

**July 6th, 2012**

giovedì 25 maggio 2017 11:18

**EXERCISE NO. 1** #MARKS: 10  
With respect to the Diffie-Hellman key establishment protocol, (1) present it, and discuss its security with respect to both a (2) passive and an (3) active adversary

**EXERCISE NO. 2** #MARKS: 10  
Let us consider the protocol below aimed at establishing a session key  $K_{AB}$  between Alice and Bob. In the protocol,  $n_A$  and  $n_B$  denote two nonces that are generated by Alice and Bob, respectively;  $K_B$  denotes the public key of Bob; and, finally,  $P_A$  denotes the shared secret password between Alice and Bob.

M1  $A \rightarrow B \quad \{n_A, K_{AB}\}_{K_B}$   
M2  $B \rightarrow A \quad \{n_B, n_A\}_{K_{AB}}$   
M3  $A \rightarrow B \quad \{n_B, P_A\}_{K_{AB}}$

1) Analyse the protocol and verify whether it fulfils the key authentication and the key confirmation requirements. Specify the assumptions under which the requirements are fulfilled.  
2) Let us suppose that a session key  $K_{AB}$  is compromised. (a) Discuss the consequences.  
(b) Improve the protocol in order to limit at the minimum the effects of session key compromization.

**EXERCISE NO. 3** #marks: 10  
Let us consider the Vernam Cipher (One-Time Pad).

## Exercise 2

- ! • M1 is encrypted by means of  $K_B$

My guess

1. Key confirmation is provided by M3 because A encrypts it by means of  $K_{AB}$   
Key authentication is provided by M3 because A inserts  $P_A$  in the message.
2.  $K_{AB}$  compromised:
  - a. ...
  - b. ...

BAN logic

July 17th,  
2012

1. Dopo M1, Bob non può sapere da chi sta arrivando, non si può applicare il primo postulato (Bob doesn't believe, it just sees).  
Non ha nessun belief su  $K_{AB}$ , e nemmeno sulla sua freschezza, perché  $n_A$  non è gestito da Bob.

2. Bob encripta M2  $K_{AB}$  assumendo che solo Alice lo abbia.  
Stesso errore in old SSL.  
Un avversario potrebbe ri-eseguire lo stesso protocollo con  $K_{AB}$  tutte le volte che vuole.
  - a. Perdendo  $K_{AB}$ , si perdono le sessioni passate, a meno che non si utilizzino sessioni ephemeral.  
Perdendo  $K_{AB}$ , si perdono le sessioni future.

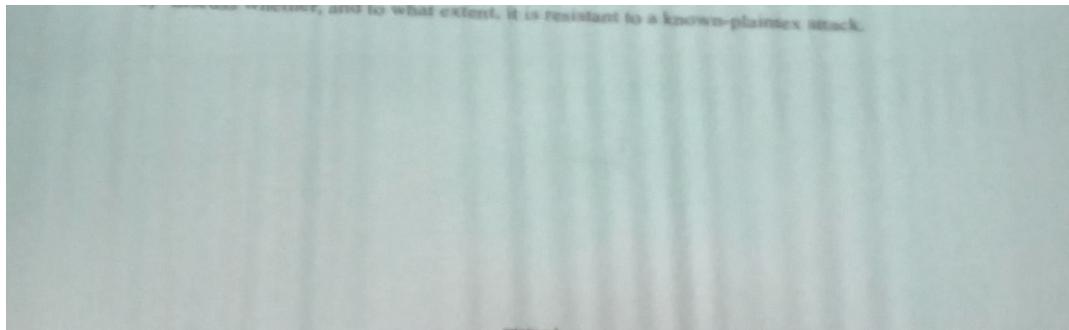
- b. Ad esempio,  $P_A$  si potrebbe inserire subito in M1.  
Per dichiarare la freschezza di M1, Bob potrebbe inviare un nonce in M0.

M0	$B \rightarrow A$	$n_B$
M1	$A \rightarrow B$	$\{n_A, n_B, P_A, K_{AB}\}_{K_B}$
M2	$B \rightarrow A$	$\{n_A, n_B\}_{K_{AB}}$
M3	$A \rightarrow B$	$\{n_B, n_A\}_{K_{AB}}$

Con  $K_{AB}$  compromessa, all'avversario manca comunque la password  $P_A$ .  
Se  $K_{AB}$  viene compromessa, viene compromessa però anche la password  $P_A$ .

**EXERCISE NO. 3** #MARKS: 10  
Let us consider the Vernam Cipher (One-Time Pad).

- 1) discuss whether, and to what extent, it is resistant to a ciphertext-only attack.
- 2) discuss whether, and to what extent, it is resistant to a known-plaintext attack.



## September 14th, 2013

giovedì 25 maggio 2017 10:45

### Exercise 2

**EXERCISE NO. 2**

The figure shows an identification protocol that allows a mobile station (MS) to identify itself w.r.t. an access point (AP) where  $c$  is a 128-bit random challenge,  $r$  is the corresponding response, and  $v$  is 24-bit random initialization vector.

**#MARKS: 10**

**Protocol:**

- M1 MS → AP: REQ
- M2 AP → MS:  $c$
- M3 MS → AP:  $v, r$
- M4 AP → MS: YES|NO

Upon receiving message M2 carrying a challenge  $c$  from AP, MS generates an initialization vector  $v$  at random, computes the response  $r$  by encrypting  $c, r = \text{SPRG}(k||v)|_{128} \oplus c$ , where  $\text{SPRG}(k||v)|_{128}$  is a 128-bit sequence generated by a secure pseudo-random generator SPRG. The generator is seeded by  $k||v$ , where  $k$  is a long-term cryptographic key secretly shared by AP and MS.

Upon receiving the response  $r$  from MS in message M3, AP computes  $r' = \text{SPRG}(k||v)|_{128} \oplus c$ , and returns  $r' == r$  to the user.

A. Does this protocol guarantee identification? Can a passive adversary impersonate a mobile station?

Let us suppose now that AP sends MS the initialization vector  $v$  together with the challenge  $c$  in message M2 which becomes  $\langle c, v \rangle$  ( $v$  in M3 is not necessary anymore)

B. Define a dictionary attack against this variant of the protocol and evaluate the size in bytes of the dictionary.<sup>1</sup>



September  
14th, 2013

- A. This protocol guarantees identification because the challenge is encrypted by XORing it with a random number seeded with  $k$ , that is a long-term shared secret key between the AP and the MS. Only the MS knows  $k$ , so only that MS can generate such a random number.

If an adversary wants to impersonate the MS, it has to guess the  $k$  shared key in order to reply to the challenge correctly.

The adversary cannot reply M3 because the AP changes its challenge  $c$  at every protocol iteration (the probability of re-using the same  $c$  key value is very low).  $v$  though is relatively slow.

The only weakness is on the shared secret key  $k$ .

$z = \text{SPRG}(k | v)$  is called **keystream**.

$$r = z \oplus c$$

- a. Eavesdropping

- i. All of them are public quantities, not related to each other.
- ii.  $z$  can be calculated by doing  $z = r \oplus c$

- b. The adversary can start a new protocol instance

This can be done because  $v$  is always the same: it doesn't need to discover  $k$ , it just uses  $z$ .

- i. It receives  $c'$  as a challenge
- ii. It can send  $v$  and  $r' = z \oplus c'$ , with a previously spoofed  $z$ .

This protocol was used by WEP.

- B. The adversary can still calculate  $z$ , but the AP could use another initialization vector  $v$ , so the previous attack could not be performed anymore.

The adversary could build a dictionary of  $(v, z)$  pairs.

If the AP re-uses a  $v$ , the adversary would have a corresponding  $z$ .

$v$  are on 24-bits, so they're not so much: the dictionary would be  $2^{24} \text{ entries} \cdot \frac{128 \text{ bits for each keystream}}{8 \frac{\text{bits}}{\text{Byte}}} = 2^{24} \cdot 2^4 = 2^{28} = 256 \text{ MB}$

If  $v$  is generated by means of a counter, everytime the AP is rebooted, the  $v$  is generated from 0, and so on.

Otherwise,  $2^{24}$  IVs is not a large number of them.

## SECURITY IN NETWORKED COMPUTING SYSTEMS

January 17 , 2017

Name \_\_\_\_\_

Serial no. \_\_\_\_\_

**EXERCISE NO. 1**

With reference to the RSA scheme, where  $d$  and  $e$  are the private and public exponents respectively,

1. Illustrate the square-and-multiply algorithm for modular exponentiation and discuss its performance;
2. In the light of point 1, argue why the usual choice for public exponent  $e$  is either 3 or  $2^{16}+1$ ;
3. Argue whether the same optimization can be done on private exponent  $d$ .

## February 17th, 2017

giovedì 25 maggio 2017 11:59

Exercise 2 solution



February  
17th, 2017 -



February  
17th, 2017 -

1. It does satisfy the password's confidentiality, because the first message is encrypted by means of ...
2. Kcs and M1, M2, M3 compromised
  - a. L'avversario può decifrare M2, scoprendo  $n_c$ .  
In M1, non ci sono più aspetti random, e può fare un brute-force attack sulla password  $\pi$ .  
I brute-force attacks sulle password sono sempre da prendere in considerazione, perché sono scelte dagli utenti e possono essere deboli.
  - b. M1 è attribuibile al client perché c'è un segreto  $\pi$  (terzo punto del primo postulato).  
Il server non può stabilire se il messaggio è fresco: questo soffre del replay attack.  
Con la BAN al primo messaggio, si vede subito che non c'è una prova di freschezza, e perciò metteva subito in risalto il problema del replay attack.
  - c. "clocks are not synchronized": questo non è modificabile.  
M2 è per la key-confirmation.  
Problemi da risolvere:
    - Freschezza M1

M0	$S \rightarrow C$	$n_s$
M1	$C \rightarrow S$	$E_{PK_S}(C, S, n_C, n_s, \pi, K_{CS})$
M2	$S \rightarrow C$	$E_{K_{CS}}(S, C, n_C)$

- Poca randomicità di M1

Permetteva di effettuare un brute-force attack alla password.  
Con la modifica sulla freschezza, questo è sempre possibile.  
Questo si può evitare, invece di inserire  $n_c$  in M2, inserendo  $H(n_c)$ .  
 $H()$  è pre-image resistance.

EXERCISE NO. 2 #MARKS: 12

A client  $C$  and a server  $S$  share a password  $\Pi$ . Furthermore, client  $C$  knows the public key  $PK_S$  of server  $S$ . Client and server are equipped with computationally secure hash functions, symmetric and asymmetric ciphers. Finally, client and server clocks are not synchronized. Under these assumptions, client and server attempt to establish a symmetric session key  $K_{CS}$  by means of the following key establishment protocol:

$$\begin{array}{ll} M1 & C \rightarrow S: E_{PK_S}(C, S, n_C, \Pi, K_{CS}) \\ M2 & S \rightarrow C: E_{K_{CS}}(S, C, n_C, n_s) \\ M3 & C \rightarrow S: n_s \end{array}$$

where the kind of encryption scheme, symmetric or asymmetric, is clear from the context.

1. Argue whether the protocol satisfies the confidentiality of the password  $\Pi$  in the case of *ciphertext-only* attack.
2. Assume now the adversary gets hold of a session key  $K_{CS}$  and records the protocol instance  $(\overline{M1}, \overline{M2}, \overline{M3})$  that led to that key establishment.
  - a. Argue whether the confidentiality of password  $\Pi$  is still guaranteed under this assumption (hint: considers an off-line guessing attack).
  - b. Argue whether the protocol suffers from a replay attack.
  - c. If the protocol suffers from any of these attacks, modify it in order to prevent them.

**EXERCISE NO. 1 (LMCE, LMECS)**

1. Introduce the Diffie-Hellman key exchange scheme.
2. Argue about its security w.r.t. a passive adversary.
3. Argue about its vulnerability to the man-in-the-middle attack, and propose a solution.

REMARKS 12

**EXERCISE NO. 2 (LMCE)**

REMARKS 13

Let us consider the modified version of the Diffie-Hellman protocol reported below and aimed at establishing a session key  $K_{AB} = g^{x+y} \text{ mod } p$ , between user Alice and server Bob, with  $p$  a secret password shared between Alice and Bob.

$$\begin{aligned} M1: A \rightarrow B: & A, (g^x \text{ mod } p), \\ M2: B \rightarrow A: & B, (g^y \text{ mod } p), \end{aligned}$$

- 1) Which of these are drawbacks of using the protocol (argue the answer)?
  - a) It is vulnerable to offline password-dictionary attacks.
  - b) It requires server Bob to store passwords in the clear-text.
  - c) It is vulnerable to the man-in-the-middle attack.
- (2) Does the protocol guarantees identification, i.e.,  $A$  knows that  $B$  is present and/or vice versa (argue the answer)?

- A. No;
- B. Yes, A w.r.t. B;
- C. Yes, B w.r.t. A;
- D. Yes, both.

(3) Extend the protocol in order to achieve mutual authentication.

# February 20th, 2017

martedì 23 maggio 2017 16:34

## Exercise 1

## Exercise 2

?

1. ?
2. Yes, A w.r.t. B
3. Yes, B w.r.t. A
4. Yes, both.

(3) Extend the protocol in order to achieve mutual authentication.

## Exercise 3



February  
20th, 2017

Let  $K_A$  be the public key of Alice,  $SP(x)$  be the digital signature of principal P on item x, CA be a Certification Authority (trusted by all principals of the system), and finally  $H$  a secure hash function. Which of the following **certificates** are useful to establish a secure channel with Alice (do not consider the validity interval)? Argue why.

1. "Alice" ||  $S_{CA}(H("Alice" || K_A))$ 
  - Non c'è modo di prelevare la chiave pubblica di A dal certificato.
2. "Alice" ||  $K_A$  ||  $S_A("Alice" || K_A)$ 
  - Il certificato è autofirmato, non è firmato da CA.
3. "Alice" ||  $K_A$  ||  $S_{CA}("Alice" || H(K_A))$ 
  - Scelta meno efficiente, ma dal punto di vista della certificazione può funzionare.
4. "Alice" ||  $K_A$  ||  $S_{CA}(H("Alice" || K_A))$ 
  - Scelta canonica.
5. "Alice" ||  $K_A$  ||  $S_{CA}(K_A)$ 
  - Non c'è un legame nome-chiave
6. "Alice" ||  $K_A$  ||  $S_B("Alice" || H(K_A) || "issuer:Bob")$  ||  $S_{CA}("Bob" || K_B)$ 
  - Bob rilascia il certificato, può non avere validità.
7. "Alice" ||  $K_A$  ||  $S_B("Alice" || H(K_A) || "issuer:Bob")$  ||  $S_{CA}("Bob, CA=Yes" || K_B)$ 
  - È ancora Bob a rilasciare il certificato.
  - Nell'ultima parte, viene detto che la CA delega Bob a rilasciare certificati.

