COMPSCI 314 S1 C

Data Communications Fundamentals

Lecture Slides, Set #4

Clark Thomborson

31 March 2006

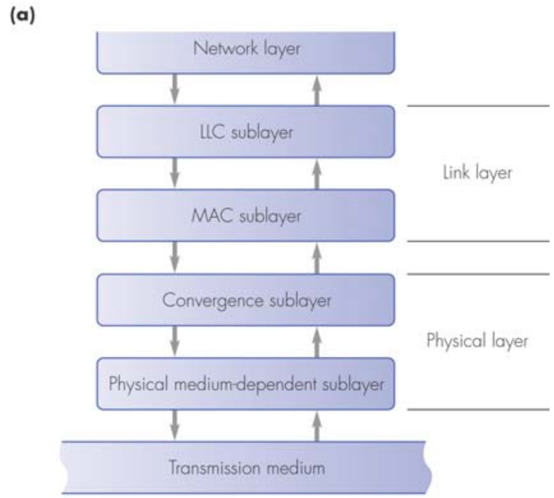


Figure 3.17 LAN protocols: (a) protocol framework

(b)

IEEE	802.1	Station management
	802.1d	Transparent bridges
	802.1Q	Virtual LANs
	802.2	Logical link control (LLC)
	802.3	CSMA/CD (Ethernet) bus
	802.3u	Fast Ethernet
	802.3x	Hop-by-hop switch flow control
	802.3z	Gigabit Ethernet
	802.3ae	10 Gigabit Ethernet

Figure 3.17 LAN protocols: (b) examples

IEEE 802 Standard Family

- IEEE 802.1 Bridging & Management
- IEEE 802.2: Logical Link Control
- IEEE 802.3: CSMA/CD Access Method
- IEEE 802.5: Token Ring Access Method
- IEEE 802.11: Wireless
- IEEE 802.15: Wireless Personal Area Networks
- IEEE 802.16: Broadband Wireless Metropolitan Area Networks
- IEEE 802.17: Resilient Packet Rings

Source: http://standards.ieee.org/getieee802/portfolio.html

IEEE 802 Standard LANs

IEEE 802 Standard Local Area Networks are

- “optimized for a moderate-sized geographic area, such as a single office building, a warehouse, or a campus;
- … a peer-to-peer communication network that enables stations to communicate directly on a point-to-point, or point-to-multipoint, basis without requiring them to communicate with any intermediate switching nodes.
- LAN communication takes place at moderate-to-high data rates, and with short transit delays, on the order of a few milliseconds or less.
- A LAN is generally owned, used, and operated by a single organization.
- This is in contrast to Wide Area Networks (WANs) that interconnect communication facilities in different parts of a country or are used as a public utility.”

Source: http://standards.ieee.org/getieee802/download/802-2001.pdf

IEEE 802 Standard MANs

- “A MAN is optimized for a larger geographical area than is a LAN, ranging from several blocks of buildings to entire cities.
- As with local networks, MANs can also depend on communications channels of moderate-to-high data rates.
- A MAN might be owned and operated by a single organization, but it usually will be used by many individuals and organizations.
- MANs might also be owned and operated as public utilities.
- They will often provide means for internetworking of local networks.”

Source: http://standards.ieee.org/getieee802/download/802-2001.pdf

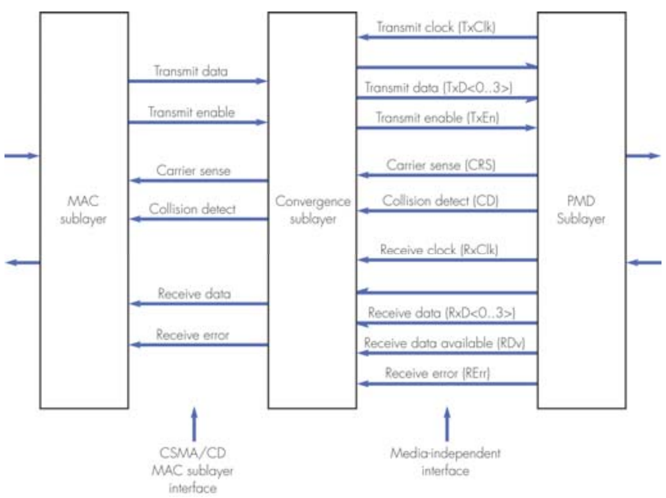
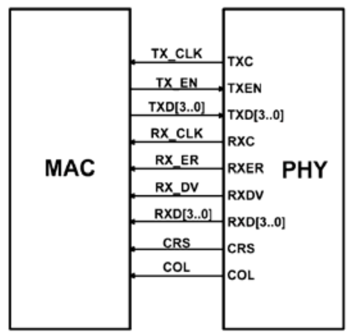


