encrypted by Alice and Bob (if the shared secret is the symmetrical key used for encryption and decryption).



Key transport scheme

If Alice encrypts a message with Bob's public key, only Bob can decrypt the message using his private key. No one else can decrypt the message, since Bob's private key is required for this purpose. In other words, in this way Alice can send a secret (for example, the symmetric key used in encryption and decryption) to Bob.

The public key algorithm Rivest, Shamir, and Adleman (RSA) is a key transport scheme. RSA was patented, but the patent expired in 2000. Due (among others) to this fact, RSA is widely used.

Digital signature (for authentication)

As an alternative way of using private and public keys, if Alice encrypts a message with her private key, anybody can decrypt the message using Alice's public key. No one else can encrypt the message in such a way that decrypting the message with Alice's public key will give a valid result. In other words, Alice has authenticated herself by providing a digital signature.

Digital signature = authentication

RSA vs. DSA

In addition to secure key transport, the public key encryption method RSA also offers authentication using a digital signature. Another algorithm that can be used for this purpose is Digital Signature Algorithm (DSA).

RSA: Key management + authentication

DSA: Only authentication, no key management

Diffie-Hellman: Only key management, no authentication.

Symmetrical encryption (for confidentiality)

Public key cryptography algorithms are far too slow to be used for encrypting the actual traffic to be carried over the communication link directly. For this purpose symmetrical encryption (= encryption and decryption are performed with the same key) must be used.

Some widely used symmetrical encryption algorithms are Advanced Encryption Standard (AES) and 3-fold Data Encryption Standard (3DES) for encrypting blocks of information, and Rivest Cipher 4 (RC4) for encrypting streams of information.

Message digests (for integrity protection)

In packet-based transmission, integrity protection is ensured by using a message digest or hash algorithm to produce a Message Authentication Code (MAC) field that is appended to the data (usually before the encryption).

If an attacker changes the content of the message during transmission, the calculated MAC and transmitted MAC at the receiving end will nor match.

Two widely used message digest or hash algorithms are Message Digest 5 (MD5) and Secure Hash Algorithm 1 (SHA-1).

Certificates

Key agreement (e.g. Diffie-Hellman) or key transport (e.g. RSA) schemes are vulnerable to man-in-the-middle attacks. A solution to this problem is to send the public key over the communication link using a signed certificate.

A certificate is a document that contains, along with the public key of the sender, the name of the certificate holder as well as the digital signature of an independent and trusted third party, called certification authority, to ensure the validity of the transmitted information. The certificate format is usually based on ITU-T recommendation X.509.

Key length

In the same way as long passwords make password guessing impractical, long keys make exhaustive searches impractical.

Every additional bit in the key doubles search time (and doubles the number of possible keys), so adding even a few bits to a key's length greatly increases the time needed to perform an exhaustive search.

In addition to using long keys, it is important to change keys frequently.

Putting it all together

There are complete security solutions that incorporate the various security mechanisms presented on the previous slides, that is key management schemes (Diffie-Hellman, RSA), authentication methods (RSA, DSA), encryption methods (AES, 3DES, RC4), integrity protection methods (MD5, SHA-1), additional security measures (e.g. anti-replay protection) and certificate management.

Important security solutions are SSL (TLS), SSH and IPSec. For wireless networks, there is Wired Equivalent Privacy (WEP) and Wi-Fi Protected Access (WPA).

SSL (TLS)

Secure Socket Layer (SSL) is a transport layer protocol (running on top of TCP) that offers security features for applications running on top of SSL, for example HTTP over SSL (HTTPS), Simple Mail Transfer Protocol (SMTP) over SSL, or Lightweight Directory Access Protocol (LDAP) over SSL (LDAPS). These are client-server types of applications.

The IETF adopted version 3.0 of the SSL protocol in 1999, renamed it Transport Layer Security (TLS) version 1.0 protocol and defined it in RFC 2246. SSLv3 and TLSv1 are compatible so far as the basic operation is concerned.

