8

NETWORK SECURITY

Adversary	Goal
Student	To have fun snooping on people's e-mail
Cracker	To test out someone's security system; steal data
Sales rep	To claim to represent all of Europe, not just Andorra
Businessman	To discover a competitor's strategic marketing plan
Ex-employee	To get revenge for being fired
Accountant	To embezzle money from a company
Stockbroker	To deny a promise made to a customer by e-mail
Con man	To steal credit card numbers for sale
Spy	To learn an enemy's military or industrial secrets
Terrorist	To steal germ warfare secrets

Fig. 8-1. Some people who cause security problems and why.

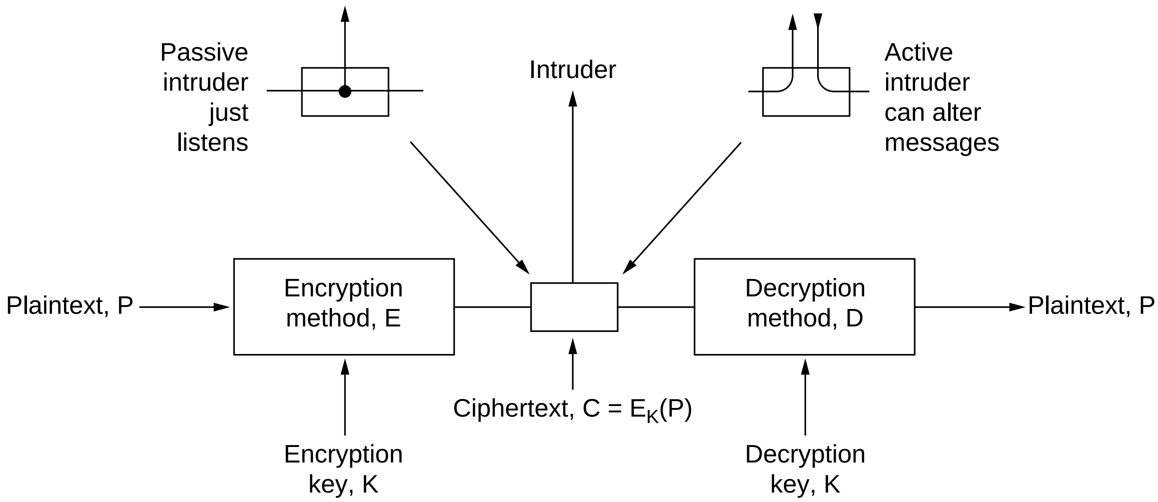


Fig. 8-2. The encryption model (for a symmetric-key cipher).

M	E	G	A	B	U	C	K
7	4	5	1	2	8	3	6
p	l	e	a	s	e	t	r
a	n	s	f	e	r	o	n
e	m	i	l	l	i	o	n
d	o	l	l	a	r	s	t
o	m	y	s	w	i	s	s
b	a	n	k	a	c	c	o
u	n	t	s	i	x	t	w
o	t	w	o	a	b	c	d

Plaintext

pleasetransferonemilliondollarsto
myswissbankaccountsixtwotwo

Ciphertext

AFLLSKSOSELAWAIATOOSSCTCLNMOMANT
ESILYNTWRNNTSOWDPAEDOBUOERIRICXB

Fig. 8-3. A transposition cipher.

Message 1:	1001001	0100000	1101100	1101111	1110110	1100101	0100000	1111001	1101111	1110101	0101110
Pad 1:	1010010	1001011	1110010	1010101	1010010	1100011	0001011	0101010	1010111	1100110	0101011
Ciphertext:	0011011	1101011	0011110	0111010	0100100	0000110	0101011	1010011	0111000	0010011	0000101
Pad 2:	1011110	0000111	1101000	1010011	1010111	0100110	1000111	0111010	1001110	1110110	1110110
Plaintext 2:	1000101	1101100	1110110	1101001	1110011	0100000	1101100	1101001	1110110	1100101	1110011

Fig. 8-4. The use of a one-time pad for encryption and the possibility of getting any possible plaintext from the ciphertext by the use of some other pad.

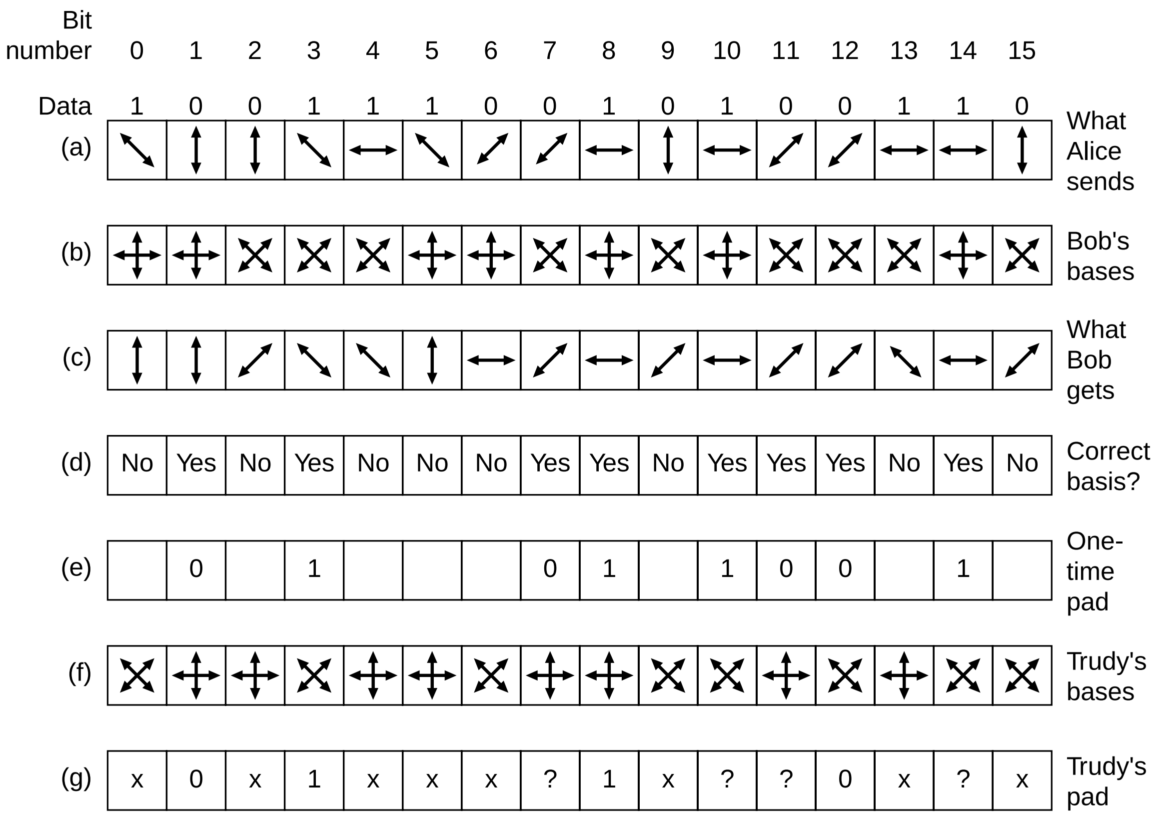


Fig. 8-5. An example of quantum cryptography.

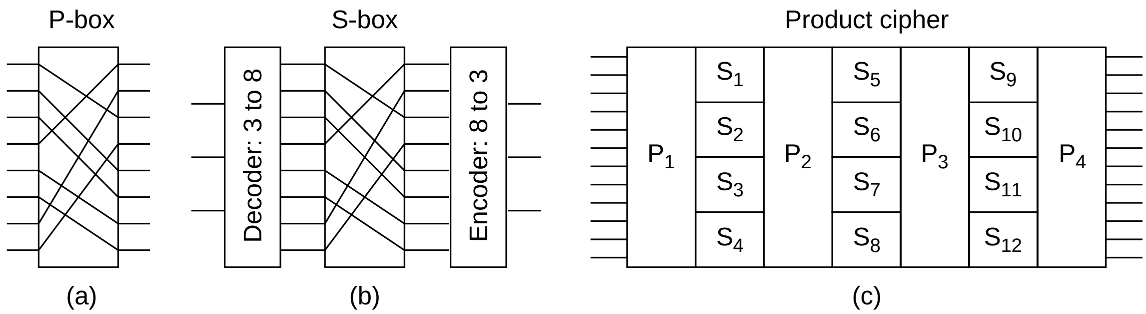


Fig. 8-6. Basic elements of product ciphers. (a) P-box. (b) S-box. (c) Product.

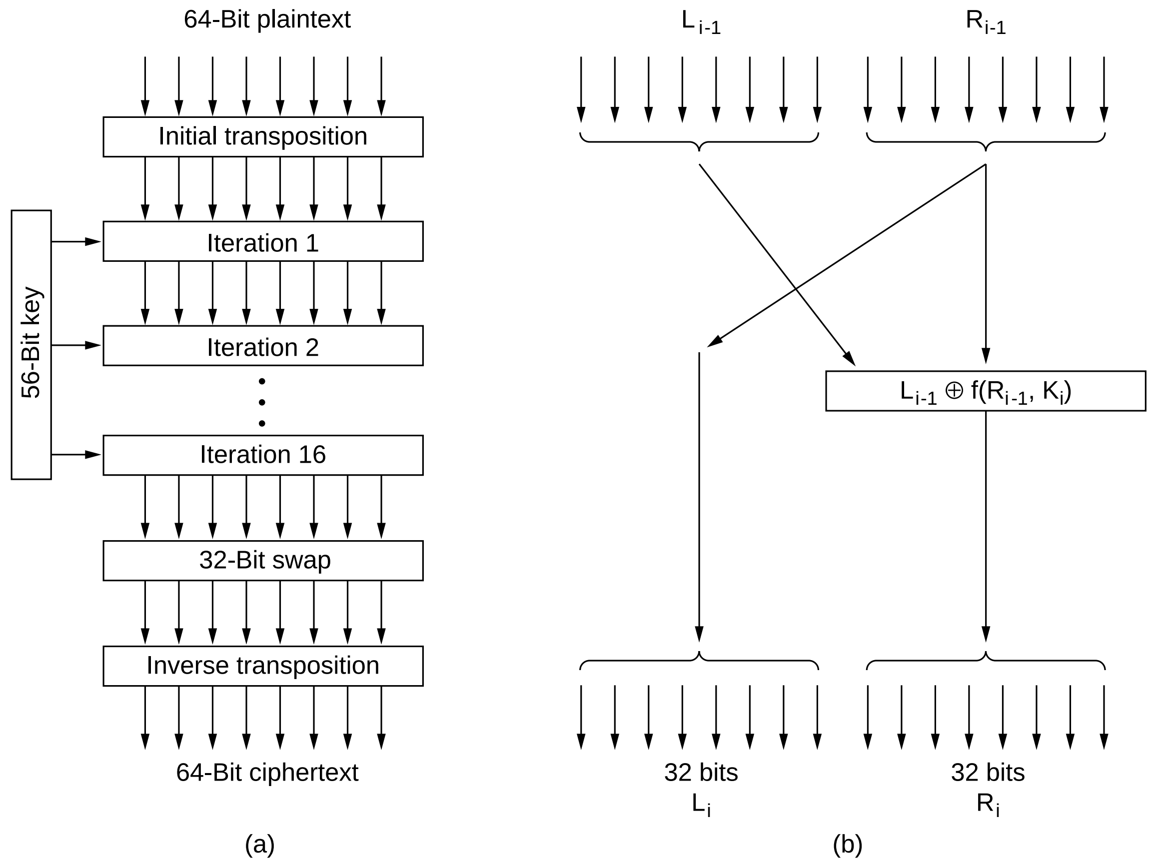


Fig. 8-7. The data encryption standard. (a) General outline.
(b) Detail of one iteration. The circled + means exclusive OR.

