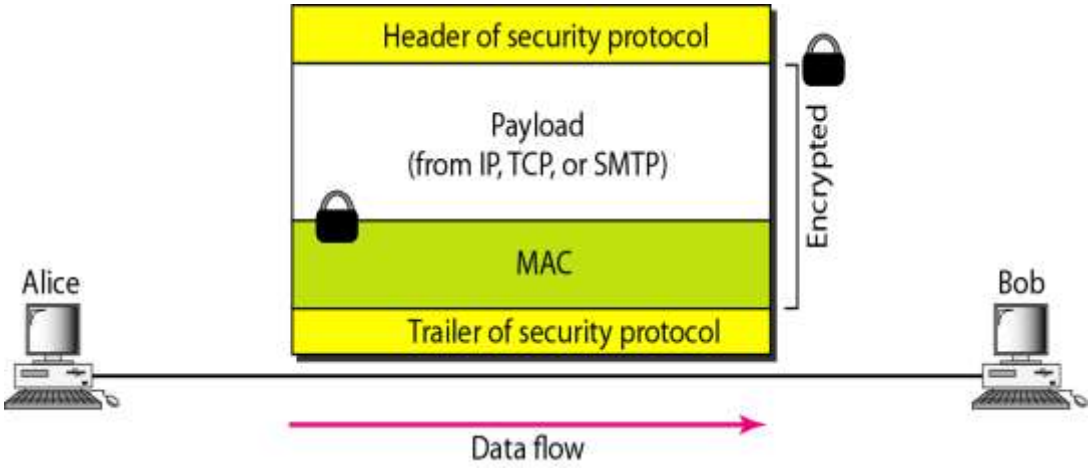
Data Communications and Networking Fourth Edition

Chapter 6 - Internet Security:

IPSec, SSL/TLS, PGP, VPN, and Firewalls

Common structure of three security protocols



- Certain security aspects, particularly *privacy* and *message authentication*, can be applied to the *network*, *transport*, and *application* layers of the Internet model.
  - IPSec protocol adds *authentication* and *confidentiality* to the IP protocol
  - How SSL (or TLS) does the same for the TCP protocol
  - How PGP does it for the SMTP protocol (e-mail)

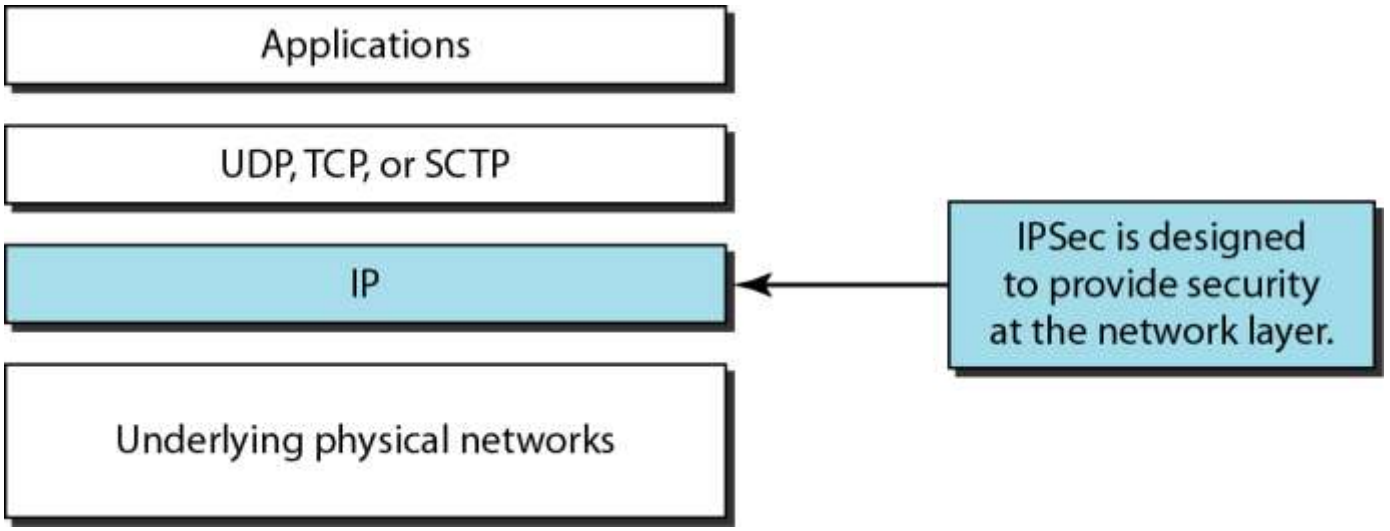
IPSecurity (IPSec)

IPSecurity (IPSec) is a collection of protocols designed by the Internet Engineering Task Force (IETF) to provide security for a packet at the network level.

Topics discussed in this section:

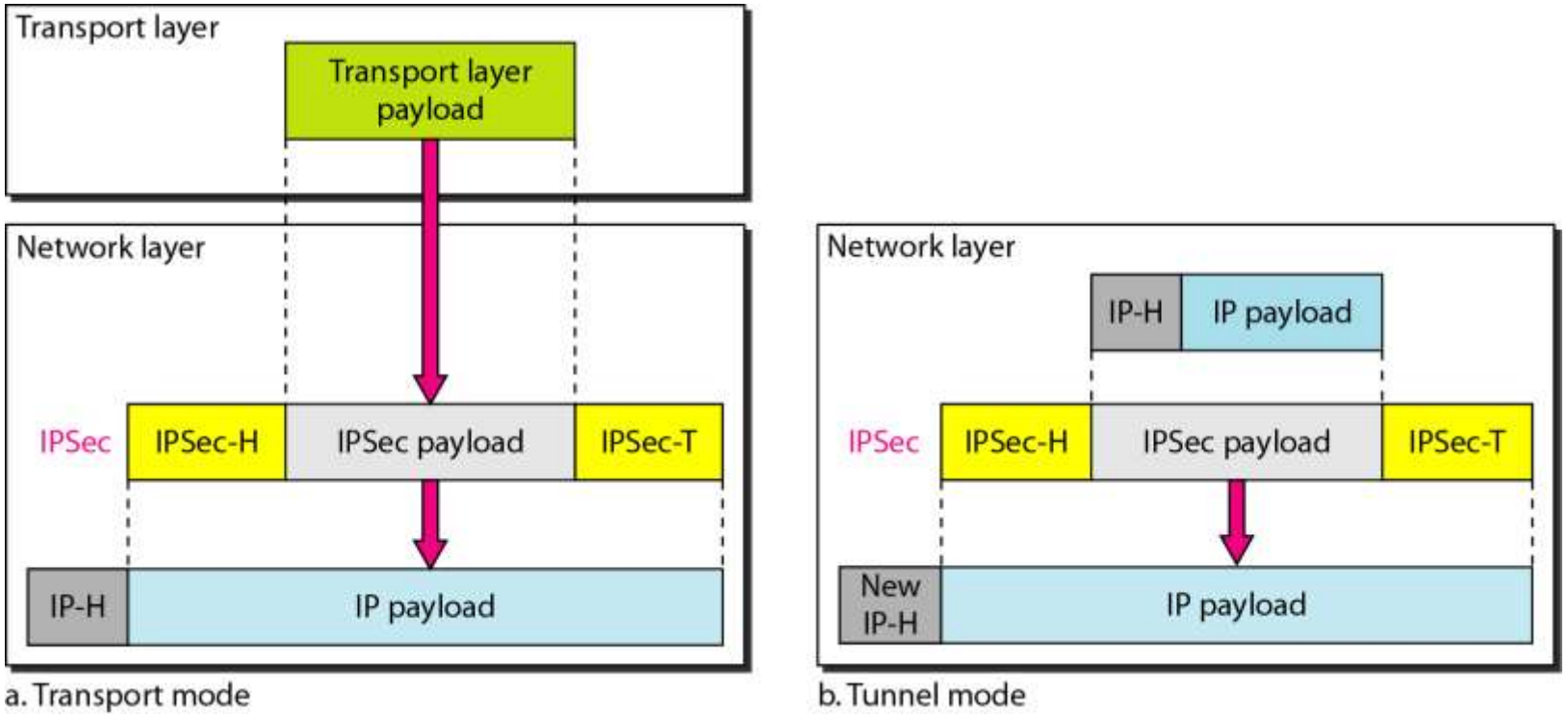
Two Modes
Two Security Protocols
Security Association
Internet Key Exchange (IKE)
Virtual Private Network

TCP/IP protocol suite and IPSec



IPSec helps to create *authenticated* and *confidential* packets for the IP layer.

Transport mode *and* ***tunnel modes*** *of IPSec protocol*



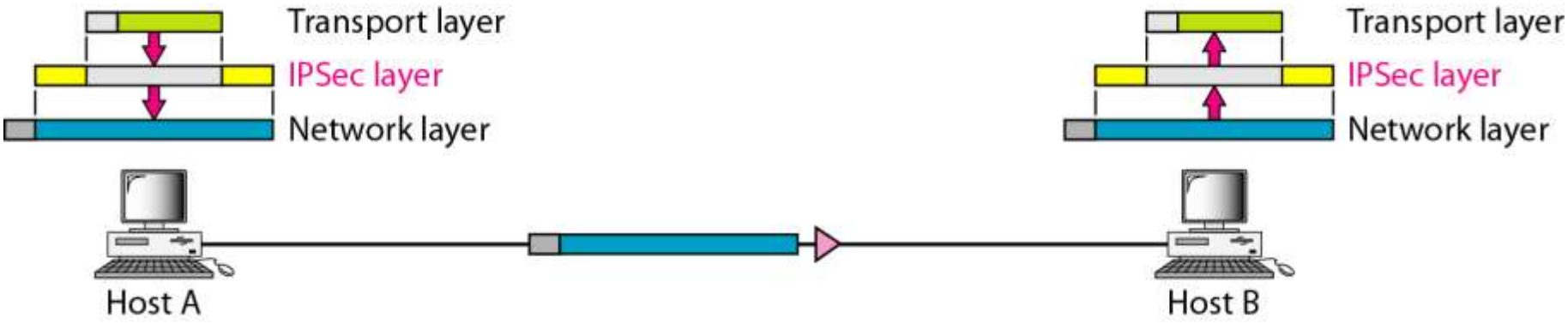
Transport mode

IPSec does not protect the IP header; it only protects the information coming from the transport layer.

Normally used when we need *host-to-host* (end-to-end) protection of data

The sending host uses IPSec to *authenticate* and/or *encrypt* the payload delivered from the transport layer.

Transport mode in action



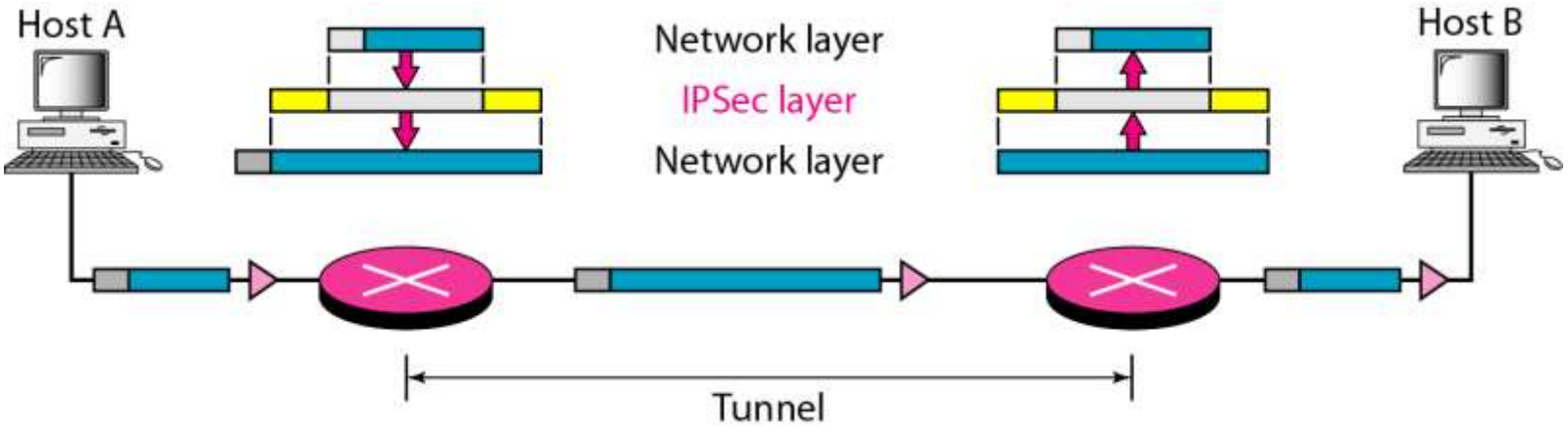
Tunnel mode

IPSec protects the entire IP packet including the original IP header.

