3. MD5, SECURE HASH ALGORITHM

- **Explain about MD5 in detail. (April/May'11, April/May'10 & May/June'12)**
- **Explain Secure Hashing Algorithm (SHA) (April/May'15, Nov/Dec'13, May/June'13 & April/May'10)**

- **Explain the process of deriving eighty 64-bit words from the 1024-bits for processing of a single block and also discuss single round function in SHA-512 algorithm. Show the results of W_{16}, W_{17}, W_{18} and W_{19} (Nov/Dec'14)**

a. MESSAGE DIGEST 5: MD5

- ✓ Developed by Ron Rivest at MIT
- ✓ Input: a message of arbitrary length
- ✓ Output: 128-bit message digest
- ✓ 32-bit word units, 512-bit blocks

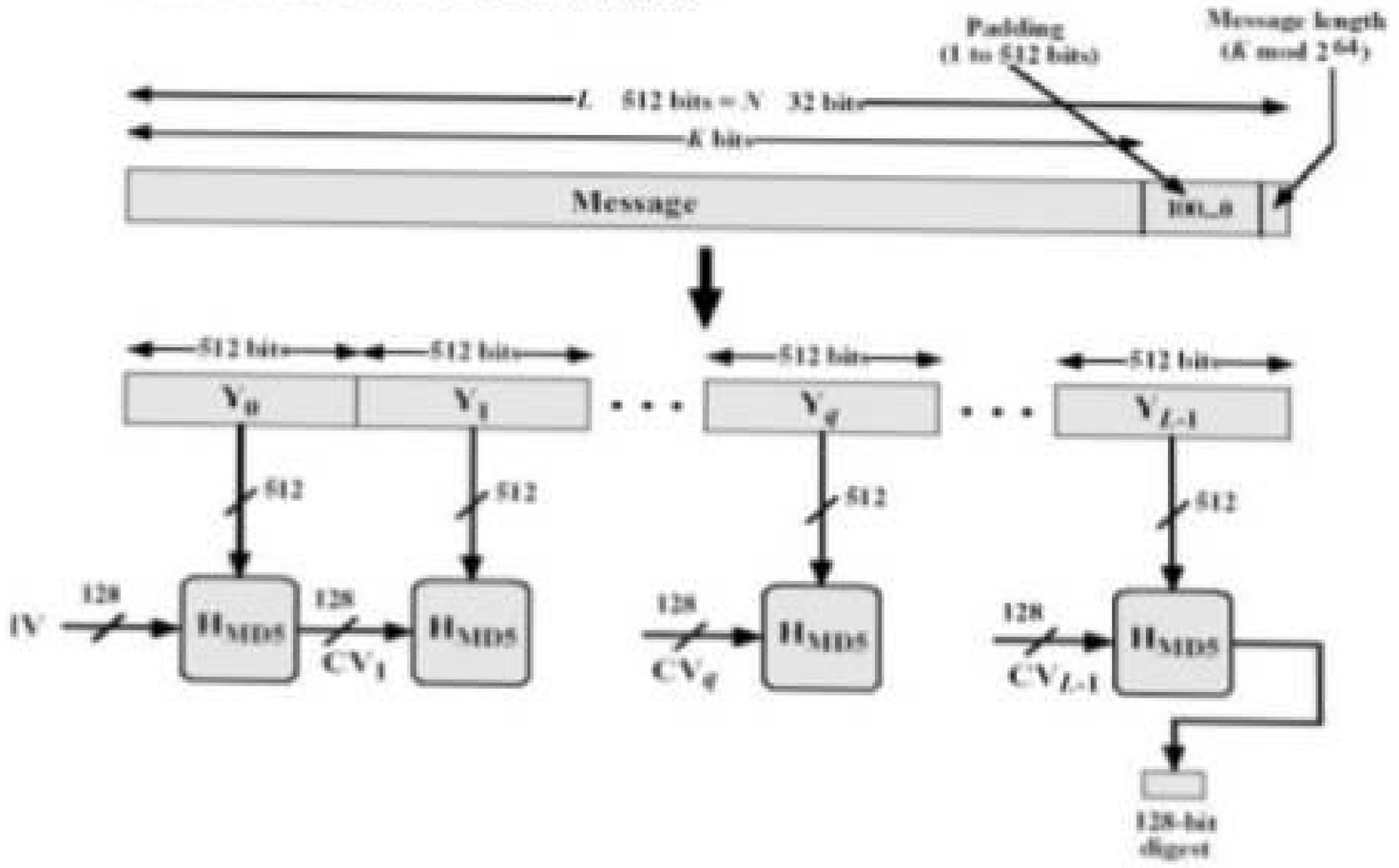


Figure: MD5 Logic

MD5 Logic:

Step 1: Append padding bits

- ✓ The message is Padded so that its bit length ≡ 448 mod 512 (i.e., the length of padded message is 64 bits less than an integer multiple of 512 bits)
- ✓ Padding is always added, even if the message is already of the desired length (1 to 512 bits)
- ✓ Padding bits: 1000....0 (a single 1-bit followed by the necessary number of 0-bits)

Step 2: Append length

- ✓ A 64-bit length: contains the length of the original message modulo 264
- ✓ The expanded message is $Y_0, Y_1, \ldots, Y_{L-1}$; the total length is L × 512 bits
- ✓ The expanded message can be thought of as a multiple of 16 32-bit words
- ✓ Let M[0 ... N-1] denote the word of the resulting message, where N = L × 16

Step 3: Initialize MD buffer

- ✓ 128-bit buffer (four 32-bit registers A,B,C,D) is used to hold intermediate and final results of the hash function
- ✓ A,B,C,D are initialized to the following values

- word D : 76 54 32 10

Step 4: Process message in 512-bit (16-word) blocks

- ✓ Heart of the algorithm called a *compression function* Consists of 4 rounds
- ✓ The 4 rounds have a similar structure, but each uses a different *primitive logical functions*, referred to as F, G, H, and I
- ✓ Each round takes as input the current 512-bit block (Yq), 128-bit buffer value ABCD and updates the contents of the buffer
- ✓ Each round also uses the table T[1 … 64], constructed from the sine function;
  - T[i] = 232 × abs(sin(i))
- ✓ The output of 4th round is added to the CVq to produce CVq+1

Step 5: Output

- ✓ After all L 512-bit blocks have been processed, the output from the Lth stage is the 128-bit message digest

$$CV_0 = IV$$

$$CVq+1 = SUM32(CVq, RFI[Yq, RFH[Yq, RFG[Yq, RFF[Yq, CVq]]])$$

$$MD = CV_L$$

Where IV = initial value of the ABCD buffer, defined in step 3

Yq = the qth 512-bit block of the message

L = the number of blocks in the message (including padding and length fields) CVq = chaining variable processed with the qth block of the message