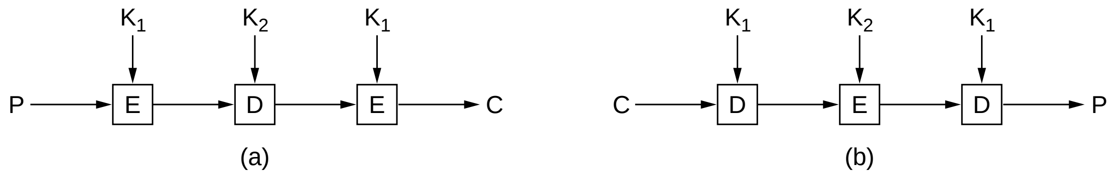


Fig. 8-8. (a) Triple encryption using DES. (b) Decryption.

```
#define LENGTH 16                         /* # bytes in data block or key */
#define NROWS 4                           /* number of rows in state */
#define NCOLS 4                           /* number of columns in state */
#define ROUNDS 10                         /* number of iterations */
typedef unsigned char byte;               /* unsigned 8-bit integer */

rijndael(byte plaintext[LENGTH], byte ciphertext[LENGTH], byte key[LENGTH])
{
 int r;                                   /* loop index */
 byte state[NROWS][NCOLS];                /* current state */
 struct {byte k[NROWS][NCOLS];} rk[ROUNDS + 1];      /* round keys */

 expand_key(key, rk);                     /* construct the round keys */
 copy_plaintext_to_state(state, plaintext);   /* init current state */
 xor_roundkey_into_state(state, rk[0]);   /* XOR key into state */

 for (r = 1; r <= ROUNDS; r++) {
      substitute(state);                  /* apply S-box to each byte */
      rotate_rows(state);                 /* rotate row i by i bytes */
      if (r < ROUNDS) mix_columns(state); /* mix function */
      xor_roundkey_into_state(state, rk[r]);  /* XOR key into state */
 }
 copy_state_to_ciphertext(ciphertext, state);      /* return result */
}
```

Fig. 8-9. An outline of Rijndael.

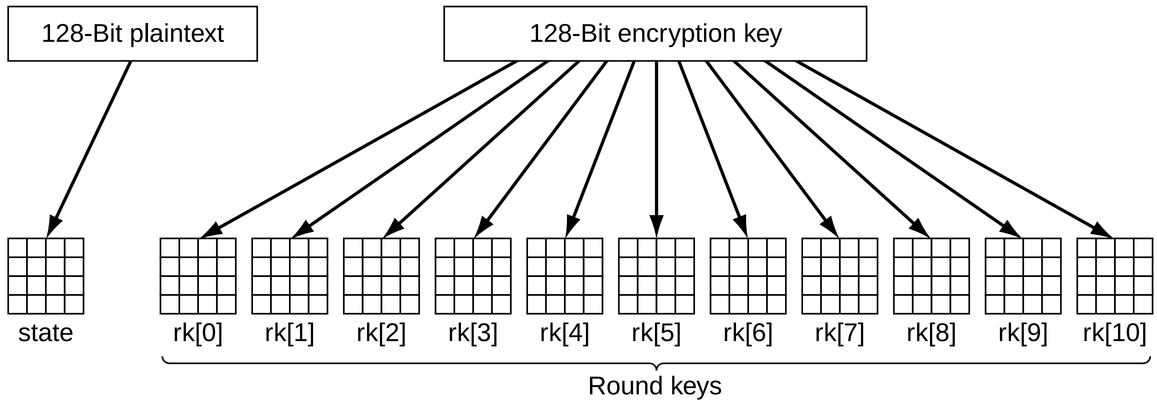


Fig. 8-10. Creating of the *state* and *rk* arrays.

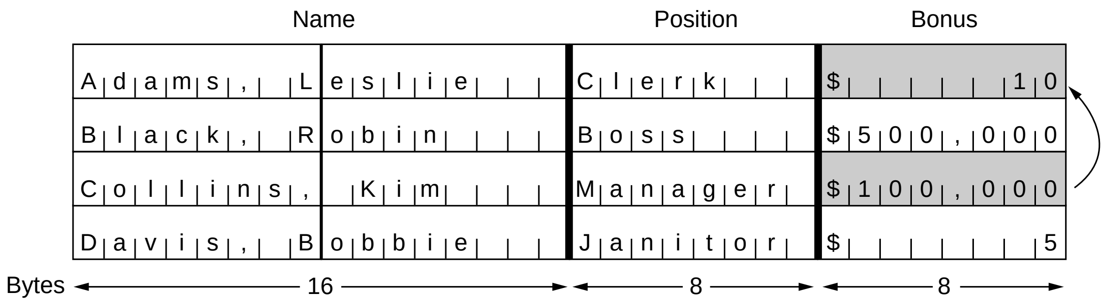


Fig. 8-11. The plaintext of a file encrypted as 16 DES blocks.

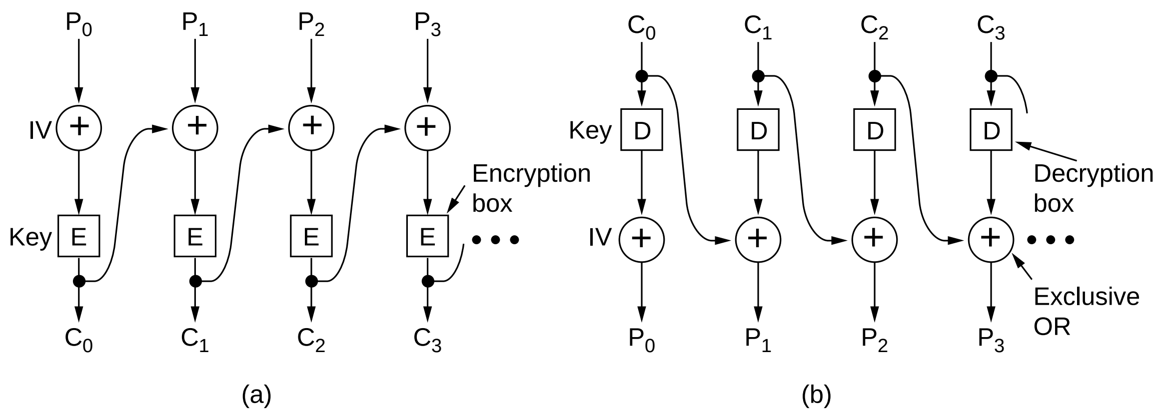


Fig. 8-12. Cipher block chaining. (a) Encryption. (b) Decryption.

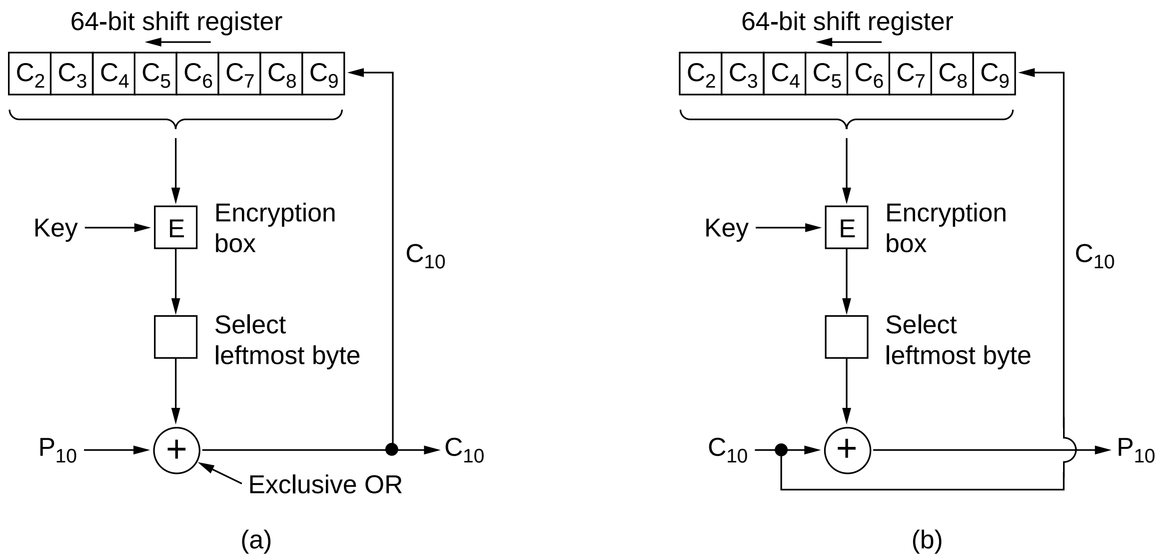


Fig. 8-13. Cipher feedback mode. (a) Encryption. (b) Decryption.

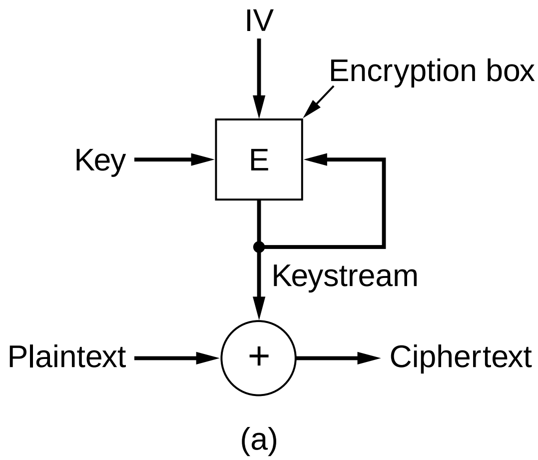


Fig. 8-14. A stream cipher. (a) Encryption. (b) Decryption.

