Pretty Good Privacy (PGP) Security for Email

G.P. Biswas

Prof/CSE, IIT, Dhanbad

PGP

- Developed by Phil Zimmerman in 1995.
- Documentation and source code is freely available.
- The package is independent of operating system and processor.
- PGP does not rely on any setup/establishment.
- It is very popular and used extensively since 1995.

- PGP combines the best available cryptographic algorithms to achieve secure e-mail communication.
- It is assumed that all users are using public key cryptography and have generated a private/public key pair.
- Either RSA (with RSA digital signatures) or ElGamel (with DSA) can be used.
- All users also use a symmetric key system such as triple-DES or Rijndael.

What does PGP do?

PGP consists of 5 components:

1. Authentication
2. Confidentiality
3. Compression
4. E-mail compatibility
5. Segmentation

Summary of PGP Services

Function	Algorithms Used	Description
Digital signature	DSS/SHA or RSA/SHA	A hash code of a message is created using SHA-1. This message digest is encrypted using DSS or RSA with the sender's private key and included with the message.
Message encryption	CAST or IDEA or Three-key Triple DES with Diffie-Hellman or RSA	A message is encrypted using CAST-128 or IDEA or 3DES with a one-time session key generated by the sender. The session key is encrypted using Diffie-Hellman or RSA with the recipient's public key and included with the message.
Compression	ZIP	A message may be compressed for storage or transmission using ZIP.
E-mail compatibility	Radix-64 conversion	To provide transparency for e-mail applications, an encrypted message may be converted to an ASCII string using radix-64 conversion.

PGP Authentication

- Combination of SHA-1 and RSA provides an effective digital signature scheme
  - Because of the strength of RSA the recipient is assured that only the possessor of the matching private key can generate the signature
  - Because of the strength of SHA-1 the recipient is assured that no one else could generate a new message that matches the hash code
- As an alternative, signatures can be generated using DSS/SHA-1
- Detached signatures are supported
  - Each person's signature is independent and therefore applied only to the document

PGP Confidentiality

- Provided by encrypting messages to be transmitted or to be stored locally as files
  - In both cases the symmetric encryption algorithm CAST-128 may be used
  - Alternatively IDEA or 3DES may be used
  - The 64-bit cipher feedback (CFB) mode is used

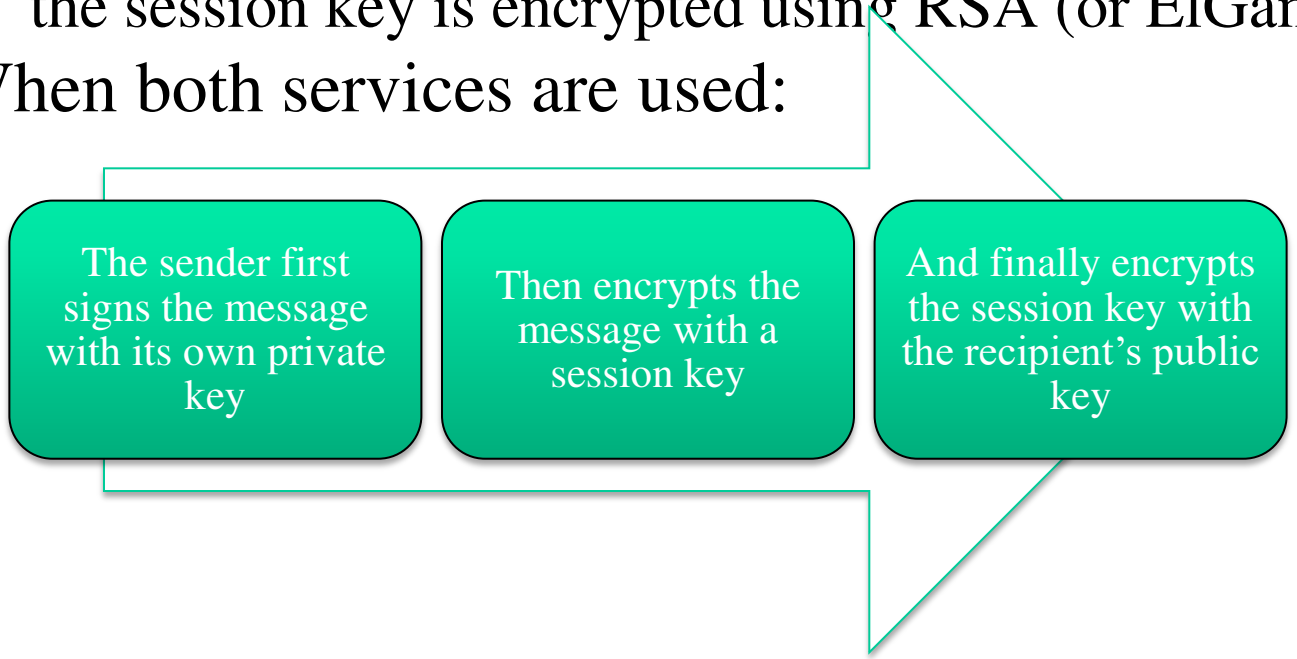
In PGP each symmetric key is used only once