IPSec adds a new IP header, which has different information than the original IP header

Normally used when either the sender or the receiver is not a host (e.g., router)

Tunnel mode in action



Two Security Protocols

IPSec defines two protocols:

- The *Authentication Header (AH)* Protocol
- The *Encapsulating Security Payload (ESP)* Protocol

Used to provide *authentication* and/or *encryption* for packets at the IP level.

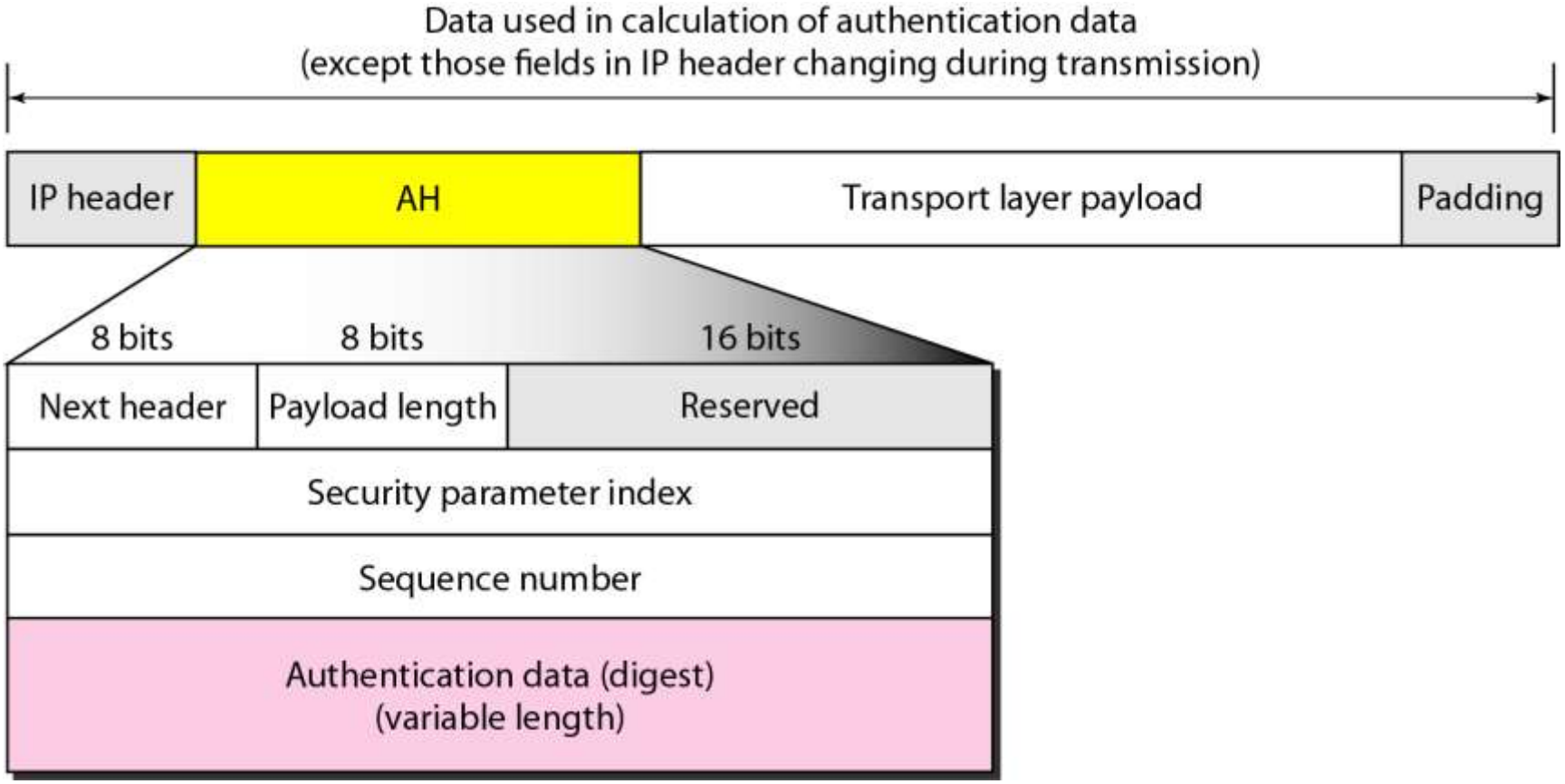
Authentication Header (AH) Protocol in transport mode

AH protocol provides *source authentication* and *data integrity* (of the payload), *but not privacy*.

AH uses a hash function and a symmetric key to create a message digest

The digest is inserted in the authentication header, and the AH is placed in the packet based on the IPSec mode.

Authentication Header (AH) Protocol in transport mode



Authentication Header (AH) Protocol in transport mode

Field Description

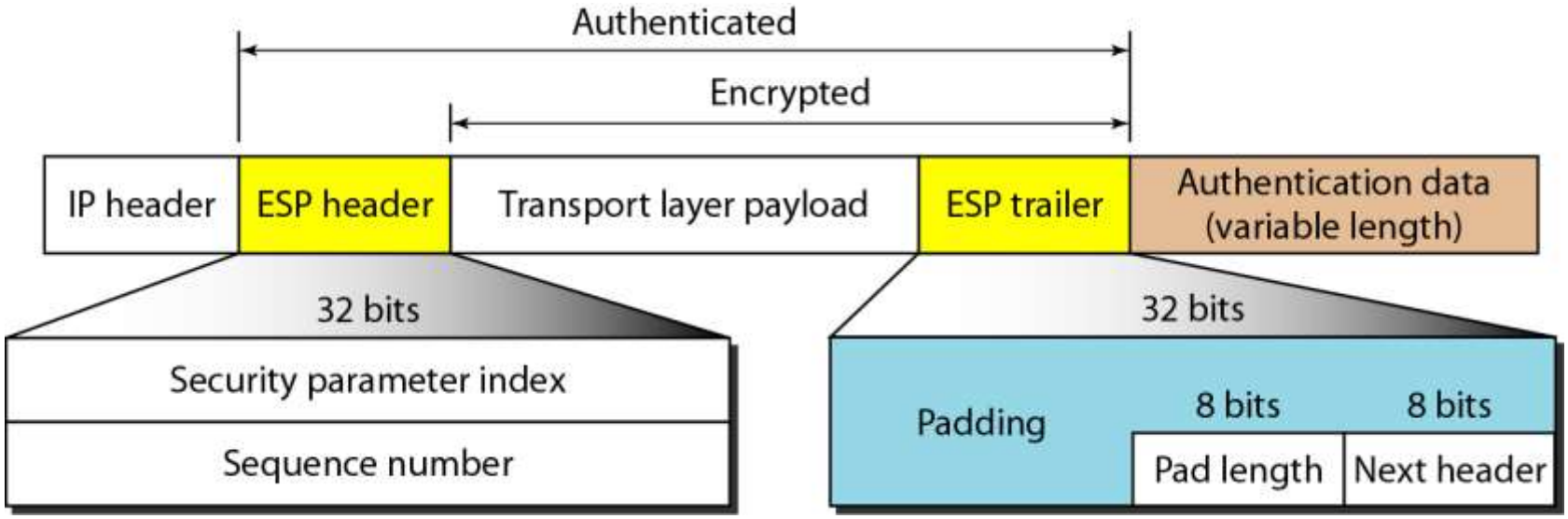
- **Next header** (8-bit): defines the type of payload carried by the IP datagram (such as TCP, UDP, ICMP, or OSPF).
- **Payload length** (8-bit): defines the length of the AH in 4-byte multiples, but it does not include the first 8 bytes.
- **Security parameter index** (32-bit): plays the role of a *virtual-circuit identifier* and is the same for all packets sent during a connection called a *Security Association (SA)*.
- **Sequence number** (32-bit): provides ordering information for a sequence of datagrams.
  - Prevents a playback.
  - Is not repeated even if a packet is retransmitted.
  - When SN reaches 232; a new connection must be established.
- **Authentication data**: is the result of applying a hash function to the entire IP datagram except for the fields that are changed during transit (e.g., time-to-live),

Authentication Header (AH) Protocol in transport mode

The steps of adding an AH

1. An authentication header is added to the payload with the authentication data field set to zero.
2. Padding may be added to make the total length even for a particular hashing algorithm.
3. Hashing is based on the total packet. However, only those fields of the IP header that do not change during transmission are included in the calculation of the message digest (authentication data).
4. The authentication data are inserted in the authentication header.
5. The IP header is added after the value of the protocol field is changed to 51.

Encapsulating Security Payload (ESP) Protocol in transport mode



Encapsulating Security Payload (ESP) Protocol in transport mode

ESP provides source authentication, data integrity, and privacy.

ESP's authentication data are added at the end of the packet which makes its calculation easier

In AH, part of the IP header is included in the calculation of the authentication data; in ESP, it is not.

ESP does NOT authenticate the IP header at all.

Encapsulating Security Payload (ESP) Protocol in transport mode

The ESP procedure follows these steps

1. An ESP trailer is added to the payload.
2. The payload and the trailer are encrypted.
3. The ESP header is added.
4. The ESP header, payload, and ESP trailer are used to create the authentication data.
5. The authentication data are added to the end of the ESP trailer.
6. The IP header is added after the protocol value is changed to 50.

IPSec services

Services	*AH*	*ESP*
Access control	Yes	Yes
Message authentication (message integrity)	Yes	Yes
Entity authentication (data source authentication)	Yes	Yes
Confidentiality	No	Yes
Replay attack protection	Yes	Yes

IPSec Services

Access Control: by using a Security Association Database (SADB). When a packet arrives at a destination, and there is no security association already established for this packet, the packet is discarded.

Message Authentication: The integrity of the message is preserved in both AH and ESP by using authentication data.

Entity Authentication: The security association and the keyed-hashed digest of the data sent by the sender authenticate the sender of the data in both AH and ESP.

Confidentiality The encryption of the message in ESP provides confidentiality.

IPSec Services

Replay Attack Protection In both protocols, the replay attack is prevented by using sequence numbers and a sliding receiver window

- Unique sequence number when the security association (SA) is established
- The number starts from 0 to 232 – 1.
- If sequence number = max =>
  1. Reset to zero,
  2. Old SA is deleted
  3. A new SA is established

To prevent processing of duplicate packets, IPSec mandates the use of a fixed-size window at the receiver (determined by the receive - default value of 64.)

Security Association (security association database (SADB))

- **Goal:** establish a set of security parameters

  like:

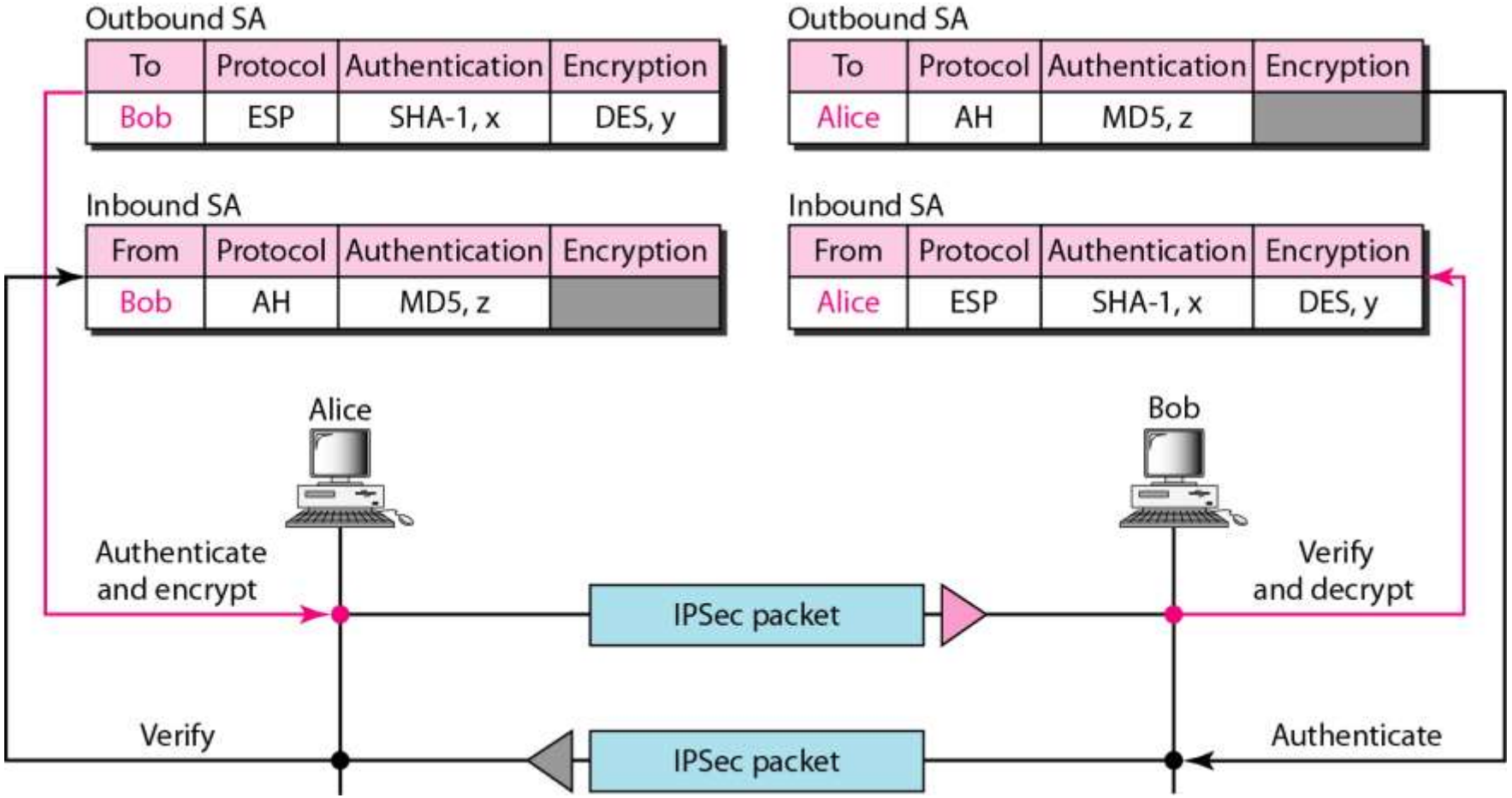
  - Security Protocol (AH or ESP), SPI (Security Parameters Index), Encryption Algorithm (AES-CBC, ChaCha20, 3DES), Integrity / Authentication Algorithm (HMAC-SHA-1), Keying Material, Sequence Number State, Mode of Operation, Lifetime…

  Methods:

  - Security parameters related to each datagram can be included in each datagram.
  - A set of security parameters can be established for each datagram.
  - A set of security parameters can be established between a sender and a particular receiver the first time the sender has a datagram to send.