**18 February 2014**

NAME \_\_\_\_\_ SERIAL NO. \_\_\_\_\_

**EXERCISE NO. 1**

**#MARKS: 12**

Define a secure hash function and argue the relevance of its properties with respect to digital signature.

**EXERCISE NO. 2**

**#MARKS: 6**

In an access control system (ACS), Alice brings a personal device that is equipped with a symmetric cypher, a collision-resistant hash function, a random number generator, and a short-range wireless communication device. Alice and the ACS share a password  $\Pi_A$ .

Design a challenge-response protocol that allows Alice to prove ACS her presence. The key  $K_A$  shared by Alice and ACS for the challenge response protocol is derived from the password. The protocol must i) guarantee the authentication of Alice; ii) be resistant to replay-attacks; and iii) prevent offline password-guessing attack.

**EXERCISE NO. 3**

**#marks: 12**

Let  $(S, D)$  be a secure digital signature scheme with appendix. Let  $S$  and  $D$  be the signature and verification algorithm, respectively. Furthermore, let  $K_P$  be principal  $P$ 's public key, and  $CA$  a Certification Authority that is trusted by all principals of the system. Finally let  $H$  be a secure hash function. Which of the following *certificates* are useful to establish a secure channel with Alice? Argue why.<sup>1</sup>

- (A) "Alice" ||  $K_A$  ||  $S_{CA}(\text{Alice})$
- (B) "Alice" ||  $K_A$  ||  $S_{CA}(K_A)$
- (C) "Alice" ||  $K_A$  ||  $S_A(H(\text{"Alice"}) || K_A)$
- (D) "Alice" ||  $K_A$  ||  $S_{CA}(\text{"Alice"} || H(K_A))$
- (E) "Alice" ||  $K_A$  ||  $S_{CA}(H(\text{"Alice"}) || K_A)$
- (F) "Alice" ||  $K_A$  ||  $S_{Bob}(\text{"Alice"} || K_A)$  || "Bob" ||  $K_B$  ||  $S_{CA}(\text{"Bob"}) || K_B$
- (G) "Alice" ||  $K_A$  ||  $S_{Bob}(\text{"Alice"} || K_A)$  || "Bob, CA: yes" ||  $K_B$  ||  $S_{CA}(\text{"Bob, CA: yes"}) || K_B$ .

---

<sup>1</sup> Neglect any issue related to time.

29 January 2014

**SOLUTION**

**EXERCISE #1**

*See theory.*

**EXERCISE #2.**

$M1 \quad A \rightarrow S : \quad A$

$M2 \quad S \rightarrow A : \quad n_s$

$M3 \quad A \rightarrow S : \quad E_{K_A}(A, n_s, s_A)$

where  $K_A = h(\Pi_A)|_k$ . Notice that  $s_A$  is a random salting quantity aimed at avoiding an offline password-guessing attack.

**EXERCISE #3.**

- A. Certificate A does not link KA to Alice
- B. Certificate A does not link KA to Alice
- C. Certificate B is self-signed and Alice is not a trusted authority
- D. Certificate C is fine.
- E. Certificate C is fine.
- F. Bob, who is not a trusted authority, signed certificate D.
- G. Certificate E is fine: CA delegates B to sign certificates.

**SECURITY IN NETWORKED COMPUTING SYSTEMS**  
*Computer Engineering*

**18 September 2014**

NAME \_\_\_\_\_ SERIAL NO. \_\_\_\_\_

♥ : MCE, SSI, SnR; ♠ : DSS

**EXERCISE NO. 1 (♥, ♠)**

**#MARKS: 10**

With reference to hash functions,

1. Provide the definition of the *pre-image resistance*, *second-preimage resistance* and *collision resistance* properties;
2. Argue about the relevance of these properties w.r.t. to a digital signature scheme;
3. Argue about the security of hash functions with respect to black box attacks.

**EXERCISE NO. 2 (♥, ♠)**

**#MARKS: 12**

Let us consider the following *secret sharing scheme* that allows us to share a secret  $x$  between two non-colluding people so that each person alone is not able to reconstruct the secret.

1. Let  $x$  be a secret bit-string  $t$ -bit long;
2. Generate a  $t$ -bit truly random key  $k$ ;
3. Compute a *share*  $s_i$  s.t.,  $s_i = x_i \oplus k_i$ ,  $0 \leq i \leq t-1$ ;
4. Give the key  $k$  to Alice and the share  $s$  to Bob.

The candidate answers the following questions.

- Question A. Under the assumption that Alice and Bob do not collude, is the scheme perfectly secure? In other words, can Alice or Bob alone derive any information about the secret  $x$ ?
- Question B. What about if Alice and Bob collude?
- Question C. Generalize the scheme in order to share a secret among  $n$  users.
- How many key bits do we need to share a  $t$ -bit secret among  $n$  users?

**EXERCISE NO. 3 (♥)**

**#marks: 8**

How the problem of delegation is solved in Kerberos?

Describe the protocols for proxiable and forwardable tickets.

**EXERCISE NO.4 (♠)**

**#MARKS: 8**

Let  $(S, D)$  be a secure digital signature scheme with appendix. Let  $S$  and  $D$  be the signature and verification algorithm, respectively. Furthermore, let  $K_P$  be principal  $P$ 's public key, and  $CA$  a Certification Authority that is trusted by all principals of the system. Finally let  $H$  be a secure hash function. Which of the following *certificates* are useful to establish a secure channel with Alice? Argue why.<sup>1</sup>

- (A) “Alice” ||  $K_A$  ||  $S_{CA}(Alice)$
- (B) “Alice” ||  $K_A$  ||  $S_{CA}(K_A)$
- (C) “Alice” ||  $K_A$  ||  $S_A(H(“Alice” ||  $K_A$ ))$
- (D) “Alice” ||  $K_A$  ||  $S_{CA}(“Alice” || H( $K_A$ ))$
- (E) “Alice” ||  $K_A$  ||  $S_{CA}(H(“Alice” ||  $K_A$ ))$
- (F) “Alice” ||  $K_A$  ||  $S_{Bob}(“Alice” ||  $K_A$ )$  || “Bob” ||  $K_B$  ||  $S_{CA}(“Bob” ||  $K_B$ )$
- (G) “Alice” ||  $K_A$  ||  $S_{Bob}(“Alice” ||  $K_A$ )$  || “Bob, CA: yes” ||  $K_B$  ||  $S_{CA}(“Bob, CA: yes” ||  $K_B$ )$ .

<sup>1</sup> Neglect any issue related to time.

**18 September 2014**

NAME \_\_\_\_\_ SERIAL NO. \_\_\_\_\_

## **SOLUTION**

### **Exercise n.1**

See theory

### **Exercise n.2**

**Question A.** Under the assumption that users do not collude, the scheme is perfectly secure because the key is perfectly random and the share is the result of one-time pad.

**Question B.** If Alice and Bob collude, they can compute the secret.

**Question C.** Define  $(n - 1)$  keys,  $k_1, k_2, \dots, k_{n-1}$ , and compute a share  $s = x \oplus k_1 \oplus k_2 \oplus \dots \oplus k_{n-1}$ . Then distribute share and keys to  $n$  different, non-colluding users. Notice that we need the presence (or collusion) of  $n$  users in order to reconstruct the secret.

A possible different approach is to 1) generate a single key  $k$ ; 2) compute a share  $s = x \oplus k$ , and finally 3) split  $k$  into  $(n - 1)$  pieces,  $P = \{p_1, p_2, \dots, p_n\}$ . Each piece requires  $v = t / (n - 1)$  bits. This solution is less secure than the previous one because  $(n - 2)$  users may be sufficient to reconstruct the secret. Let us suppose that  $(n - 2)$  users are present (or colluding) and that one user, say user  $u_i$  is missing. Then, reconstructing the secret through a brute force attack is a matter of attempting all possible sequences of  $v$  bits in order to guess the missing key piece. Its complexity is  $O(2^v) = O\left(2^{\frac{t}{n-1}}\right) = O\left(\sqrt[n-1]{2^t}\right)$ . This computation complexity could be affordable for an adversary also for a small value of  $n$ .

**Question D.** The number of key bits is  $(n - 1) \times t$ .

### **Exercise n.3**

See theory

### **Exercise n.4**

- A. Certificate A does not link KA to Alice
- B. Certificate A does not link KA to Alice
- C. Certificate B is self-signed and Alice is not a trusted authority
- D. Certificate C is fine.
- E. Certificate C is fine.
- F. Bob, who is not a trusted authority, signed certificate D.
- G. Certificate E is fine: CA delegates B to sign certificates.

September 18th, 2014

giovedì 25 maggio 2017 11:40



snscs-140918

SICUREZZA NELLE RETI  
Laurea Specialistica in Ingegneria InformaticaSICUREZZA DEI SISTEMI SOFTWARE (6/9 CFU)  
Laurea Magistrale in Ingegneria InformaticaSECURITY IN NETWORKED COMPUTING SYSTEMS  
Computer Engineering

18 September 2014

NAME \_\_\_\_\_ SERIAL NO. \_\_\_\_\_

▼ : MCE, SSI, SnR; ▲ : DSS

## EXERCISE NO. 1 (▼,▲)

#MARKS: 10

With reference to hash functions,

1. Provide the definition of the *pre-image resistance*, *second-preimage resistance* and *collision resistance* properties;
2. Argue about the relevance of these properties w.r.t. to a digital signature scheme;
3. Argue about the security of hash functions with respect to black box attacks.

## EXERCISE NO. 2 (▼,▲)

#MARKS: 12

Exercise 2

September  
18th, 2014Let us consider the following *secret sharing scheme* that allows us to share a secret  $x$  between two non-colluding people so that each person alone is not able to reconstruct the secret.

1. Let  $x$  be a secret bit-string  $t$ -bit long;
2. Generate a  $t$ -bit truly random key  $k$ ;
3. Compute a *share*  $s_i$ , s.t.,  $s_i = x_i \oplus k_i$ ,  $0 \leq i \leq t-1$ ;
4. Give the key  $k$  to Alice and the share  $s$  to Bob.

The candidate answers the following questions.

- Question A. Under the assumption that Alice and Bob do not collude, is the scheme perfectly secure? In other words, can Alice or Bob alone derive any information about the secret  $x$ ?
- Question B. What about if Alice and Bob collude?
- Question C. Generalize the scheme in order to share a secret among  $n$  users.
- How many key bits do we need to share a  $t$ -bit secret among  $n$  users?

## EXERCISE NO. 3 (▼)

#marks: 8

How the problem of delegation is solved in Kerberos?

Describe the protocols for proxiable and forwardable tickets.

## EXERCISE NO.4 (▲)

#MARKS: 8

Let  $(S, D)$  be a secure digital signature scheme with appendix. Let  $S$  and  $D$  be the signature and verification algorithm, respectively. Furthermore, let  $K_P$  be principal  $P$ 's public key, and  $CA$  a Certification Authority that is trusted by all principals of the system. Finally let  $H$  be a secure hash function. Which of the following certificates are useful to establish a secure channel with Alice? Argue why.<sup>1</sup>

- (A) "Alice" ||  $K_A$  ||  $S_{CA}(Alice)$   
 (B) "Alice" ||  $K_A$  ||  $S_{CA}(K_A)$   
 (C) "Alice" ||  $K_A$  ||  $S_A(H("Alice" ||  $K_A$ ))$   
 (D) "Alice" ||  $K_A$  ||  $S_{CA}("Alice" || H(K_A))$   
 (E) "Alice" ||  $K_A$  ||  $S_{CA}(H("Alice" ||  $K_A$ ))$   
 (F) "Alice" ||  $K_A$  ||  $S_{Bob}("Alice" ||  $K_A$ )$  || "Bob" ||  $K_B$  ||  $S_{CA}("Bob" || K_B)$   
 (G) "Alice" ||  $K_A$  ||  $S_{Bob}("Alice" ||  $K_A$ )$  || "Bob, CA: yes" ||  $K_B$  ||  $S_{CA}("Bob, CA: yes" || K_B)$ .

Exercise 2

September  
18th, 2014

- A. No.  
 Lo schema è perfettamente sicuro secondo Shannon perché è a tutti gli effetti un One-Time Pad.  
 B. They can derive  $x$  XORing  $s$  and  $k$ .  
 C. Per condividere un segreto  $x$  tra  $n$  utenti, lo si può cifrando con  $n - 1$  chiavi  $s = x \oplus k_1 \oplus k_2 \oplus \dots \oplus k_{n-1}$   
 a. Alternativa: dividere  $s$  e  $k$  in  $n/2$  parti, dando da  $s_{1:n/2}$  a  $s_{n/2:n}$  alla prima metà di utenti, e così via per  $k$ .



September

18th, 2014

Problema: ogni utente ha qualcosa lungo  $\frac{t}{n/2} = \frac{2t}{n}$ .Con  $n$  grande, questa quantità diventa piccola.  
 Se  $n-1$  colludono, si può effettuare un brute-force attack per l'ultimo che non ha colluso, in quanto il suo  $s_{i:n}$  ( $i$  o  $k_{i:n}$ ) è su pochi bit.  
 Questo non è più un caso di One-Time-Pad.D.  $(n-1) \cdot t$ <sup>1</sup> Neglect any issue related to time.

**SECURITY IN NETWORKED COMPUTING SYSTEMS**

*Computer Engineering*

**15 June 2015**

**EXERCISE NO. 1**

**#MARKS: 10**

With reference to the Diffie-Hellmann key establishment scheme,

1. Describe the scheme;
2. Argue its security with respect to a passive adversary;
3. Argue its security with respect to an active adversary.

**EXERCISE NO. 2**

**#MARKS: 10**

Let us consider an implementation of One-Time Pad (OTP) on  $n$  bit that makes it a perfect cipher.