- Although referred to as a session key, it is in reality a one-time key
- Session key is bound to the message and transmitted with it
- To protect the key, it is encrypted with the receiver's public key

- As an alternative to the use of RSA for key encryption, PGP uses ElGamal, a variant of Diffie-Hellman that provides encryption/decryption

PGP Confidentiality and Authentication

- Both services may be used for the same message
  - First a signature is generated for the plaintext message and prepended to the message
  - Then the plaintext message plus signature is encrypted using CAST-128 (or IDEA or 3DES) and the session key is encrypted using RSA (or ElGamal)
- When both services are used:

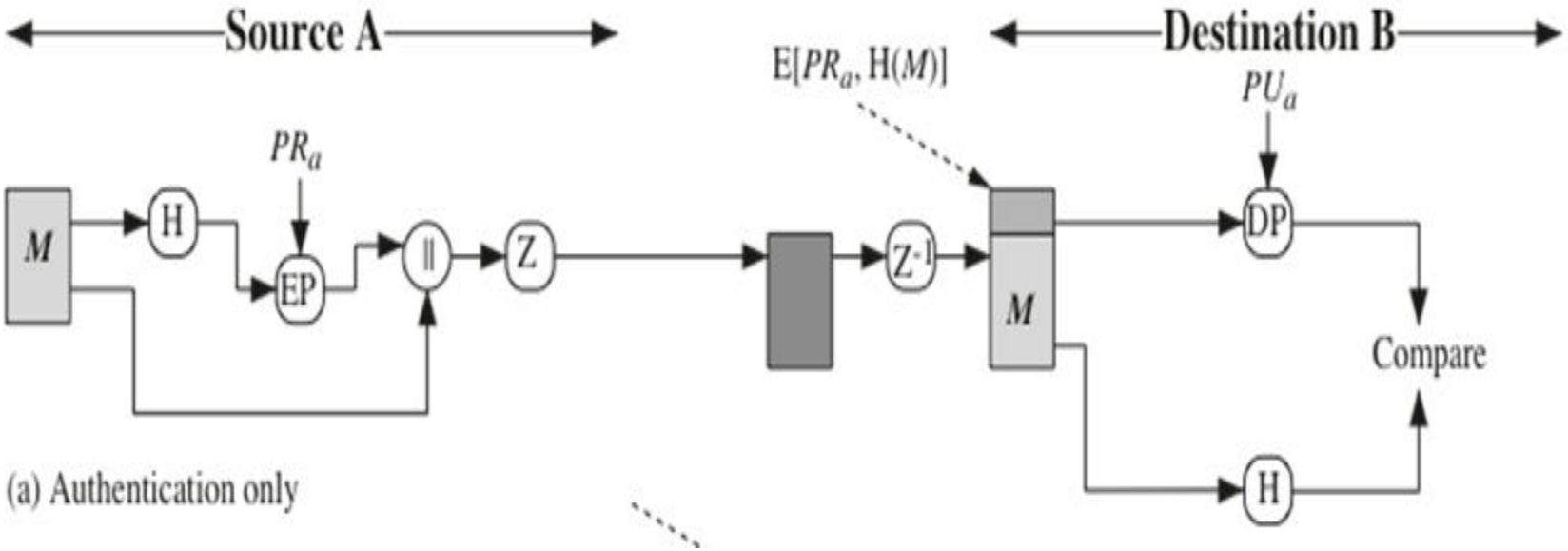


PGP Compression

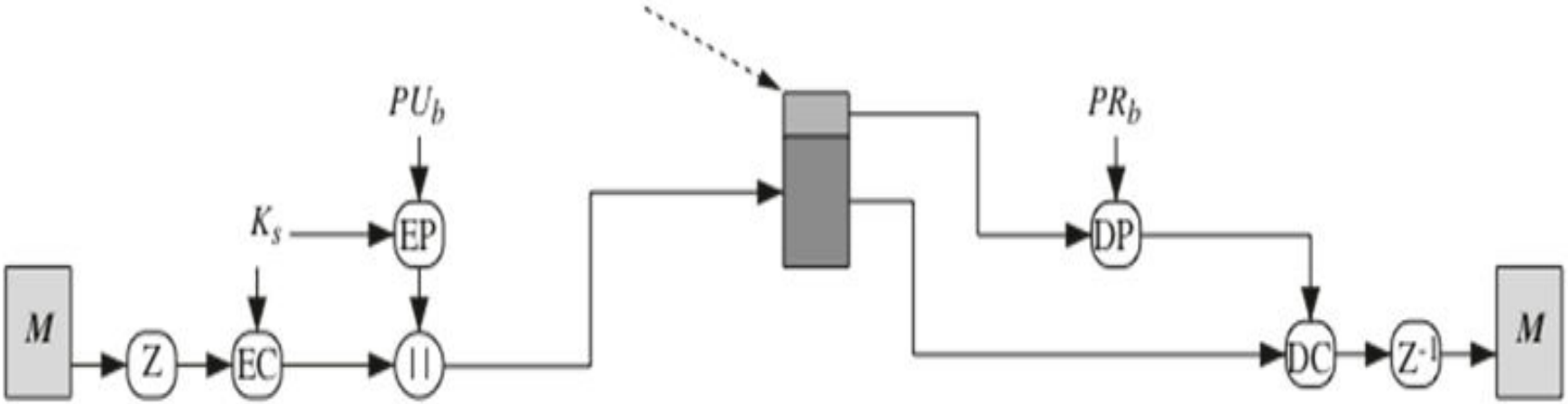
- As a default, PGP compresses the message after applying the signature but before encryption
  - This has the benefit of saving space both for e-mail transmission and for file storage
  - The placement of the compression algorithm is critical
    - Applying the hash function and signature after compression would constrain all PGP implementations to the same version of the compression algorithm
    - Message encryption is applied after compression to strengthen cryptographic security
  - The compression algorithm used is ZIP