- IPSec changes a connectionless protocol, IP, to a connection-oriented protocol
- This SA can be used all the time (unless a new one is established)

Simple inbound and outbound security associations

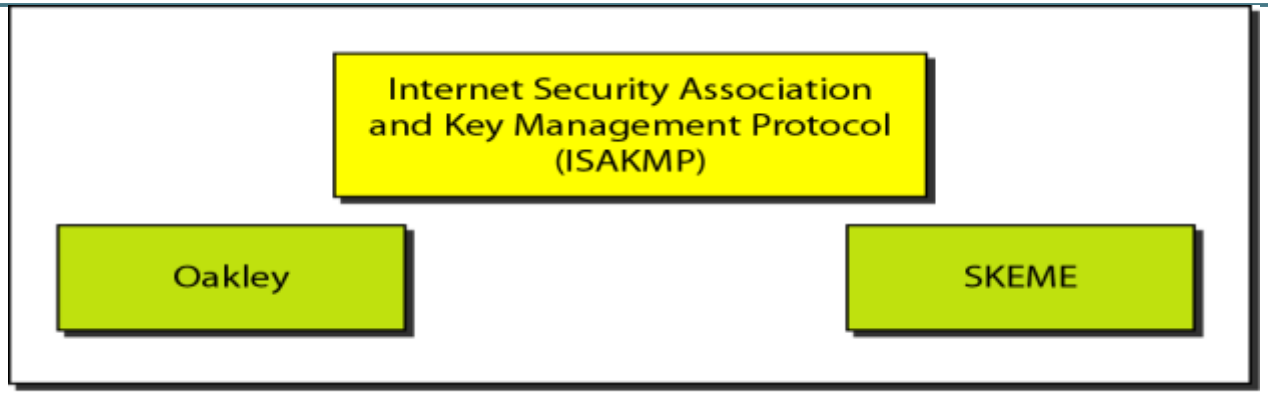


Internet Key Exchange (IKE)

IKE protocol creates SAs for IPSec in the SADB.

Uses 3 protocols: Oakley, SKEME and ISAKMP

IKE components



Internet Key Exchange (IKE)

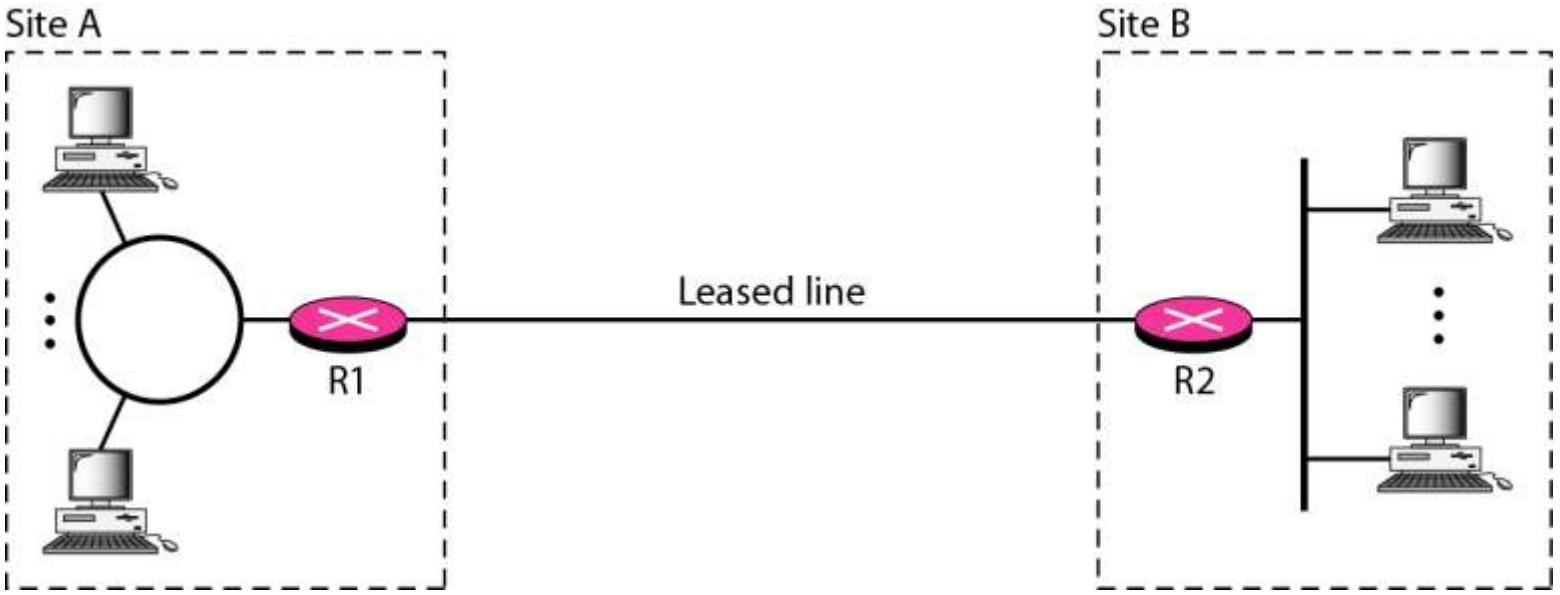
- Oakley Protocol
  - Developed by Hilarie Orman
  - Based on the Diffie-Hellman key-exchange method
- SKEME Protocol
  - Designed by Hugo Krawcyzk,
  - Uses public-key encryption for entity authentication in a key-exchange protocol.
- ISAKMP Protocol
  - Developed by the National Security Agency (NSA)
  - Allows the IKE exchanges to take place in standardized, formatted messages to create SAs

Virtual Private Network (VPN)

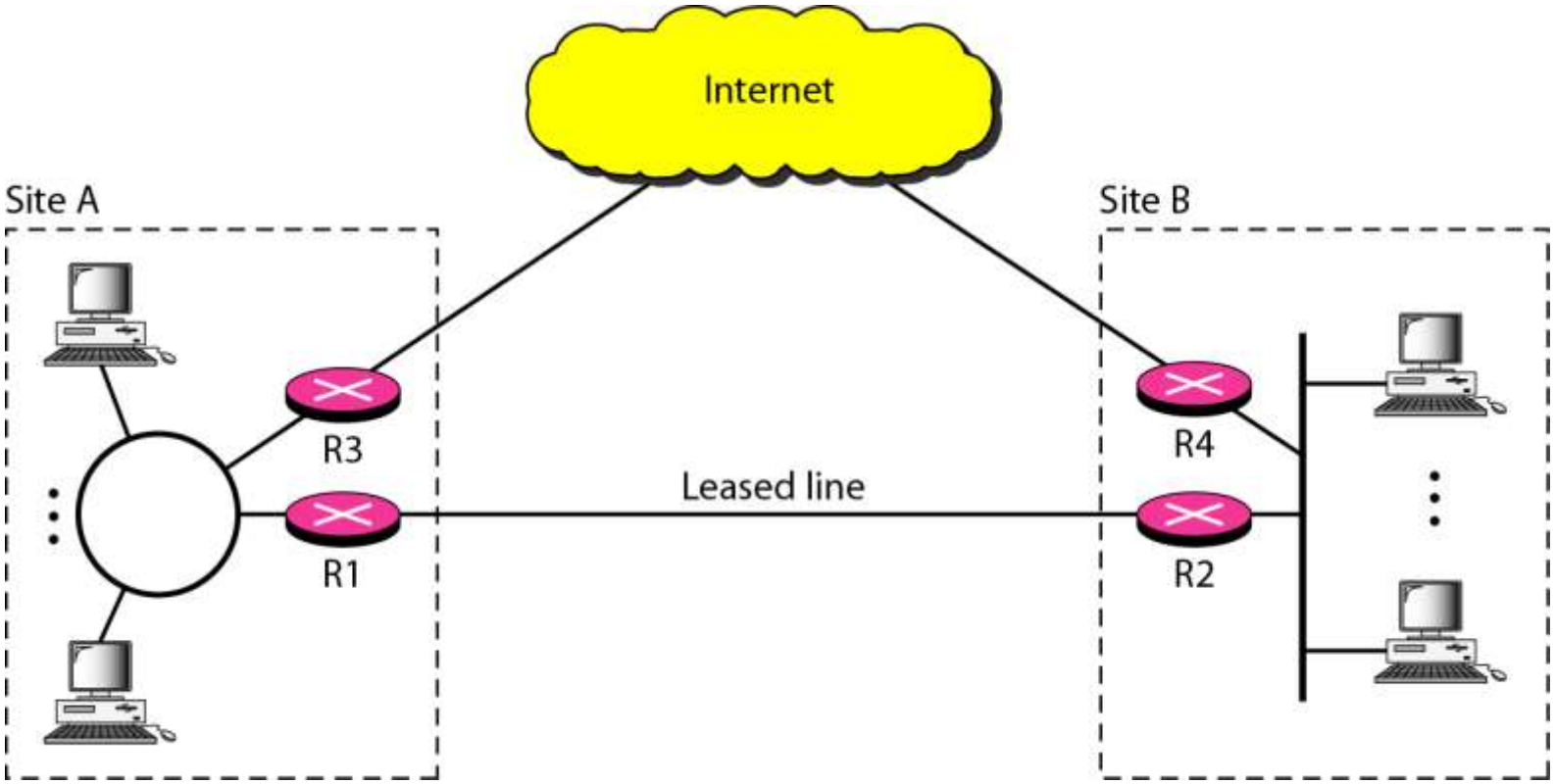
- **Goal:** allows access to shared resources and, at the same time, provides privacy
- Intranet / extranet
- Addressing:
  - Select addresses from Internet authorities.
  - Create addresses without register them at the Internet authorities
  - Use the reserved IP addresses

Prefix	*Range*	*Total*
10/8	10.0.0.0 to 10.255.255.255	2^{24}
172.16/12	172.16.0.0 to 172.31.255.255	2^{20}
192.168/16	192.168.0.0 to 192.168.255.255	2^{16}