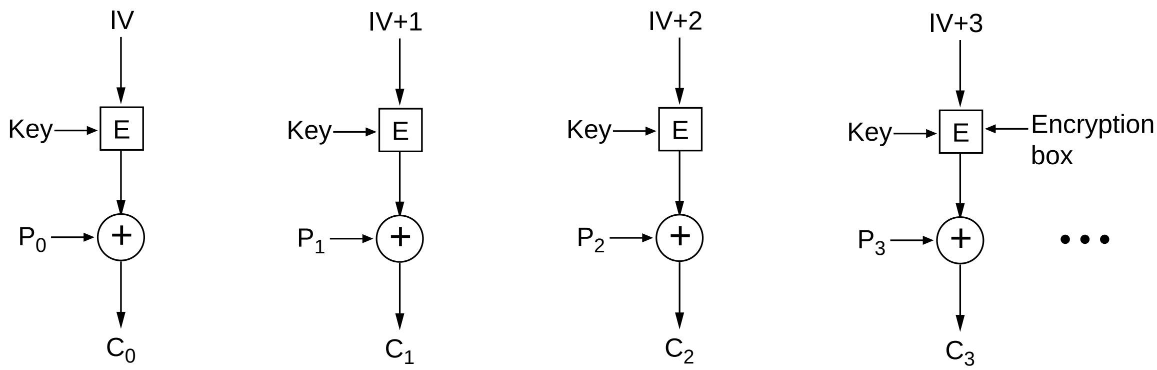


Fig. 8-15. Encryption using counter mode.

Cipher	Author	Key length	Comments
Blowfish	Bruce Schneier	1–448 bits	Old and slow
DES	IBM	56 bits	Too weak to use now
IDEA	Massey and Xuejia	128 bits	Good, but patented
RC4	Ronald Rivest	1–2048 bits	Caution: some keys are weak
RC5	Ronald Rivest	128–256 bits	Good, but patented
Rijndael	Daemen and Rijmen	128–256 bits	Best choice
Serpent	Anderson, Biham, Knudsen	128–256 bits	Very strong
Triple DES	IBM	168 bits	Second best choice
Twofish	Bruce Schneier	128–256 bits	Very strong; widely used

Fig. 8-16. Some common symmetric-key cryptographic algorithms.

Plaintext (P)			Ciphertext (C)		After decryption	
Symbolic	Numeric	P^3	P^3 (mod 33)	C^7	C^7 (mod 33)	Symbolic
S	19	6859	28	13492928512	19	S
U	21	9261	21	1801088541	21	U
Z	26	17576	20	1280000000	26	Z
A	01	1	1	1	01	A
N	14	2744	5	78125	14	N
N	14	2744	5	78125	14	N
E	05	125	26	8031810176	05	E
Sender's computation				Receiver's computation		

Fig. 8-17. An example of the RSA algorithm.

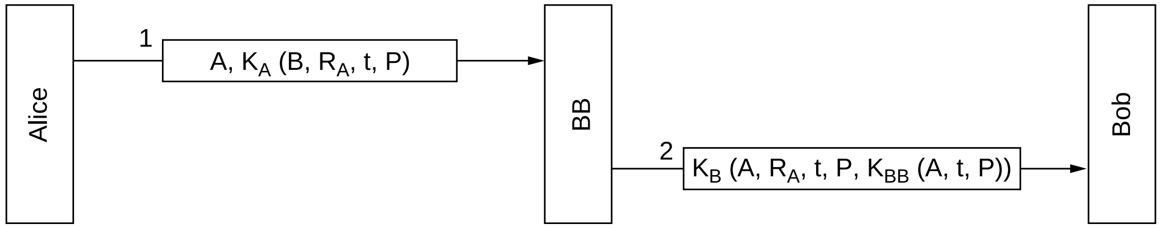


Fig. 8-18. Digital signatures with Big Brother.

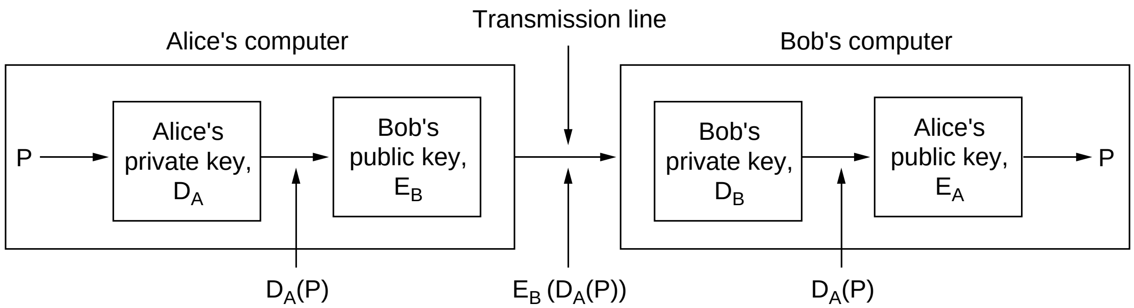


Fig. 8-19. Digital signatures using public-key cryptography.

Fig. 8-20. Digital signatures using message digests.

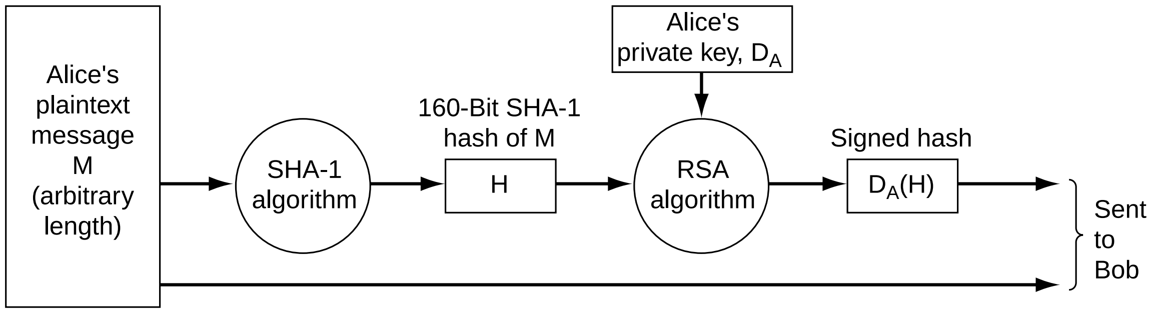


Fig. 8-21. Use of SHA-1 and RSA for signing nonsecret messages.

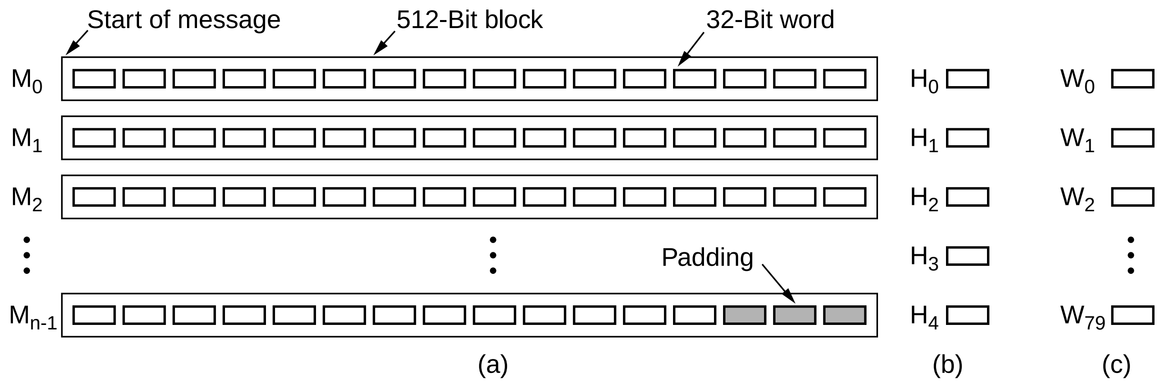


Fig. 8-22. (a) A message padded out to a multiple of 512 bits. (b) The output variables. (c) The word array.

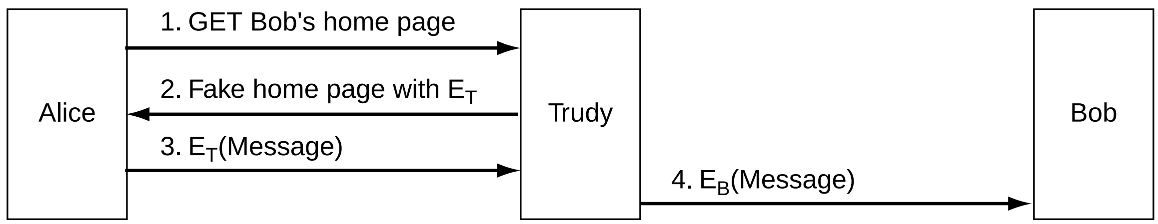


Fig. 8-23. A way for Trudy to subvert public-key encryption.

I hereby certify that the public key
    19836A8B03030CF83737E3837837FC3s87092827262643FFA82710382828282A
belongs to
    Robert John Smith
    12345 University Avenue
    Berkeley, CA 94702
    Birthday: July 4, 1958
    Email: bob@superdupernet.com

SHA-1 hash of the above certificate with the CA's private key

Fig. 8-24. A possible certificate and its signed hash.

Field	Meaning
Version	Which version of X.509
Serial number	This number plus the CA's name uniquely identifies the certificate
Signature algorithm	The algorithm used to sign the certificate
Issuer	X.500 name of the CA
Validity period	The starting and ending times of the validity period
Subject name	The entity whose key is being certified
Public key	The subject's public key and the ID of the algorithm using it
Issuer ID	An optional ID uniquely identifying the certificate's issuer
Subject ID	An optional ID uniquely identifying the certificate's subject
Extensions	Many extensions have been defined
Signature	The certificate's signature (signed by the CA's private key)

Fig. 8-25. The basic fields of an X.509 certificate.

