Network Security

Elements of Applied Cryptography

Analisi e progetto di protocolli crittografici

- **La logica BAN**
- **Principi di progettazione**
- **Casi di studio: Needham-Schroeder, Otway-Rees; SSL (old version); X509; GSM**

Il problema

Security protocols are three-line programs that people still manage to get wrong.

Roger M. Needham

La Logica BAN

- La logica BAN prende il nome dai suoi inventori: Burrows, Abadi, Needham
- La logica BAN è una logica di "belief and action"
- La logica non può essere utilizzata per provare che un protocollo è errato, tuttavia
- quando non è possibile provare che un protocollo è corretto, quel protocollo deve essere trattato con grande sospetto

Formalismo

$P \mid\equiv X$ **P believes X**: P si comporta come se X fosse vero

$P \triangleleft X$ **P sees X**: P ha ricevuto un messaggio che contiene X, nel passato o in questa esecuzione del protocollo; P può leggere X e ripeterlo

$P \mid\sim X$ **P once said X**: P ha inviato un messaggio che contiene X; P *believed* X quando lo inviò

$P \Rightarrow X$ **P controls X**. P è un autorità su X e bisogna fidarsi a questo riguardo

$\#(X)$ **X è fresh**

$P \overset{K}{\leftrightarrow} Q$ **K è una chiave condivisa tra P e Q**

$P \overset{K}{\rightleftharpoons} Q$ **X è un segreto condiviso tra P e Q**

$\overset{K}{\mapsto} P$ **K è la chiave pubblica di P**

$\langle X \rangle_Y$ **X è combinato con Y (segreto)**

$\{X\}_K$ **X è stato cifrato con K**

Formalismo: esempi

$A \mid\equiv \#(N_a)$ A crede che la quantità N_a sia fresh

$A \mid\equiv A \overset{K}{\leftrightarrow} B$ A crede che K sia una chiave condivisa con B

$T \mid\equiv A \overset{K}{\leftrightarrow} B$ T crede che K sia una chiave per la comunicazione tra A e B

$A \mid\equiv T \Rightarrow A \overset{K}{\leftrightarrow} B$ A crede che T sia competente nella generazione di chiavi di sessione

$A \mid\equiv T \Rightarrow \#\left(A \overset{K}{\leftrightarrow} B \right)$ A crede che T sia competente nella generazione di chiavi di sessione "fresh"

Preliminari

- Nello studio dei protocolli di autenticazione si considerano due epoche: il **presente** ed il **passato**
  - Il presente comincia con l'inizio dell'esecuzione di un protocollo
- I belief raggiunti nel presente sono **stabili** per tutta la durata del protocollo
- Si assume che quando **P dice X allora P crede X**
- I belief del passato non è detto che valgano nel presente

Postulati: message meaning rule

$$\frac{P \mid\equiv Q \overset{K}{\leftrightarrow} P, P \triangleleft \{X\}_K}{P \mid\equiv Q \mid\sim X}$$

Se *K* è una chiave condivisa tra *P* e *Q*, e *P* vede un messaggio cifrato con *K* contenente *X* (e P non ha trasmesso quel messaggio), allora *P* crede che *X* sia stato trasmesso da *Q*

$$\frac{P \mid\equiv \overset{K}{\mapsto} Q, P \triangleleft \{X\}_{K^{-1}}}{P \mid\equiv Q \mid\sim X}$$

Se *K* è la chiave pubblica di *Q*, e *P* vede un messaggio firmato con K^{-1} contenente *X*, allora *P* crede che *X* sia stato trasmesso da *Q*

$$\frac{P \mid\equiv Q \overset{Y}{\rightleftharpoons} P, P \triangleleft \langle X \rangle_Y}{P \mid\equiv Q \mid\sim X}$$

Se *Y* è un segreto condiviso tra *P* e *Q*, e *P* vede un messaggio in cui Y è combinato con *X* (e P non ha trasmesso quel messaggio), allora *P* crede che *X* sia stato trasmesso da *Q*

Postulati: nonce verification rule

$$\frac{P \mid\equiv \#(X), P \mid\equiv Q \mid\sim X}{P \mid\equiv Q \mid\equiv X}$$

