COMPSCI 314 S2 T

Data Communications Fundamentals

Lecture Slides, Set #5

Clark Thomborson

10 September 2007

What sort of a student are you?

- “C” student: attends most lectures, reads the textbook only when it seems necessary to do homework, learns terminology and basic concepts.
- “B” student: pays attention during lectures, reads textbook carefully *before* the lecture on that topic, completes homework, develops a competent understanding of the material emphasized in homework and lectures.
- “A” student: asks and answers their own questions, using outside sources if necessary; develops a critical and appreciative understanding of the textbook and the lecturer; works creatively with what they have learned.

Critical and Appreciative Views

- “I think networks, as taught in COMPSCI 314, are a boring and bewildering mess of detail. Where are the underlying scientific concepts?”
- “I want to learn how to set up and administer a network for a small office – or for a large corporation!”
- “I want to learn enough to get hired as a network administrator when I graduate.”
- What is your current view of this class? Can you develop a view that will motivate you to be an “A” student?

Security 101 (not in either Shay or Halsall)

Properties of secure data: **CIA**

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

These properties apply to systems as well as to data.

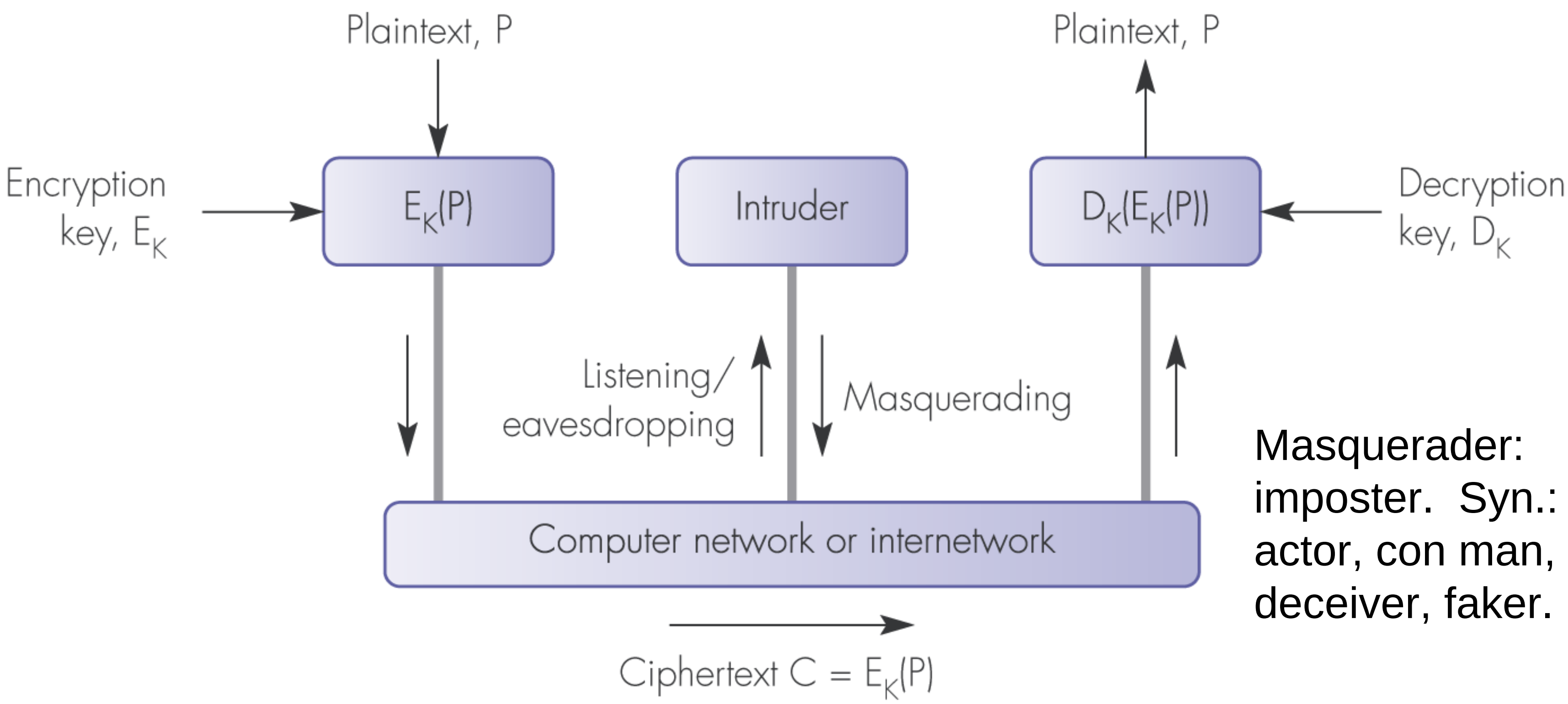
- Confidentiality: information about a secure system is revealed on a "need to know" basis.
  - All users of a system need to know something about it...
- Integrity: no unauthorised changes to the system!
- Availability: authorised people *can* make changes.