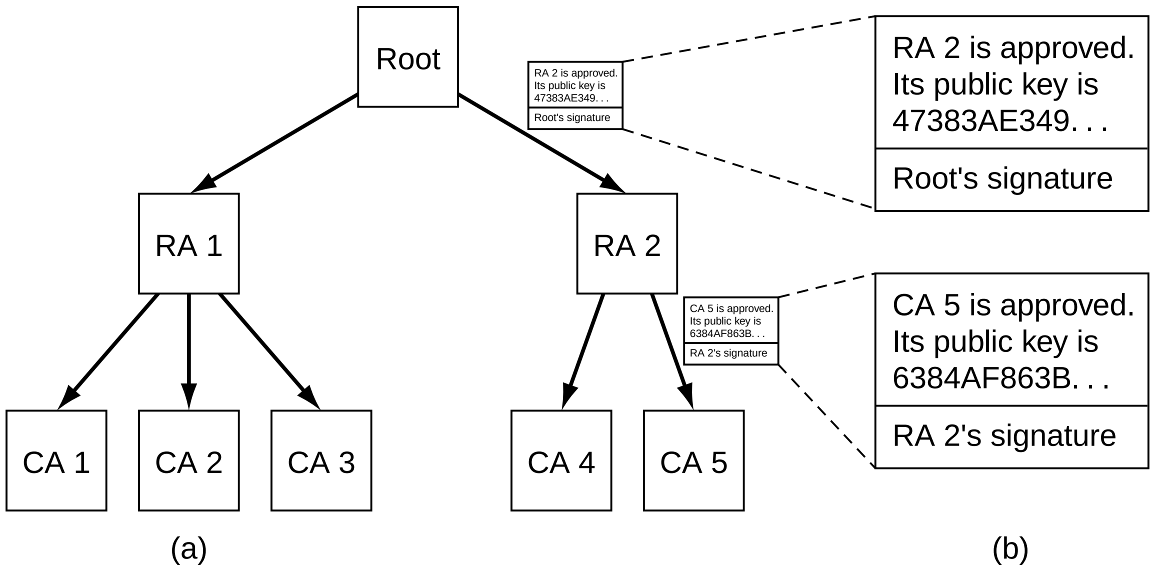


Fig. 8-26. (a) A hierarchical PKI. (b) A chain of certificates.

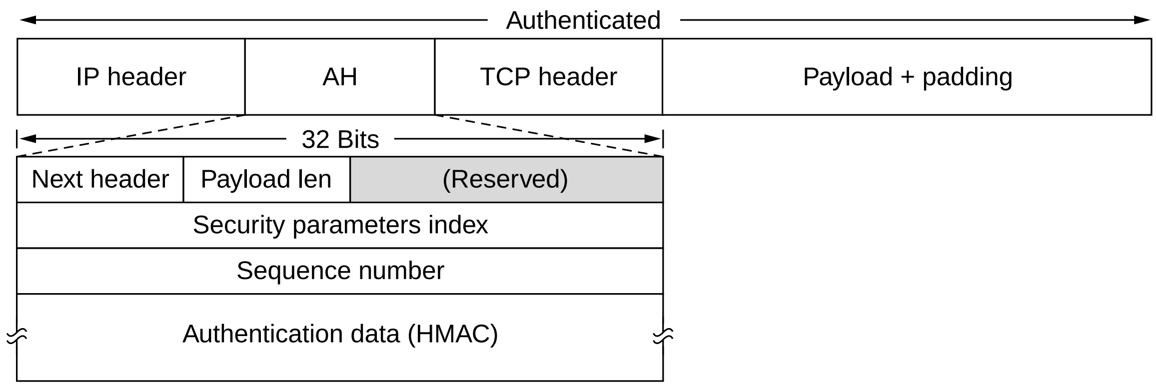


Fig. 8-27. The IPsec authentication header in transport mode for IPv4.

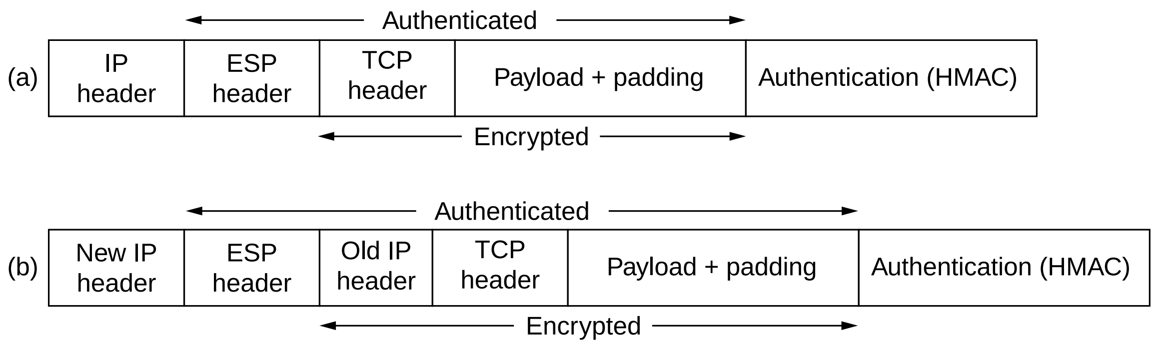


Fig. 8-28. (a) ESP in transport mode. (b) ESP in tunnel mode.

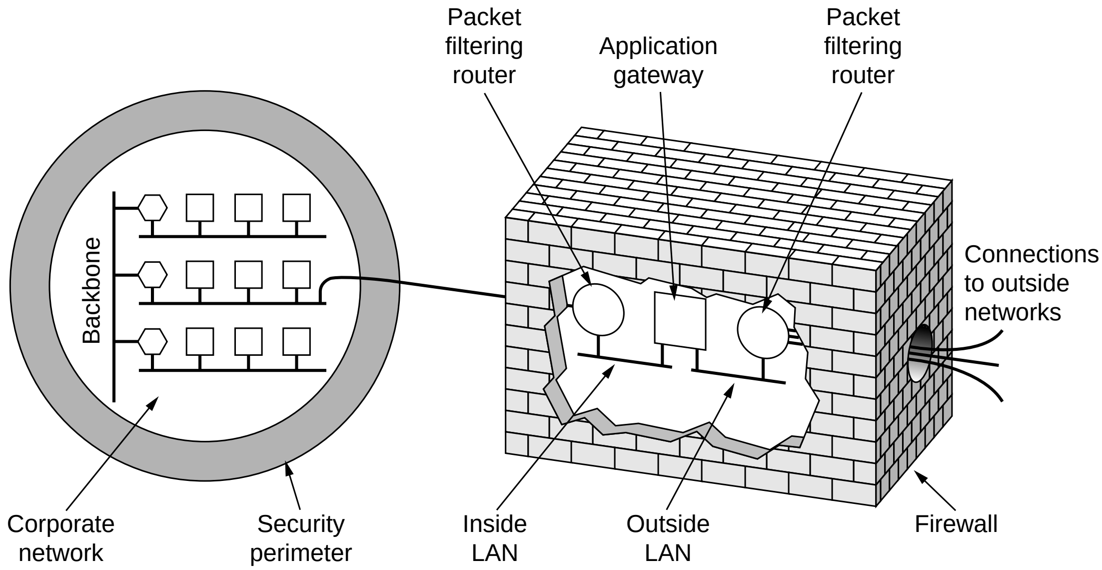


Fig. 8-29. A firewall consisting of two packet filters and an application gateway.

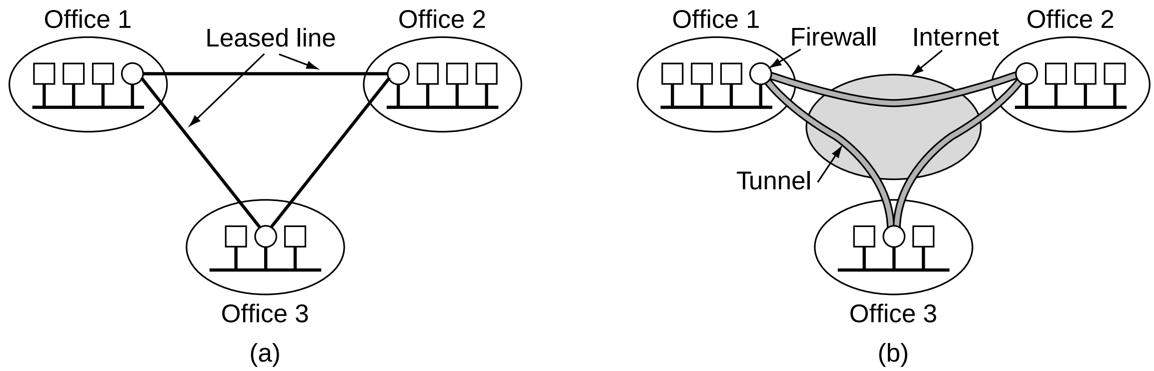


Fig. 8-30. (a) A leased-line private network. (b) A virtual private network.

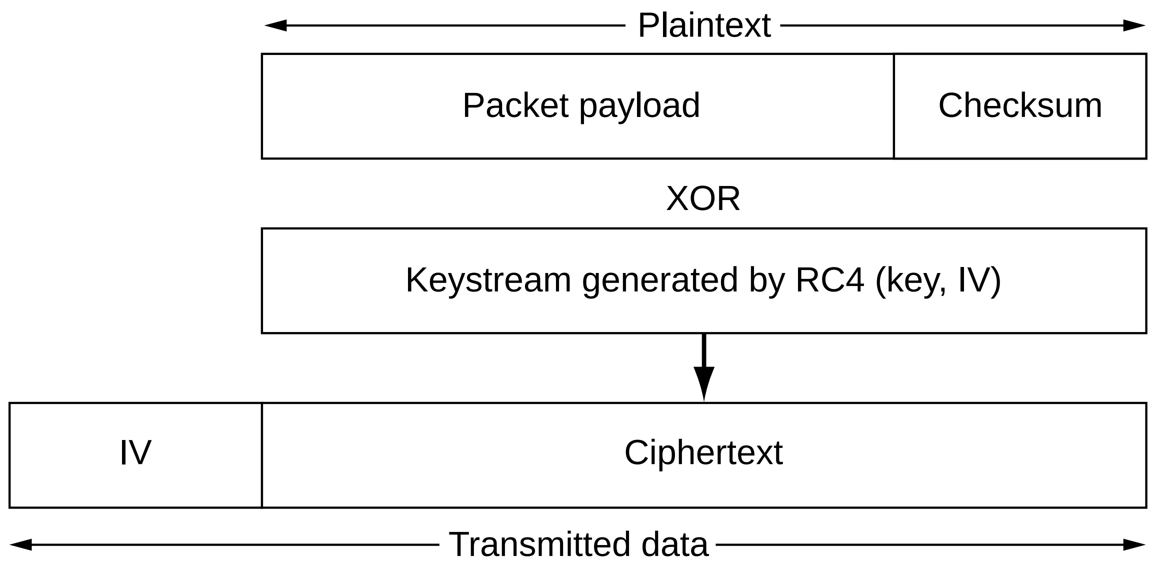


Fig. 8-31. Packet encryption using WEP.