- Se *P* crede che *Q* abbia detto *X* e se P crede che *X* sia una quantità *fresh*, allora *P* crede che Q creda X (ora, cioè in questa esecuzione del protocollo)
- Se *P* crede che *X* sia stato inviato da *Q*, se *P* crede che *X* sia *fresh*, allora *P* crede che *Q* abbia inviato *X* in questa esecuzione del protocollo

Postulati: jurisdiction rule

$$\frac{P \mid\equiv Q \mid\equiv X, P \mid\equiv Q \Rightarrow X}{P \mid\equiv X}$$

- Se *P* crede che *Q* creda *X* e se *P* crede che *Q* sia un'autorità su *X*, allora anche *P* crede *X*
- Se *P* crede che *Q* dica *X* e se *P* si fida di *Q* per quanto riguarda *X*, allora anche *P* crede *X*

Altri postulati

$$\frac{P \mid\equiv X, \quad P \mid\equiv Y}{P \mid\equiv (X,Y)} \qquad \frac{P \mid\equiv (X,Y)}{P \mid\equiv X, \quad P \mid\equiv Y} \qquad \frac{P \mid\equiv Q \mid\equiv (X,Y)}{P \mid\equiv Q \mid\equiv X} \qquad \frac{P \mid\equiv Q \mid\sim (X,Y)}{P \mid\equiv Q \mid\sim X}$$

$$\frac{P \mid\equiv \#(X)}{P \mid\equiv \#(X,Y)}$$

$$\frac{P \lhd (X,Y)}{P \lhd X} \qquad \frac{P \lhd \langle X \rangle_Y}{P \lhd X}$$

$$\frac{P \mid\equiv Q \overset{K}{\leftrightarrow} P, \quad P \lhd \{X\}_K}{P \lhd X} \qquad \frac{P \mid\equiv \overset{K}{\mapsto} P, \quad P \lhd \{X\}_K}{P \lhd X} \qquad \frac{P \mid\equiv \overset{K}{\mapsto} Q, \quad P \lhd \{X\}_{K^{-1}}}{P \lhd X}$$

$$\frac{P \mid\equiv R \overset{K}{\leftrightarrow} R'}{P \mid\equiv R' \overset{K}{\leftrightarrow} R} \qquad \frac{P \mid\equiv Q \mid\equiv R \overset{K}{\leftrightarrow} R'}{P \mid\equiv Q \mid\equiv R' \overset{K}{\leftrightarrow} R} \qquad \frac{P \mid\equiv R \overset{K}{\rightleftharpoons} R'}{P \mid\equiv R' \overset{K}{\rightleftharpoons} R} \qquad \frac{P \mid\equiv Q \mid\equiv R \overset{K}{\rightleftharpoons} R'}{P \mid\equiv Q \mid\equiv R' \overset{K}{\rightleftharpoons} R}$$

Protocollo idealizzato

Tipicamente ogni passo di protocollo si rappresenta come

$$A \rightarrow B : messaggio$$

Ad esempio:

$$A \rightarrow B : \{A, K_{ab}\}_{K_{bs}}$$

Questa notazione informale è ambigua; il protocollo viene quindi idealizzato

$$A \rightarrow B : \left\{ A \overset{K_{ab}}{\leftrightarrow} B \right\}_{K_{bs}}$$

la specifica risultante è più chiara e si può dedurre la formula

$$B \lhd A \overset{K_{ab}}{\leftrightarrow} B$$

Analisi di un protocollo

L'analisi di un protocollo si articola sui seguenti passi:

1. Derivare il protocollo idealizzato dal protocollo reale
2. Determinare le ipotesi di partenza
3. Applicare i postulati a ciascun passo in modo da determinare i **belief** raggiunti dai partecipanti ai vari passi del protocollo
4. Derivare le conclusioni

Analisi di un protocollo

[*ipotesi*] S_1 [*asserzione* 1]

....

[*asserzione i*-1] S_i [*asserzione i*]

...