RFx = round function using primitive logical function x

MD = final message digest value

SUM32 = addition modulo 232 performed separately on each word

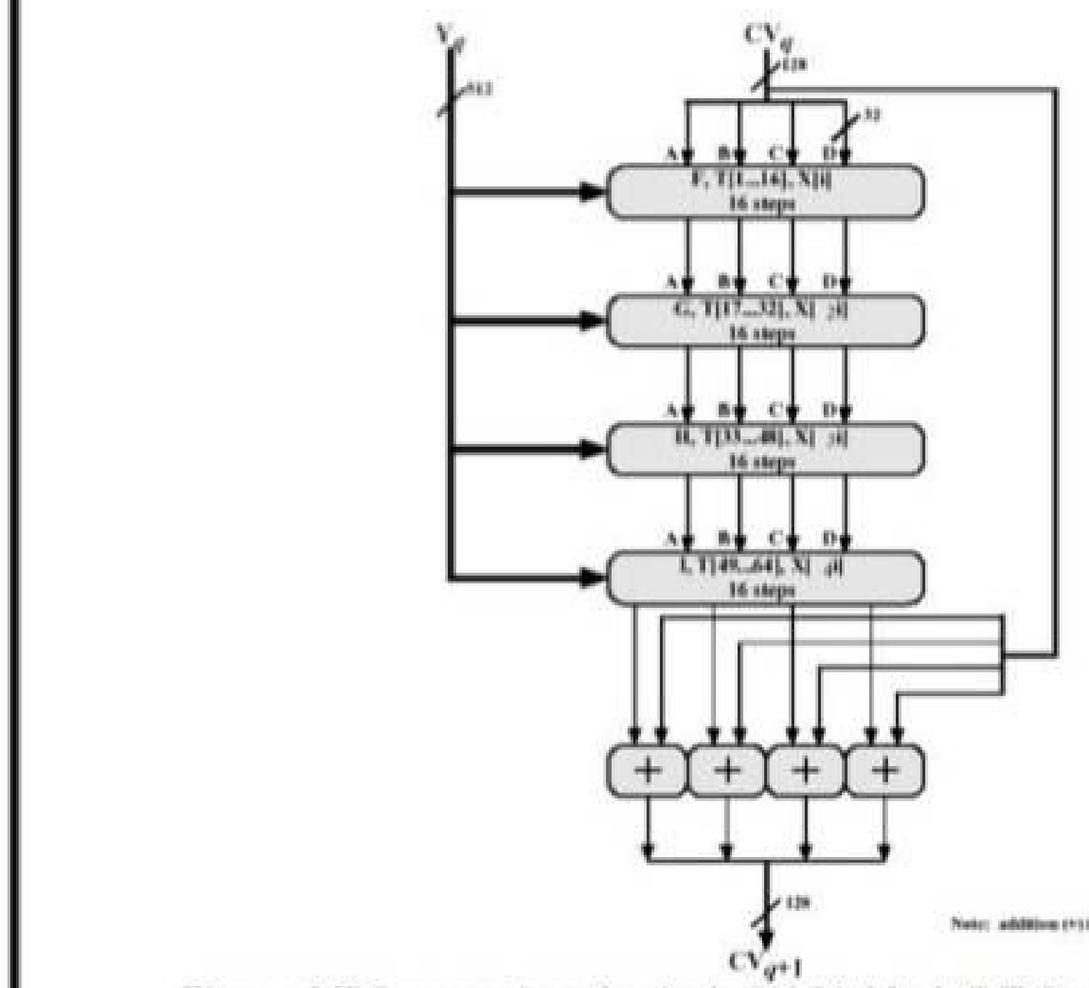


Figure: MD5 processing of a single 512-bit block (MD5 compression function)

MD5 Compression Function:

- ✓ Each round consists of a sequence of 16 steps operating on the buffer ABCD
- ✓ Each step is of the form

$$a \leftarrow b + ((a + g(b, c, d) + X[k] + T[i] <<< s)$$

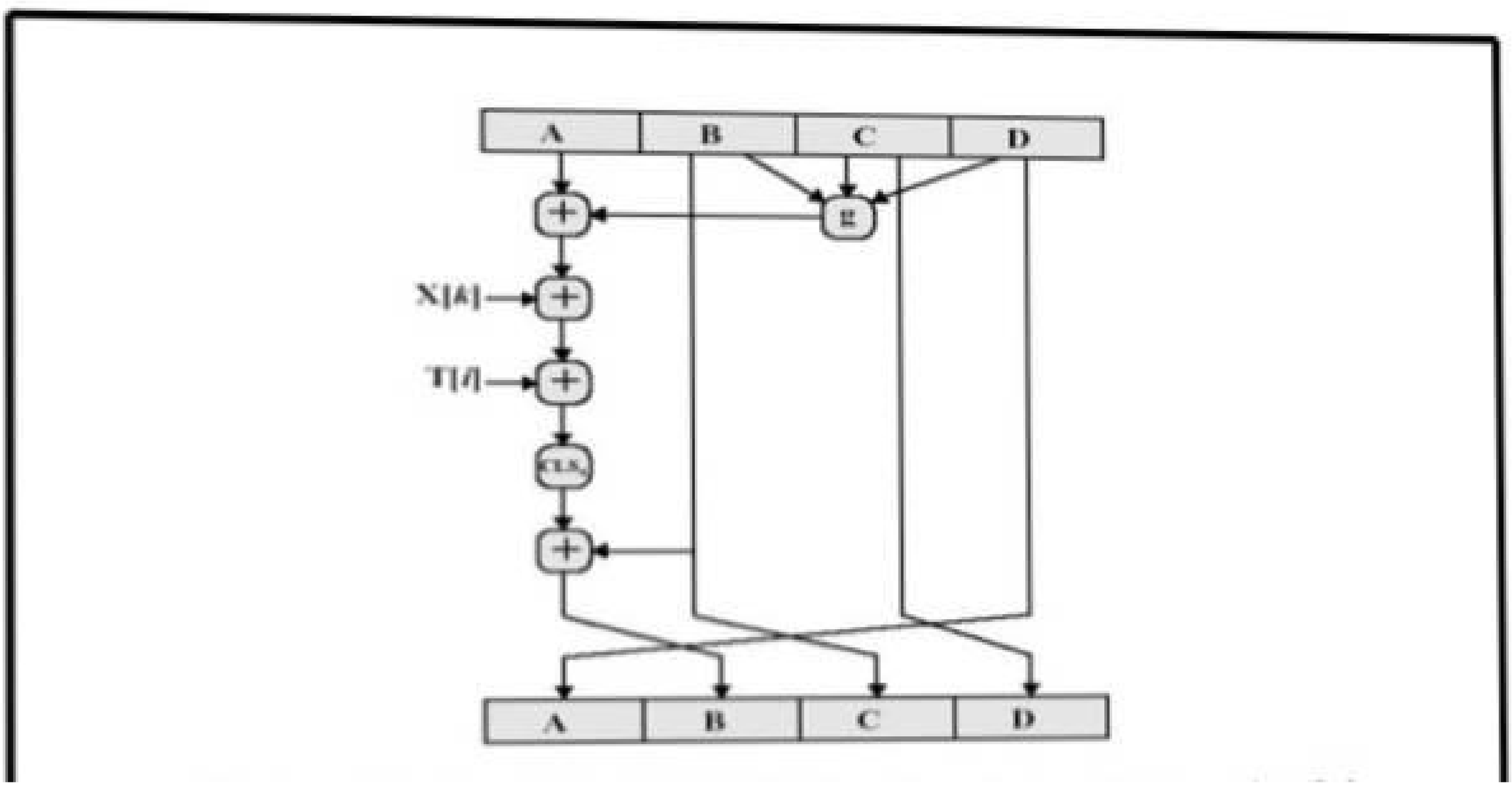
where

a,b,c,d = the 4 words of the buffer, in a specified order that varies across steps

g = one of the primitive functions F, G, H, I

<<s = circular left shift (rotation) of the 32-bit arguments by s bits

X[k] = M[q × 16 + k] = the kth 32-bit word in the qth 512-bit block of the



5. DIGITAL SIGNATURE ALGORITHM

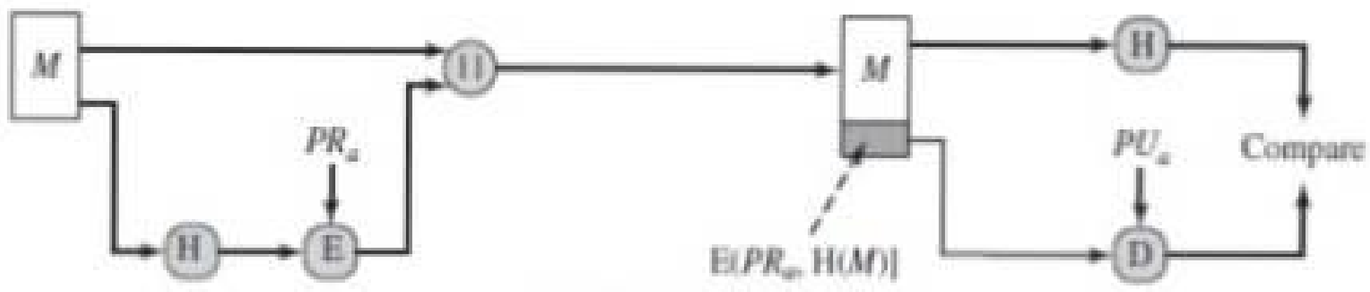
- **Explain Digital Signature Standard. (May/June'14)**
- **Give the details of digital signature algorithm. (May/June'07)**
- **With a neat sketch, explain signing and verifying functions of DSA. (May/June'12)**

NIST DIGITAL SIGNATURE ALGORITHM

- The National Institute of Standards and Technology (NIST) has published Federal Information Processing Standard FIPS 186, known as the Digital Signature Algorithm (DSA).
- The DSA makes use of the Secure Hash Algorithm (SHA).
- The DSA was originally proposed in 1991 and revised in 1993, 1996 and then 2000 an expanded version of the standard was issued as FIPS 186-2, subsequently updated to FIPS 186-3 in 2009.
- The DSA uses an algorithm that is designed to provide only the digital signature function. Unlike RSA, it cannot be used for encryption or key exchange. Nevertheless, it is a public-key technique.

