UNIT IV
NETWORK SECURITY

UNIT IV - NETWORK SECURITY 9

Authentication Applications: Kerberos – X.509 Authentication Service – Electronic Mail Security – PGP – S/MIME – IP Security – ISAKMP.

Text Books:

1. William Stallings, "Cryptography and Network Security - Principles and Practices", Prentice Hall of India, Third Edition, 2003.
2. Behrouz A. Forouzan, "Cryptography and Network Security", Tata McGraw Hall, Second Edition,2011.

Reference Books:

1. AtulKahate, "Cryptography and Network Security", Tata McGraw-Hill, 2003.
2. BruceSchneier, "Applied Cryptography", John Wiley and Sons Inc, 2001.
3. Charles P. Pfleeger, Shari Lawrence Pfleeger, "Security in Computing", Third Edition, Pearson Education, 2003.
4. Wenbo Mao, "Modern Cryptography: Theory and Practice", Pearson Education, Second edition, 2007.

Web References:

1. http://nptel.ac.in/courses/106105031/
2. http://www.cse.iitk.ac.in/users/braman/cs425/slides/security-overview.pdf

Authentication Applications-KERBEROS

Ref:Cryptography and Network Security

Atul Kahate

Course Outcome

- Explain various authentication algorithms for network security

Learning Outcome

- Understand authentication protocol

Specific Outcome

- Understand authentication application kerberos

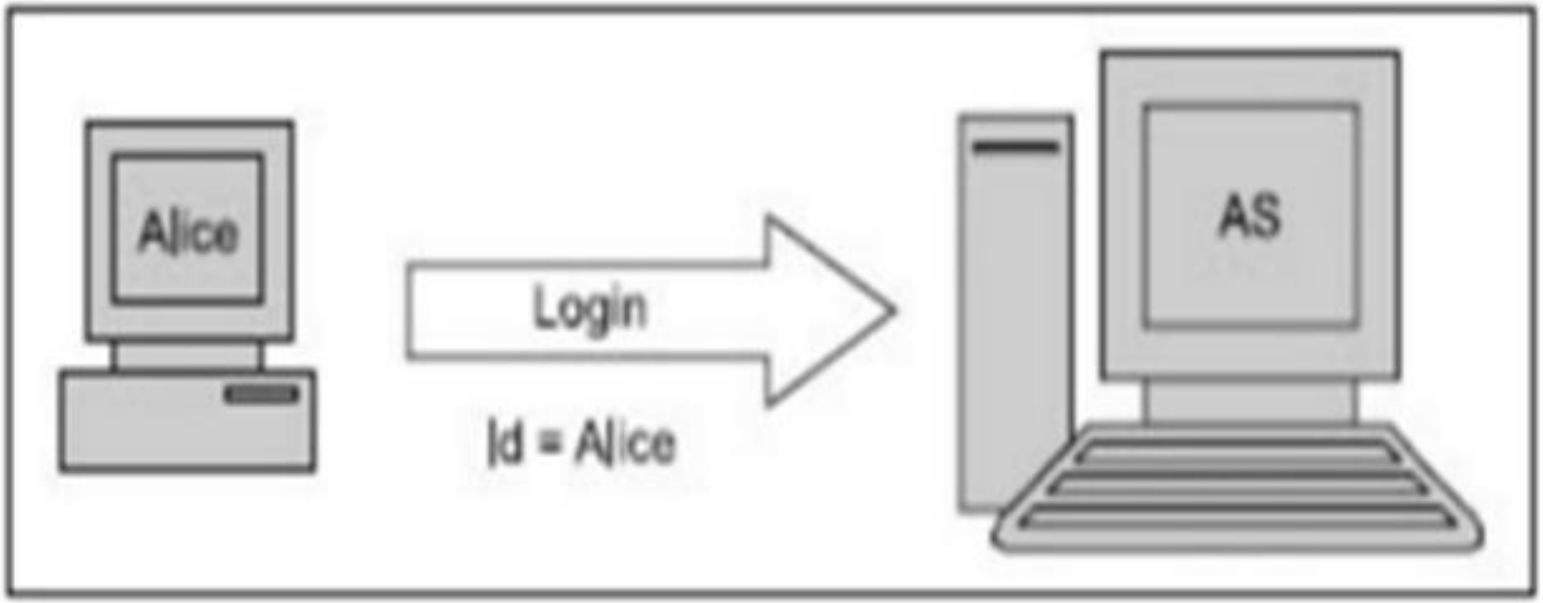
KERBEROS

- Many real time systems use an authentication protocol called as Kerberos.
- This is based on Needham-Schroeder protocol.
- Designed at MIT to allow the workstations to use the network resources in a safe manner.
- Kerberos signifies a multi headed dog in the greek mythology(apparently used to keep outsiders away).
- two versions in use: 4 & 5

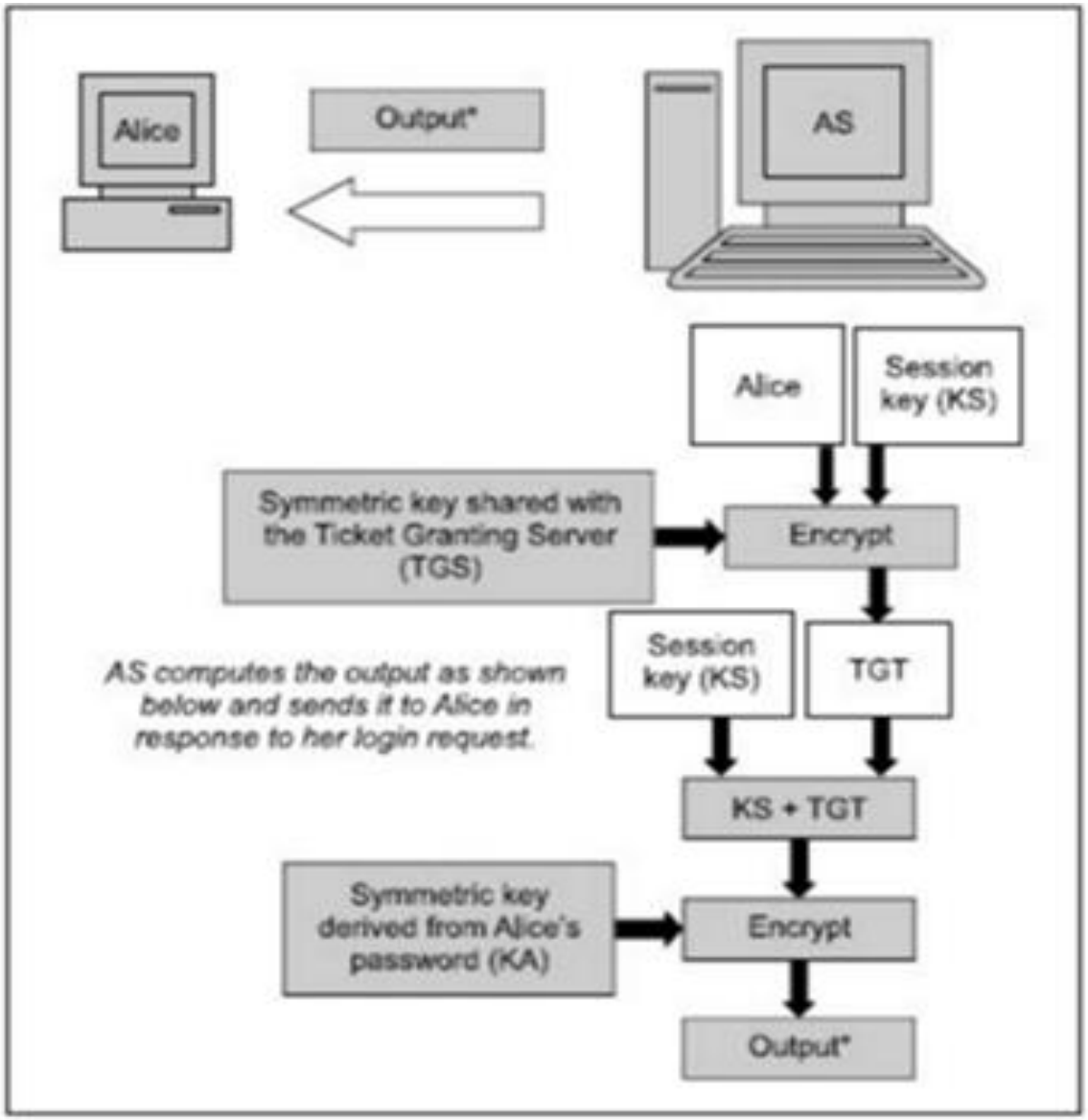
Working

- There are four parties involved in the kerberos protocol
  - Alice:The client Workstation
  - Authentication Servers(AS):Verifies (authenticates)the user during login
  - Ticket Granting Server(TGS):Issues tickets to certify proof of identity.
  - Bob:The server offering services such as network printing,file sharing or an application program.

- There are three primary steps in the Kerberos protocol.
  - **Step 1:Login**

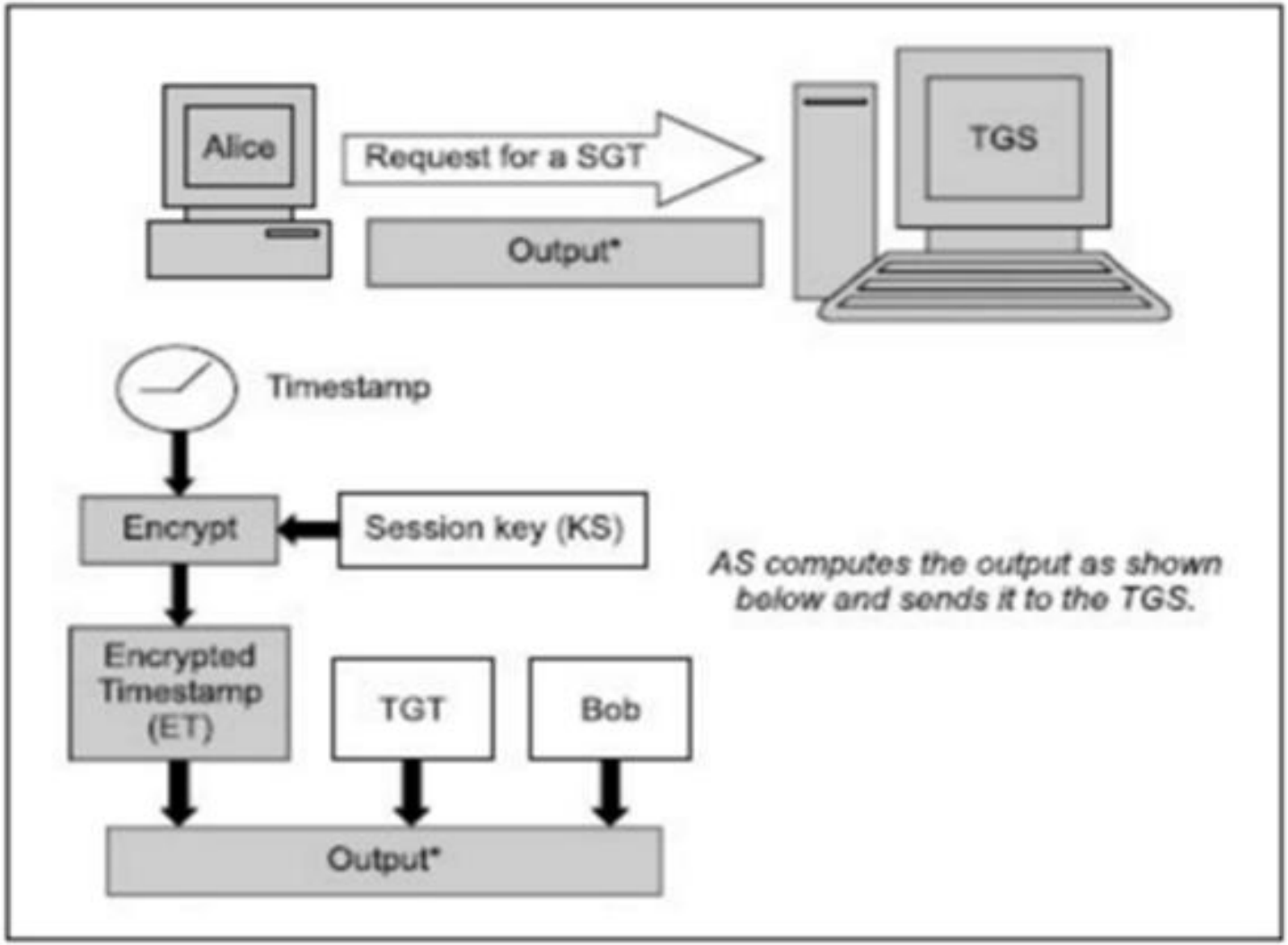


Alice sends a login request to AS

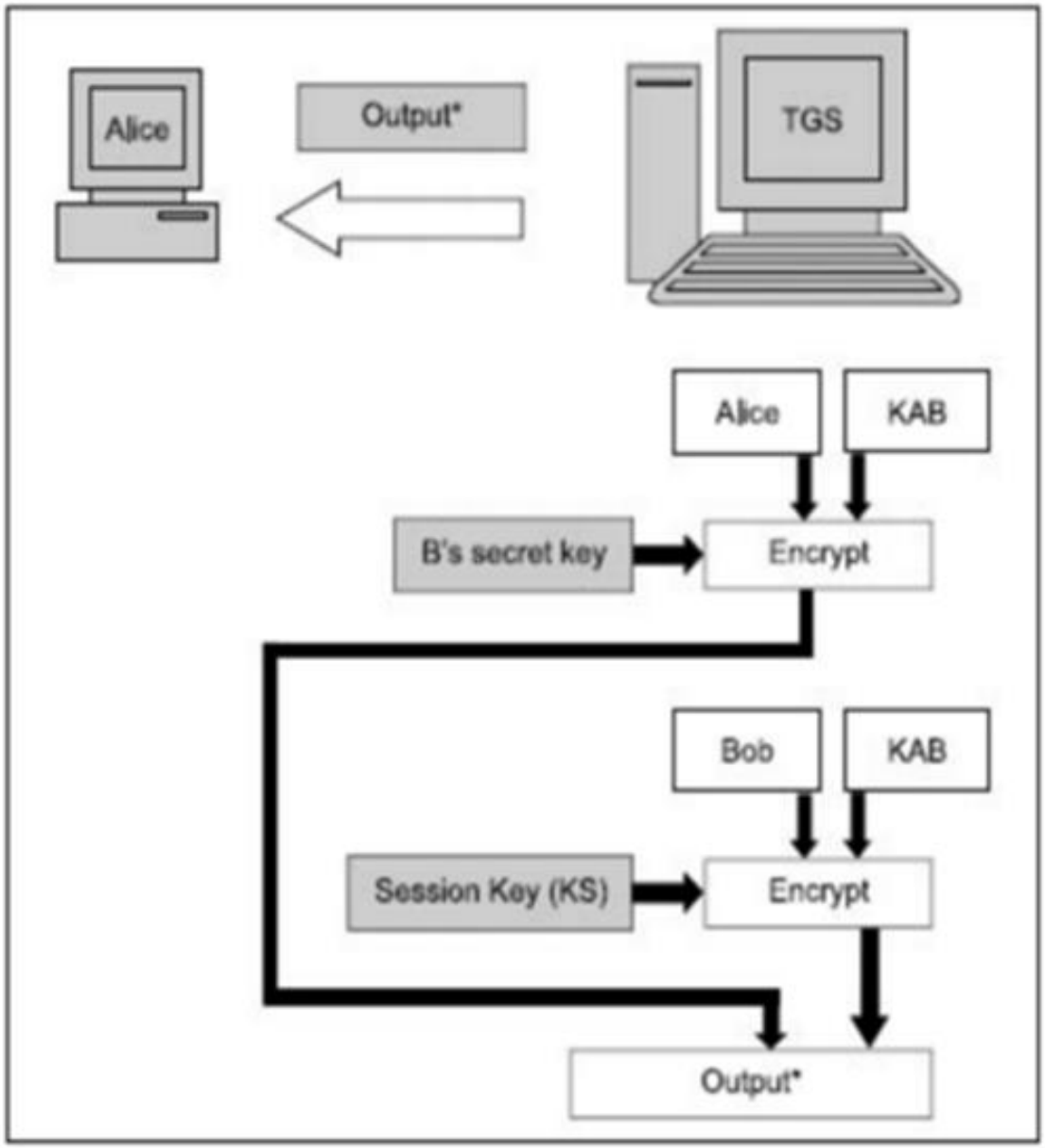


AS sends back encrypted session key and TGT to Alice

Step 2:Obtaining a service granting ticket(SGT)

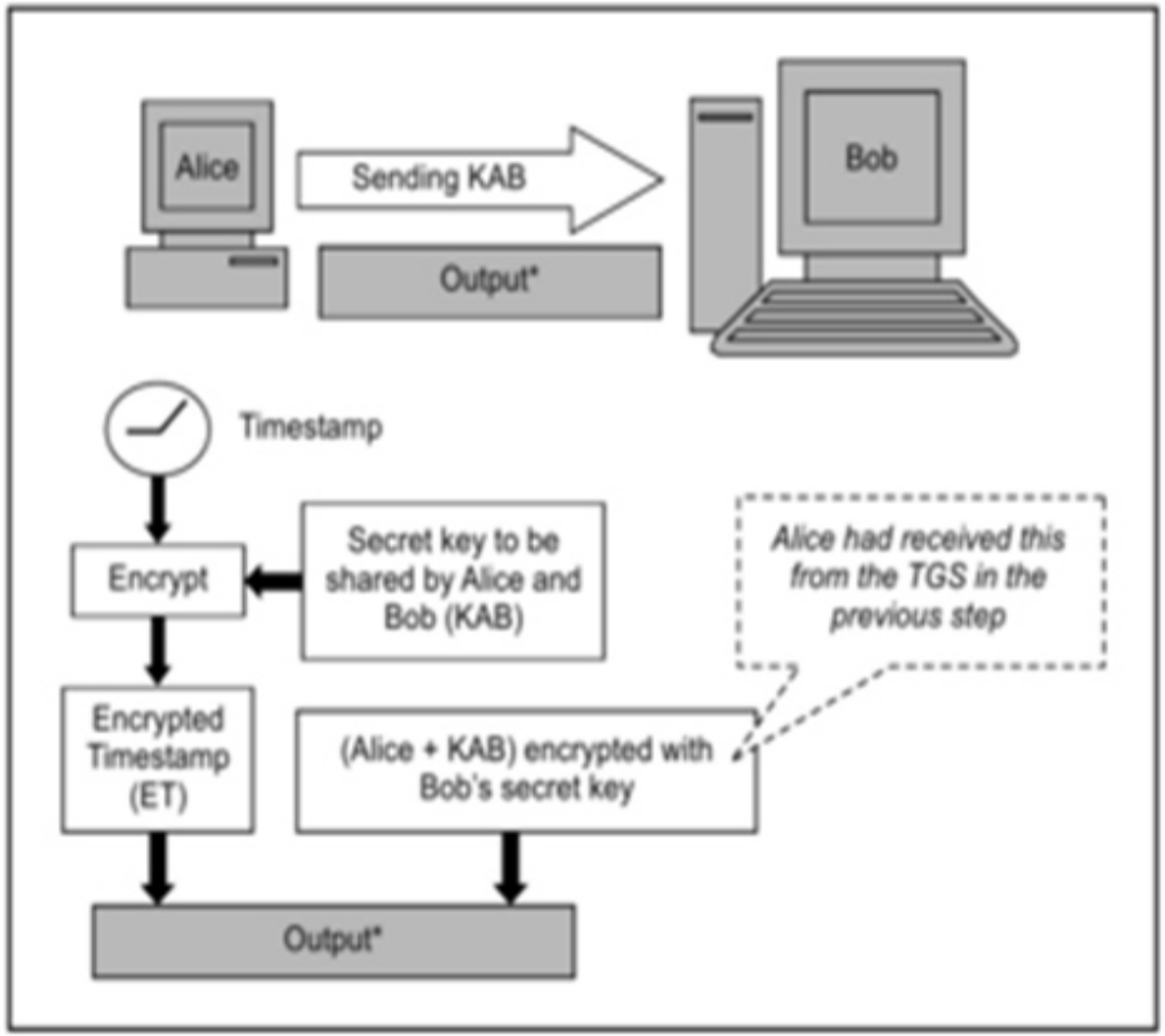


Alice sends a request for a SGT to the TGS

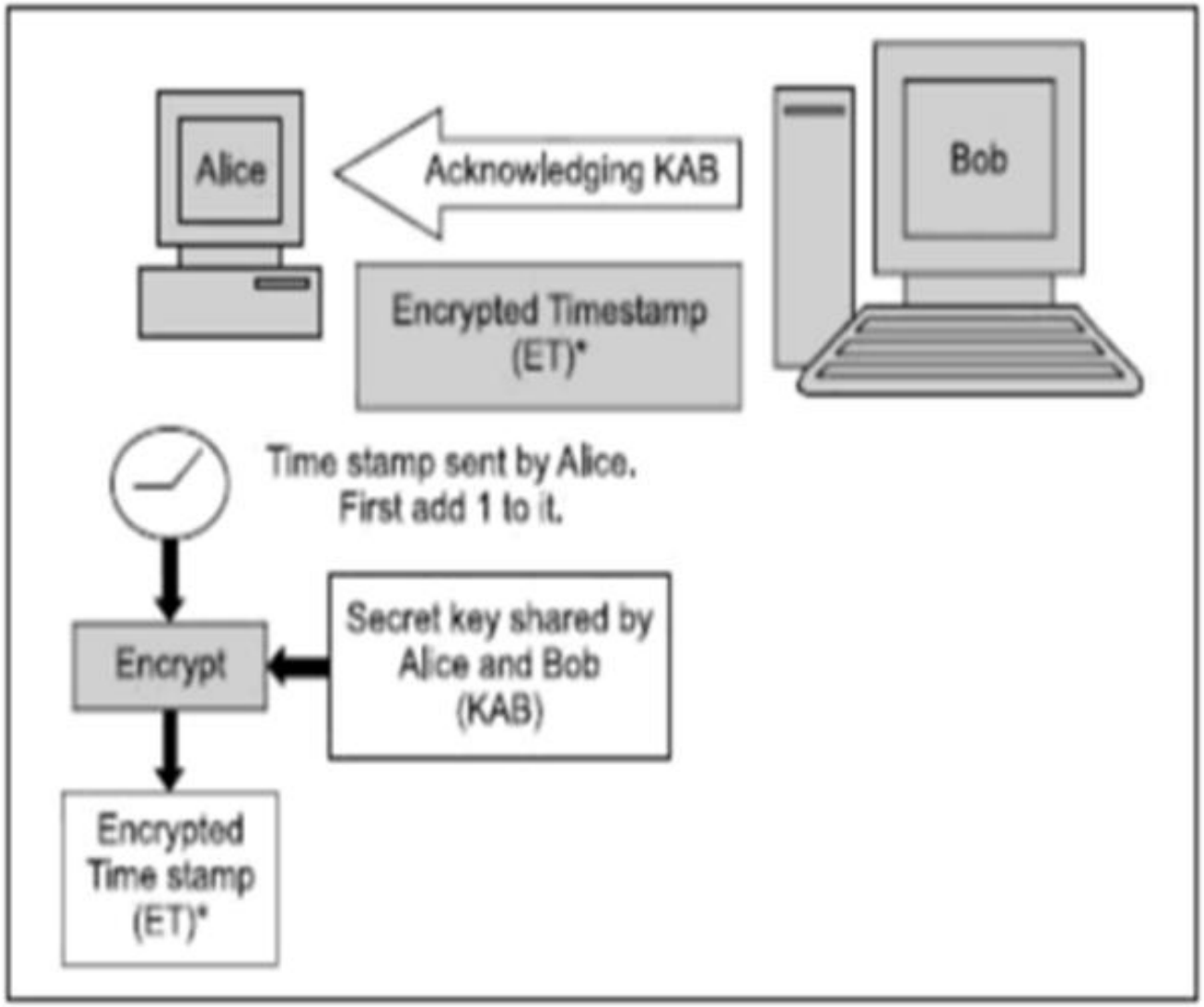


TGS sends response back to Alice

Step 3:User contacts Bob for accessing the server:



Alice sends KAB securely to Bob



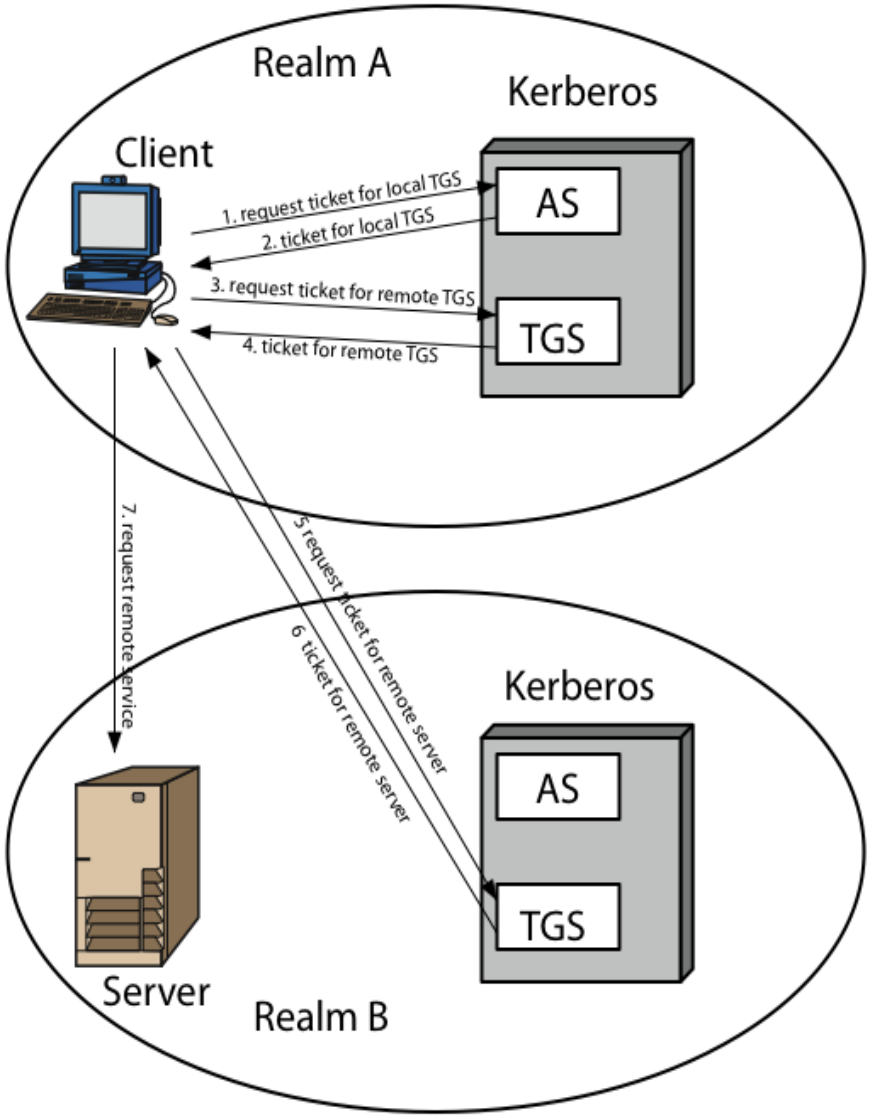
Bob acknowledges the receipt of KAB

- If alice wants to communicates with carol,she simply needs to obtain another shared key from the TGS specifying carol instead of bob.
- As long as alice communicating with bob alone she need not get a new ticket.
- Since she needs to sign only once this mechanism is called single sign on (SSO).

Kerberos Realms(william stallings)

- a Kerberos environment consists of:
  - a Kerberos server
  - a number of clients, all registered with server
  - application servers, sharing keys with server
- this is termed a realm
  - typically a single administrative domain
- if have multiple realms, their Kerberos servers must share keys and trust

Kerberos Realms



Kerberos version 5

- Version 5 overcomes some of the shortcomings of version 4.
- Version 4 demands the use of DES.
- Version 5 allows the flexibility in terms of allowing the choice of other algorithms.
- developed in mid 1990's
- provides improvements over v4
  - addresses environmental shortcomings
    - encryption alg, network protocol, byte order, ticket lifetime, authentication forwarding, interrealm auth
  - and technical deficiencies
    - double encryption, non-std mode of use, session keys, password attacks

THANK YOU

X.509 AUTHENTICATION SERVICE

Ref:Cryptography and Network Security-
Behrouz A Forouzan

Course Outcome

- Explain various authentication algorithms for network security

Learning Outcome

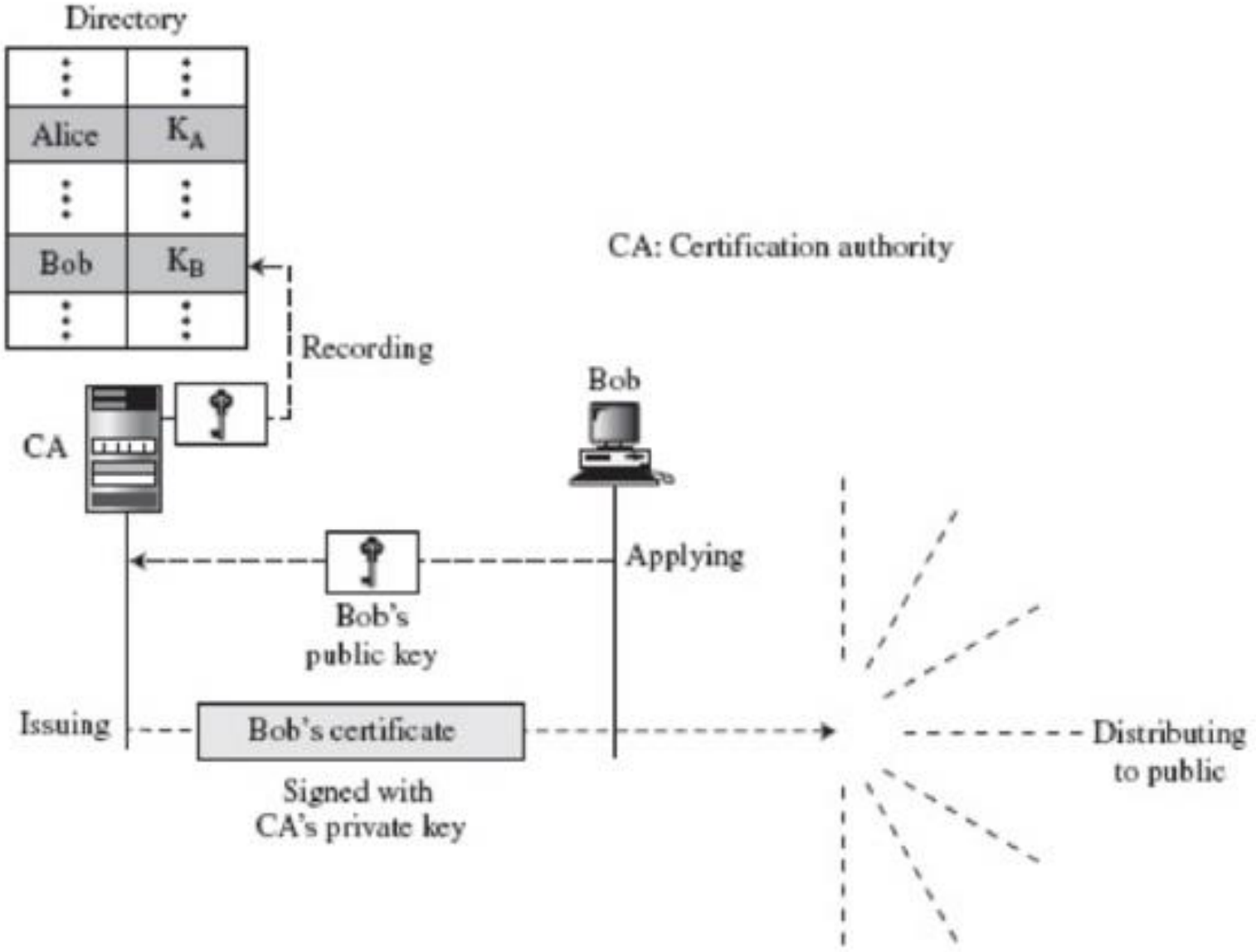
- Understand authentication services

Specific Outcome

- Understand authentication service X.509

Certificate authority

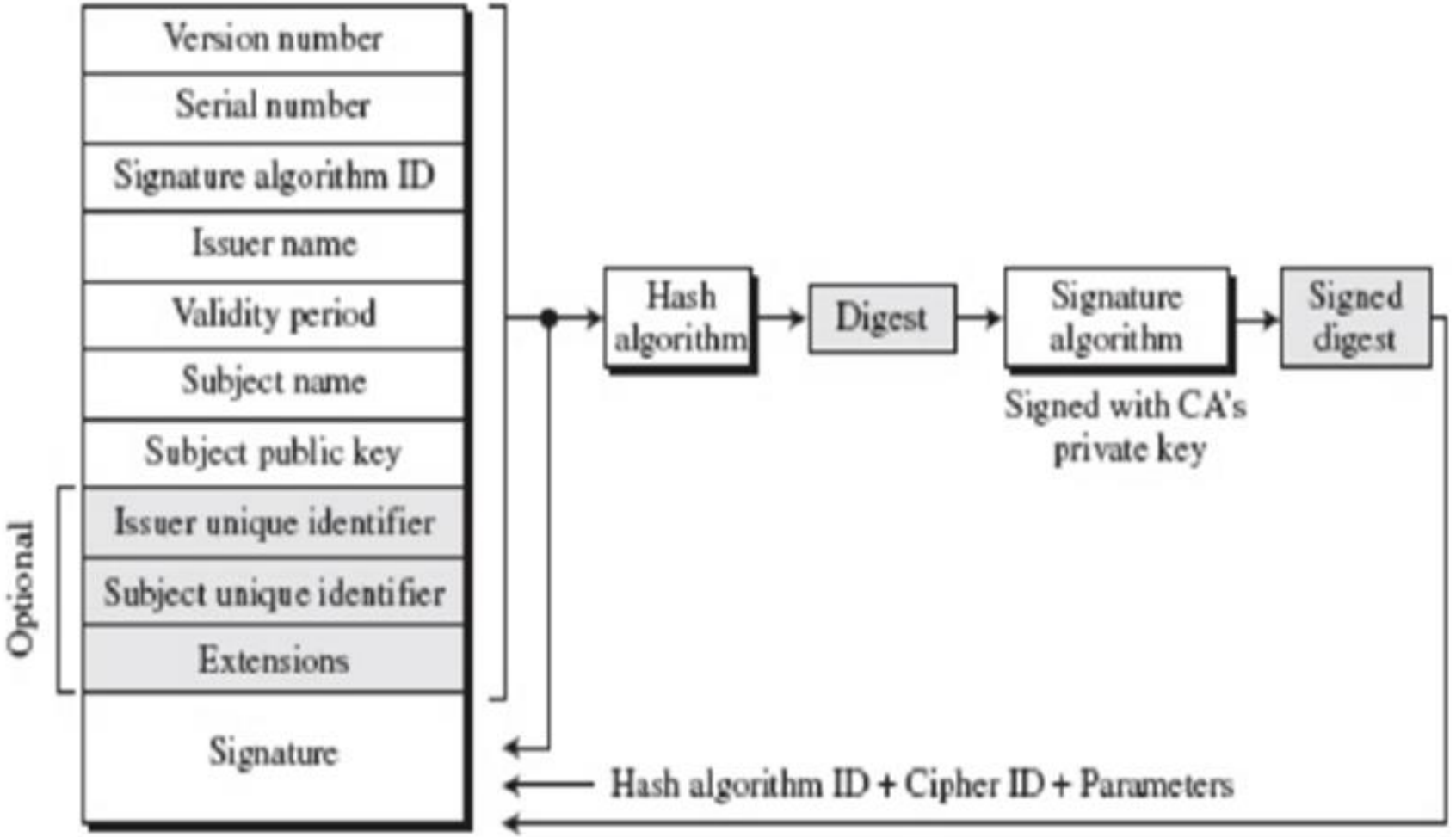
- **Certification authority (CA)** is a federal or state organization that binds a public key to an entity and issue a certificate.
- CA has a well known public key which cannot be forged.


Directory
Alice
K_A
Bob
K_B
Recording
CA
CA: Certification authority
Bob
Applying
Bob's
public key
Issuing
Bob's certificate
Signed with
CA's private key
Distributing
to public

- Although the use of CA has solved the problem of public key fraud,it has created side effect.
- Each certificate may have different format.
- Anything that needs to be used universally must have universal format.
- To remove this side effect ,the ITU has designed X.509 ,a recommendation that has been accepted by the internet with some changes.

- X.509 is a way to describe the certificate in a structured way.
- It uses a well known protocol called as ASN.1(Abstract Syntax Notation 1).

X.509 Certificate format



- A certificate has the following fields

- *Version Number* This field defines the version of X.509 of the certificate. The version number started at 0; the current version (third version) is 2.

- *Serial Number* This field defines a number assigned to each certificate. The value is unique for each certificate issuer.

- *Signature Algorithm ID* This field identifies the algorithm used to sign the certificate. Any parameter that is needed for the signature is also defined in this field.

- *Issuer Name* This field identifies the certification authority that issued the certificate. The name is normally a hierarchy of strings that defines a country, a state, organization, department, and so on.

- *Validity Period* This field defines the earliest time (not before) and the latest time (not after) the certificate is valid.

Contd…

- *Subject Name* This field defines the entity to which the public key belongs. It is also a hierarchy of strings. Part of the field defines what is called the *common name*, which is the actual name of the beholder of the key.

- *Subject Public Key* This field defines the owner's public key, the heart of the certificate. The field also defines the corresponding public-key algorithm (RSA, for example) and its parameters.

- *Issuer Unique Identifier* This optional field allows two issuers to have the same *issuer* field value, if the *issuer unique identifiers* are different.

Contd…

- *Subject Unique Identifier* This optional field allows two different subjects to have the same *subject* field value, if the *subject unique identifiers* are different.
- **Extensions.** This optional field allows issuers to add more private information to the certificate.
- **Signature.** This field is made of three sections. The first section contains all other fields in the certificate. The second section contains the digest of the first section encrypted with the CA's public key. The third section contains the algorithm identifier used to create the second section.

Certificate Renewal

- Each certificate has a period of validity.
- If there is no problem with the certificate, the CA issues a new certificate before old one expires.
- The process is like renewal of credit cards by a credit card company.

Certificate revocation

- In some cases a certificate must be revoked before it expires. Examples
  - The usr's (subject's)private key(corresponding to the public key listed in the certificate)might have been compromised.
  - The CA is no longer willing to certify the user. For example ,the users certificate relates to an organization that she no longer works.

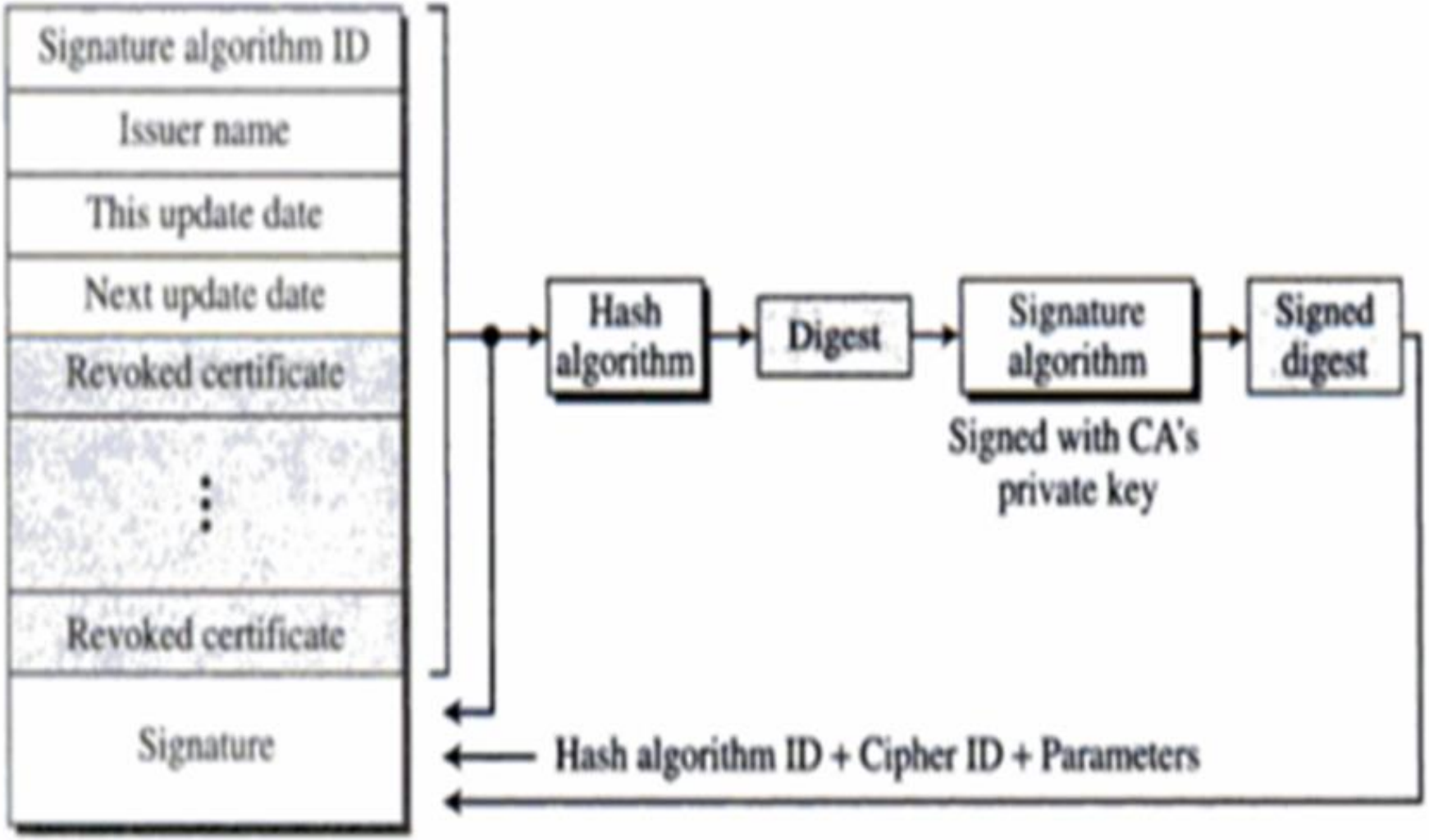
- The CA's private key which can verify certificates, may have been compromised in this case the CA needs to revoke all unexpired certificates.
- The revocation is done by periodically issuing a certificate revocation list(CRL).The list contains all revoked certificates that are not expired on the date the CRL is issued.
- When a user wants to use a certificate, she first needs to check the directory of the corresponding CA for the last certificate revocation list.

Fields in the Revocation

A certificate revocation list has the following fields:

- **Signature algorithm ID.** This field is the same as the one in the certificate.
- **Issuer name.** This field is the same as the one in the certificate.
- **This update date.** This field defines when the list is released.
- **Next update date.** This field defines the next date when the new list will be released.
- **Revoked certificate.** This is a repeated list of all unexpired certificates that have been revoked. Each list contains two sections: user certificate serial number and revocation date.
- **Signature.** This field is the same as the one in the certificate list.

Certificate revocation format



Delta Revocation

- To make revocation more efficient, the delta revocation list (delta CRL)has been introduced.
- A delta CRL is created and posted on the directory if there are changes after this update date and next update date.
- For example ,if CRLs are issued every month but there are revocations in between, the CA can create a delta CRL when there is a change during the month.
- However ,a delta CRL contains only the changes made after the last CRL.