[*asserzione n*-1] S_n [*conclusioni*]

Assertion i-1 $A \mid\equiv A \overset{K}{\leftrightarrow} B$

Step i $A \rightarrow B : \{X\}_K$ applicando il postulato message meaning

Assertion i $A \mid\equiv A \overset{K}{\leftrightarrow} B, B \mid\equiv A \mid\sim X$

Obiettivi di un protocollo

L'obiettivo di un protocollo dipende, tipicamente, dal contesto

- Un protocollo di distribuzione delle chiavi, tipicamente, ha l'obiettivo

  $A \mid\equiv A \overset{K}{\leftrightarrow} B$ $B \mid\equiv A \overset{K}{\leftrightarrow} B$ (*key authentication*)

  e spesso $A \mid\equiv B \mid\equiv A \overset{K}{\leftrightarrow} B$ $B \mid\equiv A \mid\equiv A \overset{K}{\leftrightarrow} B$ (*key confirmation*)

  ed anche $A \mid\equiv \#\left(A \overset{K}{\leftrightarrow} B\right)$ $B \mid\equiv \#\left(A \overset{K}{\leftrightarrow} B\right)$ (*key freshness*)

- Un protocollo di interazione con una certification authority

  $A \mid\equiv \overset{e_b}{\mapsto} B$

Protocollo di Needham-Schroeder (1978)

protocollo reale

$$
\begin{array}{lll}
M1 & A \rightarrow T & A, B, N_a \\
M2 & T \rightarrow A & E_{K_a}\left(N_a, B, K_{ab}, E_{K_b}\left(K_{ab}, A\right)\right) \\
M3 & A \rightarrow B & E_{K_b}\left(K_{ab}, A\right) \\
M4 & B \rightarrow A & E_{K_{ab}}\left(N_b\right) \\
M5 & A \rightarrow B & E_{K_{ab}}\left(N_b - 1\right)
\end{array}
$$

protocollo idealizzato

$$
\begin{array}{lll}
M2 & T \rightarrow A & \left\{ N_a, \left(A \overset{K_{ab}}{\leftrightarrow} B \right), \#\left(A \overset{K_{ab}}{\leftrightarrow} B \right), \left\{ A \overset{K_{ab}}{\leftrightarrow} B \right\}_{K_b} \right\}_{K_a} \\
M3 & A \rightarrow B & \left\{ A \overset{K_{ab}}{\leftrightarrow} B \right\}_{K_b} \\
M4 & B \rightarrow A & \left\{ N_b, A \overset{K_{ab}}{\leftrightarrow} B \right\}_{K_{ab}} \text{ from } B \\
M5 & A \rightarrow B & \left\{ N_b, A \overset{K_{ab}}{\leftrightarrow} B \right\}_{K_{ab}} \text{ from } A
\end{array}
$$

Protocollo di Needham-Schroeder

$$M2 \quad T \rightarrow A \quad \left\{ N_a, \left(A \overset{K_{ab}}{\leftrightarrow} B \right), \#\left(A \overset{K_{ab}}{\leftrightarrow} B \right), \left\{ A \overset{K_{ab}}{\leftrightarrow} B \right\}_{K_b} \right\}_{K_a}$$

Dopo aver ricevuto N_a, T ha detto che K_{ab} è "buona" per parlare con Bob

$$M3 \quad A \rightarrow B \quad \left\{ A \overset{K_{ab}}{\leftrightarrow} B \right\}_{K_b}$$

T ha detto che K_{ab} è buona per parlare con Alice

$$M4 \quad B \rightarrow A \quad \left\{ N_b, A \overset{K_{ab}}{\leftrightarrow} B \right\}_{K_{ab}} \text{ from } B$$

Dopo aver aver ricevuto K_{ab}, *Bob* ha detto che K_{ab} è buona per parlare con Alice

$$M5 \quad A \rightarrow B \quad \left\{ N_b, A \overset{K_{ab}}{\leftrightarrow} B \right\}_{K_{ab}} \text{ from } A$$

Dopo aver aver ricevuto N_b, *Alice* ha detto che K_{ab} è buona per parlare con *Bob*

Principio 1. Ogni messaggio deve dire cosa significa; l'interpretazione di un messaggio deve dipendere solo dal suo contenuto; deve essere possibile scrivere una frase che descrive tale contenuto