The RSA Approach:

- In the RSA approach, the message to be signed is input to a hash function that produces a secure hash code of fixed length.
- This hash code is then encrypted using the sender's private key to form the signature.
- Both the message and the signature are then transmitted. The recipient takes the message and produces a hash code.
- The recipient also decrypts the signature using the sender's public key.
- If the calculated hash code matches the decrypted signature, the signature is accepted as valid. Because only the sender knows the private key, only the sender could have produced a valid signature.

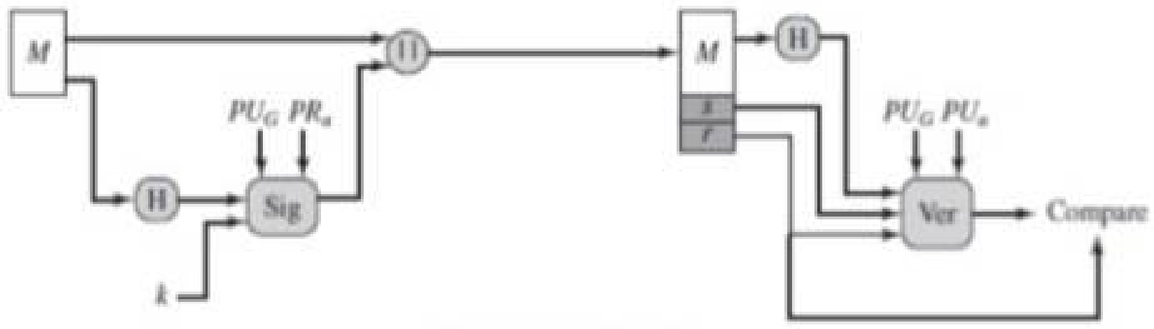


(a) RSA approach

The DSA Approach:

- The DSA approach also makes use of a hash function. The hash code is provided as input to a signature function along with a random number k generated for this particular signature.
- The signature function also depends on the sender's private key (PR_a) and a set of parameters known to a group of communicating principals. We can consider this set to constitute a global public key (PU_G). The result is a signature consisting of two components, labeled s and r.
- At the receiving end, the hash code of the incoming message is generated. This plus the signature is input to a verification function.
- The verification function also depends on the global public key as well as the sender's public key (PU_a), which is paired with the sender's private key.

- The output of the verification function is a value that is equal to the signature component r if the signature is valid.
- The signature function is such that only the sender, with knowledge of the private key, could have produced the valid signature.



(b) DSA approach

The Digital Signature Algorithm

Global Public-Key Components

p prime number where $2^{L-1} < p < 2^L$ for $512 \le L \le 1024$ and L a multiple of 64; i.e., bit length of between 512 and 1024 bits in increments of 64 bits

q prime divisor of $(p - 1)$, where $2^{N-1} < q < 2^N$ i.e., bit length of N bits

g $= h^{(p-1)/q} \bmod p$, where h is any integer with $1 < h < (p - 1)$ such that $h^{(p-1)/q} \bmod p > 1$