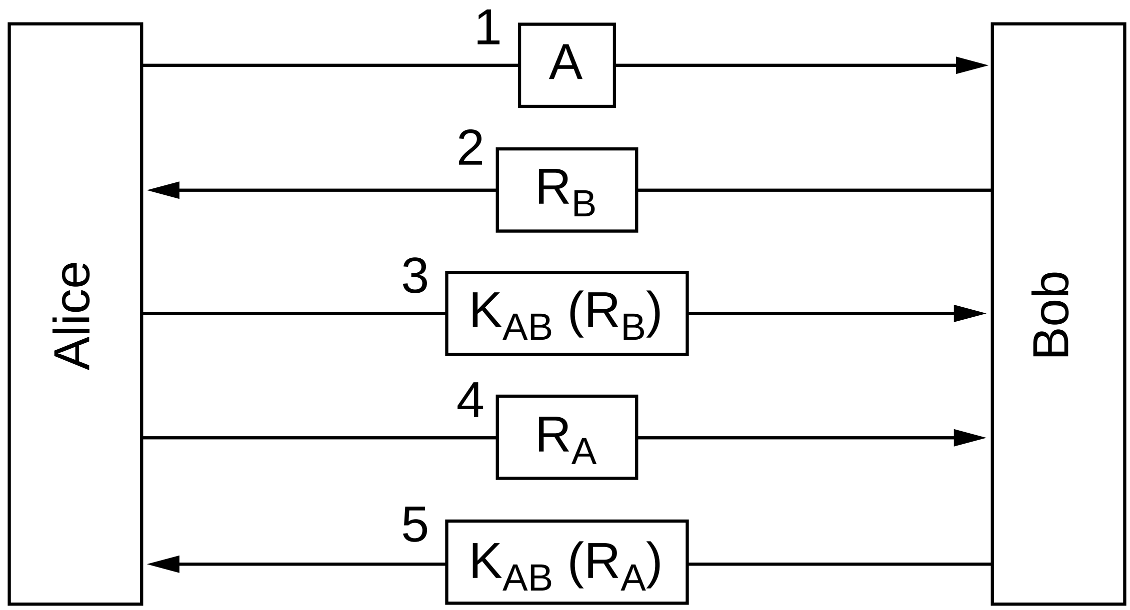


Fig. 8-32. Two-way authentication using a challenge-response protocol.

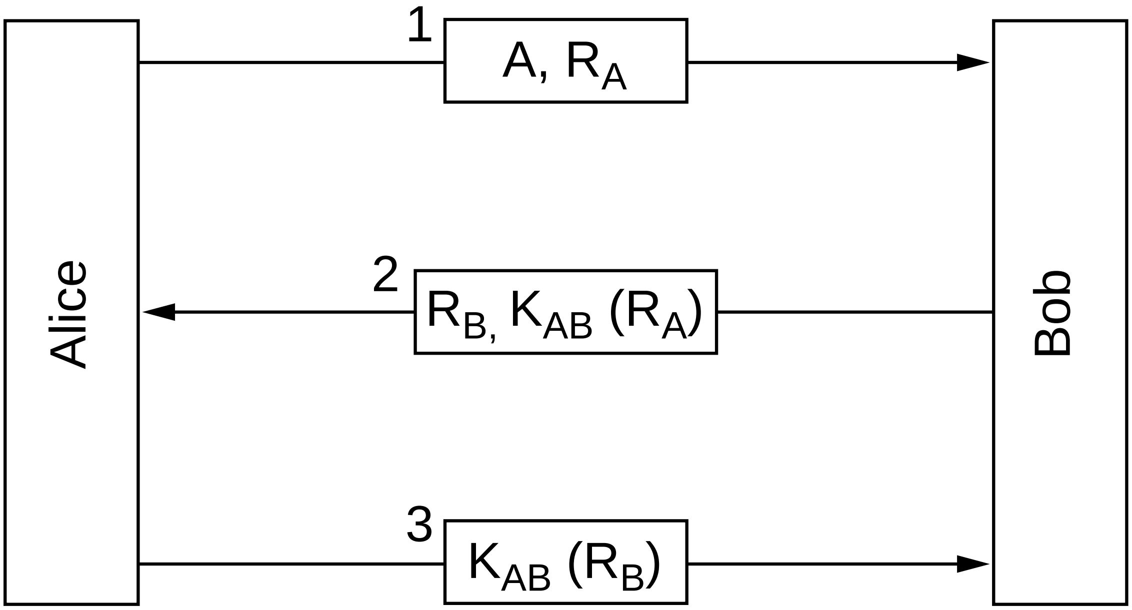


Fig. 8-33. A shortened two-way authentication protocol.

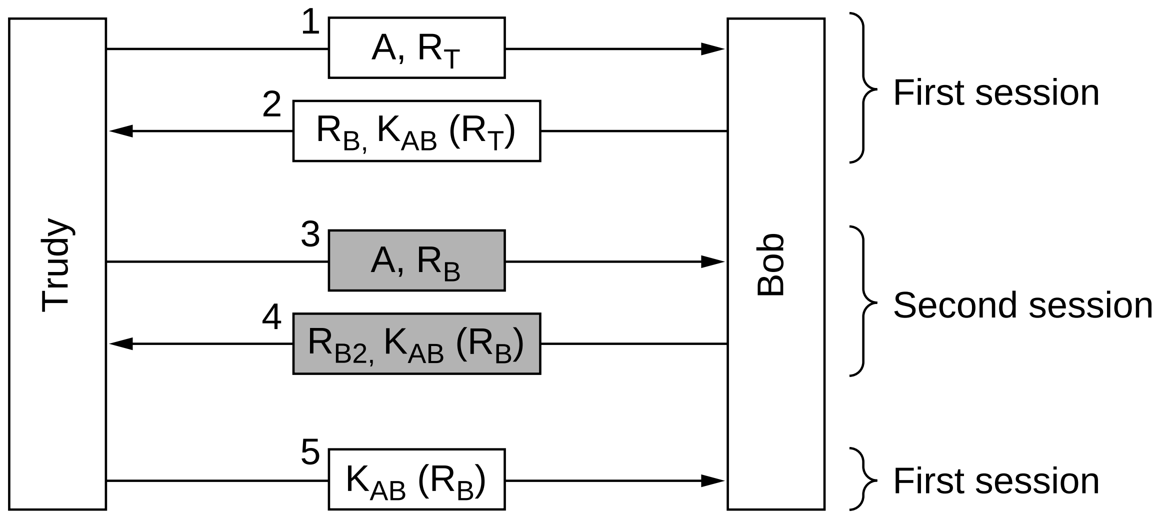


Fig. 8-34. The reflection attack.

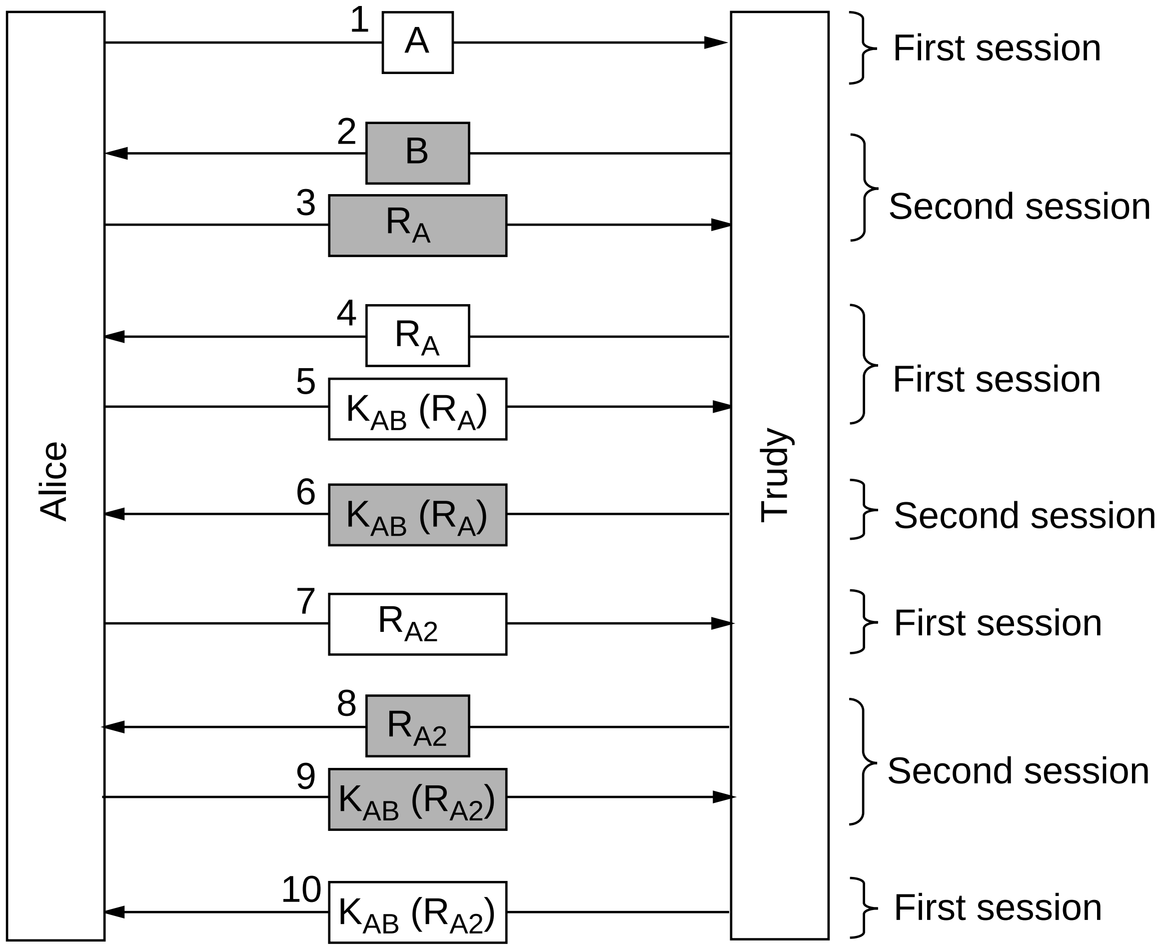


Fig. 8-35. A reflection attack on the protocol of Fig. 8-0.