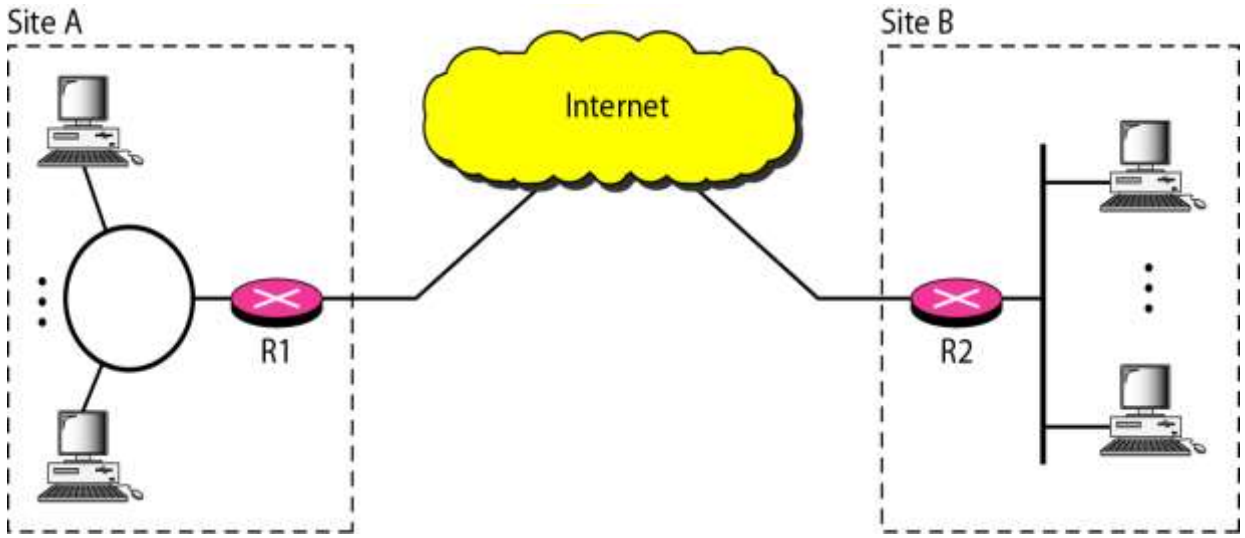
Private network



Hybrid network

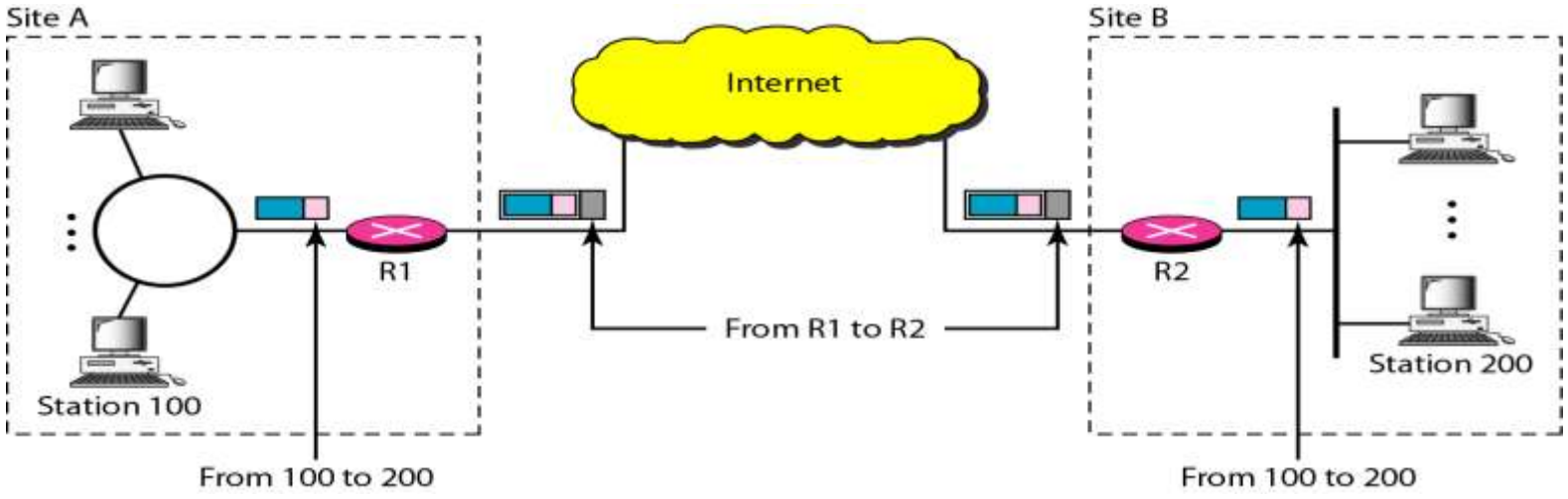


Virtual private network



- It is virtual because it does not use real private WANs
- The network is physically public but virtually private

Addressing in a VPN



- VPN technology uses IPSec in the tunnel mode to provide authentication, integrity, and privacy.
- Each IP datagram destined for private use in the organization is encapsulated in another datagram
- The public network (Internet) is responsible for carrying the packet from Rl to R2
- Outsiders cannot decipher the contents of the packet or the source and destination addresses

SSL/TLS

Two protocols are dominant today for providing security at the transport layer: the Secure Sockets Layer (SSL) Protocol and the Transport Layer Security (TLS) Protocol.

Topics discussed in this section:

SSL Services
Security Parameters
Sessions and Connections
Four Protocols
Transport Layer Security

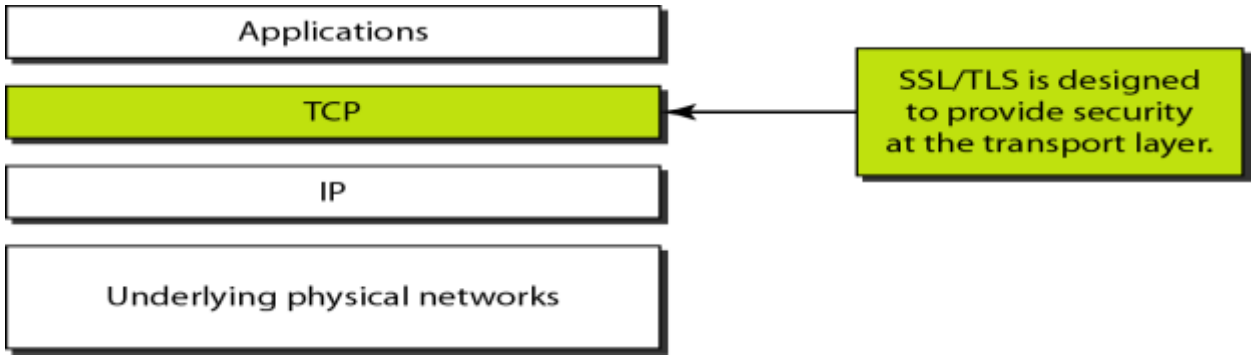
SSL/TLS

Goal: provide end-to-end security services for applications that use a reliable transport layer protocol such as TCP.

For example: when a customer shops online, the following security services are desired:

- The customer and the vendor need to authenticate each other (customer is not an imposter, not sharing credit card number) – (**entity authentication**)
- The customer and the vendor need to be sure that the contents of the message are not modified during transition (**message integrity**).
- The customer and the vendor need to be sure that an imposter does not intercept sensitive information such as a credit card number (**confidentiality**).

Location of SSL and TLS in the Internet model



Goal: designed to provide security and compression services to data generated from the application layer.

- Receives data from any application layer protocol,
- Usually, the protocol is HTTP
- The application data are *compressed* (optional), *signed*, and *encrypted*
- Then passed to a reliable transport layer protocol such as TCP
- Developed by Netscape in 1994
- Version 1, 2, and 3

Secure Socket Layer (SSL) Services

- *Fragmentation*

First, SSL divides the data into blocks of 214 bytes or less.

- *Compression*

Each fragment of data is compressed by using one of the lossless compression methods negotiated between the client and server (**optional**).

- *Message Integrity*

SSL uses a keyed-hash function to create a MAC.

- *Confidentiality*

The original data and the MAC are encrypted using symmetric key cryptography.

- *Framing*

A header is added to the encrypted payload. The ,payload is then passed to a reliable transport layer protocol.

Secure Socket Layer (SSL) Security Parameters

Cipher suites and Cryptographic secrets

- The combination of *key exchange*, *hash*, and *encryption algorithms* defines a cipher suite for each SSL session.
- General Format:

  SSL_key-exchange-algo_WITH_encryption-algo_hash-algo

- Example: SSL_DHE_RSA_WITH_DES_CBC_SHA
  - **Key exchange:** DHE_RSA
  - **Encryption algorithm:** DES_CBC
  - **Hash algorithm:** SHA
- Note: DH is fixed Diffie-Hellman, DHE is ephemeral Diffie-Hellman, and DH-anon is anonymous Diffie-Hellman.

SSL cipher suite list

Cipher Suite	*Key Exchange Algorithm*	*Encryption Algorithm*	*Hash Algorithm*
SSL_NULL_***WITH***_**NULL**_NULL	NULL	NULL	NULL
SSL_RSA_***WITH***_**NULL**_MD5	RSA	NULL	MD5
SSL_RSA_***WITH***_**NULL**_SHA	RSA	NULL	SHA
SSL_RSA_***WITH***_**RC4_128**_MD5	RSA	RC4_128	MD5
SSL_RSA_***WITH***_**RC4_128**_SHA	RSA	RC4_128	SHA
SSL_RSA_***WITH***_**IDEA_CBC**_SHA	RSA	IDEA_CBC	SHA
SSL_RSA_***WITH***_**DES_CBC**_SHA	RSA	DES_CBC	SHA
SSL_RSA_***WITH***_**3DES_EDE_CBC**_SHA	RSA	3DES_EDE_CBC	SHA
SSL_DH_anon_***WITH***_**RC4_128**_MD5	DH_anon	RC4_128	MD5
SSL_DH_anon_***WITH***_**DES_CBC**_SHA	DH_anon	DES_CBC	SHA
SSL_DH_anon_***WITH***_**3DES_EDE**_CBC_SHA	DH_anon	3DES_EDE_CBC	SHA

SSL cipher suite list (continued)

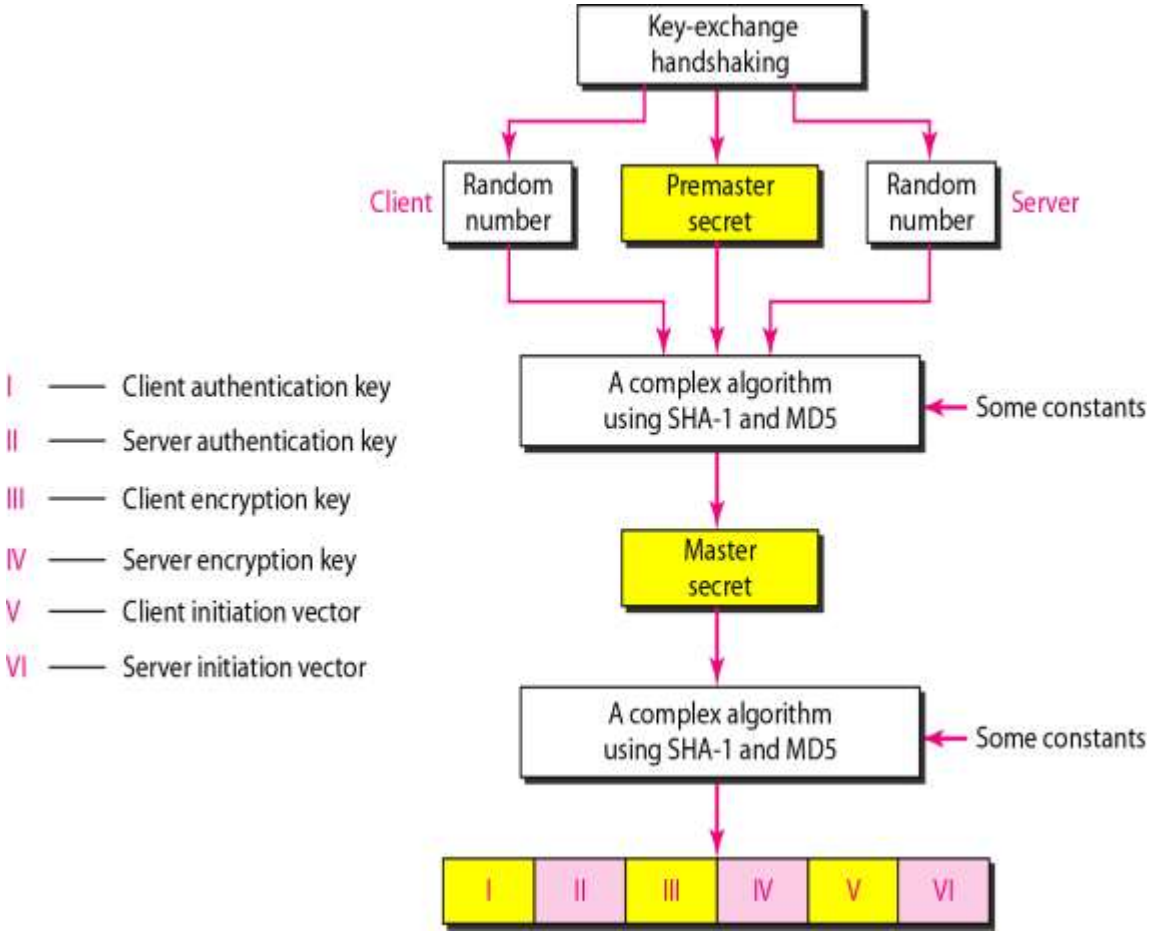
Cipher Suite	*Key Exchange Algorithm*	*Encryption Algorithm*	*Hash Algorithm*
SSL_DHE_RSA_***WITH***_DES_CBC_SHA	DHE_RSA	DES_CBC	SHA
SSL_DHE_RSA_***WITH***_3DES_EDE_CBC_SHA	DHE_RSA	3DES_EDE_CBC	SHA
SSL_DHE_DSS_***WITH***_DES_CBC_SHA	DHE_DSS	DES_CBC	SHA
SSL_DHE_DSS_***WITH***_3DES_EDE_CBC_SHA	DHE_DSS	3DES_EDE_CBC	SHA
SSL_DH_RSA_***WITH***_DES_CBC_SHA	DH_RSA	DES_CBC	SHA
SSL_DH_RSA_***WITH***_3DES_EDE_CBC_SHA	DH_RSA	3DES_EDE_CBC	SHA
SSL_DH_DSS_***WITH***_DES_CBC_SHA	DH_DSS	DES_CBC	SHA
SSL_DH_DSS_***WITH***_3DES_EDE_CBC_SHA	DH_DSS	3DES_EDE_CBC	SHA
SSL_FORTEZZA_DMS_***WITH***_NULL_SHA	FORTEZZA_DMS	NULL	SHA
SSL_FORTEZZA_DMS_***WITH***_FORTEZZA_CBC_SHA	FORTEZZA_DMS	FORTEZZA_CBC	SHA
SSL_FORTEZZA_DMS_***WITH***_RC4_128_SHA	FORTEZZA_DMS	RC4_128	SHA