PGP E-mail Compatibility

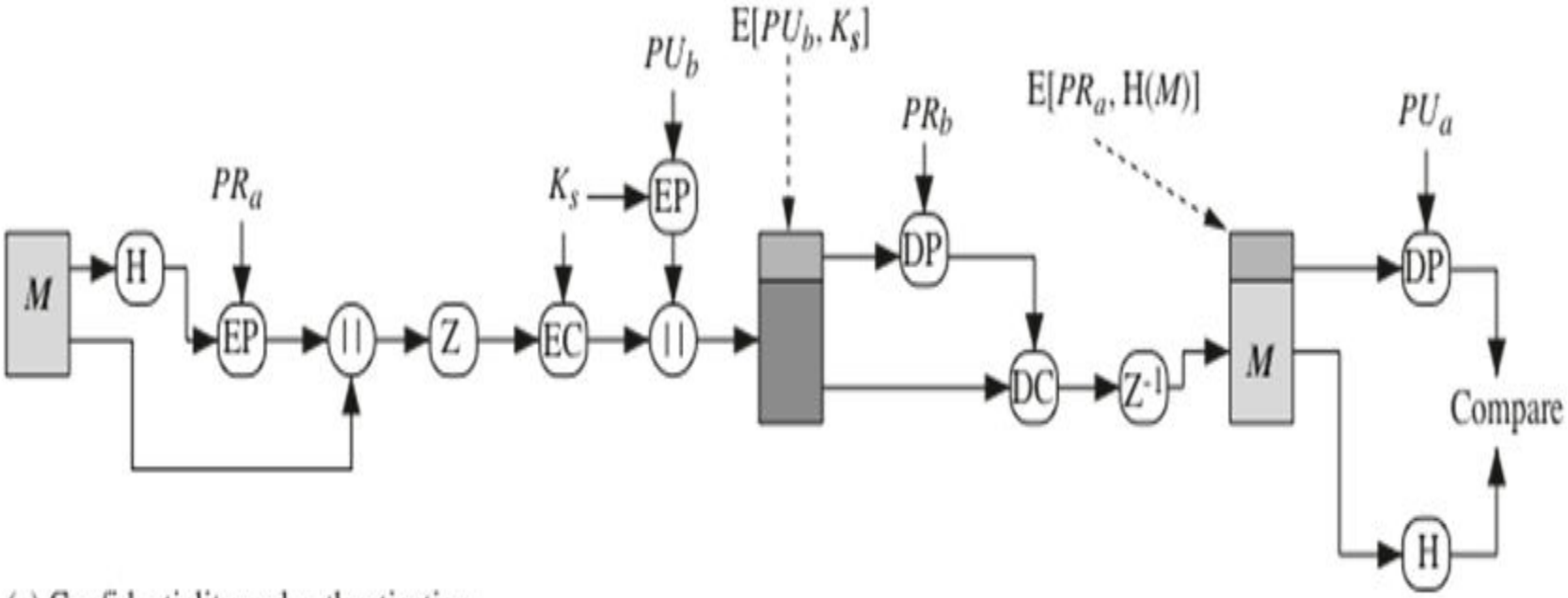
- Many electronic mail systems only permit the use of blocks consisting of ASCII text
  - To accommodate this restriction, PGP provides the service of converting the raw 8-bit binary stream to a stream of printable ASCII characters
  - The scheme used for this purpose is radix-64 conversion
    - Each group of three octets of binary data is mapped into four ASCII characters
    - This format also appends a CRC to detect transmission errors