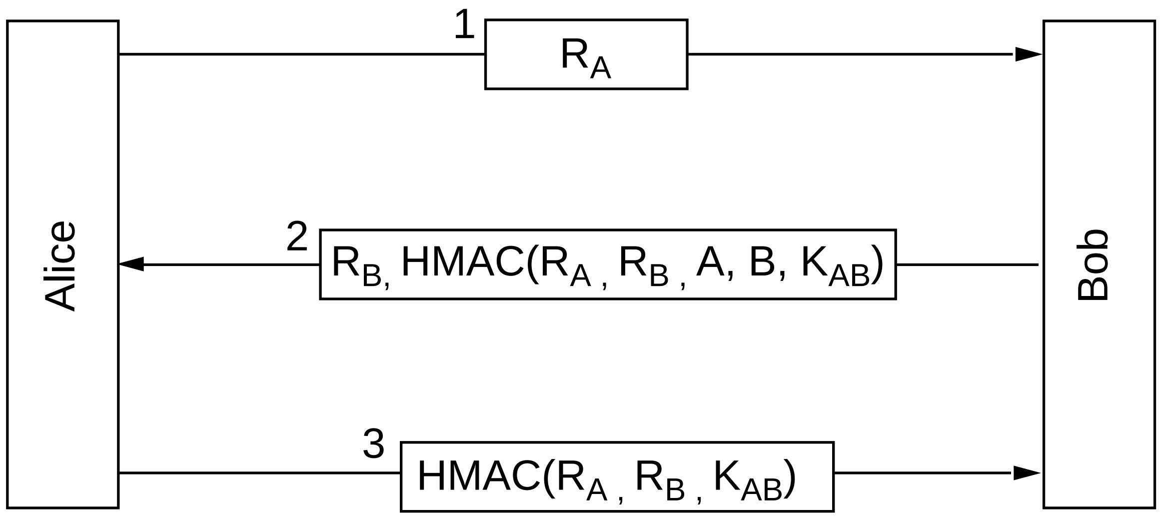


Fig. 8-36. Authentication using HMACs.

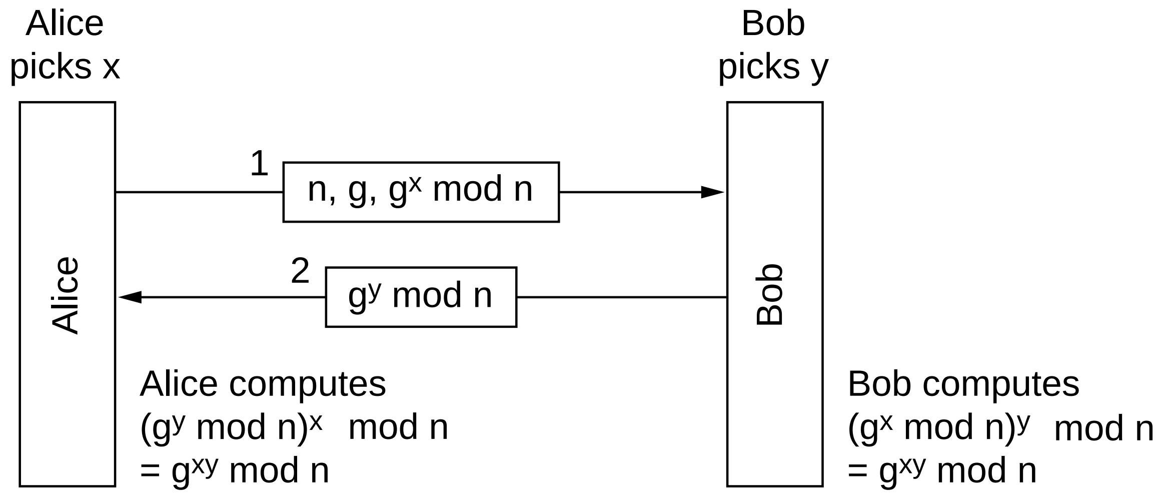


Fig. 8-37. The Diffie-Hellman key exchange.

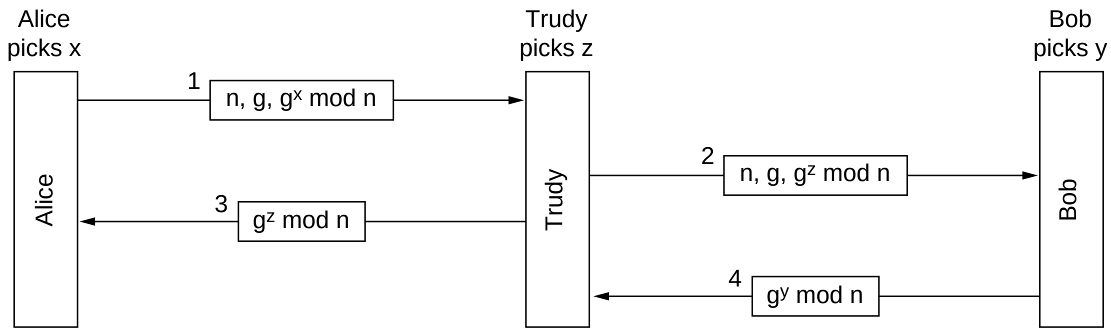


Fig. 8-38. The bucket brigade or man-in-the-middle attack.

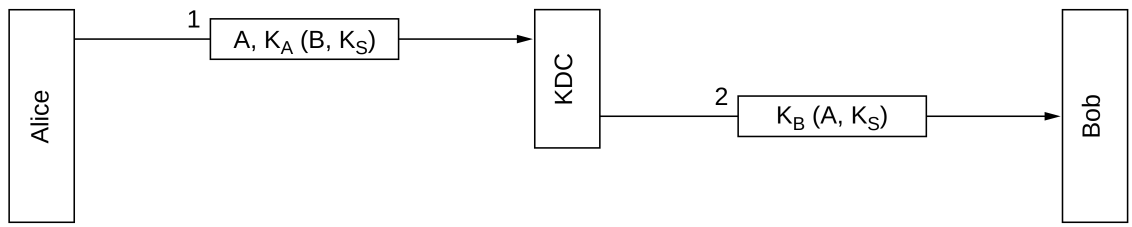


Fig. 8-39. A first attempt at an authentication protocol using a KDC.

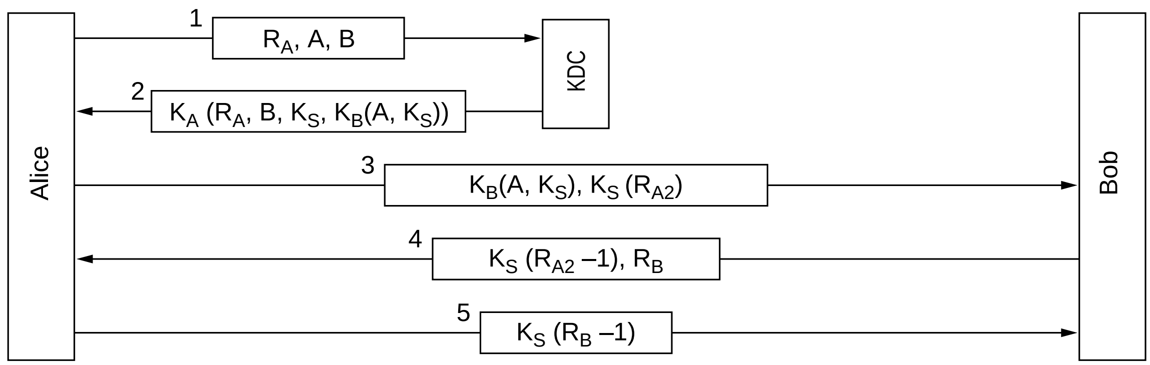


Fig. 8-40. The Needham-Schroeder authentication protocol.

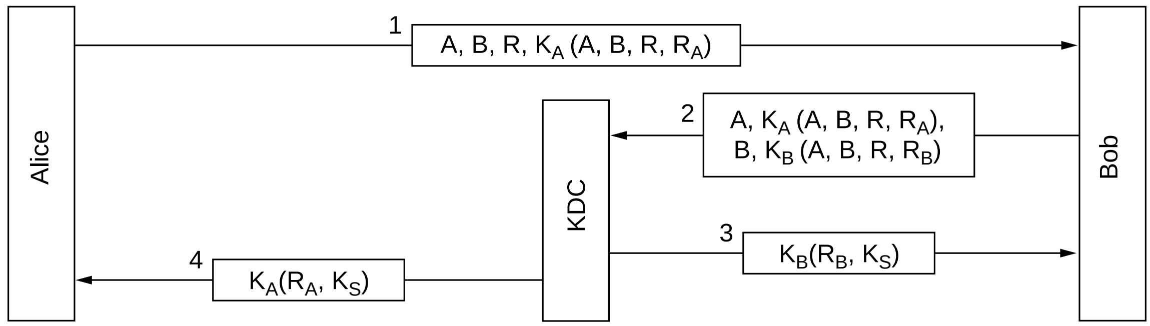


Fig. 8-41. The Otway-Rees authentication protocol (slightly simplified).

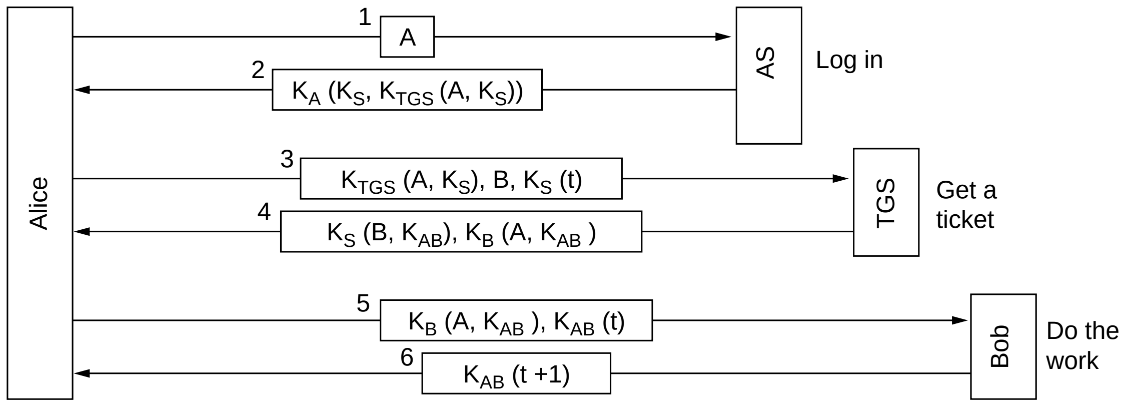


Fig. 8-42. The operation of Kerberos V4.