Security Functions

The Gold Standard, and some additional functions:

- Authentication: are you who you say you are?
  – All claims to identity can be verified.
- Authorisation: who is permitted to do which operations to what?
  – Users can't increase their own authority.
- Auditing: what has happened on this system?
  – System administrators can investigate problems.
- Identification: what human (or object) is this?
  – This is often confused with authentication (a proof of an identity) or with authorisation (a decision to allow an activity).
- Non-repudiation: can you prove this event really did happen?
  – Users can be given receipts which can be used as proofs of prior activity.

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



Masquerader: imposter. Syn.: actor, con man, deceiver, faker.

Figure 10.1 (Halsall) Data encryption: basic terminology, and two attacks that it can prevent.

Q1. An eavesdropper violates the Confidentiality property. What property is violated by a masquerader?

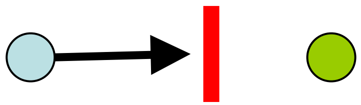
Q2. What other attacks are possible?

Attacks on a Communication Link

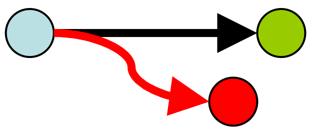
1. *Interception* or *eavesdropping*: an attacker reads a message;
2. *Modification* or *man in the middle*: an attacker changes a message;
3. *Interruption* or *denial of service*: an attacker prevents delivery;
4. *Fabrication* or *spoofing*: an attacker injects a message.

To learn more: Pfleeger, *Security in Computing*, 3rd Edition, Prentice-Hall, 2003.

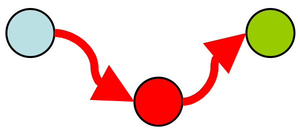
Desired message flow



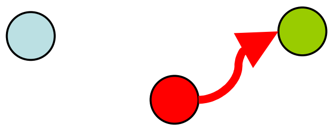
Interruption



Interception



Modification



Fabrication

How do these attacks relate to our goals and functions?

Goals of the Attacker; Vulnerabilities

- Interception, Modification, and Interruption are attacks on the CIA properties of the messaging system.
  - Fabrication can be considered an attack on integrity, in which the attacker modifies the "null message" of an idle channel.
  - Sometimes the CIA properties are violated unintentionally, and sometimes the system fails to enforce these properties.
  - Design errors and operator errors are usually considered to be security faults, even though there is no "attack".
  - An attacker's goal is to gain advantage (or cause damage) by compromising the C, I, and/or A of some secure system.
- The attacker must find, and exploit, a vulnerability in the Identification, Authentication, or Authorisation functions of the messaging system.
- An attack might be detected by the Auditing function.
- If someone is suspected of an attack, the Non-repudiation function will give evidence for (or against) them.
- Security is an immature field. Different authors use different taxonomies and terminologies.

Cryptography 101

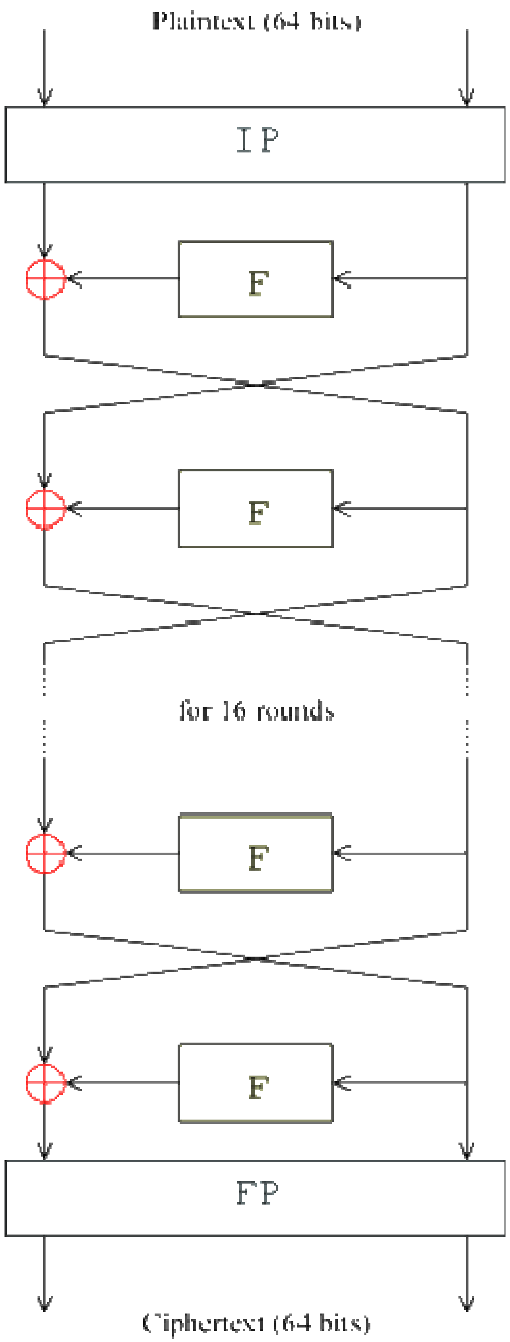
- Cryptology: the art (science) of communication with secret codes. Includes
  - Cryptography: the making of secret codes.
  - Cryptanalysis : the “breaking” of codes, so that the plaintext of a message is revealed.
- Cryptanalysis is pretty easy (for an expert ;-) unless the code has both
  - Substitutions of symbols for letters (or for sequences of letters) in the plaintext
  - Transpositions, so that the code symbols appear in a different order than the letters in the plaintext.
- It is very difficult (even for an expert) to devise a secure encryption.