Protocollo di Needham-Schroeder

Ipotesi

$A \mid\equiv A \overset{K_a}{\leftrightarrow} T$ $\quad$ $B \mid\equiv B \overset{K_b}{\leftrightarrow} T$

$T \mid\equiv A \overset{K_a}{\leftrightarrow} T$ $\quad$ $T \mid\equiv B \overset{K_b}{\leftrightarrow} T$

$T \mid\equiv A \overset{K_{ab}}{\leftrightarrow} B$

$A \mid\equiv \left(T \Rightarrow A \overset{K_{ab}}{\leftrightarrow} B \right)$ $\quad$ $B \mid\equiv \left(T \Rightarrow A \overset{K_{ab}}{\leftrightarrow} B \right)$

$A \mid\equiv \left(T \Rightarrow \#\left(A \overset{K_{ab}}{\leftrightarrow} B \right) \right)$

$A \mid\equiv \#(N_a)$ $\quad$ $B \mid\equiv \#(N_b)$

$T \mid\equiv \#\left(A \overset{K_{ab}}{\leftrightarrow} B \right)$ $\quad$ $B \mid\equiv \#\left(A \overset{K_{ab}}{\leftrightarrow} B \right)$

Obiettivi

$A \mid\equiv A \overset{K_{ab}}{\leftrightarrow} B$

$B \mid\equiv A \overset{K_{ab}}{\leftrightarrow} B$

$A \mid\equiv B \mid\equiv A \overset{K_{ab}}{\leftrightarrow} B$

$B \mid\equiv A \mid\equiv A \overset{K_{ab}}{\leftrightarrow} B$

Principio 2. Il progettista deve conoscere le relazioni di trust su cui il protocollo si basa e deve sapere perché tali relazioni sono necessarie. La ragione per cui una certa relazione è accettabile deve essere esplicita

Protocollo di Needham-Schroeder

Dopo M2

message meaning e nonce verification

$A \mid\equiv T \mid\equiv \left(A \overset{K_{ab}}{\leftrightarrow} B \right)$

$A \mid\equiv T \mid\equiv \#\left(A \overset{K_{ab}}{\leftrightarrow} B \right)$

jurisdiction rule

$A \mid\equiv \left(A \overset{K_{ab}}{\leftrightarrow} B \right)$

$A \mid\equiv \#\left(A \overset{K_{ab}}{\leftrightarrow} B \right)$

Dopo M3

message meaning

$B \mid\equiv T \mid\sim A \overset{K_{ab}}{\leftrightarrow} B$

nonce verification

$B \mid\equiv T \mid\equiv A \overset{K_{ab}}{\leftrightarrow} B$

jurisdiction rule

$B \mid\equiv A \overset{K_{ab}}{\leftrightarrow} B$

Principio 3. Una chiave può essere stata utilizzata recentemente per cifrare un nonce e tuttavia può essere vecchia o compromessa: l'uso recente di una chiave non la rende più buona

Dopo M4

message meaning

$A \mid\equiv B \mid\sim A \overset{K_{ab}}{\leftrightarrow} B$

nonce verification

$A \mid\equiv B \mid\equiv A \overset{K_{ab}}{\leftrightarrow} B$

Dopo M5

message meaning

$B \mid\equiv A \mid\sim \left(N_b, A \overset{K_{ab}}{\leftrightarrow} B \right)$