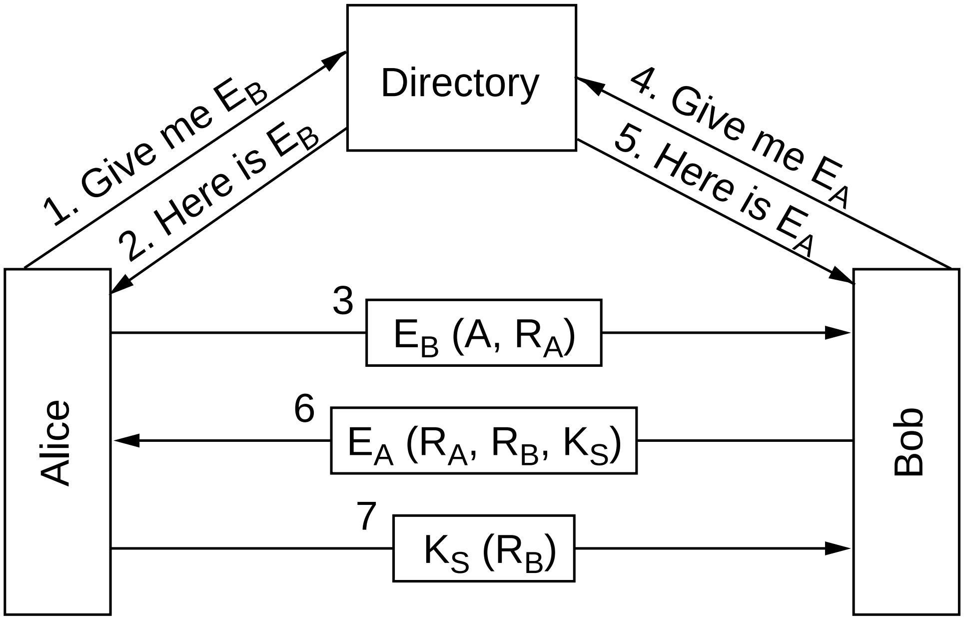


Fig. 8-43. Mutual authentication using public-key cryptography.

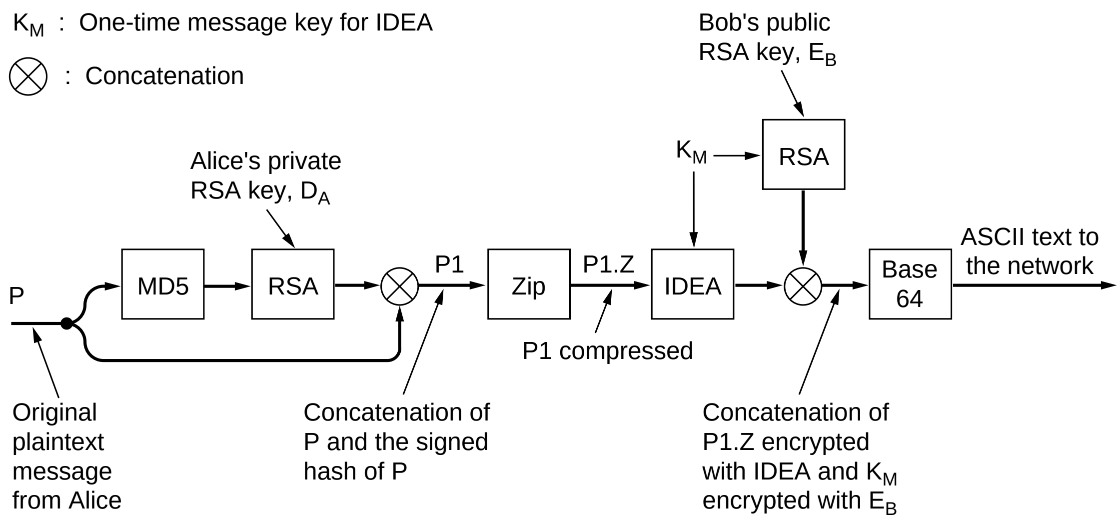


Fig. 8-44. PGP in operation for sending a message.

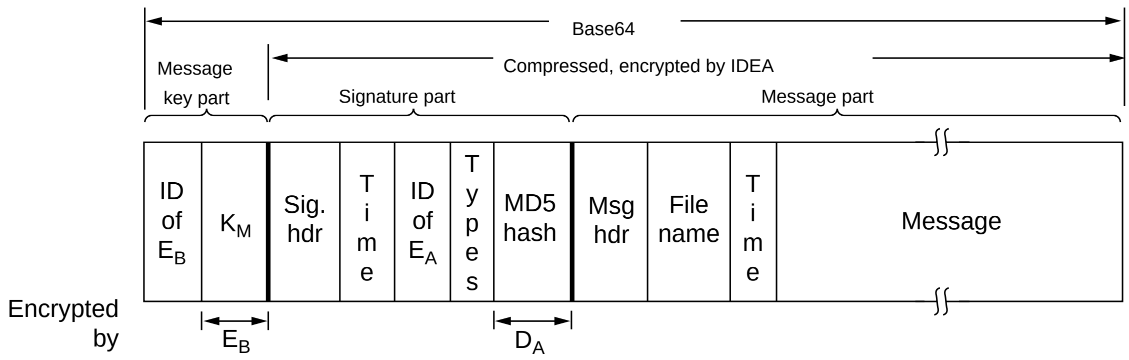


Fig. 8-45. A PGP message.

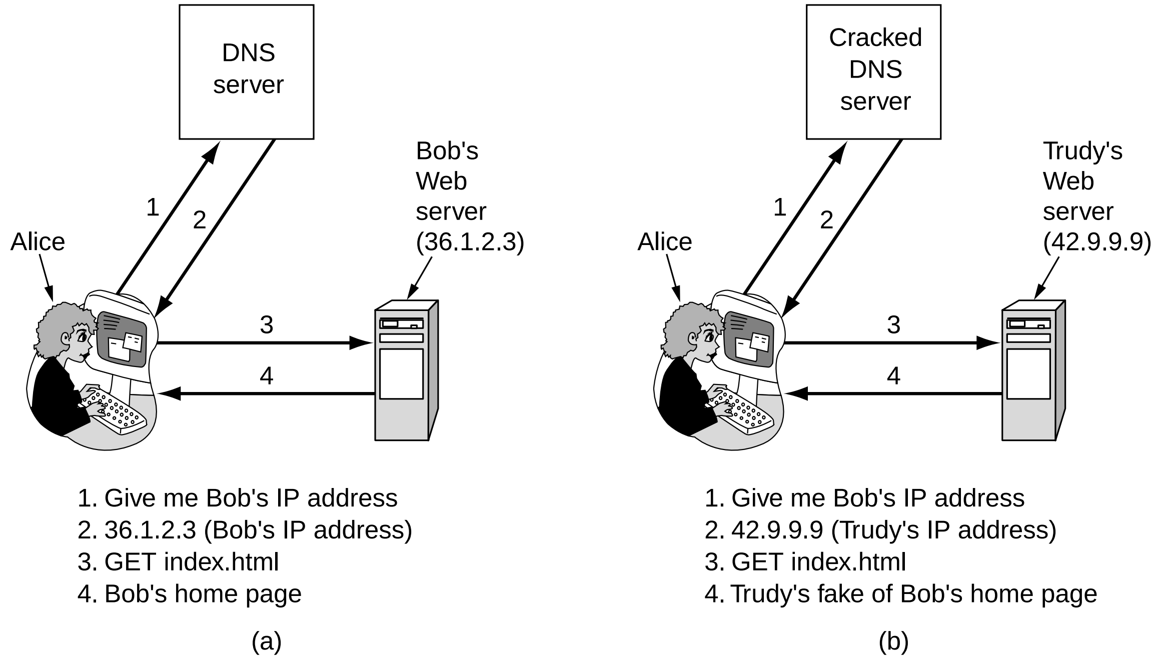
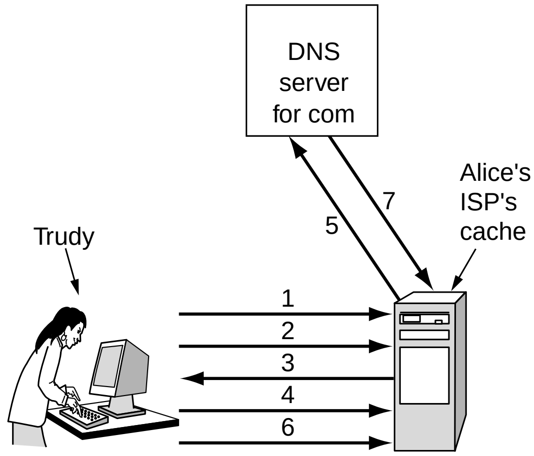


Fig. 8-46. (a) Normal situation. (b) An attack based on breaking into DNS and modifying Bob's record.



1. Look up foobar.trudy-the-intruder.com
   (to force it into the ISP's cache)
2. Look up www.trudy-the-intruder.com
   (to get the ISP's next sequence number)
3. Request for www.trudy-the-intruder.com
   (Carrying the ISP's next sequence number, n)
4. Quick like a bunny, look up bob.com
   (to force the ISP to query the com server in step 5)
5. Legitimate query for bob.com with seq = n+1
6. Trudy's forged answer: Bob is 42.9.9.9, seq = n+1
7. Real answer (rejected, too late)

Fig. 8-47. How Trudy spoofs Alice's ISP.

Domain name	Time to live	Class	Type	Value
bob.com.	86400	IN	A	36.1.2.3
bob.com.	86400	IN	KEY	3682793A7B73F731029CE2737D...
bob.com.	86400	IN	SIG	86947503A8B848F5272E53930C...

Fig. 8-48. An example RRSet for *bob.com*. The *KEY* record is Bob's public key. The *SIG* record is the top-level *com* server's signed hash of the *A* and *KEY* records to verify their authenticity.

$$\overbrace{\text{http://www.bob.com}}^{\text{Server}}\text{:}\overbrace{\text{2g5hd8bfjkc7mf6hg8dgany23xds4pe6}}^{\text{SHA-1 (Server, Server's Public key)}}\text{/}\overbrace{\text{photos/bob.jpg}}^{\text{File name}}$$

Fig. 8-49. A self-certifying URL containing a hash of server’s name and public key.

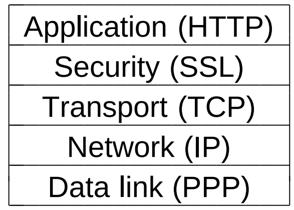


Fig. 8-50. Layers (and protocols) for a home user browsing with SSL.