Basic SSL handshake operation (1)

Before data transport can take place over a secure SSL connection, the connection must first be established using a handshake procedure.

During the SSL handshake, the client and server need to agree on the algorithms that will be used to protect the data (first phase).

Then, the server sends its public key in a signed certificate to the client, so that the client can authenticate the server (second phase) using the RSA or DSA authentication method.

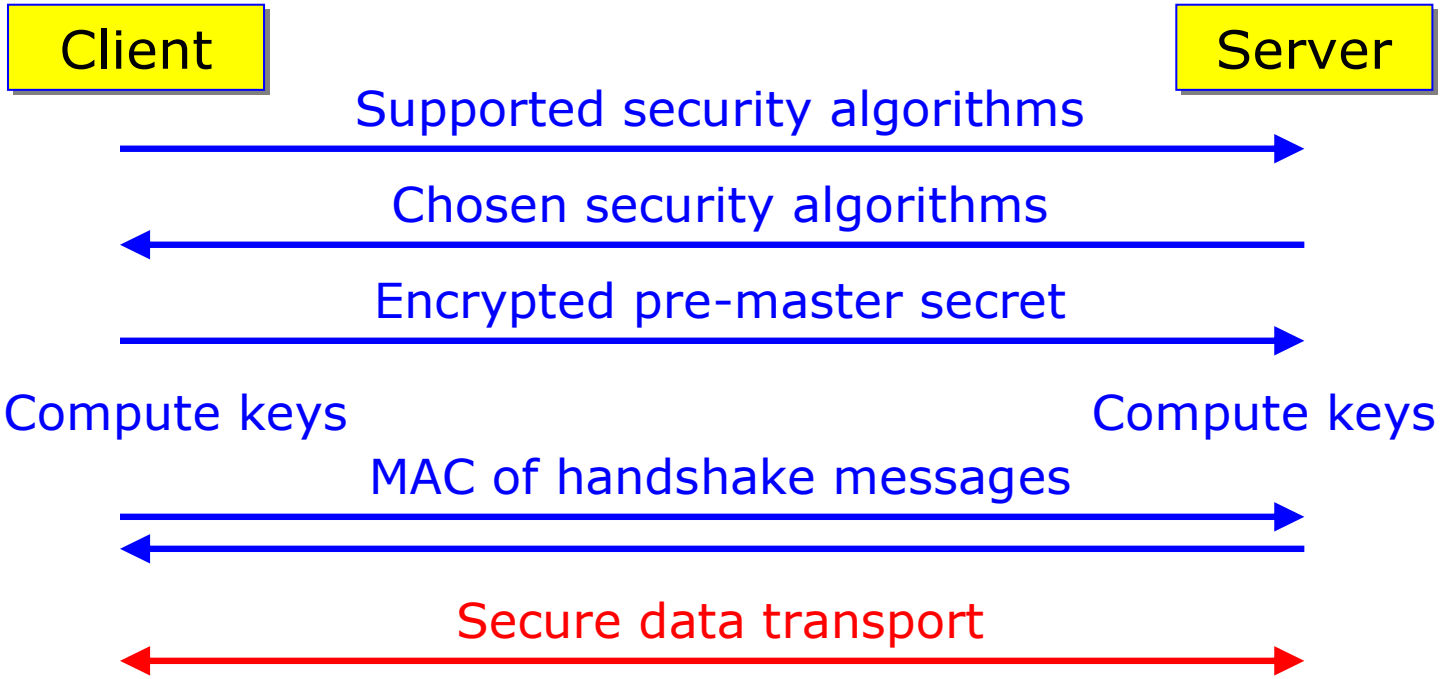
Basic SSL handshake operation (2)

The client generates a so-called pre-master secret, and sends this secret in encrypted form (using the server's public key for encryption) to the server (third phase).

Both the server and client side use the pre-master secret for generating the actual keys for symmetrical encryption as well as the message authentication code (MAC).