Figure 3.18 Fast Ethernet media-independent interface

Do you think there are enough signals on these interfaces?

Media Independent Interface (MII)

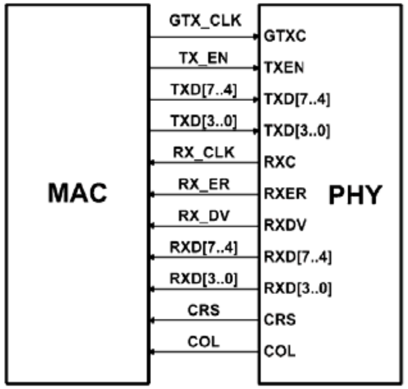


- Source: datasheet for Realtek RTL8212N Integrated 10/100/1000 Single/Dual Ethernet Transceiver

ftp://202.65.194.18/cn/phy/rtl8212rtl8212nrtl8211n/RTL8212_RTL8212N_RTL8211N_DataSheet_1.2.pdf

- How does this compare with Figure 3.18?

Gigabit Media Independent Interface (GMII)



- What is the clock rate on this interface?
- Can you guess (remember) what the acronyms mean?

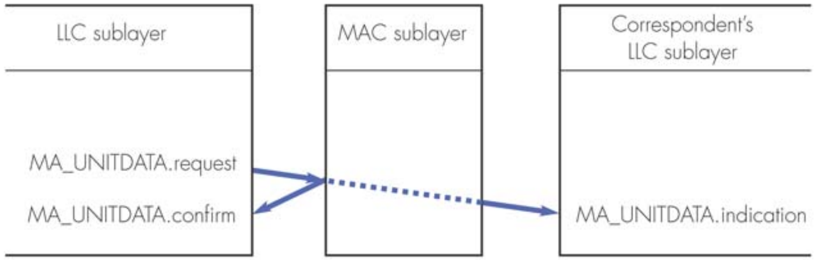


Figure 3.19 MAC user service primitives for CSMA/CD

Do you remember where time sequence diagrams were defined in your text?

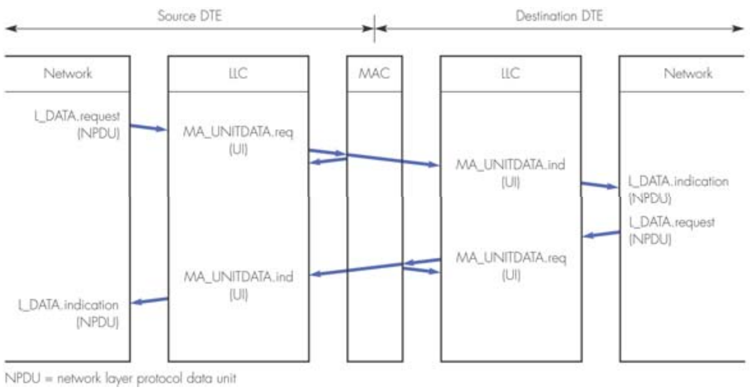


Figure 3.20 LLC/MAC sublayer interactions

Does this give you a better understanding of protocol layers?