Secure Socket Layer (SSL) Security Parameters

Cipher suites and **Cryptographic secrets**

- To achieve message integrity and confidentiality, SSL needs six cryptographic secrets: *four keys*, and *two IVs*.
- The client and the server need
  - One key for message authentication,
  - One key for encryption,
  - One IV for block encryption.
- Each direction requires different keys (prevents that an attack does not affect the other direction)
- These parameters are generated by using a negotiation protocol

Creation of cryptographic secrets in SSL



1- client and server exchange two random numbers

2- client and server exchange one premaster secret by using one of the key exchange algorithms

3- A 48-byte master secret is created from the premaster secret

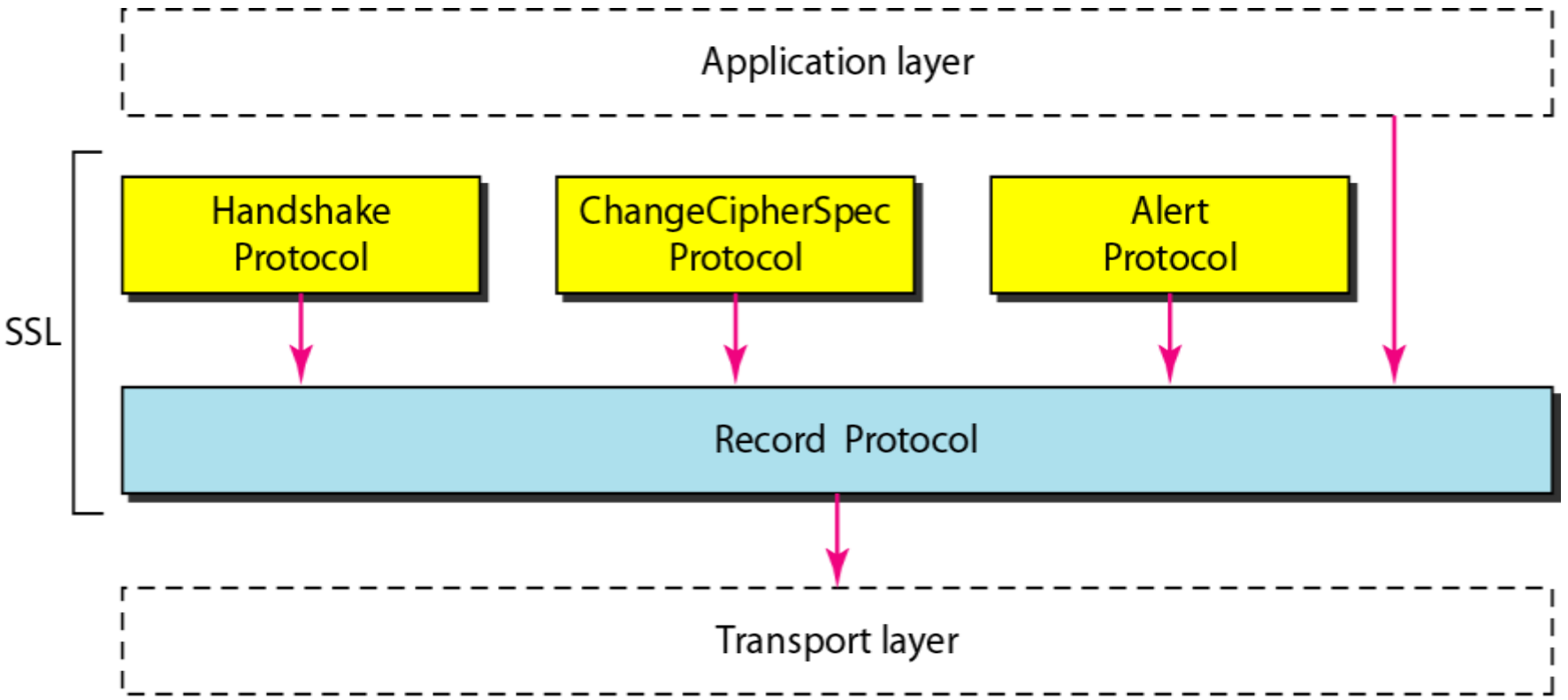
4. The master secret is used to create variable-length secrets

Sessions and Connections

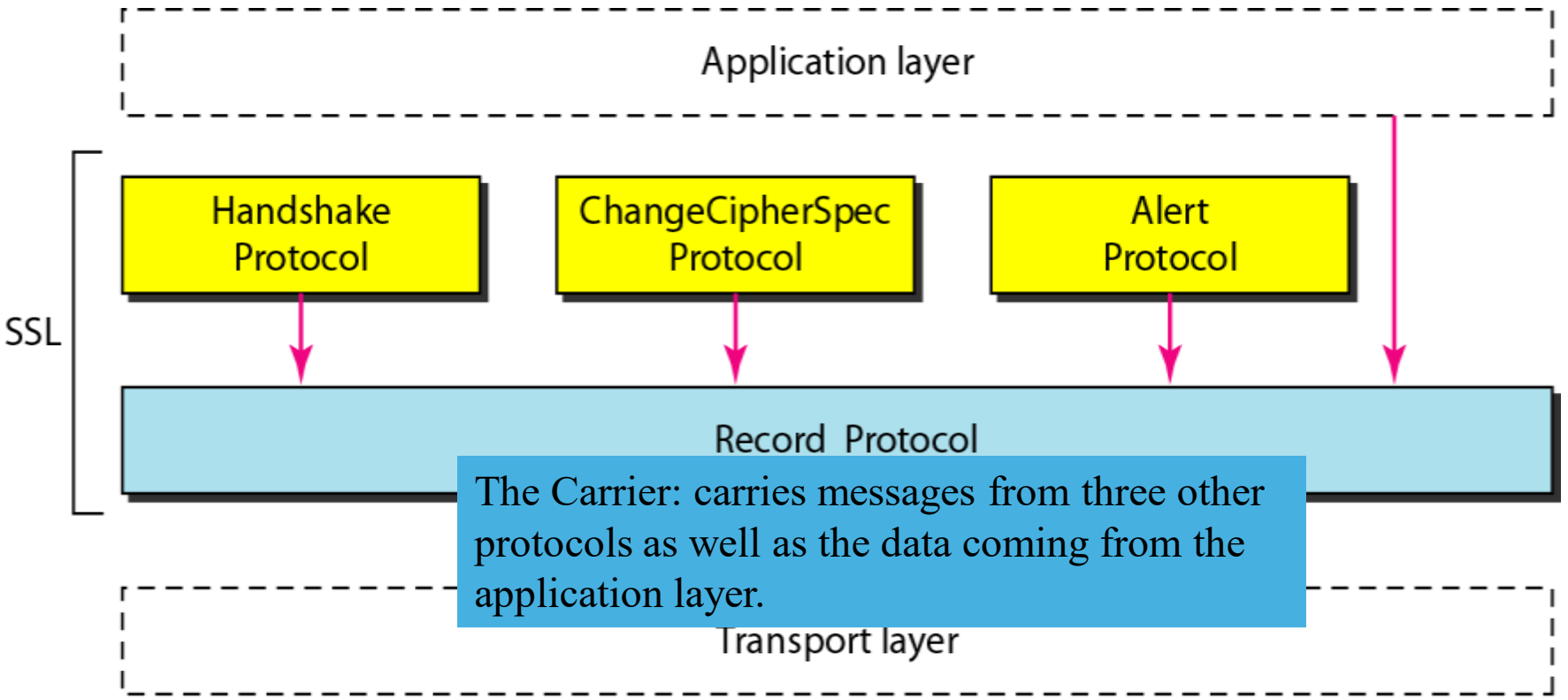
- IP is a connectionless protocol; TCP is a connection-oriented protocol
- Association in IPSec transforms the connectionless IP to a connection-oriented secured protocol

- **Session**: association that can last for a long time.
  - Some of the security parameters are created during the session establishment and are in effect until the session is terminated (e.g. cipher suite and master key).
- **Connection**: established and broken several times during a session.
  - Some of the security parameters must be recreated (or occasionally resumed) for each connection (e.g. six secrets).
- SSL defines four protocols in two layers.