User's Private Key

x random or pseudorandom integer with $0 < x < q$

User's Public Key

$y = g^x \bmod p$

User's Per-Message Secret Number

k random or pseudorandom integer with $0 < k < q$

Signing

$r = (g^k \bmod p) \bmod q$

$s = [k^{-1} (H(M) + xr)] \bmod q$

Signature $= (r, s)$

Verifying

$w = (s')^{-1} \bmod q$

$u_1 = [H(M')w] \bmod q$

$u_2 = (r')w \bmod q$

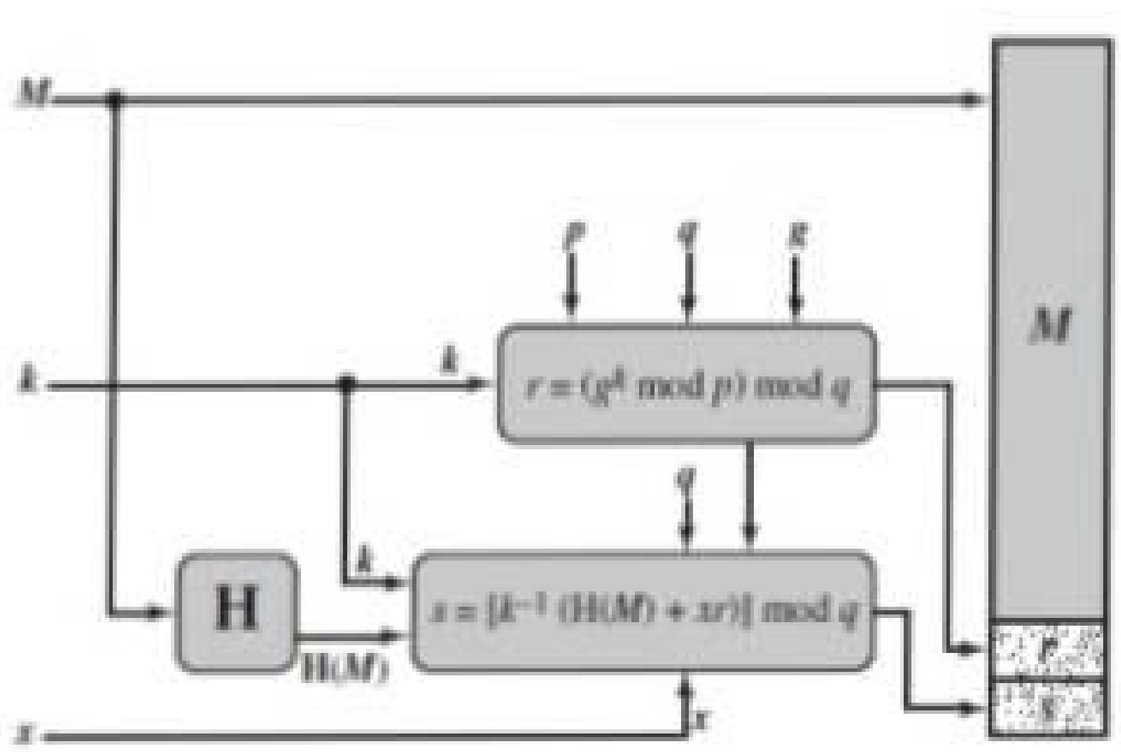
$v = [(g^{u1} y^{u2}) \bmod p] \bmod q$

TEST: $v = r'$

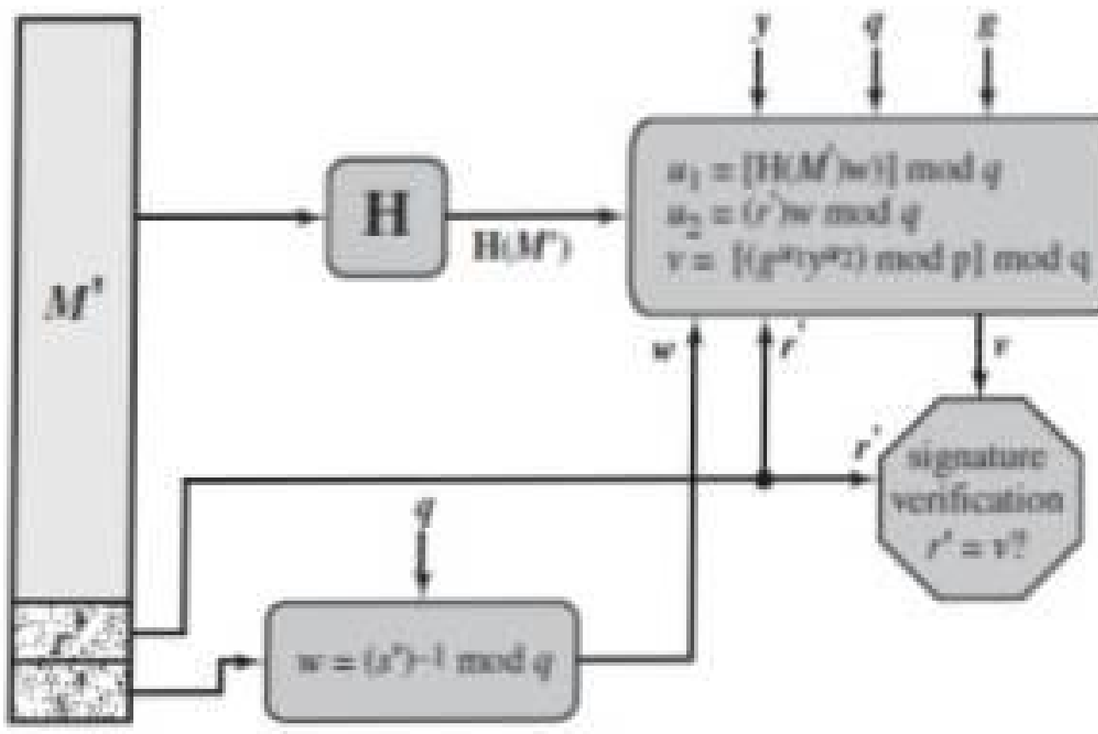
M = message to be signed

H(M) = hash of M using SHA-1

M', r', s' = received versions of M, r, s



(a) Signing



(b) Verifying

Figure: DSA Signing and Verifying

4. FIREWALL

❖ **Explain in detail about Firewall design principles explain types of firewalls in detail. (April/May 2011, May/June 2011, Nov/Dec 2011)**

Firewall Design Principles

- Centralized data processing system, with a central mainframe supporting a number of directly connected terminals
- Local area networks (LANs) interconnecting PCs and terminals to each other and the mainframe
- Premises network, consisting of a number of LANs, interconnecting PCs, servers, and perhaps a mainframe or two
- Enterprise-wide network, consisting of multiple, geographically distributed premises networks interconnected by a private wide area network (WAN)
- Internet connectivity, in which the various premises networks all hook into the Internet and may or may not also be connected by a private WAN

Firewall Characteristics

1. All traffic from inside to outside, and vice versa, must pass through the firewall. This is achieved by physically blocking all access to the local network except via the firewall. Various configurations are possible, as explained later in this section.
2. Only authorized traffic, as defined by the local security policy, will be allowed to pass. Various types of firewalls are used, which implement various types of security policies, as explained later in this section.
3. The firewall itself is immune to penetration. This implies that use of a trusted system with a secure operating system.

Firewalls focused primarily on service control, but they have since evolved to provide all four:

- **Service control:** Determines the types of Internet services that can be accessed, inbound or outbound.
- **Direction control**: Determines the direction in which particular service requests may be initiated and allowed to flow through the firewall.
- **User control**: Controls access to a service according to which user is attempting to access it.
- **Behavior control**: Controls how particular services are used. For example, the firewall may filter e-mail to eliminate spam, or it may enable external access to only a portion of the information on a local Web server.

Capabilities are within the scope of a firewall:

1. A firewall defines a single choke point that keeps unauthorized users out of the protected network, prohibits potentially vulnerable services from entering or leaving the network, and provides protection from various kinds of IP spoofing and routing attacks.
2. A firewall provides a location for monitoring security-related events.
3. A firewall is a convenient platform for several Internet functions that are not security related.
4. A firewall can serve as the platform for IPSec.

Firewalls have their limitations, including the following:

1. The firewall cannot protect against attacks that bypass the firewall. Internal systems may have dial-out capability to connect to an ISP.
2. The firewall does not protect against internal threats, such as a disgruntled employee or an employee who unwittingly cooperates with an external attacker.
3. The firewall cannot protect against the transfer of virus-infected programs or files.

Types of Firewalls

Packet-Filtering Router

✓ A packet-filtering router applies a set of rules to each incoming and outgoing IP

- **Source and destination transport-level address:** The transport level (e.g., TCP or UDP) port number, which defines applications such as SNMP or TELNET
- **IP protocol field:** Defines the transport protocol
- **Interface:** For a router with three or more ports, which interface of the router the packet came from or which interface of the router the packet is destined for.

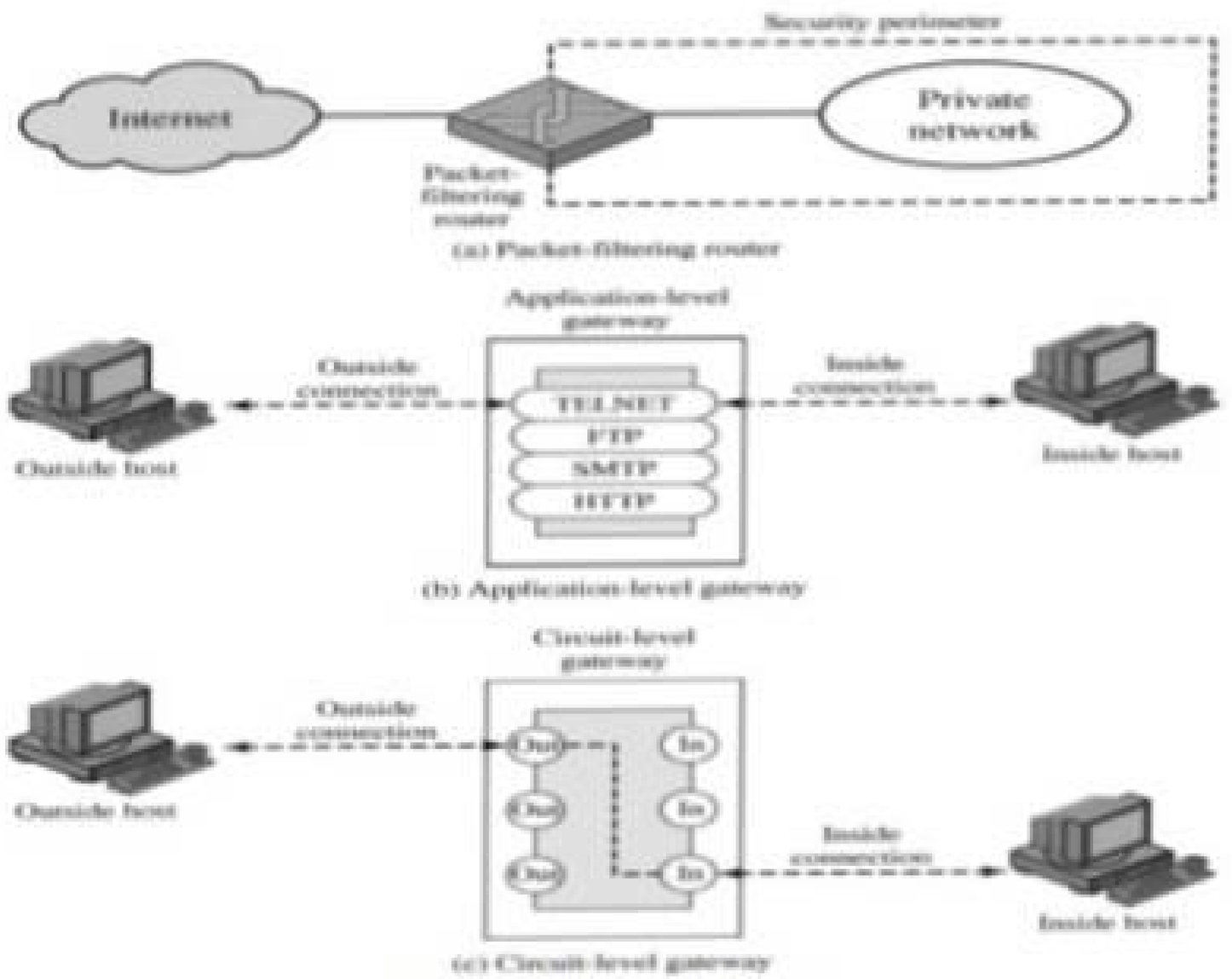


Figure: Firewall Types

Two default policies are possible:

- **Default = discard**: That which is not expressly permitted is prohibited.
- **Default = forward**: That which is not expressly prohibited is permitted.