Security 101

Data security: CIA

- Confidentiality: no unauthorised user can read
- Integrity: no unauthorised user can write
- Availability: all authorised users can read and write

Important security functions:

- Authentication: who is trying to do this?
  - UserID X can't impersonate userID Y.
- Authorisation: Who is permitted to do which operations to what?
  - Users can't add anything to their list of authorised actions.
- Auditing: what has happened on this system?
  - System administrators can investigate problems.
- Identification: what human is supposed to be logged in as userID X?
  - People can be held responsible for actions authorised by userIDs.
- Non-repudiation: did this user really do that?
  - Users can be held accountable for their actions.

How does this list compare with section 10.1 of your text?

To learn more: Lampson, "Computer Security in the Real World", *IEEE Computer 37:6*, June 2004.

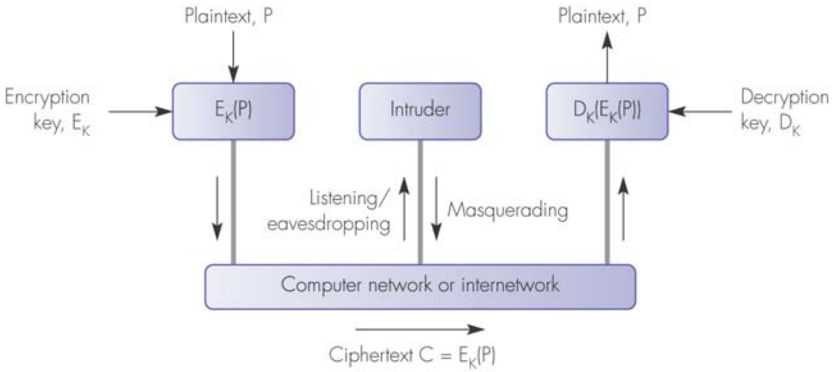


Figure 10.1 Data encryption terminology

Are there any attacks not shown here? (Hint: think CIA.)

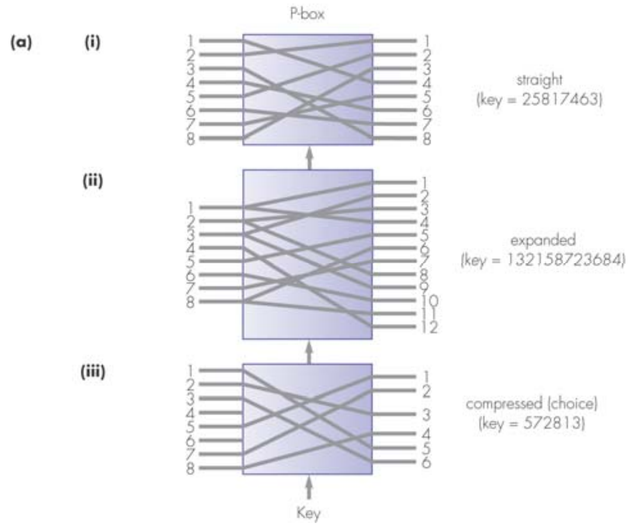


Figure 10.2 Product cipher components: (a) P-box examples

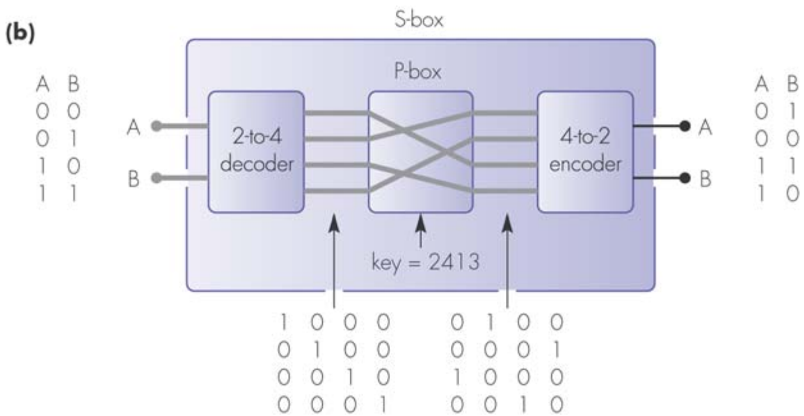


Figure 10.2 Product cipher components: (b) S-box example

How many different keys are there? How many bits of key information?