Electronic mail Security

Ref:Cryptography and Network Security- Behrouz A Forouzan

Course Outcome

- Explain various authentication algorithms for network security

Learning Outcome

- Explain various cryptographic algorithms used for email security

Specific Outcome

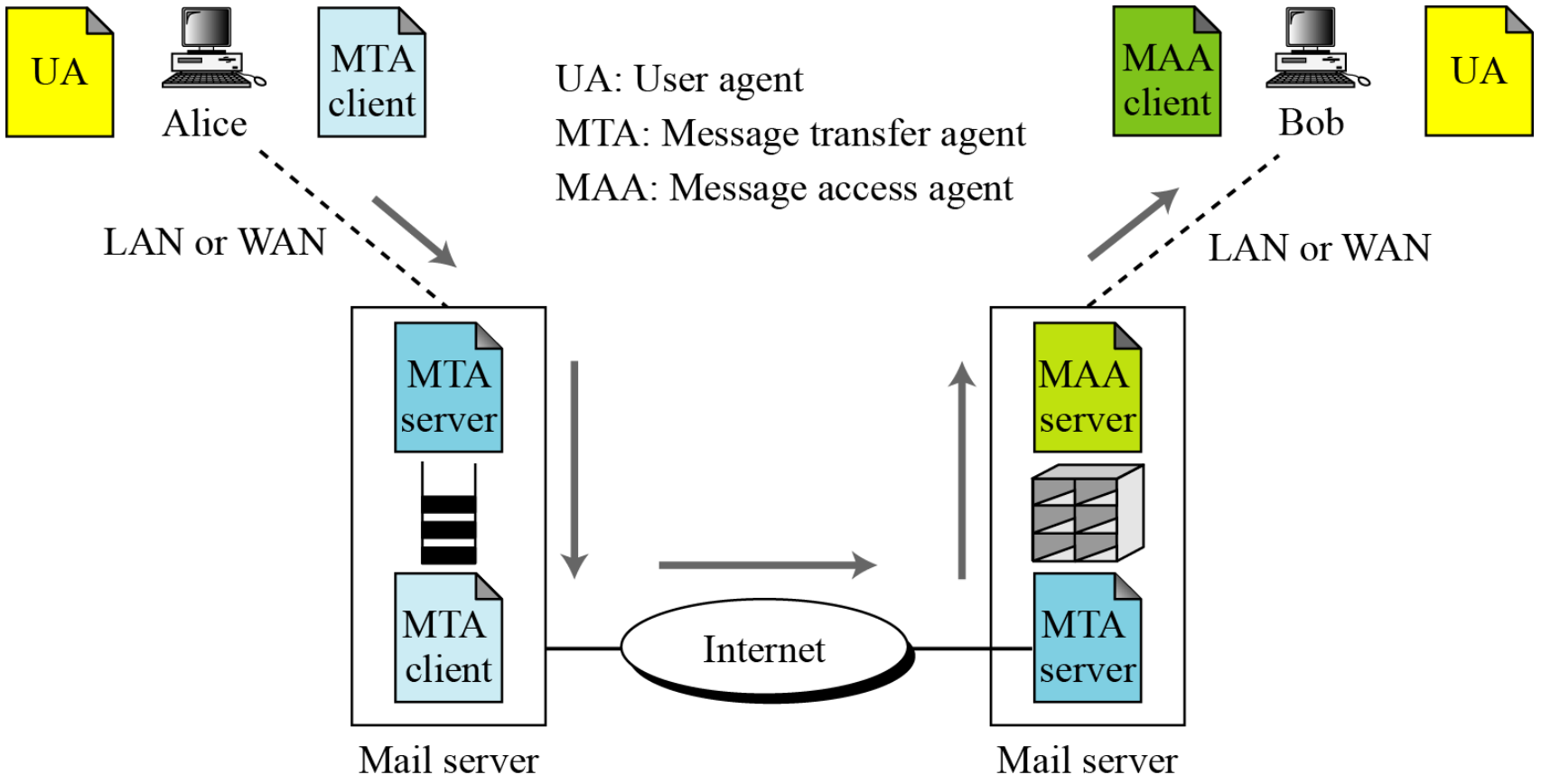
- Discuss more on email security architecture

Email Security

Protocols that provide security services for Email are

- Pretty Good Privacy
- Secure/Multipurpose Internet Mail Extension (S/MIME)

E-mail architecture



- When Alice need to send a message to Bob, she invokes a User Agent(UA) to prepare a message.
- She uses MTA(Message Transfer Agent) to send the message to the mail server at her site.
- MTA is a Client /Server Program with client installed at Alice computer and server installed at mail server.
- When message arrives at destination mail server, it is stored in Bob’s mailbox.
- When Bob retrieves his message,he invokes a program Message Access Agent (MAA)

Architecture of Email System

- Sending of email from Alice to Bob is a store-retrieve activity.
- Communication between Alice and Bob is through MTA and MAA.
- MTA Client program is a Push Program & MAA Client program is a Pull Program.
- Alice and Bob cannot communicate directly through MTA which requires MTA sever keep running all the time.

Cryptographic Algorithms

In e-mail security, the sender of the message needs to include the name or identifiers of the algorithms used in the message.

Eg., Triple DES for Encryption/Decryption and MD5 for Hashing

Cryptographic Secrets

In e-mail security, the encryption/ decryption is done using a symmetric-key algorithm, but the secret key to decrypt the message is encrypted with the public key of the receiver and is sent with the message.

Certificates

It is obvious that some public-key algorithms must be used for e-mail security.

Pretty Good Privacy (PGP)

Ref:Cryptography and Network Security-
Behrouz A Forouzan

Course Outcome

- Explain various authentication algorithms for network security

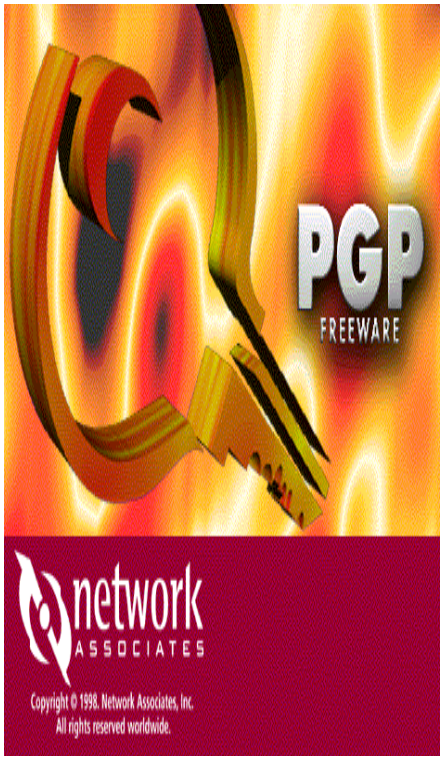
Learning Outcome

- Explain various cryptographic algorithms used for email security

Specific Outcome

- Explain PGP architecture

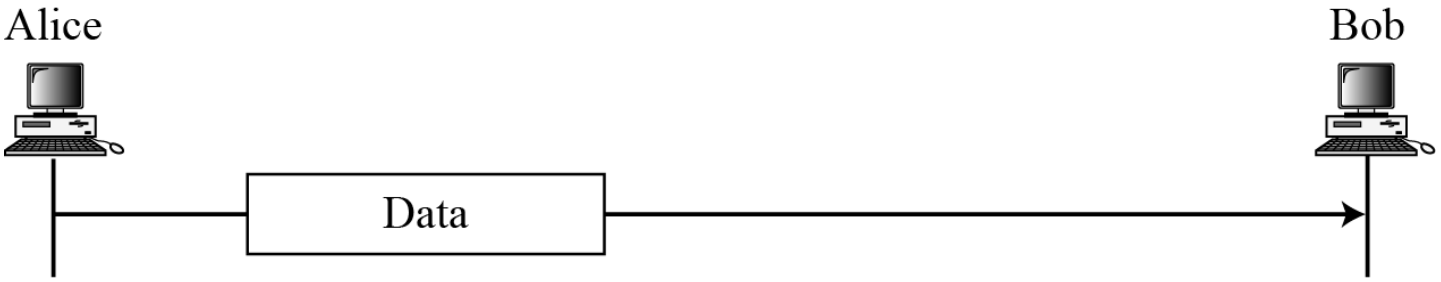
- Pretty Good Privacy (PGP) can be used to create a secure e-mail message or to store a file securely for future retrieval.
- It was invented by Phil Zimmermann to provide e-mail with privacy, integrity and authentication.



PGP is an open-source freely available software package for e-mail security. It provides authentication; confidentiality; compression; e-mail compatibility; and segmentation and reassembly.

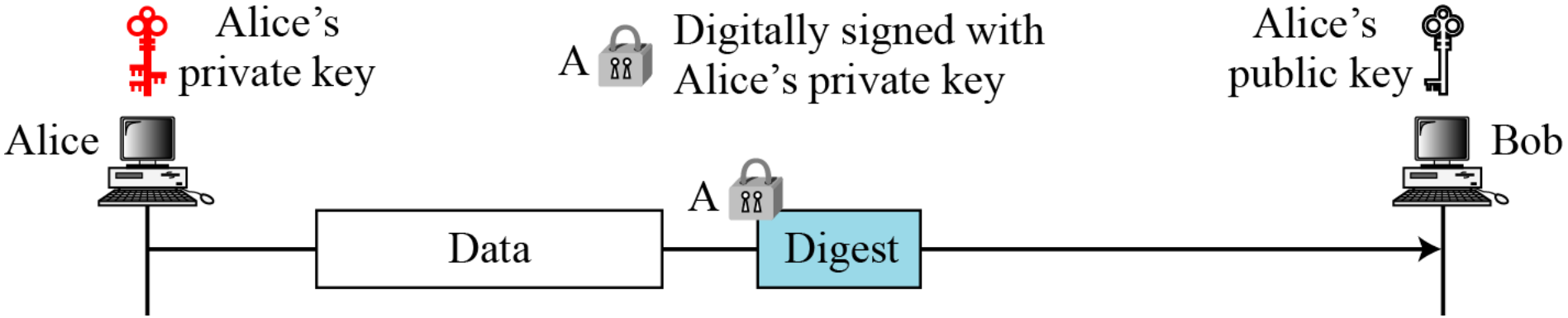
Scenarios

Plaintext



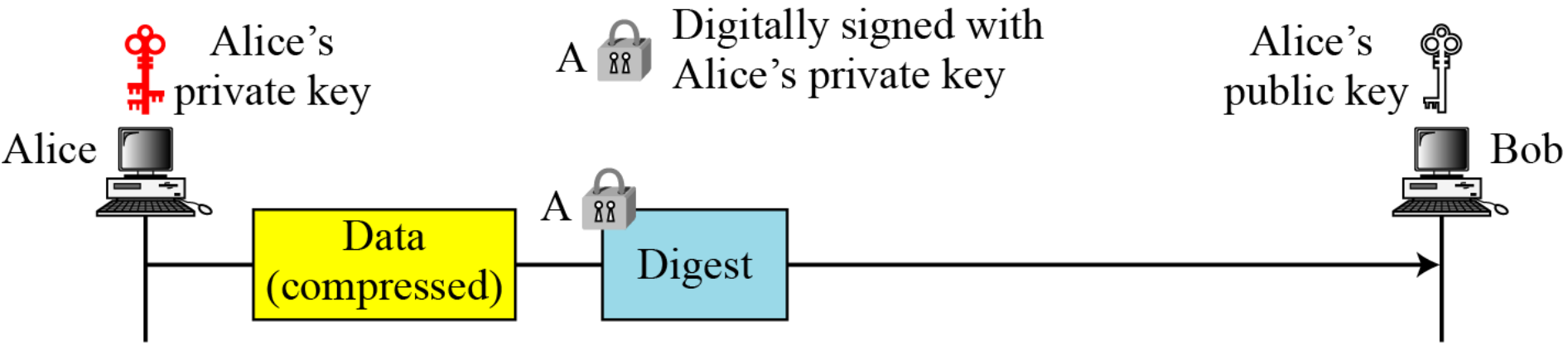
A plaintext message

Message Integrity



An authenticated message

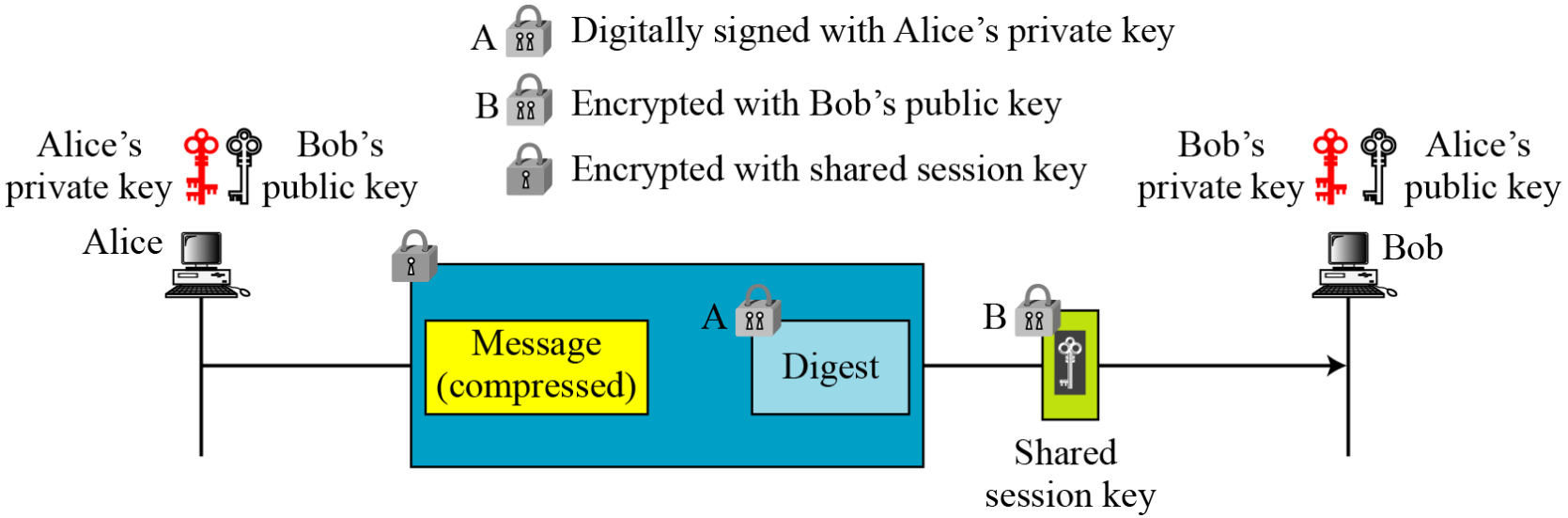
Compression



A message may be compressed, for storage or transmission, using ZIP.

A compressed message

Confidentiality with One-Time Session Key



A confidential message

Two Cases may arise(Refer fig)

1. **Alice needs to send a message to another person in the community**.

a. She uses her private key to sign the digest

b. She uses the receivers public key to encrypt a newly created session key.

c. She encrypts the message & signed digest with the session key created.

2. **Alice receives a message from another person in the community**.

a. She uses her private key to decrypt the session key

b. She uses the session key to decrypt message & digest.

c. She uses her public key to verify the digest.

- **Code Conversion**
- Another service provided by PGP is code conversion. PGP uses Radix-64 conversion.
- To provide transparency for email applications, an encrypted message may be converted to an ASCII string using radix 64 conversion.

- **Segmentation**
- PGP allows segmentation of the message after it has been converted to Radix-64 .
- To accommodate maximum message size limitations, PGP performs segmentation and reassembly.

PGP Algorithms

- **Symmetric encryption:**
  - DES, 3DES, AES and others.
- **Public key encryption of session keys:**
  - RSA or ElGamal.
- **Hashing:**
  - SHA-1, MD-5 and others.
- **Signature:**
  - RSA, DSS, ECDSA and others.

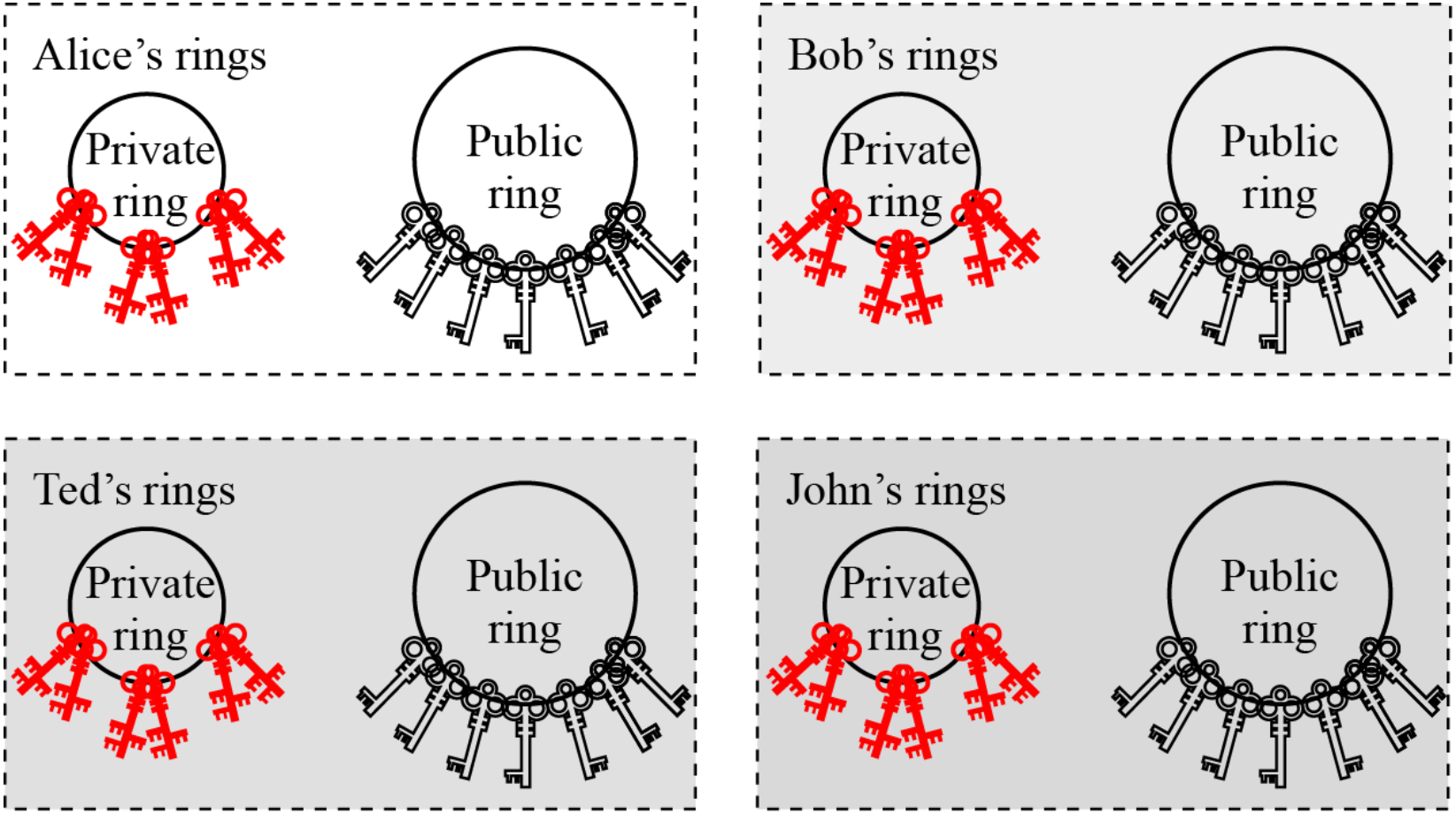
Key Rings

- Each User need to have two sets of rings:
  - A ring of private/Public keys
  - A ring of Public keys of other people

PGP use:

public keys for encrypting session keys / verifying signatures.
private keys for decrypting session keys / creating signatures.

Key rings in PGP



Pretty Good Privacy (PGP)

Ref:Cryptography and Network Security-
Behrouz A Forouzan

Course Outcome

- Explain various authentication algorithms for network security

Learning Outcome

- Explain various cryptographic algorithms used for email security

Specific Outcome

- Explain PGP architecture

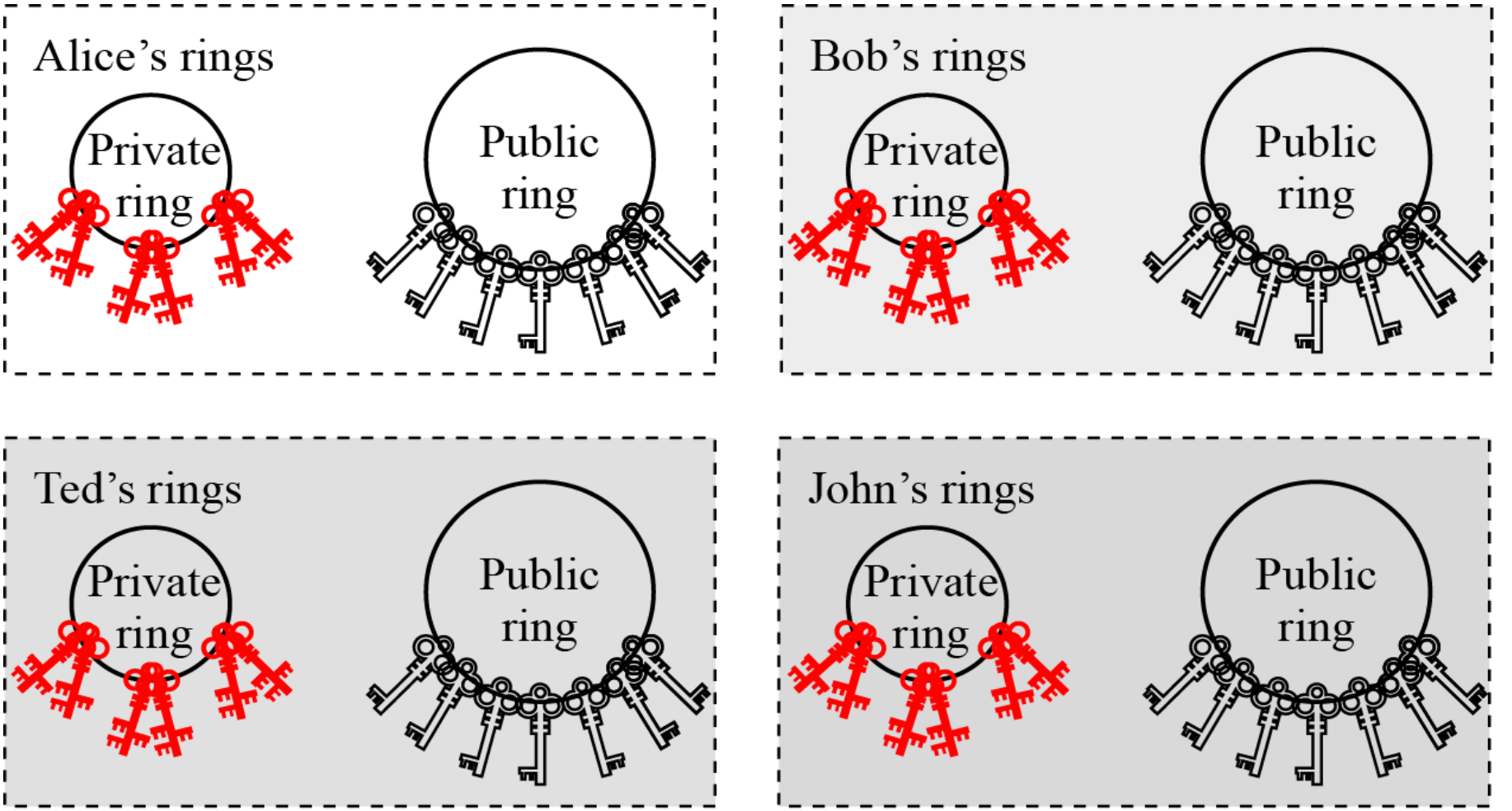
Key Rings

Each User need to have two sets of rings:

- A ring of private/Public keys
- A ring of Public keys of other people

Key Rings

Key rings in PGP



PGP Algorithms

Table 16.1 *Public-key algorithms*

ID	*Description*
1	RSA (encryption or signing)
2	RSA (for encryption only)
3	RSA (for signing only)
16	ElGamal (encryption only)
17	DSS
18	Reserved for elliptic curve
19	Reserved for ECDSA
20	ElGamal (for encryption or signing)
21	Reserved for Diffie-Hellman
100–110	Private algorithms

For signing the digest or encrypting messages

Table 16.2 *Symmetric-key algorithms*

ID	*Description*
0	No Encryption
1	IDEA
2	Triple DES
3	CAST-128
4	Blowfish
5	SAFER-SK128
6	Reserved for DES/SK
7	Reserved for AES-128
8	Reserved for AES-192
9	Reserved for AES-256
100–110	Private algorithms

Table 16.3 *Hash Algorithms*

ID	*Description*
1	MD5
2	SHA-1
3	RIPE-MD/160
4	Reserved for double-width SHA
5	MD2
6	TIGER/192
7	Reserved for HAVAL
100–110	Private algorithms

Table 16.4 *Compression methods*

ID	*Description*
0	Uncompressed
1	ZIP
2	ZLIP
100–110	Private methods

PGP Certificates

X.509 Certificates

Protocols that use X.509 certificates depend on the hierarchical structure of the trust.