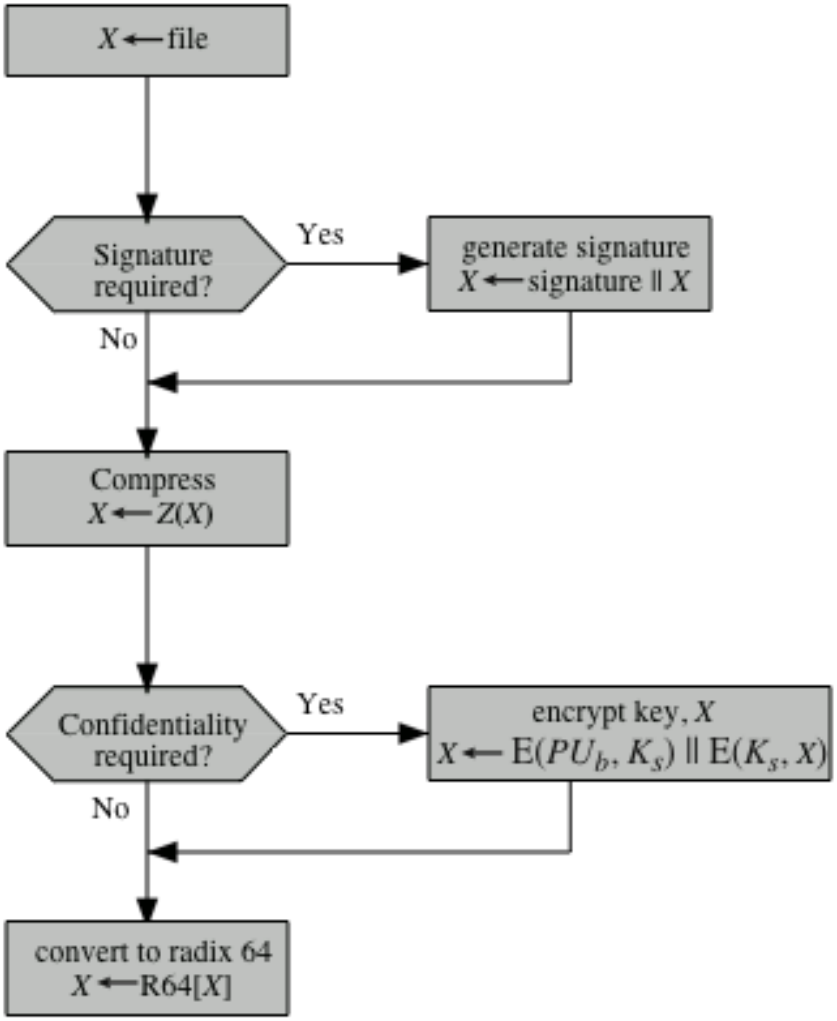
(a) Authentication only



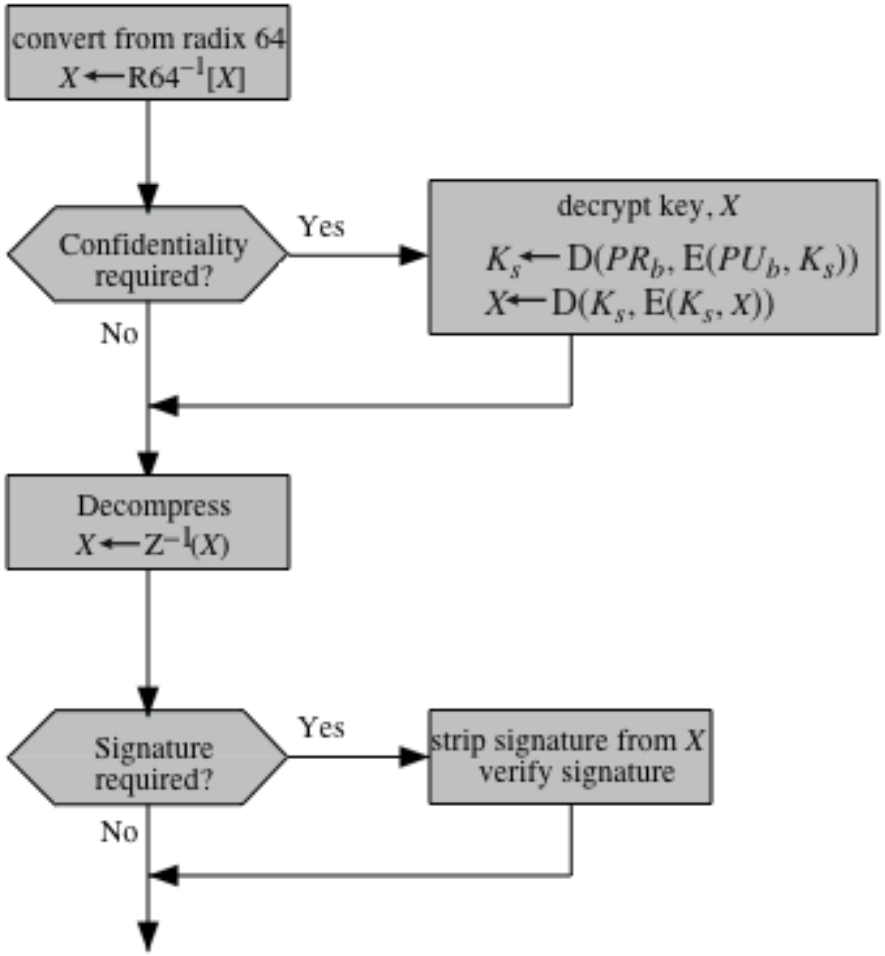
(b) Confidentiality only



(c) Confidentiality and authentication



(a) Generic Transmission Diagram (from A)