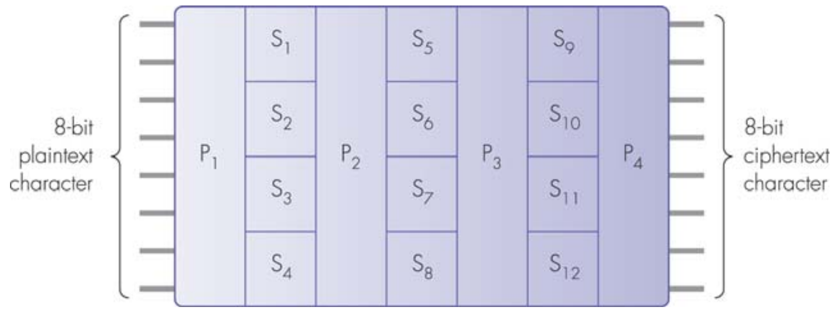


Figure 10.3 Example of a product cipher

Are four 2-bit S-boxes equivalent to one 8-bit S-box?

How many bits of key material is required to control this cipher?

One Step in a Feistel Cipher

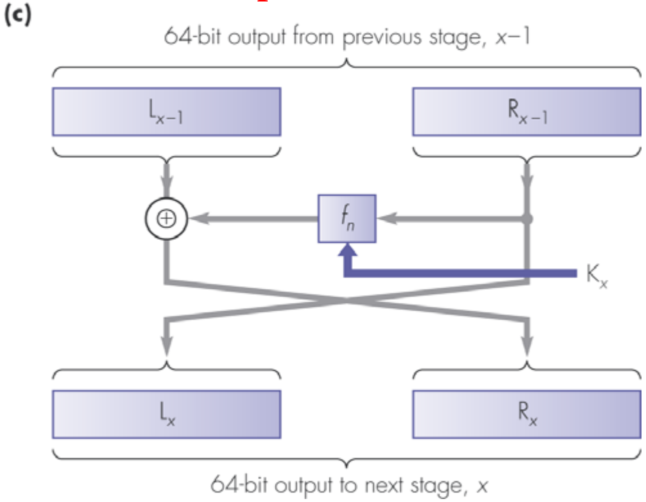


Figure 10.4 DES algorithm principles: (c) substitution operation

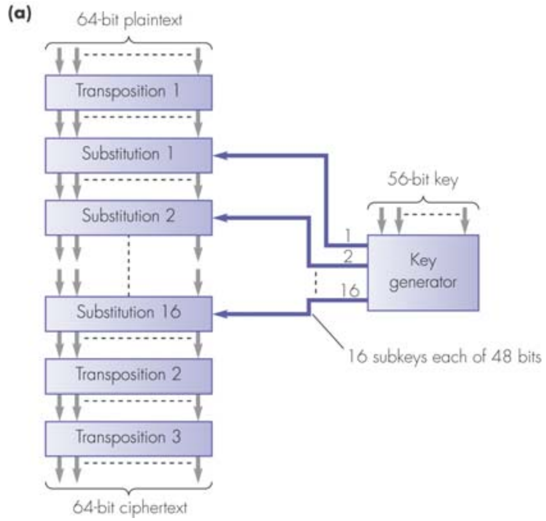


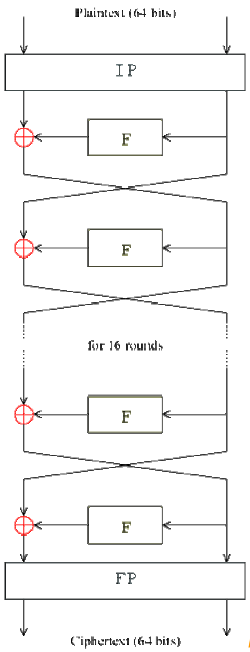
Figure 10.4 DES algorithm principles: (a) overall schematic.