In X.509, there is a single path from the fully trusted authority to any certificate.

PGP Certificates

In PGP, there is no need for CAs; anyone in the ring can sign a certificate for anyone else in the ring.

In PGP, there can be multiple paths from fully or partially trusted authorities to any subject.

The issuer of a certificate is usually called an introducer.

Trusts and Legitimacy

The entire operation of PGP is based on introducer trust, the certificate trust, and the legitimacy of the public keys.

Three levels of trust is assigned to the introducer: None, Partial, Full.
The introducers trust level specifies the trust levels issued by the introducer for other people in the ring.

Key Legitimacy

The level of key Legitimacy for a user is the weighted trust levels of that user.

1. A weight of 0 to a nontrusted certificate
2. A weight of 1/2 to a certificate with partial trust
3. A weight of 1 to a certificate with full trust

Key Ring Table

Format of private key ring table

User ID	Key ID	Public key	Encrypted private key	Timestamp
⋮	⋮	⋮	⋮	⋮

User Id – Email ID of the User
Key ID – 64 bits of the public Key
Public Key – Public key belonging to the Public/Private key pair
Encrypted Private Key – Encrypted value of private Key
Timestamp – Date and Time of the Key pair creation

Let us show a private key ring table for Alice. We assume that Alice has only two user IDs, alice@some.com and alice@anet.net. We also assume that Alice has two sets of private/public keys, one for each user ID.

Table 16.5 *Private key ring table for Example 1*

User ID	*Key ID*	*Public Key*	*Encrypted Private Key*	*Timestamp*
alice@anet.net	AB13...45	AB13...45...59	**32452398...23**	031505-16:23
alice@some.com	FA23...12	FA23...12...22	**564A4923...23**	031504-08:11

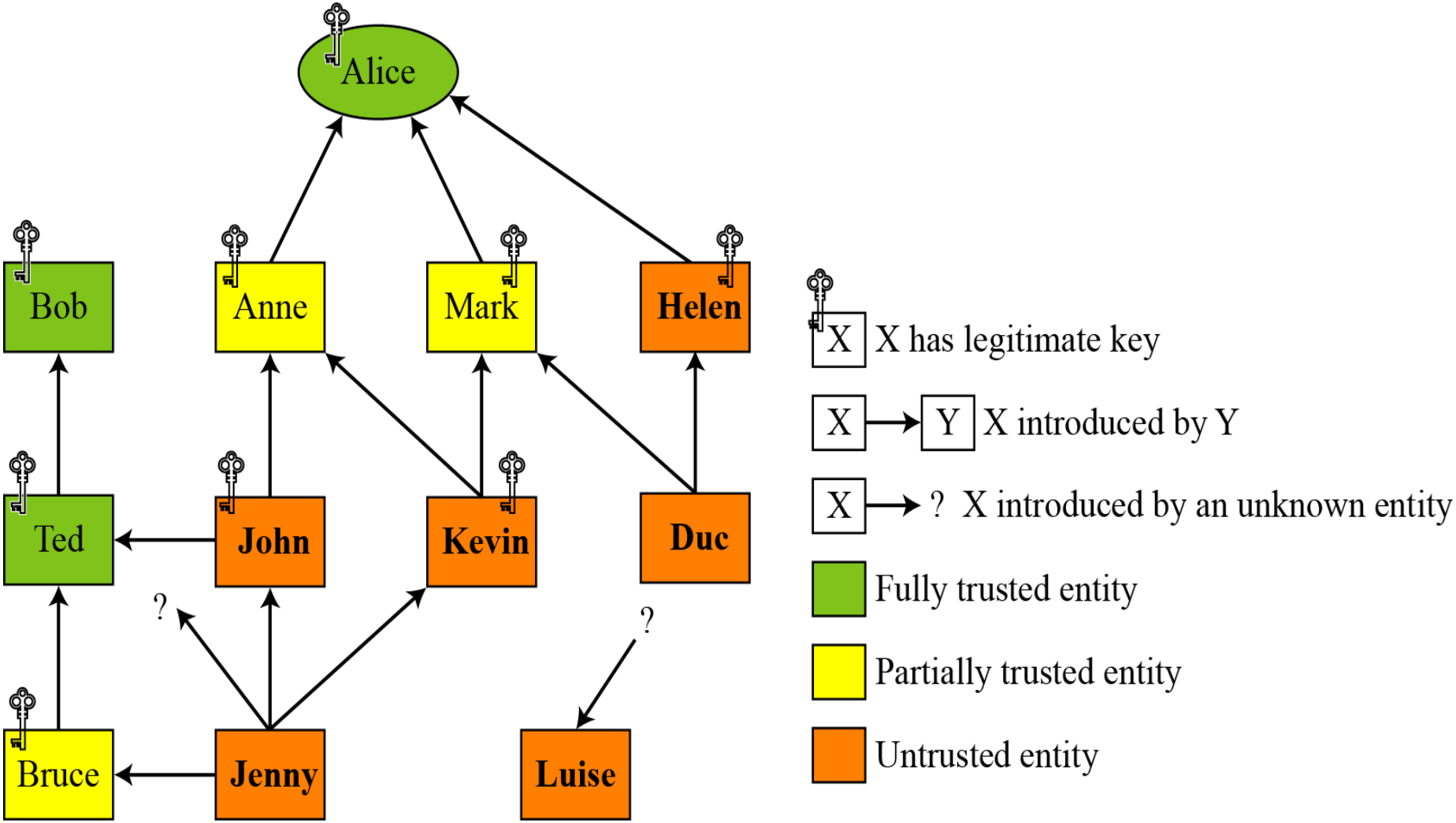
Format of a public key ring table

Public ring

User ID	Key ID	Public key	Producer trust	Certificate(s)	Certificate trust(s)	Key Legitimacy	Timestamp
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Public Key – Public Key of the Entity
Producer Trust – None, Partial, Full
Certificates – certificates signed by other entities
Certificate Trust – Certificates Trust or trusts
Key Legitimacy – calculated based on the value of the certificate trust & predefined weight for the certificate trust .

Trust Model in PGP



DR MAHALINGAM
MCET
COLLEGE OF ENGINEERING AND TECHNOLOGY
Enlightening Technical Minds

THANK YOU

Pretty Good Privacy (PGP)

Ref:Cryptography and Network Security-
Behrouz A Forouzan

Course Outcome

- Explain various authentication algorithms for network security

Learning Outcome

- Explain various cryptographic algorithms used for email security

Specific Outcome

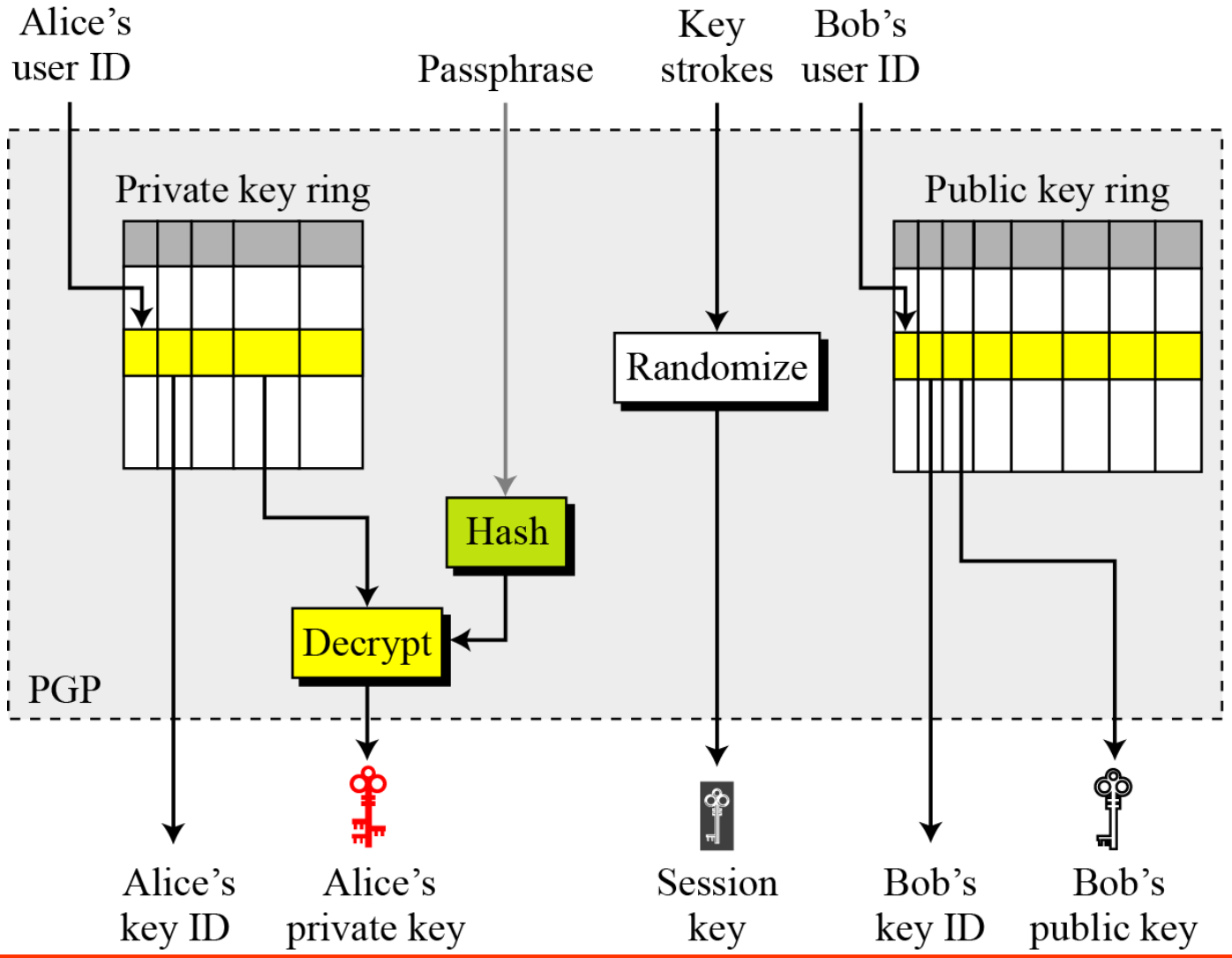
- Explain PGP architecture

Key Revocation

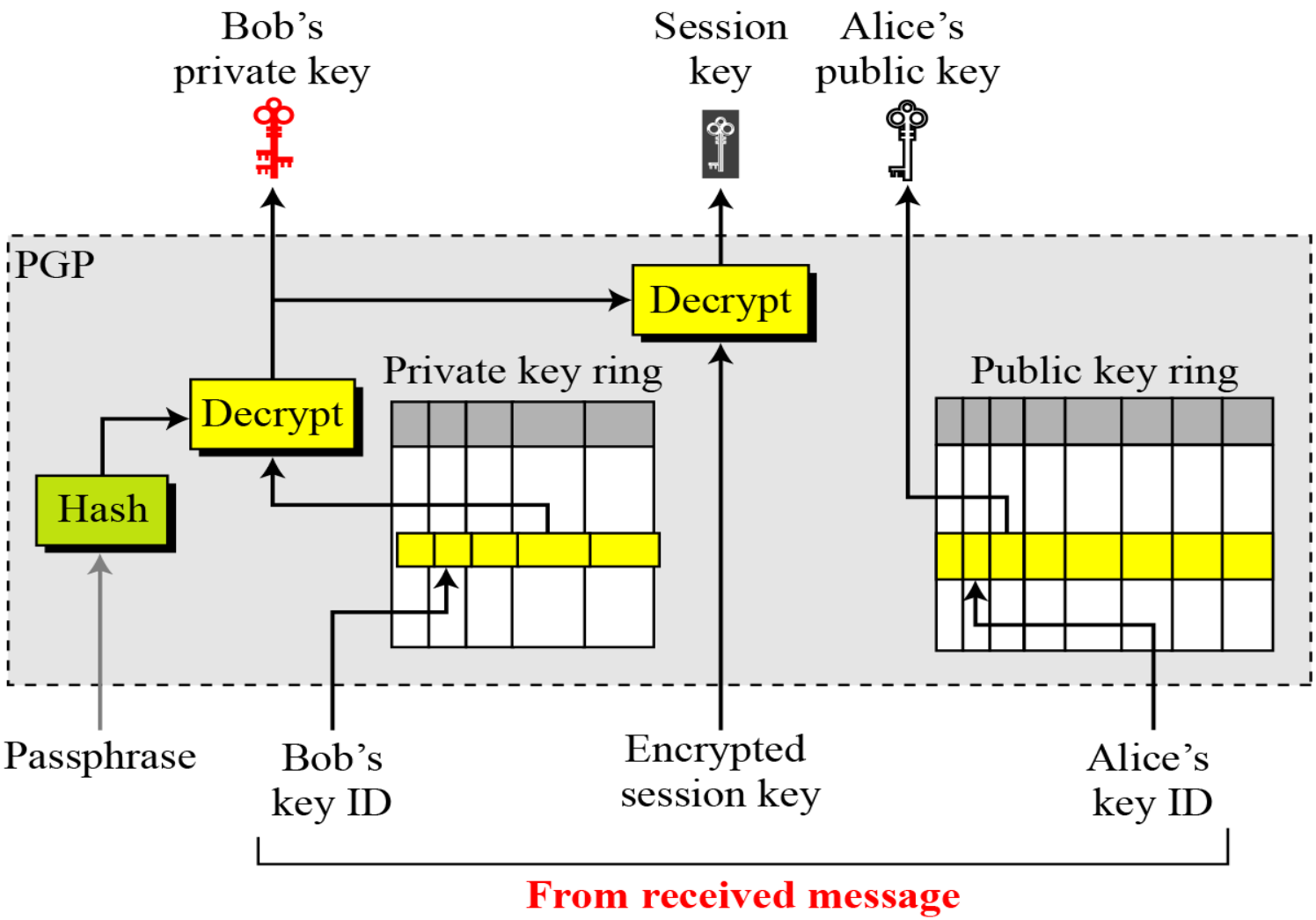
It may become necessary for an entity to revoke his or her public key from the ring. This may happen if the owner of the key feels that the key is compromised (stolen, for example) or just too old to be safe.

Extracting Information from Rings

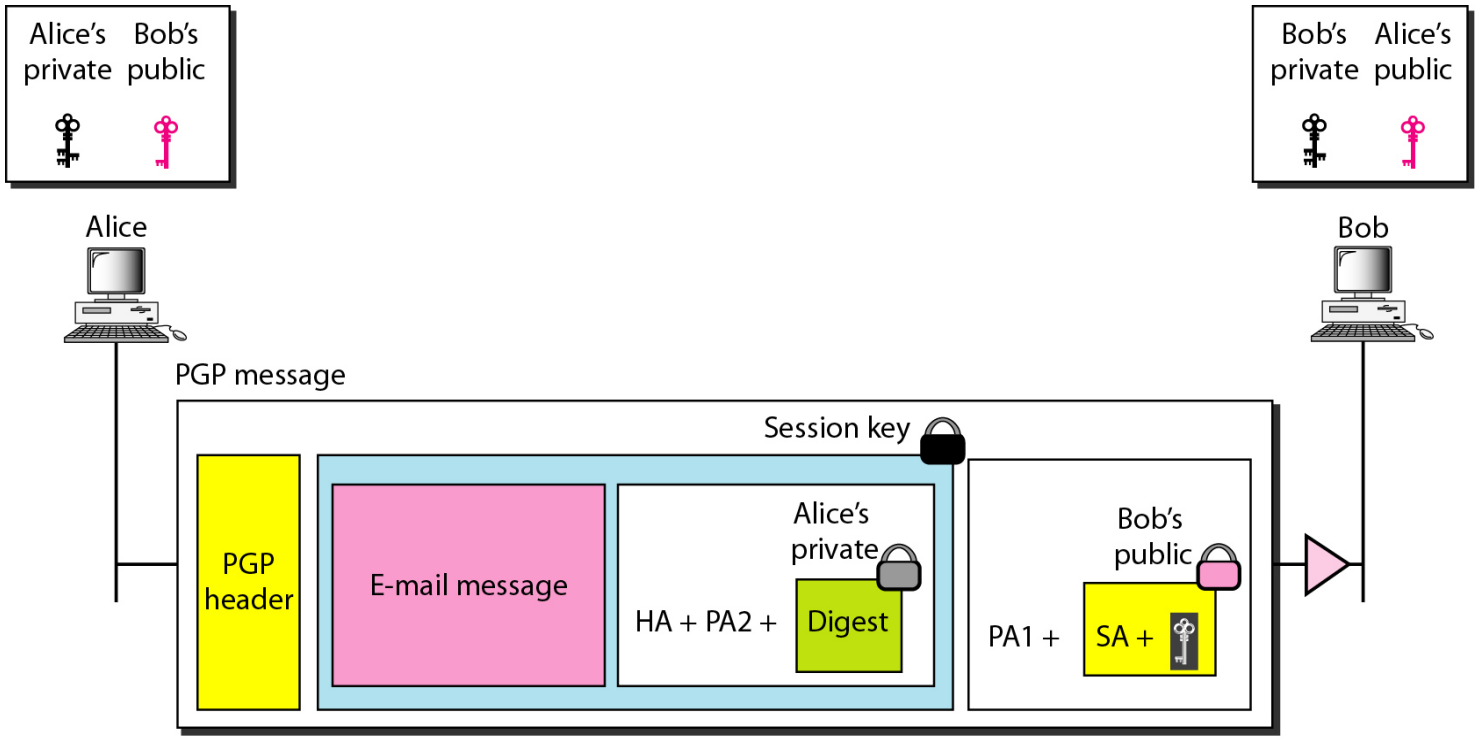
Extracting information at the sender site



Extracting information at the receiver site

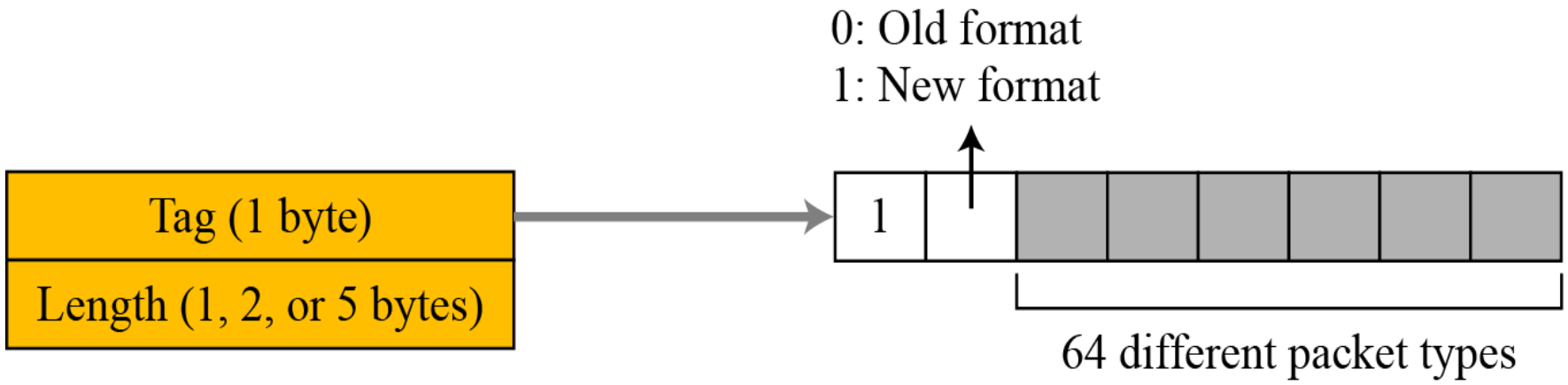


A scenario in which an e-mail message is authenticated and encrypted



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)

PGP Packets



Format of packet header

Table 16.12 *Some commonly used packet types*

Value	*Packet type*
1	Session key packet encrypted using a public key
2	Signature packet
5	Private-key packet
6	Public-key packet
8	Compressed data packet
9	Data packet encrypted with a secret key
11	Literal data packet
13	User ID packet

PGP Messages

An encrypted message - can be a sequence of two packets, a session-key packet and a symmetrically encrypted packet.

Signed message – Combination of signature packet and literal packet.

Certificate message - Combination of User ID packet and a public – key packet. Signature is calculated on the concatenation of the key and user ID.