Four SSL protocols

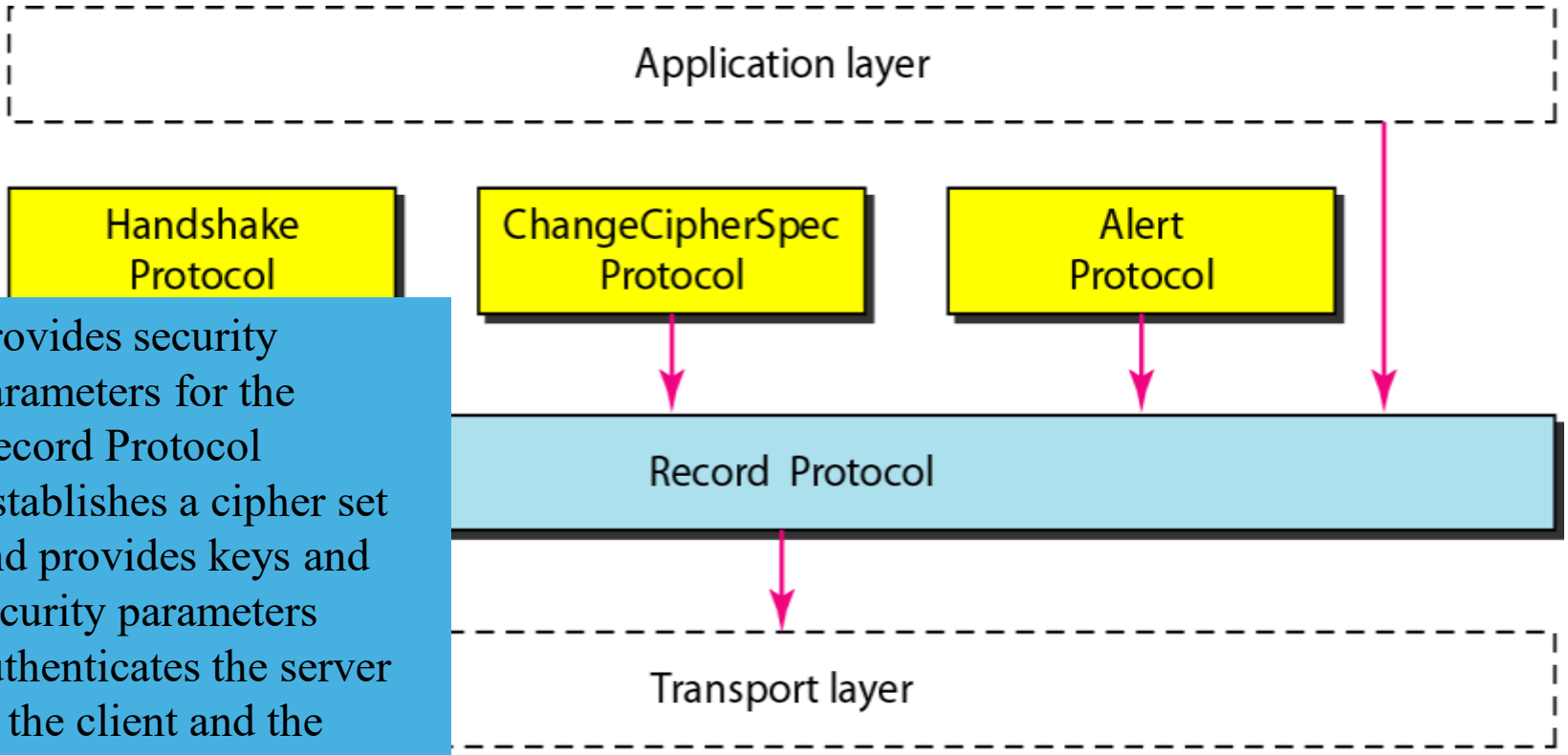


Four SSL protocols



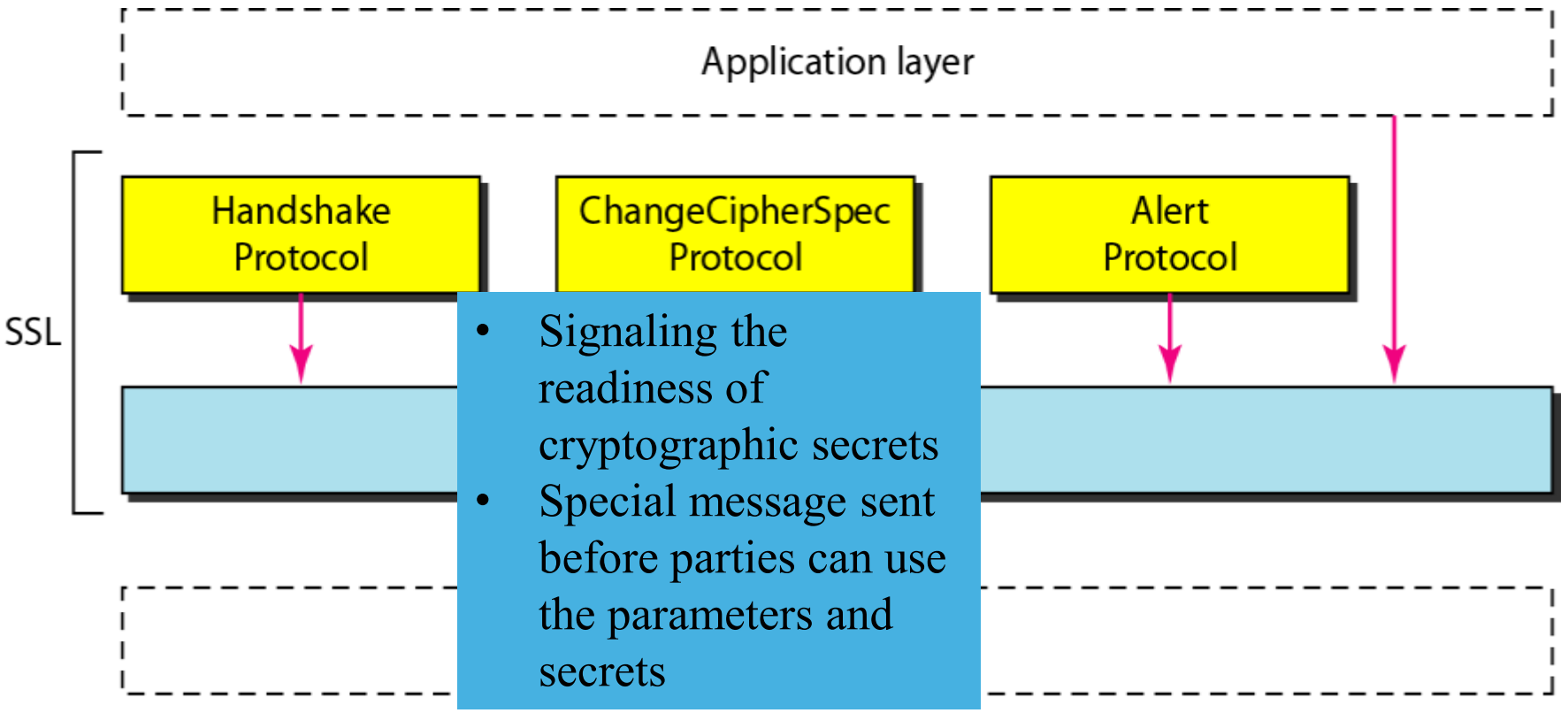
The Carrier: carries messages from three other protocols as well as the data coming from the application layer.

Four SSL protocols

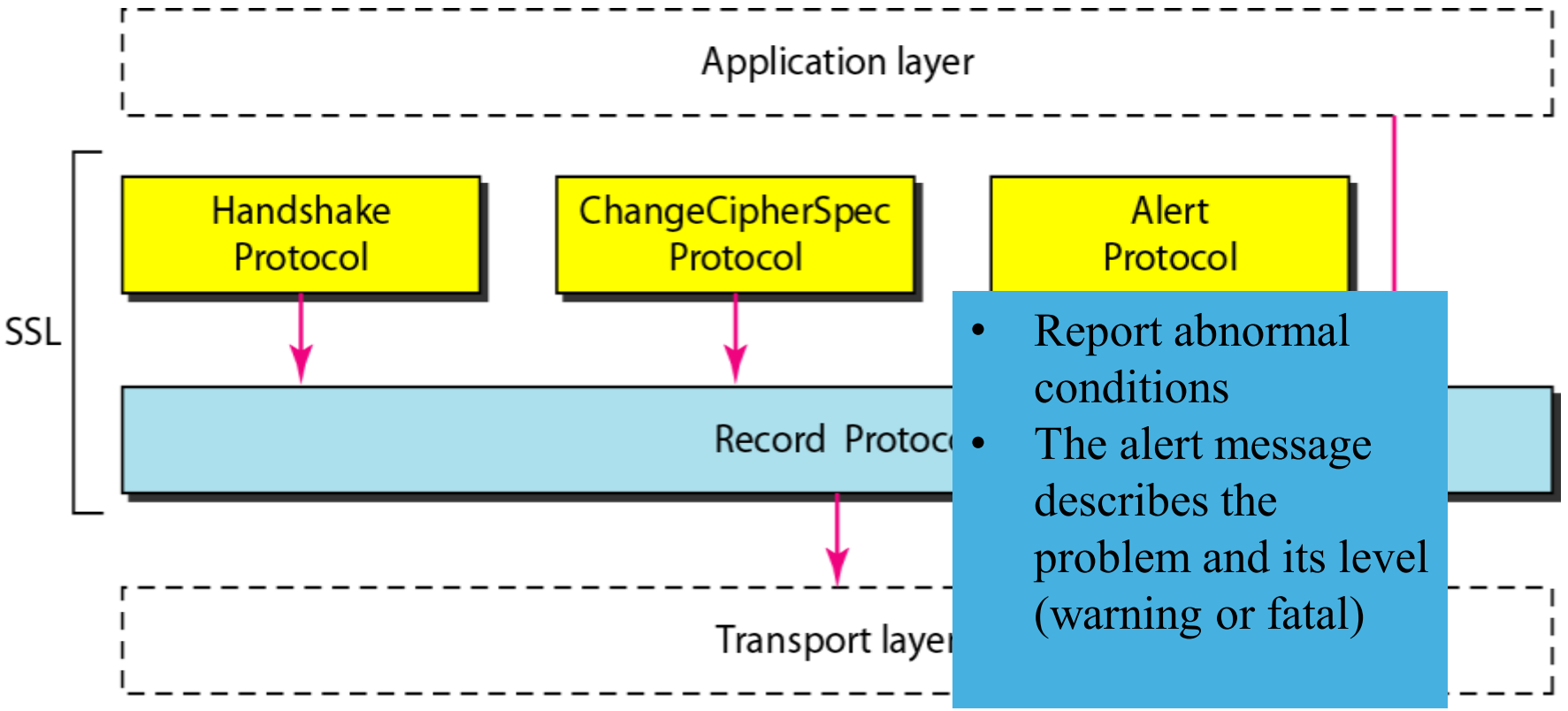


- Provides security parameters for the Record Protocol
- Establishes a cipher set and provides keys and security parameters
- authenticates the server to the client and the client to the server, if needed