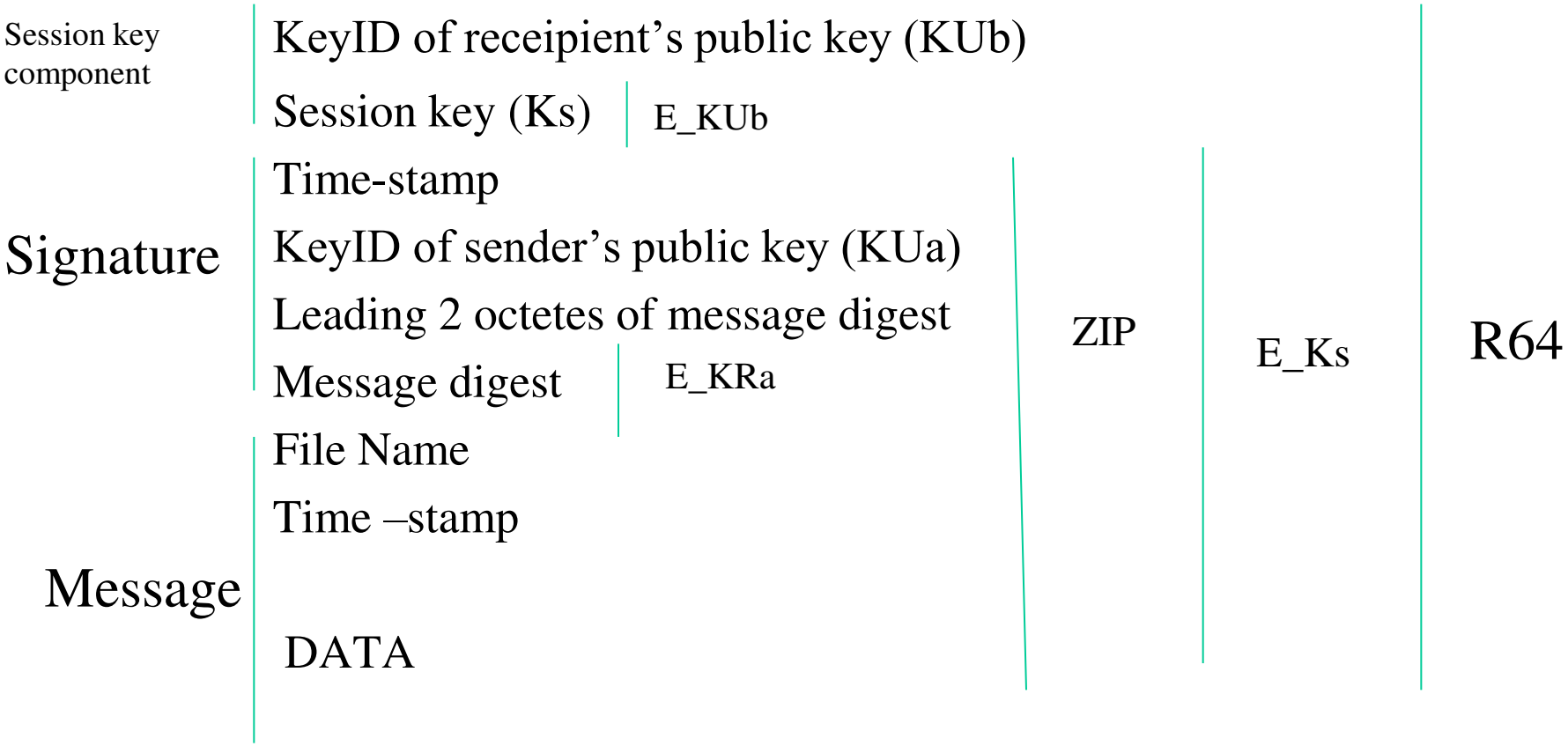
(b) Generic Reception Diagram (to B)