Some of the attacks that can be made on packet-filtering routers and the appropriate countermeasures are thefollowing:

- **IP address spoofing**: The intruder transmits packets from the outside with a source IP address field containing an address of an internal host.
- **Source routing attacks**: The source station specifies the route that a packet should take as it crosses the Internet, in the hopes that this will bypass security measures that do not analyze the source routing information. The countermeasure is to discard all packets that use this option.

Secure Electronic Transaction:

Secure Electronic Transaction (SET) is an open encryption and security specification designed to protect credit card transactions on the Internet.

- Confidentiality of information
- Integrity of data
- Cardholder account authentication
- Merchant authentication

SET is an open encryption and security specification designed to protect credit card transactions on the Internet. The current version, SETv1, emerged from a call for security standards by MasterCard and Visa in February 1996.

SET is not itself a payment system. Rather it is a set of security protocols and formats that enables users to employ the existing credit card payment infrastructure on an open network (Internet) in a secure fashion.

SET services:

- Provides a secure communications channel among all parties involved in a transaction.
- Provides trust by the use of X.509v3 digital certificates
- Ensures privacy because the information is only available to parties in a transaction when and where necessary.

SET Overview:

Requirements:

Business requirements for secure payment processing with credit cards over the Internet and other networks:

- Provide confidentiality of payment and ordering information: It is necessary to assure cardholders that this information is safe and accessible only to the intended recipient.
- Ensure the integrity of all transmitted data: That is, ensure that no changes in content occur during transmission of SET messages. Digital signatures are used to provide integrity.
- Provide authentication that a cardholder is a legitimate user of a credit card account. Digital signatures and certificates are used to verify that a cardholder is a legitimate user of a valid account.
- Provide authentication that a merchant can accept credit card transactions through its relationship with a financial institution.
- Ensure the use of the best security practices and system design techniques to protect all legitimate parties in an electronic commerce transaction.
- Create a protocol that neither depends on transport security mechanisms nor prevents their use.
- Facilitate and encourage interoperability among software and network providers.

Key Features of SET

SET incorporates the following features:

- Confidentiality of information.
- Integrity of data.
- Cardholder account authentication

- Provide confidentiality of payment and ordering information: It is necessary to assure cardholders that this information is safe and accessible only to the intended recipient.

- Ensure the integrity of all transmitted data: That is, ensure that no changes in content occur during transmission of SET messages. Digital signatures are used to provide integrity.

- Provide authentication that a cardholder is a legitimate user of a credit card account. Digital signatures and certificates are used to verify that a cardholder is a legitimate user of a valid account.

- Provide authentication that a merchant can accept credit card transactions through its relationship with a financial institution.

- Ensure the use of the best security practices and system design techniques to protect all legitimate parties in an electronic commerce transaction.

- Create a protocol that neither depends on transport security mechanisms nor prevents their use.

- Facilitate and encourage interoperability among software and network providers.

Key Features of SET

SET incorporates the following features:

- Confidentiality of information.
- Integrity of data.
- Cardholder account authentication
- Merchant authentication

SET Participants:

Cardholder:

In the electronic environment, consumers and corporate purchasers interact with merchants from personal computers over the Internet. A cardholder is an authorized holder of a payment card (e.g., MasterCard, Visa) that has been issued by an issuer.

Merchant:

A merchant is a person or organization that has goods or services to sell to the cardholder.

Issuer:

This is a financial institution, such as a bank, that provides the cardholder with the payment card.

Acquirer:

This is a financial institution that establishes an account with a merchant and processes payment card authorizations and payments. The acquirer also provides electronic transfer of payments to the merchant's account.

Payment gateway:

This is a function operated by the acquirer or a designated third party that processes merchant payment messages. The payment gateway interfaces between SET and the existing bankcard payment networks for authorization and payment functions.

Certification authority (CA):

This is an entity that is trusted to issue X.509v3 public-keycertificates for cardholders, merchants, and payment gateways. The success of SET will depend on the existence of a CA infrastructure available for this purpose.

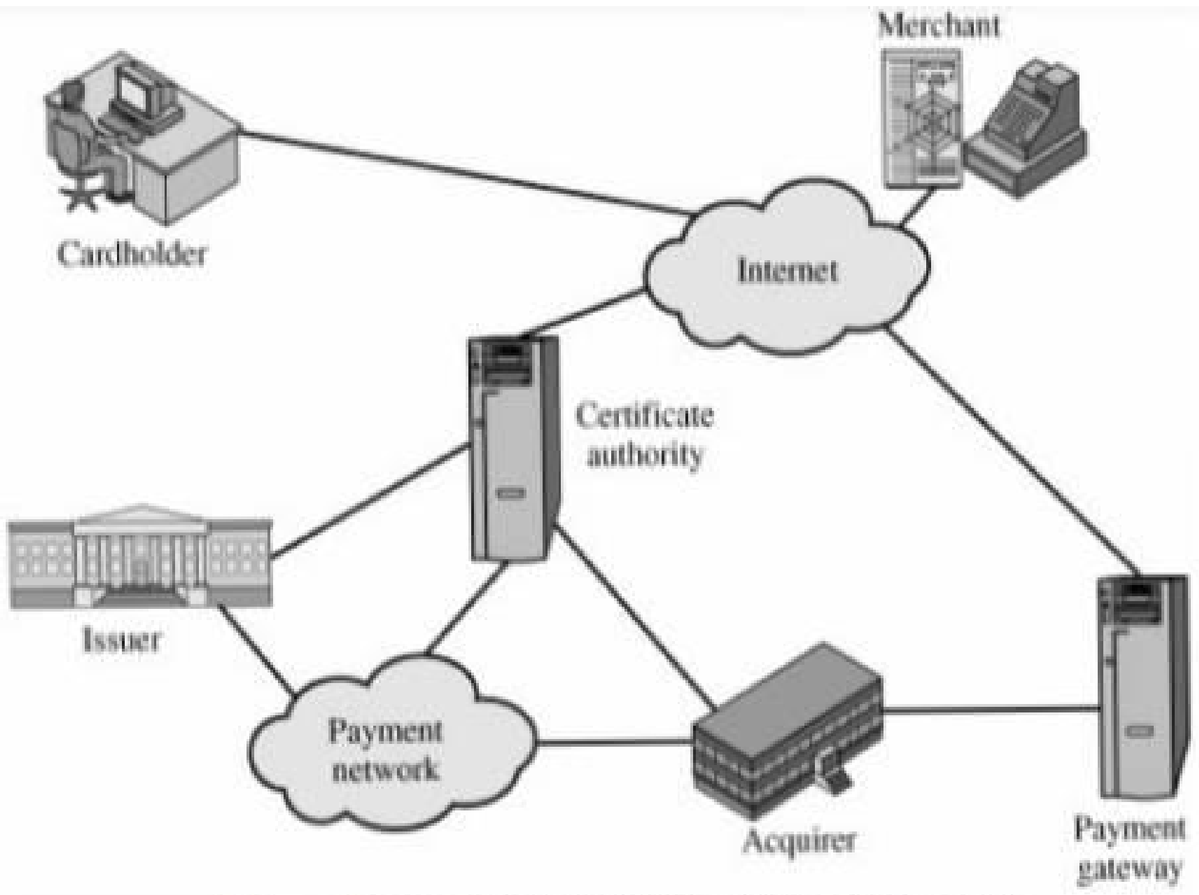


Figure: Secure Electronic Commerce Components

Sequence of events for a transaction:

- The customer opens an account
- The customer receives a certificate
- Merchants have their own certificates
- The customer places an order
- The merchant is verified.
- The order and payment are sent.
- The merchant requests payment authorization
- The merchant confirms the order.
- The merchant provides the goods or service.
- The merchant requests payment.