DR MAHALINGAM
MCET
COLLEGE OF ENGINEERING AND TECHNOLOGY
Enlightening Technical Minds

THANK YOU

S/MIME

Ref:Ref:Cryptography and Network Security-Behrouz A Forouzan

Course Outcome

- Explain various authentication algorithms for network security

Learning Outcome

- Understand the security services designed for email application

Specific Outcome

- Understand the security protocol S/MIME designed for email application

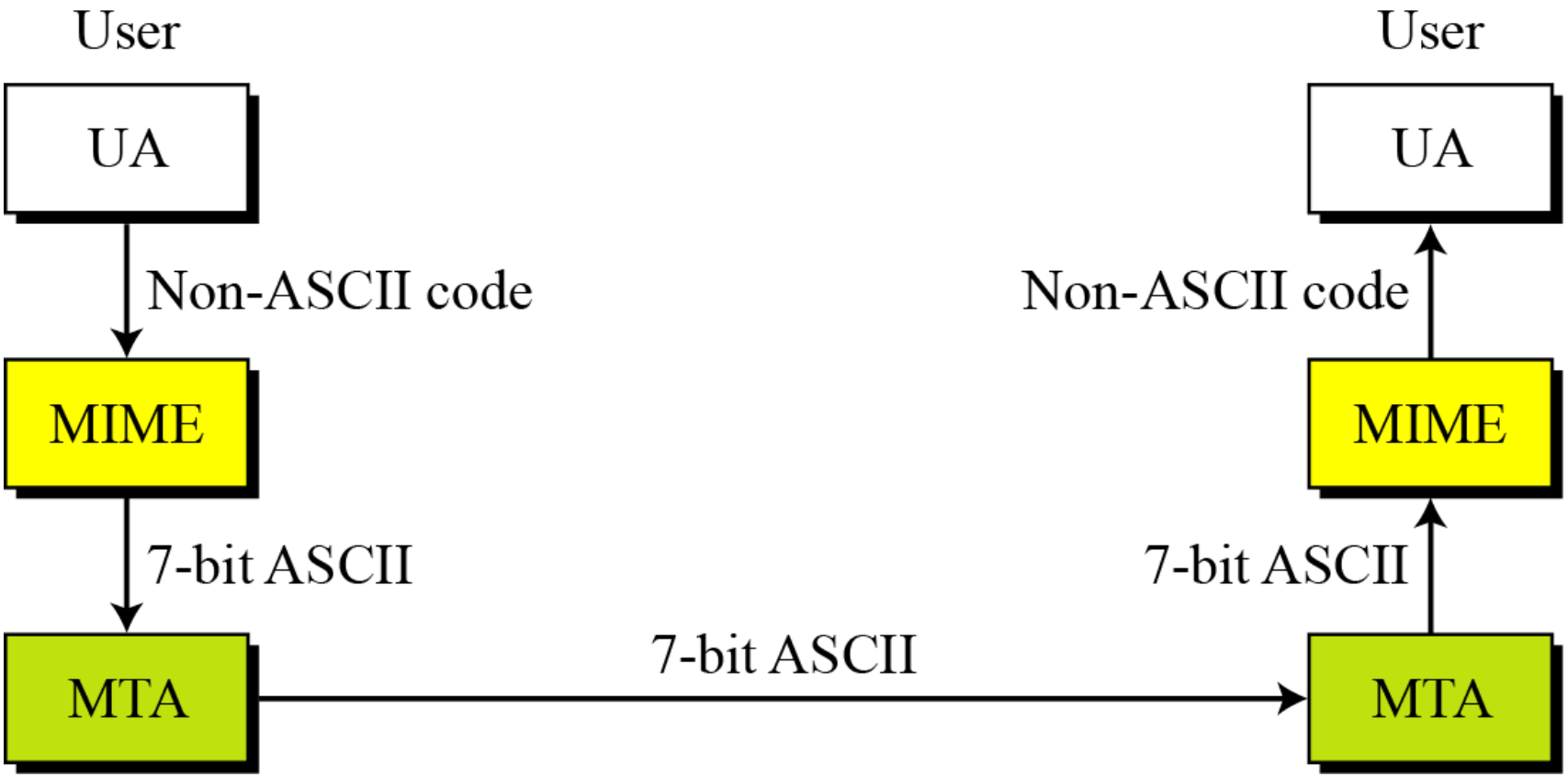
S/MIME

Another security service designed for electronic mail is Secure/Multipurpose Internet Mail Extension (S/MIME). The protocol is an enhancement of the Multipurpose Internet Mail Extension (MIME) protocol.

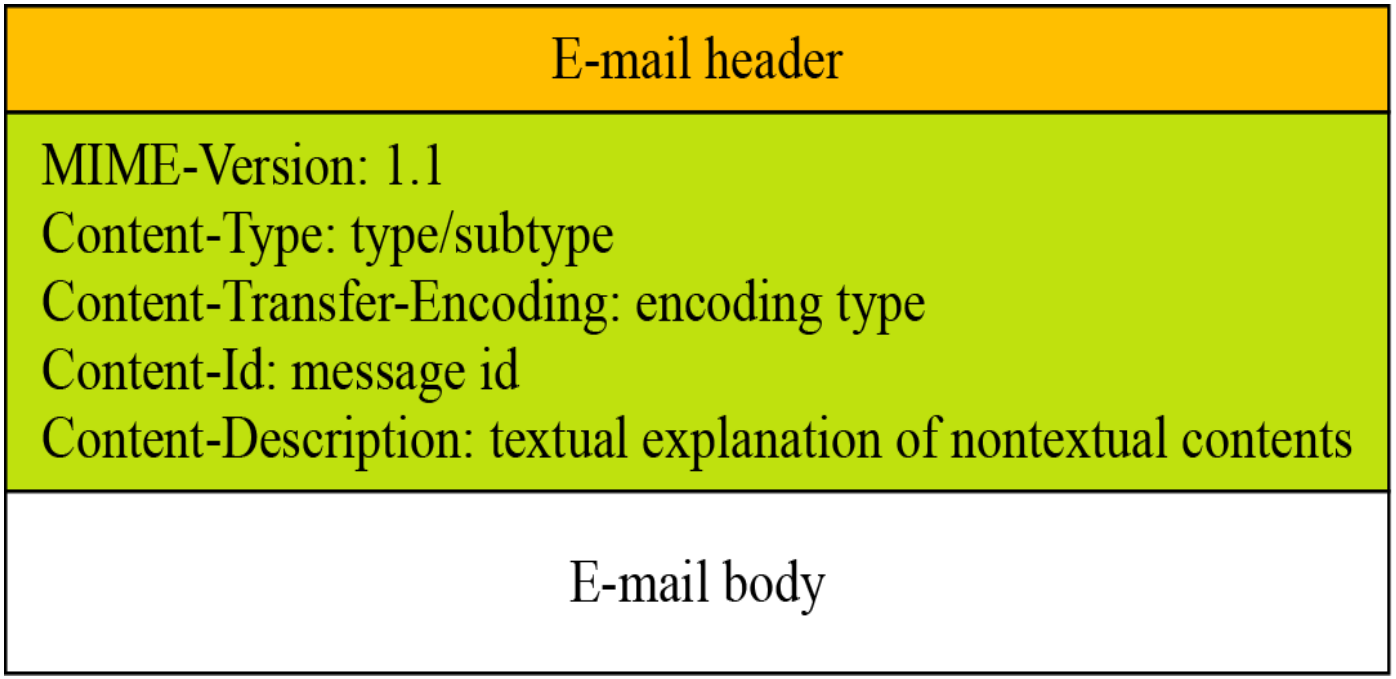
MIME

- It can send messages only in NVT 7-bit ASCII format.
- It cannot be used to send binary files or video or audio data.
- MIME is a supplementary protocol that allows non ASCII data to be sent through e mail.
- MIME transforms non ASCII data at the sender site to NVT ASCII data and delivers to the client MTA to be sent through the internet.

MIME


User
UA
Non-ASCII code
MIME
7-bit ASCII
MTA
7-bit ASCII
User
UA
Non-ASCII code
MIME
7-bit ASCII
MTA

MIME Header



MIME headers

MIME-Version

This header defines the version of MIME used. The current version is 1.1.

Content-Type

The content type and the content subtype are separated by a slash. Depending on the subtype, the header may contain other parameters.

Content-Type: <type / subtype; parameters>

Data types and subtypes in MIME

Type	*Subtype*	*Description*
Text	Plain	Unformatted.
	HTML	HTML format.
Multipart	Mixed	Body contains ordered parts of different data types.
	Parallel	Same as above, but no order.
	Digest	Similar to Mixed, but the default is message/RFC822.
	Alternative	Parts are different versions of the same message.
Message	RFC822	Body is an encapsulated message.
	Partial	Body is a fragment of a bigger message.
	External-Body	Body is a reference to another message.
Image	JPEG	Image is in JPEG format.
	GIF	Image is in GIF format.
Video	MPEG	Video is in MPEG format.
Audio	Basic	Single channel encoding of voice at 8 KHz.
Application	PostScript	Adobe PostScript.
	Octet-stream	General binary data (eight-bit bytes).

Content Transfer Encoding

Type	*Description*
7bit	NVT ASCII characters and short lines.
8bit	Non-ASCII characters and short lines.
Binary	Non-ASCII characters with unlimited-length lines.
Radix-64	6-bit blocks of data are encoded into 8-bit ASCII characters using Radix-64 conversion.
Quoted-printable	Non-ASCII characters are encoded as an equal sign followed by an ASCII code.

S/MIME

- S/MIME adds new content types to include security services to the MIME.
- These new types include the parameter "application/pkcs7-mime," in which "pkcs" defines "Public Key Cryptography Specification."

Cryptographic Message Syntax (CMS)

To define how security services, such as confidentiality or integrity, can be added to MIME content types, S/MIME has defined Cryptographic Message Syntax (CMS).

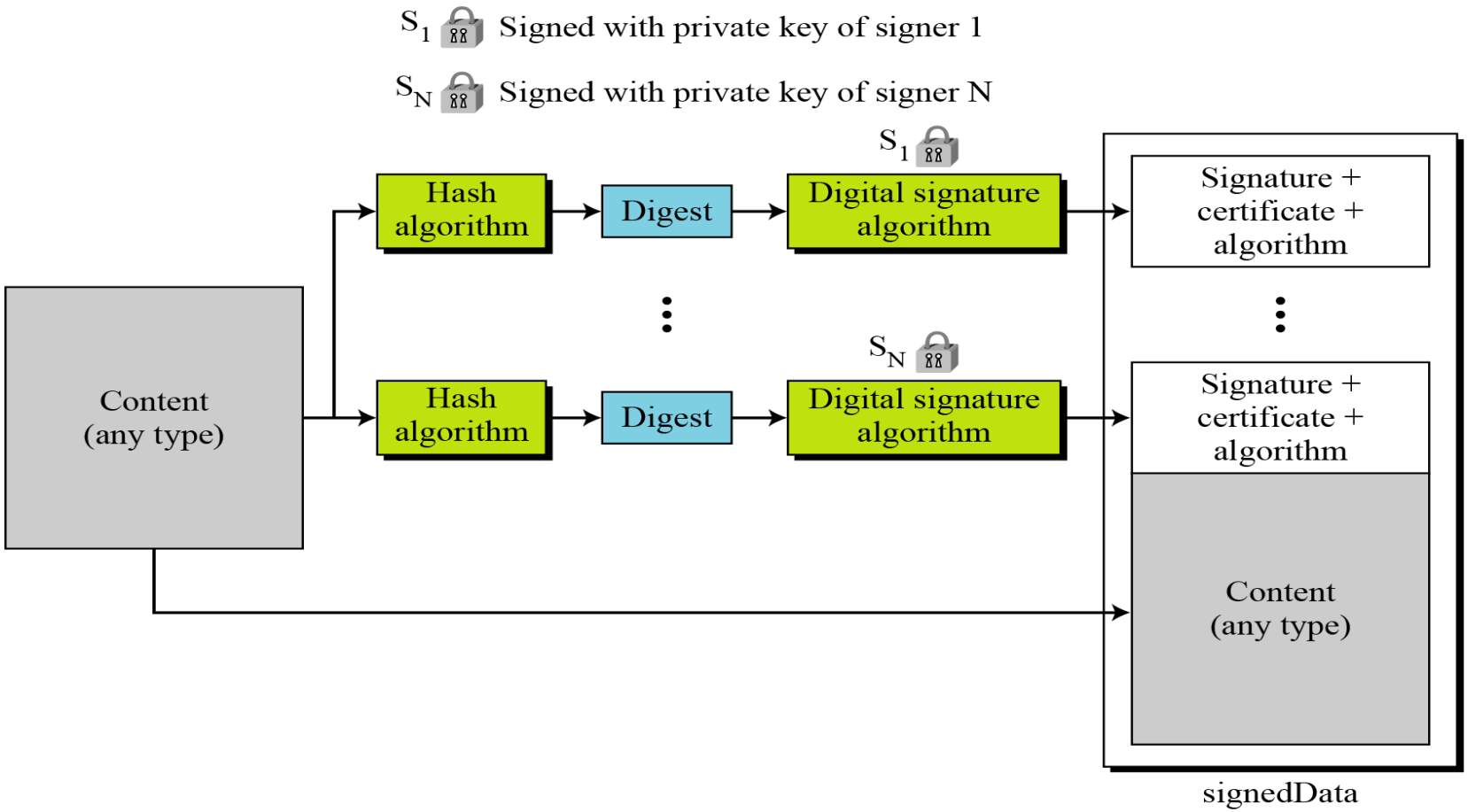
The syntax in each case defines the exact encoding scheme for each content type.

Contd…

- S/MIME provides the following cryptography security services:
  - ✓Authentication.
  - ✓Message Integrity - By using digital signing
  - ✓Non-repudiation of origin.
  - ✓Privacy and data security - By using encryption

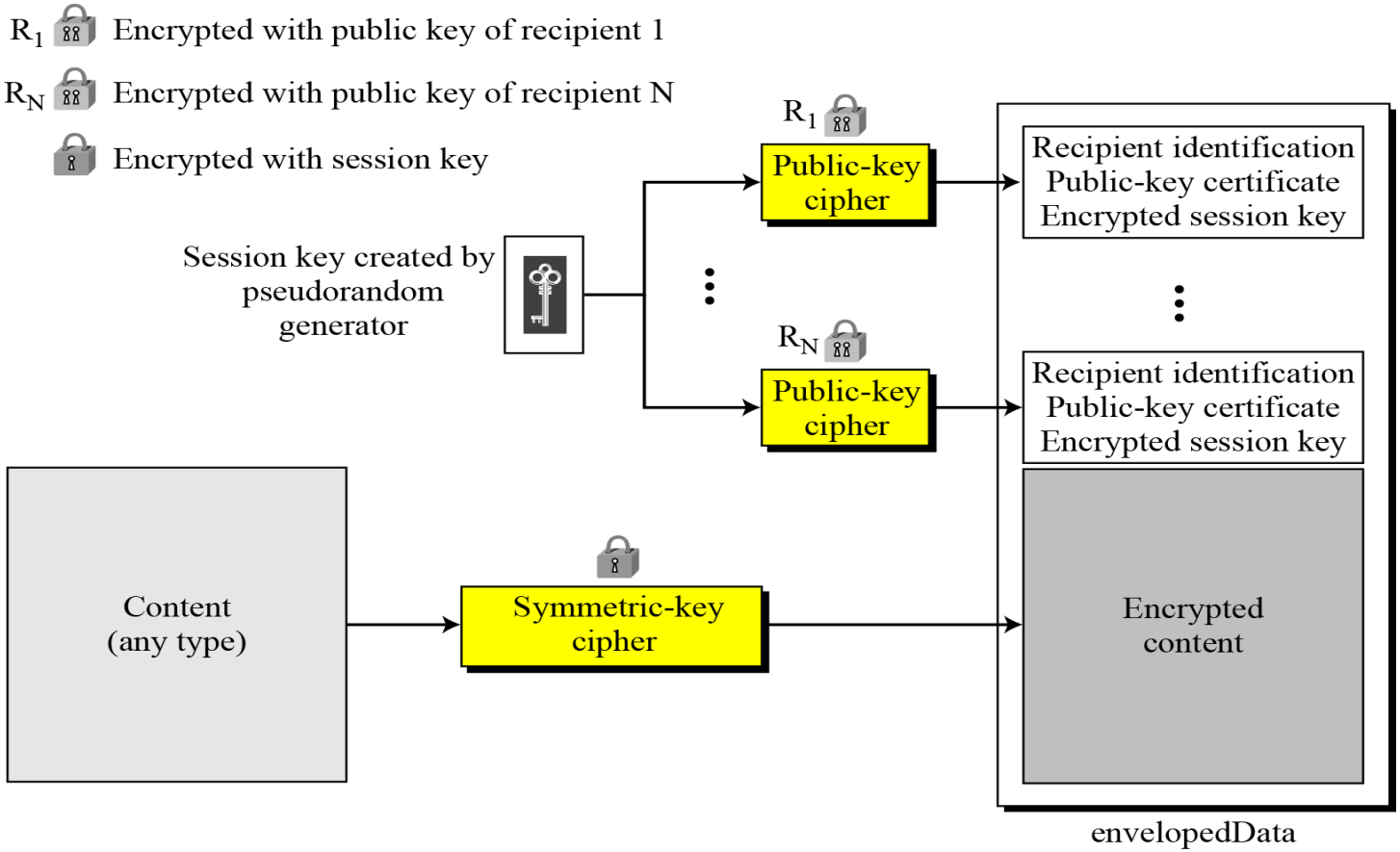
Types of messages

- ✓**Data content type**: This is an arbitary string. The object created is called data.

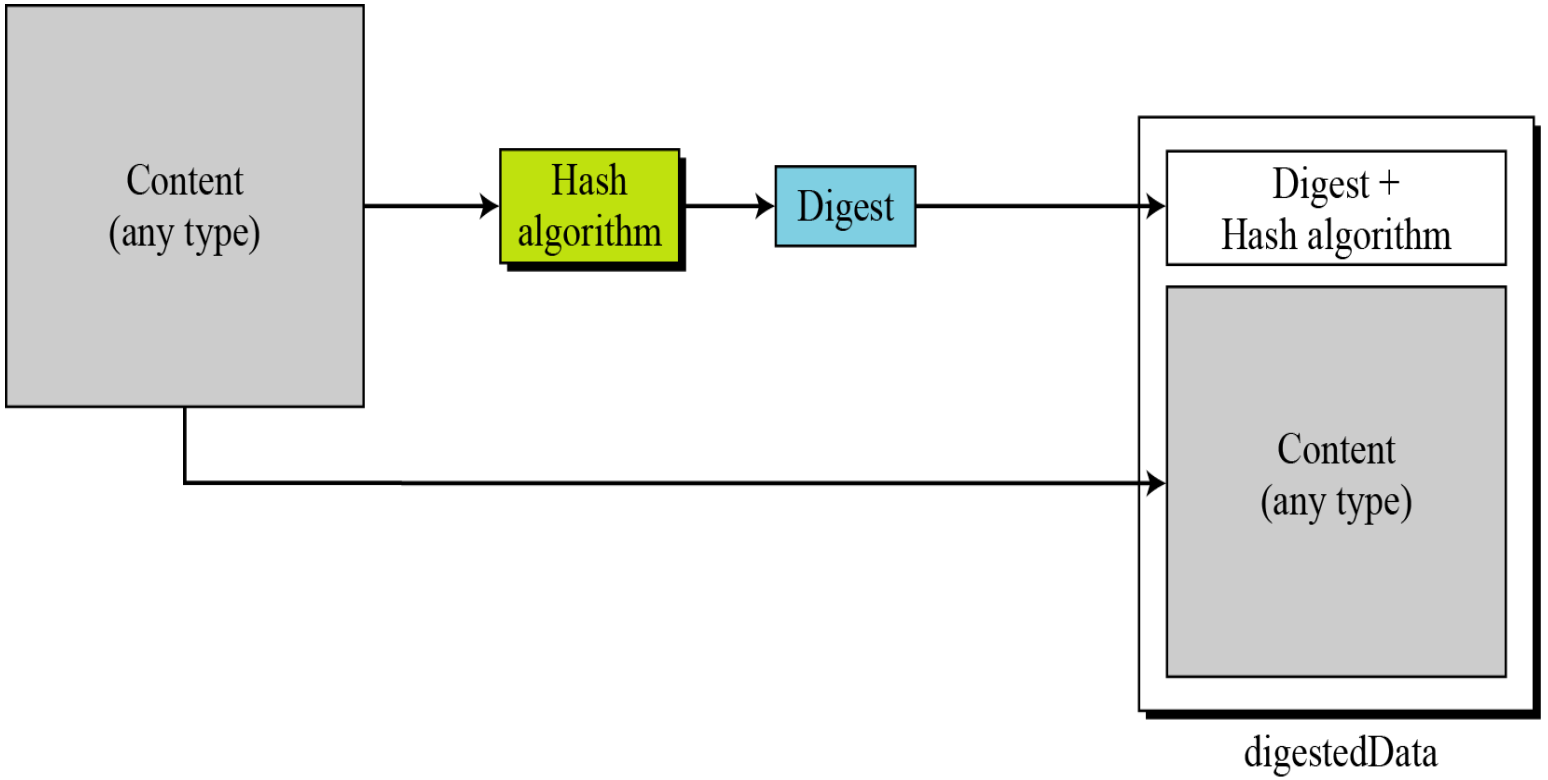


- **Signed data content type**:This provides only integrity of data. The encoded result is an object called signed data.

Enveloped data content type:This type is used to provide privacy for the message. The encoded result is an object called enveloped data.