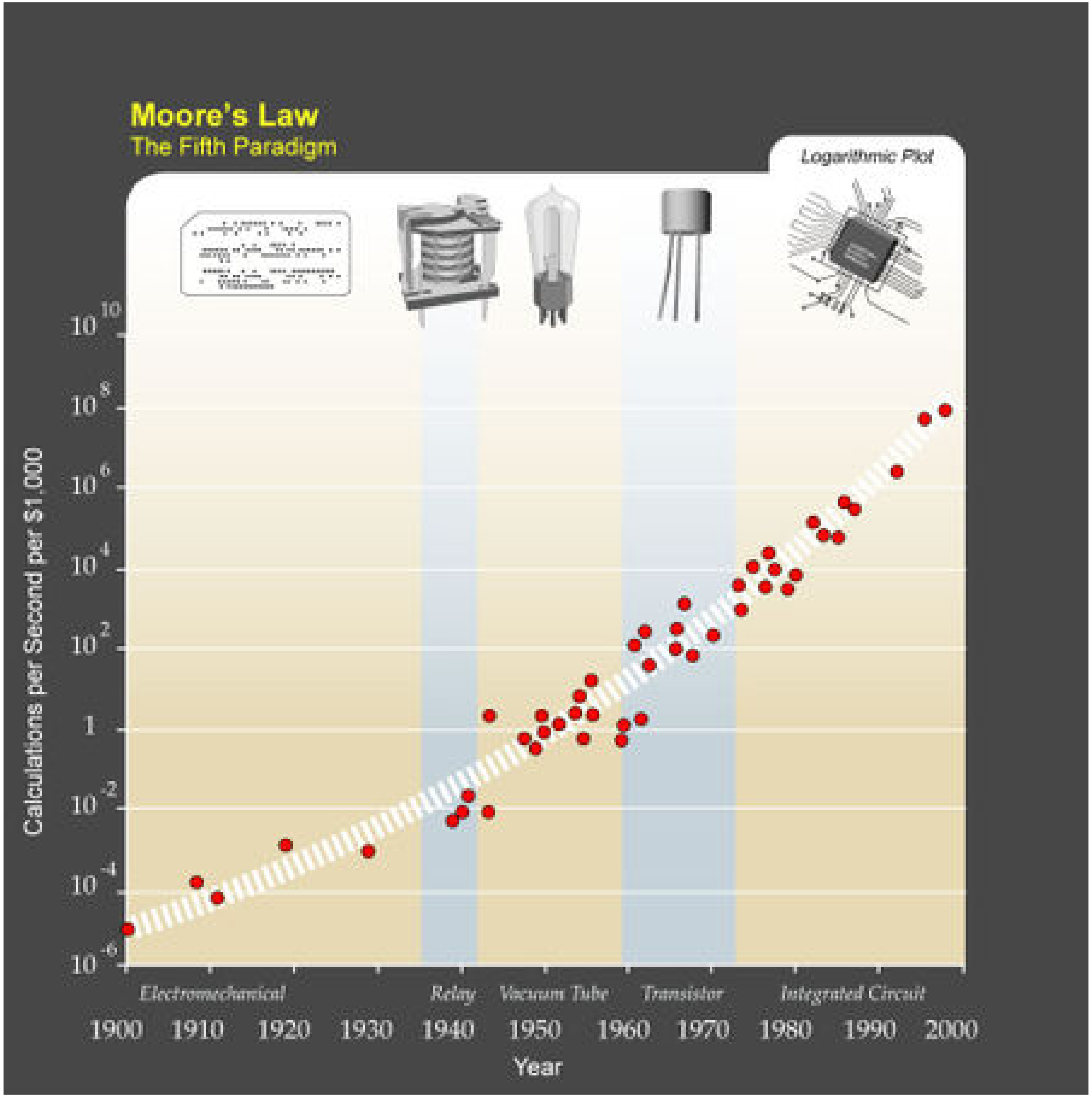
Auguste Kerckhoffs, 1835-1903 (Source: Wikipedia)

1. The system should be, if not theoretically unbreakable, unbreakable in practice.
2. (“Kerckhoffs’ Principle”) The design of a system should not require secrecy and compromise of the system should not inconvenience the correspondents.
3. The key should be memorable without notes and should be easily changeable.
4. The cryptograms should be transmittable by telegraph.
5. The apparatus or documents should be portable and operable by a single person.
6. The system should be easy, neither requiring knowledge of a long list of rules nor involving mental strain.

