CORSO DI SICUREZZA INFORMATICA 1 (A.A. 2007/2008)

Prof. A. Armando

(3 Luglio 2008)

Si risponda alle domande utilizzando lo spazio apposito.
Non è consentito l'utilizzo di libri, appunti, nè dispositivi elettronici di alcun tipo.

Nome e Cognome: _____

Matricola: _____

---

1. ***Crittografia simmetrica***

   Si consideri l'algoritmo di cifratura a blocchi definito da:

$$\begin{aligned} C_0 &= IV \\ C_i &= E_K(C_{i-1}) \oplus P_i \end{aligned} \tag{1}$$

   dove IV e' il vettore di inizializzazione. Si definisca l'operazione di decifrazione.

   Soluzione.

$$P_i = E_K(C_{i-1}) \oplus C_i$$

   Infatti, se moltiplichiamo ambo il lati (1) di per $E_K(C_{i-1})$ otteniamo:

$$E_K(C_{i-1}) \oplus C_i = E_K(C_{i-1}) \oplus (E_K(C_{i-1}) \oplus P_i)$$

   Sfruttando le proprietà dello $\oplus$ otteniamo:

$$E_K(C_{i-1}) \oplus C_i = P_i$$

2. ***Funzioni di Hash***

Si consideri la seguente funzione di hash. I messaggi sono visti come sequenza di numeri $M = a_1 a_2 \cdots a_l$*. La funzione di hash è calcolata nel seguente modo:*

$$H(M) = \left(\sum_{i=1}^{l} a_i \right) \mod n$$

per un dato valore predefinito di n*. Indicare quali delle seguenti proprietà sono soddisfatte dalla funzione* H*. Giustificare le risposte date.*

Soluzione.

☑ H can be applied to a block of data of any size.

☑ H produces a fixed-length output.
se si considera la rappresentazione binaria dei numeri.

☑ $H(x)$ is relatively easy to compute for any given x.
Il calcolo di H ha complessità lineare nella lunghezza di M.

☐ One-way property
Se k appartiene all'intervallo $[0..n-1]$ allora il messaggio M contentene comue unico numero k è tale che $H(M) = k$.

☐ Weak collision resistance
Se M' è una permutazione di M, allora $H(M') = H(M)$, ma chiaramente $M' \neq M$.

☐ Strong collision resistance
Vedi risposta alla domanda precedente.

3. ***Protocolli di Sicurezza***

Si consideri il Needham-Schroeder Secret-Key (NSSK) protocol *presentato a lezione:*

1. $A \to S : A, B, N_a$
2. $S \to A : \{N_a, B, K_{AB}, \{K_{AB}, A\}_{K_{BS}}\}_{K_{AS}}$
3. $A \to B : \{K_{AB}, A\}_{K_{BS}}$
4. $B \to A : \{N_b\}_{K_{AB}}$
5. $A \to B : \{N_b - 1\}_{K_{AB}}$

dove N_a *e* N_b *sono nonces,* K_{AS} *è la* long term key *tra* A *e* S *e* K_{BS} *è la* long term key *tra* B *e* S *e* K_{AB} *è una nuova chiave di sessione. Si assuma che il* key server S *sia fidato. L'obiettivo del protocollo è di generare una nuova chiave di sessione* K_{AB} *e far sì che sia condivisa tra* A *e* B*. Siccome* K_{AB} *può essere utilizzata per cifrare grosse moli di dati, al fine di contrastare possibili tentativi di ricostruirla tramite crittoanalisi,* K_{AB} *viene usata per un certo lasso di tempo scaduto il quale il protocollo viene rieseguito per generare una nuova chiave di sessione.*

(a) *Si consideri la versione del protocollo NSSK ottenuta eliminando la nonce* N_A *dai primi due messaggi, ovvero:*

1. $A \to S : A, B$
2. $S \to A : \{B, K_{AB}, \{K_{AB}, A\}_{K_{BS}}\}_{K_{AS}}$
3. $A \to B : \{K_{AB}, A\}_{K_{BS}}$
4. $B \to A : \{N_b\}_{K_{AB}}$
5. $A \to B : \{N_b - 1\}_{K_{AB}}$

e si discutano le possibili conseguenze negative di questa semplificazione.