1. Let  $k_0 = \overbrace{000\dots000}^n$  be a key and  $m$  an  $n$ -bit message. Compute the cipher-text  $c = m \oplus k_0$ .
2. Is there any advantage, or disadvantage, in removing key  $k_0$  from the set of possible keys?
3. Let us suppose that  $c = \text{"The password of my bank account is my wife's birthday"}$ . Which are the most probable plaintext messages (determine at least two)? Which are the corresponding keys?

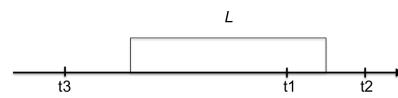
**EXERCISE NO. 3**

**#marks: 10**

Let us consider the “simplified” certificate  $Cert_A = A, pubK_A, L, \sigma$ , with  $\sigma = S_{CA}(t)$  and  $t = H(A, ||pubK_A||L)$ , where  $A$  is the user identifier,  $pubK_A$  is the user’s public key,  $L$  is the validity period,  $H$  is a collision-resistant hash function and  $S$  is a secure digital signature scheme.

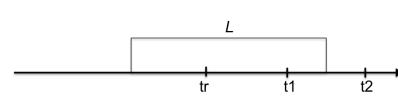
With reference to the figure on the right, give a motivated answer to the following questions:

1. Is  $Cert_A$  valid at time  $t = t1$ ?
2. Is  $Cert_A$  valid at time  $t = t2$ ?
3. Is  $Cert_A$  valid at time  $t = t3$ ?



With reference to the figure on the right, assume that  $Cert_A$  has been revoked at time  $t = tr$ , give a motivated answer to the following questions:

4. Is  $Cert_A$  certainly present in CRL at time  $t = t1$ ?
5. Is  $Cert_A$  certainly present in CRL at time  $t = t2$ ?



**SECURITY IN NETWORKED COMPUTING SYSTEMS**

*Computer Engineering*

**15 June 2015**

**SOLUTION**

**Exercise n.1**

*See theory.*

**Exercise n.2**

**Question 1.**

$c = m$

**Question 2.**

If you remove  $k_0$ , than the number of keys become  $2^n - 1$ . It follows that the number of keys becomes smaller than the number of messages and therefore the resulting cipher is not perfect anymore.

**Question 3.**

Four possible messages are:

The password of my bank account is my wife's birthday  
The password of my bank account is my aunt's birthday  
The password of my bank account is b4nk-P4ssw0rd12345  
Love of my life you left me. You have broken my heart

The respective keys are obtained by computing  $k_i = c \oplus m_i$ .

**Exercise n. 3**

**1:** valid

**2, 3:** invalid as outside the validity interval

**4:** the certificate is certainly in CRL

**5:** the certificate may not be in CRL because, it is not valid anymore, it might have been removed to shorten the CRL itself.

**SICUREZZA NELLE RETI SICUREZZA DEI SISTEMI SOFTWARE (6/9 CFU)**

*Laurea Specialistica in Ingegneria Informatica*

*Laurea Magistrale in Ingegneria Informatica*

**SECURITY IN NETWORKED COMPUTING SYSTEMS**

*Computer Engineering*

**15 june 2015**

**SECURITY IN NETWORKED COMPUTING SYSTEMS***Computer Engineering***15 July 2015****EXERCISE NO. 1****#MARKS: 10**

In RSA, encryption/decryption performance depends on the key's binary representation.

1. Argue the above sentence.
2. Argue why the public exponent  $e = 3$  or  $e = 2^{16}+1$  are a good choice.
3. Argue whether a private exponent  $d$  with only two 1's in its binary representation is a good choice or not.
4. Is a public exponent with only one 1 in its binary representation a viable solution?

**EXERCISE NO. 2****#MARKS: 12**

Alice and Bob share a password (or PIN)  $P$ . For identification, they run the following challenge-response protocol:

$$\begin{aligned} M1 \ A \xrightarrow{\quad} & B: \text{ CHL} \\ M2 \ B \xrightarrow{\quad} & A: \text{ RSP} \end{aligned}$$

Indicate which one of the following implementations of CHL and RSP is secure w.r.t. to an off-line password-guessing attack.

1.  $\text{CHL} = r_a$  and  $\text{RSP} = \{r_a\}_K$ ;
2.  $\text{CHL} = r_a$  and  $\text{RSP} = H_K(r_a)$ ;
3.  $\text{CHL} = r_a$  and  $\text{RSP} = \{r_a, r_b\}_K$ ;
4.  $\text{CHL} = r_a$  and  $\text{RSP} = H_K(r_a \| r_b)$ ;
5.  $\text{CHL} = r_a$  and  $\text{RSP} = \{H_K(r_a)\}_{\Pi_A}$ ;
6.  $\text{CHL} = r_a$  and  $\text{RSP} = \{H_K(r_a), r_b\}_{\Pi_A}$ .

where: **i)**  $\{\cdot\}_\kappa$  denotes encryption by means of key  $\kappa$  (whether symmetric or asymmetric depends on the context), **ii)**  $H_k(\cdot)$  denotes a secure MAC; **iii)**  $r_x$  is a random number generated by  $X$ ; **iv)**  $K$  is a symmetric key, with  $K = f(P)$  and  $f(\cdot)$  a deterministic function; and, finally, **v)**  $\Pi_A$  is A's public key, known to B.

**EXERCISE NO. 3****#marks: 8**

Indicate which of the following certificates is correct.

1.  $A, \Pi_A, L, S_{CA}(H(A\|\Pi_A\|L))$
2.  $A, \Pi_A, L, S_{CA}(H(\Pi_A\|L))$
3.  $A, \Pi_A, L, S_{CA}(H(A\|\Pi_A))$
4.  $A, \Pi_A, L, S_{CA}(A\|\Pi_A\|L)$

**SECURITY IN NETWORKED COMPUTING SYSTEMS***Computer Engineering***15 July 2015****SOLUTION****Exercise n.1**

For the relationship between performance and key-bit configuration see theory. From that relationship, it follows that the number of multiplications depends on the number of bits equal to 1.

We choose those values for  $e$  because they contain only two 1's in their binary representation.

Selecting  $d$  so that it has only two 1's in its binary representation would reduce the private key space and make it vulnerable to exhaustive key search.

If  $e$  contains only one 1 in its binary representation, then its value is either 1 or even. In both case, it does not fulfill the constraint specified by the RSA key generation algorithm.

**Exercise n.2**

**Case 1 is insecure.** The adversary guesses a password, computes a key and decrypts RSP. Then (s)he compares the resulting plaintext with  $r_a$ . It follows the complexity of the attack is equal to the password-guessing attack.

**Case 2 is insecure.** The adversary guesses a password and computes a key. Then (s)he computes the MAC of  $r_a$  and checks whether it is equal to RSP. It follows the complexity of the attack is equal to the password-guessing attack.

**Case 3 is insecure.** The adversary guesses a password, computes a key and decrypts RSP. Then (s)he compares the resulting plaintext with the first field of the resulting plaintext. It follows the complexity of the attack is equal to the password-guessing attack.

**Case 4 is insecure.** A is not able to verify RSP because (s)he does not know  $r_b$ .

**Case 5 is insecure.** The adversary guesses a password and computes a key. Then (s)he computes the MAC of  $r_a$ , encrypts it by means of  $\Pi_A$ , and checks whether it is equal to RSP. It follows the complexity of the attack is equal to the password-guessing attack.

**Case 6 is secure,** because  $r_b$  randomizes RSP.

**Exercise n. 3**

**Case 1 and 4** are secure.

**Case 2** is insecure because it doesn't link the user identifier to his/her key.

**Case 3** is insecure because it doesn't link the period of validity to the user's key and user's identifier

# 15 July 2015

martedì 23 maggio 2017 14:43



sncs-150701

## Exercise number 1

1. Square-and-multiply algorithm
2. See notes
3. No. Altrimenti si ridurrebbe lo spazio delle chiavi.
4. Uno non cifra niente, e un numero pari non serve a niente.  
e dev'essere co-primo rispetto a phi = (p -1)(q -1).

## Exercise number 2

3. I falsi positivi (di K) possono essere eliminati analizzando altre coppie di r\_a ed r\_b.
4. L'avversario per fare l'attacco dovrebbe provare tutti i possibili r\_b per ogni password da provare.  
L'attacco è difficile, ma A non conosce r\_b.
5. La chiave pubblica ce l'ha anche l'avversario, perciò la sfrutta sempre.  
Prova un K (generato da una P), fa l'hash e la cripta con la chiave pubblica di A.  
Siccome l'avversario parte dalla password, l'attacco è molto semplice.
6. Lo stesso attacco nel 5, ma per ogni chiave K generata, occorre provare tutti i valori di r\_b.  
r\_b di solito ha la stessa dimensione delle chiavi, perciò si tratta di tanti valori da provare.

## Exercise number 3

1. Certificato giusto, con l'Hash che lega le tre entità.  
È una versione più efficiente da calcolare.
2. Non c'è legame tra
3. .
4. Certificato classico, versione meno efficiente della digital signature.

**SECURITY IN NETWORKED COMPUTING SYSTEMS**

*Computer Engineering*

**22 July 2015**

**EXERCISE NO. 1**

**#MARKS: 10**

With reference to a perfect cipher,

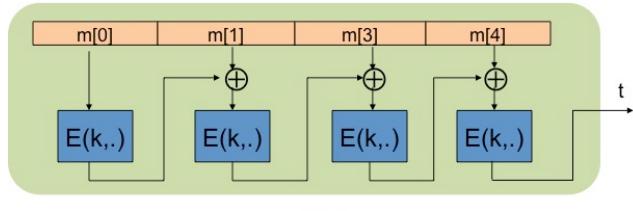
1. State the Shannon's formal definition of perfect cipher and give an intuitive explanation of the definition;
2. Prove that in a perfect cipher the number of keys cannot be smaller than the number of messages;
3. Argue whether an asymmetric cipher can be perfect or not.

**EXERCISE NO. 2**

**#MARKS: 10**

State the security definition of a Message Authentication Code (MAC).

List and briefly introduce the main methods of building a MAC out of other cryptographic primitives.

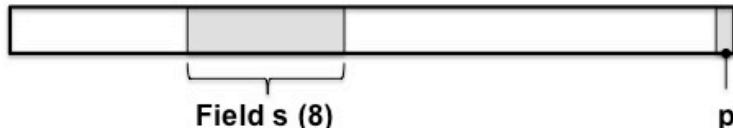


Show that the rawCBC-MAC (see figure) is insecure. Hint: compute the MAC  $t_1$  of the single block message  $m_1$  and the MAC  $t_2$  of the two-block message  $m_2 = (m_1, m_1 \oplus t_1)$ . Generalize the attack for an arbitrary number of blocks message.

**EXERCISE NO. 3**

**#marks: 10**

Let us assume that a plaintext  $P$  has the format specified in the figure where  $s$  is an 8-bit field that specifies an amount of money and  $p$  is a *parity bit* s.t.



$p$  is 0 if the number of 1s in the plaintext (bit  $p$  excluded) is even; it is 1 otherwise. The whole plaintext is encrypted by means of one-time-pad.

Q1. Does this encryption scheme suffer from malleability? Motivate the answer.

Q2. Assume that field  $s$  specifies the value 130. Argue whether and how, it is possible to modify the cipher-text so that the decrypted plaintext specifies 146 in the field  $s$  and such a modification goes undetected.

Q3. Propose a possible countermeasure.

**SECURITY IN NETWORKED COMPUTING SYSTEMS***Computer Engineering***22 July 2015****SOLUTION****Exercise n.1**

Q1, Q2, Q3. See theory.

**Exercise n.2**

Q1. See theory

Q2. See theory.

Q3. Compute  $t_1$  and  $t_2$  as hinted, and verify that  $t_1 = t_2$ . This means that a collision has arisen. One possible way to generalize is to consider a message having the following structure  $m = (m_1, m_1 \oplus t_1, \dots, m_1 \oplus t_1)$ , where  $m_1$  is a single block message and  $t_1$  is  $m_1$ 's tag.

**Exercise n. 3**

**Q1.** The encryption scheme is malleable. A simple way to prove it is the following. Let  $P[i]$ ,  $C[i]$ , and  $K[i]$  be the  $i$ -th bit of the plaintext, ciphertext and key, respectively, s.t.  $C[i] = P[i] \text{ xor } K[i]$ . Notice that  $P[0] = p$ . Finally let  $C'$  be the modified ciphertext and  $P'$  the resulting plaintext after decryption. Notice that an adversary can easily complement a bit of the plaintext by operating on the ciphertext. Assume that the adversary wishes to complement bit  $i$ . Then, (s)he computes  $C'[i] = C[i] \text{ xor } 1 = (P[i] \text{ xor } K[i]) \text{ xor } 1 = (P[i] \text{ xor } 1) \text{ xor } K[i] = P'[i] \text{ xor } K[i]$ . It follows that  $P'[i] = P[i] \text{ xor } 1$ , that is,  $P'[i]$  is the complement of  $P[i]$  ( $P'[i] = \sim P[i]$ ).

In order for the attack to go undetected, and the scheme to be malleable, the parity bit must be consistently modified as well. Notice that since  $P[i]$  is complemented the number of 1 either increment or decrement by one. In both cases the parity bit has to be complemented as well. This implies that  $C'[0] = C[0] \text{ xor } 1$ .

**Q2.** The attack consists in complementing  $C[0]$  and the 5-th most significant bit in  $s$ .

**Q3.** The problem can be solved by replacing the parity bit by a tag resulting from a secure hash function.

# SECURITY IN NETWORKED COMPUTING SYSTEMS

Master in Computer Engineering

14 September 2015

❖: all

◆: all but DSS

## EXERCISE NO. 1 (❖)

#MARKS: 10

With reference to the CBC encryption mode,

1. Illustrate the scheme and present the related equations;
2. Discuss the advantages and disadvantages w.r.t. ECB;
3. Argue whether it can be used with an asymmetric cipher.