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.
- 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.




Moore's Law
The Fifth Paradigm
Logarithmic Plot
Calculations per Second per $1,000
10^10
10^8
10^6
10^4
10^2
1
10^-2
10^-4
10^-6
Electromechanical
Relay
Vacuum Tube
Transistor
Integrated Circuit
1900 1910 1920 1930 1940 1950 1960 1970 1980 1990 2000
Year

Triple DES

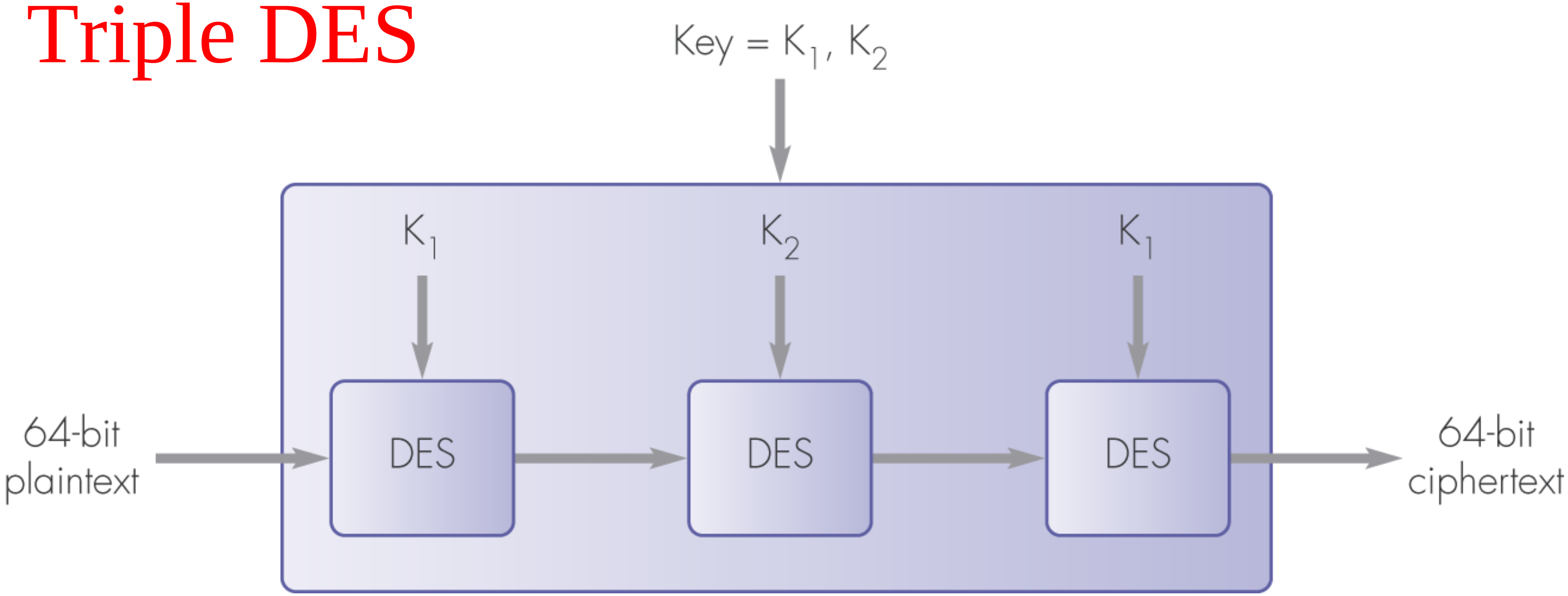


Figure 10.5 (Halsall) Triple DES schematic

- 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

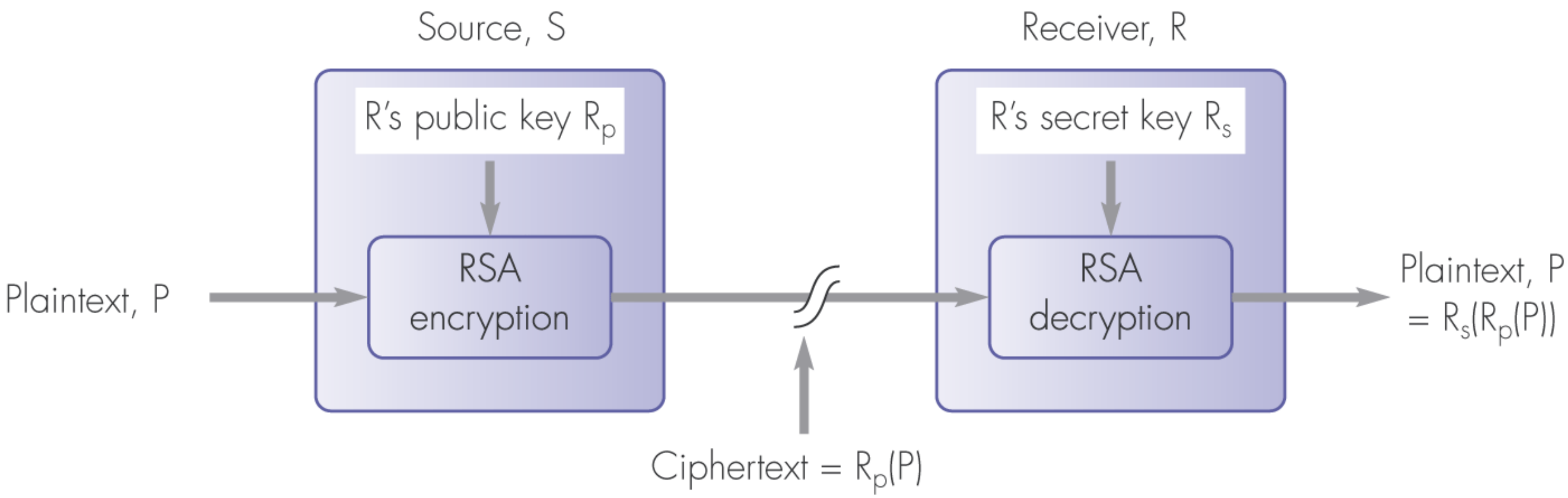


Figure 10.8 (Halsall) RSA schematic

- Two different keys!! Sender: (S_P, S_S); Receiver: (R_P, R_S).
  - Shay uses slightly different notation: (E_k, D_k), $(E_{k'}, D_{k'})$
- If you publish your public key, then anyone who reads this publication can send you an encrypted message.
- Your private key is required to decrypt.

Non-repudiation

- Any public-key cryptographic system, e.g. RSA, can be used for non-repudiable messaging.
- Encrypt a plaintext message P with your secret signing key s: $S_S(P)$.
  - This is a strange use of encryption.
  - Anyone can read your encrypted message, because the public half S_P of your signing keypair (S_S, S_P) is *not* a secret.
- You are the only person who can efficiently compute $S_P(P)$, unless you reveal your secret signing key S_S to someone else.
  - But… if you *don't* give someone else a copy of your keys, what happens if you lose them?!
  - Key management is *very* difficult in practice.

Secret and Non-repudiable Messaging

- You can send a secret non-repudiable message $R_P(S_S(P))$,