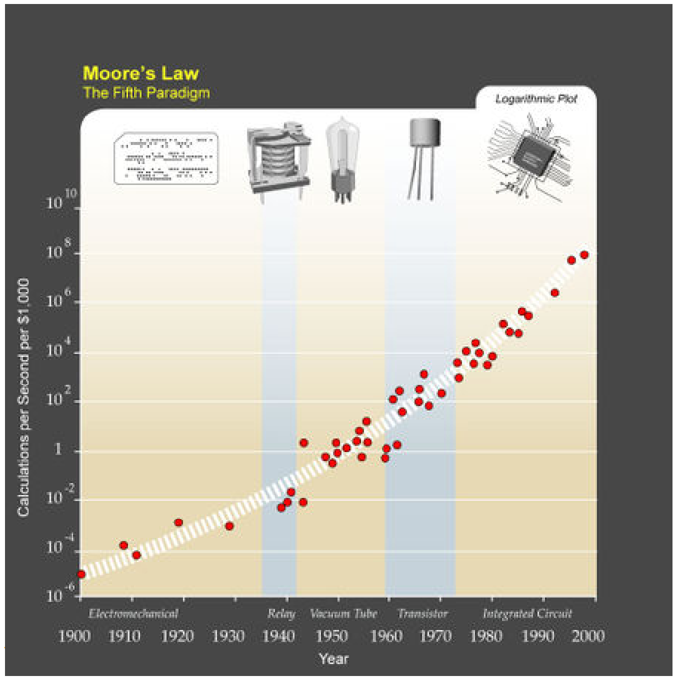
Note: Transpositions 1, 2, and 3 are fixed permutations (not keyed).

Why can't we combine Transpositions 2 and 3?

DES

- IP = Initial permutation
- F = Feistel function (keyed)
- FP = Final permutation = IP^{-1}
- Source: http://en.wikipedia.org/wiki/Data_Encryption_Standard, version 17:42, 24 March 2006.
- Do you believe this version of Wikipedia, or your textbook?
- Only 56 bits of key is required: is this a feature or a bug?
- In July 1998, the EFF's DES cracker (Deep Crack) broke a DES key in 56 hours. Cost: $250,000.





Triple DES

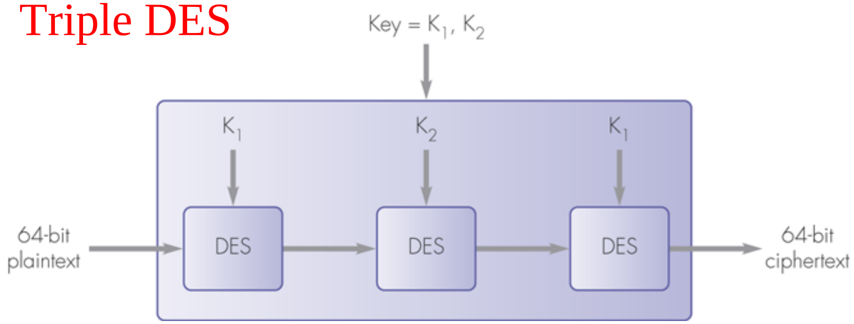


Figure 10.5 Triple DES schematic

- 25 October 1999: 3DES preferred by NIST; single DES permitted only in legacy systems.
- 26 November 2001: The Advanced Encryption Standard is published.
- 19 May 2005: NIST withdraws DES standard.

Rivest, Shamir, Adleman

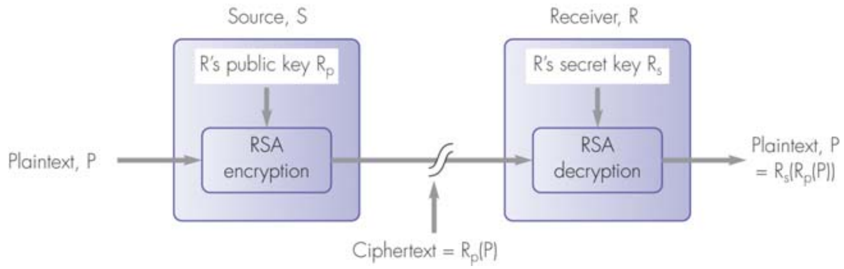


Figure 10.8 RSA schematic

Two different keys! Everyone knows your public key.