Figure 19.2 Transmission and Reception of PGP Messages

Key Rings

- Each user maintains private key and public key rings using certain formats as given below:
- Private key ring:
  - Timestamp, KeyID (KU mod 2^64), Public key KU, Encrypted private key, UserID
- Public key ring:
  - Time stamp, KeyID, Public key, Owner trust, UserID, Key Legitimacy, Signature, Signature-trust

General PGP Message Format



Data Compression using ZIP

- ZIP is written by Jean-lup Gailly, Mark Adler and Richard Wales. It is a free-ware package written in C.
- ZIP is functionally equivalent to PKZIP
- ZIP (and similar algorithms) is developed by Jacob Ziv and Abraham Lempel referred as LZ77. A version of this algorithm is used in the zip compression scheme as PKZIP, gzip, zipit etc.

- LZ77 and its variants exploit the fact that words and phrases within a text stream are likely to be repeated.
- When a repetition occurs, the repeated sequence can be replaced by short code.
- The compression program scans for such repetition and develops codes on the fly to replace the repeated sequence. The algorithm must be defined in such a way that the decompression program is able deduce the current mapping between codes and sequences of source data.
- For example,
  - the brown fox jumped over the brown foxy jumping frog
- It is a 53 octets = 424 bits long. In LZ77, initially each character is mapped into 9-bit pattern consisting of a binary 1 followed by 8-bit ASCII representation.
- the brown fox (its length is 13 ASCIIs) is repeated so represented as <00b><26d><13d> or 00 00011010 1101 which is 14-bit length
- 00: 8-bit pointer and 4-bit length and 01: 12-bit pointer and 6-bit length (it follows 00, pointer 26 decimal and length 13 decimal)

and so on

PGP E-Mail Compatibility

Many electronic mail systems can only transmit blocks of ASCII text. This can cause a problem when sending encrypted data since ciphertext blocks might not correspond to ASCII characters which can be transmitted.

PGP overcomes this problem by using **radix-64 conversion**.

Radix-64 conversion

Suppose the text to be encrypted has been converted into binary using ASCII coding and encrypted to give a ciphertext stream of binary.

Radix-64 conversion maps arbitrary binary into printable characters as follows:

Radix-64 conversion

1. The binary input is split into blocks of 24 bits (3 bytes).
2. Each 24 block is then split into four sets each of 6-bits.
3. Each 6-bit set will then have a value between 0 and 2^6-1 (=63).
4. This value is encoded into a printable character.

6 bit value	Character encoding	6 bit value	Character encoding	6 bit value	Character encoding	6 bit value	Character encoding
0	A	16	Q	32	g	48	w
1	B	17	R	33	h	49	x
2	C	18	S	34	i	50	y
3	D	19	T	35	j	51	z
4	E	20	U	36	k	52	0
5	F	21	V	37	l	53	1
6	G	22	W	38	m	54	2
7	H	23	X	39	n	55	3
8	I	24	Y	40	o	56	4
9	J	25	Z	41	p	57	5
10	K	26	a	42	q	58	6
11	L	27	b	43	r	59	7
12	M	28	c	44	s	60	8
13	N	29	d	45	t	61	9
14	O	30	e	46	u	62	+
15	P	31	f	47	v	63	/
						(pad)	=

PGP Segmentation

Another constraint of e-mail is that there is usually a maximum message length.

PGP automatically blocks an encrypted message into segments of an appropriate length.

On receipt, the segments must be re-assembled before the decryption process.