nonce verification

$B \mid\equiv A \mid\equiv A \overset{K_{ab}}{\leftrightarrow} B$

Protocollo Otway-Rees

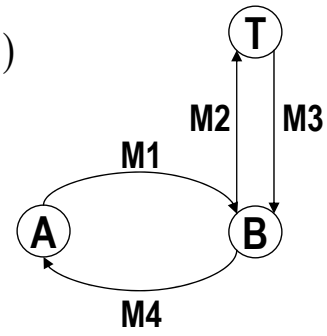
Protocollo reale

M1. $A \to B: \quad M, A, B, E_{K_A}(N_A, M, A, B)$

M2. $B \to T: \quad M, A, B, E_{K_A}(N_A, M, A, B), E_{K_B}(N_B, M, A, B)$

M3. $T \to B: \quad M, E_{K_A}(N_A, K_{ab}), E_{K_B}(N_B, K_{ab})$

M4. $B \to A: \quad M, E_{K_A}(N_A, K_{ab})$



Protocollo ideale

M1. $A \to B: \quad \{N_A, M, A, B\}_{K_a}$

M2. $B \to T: \quad \{N_A, M, A, B\}_{K_a}, \{N_B, M, A, B\}_{K_b}$

M3. $T \to B: \quad \left\{N_a, A \overset{K_{ab}}{\leftrightarrow} B, B \mid\sim M\right\}_{K_a}, \left\{N_b, A \overset{K_{ab}}{\leftrightarrow} B, A \mid\sim M\right\}_{K_b}$

M4. $B \to A: \quad \left\{N_b, A \overset{K_{ab}}{\leftrightarrow} B, A \mid\sim M\right\}_{K_a}$

Protocollo Otway-Rees

M1. $A \to B: \quad \{N_A, M, A, B\}_{K_a}$

M2. $B \to T: \quad \{N_A, M, A, B\}_{K_a}, \{N_B, M, A, B\}_{K_b}$

M3. $T \to B: \quad \left\{N_a, A \overset{K_{ab}}{\leftrightarrow} B, B \mid\sim M\right\}_{K_a}, \left\{N_b, A \overset{K_{ab}}{\leftrightarrow} B, A \mid\sim M\right\}_{K_b}$

M4. $B \to A: \quad \left\{N_a, A \overset{K_{ab}}{\leftrightarrow} B, B \mid\sim M\right\}_{K_a}$

M1: Alice dice che M è una transazione con Bob ed **N_a è un altro nome per Alice in M**

M2: Bob dice che M è una transazione con Bob ed **N_b è un altro nome per Bob in M**

M3: Dopo aver ricevuto N_b, T dice che K_{ab} è buona e che Alice credeva di essere in M

M4: Dopo aver ricevuto N_a, T dice che K_{ab} è buona e che Bob credeva di essere in M

Protocollo di Otway-Rees

Ipotesi

$$A \mid\equiv A \overset{K_a}{\leftrightarrow} T \qquad B \mid\equiv A \overset{K_b}{\leftrightarrow} T$$

$$T \mid\equiv A \overset{K_a}{\leftrightarrow} T \qquad T \mid\equiv A \overset{K_b}{\leftrightarrow} T$$

$$T \mid\equiv A \overset{K_{ab}}{\leftrightarrow} B$$

$$A \mid\equiv \left(T \Rightarrow A \overset{K}{\leftrightarrow} B \right) \qquad B \mid\equiv \left(T \Rightarrow A \overset{K}{\leftrightarrow} B \right)$$

$$A \mid\equiv \left(T \Rightarrow B \mid\sim M \right) \qquad B \mid\equiv \left(T \Rightarrow A \mid\sim M \right)$$

$$A \mid\equiv \#\left(N_a \right) \qquad B \mid\equiv \#\left(N_b \right)$$

$$A \mid\equiv \#\left(M \right)$$

Risultati

$$A \mid\equiv A \overset{K_{ab}}{\leftrightarrow} B$$

$$B \mid\equiv A \overset{K_{ab}}{\leftrightarrow} B$$

$$A \mid\equiv B \mid\equiv M$$

$$B \mid\equiv A \mid\sim M$$

Protocollo di Otway-Rees

dopo M2

$$T \mid\equiv A \mid\sim \left(N_a, M, A, B \right) \quad T \mid\equiv B \mid\sim \left(N_b, M, A, B \right)$$

dopo M3

$$B \mid\equiv T \mid\sim \left(N_b, A \overset{K_{ab}}{\leftrightarrow} B, A \mid\sim M \right)$$

data la freschezza di N_b per Bob, allora

$$B \mid\equiv T \mid\equiv \left(N_b, A \overset{K_{ab}}{\leftrightarrow} B, A \mid\sim M \right)$$

$$B \mid\equiv A \overset{K_{ab}}{\leftrightarrow} B, \quad B \mid\equiv A \mid\sim M$$

dato il trust che Bob ripone in T per quanto riguarda le chiavi e la sua capacità di fare relay, allora