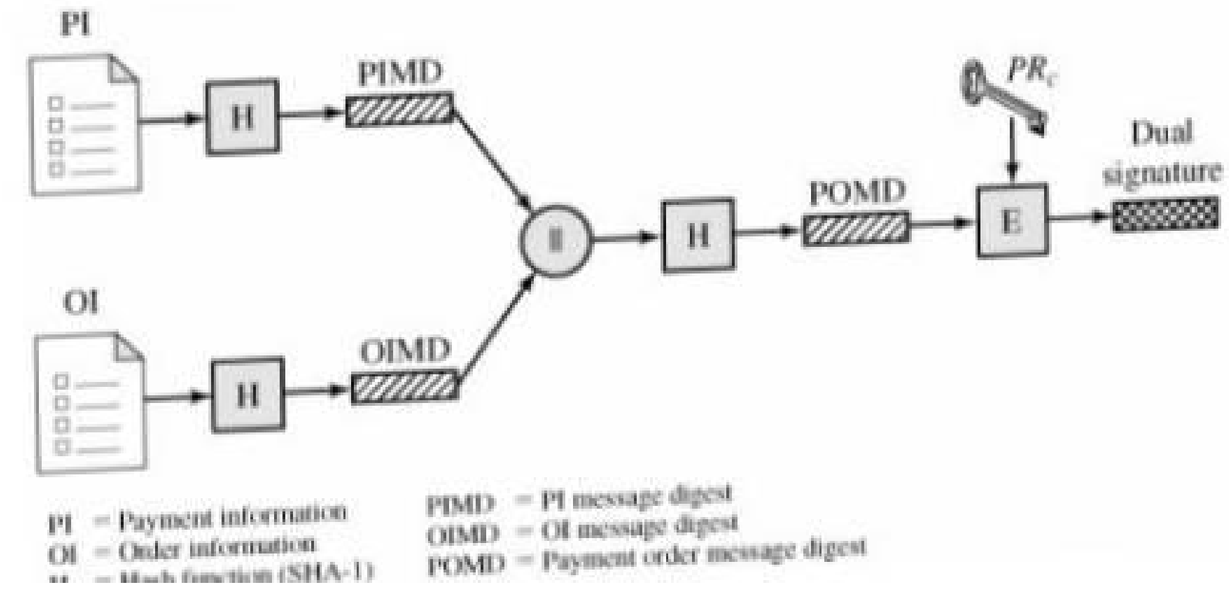
Dual Signature:

SET dual signature : The purpose of the dual signature is to link two messages that are intended for two different recipients. In this case, the customer wants to send the order information (OI) to the merchant and the payment information (PI) to the bank. The merchant does not need to know the customer's credit card number, and the bank does not need to know the details of the customer's order. The customer is afforded extra protection in terms of privacy by keeping these two items separate.

The customer takes the hash (using SHA-1) of the PI and the hash of the OI. These two hashes are then concatenated and the hash of the result is taken. Finally, the customer encrypts the final hash with his or her private signature key, creating the dual signature. The operation can be summarized as

$$DS = E(PRc, [H(H(PI)||H(OI))])$$

where PRc is the customer's private signature key.



Payment Processing:

- Purchase request
- Payment authorization
- Payment capture

SET Transaction Types:

Cardholder registration:

Cardholders must register with a CA before they can send SET messages to merchants.

Merchant registration:

Merchants must register with a CA before they can exchange SET messages with customers and payment gateways.

Purchase request:

Message from customer to merchant containing OI for merchant and PI for bank.

Payment authorization:

Exchange between merchant and payment gateway to authorize a given amount for a purchase on a given credit card account.

Payment capture:

Allows the merchant to request payment from the payment gateway.

Certificate inquiry and status:

The cardholder or merchant sends the Certificate Inquiry message to determine the status of the certificate request and to receive the certificate if the request has been approved.

Purchase inquiry:

Allows the cardholder to check the status of the processing of an order after the purchase response has been received.

Authorization reversal:

Allows a merchant to correct previous authorization requests. If the order will not be completed, the merchant reverses the entire authorization.

Capture reversal :

Allows a merchant to correct errors in capture requests such as transaction amounts that were entered incorrectly by a clerk.

Credit:

Allows a merchant to issue a credit to a cardholder's account such as when goods are returned or were damaged during shipping.

Credit reversal:

Allows a merchant to correct a previously request credit. **Payment Gateway certificate request**:

Allows a merchant to query the payment gateway and receive a copy of the gateway's current key-exchange and signature certificates.

Batch administration:

Allows a merchant to communicate information to the payment gateway regarding merchant batches.

1. PREETY GOOD PRIVACY

- **Explain Pretty Good Privacy in detail. (16 Marks) May/June'14,Nov/Dec'12**
- **Explain PGP message generation and reception.(16 Marks) Apr/May'11**
- **Illustrate the confidentiality service provided by PGP.(8 Marks) (May/June'2007)**
- **For what purpose Zimmerman developed PGP? Brief the various services provided by PGP. Discuss the threats faced by an e-mail and explain its security requirements to provide a secure e-mail service. (16 Marks) (Nov/Dec '14)**

Pretty Good Privacy

Definition of PGP: **(2 Marks Nov/Dec'2013)**

PGP provides confidentiality and authentication service that can be used for electronic mail and file storage applications.

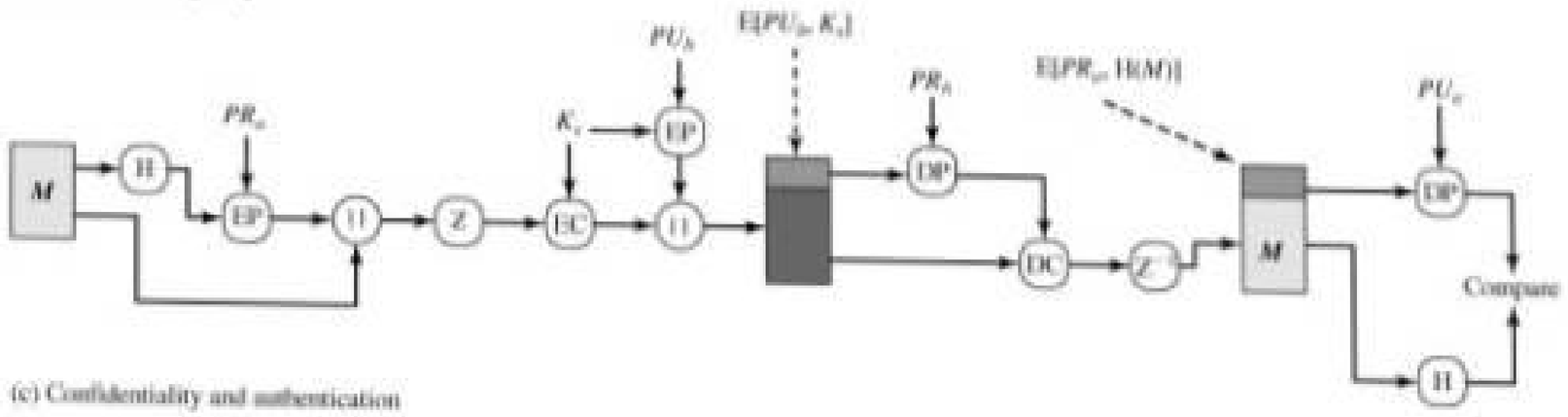
Pretty Good Privacy is an open-source freely available software package for e-mail security. It provides authentication through the use of digital signature; confidentiality through the use of symmetric block encryption; compression using the ZIP algorithm; e-mail compatibility using the radix-64 encoding scheme; and segmentation and reassembly to accommodate long e-mails.

PGP provides a confidentiality and authentication service that can be used for electronic mail and file storage applications.