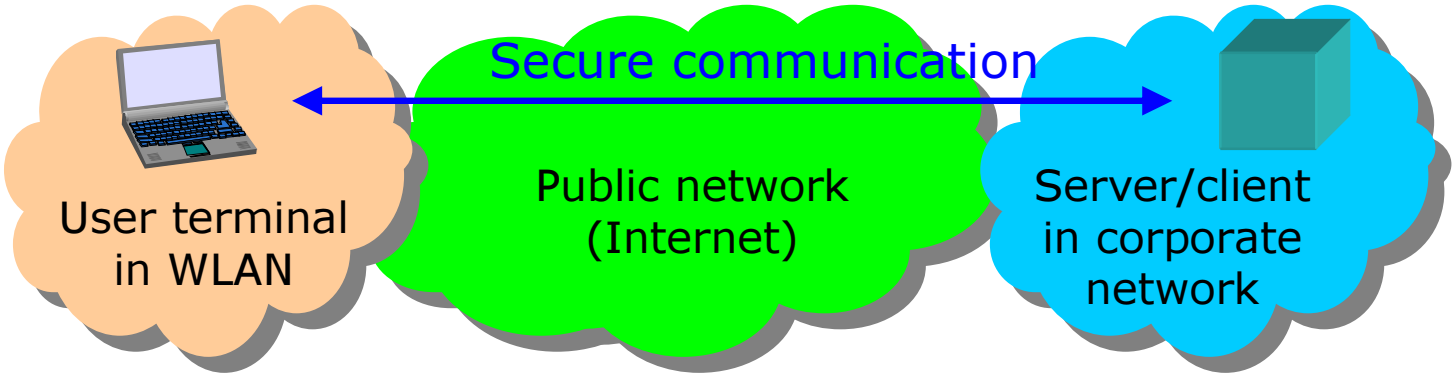
Finally, the client and server both calculate the MAC of the complete handshake information up to this point and send this information to the other side (fourth phase). Now the data communication can start.

Basic SSL handshake operation (3)



Virtual Private Network (VPN)

A virtual private network (VPN) can be used within a public telecommunication infrastructure, such as the Internet, to provide remote offices or individual users with secure access to their organisation's network.



Implementing VPN using IPSec

Secure VPN connections can be implemented using the IPSec protocol.

There exist many security solutions at higher protocol layers (e.g. SSL, SSH). However, IPSec is the only widely available and standardised protocol (rather: set of protocols) that operates (operate) at the network (or IP) layer.

IPSec is specified by the IETF in RFC 2401.