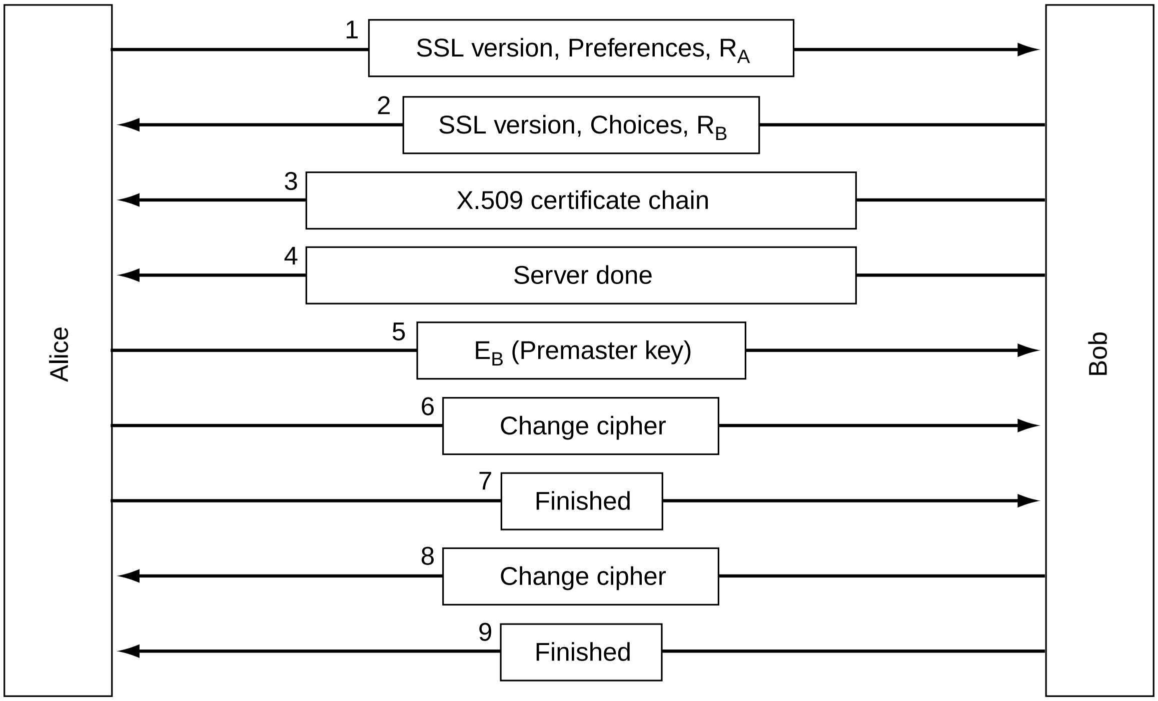


Fig. 8-51. A simplified version of the SSL connection establishment subprotocol.

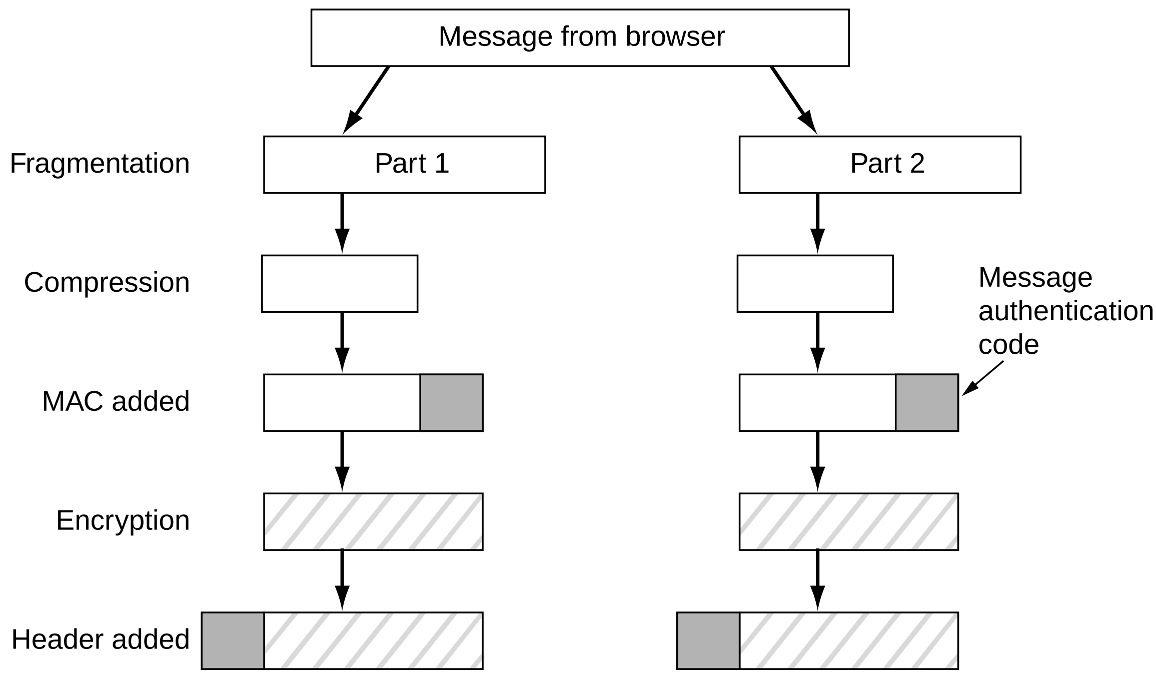


Fig. 8-52. Data transmission using SSL.

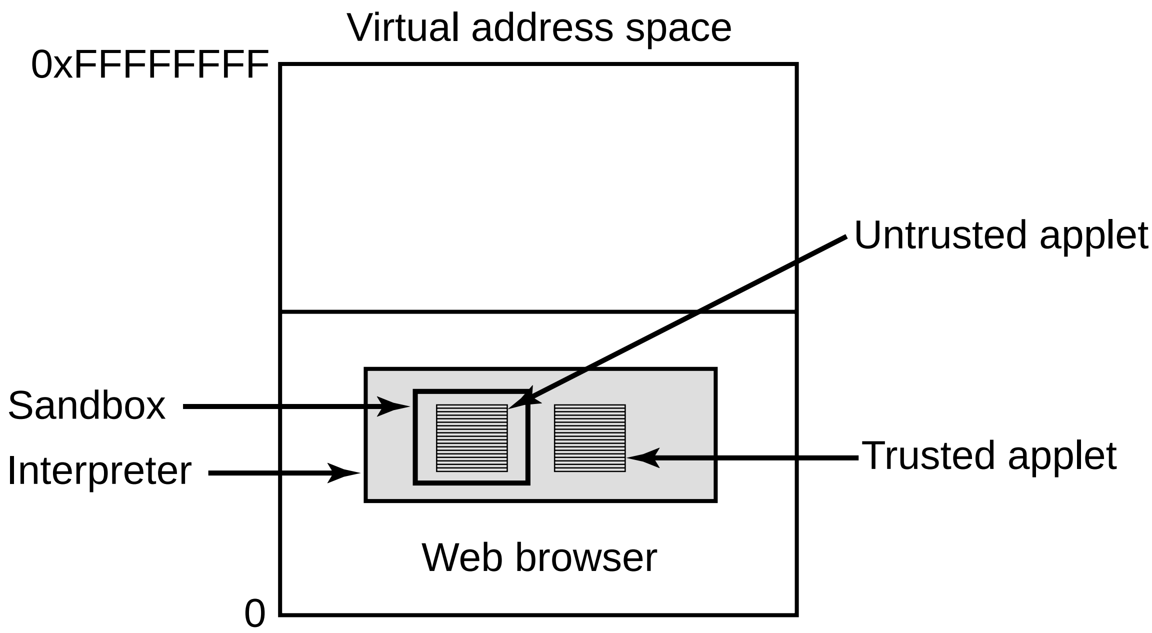


Fig. 8-53. Applets can be interpreted by a Web browser.

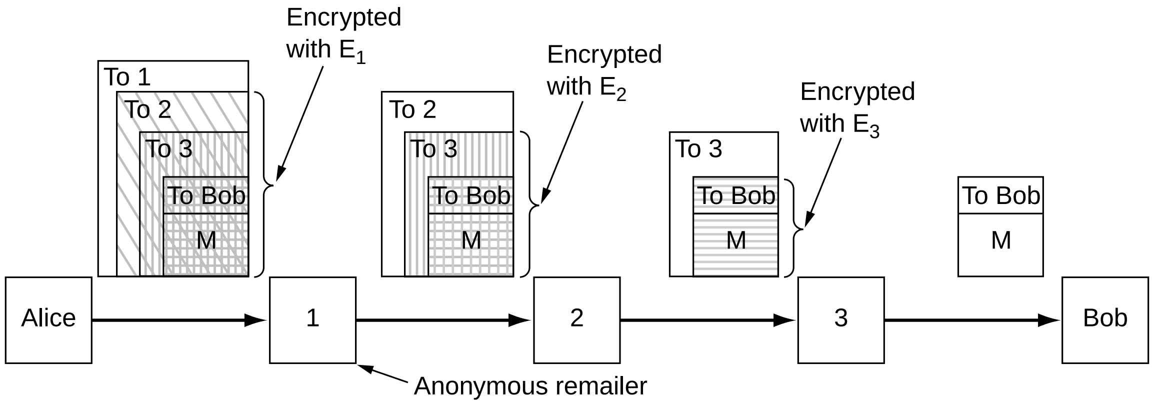
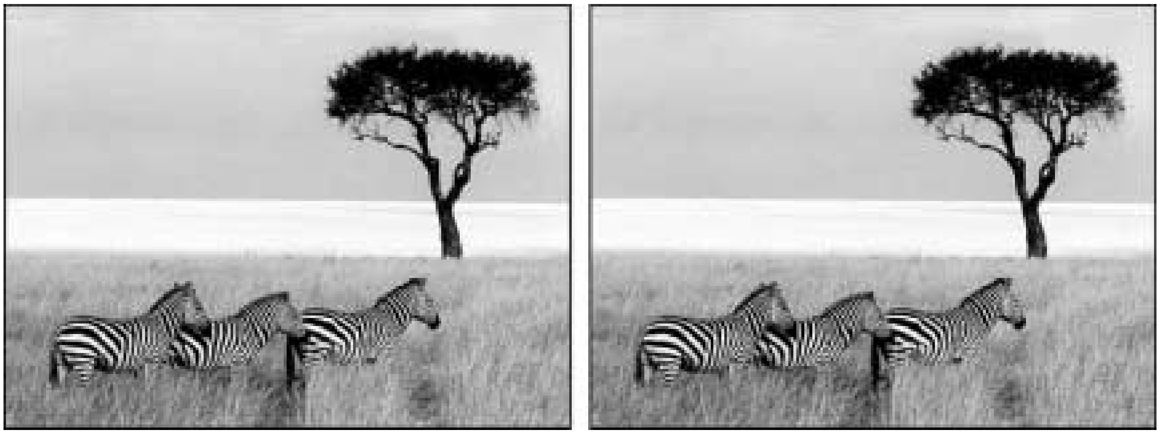


Fig. 8-54. How Alice uses 3 remailers to send Bob a message.



(a) (b)

Fig. 8-55. (a) Three zebras and a tree. (b) Three zebras, a tree, and the complete text of five plays by William Shakespeare.