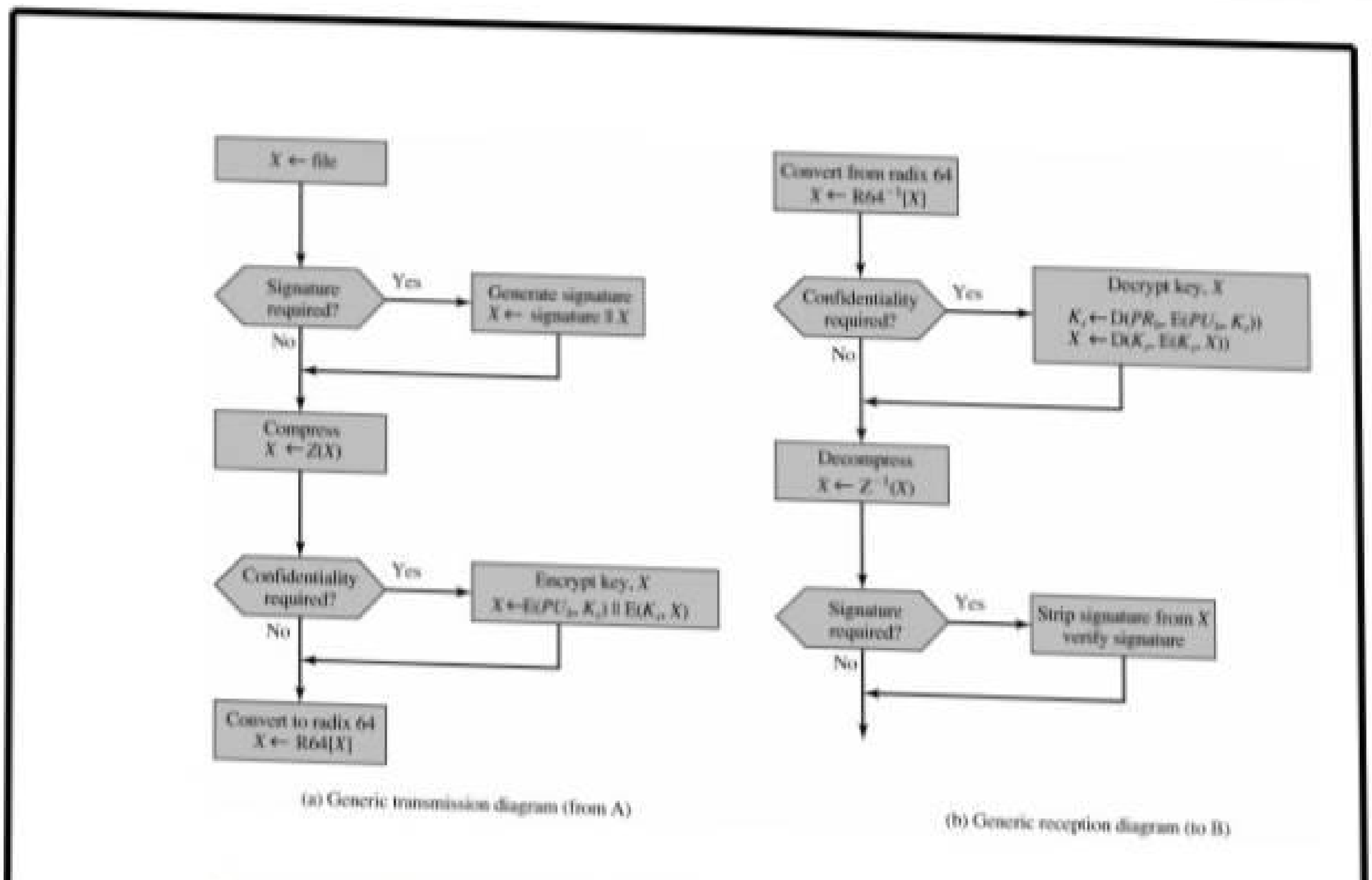
1. Selected the best available cryptographic algorithms as building blocks
2. Integrated these algorithms into a general-purpose application that is independent of operating system and processor and that is based on a small set of easy-to- use commands
3. Made the package and its documentation, including the source code, freely available via the Internet, bulletin boards, and commercial networks such as AOL (America On Line)
4. Entered into an agreement with a company (Viacrypt, now Network Associates) to provide a fully compatible, low-cost commercial version of PGP.

Confidentiality and Authentication

(b) Confidentiality only



(c) Confidentiality and authentication



(a) Generic transmission diagram (from A)

(b) Generic reception diagram (to B)

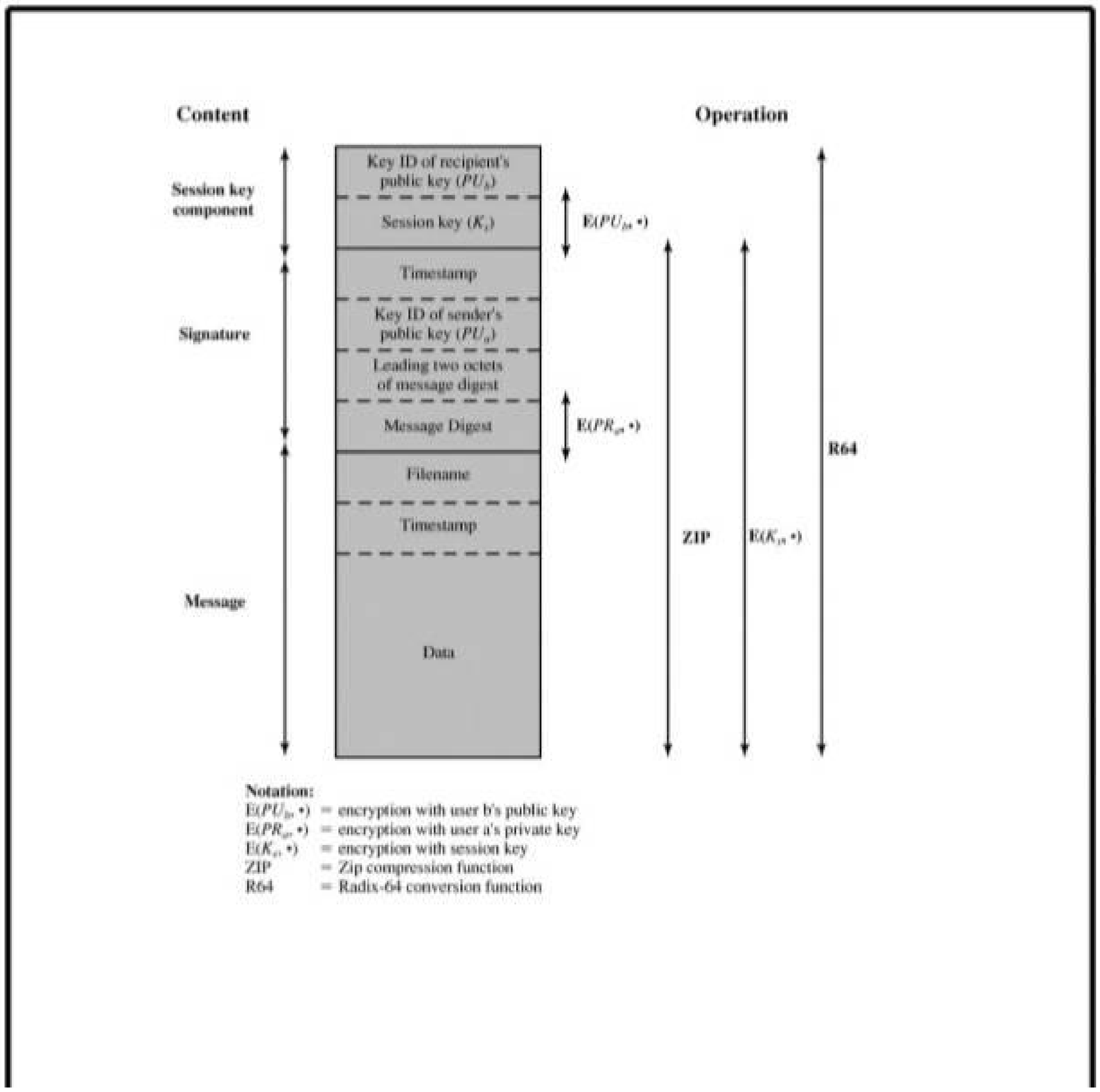
Cryptographic Keys and Key Rings

1. A means ofType equation here. generating unpredictable session keys is needed.
2. We would like to allow a user to have multiple public-key/private-key pairs.
3. Each PGP entity must maintain a file of its own public/private key pairs as well as a file of public keys of correspondents.