  - if you know your recipient's public encryption key R_P
  - and if they know your public signing key S_P.

Can you specify the computation that should be done by the receiver after receiving message M?

1. Sender $\rightarrow$ Receiver: $M = R_P(S_S(P))$
2. Receiver: P = ________________________

Efficient Non-repudiation

- 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.

➢ Efficiency trick: sign only the hash. Send P | S_S(MD5(P)).

Why do we use such long hashes?

- Shay has a nice explanation of why we don't use 64-bit secure hashes for signed messages.
  - An attacker who creates 2^{32} different versions of a message has a good chance of finding two versions with the same hash.
- Here's another birthday-attack scenario:
  - M = "The NZSE will go up on 12 Sept 07", M' = "The NZ sharemarket will go down on Friday 12 September", with H(M) = H(M').
  - M and M' are datestamped and signed by trusted third parties, such as Verisign, prior to 12 September.
  - The attacker puts many such messages on their webpage, apparently "proving" that they have an infallible method of predicting future movements in the NZSE.
- Short hash signatures are vulnerable to repudiation attacks.
  - "Salt" (a random number) can be added to short messages in secure protocols to increase their resilience.
  - Great expertise is required to design a secure cryptographic protocol!
- I *won't* expect you to design a secure cryptographic protocol, or to know the details of any cryptographic algorithm.
  - I *will* expect you to be able to do simple assessments of simple systems.