## EXERCISE NO. 2 (◆)

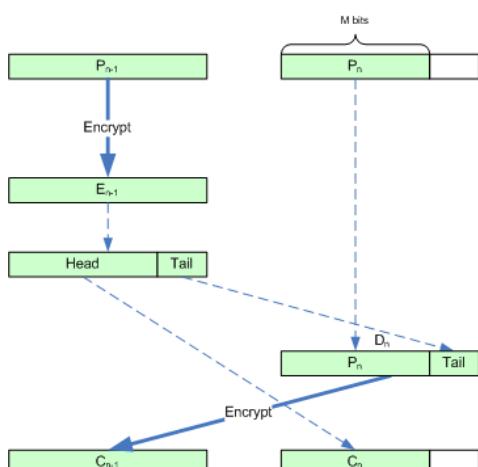
#MARKS: 10

Alice and Bob wish to establish a session key  $K_{AB}$ . To this purpose they run a key establishment protocol that exploits the presence of Trent, a trusted third party that plays the key server role. Alice and Bob share a long-term secret key with Trent. Let  $K_A$  and  $K_B$  these keys, respectively. Design the key establishment protocol that has to fulfil the following requirements:

- A. Clocks are not synchronized;
- B. The protocol is not subject to replay;
- C. At the end of the protocol, each principal has the proof that the peer holds the key;
- D. Alice cannot talk to Trent directly but indirectly through Bob

## EXERCISE NO. 3 (❖)

#marks: 10



The picture shows the encryption steps of the ECB ciphertext stealing (ECB-CTS) encryption mode. This mode allows for processing of messages that are not evenly divisible into blocks without resulting in any expansion of the ciphertext (no padding is required).

In the picture, broken arrows denote bit copy whereas solid arrows labeled “Encryption” denote encryption by means of key  $K$ .

1. Determine the decryption steps.
2. Argue about bit-error propagation

In answering the questions assume: i) blocks are numbered from 1 to  $n$ ; ii)  $B$ -bit block; iii) function  $Head(blk, n)$  that returns the  $n$  most-significant bits of block  $blk$ ; iv) function  $Tail(blk, n)$  that returns the  $n$  least-significant bits of block  $blk$ .

# SECURITY IN NETWORKED COMPUTING SYSTEMS

*Master in Computer Engineering*

14 September 2015

## SOLUTION

### EXERCISE N.1

*See theory.*

### EXERCISE N.2

M1	$A \rightarrow B:$	$A, B, n_A$
M2	$B \rightarrow T:$	$A, B, n_A, n_B$
M3	$T \rightarrow B:$	$\{A, B, n_A, K_{AB}\}_{K_A}, \{A, B, n_A, n_B, K_{AB}\}_{K_B}$
M4	$B \rightarrow A:$	$\{A, B, n_A, K_{AB}\}_{K_A}, \{A, B, n_A, n_B\}_{K_{AB}}$
M5	$B \rightarrow A:$	$\{A, B, n_B, n_A\}_{K_{AB}}$

### EXERCISE N.3

#### Question #1.

$$P_i = \mathcal{D}(K, C_i), 1 \leq i < n-1$$

$$D_n = \mathcal{D}(K, C_{n-1})$$

$$P_n = \text{Head}(D_n, M)$$

$$E_{n-1} = C_n \parallel \text{Tail}(D_n, B - M)$$

$$P_{n-i} = \mathcal{D}(K, E_{n-1})$$

#### Question #2.

A bit-error in any block  $C_i, i < n-1$ , affects  $P_i$  only.

A bit-error in  $C_{n-1}$  causes the block-wide loss of  $P_n$  and  $P_{n-1}$ .

A bit-error in  $C_n$  causes the block-wide loss of  $P_{n-1}$ .

# 14 September 2015

martedì 23 maggio 2017 15:17



sncs-150914

## Exercise 1

1. .
  2. .
  3. In linea teorica, un cifrario asimmetrico può essere utilizzato anche per grosse quantità. in cui ogni blocco può avere una rappresentazione minore di n.
- ? Si può fare (non conviene dal punto di vista delle prestazioni),**

## Exercise 2

- My unfinished attempting

1	$A \rightarrow B$	$A, n_a, \{A, B\}_Ka$
2	$B \rightarrow T$	$\{A, B\}_Ka$
3	$T \rightarrow B$	$\{Kab\}_Kb, \{Kab\}_Ka$
4	$B \rightarrow A$	$\{Kab\}_Ka$

- I basically have to copy Kerberos
- Especially for tickets and authenticators that will confirm keys

## Exercise 3 (molto molto facile)

CTS is an encrypting mode that avoids expanding the ciphertext by adding padding.

1. O in forma algoritmica, o come disegno.
  - Algoritmica (mia)
    - È giusta, ma più in generale, avrei dovuto scrivere come si decrittano **tutti** i blocchi, e non solo gli ultimi due (mostrati in figura, di incide n-1 ed n)

$P_n = \text{Head}(D(C_{n-1}), M)$   
 $P_{n-1} = D(\text{Tail}(D(C_{n-1}), B-M) || C_n)$

- Algoritma (prof)

$P_i = D(K, C_i), i = 1, n-2$   
 $D = D(K, C_{n-1})$   
 $P_n = \text{Head}(D, M)$   
 $E = C_n || \text{Tail}(D, B-M)$   
 $P_{n-1} = D(K, E)$

2.  $P'_i = D(K, C'_i)$ ,  $i = 1, n-2$

$P'_i$  viene cambiato in maniera random, quindi se  $C_i$  viene alterato, viene perso tutto il suo corrispondente blocco.

$$D' = D(K, C'_n)$$

anche questo è random

$$P'_n = \text{Head}(D', M)$$

$$E' = C_n \mid\mid \text{Tail}(D', B-M)$$

$$P'_n = D(K, E')$$

Un errore su  $C_n$  si riflette sia su  $P_{n-1}$  che su  $P_n$  (situazione peggiore)

Sempre da sopra, si vede che un errore su  $C_n$  si propaga solo su  $P_{n-1}$  (anche dalla figura si vede molto bene).

05 February 2016

❖: all

◆: all but LMECS

■: only LMECS

**EXERCISE NO. 1 (❖)**

**#MARKS: 10**

With reference to the Shannon's theory,

1. Give the definition of perfect cipher;
2. Give the physical interpretation of the definition;
3. Prove the Shannon's theorem.

**EXERCISE NO. 2 (❖)**

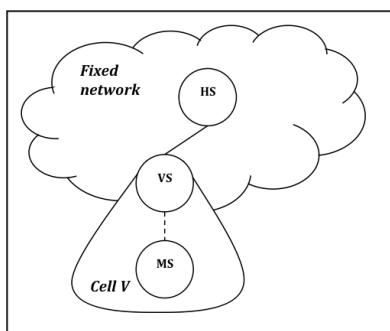
**#MARKS: 10**

Let  $K_A$  be the public key of Alice,  $S_P(x)$  be the digital signature of principal  $P$  on item  $x$ , CA be a Certification Authority (trusted by all principals of the system), and finally  $H()$  a secure hash function. Validity period apart, which of the following certificates are useful to establish a secure channel with Alice? Argue why.

- (A) "Alice" ||  $S_{CA}(H("Alice" || K_A))$
- (B) "Alice" ||  $K_A$  ||  $S_A(H("Alice" || K_A))$
- (C) "Alice" ||  $K_A$  ||  $S_{CA}("Alice")$
- (D) "Alice" ||  $K_A$  ||  $S_{CA}("Alice" || H(K_A))$
- (E) "Alice" ||  $K_A$  ||  $S_{CA}(H("Alice" || K_A))$
- (F) "Alice" ||  $K_A$  ||  $S_{CA}(K_A)$
- (G) "Alice" ||  $K_A$  ||  $S_B("Alice" || H(K_A) || "issuer: Bob")$  ||  $S_{CA}("Bob" || K_B)$
- (H) "Alice" ||  $K_A$  ||  $S_B("Alice" || H(K_A) || "issuer: Bob")$  ||  $S_{CA}("Bob" || "CA=Yes" || K_B)$

**EXERCISE NO. 3 (◆)**

**#MARKS: 10**



In a roaming system, a mobile station MS, whose home server is HS, is visiting a cell V served by server VS. MS and HS share the key  $k_{mh}$ . Furthermore, HS and VS belong to the fixed infrastructure and share the key  $k_{vh}$ . Design a key establishment protocol that fulfils the following requirements:

1. Establish of secret shared key  $k_{mv}$  between MS and VS;
2. Mutual authentication between VS and MS;
3. Resistance against replay attacks.

Assume that MS, HS e VS use the same cipher  $E()$  and that their clocks synchronized. Analyse the protocol by means of the BAN logic and argue that it fulfils requirements 1–3. What if the clocks are not synchronized?

**EXERCISE NO. 4 (■)**

**#MARKS: 10**

With reference to the CBC encryption mode,

4. Draw the encryption and decryption scheme;
5. State the CBC equations;
6. Briefly discuss pros and cons w.r.t. ECB.

05 February 2016

# SOLUTION

## EXERCISE N.1

*See theory.*

## EXERCISE N.2

- A. "Alice" ||  $S_{CA}(H("Alice" || K_A))$ . Not valid because it doesn't carry a key.
- B. "Alice" ||  $K_A$  ||  $S_A(H("Alice" || K_A))$ . Not valid because it is self-signed and Alice is not trusted.
- C. "Alice" ||  $K_A$  ||  $S_{CA}("Alice")$ . Not valid because key and name are not linked together.
- D. "Alice" ||  $K_A$  ||  $S_{CA}(Alice" || H(K_A))$ . Valid.
- E. "Alice" ||  $K_A$  ||  $S_{CA}(H("Alice" || K_A))$ . Valid.
- F. "Alice" ||  $K_A$  ||  $S_{CA}(K_A)$ . Not valid because key and name are not linked together.
- G. "Alice" ||  $K_A$  ||  $S_B("Alice" || H(K_A) || "issuer: Bob")$  ||  $S_{CA}("Bob" || K_B)$ . Not valid because the certificate is issued by Bob who is not trusted to do so.
- H. "Alice" ||  $K_A$  ||  $S_B("Alice" || H(K_A) || "issuer: Bob")$  ||  $S_{CA}("Bob" || "CA=Yes" || K_B)$ . Valid because the certificate is issued by Bob who has been delegated by the CA to do so.

## EXERCISE N. 3

**The protocol**

PROTOCOL (real)

1)  $MS \rightarrow VS : MS, VS, t_m$

2)  $V S \rightarrow H S : MS, VS, t_m, t_v$

3)  $H S \rightarrow V S : \{ MS, VS, t_m, t_v, k_{vh} \}_{K_{hm}}$ ,

$\{ VS, MS, t_v, t_m, k_{vh} \}_{K_{hv}}$

4)  $V S \rightarrow M S : \{ MS, VS, t_m, t_v, K_{vh} \}_{K_{vm}}$ ,

$\{ VS, MS, t_m \}_{K_{vm}}$

5)  $M S \rightarrow V S : \{ MS, VS, t_v \}_{K_{vm}}$

**Assumptions**

ASSUMPTIONS (only the most important ones)

1.  $VS \models HS \Rightarrow (MS \xleftarrow{k_{mv}} VS)$

2.  $MS \models HS \Rightarrow (MS \xleftarrow{k_{mv}} VS)$

3.  $HS \models (MS \xleftarrow{k_{mv}} VS)$

4.  $VS \models \#(t_v)$

5.  $MS \models \#(t_m)$

Proof

PROOF (sketch)

AFTER M3

$$VS \models MS \xleftarrow{k_{vm}} VS \quad (A1)$$

AFTER M4

$$MS \models MS \xleftarrow{k_{vm}} VS \quad (A2)$$

$$MS \models VS \models MS \xleftarrow{k_{vm}} VS \quad (A3)$$

AFTER M5

$$VS \models MS \models MS \xleftarrow{k_{vm}} VS \quad (A4)$$

# SECURITY IN NETWORKED COMPUTING SYSTEMS

*Master in Computer Engineering, Master in Embedded Computing Systems*

16 February 2016

Candidate \_\_\_\_\_ Serial no. \_\_\_\_\_

## EXERCISE NO. 1

#MARKS: 12

- (1) Describe the Diffie-Hellmann protocol
- (2) Argue about its security w.r.t. a passive adversary
- (3) Argue about its security w.r.t. a MITM attack

## EXERCISE NO. 2

#MARKS: 12

Alice and Bob use one-time pad (OTP) and have agreed on a perfectly random key K. Alice will send Bob the answer to the question "Are you taking SNCS?" as either 'Y' or 'N' encoded in their ASCII representation (1011001 and 1001110, respectively). The adversary knows nothing about the key but intercepts the cipher-text  $c$  exchanged between Alice and Bob: 1001110.

- (1) What cipher-text  $c'$  should the adversary send Bob to flip Alice's message?
- (2) Alice decides to use a secure hash function  $h(\cdot)$  and transmits the cypher-text  $c''$  obtained from enciphering  $m \parallel h(m)$ , where  $m$  is the clear-text message and  $\parallel$  is the concatenation operator. Can it be of any help?
- (3) Without using anything other than OTP, how can Alice and Bob solve this problem? (Hint: use two keys)

## EXERCISE NO. 3

#MARKS: 6

1. What are certificates for?
  - A. Establishing an indissoluble link between an identifier and a public key.
  - B. Establishing an indissoluble link between a public key and the owner of the certificate.
  - C. Establishing the privileges of the owner of the certificate.
  - D. Establishing the trustworthiness of the certificate owner.
2. What are the Certification Authority's obligations before releasing a certificate?
3. Describe the minimum set of data fields that you expect to find in a certificate.

# SECURITY IN NETWORKED COMPUTING SYSTEMS

*Master in Computer Engineering, Master in Embedded Computing Systems*

16 February 2016

## SOLUTION

### EXERCISE N.1

*See theory.*

### EXERCISE N.2

#### **Question 1.**

Let  $c = msg \square K$  where  $msg$  may assume two values  $mY$  and  $mN$ . In order to flip Alice's message, the adversary has to transmit  $c' = c \oplus (mY \oplus mN)$ .

Substituting the exercise data we obtain the following.  $(mY \oplus mN) = 1011001 \oplus 1001110 = 0010111$ . Therefore,  $c' = c \oplus (mY \oplus mN) = 1001110 \oplus 0010111 = 1011001$ .