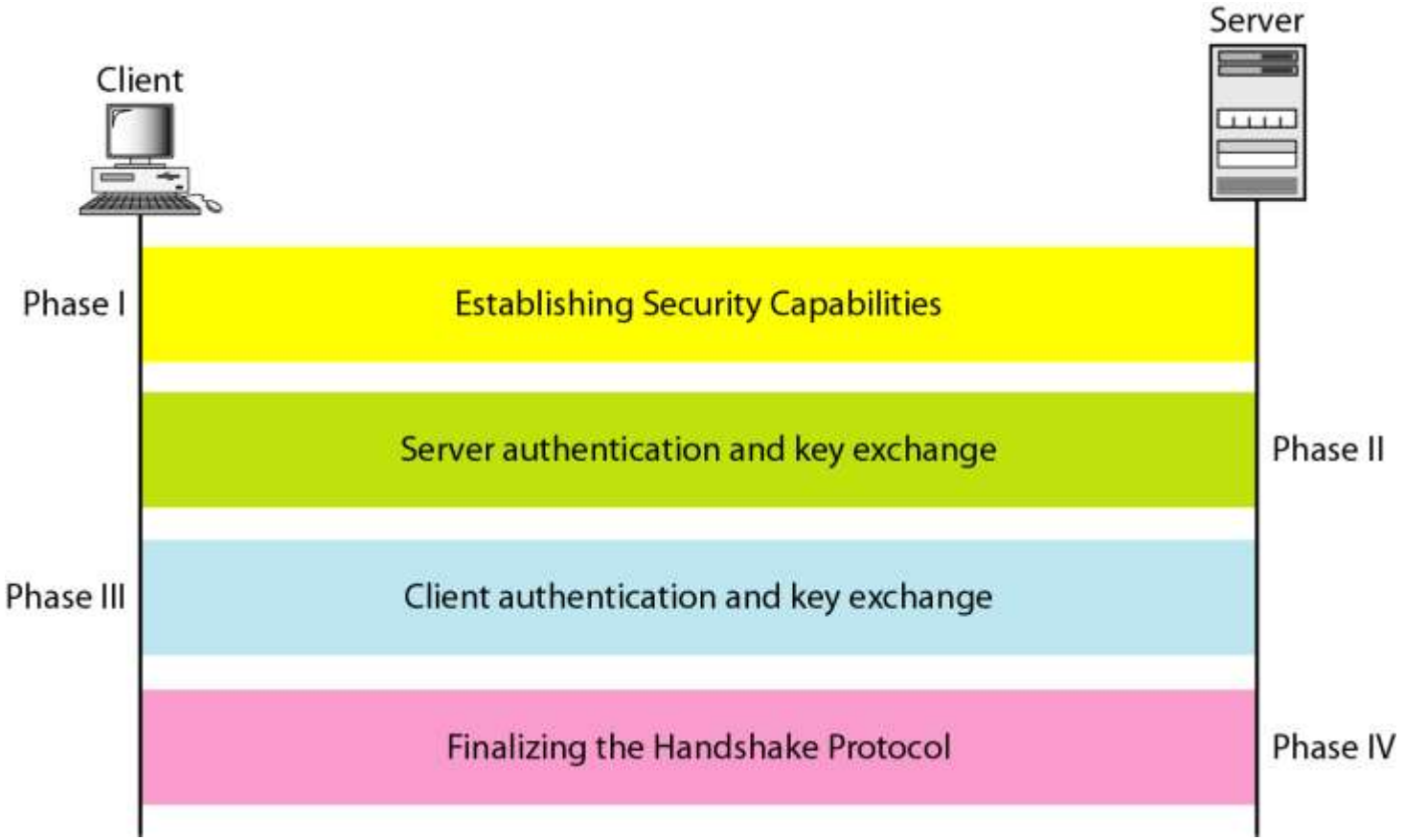
Four SSL protocols



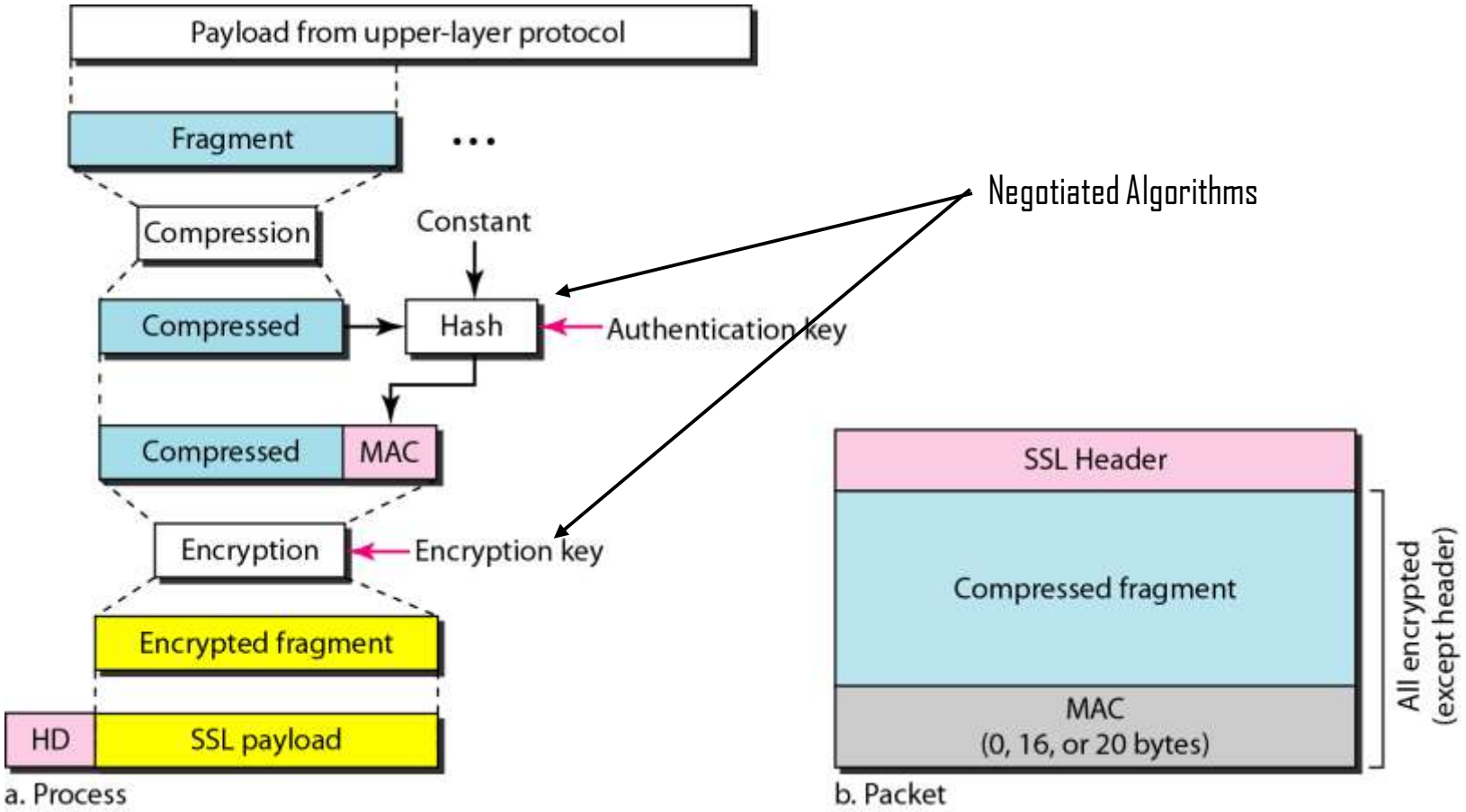
Four SSL protocols



Handshake Protocol



Processing done by the Record Protocol



Transport Layer Security (TLS)

- IETF standard version of SSL.
- Differences:
  - SSLv3.0 discussed in this section is compatible with TLSv1.0.
  - TLS cipher suite does not support Fortezza.
  - TLS uses a pseudorandom function (PRF) to create the master key and the key materials for the cryptographic secrets
  - TLS deletes some alert messages and adds some new ones.
  - TLS changes details of some messages in Handshake protocol.
  - TLS uses HMAC instead of MAC in the Record protocol.

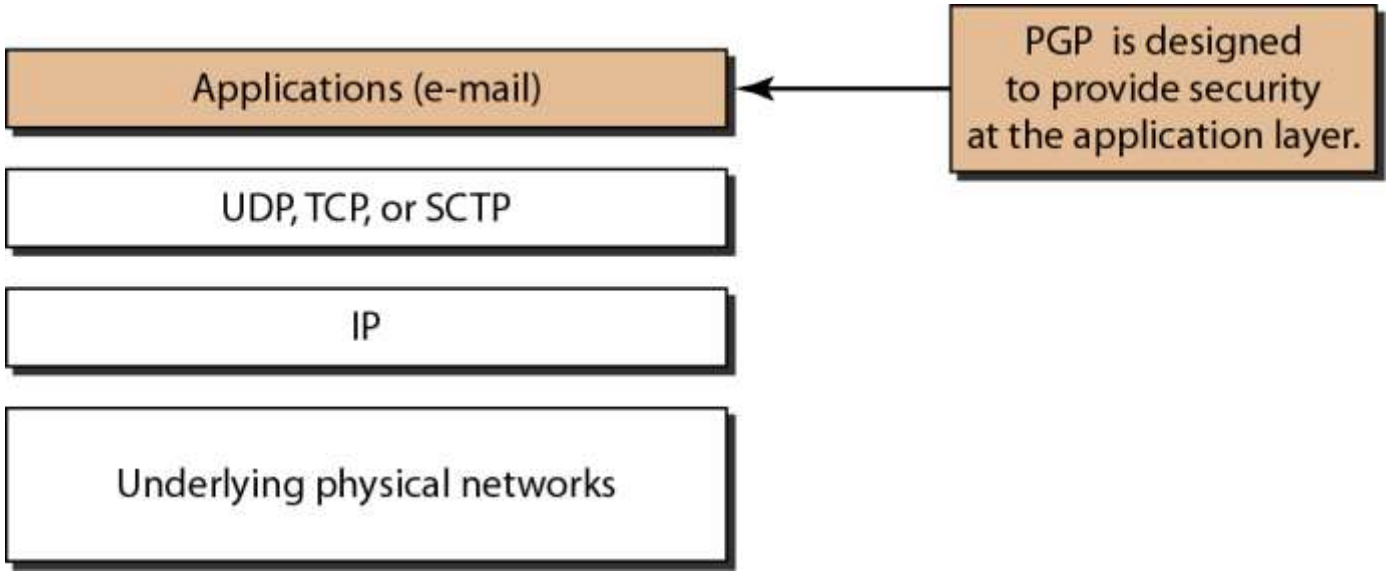
PGP

One of the protocols to provide security at the application layer is Pretty Good Privacy (PGP). PGP is designed to create authenticated and confidential e-mails.

Topics discussed in this section:

Security Parameters
Services
A Scenario
PGP Algorithms
Key Rings
PGP Certificates

Position of PGP in the TCP/IP protocol suite



Pretty Good Privacy (PGP)

- Sending an e-mail is a one-time activity (*Alice sends a message to Bob; sometime later, Bob reads the message and may or may not send a reply*)
- In e-mail, there is ***no session*** and ***no handshake***. To negotiate the algorithms for encryption and authentication.

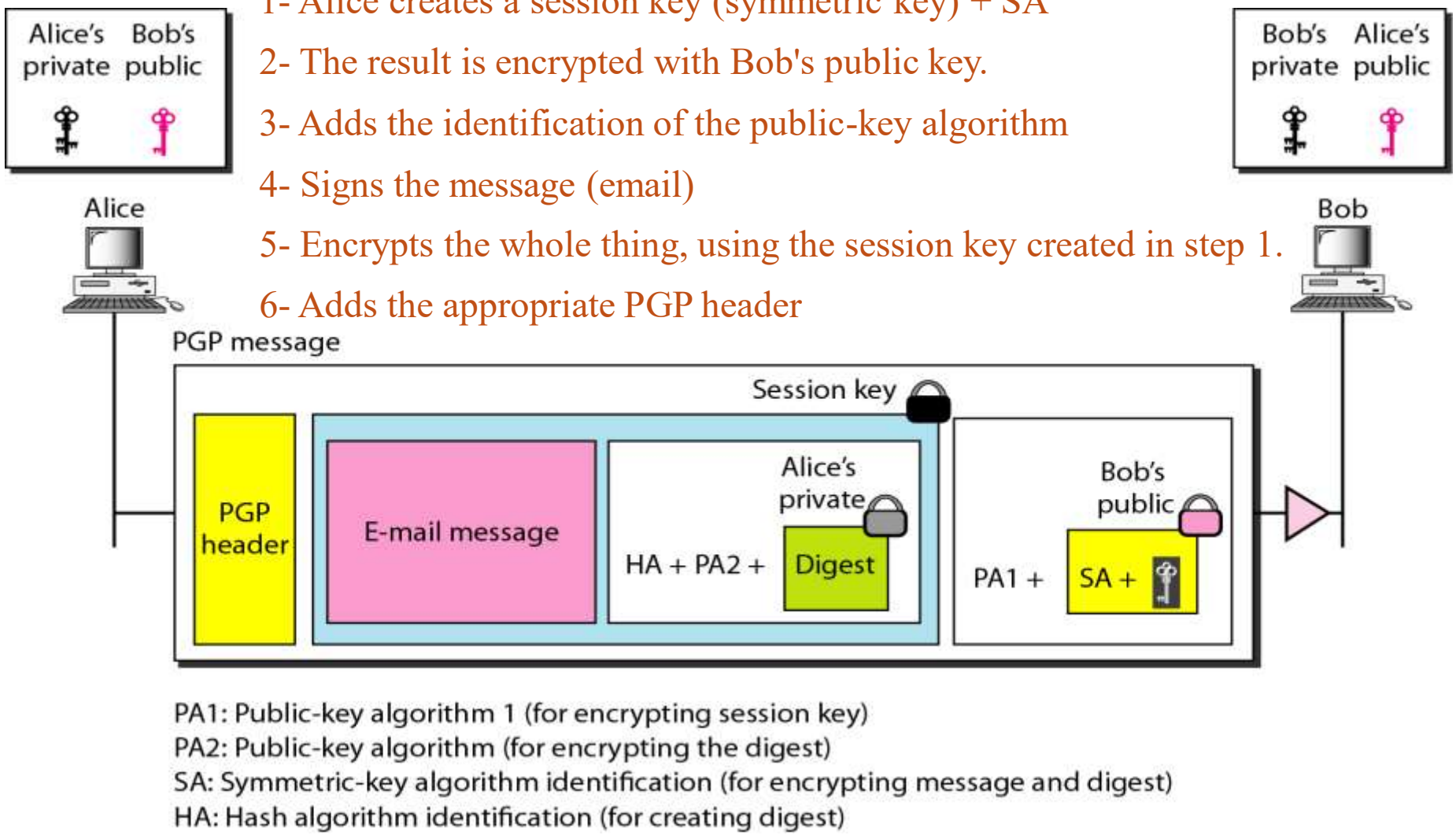
=> *security of a unidirectional message*

Solution: ***the sender of the message includes the identifiers of the algorithms used in the message as well as the values of the keys.***

Pretty Good Privacy (PGP) / Services

- Plaintext
- Message Authentication
  - Signature using Public/Private Keys
- Compression
- Confidentiality
  - One Time Session Key
  - The key is sent within the message, encrypted with the receiver public key.
- Code Conversion
  - In email only ASCII is used.
  - PGP uses Radix 64 conversion
- Segmentation

*A scenario in which an e-mail message is **authenticated** and **encrypted***



PA1: Public-key algorithm 1 (for encrypting session key)
PA2: Public-key algorithm (for encrypting the digest)
SA: Symmetric-key algorithm identification (for encrypting message and digest)
HA: Hash algorithm identification (for creating digest)

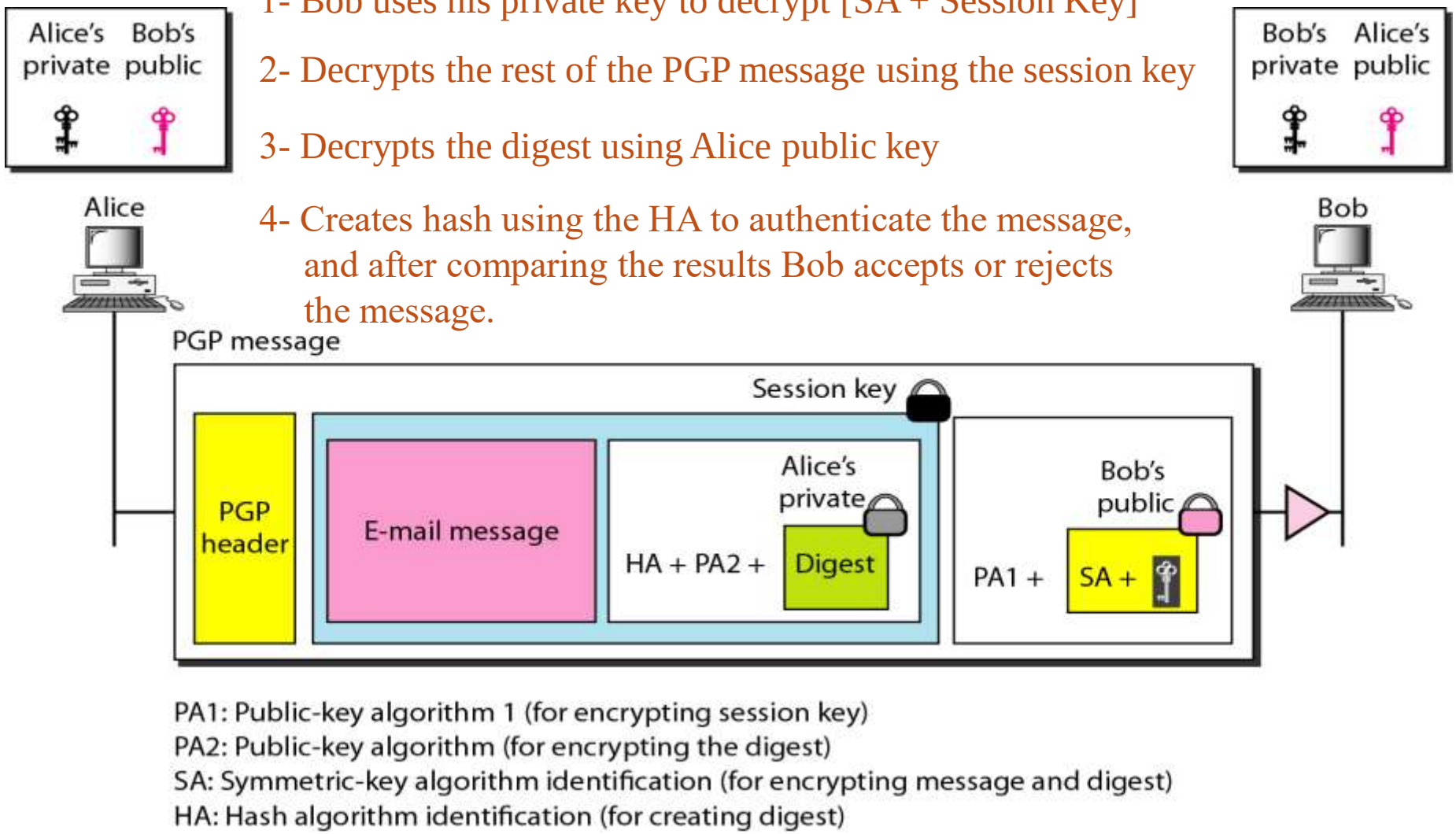
*A scenario in which an e-mail message is **authenticated** and **encrypted***