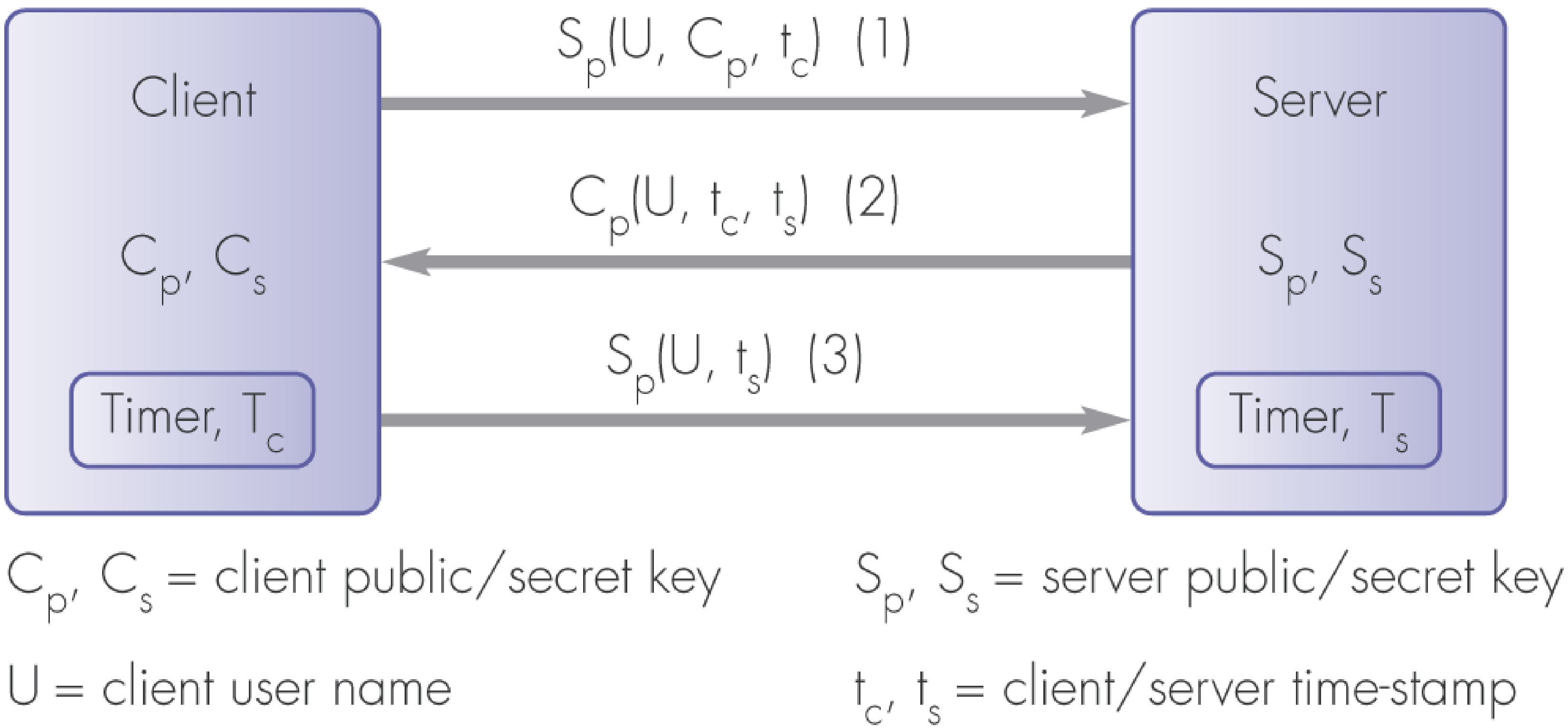


Figure 10.10 (Halsall) User authentication using a public key scheme