#### **Question 2.**

No. Let  $c = (msg \parallel H(msg)) \oplus K$  where  $msg$  may assume two values  $mY$  and  $mN$ . In order to flip Alice's message, the adversary has to transmit the cipher-text  $c'$  s.t.

$$c' = c \oplus (mY \parallel H(mY) \oplus (mN \parallel H(mN))).$$

#### **Question 3.**

A possible solution requires two keys,  $KN$  and  $KY$  such that  $KY \oplus KN \neq mY \oplus mN$  (*Uniqueness condition*). Alice and Bob share these keys. Alice transmits  $c$  such that

$$c = \begin{cases} mY \oplus KY & \text{if answer is yes} \\ mN \otimes KN & \text{if answer is no} \end{cases}$$

Upon receiving a cipher text  $c$ , Bob i) computes  $\bar{mY} = c \oplus kY$ ,  $\bar{mN} = c \oplus KN$ ; and, ii) returns message  $m$  such that:

$$m = \begin{cases} mY & \text{if } \bar{mY} = mY \\ mN & \text{if } \bar{mN} = mN \\ \perp & \text{otherwise} \end{cases}$$

The Uniqueness Condition guarantees that  $mN$  cannot be derived from  $cY$  and vice versa. Notice that in order to flip Alice's answer the adversary must be able to obtain  $cN$  from  $cY$  and vice versa. This requires the adversary to know  $KY \oplus KN$ . However, this is not possible given the Uniqueness condition, the secrecy and randomness of the keys.

# **SECURITY IN NETWORKED COMPUTING SYSTEMS**

*Master in Computer Engineering, Master in Embedded Computing Systems*

**16 February 2016**

## **EXERCISE N.3**

**Question 1.** Option B.

**Question 2.** Identify the subject and authenticate the key.

**Question 3.** Subject identifier (S), public key (pK), validity period (V), a digital signature on  $S \parallel pK \parallel V$  by the Certification Authority.

## SECURITY IN NETWORKED COMPUTING SYSTEMS

20 JUNE 2016

♦: LMCE+LMECS

■:LMCE

### EXERCISE NO. 1 (♦)

MARKS: 10

With reference to the RSA crypto-system,

1. Illustrate the key generation, encryption, decryption algorithms;
2. Argue why it is considered secure;
3. Argue whether it can be considered perfect according to Shannon's theory.

### EXERCISE NO. 2 (♦)

MARKS: 10

In an electronic auction, bidder B cast his bid  $\beta$  encrypting it by mean of the auctioneer Alice's public key  $\Pi_A$ . Let us assume that a bid is an integer number on  $b$ -bits,  $b = 32$ . Argue whether the following protocols are secure w.r.t. to a passive adversary.

1.  $B \rightarrow A: \{B, \beta\}_{\Pi_A}$
2.  $B \rightarrow A: \{B, \beta, H(\beta)\}_{\Pi_A}$
3.  $B \rightarrow A: \{B, H(\beta)\}_{\Pi_A}$
4.  $B \rightarrow A: \{B, \rho, \beta\}_{\Pi_A}$
5.  $B \rightarrow A: \{B, K\}_{\Pi_A}, \{B, \beta\}_K$

where  $H$  is a secure hash function whose output is  $r$ -bit,  $\rho$  is a random number of  $r$ -bits and,  $K$  is a random cryptographic key on  $k$  bits. Bob generates  $\rho$  and  $K$  upon casting his bid. Assuming that the encryption and the hash function are secure, determine the size in bit of  $\rho$  and  $K$  so that an attack requires  $2^{128}$  steps.

### EXERCISE NO. 3 (■)

MARKS: 10

With reference to the Kerberos system, argue the need for the Ticket Granting Service (TGS)?

# June 20th, 2016

martedì 23 maggio 2017 16:22

## Exercise 1

With reference to RSA,

1. .
2. .
3. .

## Exercise 2

In an electronic auction, bidder B cast his bid BETA encrypting it by means of the auctioneer Alice's public key  $\pi_A$ .

Let us assume that a bid is an integer number on b-bits,  $b = 32$ .

Argue whether the following protocols are secure w.r.t. a passive adversary.

1.  $B \rightarrow A: \{B, \text{Beta}\}_{\pi_A}$
2.  $B \rightarrow A: \{B, \text{Beta}, H(\text{Beta})\}_{\pi_A}$
3.  $B \rightarrow A: \{B, H(\text{Beta})\}_{\pi_A}$
4.  $B \rightarrow A: \{B, p, \text{Beta}\}_{\pi_A}$
5.  $B \rightarrow A: \{B, K\}_{\pi_A}, \{B, \text{Beta}\}_K$

Where  $H$  is a secure hash function whose output is t-bit,  $p$  is a random number of r-bits and,  $K$  is a random cryptographic key on k bits. Bob generates  $p$  and  $K$  upon casting his bid.

Assuming that the encryption and the hash function are secure, determine the size in bit of  $p$  and  $K$  so that an attack requires  $2^{128}$  steps.

## Solution



June 20th,

2016

1. Se l'avversario ha un'idea di Beta, non ha nemmeno bisogno di provare  $2^{32}$  valori per Beta.
  2. Uguale al primo ma con un hash in più.
  3. ...
  4. Dipende dalla lunghezza del numero random  $p$ .
- Beta is on 32 bits, therefore  $p$  has to be at least  $128 - 32 = 96$  bits long.
5. L'avversario deve:
    - a. Provare le possibili chiavi

## Exercise 3

**SECURITY IN NETWORKED COMPUTING SYSTEMS**

September 20, 2016

Name \_\_\_\_\_ Serial nr. \_\_\_\_\_

**EXERCISE NO. 1 (ALL)**

#MARKS: 12

With reference to the perfect ciphers,

- 1 Give the Shannon's definition;
- 2 Give an intuitive practical interpretation of the definition;
- 3 Prove the Shannon's theorem;
- 4 Argue under which assumptions one-time pad is perfect.

**EXERCISE NO. 2 (ALL)**

#MARKS: 8

When using the one-time pad (OTP) encryption scheme, it can occur that  $k = \{0\}^*$ , that is the key is a sequence of 0's. In this case ciphertext is equal to the plaintext.

Alice suggests to improve the one-time pad by only choosing non-zero keys. What do you think of this improvement? In particular, is the improved OTP still perfectly secret?

Let us suppose that you receive the message "Attack". In the improved OTP is it more likely that the plaintext is "Attack", "Defend" or "Rabbit"? What about in the original-OTP?

**EXERCISE NO. 3 (LMCE)**

#MARKS: 10

Kerberos supports delegation.

- 1 Explain the delegation problem.
- 2 Illustrate the proxy ticket solution
- 3 Argue pros and cons of proxy tickets w.r.t. forwardable tickets

# September 20th, 2016

martedì 23 maggio 2017 15:46

Not found on the professor's filesystem.

Here's a copy of it:

## Exercise 1

With reference to the perfect ciphers,

1. Give the Shannon's definition
2. Give an intuitive practical interpretation of the definition
3. Prove the Shannon's theorem
4. Argue under which assumptions OTP is perfect

## Exercise 2

When using the one-time pad (OTP) encryption scheme, it can occur that  $k = [0]^*$ , that is the key is a sequence of 0's. In this case ciphertext is equal to the plaintext.

Alice suggests to improve the OTP by only choosing non-zero keys.

What do you think of this improvement? In particular, is the improved OTP still perfectly secret?

Let's suppose that you receive the message "Attack". In the improved OTP is it more likely that the plaintext is "Attack", "Defend" or "Rabbit"? What about in the original-OTP?

### *Solution*

Non è perfetto perché la probabilità a priori e a posteriori sono diverse.

Se si è in grado di dimostrare che  $|K| < |P|$ , è fatta.



September  
20th, 2016

Le 3 stringhe son 3 possibili plaintext.

Nell'OTP canonico e nell'OTP modificato, cambia la probabilità che il plaintext sia uno di quei 3 se il ciphertext è "Attack"?

- OTP-canonico: l'avversario non può sapere quale plaintext ha più probabilità
- OTP-modificato: avendo levato  $k = [0]^*$ , di sicuro il plaintext non è "Attack", la sua probabilità a priori non è più la stessa.  
Per quanto riguarda gli altri due ciphertext, non si conosce niente, nemmeno la probabilità legata.

## Exercise 3



September  
20th, 2016  
Kerberos supports delegation.

1. Explain the delegation problem
2. Illustrate the proxy ticket solution
3. Argue pros and cons of proxy tickets w.r.t. forwardable tickets

**LAUREA SPECIALISTICA IN INGEGNERIA INFORMATICA**  
**SICUREZZA NELLE RETI**  
**Appello del 16 Gennaio 2005**

Nome e Cognome \_\_\_\_\_ Matricola \_\_\_\_\_

**ESERCIZIO 1** **PUNTI: 8**

**(A.A. 2004-05).** Il candidato spieghi con precisione matematica e proprietà di linguaggio il legame tra  $n$ , il numero di bit di un blocco, e  $k$ , il numero di bit di una chiave, in un *true random cipher*. Il candidato discuta inoltre le difficoltà pratiche legate alla realizzazione di tale tipo di cifrario.

**(A.A. 2005-06).** Il candidato dimostri che se un cifrario è perfetto allora il numero delle chiavi deve essere maggiore o uguale al numero dei messaggi.

**ESERCIZIO 2** **PUNTI: 14 (4, 5, 5)**

Si consideri il seguente protocollo di distribuzione delle chiavi orientato a stabilire una chiave di sessione  $K_{AB} = h(k_a, k_b)$  tra i processi  $A$  e  $B$ , con  $h$  funzione hash one-way:

M1	$A \rightarrow B: E_{e_b}(k_a, A)$	<ul style="list-style-type: none"><li>• <math>E</math>: cifrario asimmetrico;</li></ul>
M2	$B \rightarrow A: E_{e_a}(k_a, k_b)$	<ul style="list-style-type: none"><li>• <math>k_a</math> e <math>k_b</math>: quantità segrete scelte rispettivamente da <math>A</math> e <math>B</math>;</li></ul>
M3	$A \rightarrow B: E_{e_b}(k_b)$	<ul style="list-style-type: none"><li>• <math>e_a</math> ed <math>e_b</math>: le chiavi pubbliche, rispettivamente, di <math>A</math> e <math>B</math></li></ul>

Si assuma che

- le quantità  $k_a$  e  $k_b$  non siano mai riutilizzate;
- ciascun processo conosca la chiave pubblica dell'altro, il candidato risponda alle seguenti domande motivando le risposte.

**Quesito A.** Il protocollo garantisce la confidenzialità della chiave di sessione?

**Quesito B.** Analizzando il protocollo con la logica BAN, il candidato discuta se il protocollo garantisce le proprietà di key authentication e di key confirmation specificando le ipotesi sotto le quali tali proprietà valgono.

Si assuma adesso che l'ipotesi (i) non sia più verificata.

**Quesito C.** Il candidato verificata, quali conseguenze possono scaturire.

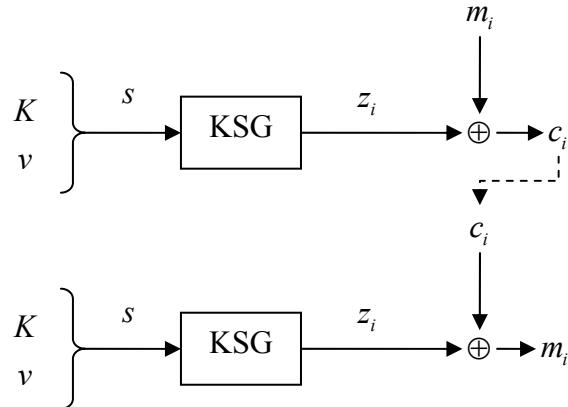
**ESERCIZIO 3** **PUNTI: 8 (4, 4)**

Si consideri il protocollo (semplificato) di identificazione di WEP (IEEE 802.11) per mezzo del quale una mobile station MS si identifica presso un access point AP per avere accesso alla rete (sia  $k$  una chiave segreta su 104 bit, condivisa tra MS ed AP):

1. MS invia una richiesta di identificazione ad AP.
2. AP genera in modo casuale una challenge  $\chi$  su 128 bit e la invia a MS.
3. MS genera in modo casuale un vettore di inizializzazione  $v$  su 24 bit, calcola la response  $\rho = E_{k,v}(\chi)$  ed invia ad AP la frame  $\langle v, \rho \rangle$ .
4. Alla ricezione della frame, AP estrae le quantità  $v$  e  $\rho$ , e verifica se  $\chi = D_{k,v}(\rho)$ . Se la verifica ha esito positivo, AP permette ad MS l'accesso alla rete; altrimenti, vieta tale accesso.

**LAUREA SPECIALISTICA IN INGEGNERIA INFORMATICA**  
**SICUREZZA NELLE RETI**  
**Appello del 16 Gennaio 2006**

Il cifrario utilizzato è definito in Figura 1, dove  $m_i$  e  $c_i$  sono, rispettivamente, l' $i$ -esimo byte del testo in chiaro e del testo cifrato. La chiave  $K$  ed il vettore di inizializzazione  $v$  sono concatenati per formare il seme  $s$  del generatore random sicuro di byte detto *Key Sequence Generator* (KSG). Con  $z_i$  si denota l' $i$ -esimo byte del *key stream* generato da KSG.



**Figura 1. Il cifrario di WEP.**

Si assuma che la chiave  $K$  è fissa ed unica per tutte le mobile station appartenenti alla rete.

**Quesito A.** Il candidato discuta se il protocollo WEP garantisce l'identificazione oppure no ovvero se un avversario (passivo) può impersonare una mobile station della rete.

Si assuma adesso di modificare il protocollo in modo tale che, al passo 2, AP generi in modo causale il vettore di inizializzazione  $v$  e lo invii in chiaro a MS insieme alla challenge<sup>1</sup>.

**Quesito B.** Il candidato definisca un dictionary attack contro questa variante del protocollo di identificazione e valuti la dimensione in byte del dizionario.

---

<sup>1</sup> Si noti che adesso non è più necessario trasmettere  $v$  insieme alla response.

**LAUREA SPECIALISTICA IN INGEGNERIA INFORMATICA**  
**SICUREZZA NELLE RETI**  
**Appello del 16 Gennaio 2006**

## SOLUZIONE

### ESERCIZIO 1

(A.A. 2004-05). Con  $n$  bit si hanno  $2^n$  possibili blocchi in chiaro. Un true random cipher realizza tutte le possibili permutazioni, cioè  $2^n!$  e richiede perciò  $k = \log_2(2^n!)$  che risulta essere  $O(n2^n)$ .

(A.A. 2005-06). Per prima cosa si osserva che affinché il cifrario sia invertibile è necessario che il numero di testi cifrati  $N_c$  sia maggiore o uguale al numero di testi in chiaro  $N_m$ ,  $N_c \geq N_m$ . La dimostrazione è per assurdo. Supponiamo che il numero di chiavi  $N_k$  sia minore del numero  $N_m$  di messaggi in chiaro,  $N_k < N_m$ . Ne segue quindi che  $N_k < N_c$ .