Your textbook spells the third name "Adelman". Who's right?

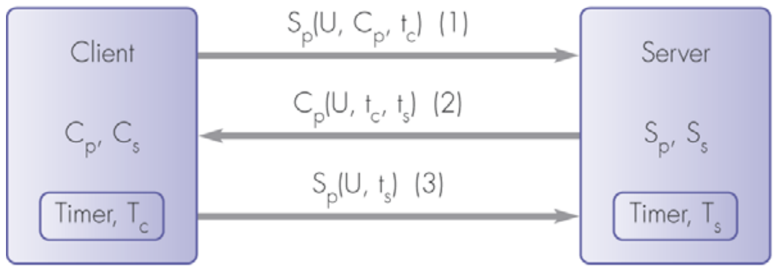
Hint: http://www.rsasecurity.com/rsalabs/node.asp?id=2083.

Nonrepudiation

- Any public-key cryptographic system, e.g. RSA, can be used for non-repudiable messaging.
- Encrypt a plaintext message P with your own secret key: $S_s(P)$
- "Everyone" can decrypt this message – they merely need to know your public key, which is not a secret.
- Only you (or people who know your secret key ;-) can efficiently compute $S_s(P)$, from the value of P and your public key S_p.
  - Don't share your cryptographic keys!
  - But… if you don't share your keys, what happens if you lose them?!
  - Key management is *very* difficult in practice.
- See Figure 10.9a: you can send a secret non-repudiable message $R_p(S_s(P))$, if you know the recipient's public key.

Efficient Nonrepudiation

- RSA was the first practical public key cryptosystem.
- Even with hardware acceleration, it is still unacceptably slow for many applications.
- The throughput of an RSA-encrypted message is approx 1 MB/s on a modern PC,
  - plus a fraction of a second for an initial Diffie-Hellman key-exchange, in cases where public keys aren't available.
  - Approx. 8 seconds to transfer 1 MB to a PDA. Source: https://www.cs.tcd.ie/publications/tech-reports/reports.03/TCD-CS-2003-46.pdf.
- Use a message digest algorithm such as MD5 or SHA – these produces a short (e.g. 128-bit) hash "signature" of a message.
- See Figure 10.9: send both P and $S_s(MD(P))$.



C_p, C_s = client public/secret key S_p, S_s = server public/secret key

U = client user name t_c, t_s = client/server time-stamp

Figure 10.10 User authentication using a public key scheme

- Has this user proved their identity to the server? (Authentication)
- Is this user allowed to use this service? (Authorization)
- Can an attacker use a copy of message 3 to gain service? (Eavesdrop, then Replay; or Intercept, then Inject)

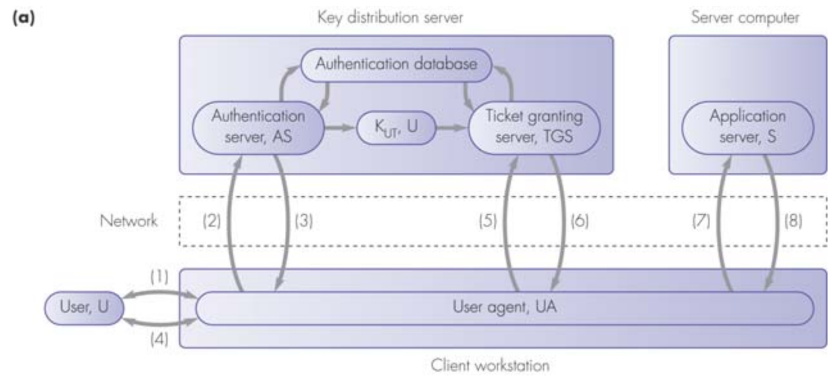
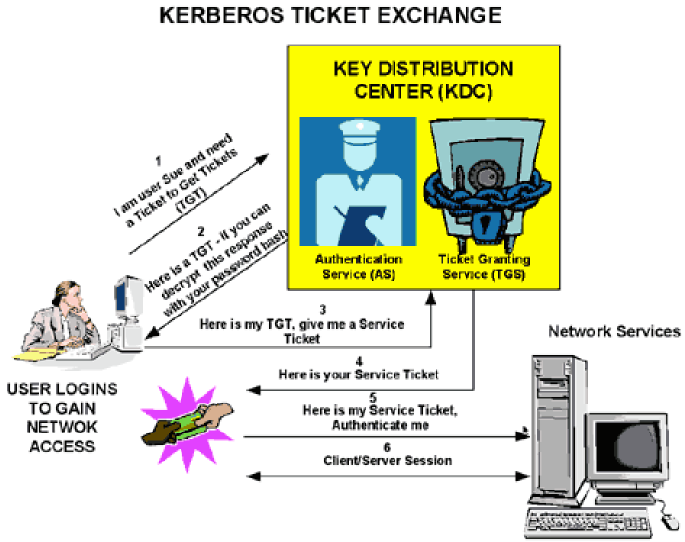