- *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)*

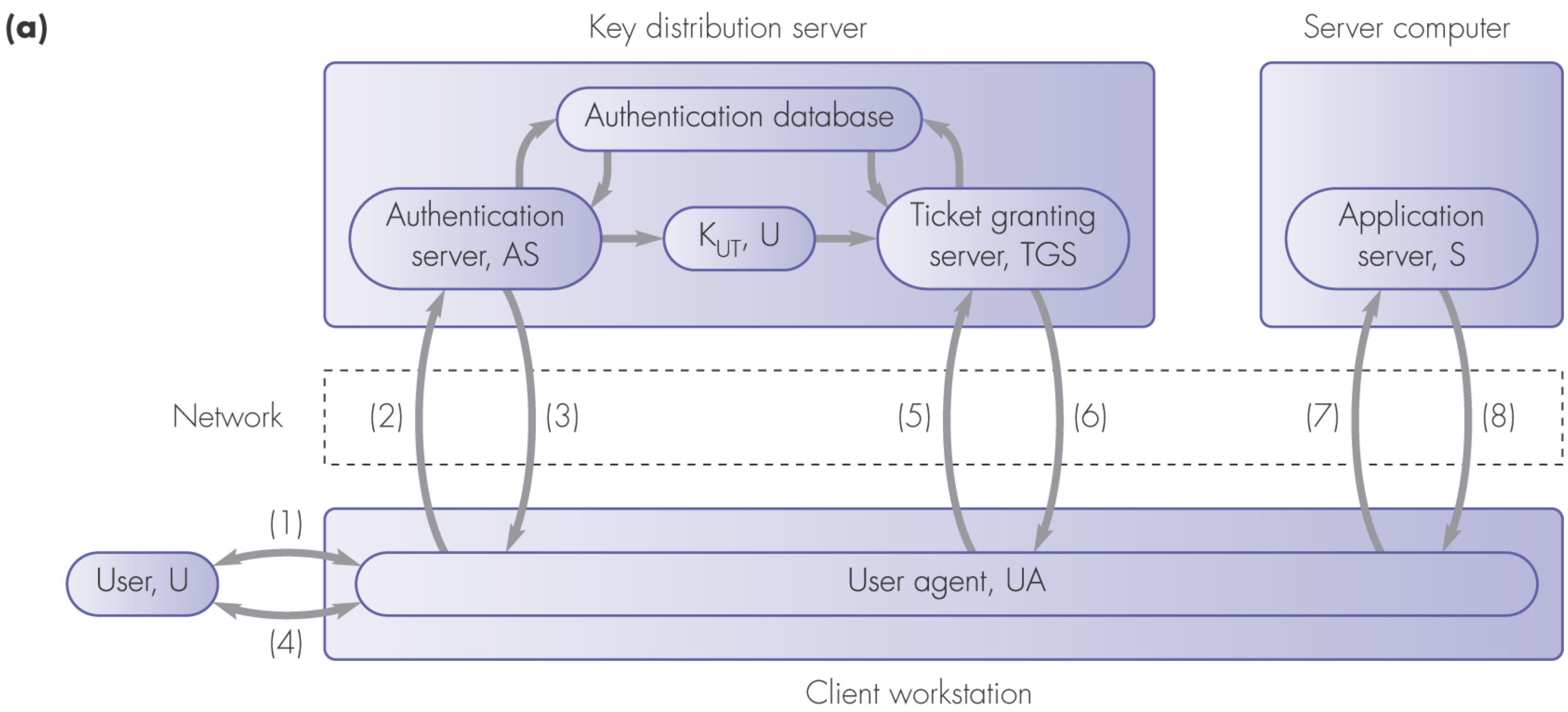
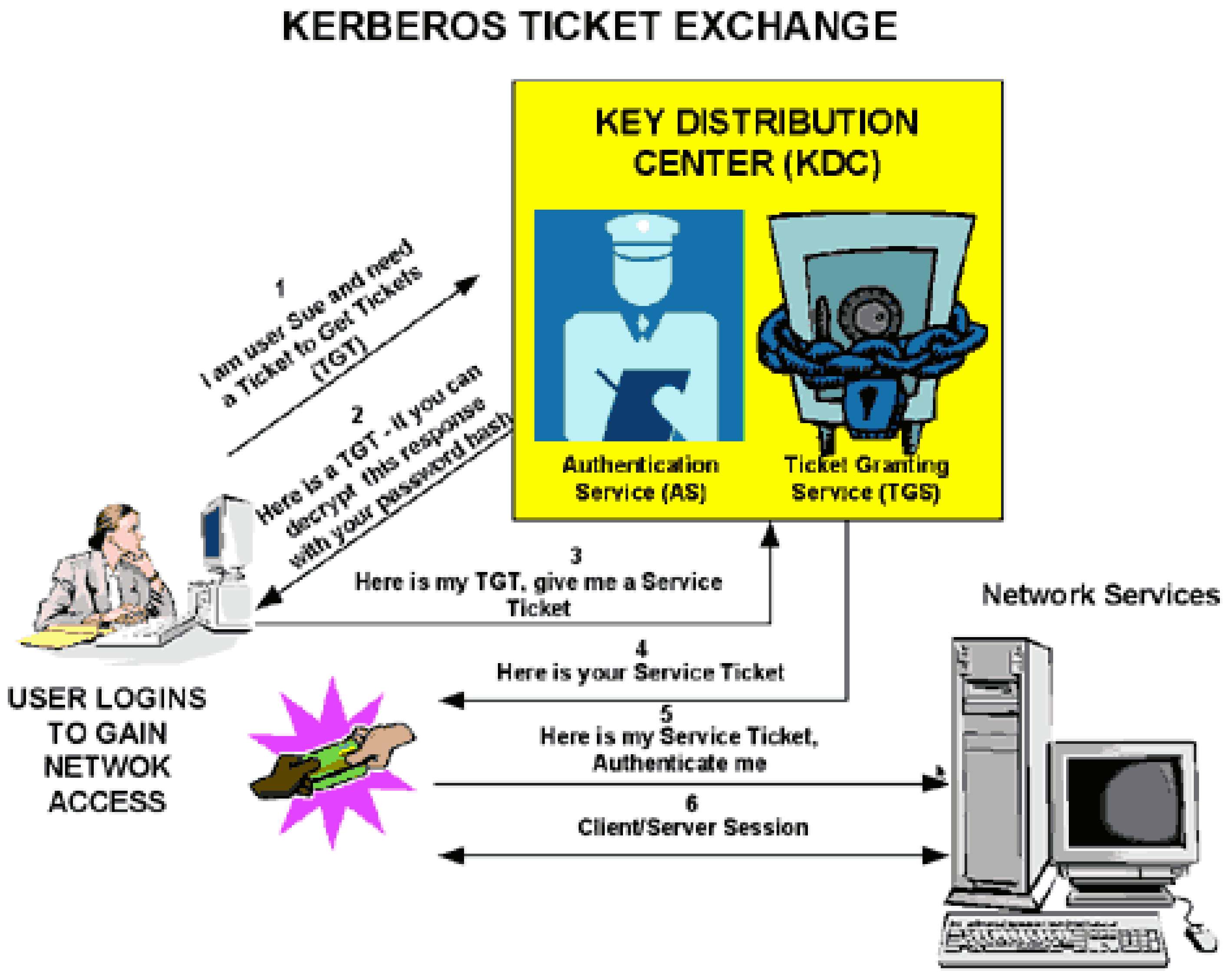