Sia  $m$  un messaggio che ha probabilità non nulla di essere trasmesso,  $P(M = m) \neq 0$ . Dalla condizione  $N_k < N_c$  segue che esiste un crittogramma  $c$  che non è immagine di  $m$ . Perciò,  $P(M = m, C = c) = 0 \neq P(M = m)$ , contro l'ipotesi che il cifrario sia perfetto.

### ESERCIZIO 2

**Quesito A.** Il protocollo garantisce la confidenzialità. Per determinare il valore di  $K_{AB}$  è necessario conoscere le quantità  $k_a$  e  $k_b$ . Tuttavia queste quantità viaggiano in rete in forma cifrata.

**Quesito B.** Il protocollo è una versione semplificata del protocollo Needham-Schroeder a chiave pubblica. Il protocollo garantisce sia la key authentication sia la key confirmation. Formalmente,

$$A \equiv B \equiv A \xrightarrow{k_b} B \text{ e } B \equiv A \equiv A \xrightarrow{k_a} B.$$

**Quesito C.** Se l'ipotesi (i) non è verificata, il protocollo non garantisce la proprietà di key authentication. Supponiamo che la quantità  $k_a$  sia riutilizzata da  $A$  e che un avversario  $M$  abbia registrato i messaggi M1 ed M2 relativi all'esecuzione del protocollo in cui  $k_a$  è stata utilizzata la prima volta. L'avversario  $M$  potrebbe eseguire il seguente attacco:

- l'avversario  $M$  induce  $A$  ad iniziare una nuova istanza del protocollo con  $B$ ;
- quando  $A$  invia il messaggio M1' relativo alla nuova esecuzione del protocollo, contenente la quantità riutilizzata  $k_a$ , l'avversario  $M$  determina che M1=M1', e risponde con M2. Alla ricezione di questo messaggio il processo A crede di parlare effettivamente con  $B$ .

Si noti che questo attacco ha come effetto collaterale il riutilizzo della vecchia chiave di sessione  $K_{ab}$  (la quantità  $k_b$  è contenuta nel messaggio M2 replicato da  $M$ ). L'avversario  $M$  potrebbe non conoscere tale chiave ma potrebbe comunque replicare vecchi messaggi relativi alla sessione  $K_{ab}$  che  $A$  considererebbe come provenienti da  $B$ . Il danno sarebbe massimo se  $M$ , per altre vie, fosse riuscito ad impadronirsi di  $K_{ab}$ .

Considerazioni simili possono essere fatte per  $B$  se questo processo riutilizza la quantità  $k_b$ .

**LAUREA SPECIALISTICA IN INGEGNERIA INFORMATICA**

**SICUREZZA NELLE RETI**

**Appello del 16 Gennaio 2006**

**ESERCIZIO 3**

**Quesito A.** Il protocollo non garantisce l'identificazione. Supponiamo che un avversario  $A$  intercetti una coppia challenge-response. Più precisamente,  $A$  intercetta la challenge e la frame  $f$  che trasporta la relativa response. Tale frame oltre a response trasporta anche il vettore di inizializzazione  $v$  utilizzato per cifrare la challenge. Dalla challenge (testo in chiaro) e dalla response (crittogramma) è possibile ricavare il keystream  $\{z_i(v)\}$  facendo l'or-esclusivo tra le due quantità. A questo punto l'avversario conosce una coppia (vettore di inizializzazione, keystream)  $(v, \{z_i(v)\})$  che può riutilizzare in (un numero indefinito di) altre esecuzioni del protocollo di identificazione.

**Quesito B.** Come nel caso precedente l'avversario riesce a violare il protocollo riutilizzando una coppia  $(v, \{z_i(v)\})$ . Il keystream può essere ricavato come nel caso precedente mentre  $v$  può essere ricavato dalla frame che trasporta la challenge. Differentemente a prima, adesso è AP che seleziona  $v$ . L'avversario quindi deve costruirsi un dizionario  $DIZ = \{(v, \{z_i(v)\}) \mid v \in V_{24}\}$ , con  $V_{24}$  l'insieme di tutte le sequenze di 24 bit.  $DIZ$  ha  $2^{24}$  voci, ciascuna di  $\text{sizeof}(v) + \text{sizeof}(\{z_i(v)\})$  byte. La quantità  $v$  occupa 3 byte. Il keystream  $\{z_i(v)\}$  ha per definizione la stessa dimensione della challenge, cioè  $128/8 = 16$  byte. Ne segue quindi che una voce del dizionario occupa 19 byte. Ne segue perciò che il  $DIZ$  ha una dimensione pari a  $19 \times 2^{24} \cong 20$  Mbyte.

**LAUREA SPECIALISTICA IN INGEGNERIA INFORMATICA**  
**SICUREZZA NELLE RETI**  
**Appello del 17 Febbraio 2005**

Nome e Cognome \_\_\_\_\_ Matricola \_\_\_\_\_

**ESERCIZIO 1**

**PUNTI: 14 (3, 3, 2, 3, 3)**

Con riferimento al sistema di crittografia One-time Pad, il candidato, con precisione matematica e proprietà di linguaggio,

1. descriva gli algoritmi di generazione della chiave, di cifratura e di decifratura;

2. specifichi le condizioni sotto le quali il cifrario è perfetto;

3. illustri le implicazioni pratiche di tali condizioni.

Siano  $P = \text{"MARIOROSSI0850EURO"}$ ,  $K = \text{"5RTIOX1LQB39DEMZAN"}$ ,

4. si determini  $C$  sapendo

- che  $C[i]$ ,  $P[i]$  e  $K[i]$  sono la rappresentazione dell'i-esimo carattere di  $C$ ,  $P$  e  $K$ , rispettivamente;
- che il generico carattere è rappresentato dall'intero che esprime la sua posizione, a partire da zero, nell'alfabeto  $\mathcal{A} = \{\text{'A'}, \text{'B'}, \dots, \text{'Z'}, \text{'0'}, \text{'1'}, \dots, \text{'9'}\}$ ;
- che  $C[i] = P[i] \oplus K[i]$  per ogni carattere di  $P$ .

5. si determini la probabilità che un avversario, operando sul solo testo cifrato, riesca ad incrementare la cifra più significativa dell'importo sapendo che tale importo è minore di 1000 euro.

**SOLUZIONE.**

**Punti 1–3.** Vedere il materiale didattico.

**Punto 4.** Eseguire le somme modulo 31.<sup>1</sup>

**Punto 5.** L'avversario ha la certezza del successo del suo attacco. Sapendo che l'importo è minore di 1000 euro, l'avversario sa che la cifra più significativa è rappresentata dal carattere  $p = \text{'0'}$ . Conoscendo il carattere  $c$  del crittogramma corrispondente a  $p$  ('Q'), l'avversario è in grado di determinare il carattere  $k$  della chiave associato a  $p$  e  $c$  come  $p \oplus c$ . Tale carattere è '3'. A questo punto l'avversario può calcolare il carattere  $c'$  da sostituire nel crittogramma al posto di 'Q',  $c' = k \oplus \text{'1'} = \text{'R'}$ .

Più semplicemente si poteva osservare che  $c' = p \oplus 1 \oplus k = c \oplus 1$ .

**ESERCIZIO 2**

**PUNTI: 8 (4, 4)**

1. Bob riceve il messaggio  $\langle \text{Alice}, m, \Pi, \sigma \rangle$  e verifica con successo la firma digitale  $\sigma$  di  $m$  per mezzo della chiave pubblica  $\Pi$ . Indicare quale delle seguenti conclusioni Bob può correttamente trarre, motivando la scelta:

---

<sup>1</sup> Oppure modulo 36 se si considera l'alfabeto inglese. Noi consideriamo quello italiano.

**LAUREA SPECIALISTICA IN INGEGNERIA INFORMATICA**  
**SICUREZZA NELLE RETI**  
**Appello del 31 Gennaio 2006**

- a) Il messaggio  $m$  è stato firmato con la chiave privata di Alice;
  - b) Il messaggio  $m$  è stato firmato con la chiave privata corrispondente a  $\Pi$ ;
  - c) Il messaggio  $m$  è stato firmato da Alice.
2. Sia  $T$  un'autorità di certificazione di fiducia di Bob, di cui Bob conosce la chiave pubblica  $\Pi_T$ . Sia inoltre  $X(T, A)$  un certificato rilasciato da  $T$  ad *Alice*. Bob riceve il messaggio  $\langle A, m, \sigma, X(T, A) \rangle$  e verifica con successo le firme digitali. Quale delle precedenti conclusioni Bob può adesso correttamente trarre? Motivare la scelta.

**SOLUZIONE.**

**Quesito 1.** Bob può giungere alla conclusione b.

**Quesito 2.** Bob può giungere alla conclusione a.

**ESERCIZIO 3**

**PUNTI: 10 (3, 4, 3)**

Con riferimento a Kerberos 5, il candidato

1. illustri il protocollo semplificato;
2. lo analizzi con la logica BAN;
3. discuta l'impatto che il *ticket lifetime* e l'*authenticator lifetime* hanno sulla sicurezza del protocollo.

**SOLUZIONE.** Vedere il materiale didattico.

**LAUREA SPECIALISTICA IN INGEGNERIA INFORMATICA**  
**SICUREZZA NELLE RETI**  
**Appello del 31 Gennaio 2006**

**SOLUZIONE**

**ESERCIZIO 1**

**ESERCIZIO 2**

**ESERCIZIO 3**

# Sicurezza nelle Reti

## Appello del 7 Luglio 2006

Nome e Cognome \_\_\_\_\_ Matricola \_\_\_\_\_

**QUESITO 1**
**PUNTI: 12 (5, 1, 2, 4)**

Sia  $W$  un borsellino elettronico che memorizza informazioni relative ai clienti, come nome, indirizzo e numero di carta di credito, ed esegue pagamenti per loro conto. Un cliente  $C$  viene identificato da  $W$  per mezzo di un PIN, un numero segreto su  $p$  bit condiviso tra  $C$  ed  $W$ . La Sia  $PKW$  la chiave pubblica del server  $W$ , su  $k$  bit, che si suppone pubblicamente nota.

Sia  $D$  la descrizione (una stringa di caratteri) della merce che  $C$  vuole acquistare, sia nota a  $W$  oltre che a  $C$ . Un possibile protocollo di autorizzazione dell'acquisto è costituito dai seguenti messaggi:

M1  $W \rightarrow C: W, tid, D$

M2  $C \rightarrow W: C, E_w(tid, D, PIN)$

dove  $tid$  è l'identificatore unico di transazione generato da  $W$  ed  $E_X(m)$  denota la cifratura della quantità  $m$  con la chiave pubblica di  $X$ . Il candidato risponda alle seguenti domande motivando le risposte.

- Il protocollo garantisce la confidenzialità della quantità PIN rispetto ad ciphertext-only attack? Si determini il tempo medio necessario per tale attacco nel caso che  $k = 1024$ ,  $p = 4$  ed il tempo necessario ad eseguire una cifratura/decifratura è di circa 50ms.

**SOLUZIONE.** In M2 le quantità  $tid$  e  $D$  non sono segrete (lette in M1). Inoltre  $PKW$  è pubblica. Per cui si può fare una ricerca esaustiva sul PIN. Il tempo medio è  $(50 \times 10^4)/2 = 250000$  ms = 250 s = 6 minuti e 10 secondi

- Tale attacco può essere condotto off-line?

**SOLUZIONE.** Si.

- Tenendo conto della risposta al punto 2, il server  $W$  può attribuire il messaggio M2 al customer  $C$ ?

**SOLUZIONE.** NO.

- Nel caso la risposta alla domanda 3 sia negativa, apportare una modifica al protocollo in modo da garantire l'autenticità del customer senza però assegnare alcuna coppia di chiavi pubblica-privata al customer.

**SOLUZIONE.** Basta che il customer concatensi un numero random di sua scelta alle quantità  $tid$ ,  $D$ , PIN nel messaggio M2. Il numero random deve avere una lunghezza sufficiente da rendere praticamente impossibile un attacco esaustivo ai dati.

**QUESITO 2**
**PUNTI: 8 (5, 3)**

Con precisione matematica e proprietà di linguaggio, si definisca il *true random cipher* e ne discuta i problemi relativi ad una pratica realizzabilità.

**SOLUZIONE.** Vedere materiale didattico

**QUESITO 3**
**PUNTI: 12 (3, 3, 2, 4)**

- Si spieghi con precisione matematica e proprietà di linguaggio quando un cifrario viene detto *sicuro dal punto di vista computazionale*.

**SOLUZIONE.** Vedere materiale didattico.

2. Si definiscano i due attacchi: *known-plaintext attack* e *chosen-plaintext attack*.

**SOLUZIONE.** Vedere materiale didattico.

Si consideri il cifrario a blocchi FEAL-N caratterizzato dai seguenti parametri: block size  $n = 64$  bit , key size  $k = 64$  e numero di round  $N = 4, 6, 8, 16, 24, 32$ . La Tabella 1 specifica la resistenza di FEAL a vari attacchi.

3. Si vuole garantire la confidenzialità delle informazioni per almeno un giorno rispetto ad un avversario capace di un known-plaintext attack. Si indichi quali cifrari sono da considerarsi insicuri in tal caso.

**SOLUZIONE.** Si scarta FEAL-N per N uguale a 4, 6 ed 8. Gli altri cifrari possono essere utilizzati perché l'attacco più efficiente è di tipo chosen-plaintext attack ma l'avversario non è capace di questo tipo di attacco per ipotesi.

4. Si indichi quali cifrari sono da considerarsi insicuri nel caso che si voglia garantire la confidenzialità delle informazioni per almeno un mese rispetto ad un avversario che è capace di un chosen-plaintext attack *off-line*,<sup>1</sup> che dispone di hardware convenzionale (un PC ad esempio), e che riesce ad eseguire operazioni di cifratura e decifratura in 1 ms.

**SOLUZIONE.** Si scarta FEAL 16 perché richiede circa 8 Gbytes di memoria per memorizzare offline i chosen pairs ed impiega circa 106 secondi (12 gg) per condurre l'attacco.