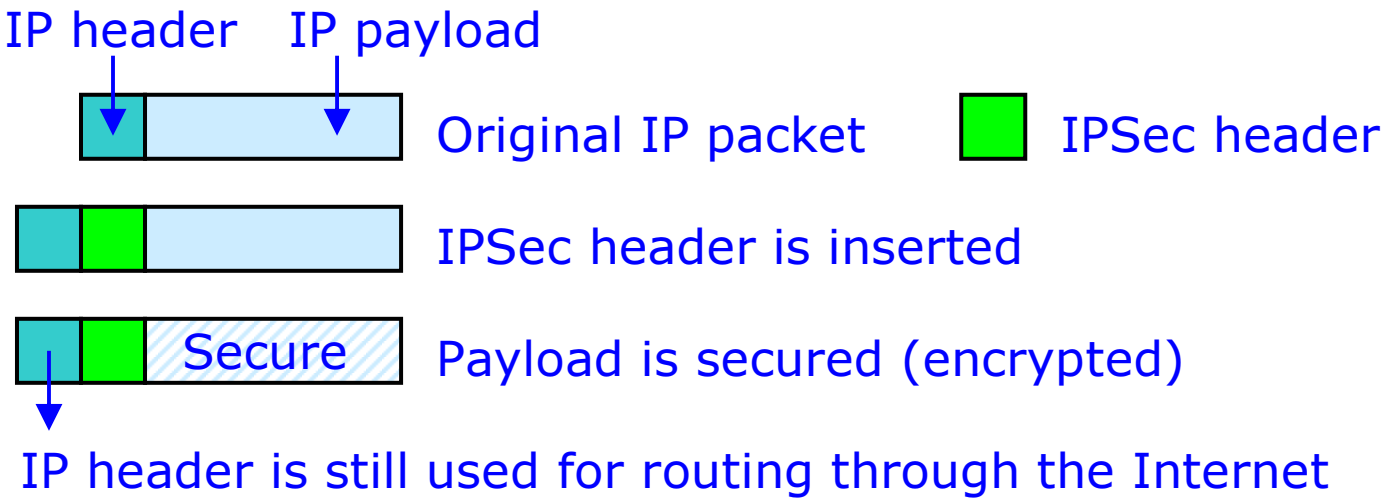
Two modes of IPSec

IPSec offers two modes:

- In Transport mode, only the IP packet payload is secured. The IP header is not encrypted, since it is used for routing the packet through the Internet. Transport mode is intended for end-to-end IPSec connections only.
- In Tunnel mode, the entire IP packet (including header) is secured. Tunnel mode is intended for applications involving security gateways.

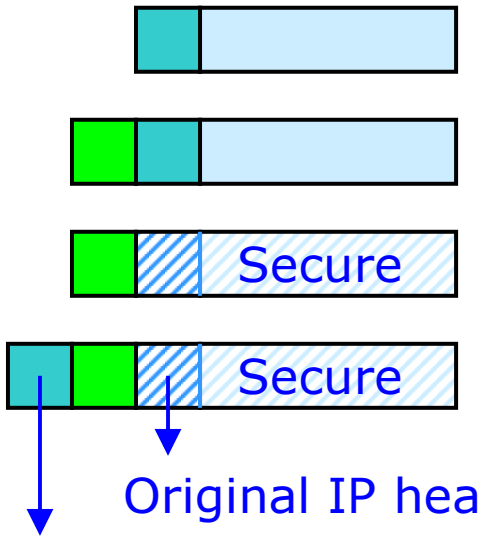
IPSec Transport mode

In Transport mode the IP headers are not encrypted:



IPSec Tunnel mode

In Tunnel mode the IP headers are also encrypted:



Original IP packet

IPSec header is appended

Whole packet is secured (encrypted)

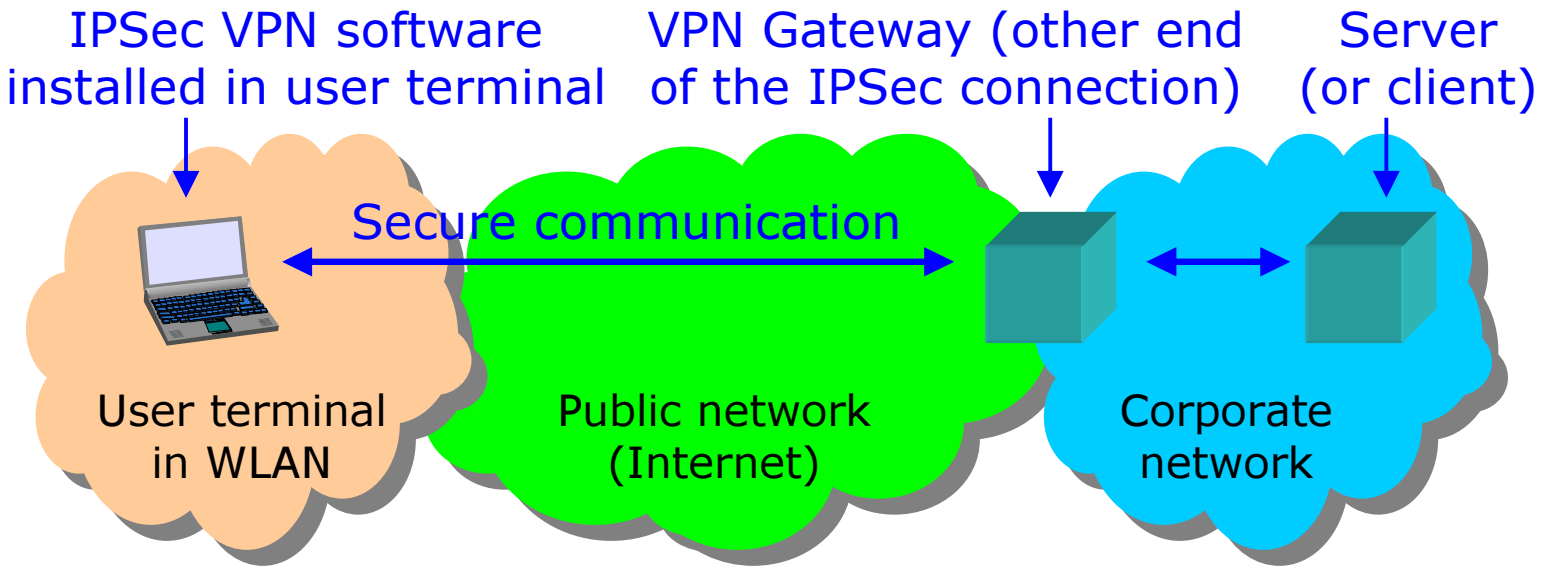
New IP header is appended (”tunneling”)