Figure 10.11a (Halsall) User authentication using Kerberos: terminology and message exchange

- *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)	$U \leftrightarrow UA$	User name, U
(2)	$UA \rightarrow AS$	(U, T, n_1)
(3)	$AS \rightarrow UA$	$K_U (K_{UT}, n_1); T_{UT}$
(4)	$U \leftrightarrow UA$	User password, K_U
(5)	$UA \rightarrow TGS$	$K_{UT} (U, t); T_{UT}, S, n_2$
(6)	$TGS \rightarrow UA$	$K_{UT} (K_{US}, n_2); T_{US}$
(7)	$UA \rightarrow S$	$K_{US} (U, t); T_{US}, n_3$
(8)	$S \rightarrow UA$	$K_{US} (n_3)$

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 $\leftrightarrow$ TGS dialog units
K_{US} = A session key to encrypt UA $\rightarrow$ 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

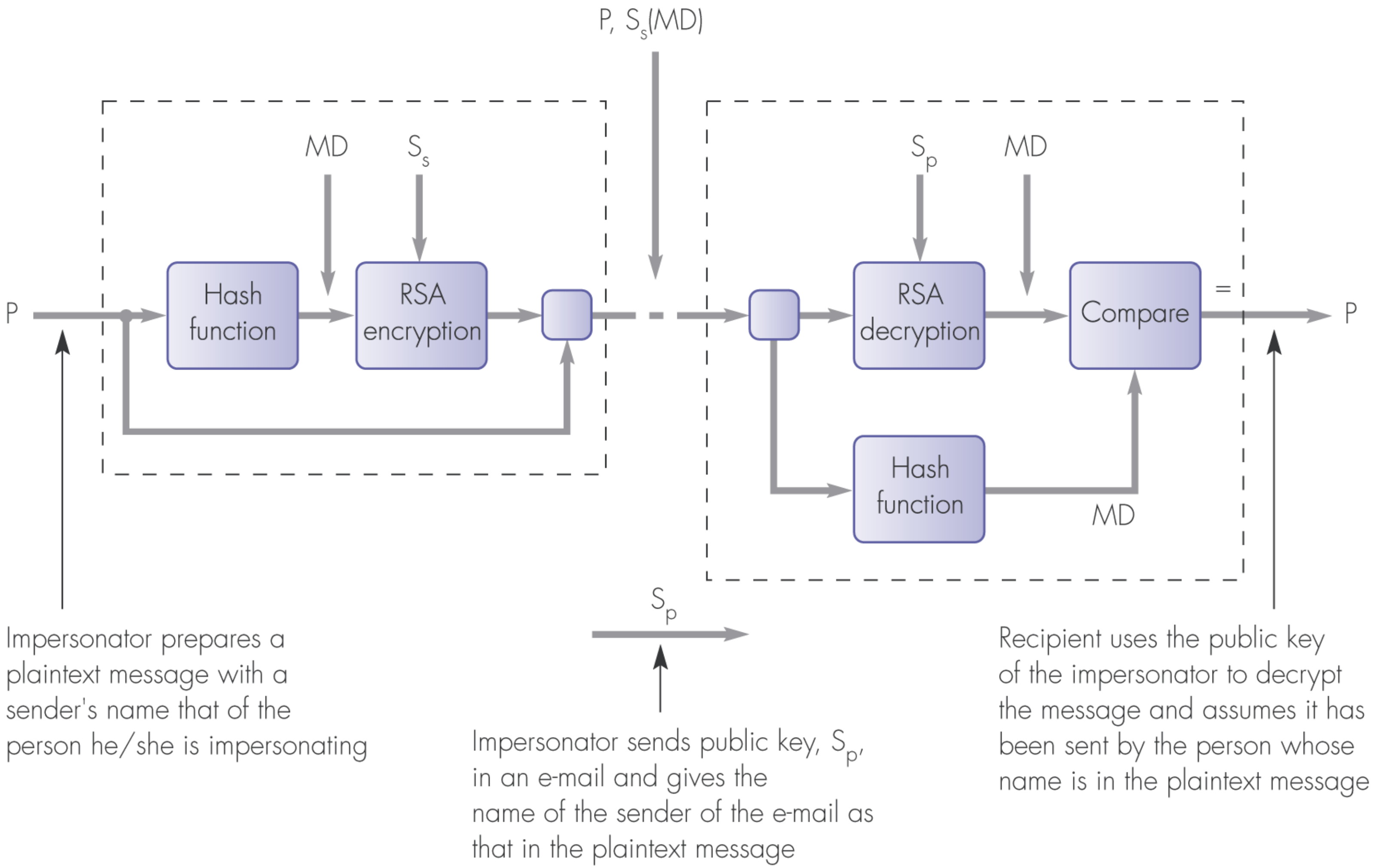


Figure 10.12 (Halsall) A possible threat when using a public key system

- *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?*