Figure 10.11 User authentication using Kerberos: (a) terminology and message exchange

- What is an advantage of separating the KDS from the application server?
- Do you see any disadvantage?

Maybe a cartoon will help…



Source: http://www.microsoft.com/technet/prodtechnol/windows2000serv/maintain/security/kerberos.mspx

(c)

	Direction	Message
(1)	U ↔ UA	User name, U
(2)	UA → AS	(U, T, n_1)
(3)	AS → UA	$K_U(K_{UT}, n_1); T_{UT}$
(4)	U ↔ UA	User password, K_U
(5)	UA → TGS	$K_{UT}(U, t); T_{UT}, S, n_2$
(6)	TGS → UA	$K_{UT}(K_{US}, n_2); T_{US}$
(7)	UA → S	$K_{US}(U, t); T_{US}, n_3$
(8)	S → UA	$K_{US}(n_3)$

$K_{UT}/K_{US}(U, t)$ are both authenticators and t is a time-stamp

(b)

K_U = The private key of the user – the user password
K_T = The private key of the TGS
K_S = The private key of the application server
K_{UT} = A session key to encrypt UA ↔ TGS dialog units
K_{US} = A session key to encrypt UA → S dialog units

TGS ticket, $T_{UT} = K_T(U, T, t_1, t_2, K_{UT})$
Application server ticket, $T_{US} = K_S(U, S, t_1, t_2, K_{US})$
t_1, t_2 = start, end of ticket lifetime

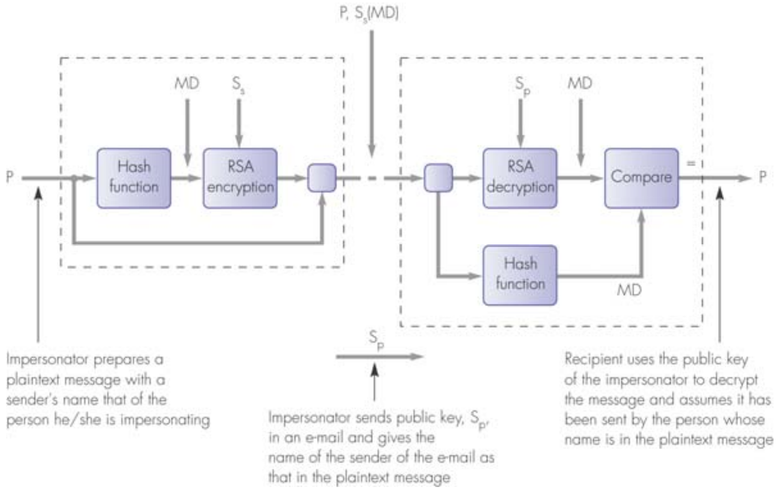


Figure 10.12 A possible threat when using a public key system

- It's surprisingly hard to be certain about who owns a public key.
- In a public key directory, who is "John Smith"? (Identification!)
- Who is Clark.Thomborson@gmail.com?