Original IP header cannot be used for routing

Instead, new IP header is used for routing through the Internet

IPSec Tunnel mode scenario

IPSec can (for instance) be used in the following way:



IPSec security features

Confidentiality

Content of IP packet (or payload) is encrypted.

Authentication

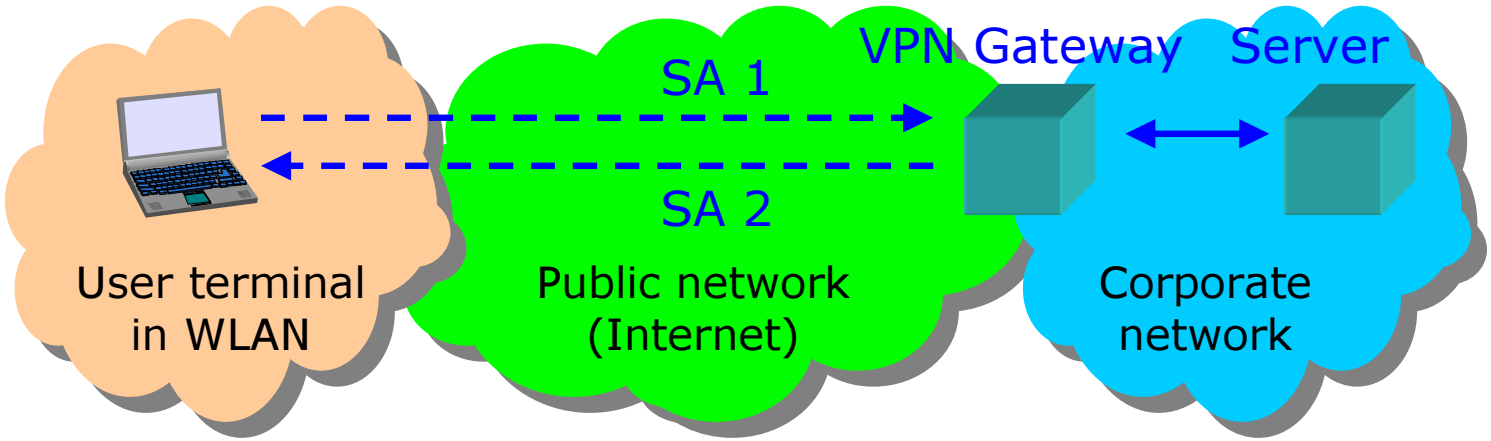
It is not possible to establish an IPSec connection if authentication fails. In the case of IPSec, authentication also ensures data integrity.

Anti-replay protection

It is not possible to send the same IP packet twice.

IPSec Security Association (SA)

Before it is possible to use the IPSec protocol between two points in an IP network, two Security Associations have to be formed - one for each transport direction.



IPSec Security Association (cont.)

Security Associations consist of agreements (by both sides) on protocols, algorithms and parameters, as well as exchange of public security keys.