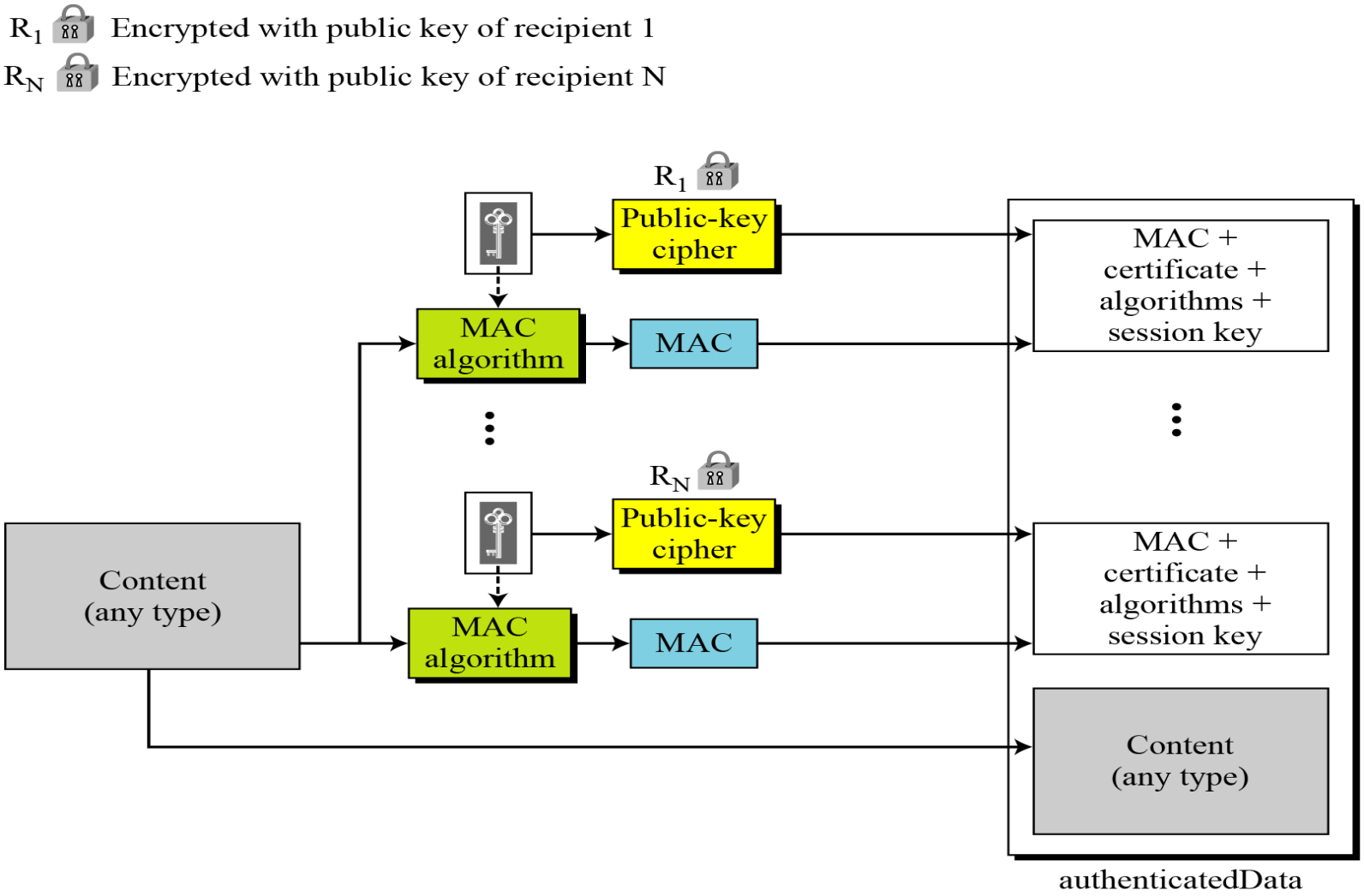


- **Digested data content type:** This type is used to provide integrity for the message. The encoded result is an object called digested data.



- **Encrypted data content type**:Used to create an encrypted version of any content type. Although this looks like enveloped data content type, the encrypted data content type has no recepient.
- It can be used to store the encrypted data instead of transmitting it.
- The user employs any key and any algorithm to encrypt the content.
- The encrypted content is stored without including the key or the algorithm.
- The object created is called encrypted data.

- **Authenticated data content type:** This type is used to provide authentication of the data



- **Key management:** Key management in S/MIME is a combination of key management used by X.509 and PGP.
- S/MIME uses public key certificates signed by the certificate authorities defined x.509.
- The user is responsible to maintain the web of trust to verify signatures as defined by PGP.

Cryptographic Algorithms

S/MIME defines several cryptographic algorithms. The term "must" means an absolute requirement; "should" means recommendation.

Algorithm	*Sender must support*	*Receiver must support*	*Sender should support*	*Receiver should support*
Content-encryption algorithm	Triple DES	Triple DES		1. AES 2. RC2/40
Session-key encryption algorithm	RSA	RSA	Diffie-Hellman	Diffie-Hellman
Hash algorithm	SHA-1	SHA-1		MD5
Digest-encryption algorithm	DSS	DSS	RSA	RSA
Message-authentication algorithm		HMAC with SHA-1		

Application

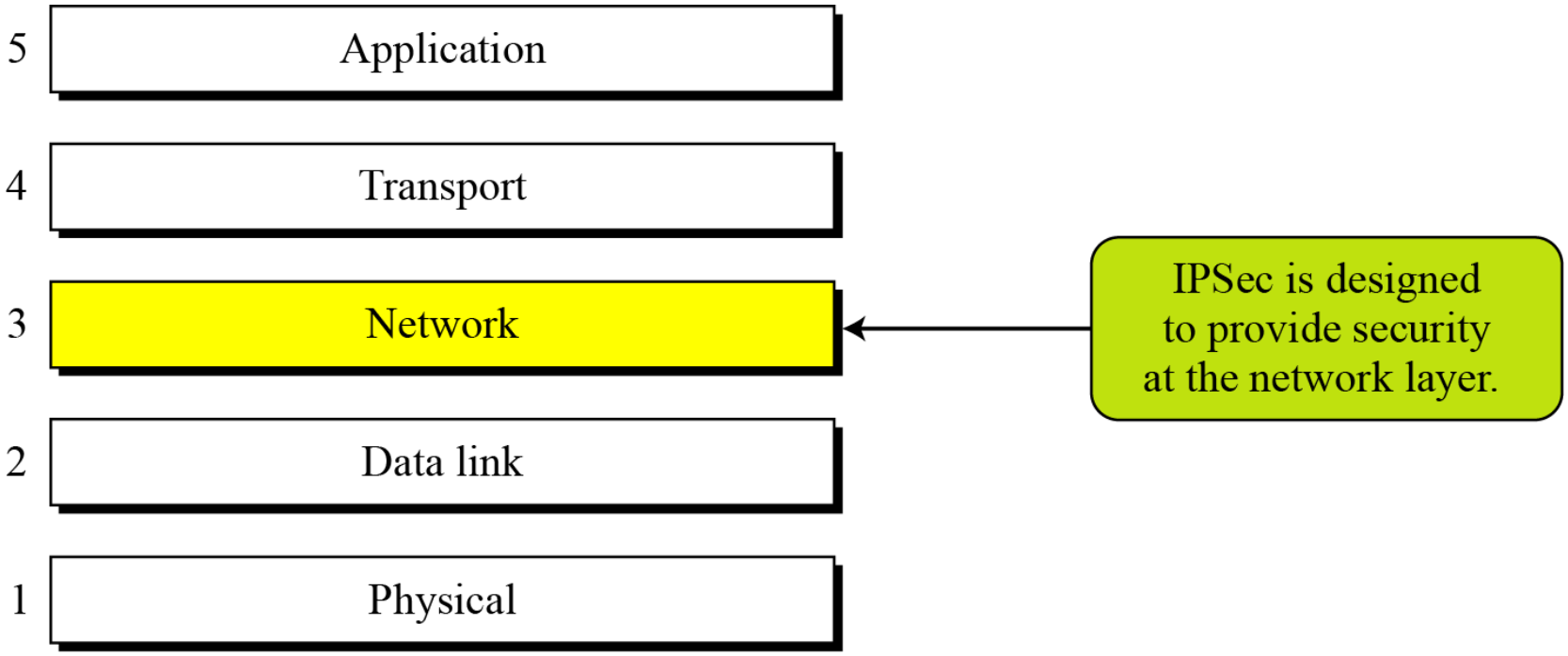
- S/MIME will be used to provide security for commercial e-mail.

Security at the Network Layer: IPSec

Ref: Cryptography and Network Security- Behrouz A Forouzan

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

TCP/IP Protocol Suite and IPSec



TWO MODES

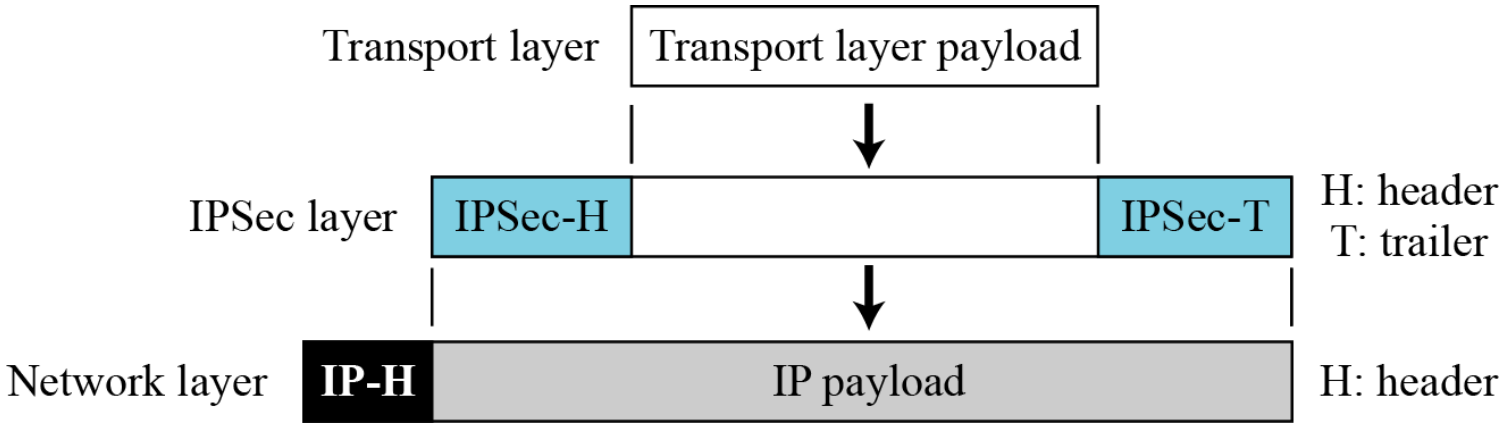
- IPSec operates in one of two different modes: transport mode & tunnel mode.

Transport Mode

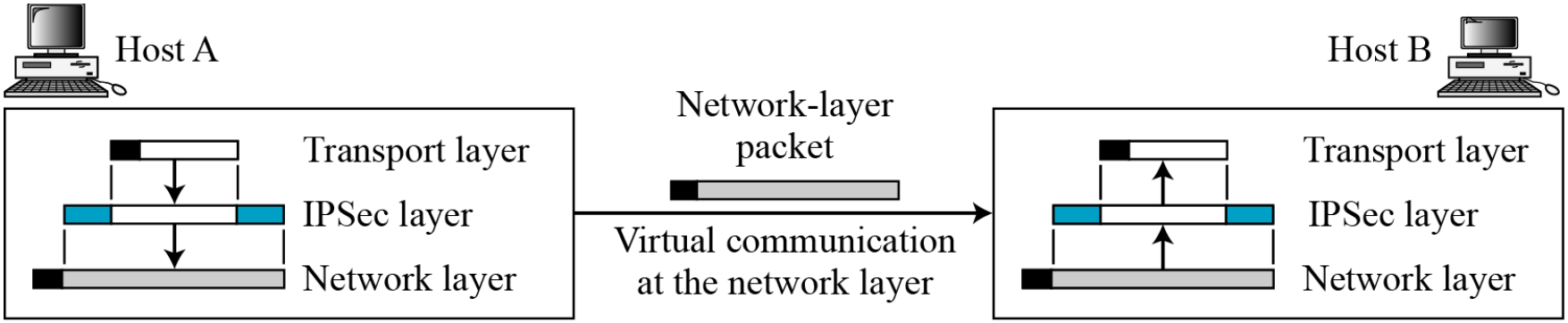
- *In transport mode, IPSec protects what is delivered from the transport layer to the network layer.*

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

- *IPSec in transport mode*



- *Transport mode in action*

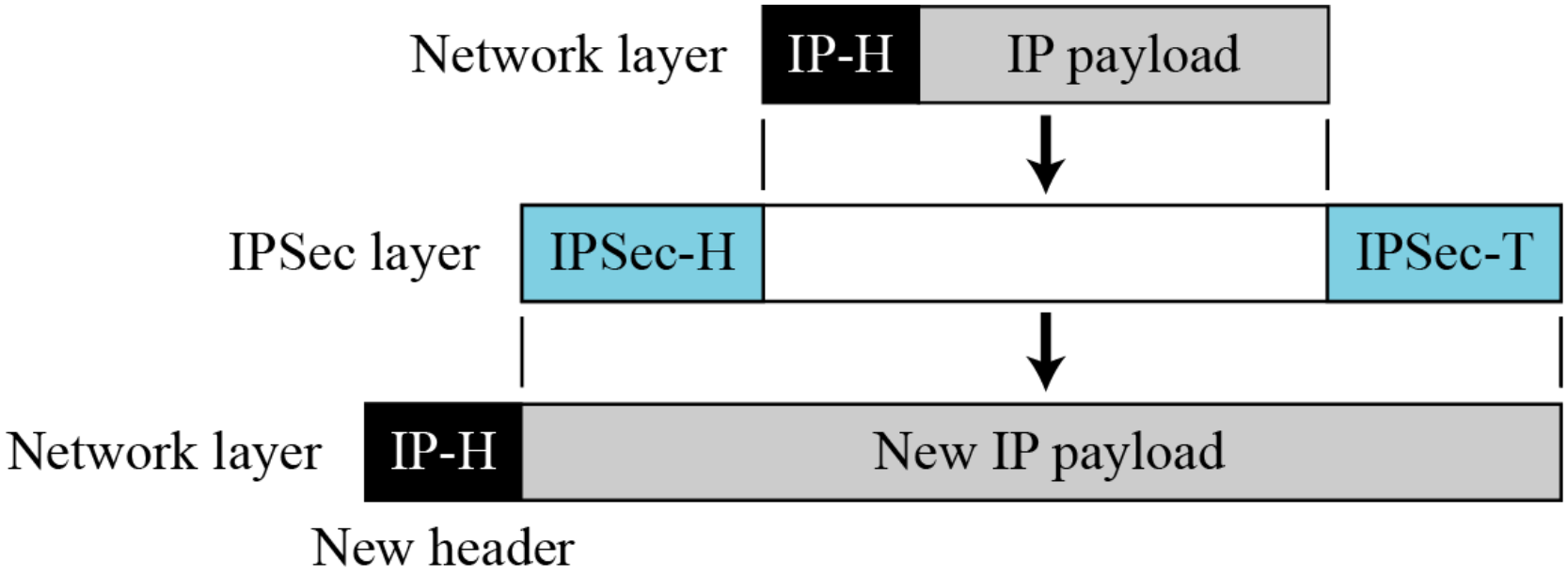


Tunnel Mode

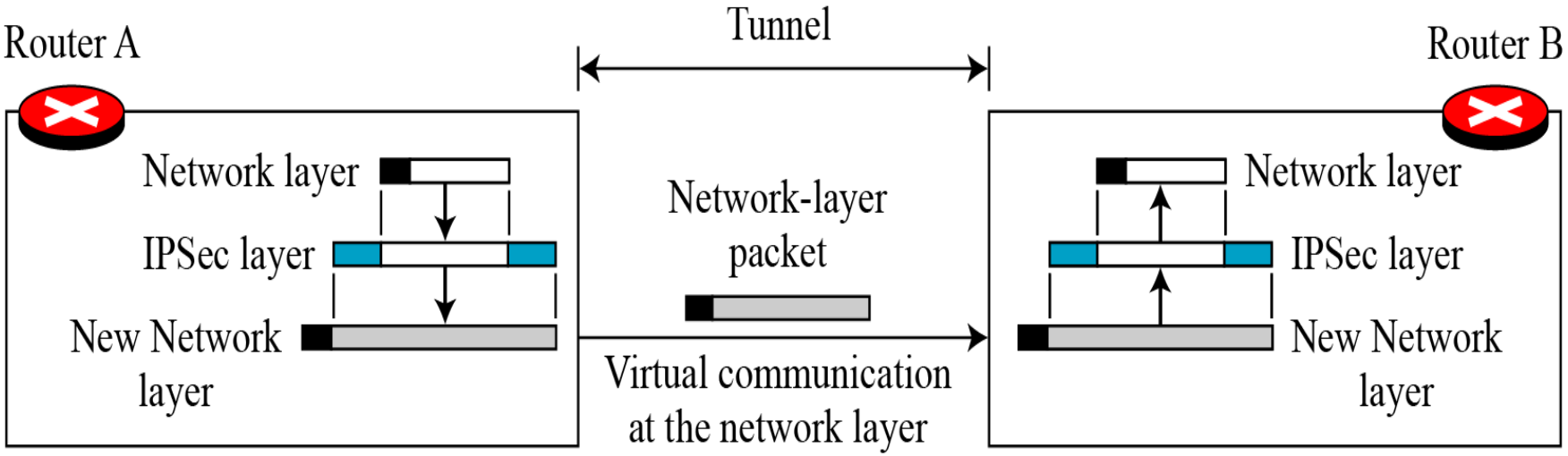
- *In tunnel mode, IPSec protects the entire IP packet. It takes an IP packet, including the header, applies IPSec security methods to the entire packet, and then adds a new IP header.*

IPSec in tunnel mode protects the original IP header.

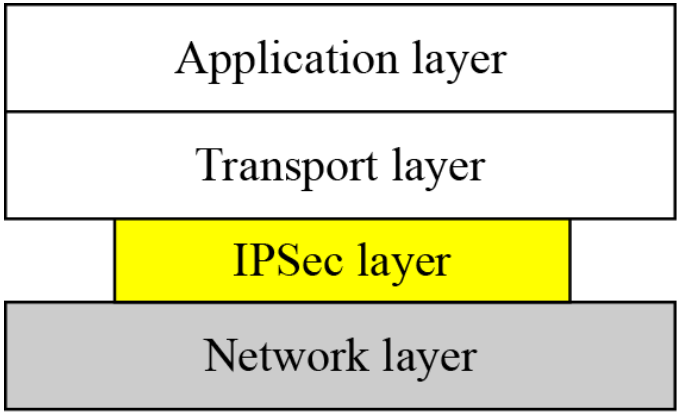
IPSec in tunnel mode



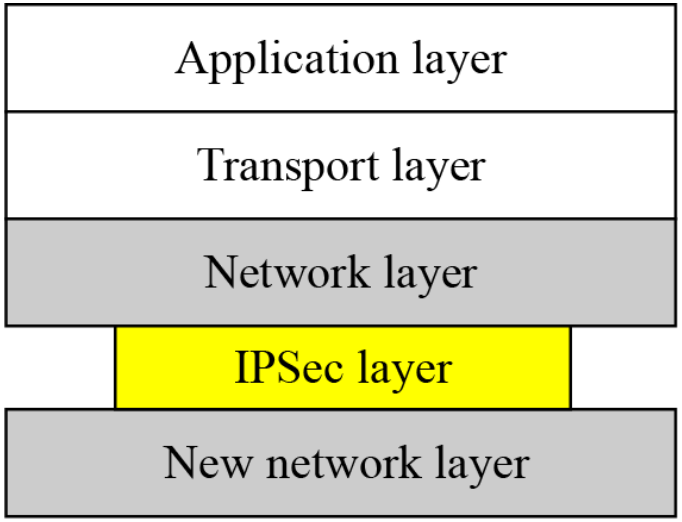
Tunnel mode in action



Transport mode versus tunnel mode



Transport Mode



Tunnel Mode

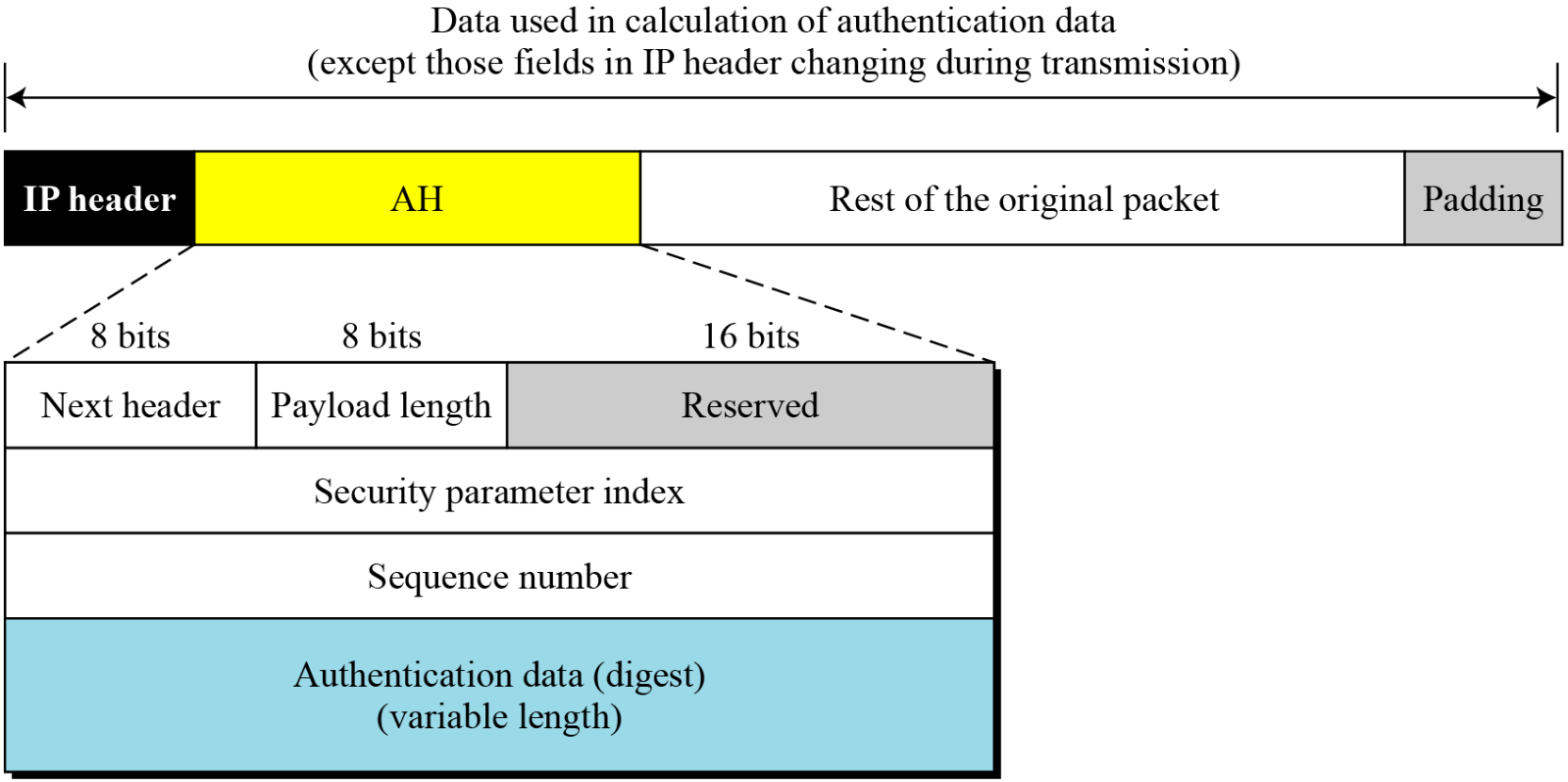
TWO SECURITY PROTOCOL

- IPSec defines two protocols—the Authentication Header (AH) Protocol and the Encapsulating Security Payload (ESP) Protocol to provide authentication and/or encryption for packets at the IP level.