1- Bob uses his private key to decrypt [SA + Session Key]

2- Decrypts the rest of the PGP message using the session key

3- Decrypts the digest using Alice public key

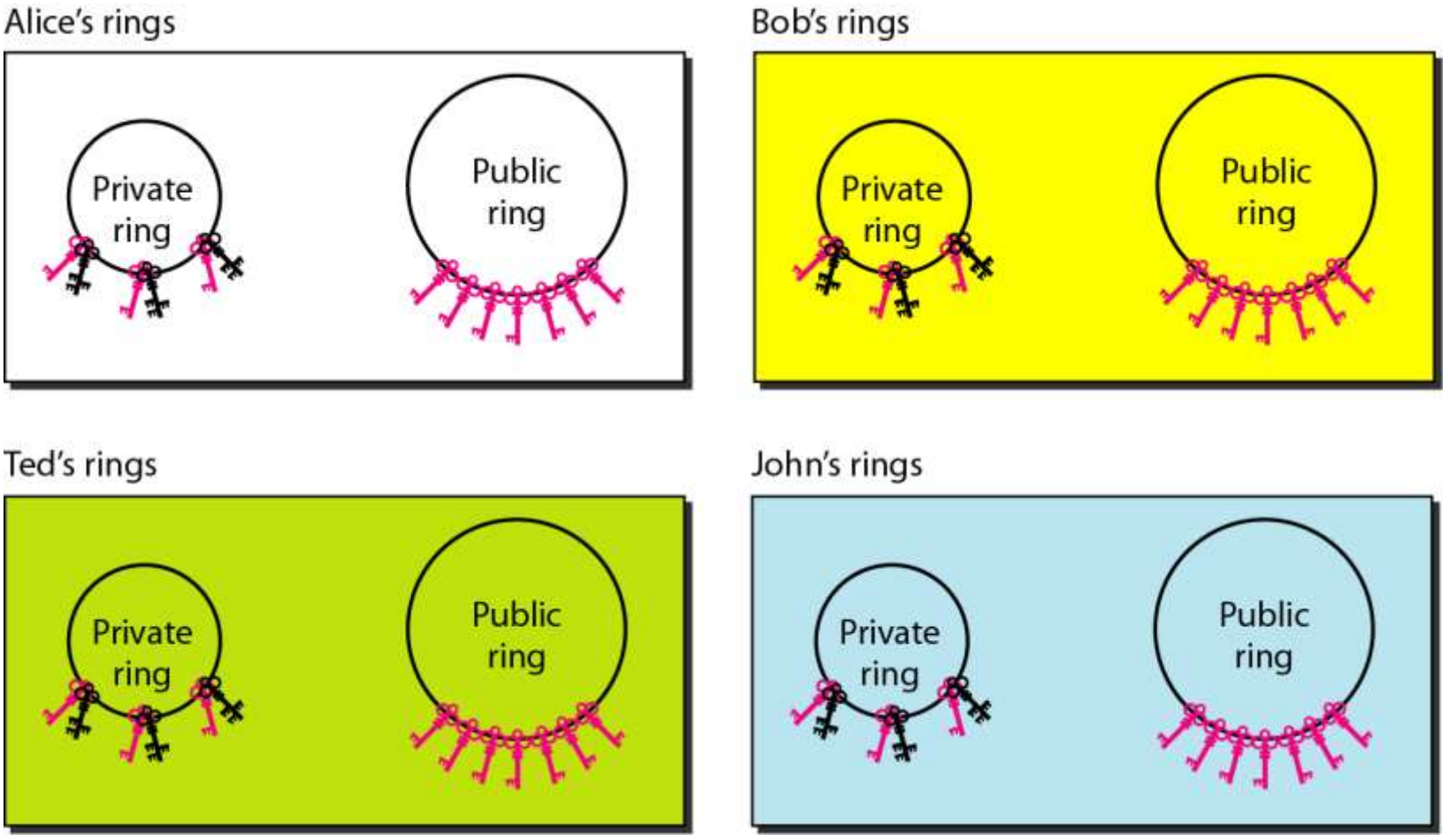
4- Creates hash using the HA to authenticate the message, and after comparing the results Bob accepts or rejects the message.



PGP Algorithms

Algorithm	*ID*	*Description*
Public key	1	RSA (encryption or signing)
	2	RSA (for encryption only)
	3	RSA (for signing only)
	17	DSS (for signing)
Hash algorithm	1	MD5
	2	SHA-1
	3	RIPE-MD
Encryption	0	No encryption
	1	IDEA
	2	Triple DES
	9	AES

Rings


Alice's rings
Private ring
Public ring
Bob's rings
Private ring
Public ring
Ted's rings
Private ring
Public ring
John's rings
Private ring
Public ring

FIREWALLS

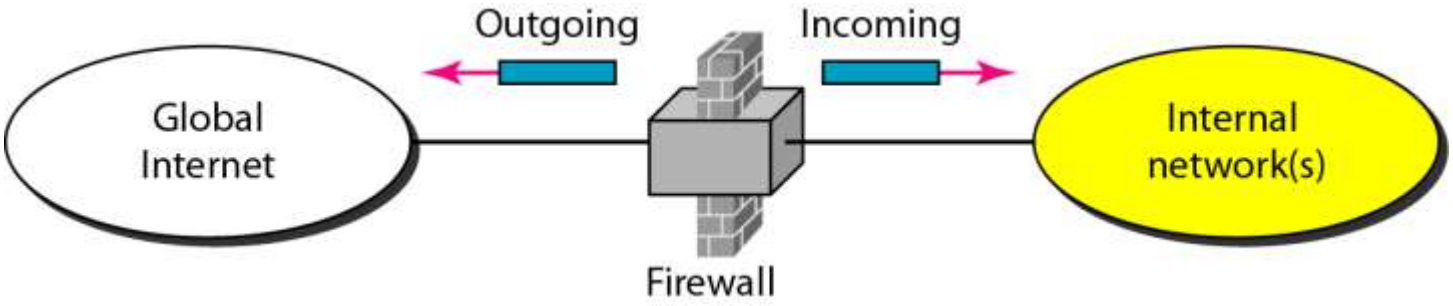
All previous security measures cannot prevent Eve from sending a harmful message to a system. To control access to a system, we need firewalls.

A firewall is a device installed between the internal network of an organization and the rest of the Internet. It is designed to forward some packets and filter (not forward) others.

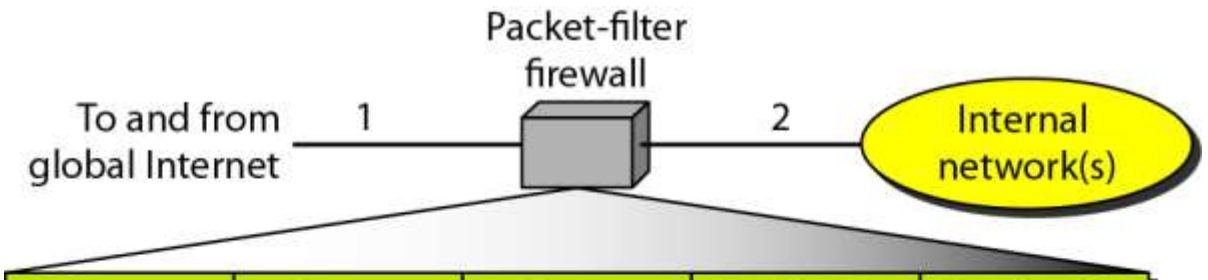
Topics discussed in this section:

Packet-Filter Firewall
Proxy Firewall

Firewall


Outgoing
Incoming
Global
Internet
Internal
network(s)
Firewall

Packet-filter firewall

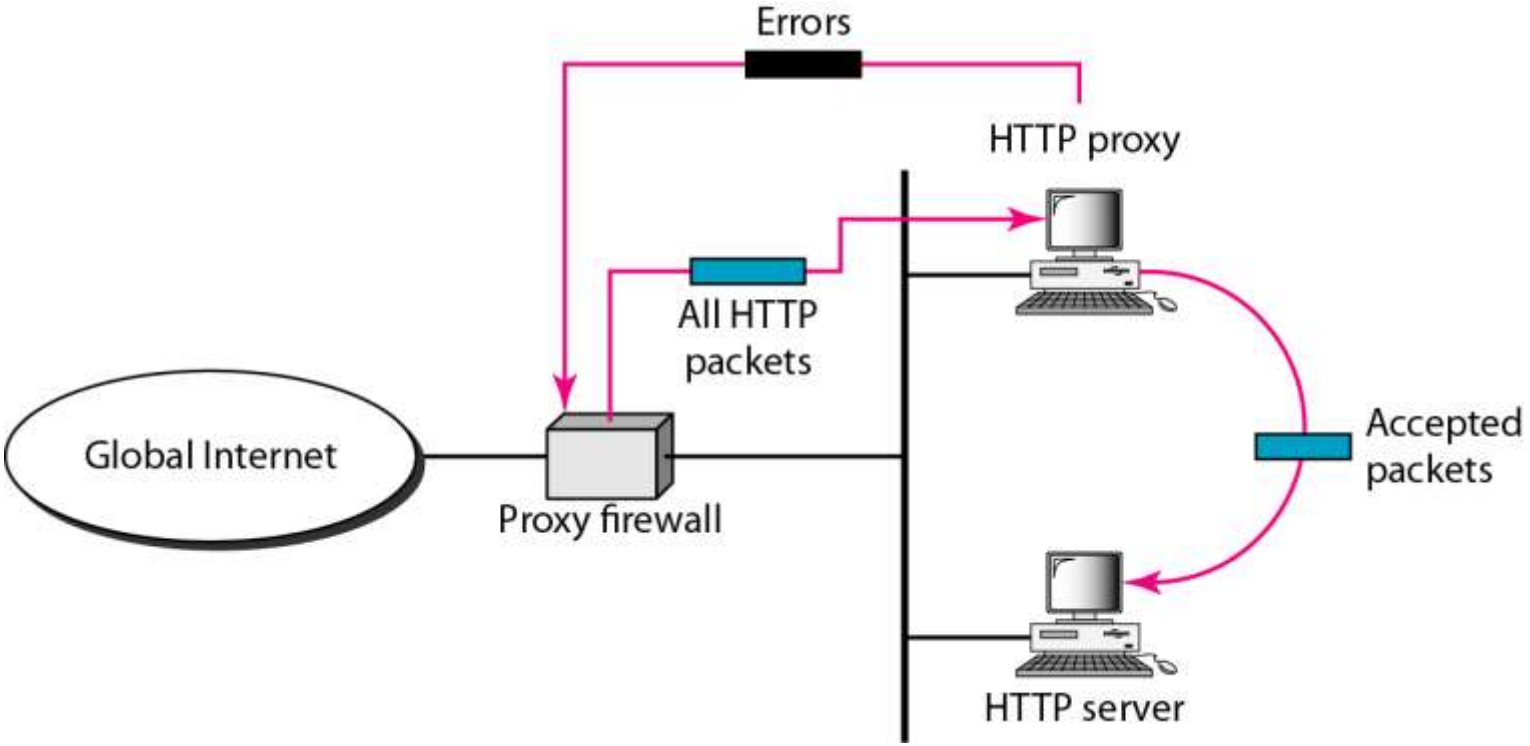


Interface	Source IP	Source port	Destination IP	Destination port
1	131.34.0.0	*	*	*
1	*	*	*	23
1	*	*	194.78.20.8	*
2	*	80	*	*

Packet-filter Firewall

- A packet-filter firewall filters at the network or transport layer.
- The packet-filter firewall is based on the information available in the network layer and transport layer headers
- Sometimes we need to filter a message based on the information available in the message itself (at the application layer) => ***proxy firewall***

Proxy firewall



Policy: Only those Internet users who have previously established business relations with the company can have access; access to other users must be blocked.