dopo M4

$$A \mid\equiv T \mid\sim \left(N_a, A \overset{K_{ab}}{\leftrightarrow} B, B \mid\sim M \right)$$

data la freschezza di N_a per Alice, allora

$$A \mid\equiv T \mid\equiv \left(N_a, A \overset{K_{ab}}{\leftrightarrow} B, B \mid\sim M \right)$$

$$A \mid\equiv A \overset{K_{ab}}{\leftrightarrow} B, \quad A \mid\equiv B \mid\equiv M$$

dato il trust che Alice ripone in T per quanto riguarda le chiavi, la sua capacità di fare relay e la freschezza di M per Alice, allora

Protocollo di Otway-Rees

- **I nonce N_a ed N_b servono non solo per la freschezza ma anche per legare i messaggi M1 ed M2 ai messaggi M3 ed M4**
- **Il nonce N_a (N_b) è un riferimento ad Alice (Bob) nell'ambito di *M***
- **Nei messaggi M1 ed M2, la cifratura non serve per la segretezza ma serve per legare Alice (Bob), N_a (N_b) ed M**

Principio 4. Avere chiare le proprietà che si richiedono ai nonce: ciò che va bene per garantire freschezza può non andare bene per garantire un'associazione tra le parti

Principio 5. Avere chiaro il motivo per cui si utilizza la cifratura

Protocollo di Otway-Rees

- Se dovessero garantire solo la freschezza, allora M1 ed M2 si potrebbero modificare in (Otway-Rees modificato)

M1. $A \rightarrow B: \quad M, A, B, N_A, E_{K_A}(M, A, B)$

M2. $B \rightarrow T: \quad M, A, B, N_A, E_{K_A}(M, A, B), N_B, E_{K_B}(M, A, B)$

Il protocollo risultante sarebbe soggetto ad un attacco che porterebbe un avversario C ad impersonare Bob (Alice) rispetto ad Alice (Bob)

Protocollo di Otway-Rees

Il protocollo risultante sarebbe soggetto ad un attacco del tipo *man-in-the-middle* che porterebbe un avversario C ad impersonare Bob (Alice) rispetto ad Alice (Bob)

- Supponiamo che l'avversario C disponga di un "vecchio" **$E_{Ka}(M', A, C)$**

M1. $A \rightarrow B[C]: \quad M, A, B, N_a, E_{K_A}(M, A, B)$

M2. $C \rightarrow T: \quad M', A, C, N_a, E_{K_A}(M', A, C), N_c, E_{K_c}(M', A, C)$

M3. $T \rightarrow C: \quad M', E_{K_a}(N_a, K_{ab}), E_{K_c}(N_c, K_{ab})$

M4. $[C]B \rightarrow A: \quad E_{K_a}(N_a, K_{ab})$

Protocollo di Otway-Rees "migliorato"

- Se il problema è quello di inserire riferimenti ad Alice e Bob in M3 ed M4, rispettivamente, allora il protocollo può essere reso **più efficiente** e **più comprensibile** come segue:

M1. $A \rightarrow B: \quad A, B, N_a$

M2. $B \rightarrow T: \quad A, B, N_a, N_b$

M3. $T \rightarrow B: \quad E_{K_A}(N_a, A, B, K_{ab}), E_{K_B}(N_b, A, B, K_{ab})$

M4. $B \rightarrow A: \quad E_{K_A}(N_a, A, B, K_{ab})$

Principio 6. Se un identificatore è essenziale per il significato di un messaggio, è prudente menzionare esplicitamente tale identificatore nel messaggio

Protocollo SSL (vecchia versione)

Gli obiettivi del protocollo sono:

- Stabilire una chiave di sessione K_{ab} tra il cliente A ed il server B
- Autenticazione mutua tra A e B

Una delle prime versioni di SSL era strutturata come segue (solo i messaggi relativi all'autenticazione del cliente):