**Tabella 1.** Attacchi a FEAL. LC sta per linear cryptanalysis e DC sta per differential cryptoanalysis.

attack method	data complexity		storage complexity	processing complexity
	known	chosen		
FEAL-4 – LC	5	—	30 Kbytes	6 minutes
FEAL-6 – LC	100	—	100 Kbytes	40 minutes
FEAL-8 – LC FEAL-8 – DC	$2^{24}$	$2^7$ pairs	280 Kbytes	10 minutes 2 minutes
FEAL-16 – DC	—	$2^{29}$ pairs		$2^{30}$ operations
FEAL-24 – DC	—	$2^{45}$ pairs		$2^{46}$ operations
FEAL-32 – DC	—	$2^{66}$ pairs		$2^{67}$ operations

---

<sup>1</sup> L'avversario accumula i chosen pair necessari e poi li utilizza come dati dell'algoritmo di crittoanalisi.

**Sicurezza nelle Reti**  
Appello del 14 Luglio 2006

# SOLUZIONE

**QUESITO 1****PUNTI: 8**

In un cifrario  $C$  esistono un messaggio  $m$  ed un crittogramma  $c$  tali che  $\text{Prob}(M = m) = p$ , con  $p < 1/4$ , e  $\text{Prob}(M=m|C=c) = 1-p$ . Spiegare se  $C$  può essere un cifrario perfetto e le conseguenze per un crittoanalista per la coppia  $(m, c)$  indicata.

Affinché il cifrario sia perfetto bisogna che  $p = 1 - p$ . Questa uguaglianza non è soddisfatta per  $p < 1/4$  e perciò il cifrario non è perfetto. Questo implica che se l'avversario vede passare in rete il crittogramma  $c$  allora può concludere che è stato trasmesso il messaggio  $m$  con una probabilità  $1 - p$ . Siccome questa probabilità è maggiore di  $p$ , la probabilità con cui viene trasmesso  $m$ , allora l'avversario ha guadagnato dall'osservazione del testo cifrato maggiori informazioni su  $m$ .

**QUESITO 2****PUNTI: 10 (8, 2)**

A e B utilizzano un crittosistema simmetrico  $E()$  ed una funzione hash con chiave  $h()$ . A e B utilizzano inoltre le chiavi segrete condivise  $K$  e  $K'$  con il crittosistema e la funzione hash, rispettivamente. Infine A e B utilizzano un meccanismo contatore-finestra (come quello di IPsec) di supporto ad un servizio anti-replay. Supponendo che il formato dei messaggi sia  $m = (c, x, d)$ , con  $x$  valore del contatore del mittente, specificare quale, o quali, dei seguenti casi fornisce il servizio anti-replay:

1.  $c = E(K, m); d = h(K', x)$
2.  $c = E(K, m \parallel x); d = h(K', x)$
3.  $c = E(K, m); d = h(K', c \parallel x)$

In caso di soluzioni equivalenti, indicare quella che permette di realizzare il servizio anti-replay con il minore carico computazionale dal punto di vista del ricevitore.

La prima soluzione non garantisce l'anti-replay perché  $m$  ed  $x$  non sono “indissolubilmente” (crittograficamente) legati tra di loro. La seconda e la terza soluzione garantiscono anti-replay con una differenza dal punto di vista dell'overhead computazionale. Nella seconda, la verifica dell'anti-replay richiede la decifratura del messaggio. Perciò saranno inutilmente decifrati anche i messaggi destinati ad essere scartati. Nella terza, la verifica viene fatta sul testo cifrato. Quindi la decifratura verrà eseguita solo dei messaggi che sono accettati.

**QUESITO 3****PUNTI: 12 (4, 4, 4)**

Con riferimento al sistema RSA, il candidato, con precisione matematica e proprietà di linguaggio,

1. descriva gli algoritmi di generazione della chiave, di cifratura e di decifratura;
2. ne discuta la “sicurezza” (relazione tra RSA ed un problema complesso); ed infine
3. illustri perché il riutilizzo del modulo può portare ad un *common modulus attack*.

Vedere materiale didattico.

# SICUREZZA NELLE RETI

Appello del 24 luglio 2009

## Esercizio 1

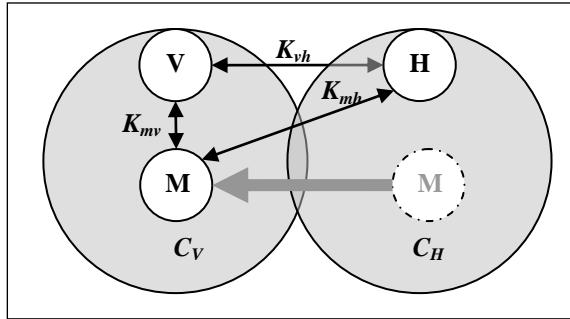
punti: 10

Con proprietà di linguaggio e precisione matematica si discuta la sicurezza dell'algoritmo RSA.

## Esercizio 2

punti: 12

Si consideri il sistema semplificato di *roaming* riportato in figura in cui una stazione mobile  $M$  registrata presso l'*home server*  $H$  si trova nella cella  $C_V$  gestita dal server *visited server*  $V$ . Per garantire la comunicazione sicura della stazione mobile  $M$  è necessario che questa stabilisca un canale sicuro (la chiave simmetrica  $K_{vm}$ ) con il visited server  $V$ . A tale proposito si assuma che:



- in virtù della registrazione, la stazione mobile  $M$  ed l'*home server*  $H$  condividono la chiave  $k_{mh}$ ;
- i server  $H$  e  $V$  fanno parte dell'infrastruttura e pertanto condividono la chiave  $k_{vh}$ ;
- la stazione mobile  $M$  ed i server  $H$ ,  $V$  utilizzano lo stesso cifrario; ed infine
- i loro clock non sono sincronizzati.

Progettare un protocollo di distribuzione delle chiavi che soddisfi i seguenti requisiti:

- permette l'autenticazione mutua di  $M$  e  $V$  quando  $M$  entra nella cella  $C_V$ ;
- permette di distribuire una chiave di sessione  $k_{mv}$  tra  $M$  e  $V$ ;
- fornisce la prova a  $V$  che  $M$  possiede  $k_{mv}$  e viceversa;
- mantiene la segretezza della chiave  $k_{mh}$ ;
- è resistente ad attacchi di replay.

Il candidato argomenti per mezzo della logica BAN che il protocollo proposto soddisfa i requisiti posti.

## Esercizio 3

punti: 8

Si descrivano le informazioni minime che devono essere contenute in un certificato digitale ed i principali obblighi che un'autorità di certificazione deve assolvere per il rilascio di tale certificato.

# Soluzione

**Risposta a.** Il protocollo garantisce la confidenzialità. Per determinare il valore di  $K_{AB}$  è necessario conoscere le quantità  $k_a$  e  $k_b$ . Tuttavia queste quantità viaggiano in rete in forma cifrata.

**Risposta b.** Il protocollo è una versione semplificata del protocollo Needham-Schroeder a chiave pubblica. Il protocollo garantisce la *key authentication*. Formalmente:  $A \stackrel{k_b}{\equiv} B \stackrel{k_b}{\equiv} A \xrightarrow{k_a} B \text{ e } B \stackrel{k_a}{\equiv} A \stackrel{k_a}{\equiv} A \xrightarrow{k_b} B$ .

**Risposta c.** Come si evince dalle formule sopra, il protocollo garantisce anche la key confirmation.

**Risposta d.** Se l'ipotesi (i) non è verificata, il protocollo non garantisce la proprietà di key authentication. Supponiamo che la quantità  $k_a$  sia riutilizzata da  $A$  e che un avversario  $M$  abbia registrato i messaggi M1 ed M2 relativi all'esecuzione del protocollo in cui  $k_a$  è stata utilizzata la prima volta. L'avversario  $M$  potrebbe eseguire il seguente attacco:

- induce  $A$  ad iniziare una nuova istanza del protocollo con  $B$ ;
- quando  $A$  invia il messaggio M1' relativo alla nuova esecuzione del protocollo, contenente la quantità riutilizzata  $k_a$ , l'avversario  $M$  determina che M1=M1', e risponde con M2. Alla ricezione di questo messaggio il processo A crede di parlare effettivamente con  $B$ .

Si noti che questo attacco ha come effetto collaterale il riutilizzo della vecchia chiave di sessione  $K_{ab}$  (la quantità  $k_b$  è contenuta nel messaggio M2 replicato da  $M$ ). L'avversario  $M$  potrebbe non conoscere tale chiave ma potrebbe comunque replicare vecchi messaggi relativi alla sessione  $K_{ab}$  che  $A$  considererebbe come provenienti da  $B$ . Il danno sarebbe massimo se  $M$ , per altre vie, fosse riuscito ad impadronirsi di  $K_{ab}$ .

Considerazioni simili possono essere fatte per  $B$  se questo processo riutilizza la quantità  $k_b$ .

# SICUREZZA NELLE RETI

Appello del 14 Gennaio 2010

## Esercizio 1

**punti: 10**

Con proprietà di linguaggio e precisione matematica il candidato (a) definisca il cifrario perfetto secondo Shannon e (b) dimostri che in tale cifrario il numero delle chiavi non può essere inferiore al numero dei messaggi.

## Esercizio 2

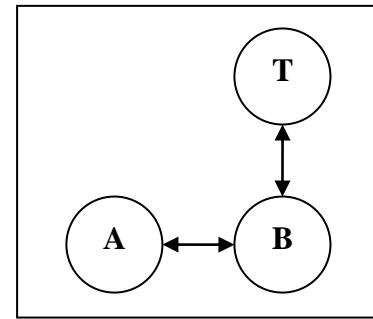
**punti: 12**

Con riferimento al sistema di comunicazione in figura, usando la logica BAN, definire un protocollo di distribuzione delle chiavi che soddisfi i seguenti requisiti:

1. A e B sono convinti che  $K_{ab}$  è la chiave di sessione;
2. A è convinto che B disponga di  $K_{ab}$  e viceversa;
3. il protocollo non è soggetto a *replay attack*;

sotto i seguenti vincoli

- A. A e B condividono una chiave segreta, rispettivamente con  $K_a$  e  $K_b$ , con T;
- B. A e B considerano T competente nella generazione delle chiavi;
- C. i clock non sono sincronizzati.



## Esercizio 3

**punti: 8**

Il candidato (a) descriva il problema del Denial of Service nel protocollo Diffie-Hellman e (b) discuta la soluzione proposta nel protocollo di Oakley.

## Soluzione

### Esercizio 2

M 1  $A \rightarrow B : A, B, n_a$

M 2  $B \rightarrow T : A, B, n_a, n_b$

M 3  $T \rightarrow B : A, B, n_b, k_{ab}, A, B, n_a, k_{ab} \quad k_a \quad k_b$

M 4  $B \rightarrow A : A, B, n_a, k_{ab} \quad k_a, A, B \quad k_{ab}$

M 5  $A \rightarrow B : B, A \quad k_{ab}$

# SICUREZZA NELLE RETI

APPELLO DEL 01 FEBBRAIO 2010

## Esercizio 1

**punti: 10**

La modalità CBC: schema, propagazione dell'errore e vantaggi rispetto alla modalità ECB.

## Esercizio 2

**punti: 12**

I processi  $A$  e  $B$  dispongono solo di un cifrario a chiave pubblica e di un cifrario a chiave simmetrica. Si assuma che  $A$  conosca la chiave pubblica  $\Pi_b$  di  $B$  e viceversa. Progettare per mezzo della logica BAN un protocollo di distribuzione delle chiavi che soddisfi i seguenti requisiti

1.  $A$  ha la prova che  $B$  dispone della chiave di sessione  $K_{ab}$  e viceversa;
2. i clock non sono sincronizzati;
3. il protocollo è resistente ad attacchi di replay;
4. la chiave di sessione è generata da uno dei due processi.

Modificare il protocollo assumendo che entrambi i processi contribuiscano alla generazione della chiave di sessione.

## Esercizio 3

**punti: 8**

Il problema della delega in Kerberos e le relative soluzioni.

## Soluzione

### Esercizio 2

$$M1 \quad A \xrightarrow{K_a} B : \quad K_a \in \Pi_b$$

$$M2 \quad B \xrightarrow{K_a} A : \quad K_a, K_{ab} \in \Pi_a$$

$$M3 \quad A \xrightarrow{K_b} B : \quad A, B \in \Pi_b$$

$$M4 \quad B \xrightarrow{K_{ab}} A : \quad K_{ab} \in \Pi_a$$

$$M5 \quad A \xrightarrow{K_{ab}} B : \quad B, A \in \Pi_b$$

**Laurea Specialistica in Ingegneria Informatica**  
**SICUREZZA NELLE RETI**  
**Appello del 19 febbraio 2010**

Nome e Cognome \_\_\_\_\_ Matricola \_\_\_\_\_

**ESERCIZIO 1**

**Punti:6**

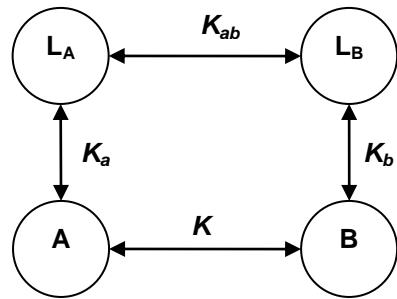
Con proprietà di linguaggio e precisione matematica, il candidato

- a) enunci le proprietà di *preimage resistance*, *2nd-preimage resistance* e *collision resistance* di una funzione hash sicura;
- b) ne discuta la rilevanza rispetto alla firma digitale; ed
- c) indichi il limite superiore della sicurezza (in termini di complessità) di una funzione hash con  $m$  bit di output.

**ESERCIZIO 2**

**punti: 14**

Due soggetti  $A$  e  $B$  stabiliscono una chiave di sessione  $K$  utilizzando il protocollo Diffie-Hellmann con parametri pubblici  $p$  e  $g$ . Al fine di evitare il MIM-attack, essi usano un sistema di certificazione online riportato in figura. Ogni soggetto fa riferimento ad una autorità di certificazione locale (ad esempio  $L_A$ ) ed ha con essa una relazione di fiducia che si concretizza nella condivisione di una chiave segreta (ad esempio  $K_a$ ). L'autorità di certificazione ne certifica il parametro pubblico (Ad esempio  $L_A$  certifica il parametro pubblico  $X_A$  di  $A$ ). Tra le autorità di certificazione locali esiste una cross-certificazione che si concretizza nella condivisione di una chiave segreta (ad esempio  $K_{ab}$ ).