Authentication Header (AH)

- The AH protocol provides source authentication and data integrity, but not privacy.

Authentication Header (AH) protocol



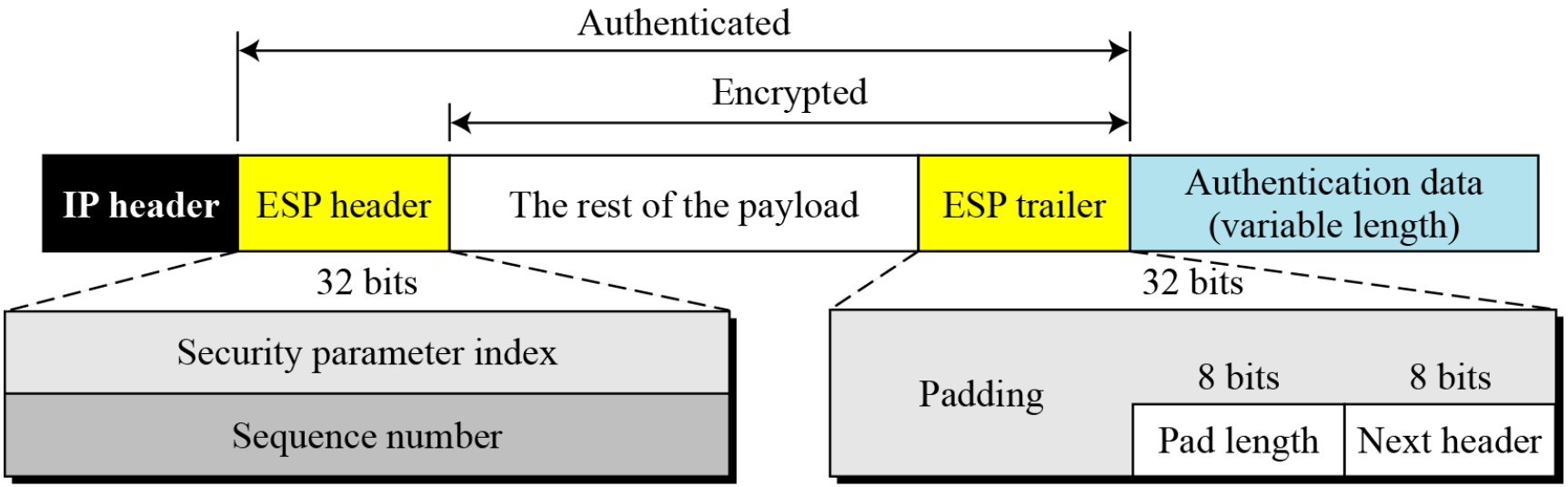
AH Header

- Next Header:defines the type of payload carried by the IP datagram(TCP,UDP,ICMP etc)
- Pay Load Length:defines the length of the authentication header
- Security parameter index: It is a 32 bit number that defines security association at the destination.
- Sequence number:provides ordering information for a sequence of datagrams.
- Authentication data:result of applying hash function to the entire IP datagram.

Encapsulating Security Payload (ESP)

- ESP provides source authentication, data integrity, and privacy.

ESP


Authenticated
Encrypted
IP header
ESP header
The rest of the payload
ESP trailer
Authentication data (variable length)
32 bits
Security parameter index
Sequence number
32 bits
Padding
8 bits
8 bits
Pad length
Next header

IPv4 and IPv6

- *IPSec supports both IPv4 and IPv6. In IPv6, however, AH and ESP are part of the extension header.*

Services Provided by IPSec

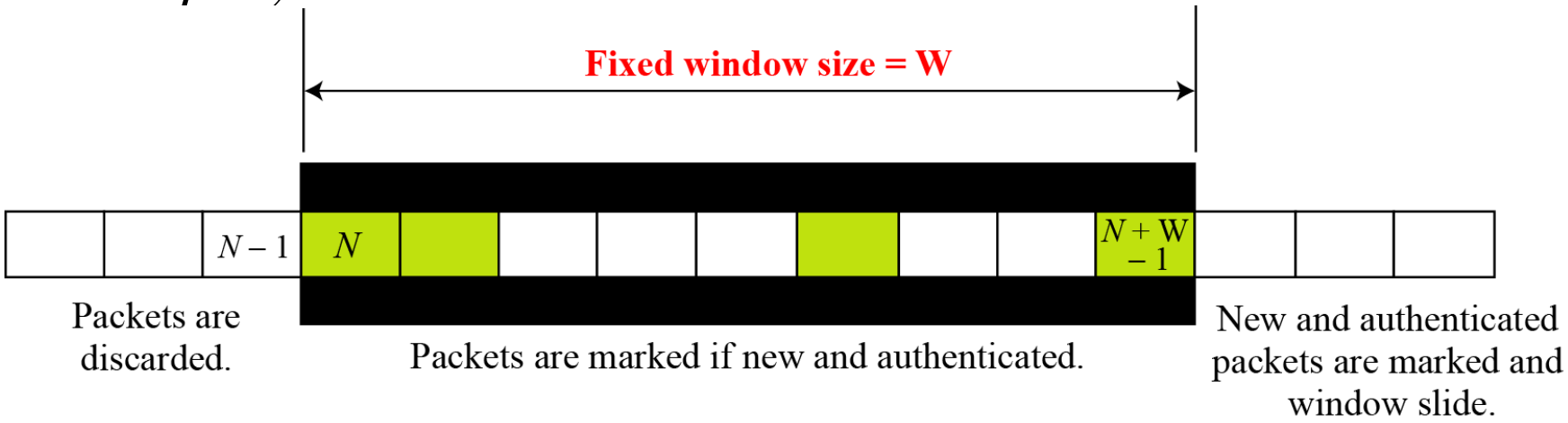
Services	*AH*	*ESP*
Access control	yes	yes
Message authentication (message integrity)	yes	yes
Entity authentication (data source authentication)	yes	yes
Confidentiality	**no**	yes
Replay attack protection	yes	yes

AH versus ESP

- *The ESP protocol was designed after the AH protocol was already in use. ESP does whatever AH does with additional functionality (privacy).*

Replay attack protection

- *Replay window*



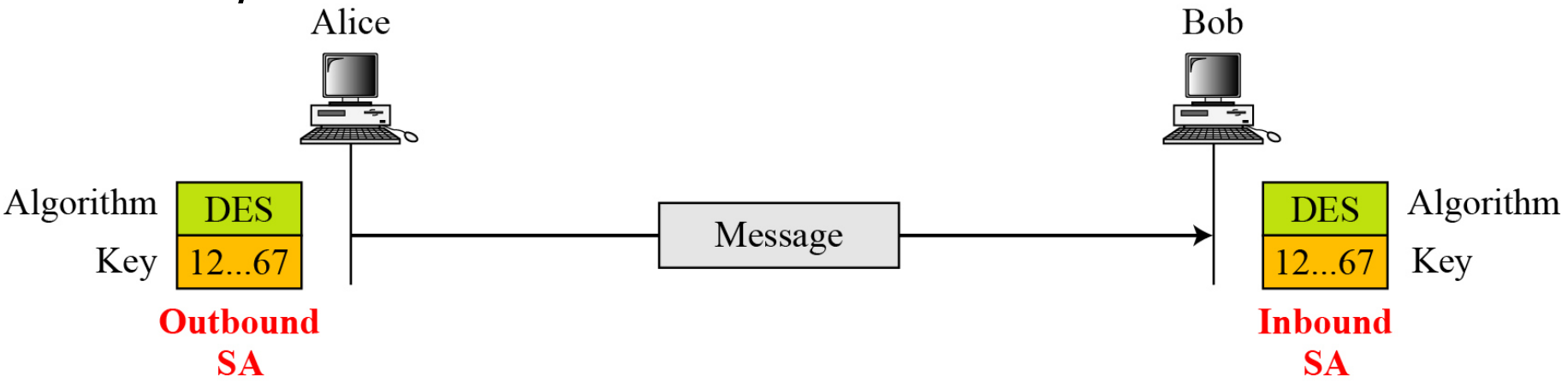
To prevent replay attack sequence number and sliding receiver window is used

SECURITY ASSOCIATION

- *Security Association is a very important aspect of IPSec. IPSec requires a logical relationship, called a Security Association (SA), between two hosts.*

Idea of Security Association

- *Simple SA*



•If alice and bob are interested only in confidentiality aspect of security,they can get a shared secret key between themselves.
•Two security associations:one outbound SA,one inbound SA.
•Each of them stores the value of the key in a variable and the name of encryption/decryption algorithm in another .
•If they are interested in integrity and authentication,they need to store more parameters.

Security Association Database (SAD)

When alice and bob wants to send message to many and receive many they have to maintain a complex database.

Index	SN	OF	ARW	AH/ESP	LT	Mode	MTU
< SPI, DA, P >							
< SPI, DA, P >							
< SPI, DA, P >							
< SPI, DA, P >							

Security Association Database

Legend:

SPI: Security Parameter Index

DA: Destination Address

AH/ESP: Information for either one

P: Protocol

Mode: IPSec Mode Flag

SN: Sequence Number

OF: Overflow Flag

ARW: Anti-Replay Window

LT: Lifetime

MTU: Path MTU (Maximum Transfer Unit)

Typical SA Parameters

Sequence Number Counter	This is a 32-bit value that is used to generate sequence numbers for the AH or ESP header.
Sequence Number Overflow	This is a flag that defines a station's options in the event of a sequence number overflow.
Anti-Replay Window	This detects an inbound replayed AH or ESP packet.
AH Information	This section contains information for the AH protocol: 1. Authentication algorithm 2. Keys 3. Key lifetime 4. Other related parameters
ESP Information	This section contains information for the ESP protocol: 1. Encryption algorithm 2. Authentication algorithm 3. Keys 4. Key lifetime 5. Initiator vectors 6. Other related parameters
SA Lifetime	This defines the lifetime for the SA.
IPSec Mode	This defines the mode, transport or tunnel.
Path MTU	This defines the path MTU (fragmentation).

SECURITY POLICY

- *Another important aspect of IPSec is the Security Policy (SP), which defines the type of security applied to a packet when it is to be sent or when it has arrived.*

Security policy database

Index	Policy
< SA, DA, Name, P, SPort, DPort >	
< SA, DA, Name, P, SPort, DPort >	
< SA, DA, Name, P, SPort, DPort >	
< SA, DA, Name, P, SPort, DPort >	

Legend:

SA: Source Address

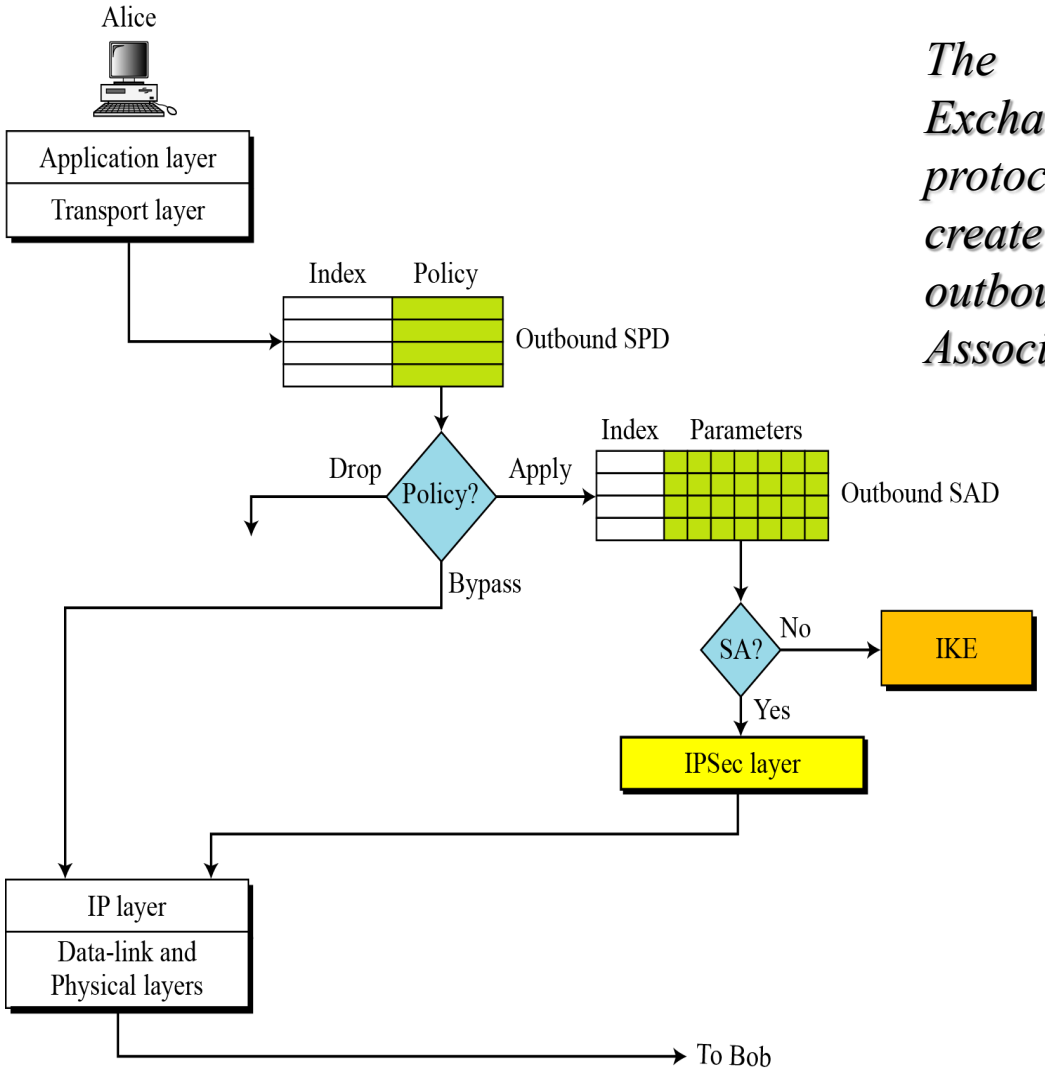
DA: Destination Address

P: Protocol

SPort: Source Port

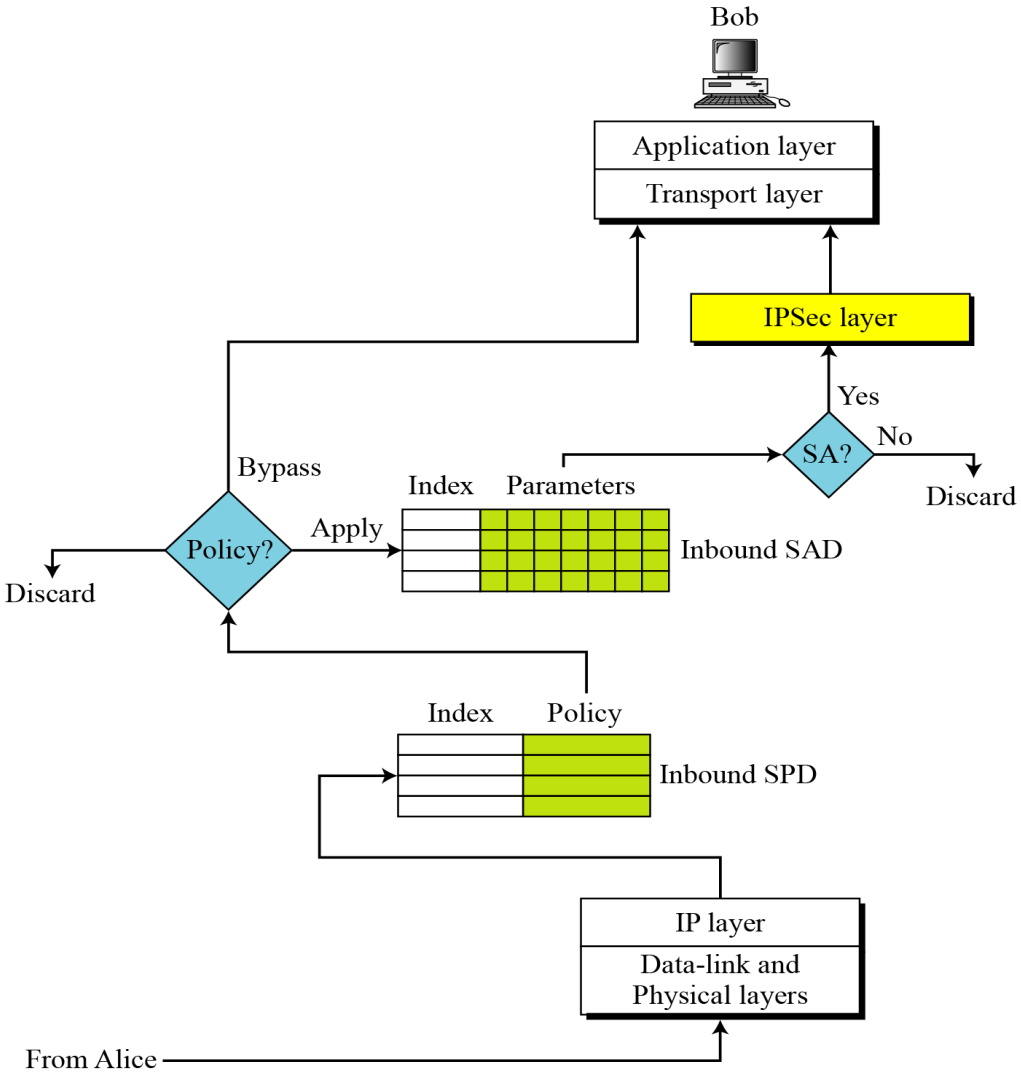
DPort: Destination Port

Outbound processing



The Internet Key Exchange (IKE) is a protocol designed to create both inbound and outbound Security Associations.

Inbound processing



ISAKMP

Ref: Cryptography and Network Security - Behrouz A Forouzan

Course Outcome

- Explain various authentication algorithms for network security

Learning Outcome

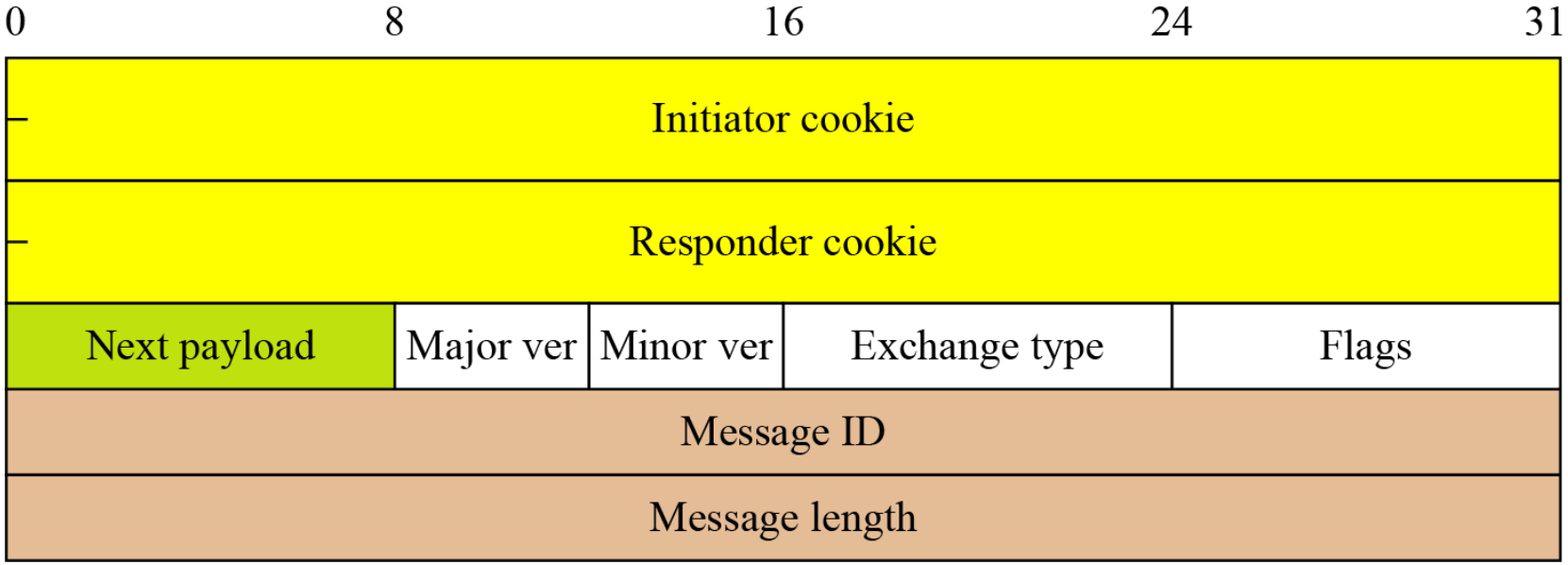
- Explain various cryptographic algorithms used for email security

Specific Outcome

- Explain ISAKMP protocol for IKE exchange

ISAKMP

The ISAKMP protocol is designed to carry messages for the IKE exchange.