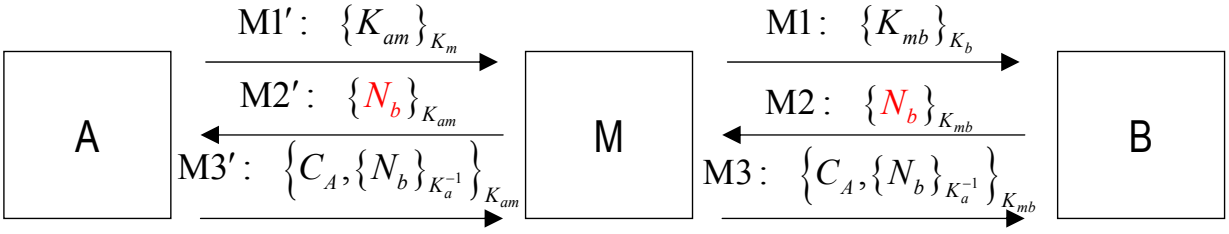
$$\text{M1.} \quad A \to B: \quad \{K_{ab}\}_{K_b}$$
$$\text{M2.} \quad B \to A: \quad \{N_b\}_{K_{ab}}$$
$$\text{M3.} \quad A \to B: \quad \left\{C_A, \{N_b\}_{K_a^{-1}}\right\}_{K_{ab}}$$

M1: Bob vede la chiave K_{ab}
M2: Dopo averla ricevuta, Bob dice di aver visto K_{ab}
M3: Dopo averlo ricevuto, Alice dice di aver visto N_b

Nei messaggi non c'è niente che autentichi il cliente, che leghi cioè il cliente A alla chiave K_{ab}

Protocollo SSL (vecchia versione)

La vecchia versione di SSL è soggetta all'attacco dell'uomo nel mezzo: l'avversario M può impersonare il client *A* rispetto al server *B*



L'attacco può essere evitato modificando il messaggio M3 come segue:

$$\text{M3} \quad A \to B: \quad \left\{C_A, \{A, B, K_{ab}, N_b\}_{K_a^{-1}}\right\}_{K_{ab}}$$

Dopo aver ricevuto N_b, Alice dice che K_{ab} è una chiave buona per comunicare con Bob

Firmare dati cifrati

Principio 7.

- Se un entità firma un messaggio cifrato, non è possibile inferire che tale entità conosce il contenuto del messaggio.
- Al contrario, se un entità firma un messaggio e poi lo cifra, allora è possibile inferire che tale entità conosce il contenuto del messaggio

Esempio: X.509

$$A \rightarrow B: \quad A, \left\{ T_a, N_a, B, X_a, \{Y_a\}_{K_b} \right\}_{K_a^{-1}}$$

il messaggio non contiene alcuna prova che il mittente (Alice) conosce Y_a

Sulle funzioni hash

Per motivi di efficienza, spesso si firma la hash di un messaggio invece dell'intero messaggio

$$A \rightarrow B: \quad \{X\}_{K_b}, \{h(X)\}_{K_a^{-1}}$$

- Il messaggio non contiene alcuna prova che il sottoscrittore (Alice) effettivamente conosce *X*,
- tuttavia, Alice si aspetta che il ricevente (Bob) si comporti come se Alice avesse piena conoscenza del messaggio
- Perciò, a meno che il sottoscrittore (Alice) non sia incauto*, firmare la hash di un messaggio equivale a firmare l'intero messaggio

* METAFORA: un manager che firma senza leggere

Postulati per le funzioni hash

$$\frac{P|\equiv Q|\sim h(X),\quad P \triangleleft X}{P|\equiv Q|\sim X}$$

Il postulato può essere generalizzato nel caso di *messaggio composto*

$$\frac{P|\equiv Q|\sim h(X_1,\cdots,X_n),\quad P \triangleleft X_1,\cdots,P \triangleleft X_n}{P|\equiv Q|\sim (X_1,\cdots,X_n)}$$

Si noti che *P* può ricevere i vari X_i da canali diversi in momenti diversi

Caso di studio: autenticazione in GSM

Protocollo reale

$$\begin{array}{lll} \text{M1.} & C \rightarrow S: & C \\ \text{M2.} & C \leftarrow S: & \rho \\ \text{M3.} & C \rightarrow S: & \sigma \end{array}$$

con

- ρ challenge random generata da *S*
- ⟨σ, K⟩ = $h(K_c, \rho)$

Ipotesi

$$S|\equiv C \overset{K_c}{\leftrightarrow} S \quad C|\equiv S \overset{K_c}{\leftrightarrow} C$$
$$S|\equiv \#(\rho)$$