Usando la logica BAN, il candidato progetti e verifichi il protocollo completo di autenticazione e definizione di una chiave di sessione tra i soggetti  $A$  e  $B$  sia nel caso in cui i clock non sono sincronizzati sia in quello in cui lo sono. Nel secondo caso, si assuma che il certificato abbia una validità di  $D$  unità di tempo.

**ESERCIZIO 3**

**punti:8**

Con proprietà di linguaggio il candidato descriva il meccanismo a finestra anti-replay di IpSec.

*Laurea Specialistica in Ingegneria Informatica*  
**SICUREZZA NELLE RETI**  
**Appello del 19 febbraio 2009**

Soluzione

**ESERCIZIO 1**

Vedi appunti.

**ESERCIZIO 2**

**Senza clock sincronizzati**

- M1. A -> B: Na
- M2. B->LB: {Na, XB}Kb
- M3. LB->LA: {Na, XB}Kab
- M4. LA->A: {Na, XB}Ka

Parallelamente a questi msg ci sono quelli M1'-M4' per la certificazione di A rispetto a B. Il protocollo si ottiene sostituendo A con B.

**Con i clock sincronizzati**

- M1. A->LA: {tA, XA}Ka
- M2. LA->LB: {tA, XA}Ka
- M3. LB->B: {tA, XA}Kb

Parallelamente a questi msg ci sono quelli M1'-M3' per la certificazione di A rispetto a B. Il protocollo si ottiene sostituendo A con B.

**ESERCIZIO 3**

Vedi appunti.

Nome e Cognome \_\_\_\_\_ Matricola \_\_\_\_\_

**ESERCIZIO 1**

**Punti:16**

Con proprietà di linguaggio e precisione matematica, il candidato i) specifichi il protocollo di Diffie-Hellman; e ii) ne argomenti la sicurezza rispetto ad un avversario passivo. Inoltre, iii) estenda il protocollo DH al caso di  $n$  processi(Group-DH), iv) valutandone la complessità in termini di numero e dimensione dei messaggi trasmessi.

**ESERCIZIO 2**

**punti: 6**

Alice e Bob utilizzano il protocollo Diffie-Hellman (DH) per stabilire una chiave di sessione. Al fine di evitare l'attacco dell'uomo-nel-mezzo, Alice e Bob mantengono un segreto condiviso a-priori  $\sigma$ . Progettare e verificare con la logica BAN un protocollo il protocollo DH-modificato che utilizza tale segreto. La verifica può dirsi conclusa con successo quando si raggiungono i beliefs  $B \stackrel{v}{\equiv} \mapsto$  e  $A \stackrel{v}{\equiv} \mapsto$ , con  $X_A$  ed  $X_B$  parametri pubblici di Alice e Bob, rispettivamente.

**ESERCIZIO 3**

**punti:8**

Con proprietà di linguaggio e precisione matematica, il candidato spieghi il problema del *keystream reuse* in WEP.

# Soluzione

## **ESERCIZIO 1**

Vedi appunti.

## **ESERCIZIO 2**

$$M1 \quad A \rightarrow \qquad \qquad n_A$$

$$M2 \quad B \rightarrow \qquad \qquad n_B$$

$$M3 \quad A \rightarrow \qquad \qquad {}_A, h_\sigma' \xrightarrow{v} \parallel$$

$$M4 \quad B \rightarrow \qquad \qquad {}_B, h_\sigma' \xrightarrow{v} \parallel$$

## **IPOTESI**

$$1. \quad A \models \xrightarrow{v}, B \models \xrightarrow{v}$$

$$2. \quad B \models \xleftarrow{-}, A \models \xrightarrow{-}$$

$$3. \quad A \models \xrightarrow{-}, B \models \xrightarrow{-}$$

$$4. \quad A \models \xrightarrow{v}, B \models \xrightarrow{v}$$

## **PROTOCOLLO IDEALIZZATO**

$$M3 \quad A \rightarrow \qquad \qquad \xrightarrow{\backslash v} \qquad \qquad \qquad \begin{array}{c} / \\ \downarrow \\ \end{array} \qquad \qquad \qquad \begin{array}{c} \backslash \\ \nearrow \\ \end{array} \qquad \qquad \qquad \begin{array}{c} / \\ \sigma \\ \end{array}$$

$$M4 \quad B \rightarrow \qquad \qquad \xrightarrow{\backslash v} \qquad \qquad \qquad \begin{array}{c} / \\ \downarrow \\ \end{array} \qquad \qquad \qquad \begin{array}{c} \backslash \\ \nearrow \\ \end{array} \qquad \qquad \qquad \begin{array}{c} / \\ \sigma \\ \end{array}$$

## **TESI**

$$1. \quad B \models \xrightarrow{v}, A \models \xrightarrow{v}$$

## **ESERCIZIO 3**

Vedi appunti.

Nome e Cognome \_\_\_\_\_ Matricola \_\_\_\_\_

**ESERCIZIO 1**

**punti: 10**

Il candidato presenti le proprietà di una funzione hash sicura (MDC) illustrandone la relazione con la firma digitale.

**ESERCIZIO 2**

**punti: 10**

Si progetti un protocollo di distribuzione delle chiavi che permette ad Alice e Bob di stabilire una chiave di sessione  $K_{ab}$  attraverso Trent, un Key Distribution Center fidato. Alice e Bob non condividono *a-priori* alcun segreto a lungo termine. Al contrario, entrambi conoscono  $\Pi_T$  la chiave pubblica di Trent. Inoltre ciascuno di essi condivide con Trent un segreto che però non può essere utilizzato come chiave crittografica. Sia, ad esempio,  $\pi_A$  il segreto che Alice condivide con Trent. Si assuma che i clock siano sincronizzati.

Al termine di un'esecuzione, il protocollo deve garantire che i) Alice (Bob) conosca la chiave di sessione; ii) Alice (Bob) sappia che Bob (Alice) conosce la chiave di sessione; iii) attacchi di reply non siano possibili.

Il candidato è libero di scegliere lo schema di comunicazione.

**ESERCIZIO 3**

**punti: 10**

Il candidato descriva il protocollo base di Kerberos discutendo il dimensionamento delle finestre temporali caratteristiche del protocollo.

# Soluzione

## ESERCIZIO 1

Vedi appunti.

## ESERCIZIO 2

### Ipotesi

$$1. \quad \forall P \in \{A, B\}, P \stackrel{\Pi_T}{\equiv} \mapsto T$$

$$2. \quad \forall P \in \{A, B\}, P \stackrel{\pi_p}{\equiv} P \rightleftharpoons T, T \stackrel{\pi_p}{\equiv} P \rightleftharpoons T$$

$$3. \quad \forall P \in \{A, B\}, P \stackrel{K_p}{\equiv} P \leftrightarrow T$$

$$4. \quad \forall P \in \{A, B\}, T \stackrel{\Pi_T}{\equiv} \left( A \xrightarrow{K_{ab}} B \right)$$

$$5. \quad \forall P \in \{A, B\}, P \stackrel{\Pi_T}{\equiv} T \Rightarrow \left( A \xrightarrow{K_{ab}} B \right)$$

$$6. \quad \forall P \in \{A, B\}, T \stackrel{\Pi_T}{\equiv} P \Rightarrow \left( P \xrightarrow{K_p} T \right)$$

$$7. \quad \forall P, Q \in \{A, B, T\}, P \stackrel{\#}{\equiv} \#(t_q)$$

### Protocollo idealizzato

$$M 1 \quad A \rightarrow B : \quad \left\{ \left\langle A, B, \tau_a, A \leftrightarrow T \right\rangle_{\pi_a}^{K_a} \right\}_{\Pi_T}$$

$$M 2 \quad B \rightarrow T : \quad \left\{ \left\langle A, B, \tau_a, A \leftrightarrow T \right\rangle_{\pi_a}^{K_a} \right\}_{\Pi_T}, \left\{ \left\langle B, A, \tau_b, B \leftrightarrow T \right\rangle_{\pi_b}^{K_b} \right\}_{\Pi_T}$$

$$M 3 \quad T \rightarrow B : \quad \left\{ B, A, \tau_b + 1, A \leftrightarrow B \right\}_{K_b}^{K_{ab}}, \left\{ A, B, \tau_a + 1, A \leftrightarrow B \right\}_{K_a}^{K_{ab}}$$

$$M 4 \quad B \rightarrow A : \quad \left\{ A, B, \tau_a + 1, A \leftrightarrow B \right\}_{K_a}^{K_{ab}}, \left\{ A, B, \tau'_b, A \leftrightarrow B \right\}_{K_{ab}}^{K_{ab}}$$

$$M 5 \quad A \rightarrow B : \quad \left\{ B, A, \tau'_b + 1, A \leftrightarrow B \right\}_{K_{ab}}^{K_{ab}}$$

### **Protocollo reale**

M 1     $A \rightarrow B : \{ A, B, \{ A, B, \tau_a, K_a, \pi_a \}_{\Pi_T} \}$

M 2     $B \rightarrow T : \{ A, B, \tau_a, K_a, \pi_a \}_{\Pi_T}, \{ B, A, \tau_b, K_b, \pi_b \}_{\Pi_T}$

M 3     $T \rightarrow B : \{ B, A, \tau_b, K_{ab} \}_{K_b}, \{ A, B, \tau_a, K_{ab} \}_{K_a}$

M 4     $B \rightarrow A : \{ A, B, \tau_a, K_{ab} \}_{K_a}, \{ A, B, \tau'_b \}_{K_{ab}}$

M 5     $A \rightarrow B : \{ B, A, \tau'_b \}_{K_{ab}}$

### **ESERCIZIO 3**

Vedi appunti.

**Appello del 13 settembre 2011**

**NOME E COGNOME** \_\_\_\_\_ **MATRICOLA** \_\_\_\_\_

**ESERCIZIO 1**

**Punti:10**

Con proprietà di linguaggio e precisione matematica, il candidato i) descriva il protocollo *one-time pad* (OTP); ii) elenchi le ipotesi sotto le quali OTP è un cifrario ideale e ne discuta le implicazioni pratiche.

**ESERCIZIO 2**

**punti: 10**

Alice e Bob utilizzano il protocollo Diffie-Hellman (DH) per stabilire una chiave di sessione. Al fine di evitare l'attacco dell'uomo-nel-mezzo, Alice e Bob mantengono un segreto condiviso a-priori con un Key Distribution Center (KDC). Siano  $\sigma_A$  e  $\sigma_B$  i segreti di Alice e Bob rispettivamente. Supponendo che i clock siano sincronizzati, progettare e verificare con la logica BAN un protocollo DH-modificato che permette ad Alice e Bob di stabilire una chiave di sessione utilizzando il KDC. La verifica può dirsi conclusa con successo quando si raggiungono i beliefs  $B \stackrel{X_A}{\equiv} \mapsto A$  e  $A \stackrel{X_B}{\equiv} \mapsto B$ , con  $X_A$  ed  $X_B$  parametri pubblici del protocollo DH di Alice e Bob, rispettivamente.

**ESERCIZIO 3**

**punti:10**

Con proprietà di linguaggio e precisione matematica, il candidato spieghi la vulnerabilità del protocollo di autenticazione di WEP.

Appello del 13 settembre 2011

NOME E COGNOME \_\_\_\_\_ MATRICOLA \_\_\_\_\_

# Soluzione

## **ESERCIZIO 1**

Vedi appunti.

## **ESERCIZIO 2**

Per ipotesi le quantità  $\sigma_A$  e  $\sigma_B$  sono dei segreti condivisi e, come precisato in sede d'esame, in assenza di ipotesi ulteriori, non possono essere considerati chiavi di cifratura. Ad esempio, con riferimento all'algoritmo DES, un segreto potrebbe essere una *weak-key*. Quello che si può invece ragionevolmente supporre è che il segreto condiviso abbia una dimensione in bit adeguata a contrastare una ricerca esaustiva sebbene questo possa venire a detimento dell'usabilità. Sulla base di queste considerazioni si può definire il seguente protocollo reale.

### PROTOCOLLO REALE

**M**  $A \xrightarrow{KC} X_A t_A h X_A t_A \sigma_A$   
**M**  $B \xrightarrow{KC} X_B t_B h X_B t_B \sigma_B$   
**M**  $KC \xrightarrow{B} X_A t_K h X_A t_K \sigma_B$   
**M**  $KC \xrightarrow{A} X_B t_K h X_B t_K \sigma_A$

### IPOTESI

1.  $A \stackrel{x_A}{\equiv} \mapsto A, B \stackrel{x_B}{\equiv} \mapsto B$
2.  $AKC \stackrel{\sigma_A}{\equiv} A, BKC \stackrel{\sigma_B}{\equiv} B$
3.  $AKC \equiv \#(t_A), BKC \equiv \#(t_B), ABKC \equiv \#(t_K)$
4.  $A \stackrel{x_B}{\equiv} B \Rightarrow \mapsto B, B \stackrel{x_A}{\equiv} A \Rightarrow \mapsto A$
5.  $A \equiv KC \stackrel{x_B}{\Rightarrow} B \equiv KC \stackrel{x_A}{\Rightarrow} A$

**SICUREZZA NELLE RETI**

**Appello del 12 settembre 2009**

**PROTOCOLLO IDEALIZZATO**

$$\begin{array}{ll} M1 \quad A \rightarrow KDC : & \left\langle \overset{x_A}{\mapsto} A, t_A \right\rangle_{\sigma_A} \\ M2 \quad B \rightarrow KDC & \left\langle \overset{x_B}{\mapsto} B, t_B \right\rangle_{\sigma_B} \\ M3 \quad KDC \rightarrow B & \left\langle A \overset{\equiv}{\mid} \overset{x_A}{\mapsto} A, t_K \right\rangle_{\sigma_B} \\ M4 \quad KDC \rightarrow A : & \left\langle B \overset{\equiv}{\mid} \overset{x_B}{\mapsto} B, t_K \right\rangle_{\sigma_A} \end{array}$$

**DIMOSTRAZIONE (SKETCH)**

Alla ricezione del messaggio M1, per le ipotesi 2 e 3,  $KDC \models A \overset{x_A}{\mapsto} A$ . Alla ricezione del messaggio 3, per le ipotesi 2, 3, 4, 5,  $B \overset{x_A}{\models} \mapsto A$ . Un ragionamento analogo può essere fatto per i messaggi M2 e M4. Si ottiene quindi  $B \overset{x_A}{\models} \mapsto A, A \overset{x_B}{\models} \mapsto B$

**ESERCIZIO 3**

Vedi appunti.