General Header

Header Fields

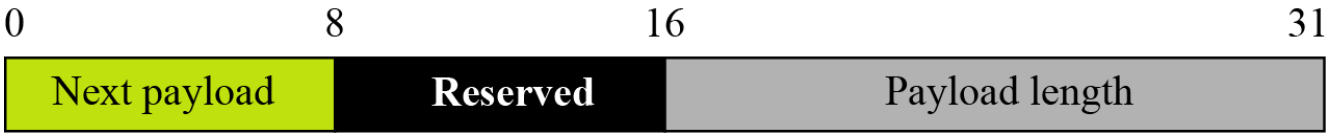
- **Initiator Cookie (32 bits)** – Cookie of entity that initiated SA establishment, notification or deletion.
- **Responder Cookie (32 bits)** – Cookie of the responder
- **Next Payload (1 octet)** – Type of first payload
- **Major/Minor Version (4 bits each)** – Version of ISAKMP in use
- **Exchange Type (1 octet)** – Type of exchange being used

- **Flags (1 octet)** – More stinking flags, encrypt, commit authentication only
- **Message ID (4 octets)** – Unique ID to identify things in Phase 2
- **Length (4 octets)** – Length of total message (headers + payloads)

Payloads

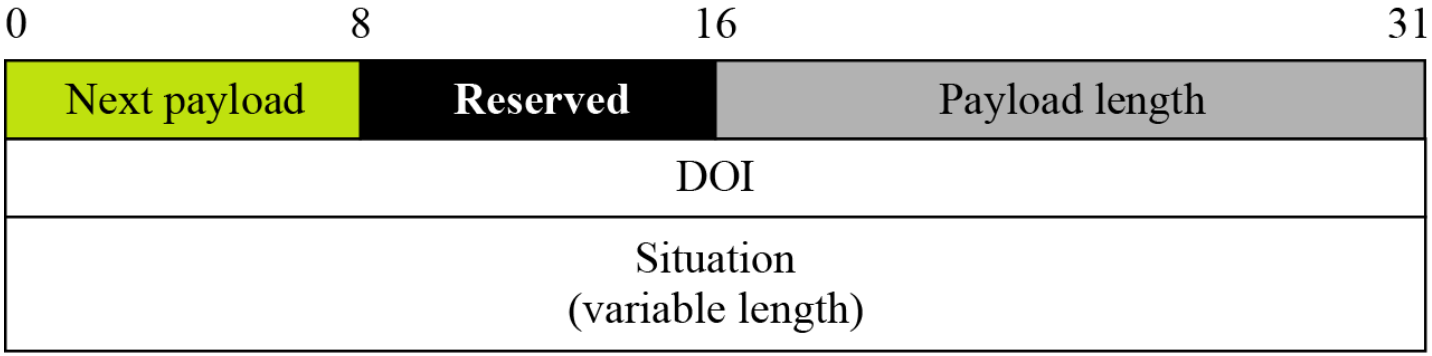
Types	Name	Brief Description
0	None	Used to show the end of the payloads
1	SA	Used for starting the negotiation
2	Proposal	Contains information used during SA negotiation
3	Transform	Defines a security transform to create a secure channel
4	Key Exchange	Carries data used for generating keys
5	Identification	Carries the identification of communication peers
6	Certification	Carries a public-key certificate
7	Certification Request	Used to request a certificate from the other party
8	Hash	Carries data generated by a hash function
9	Signature	Carries data generated by a signature function
10	Nonce	Carries randomly generated data as a nonce
11	Notification	Carries error message or status associated with an SA
12	Delete	Carries one more SA that the sender has deleted
13	Vendor	Defines vendor-specification extensions

Generic payload header

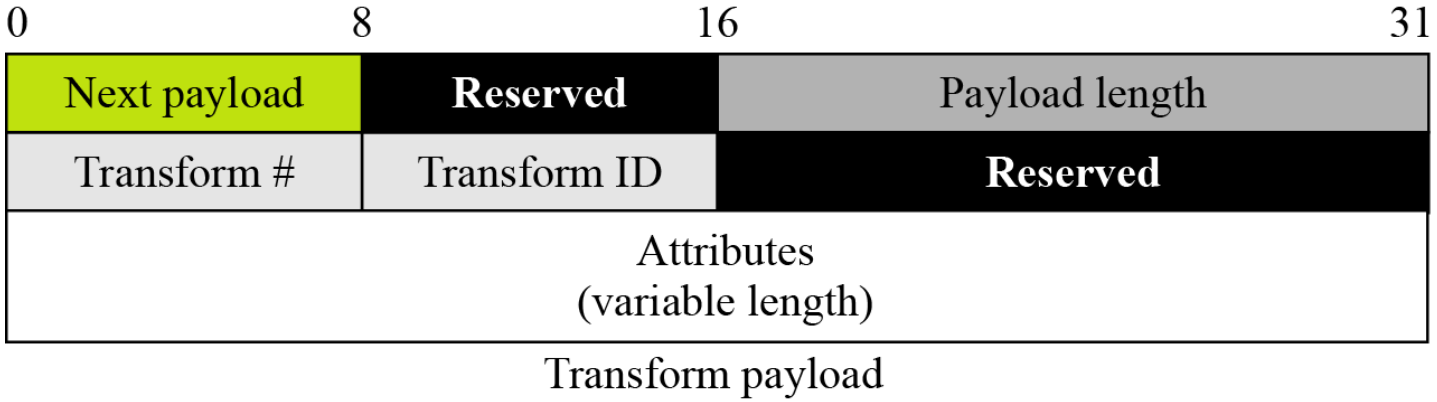


- Each payload has a generic header and some specific fields

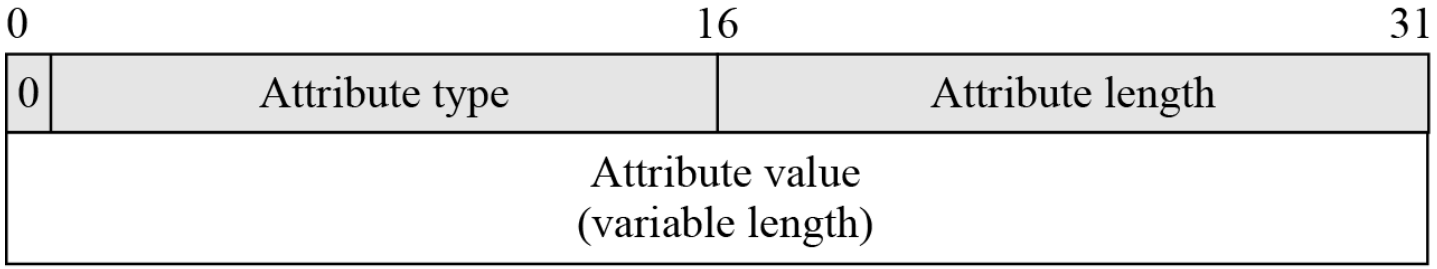
SA payload



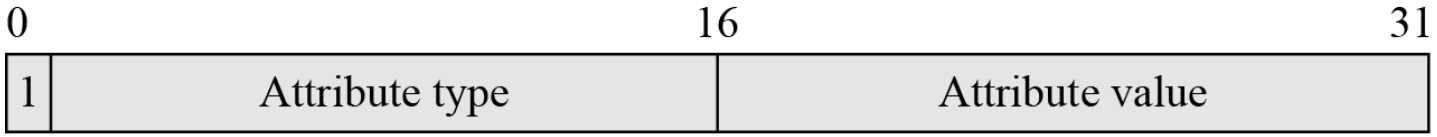
Transform payload



Transform payload



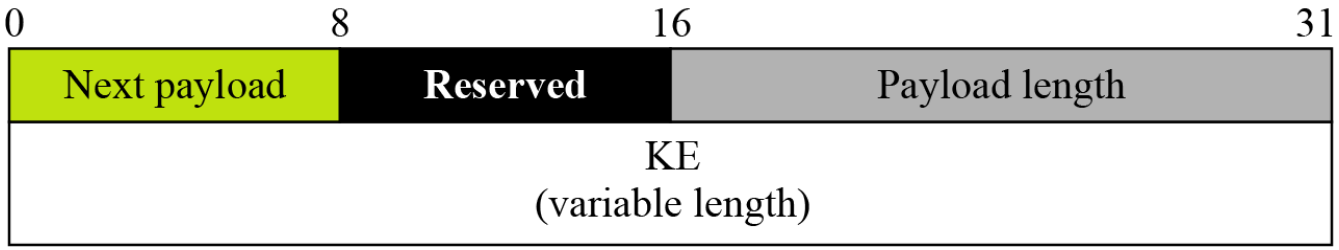
Attribute (long form)



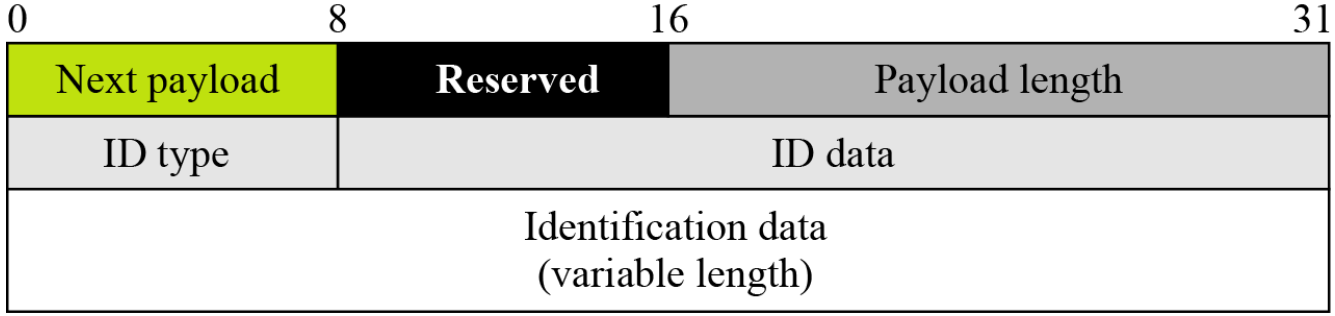
Attribute (short form)

- Payload Length (2 octets) – Length is octets of the current payload, including the generic payload header, Transform values, and all SA attributes
- Transform No. - Identifies the Transform number for the current payload.
- Transform ID – Specifies the Transform identifier from the protocol within the current proposal.
- Reserved 2 (2 octets) – Set to zero.
- SA Attributes (Variable length) – SA attributes should be represented using the Data Attributes format.

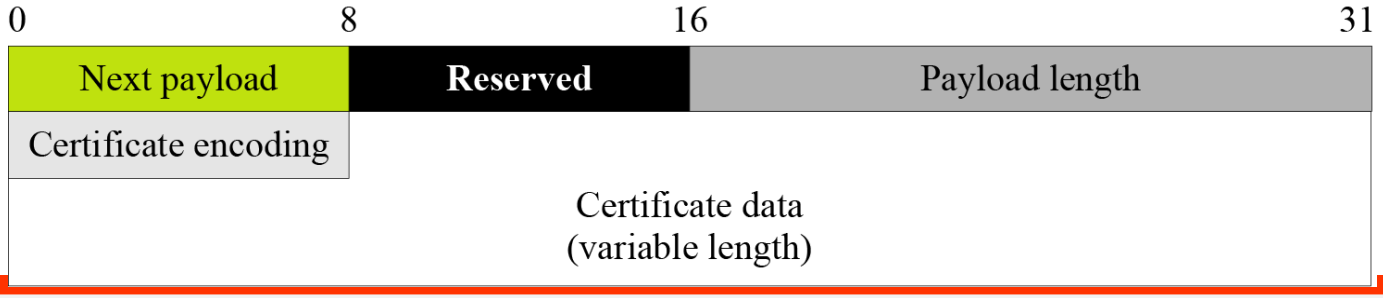
Key-exchange payload



Identification payload



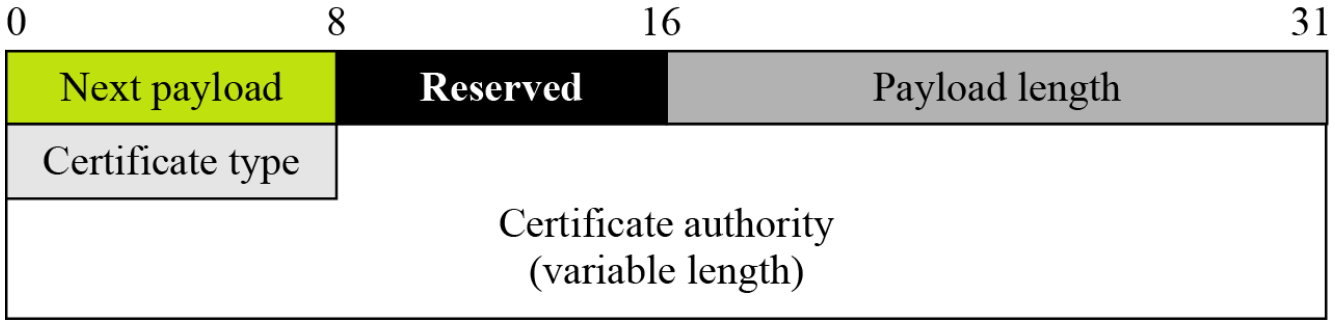
Certification payload



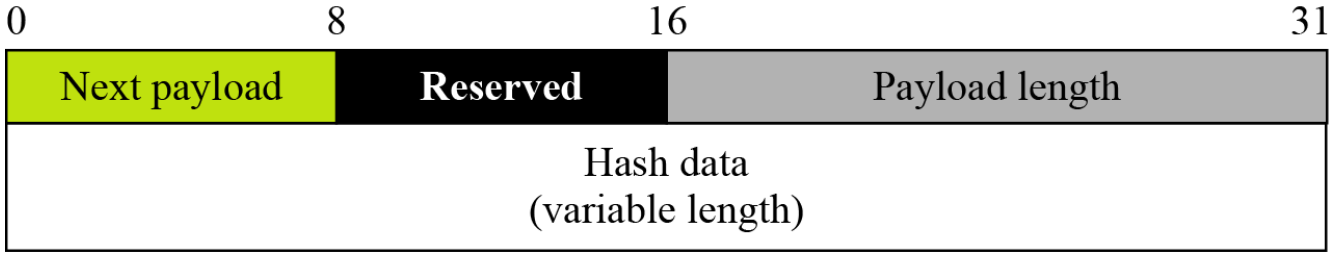
Certification types

Value	*Type*
0	None
1	Wrapped X.509 Certificate
2	PGP Certificate
3	DNS Signed Key
4	X.509 Certificate —Signature
5	X.509 Certificate—Key Exchange
6	Kerberos Tokens
7	Certification Revocation List
8	Authority Revocation List
9	SPKI Certificate
10	X.509 Certificate—Attribute

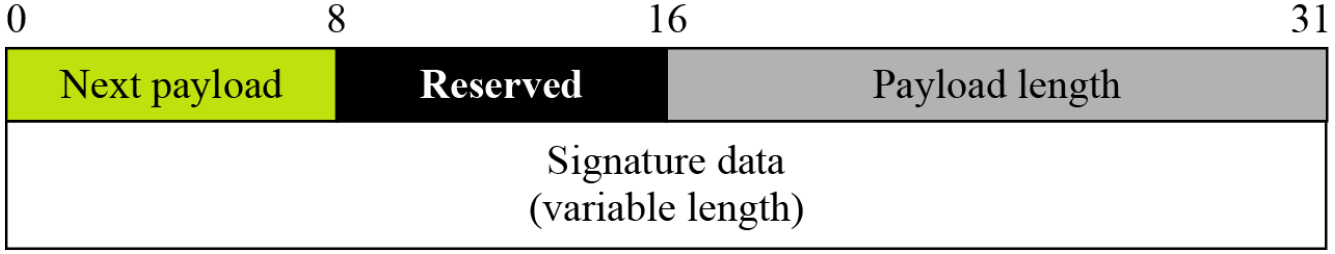
Certification request payload



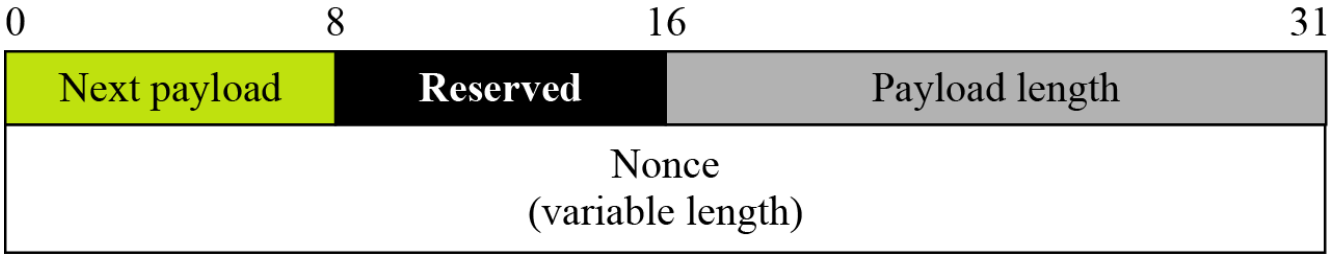
Hash payload



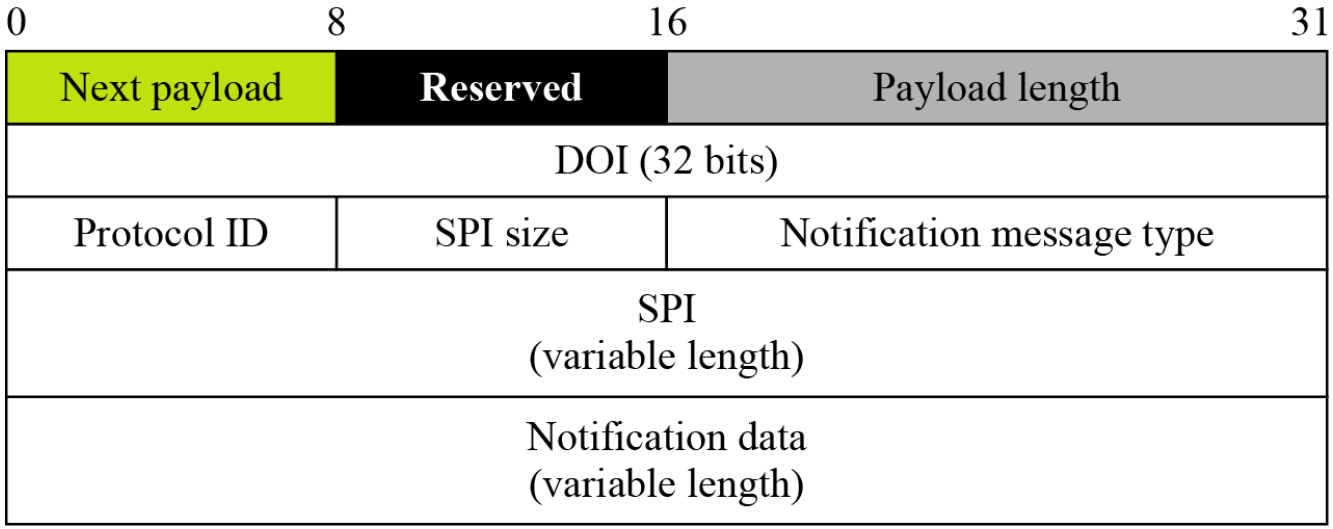
Signature payload



Nonce payload



Notification payload



Notification types

Value	*Description*	*Value*	*Description*
1	INVALID-PAYLOAD-TYPE	8	INVALID-FLAGS
2	DOI-NOT-SUPPORTED	9	INVALID-MESSAGE-ID
3	SITUATION-NOT-SUPPORTED	10	INVALID-PROTOCOL-ID
4	INVALID-COOKIE	11	INVALID-SPI
5	INVALID-MAJOR-VERSION	12	INVALID-TRANSFORM-ID
6	INVALID-MINOR-VERSION	13	ATTRIBUTE-NOT-SUPPORTED
7	INVALID-EXCHANGE-TYPE	14	NO-PROPOSAL-CHOSEN

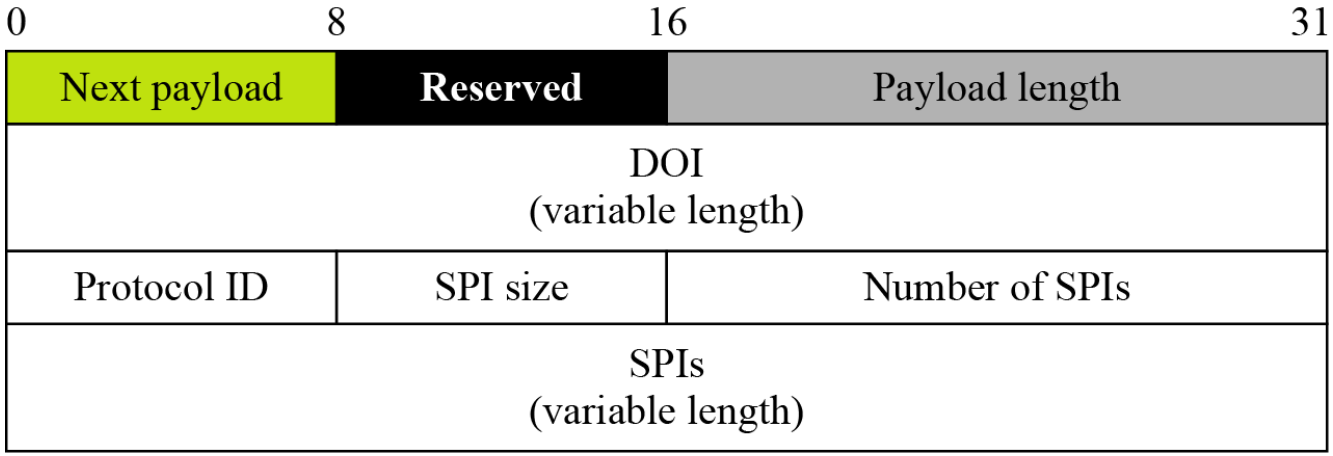
Notification types

Value	*Description*	*Value*	*Description*
15	BAD-PROPOSAL-SYNTAX	23	INVALID-HASH-INFORMATION
16	PAYLOAD-MALFORMED	24	AUTHENTICATION-FAILED
17	INVALID-KEY-INFORMATION	25	INVALID-SIGNATURE
18	INVALID-ID-INFORMATION	26	ADDRESS-NOTIFICATION
19	INVALID-CERT-ENCODING	27	NOTIFY-SA-LIFETIME
20	INVALID-CERTIFICATE	28	CERTIFICATE-UNAVAILABLE
21	CERT-TYPE-UNSUPPORTED	29	UNSUPPORTED EXCHANGE-TYPE
22	INVALID-CERT-AUTHORITY	30	UNEQUAL-PAYLOAD-LENGTHS

Status notification values

Value	*Description*
16384	CONNECTED
24576-32767	DOI-specific codes

Delete payload



Vendor payload