Protocollo idealizzato

$$\text{M3.} \quad C \rightarrow S: \quad \langle C \overset{K}{\leftrightarrow} S, \rho \rangle_{K_c}$$

Risultati

$$S|\equiv C|\equiv S \overset{K}{\leftrightarrow} C$$

Nonce: quantità predicibili

Principio 8. Una quantità predicibile può essere usata come nonce in un protocollo di challenge-response; in tal caso deve essere protetta da un replay attack

Esempio: Alice riceve una marca temporale da un Time Server
(ad esempio Alice usa la marca per sincronizzare il suo clock)

$M1 \quad A \rightarrow S \quad A, N_a$
$M2 \quad S \rightarrow A \quad \{T_s, N_a\}_{K_{as}}$

- N_a nonce predicibile
- (M2): Dopo aver ricevuto N_a, S ha detto T_s

Ipotesi

$A \mid\equiv S \overset{K_{as}}{\leftrightarrow} A$
$A \mid\equiv S \Rightarrow T_s$
$A \mid\equiv \#(N_a)$

Risultati

$A \mid\equiv S \mid\sim T_s$
$A \mid\equiv S \mid\equiv T_s$
$A \mid\equiv T_s$

Nonce: quantità predicibili

Esempio di attacco

M predice il prossimo valore di N_a

$M1 \quad M \rightarrow S \quad A, N_a$
$M2 \quad S \rightarrow M \quad \{T_s, N_a\}_{K_{as}}$ (S riceve M2 all'istante T_s)

All'istante $T_S' > T_S$, Alice inizia un'istanza del protocollo

$M1 \quad A \rightarrow S[M] \quad A, N_a$
$M2 \quad S[M] \rightarrow A \quad \{T_s, N_a\}_{K_{as}}$

Alice è indotta a credere che l'ora attuale sia $\mathbf{T_s}$ e non $\mathbf{T_s'}$

Siccome N_a è predicibile, allora va protetto

$M1 \quad A \rightarrow S \quad A, \{N_a\}_{K_{as}}$
$M2 \quad S \rightarrow A \quad \left\{T_s, \{N_a\}_{K_{as}}\right\}_{K_{as}}$

Nonce: timestamp

Principio 9. Se la freshness dei messaggi è garantita riferendosi ad un riferimento assoluto di tempo per mezzo di marche temporali, allora la differenza tra il clock locale e quello delle altre macchine deve essere molto inferiore all'intervallo di validità di un messaggio. Inoltre, il meccanismo di sincronizzazione dei clock è parte della Trusted Computig Base (TCB)

Esempio

- Kerberos. Se il clock del server può essere "rimesso indietro", è possibile il riuso degli autenticatori*
- Kerberos. Se il clock del client può essere "rimesso avanti", è possibile creare autenticatori postdatati

* l'avversario può fare di più che semplicemente stabilire una connessione perché l'applicazione non richiede esplicitamente che le comunicazioni successive siano cifrate con le chiavi non disponibili all'avversario

Sulla codifica dei messaggi

Principio 10. Deve essere possibile dedurre: (i) a quale protocollo appartiene, (ii) a quale istanza di tale protocollo appartiene, (iii) qual è il suo numero all'interno del protocollo

Se un messaggio contiene nomi e freschezza a sufficienza, il punto (ii) è automaticamente garantito

Esempio: nel protocollo Needham-Schroeder

$$M4 \quad B \rightarrow A \quad E_{K_{ab}}(N_b)$$
$$M5 \quad A \rightarrow B \quad E_{K_{ab}}(N_b - 1)$$

$N_b - 1$ serve solo per distinguire la challenge dalla response

Sarebbe stato molto più chiaro se il messaggio fosse stato codificato così:

$$M4 \quad B \rightarrow A \quad E_{K_{ab}}(\text{N-S Message 4}, N_b)$$
$$M5 \quad A \rightarrow B \quad E_{K_{ab}}(\text{N-S Message 5}, N_b)$$