Cryptographic Keys and Key Rings

1. A means ofType equation here. generating unpredictable session keys is needed.
2. We would like to allow a user to have multiple public-key/private-key pairs.
3. Each PGP entity must maintain a file of its own public/private key pairs as well as a file of public keys of correspondents.

General Format of PGP Message (from A to B)
Sketch the general format for PGP message. (2 Marks-Nov/Dec'2014)



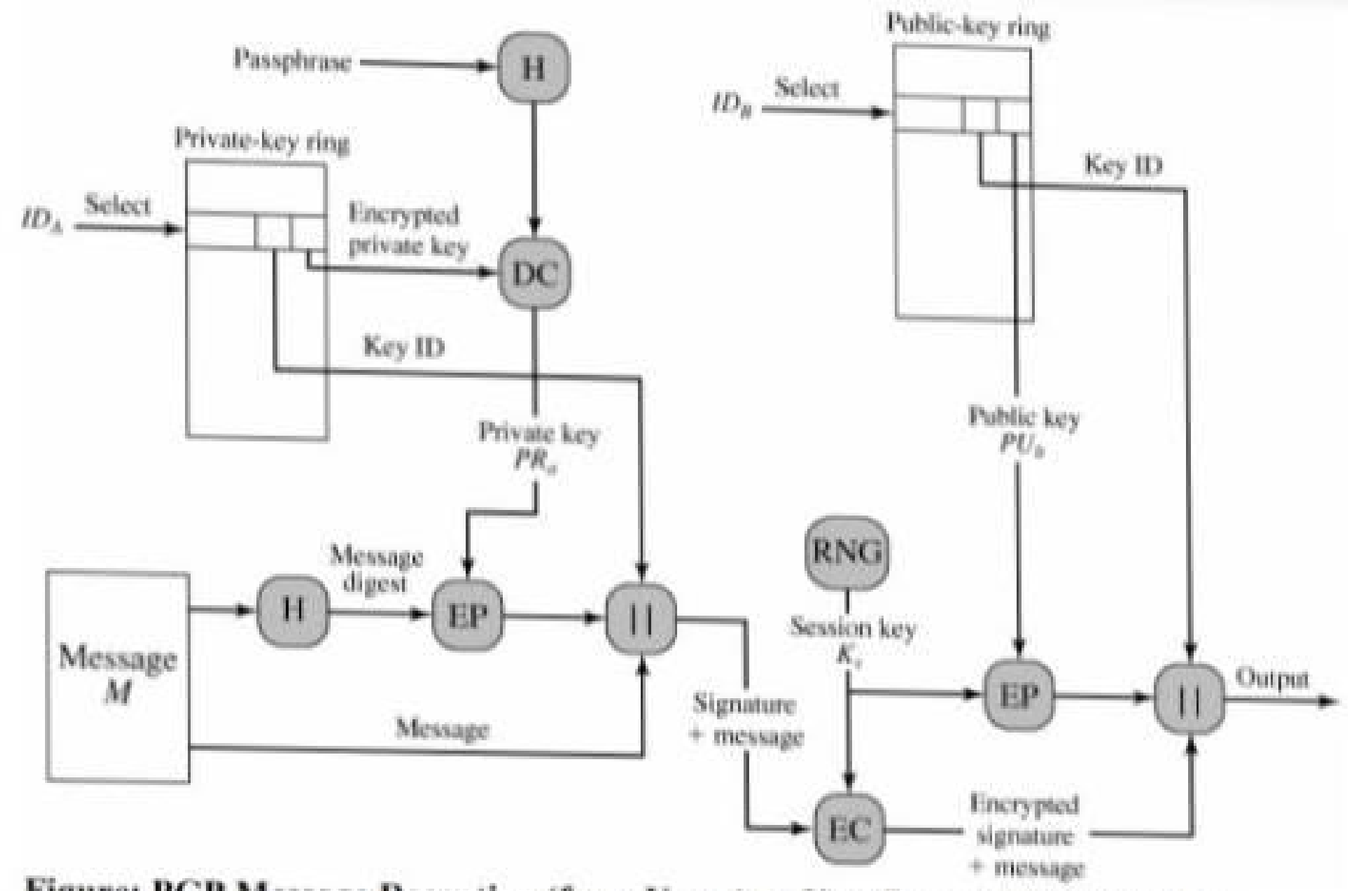


Figure: PGP Message Reception (from User A to User B; no compression or radix 64 conversion)

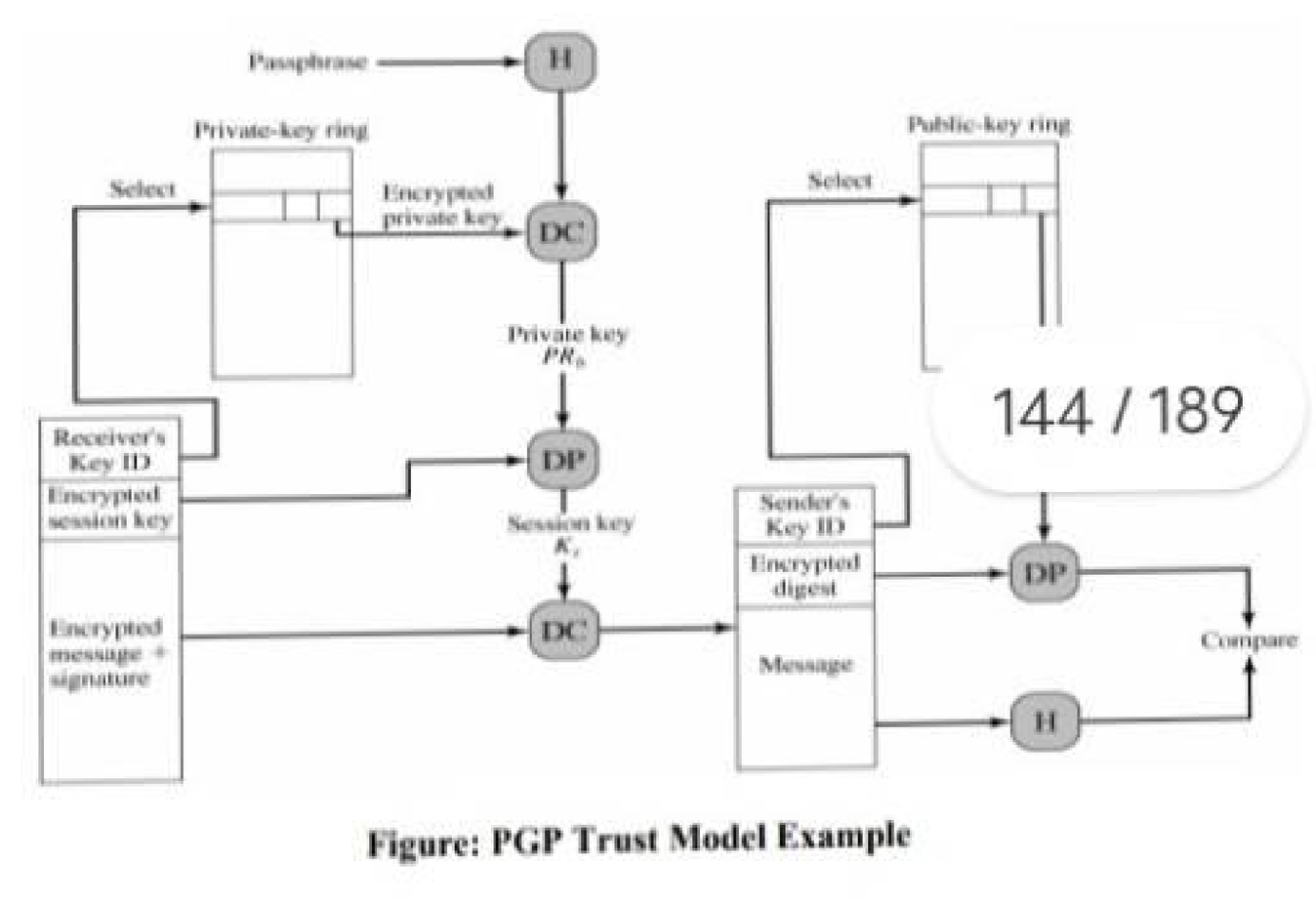


Figure: PGP Trust Model Example