Soluzione. A non è più in grado di stabilire la freshness della chiave K_{AB} ricevuta da S al passo 2. Un intruso I può dunque intercettare il messaggio mandato al passo 2 da S ad A e mandare al suo posto una vecchia versione dello stesso contenente una chiave di sessione che I è riuscito nel frattempo a ricostruire tramite crittoanalisi.

(b) *Si dimostri che anche il protocollo orginale è vulnerabile ad un attacco simile a quello discusso nella domanda (a) e si discutano possibile modifiche al protocollo per prevenire*

tale attacco.

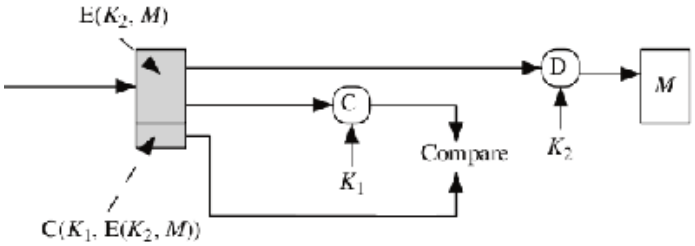
Soluzione. B non ha modo di verificare la freschezza del messaggio che riceve al passo 3 e quindi è vulnerabile ad un attacco in cui un intruso gli invia un messaggio dello stesso tipo precedente inviato e di cui è riuscito a ricostruire la chiave di sessione in esso contenuta.
Una possibile soluzione è quella adottata in Kerberos, ovvero aggiungere una *timestamp* al messaggio scambiato al passo 3. Il protocollo risultate è:

1. $A \to S : A, B, N_a$
2. $S \to A : \{N_a, B, K_{AB}, \{T, K_{AB}, A\}_{K_{BS}}\}_{K_{AS}}$
3. $A \to B : \{T, K_{AB}, A\}_{K_{BS}}$
4. $B \to A : \{N_b\}_{K_{AB}}$
5. $A \to B : \{N_b - 1\}_{K_{AB}}$

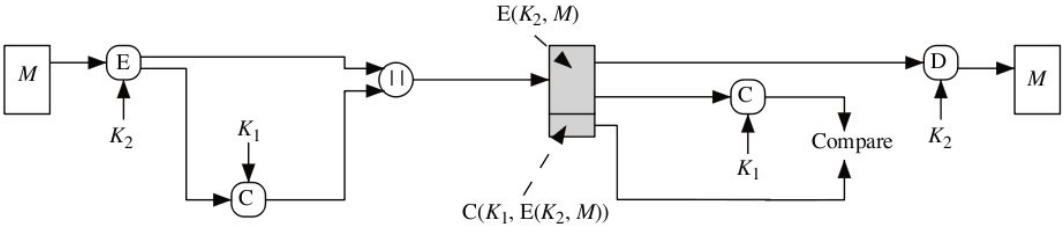
Ovviamente ciò comporta che B abbia un clock sincronizzato con S e quando riceve il messaggio al passo 3 deve verificare che T non sia troppo vecchia.

4. ***Crittografia II***

Si completi il seguente schema crittografico disegnandone il trasmettitore.



Soluzione.



5. ***Controllo degli Accessi*** *Si consideri un sistema con tre utenti: Alice, Bob e Charlie. Alice possiede il file* `alice.bat` *che può essere letto sia da Bob che da Charlie. Charlie può leggere e scrivere il file* `bob.bat`*, che è posseduto da Bob, ma Alice lo può solo leggere. Solo Charlie può leggere il file* `charlie.bat` *che gli appartiene. Ogni file può essere eseguito dagli utenti che lo posseggono.*

   (a) *Si scriva la matrice di controllo degli accessi corrispondente a tale situazione.*

Soluzione.

	`alice.bat`	`bob.bat`	`charlie.bat`
Alice	ox	r	
Bob	r	ox	
Charlie	r	rw	orx

   (b) *Si scriva la matrice di controllo degli accessi che si ottiene se Charlie dà ad Alice il permesso di leggere* `charlie.bat` *e Alice revoca a Bob il permesso di leggere* `alice.bat`*.*

Soluzione.

	`alice.bat`	`bob.bat`	`charlie.bat`
Alice	ox	r	r
Bob		ox	
Charlie	r	rw	orx