

Protocollo 1: messaggio m in chiaro e firmato (Diffie e Hellman)

- U : utente, $k_U[\text{priv}]$ e $k_U[\text{pub}]$: chiavi di U ,
- C e D : funzioni di cifratura e decifrazione di un cifrario asimmetrico

FIRMA

- U genera la firma $f = D(m, k_U[\text{priv}])$ per m
- U spedisce all'utente V la tripla $\langle U, m, f \rangle$

Protocollo 1: messaggio m in chiaro e firmato (Diffie e Hellman)

VERIFICA

- V riceve $\langle U, m, f \rangle$ e verifica l'autenticità della firma f calcolando e controllando che

$$C(f, k_U[\text{pub}]) = m$$

- L'indicazione del mittente U consente a V di selezionare la chiave pubblica $k_U[\text{pub}]$ da utilizzare nel calcolo
- I processi di firma e verifica impiegano le funzioni C e D in ordine inverso a quello standard, C e D devono essere commutative:

$$C(D(m)) = D(C(m)) = m$$

Protocollo 1: messaggio m in chiaro e firmato

Il protocollo soddisfa i requisiti della firma manuale

- f è **autentica e non falsificabile** (1)
 - k_U [priv] è nota solo a U
 - per falsificare la firma occorre conoscere k_U [priv], ma D è one-way
- il documento firmato $\langle U, m, f \rangle$ **non può essere alterato** se non da U , pena la non consistenza tra m e f (3)
- Poiché solo U può aver prodotto f , **U non può ripudiare la firma** (4)
- La firma f **non è riutilizzabile** su un altro documento $m' \neq m$ poiché è immagine di m (2)

Protocollo 1: messaggio m in chiaro e firmato

- È definito per un particolare utente U , ma non per un particolare destinatario.
- **Chiunque può convincersi dell'autenticità della firma facendo uso solo della chiave pubblica di U .**
- È solo uno schema di principio:
 - comporta lo scambio di un messaggio di lunghezza doppia dell'originale poiché la dimensione della firma è paragonabile alla dimensione del messaggio
 - il messaggio non può essere cifrato poiché è ricavabile pubblicamente dalla firma attraverso la verifica di questa.

Protocollo 2: messaggio m cifrato e firmato

FIRMA E CIFRATURA

- U genera la firma $f = D(m, k_U [\text{priv}])$ per m
- calcola il crittogramma firmato $c = C(f, k_V [\text{pub}])$
con la chiave pubblica del destinatario V
 - si incapsula la firma nel documento cifrato
- spedisce la coppia $\langle U, c \rangle$ a V.

Protocollo 2: messaggio m cifrato e firmato

DECIFRAZIONE E VERIFICA

- V riceve la coppia $\langle U, c \rangle$ e decifra il crittogramma:
$$D(c, k_V [\text{priv}]) = f$$
- Cifra tale valore con la chiave pubblica di U ottenendo
$$C(f, k_U [\text{pub}]) = C(D(m, k_U [\text{priv}]), k_U [\text{pub}]) = m$$
- V ricostruisce m, e se m è significativo attesta l'identità di U
Un utente diverso da U ha probabilità praticamente nulla di generare un crittogramma di significato accettabile se "cifrato" con la chiave pubblica di U.

Algoritmo per il protocollo 2

Cifrario RSA, con

- $\langle d_U \rangle, \langle e_U, n_U \rangle$ chiavi di U
- $\langle d_V \rangle, \langle e_V, n_V \rangle$ chiavi di V

utente U

- genera la firma del messaggio m: $f = m^{d_U} \bmod n_U$
- cifra f con la chiave pubblica di V: $c = f^{e_V} \bmod n_V$
- spedisce a V la coppia $\langle U, c \rangle$

utente V

- riceve la coppia $\langle U, c \rangle$ e decifra c: $c^{d_V} \bmod n_V = f$
- “decifra” f con la chiave pubblica di U: $f^{e_U} \bmod n_U = m$
- Se m è significativo, conclude che è autentico

Algoritmo per il protocollo 2

Per la correttezza del procedimento è necessario che:

- $n_U \leq n_V$ perché risulti $f < n_V$ e f possa essere cifrata correttamente e spedita a V
- questo impedirebbe a V di inviare messaggi firmati e cifrati a U
- ogni utente stabilisce chiavi distinte per la firma e per la cifratura:
 - si fissa pubblicamente H molto grande, ad esempio $H = 2^{1024}$
 - chiavi di firma $\langle H$
 - chiavi di cifratura $\rangle H$

Algoritmo per il protocollo 2

- Il valore elevato di H assicura che tutte le chiavi possano essere scelte sufficientemente grandi, e quindi inattaccabili in modo esauriente.
- Il meccanismo di firma si presta tuttavia a diversi attacchi

basati sulla possibilità che un crittoanalista si procuri la firma di un utente su messaggi apparentemente privi di senso.

Protocollo 2: attacco 1

Supponiamo che:

- un utente U invii una risposta automatica (ack) a MITT ogni volta che riceve un messaggio m
- il segnale di ack sia il crittogramma della firma di U su m .

Un crittoanalista attivo X può decifrare i messaggi inviati a U :

- X intercetta il crittogramma c firmato inviato da V a U , lo rimuove dal canale e lo rispedisce a U , facendogli credere che c sia stato inviato da lui.
- U spedisce automaticamente a X un ack
- X usa l'ack ricevuto per risalire al messaggio originale m applicando le funzioni del cifrario con le chiavi pubbliche di V e di U

Protocollo 2: attacco 1

1. V invia il crittogramma c a U:
 - $c = C(f, k_U[\text{pub}])$
 - $f = D(m, k_V[\text{priv}])$
2. Il crittoanalista X intercetta c , lo rimuove dal canale e lo rispedisce a U (U crede che c arrivi da X)
3. U decifra c
 $f = D(c, k_U[\text{priv}])$
e verifica la firma con la chiave pubblica di X ottenendo un messaggio
 $m' = C(f, k_X[\text{pub}])$
4. $m' \neq m$ è un messaggio privo di senso, ma il sistema manda l'ack c' a X
 $f' = D(m', k_U[\text{priv}])$
 $c' = C(f', k_X[\text{pub}])$

Protocollo 2: attacco 1

X usa c' per risalire al messaggio m

1. decifra c' e trova f'
 $D(c', k_X[\text{priv}]) = D(C(f', k_X[\text{pub}]), k_X[\text{priv}]) = f'$
2. verifica f' e trova m'
 $C(f', k_U[\text{pub}]) = C(D(m', k_U[\text{priv}]), k_U[\text{pub}]) = m'$
3. da m' ricava f
 $D(m', k_X[\text{priv}]) = D(C(f, k_X[\text{pub}]), k_X[\text{priv}]) = f$
4. verifica f con la chiave pubblica di V e trova m
 $C(f, k_V[\text{pub}]) = C(D(m, k_V[\text{priv}]), k_V[\text{pub}]) = m$

Protocollo 2: attacco 1

- Per evitare questo attacco occorre che U blocchi l'ack automatico
- l'ack deve essere inviato solo dopo un esame preventivo di m e scartando i messaggi che non si desidera firmare

Protocollo resistente agli attacchi

- Si evita la firma diretta del messaggio
- Si appone la firma digitale su una immagine del messaggio ottenuta con una funzione hash one-way e pubblica (tipo MD5, SHA)
- La firma non è più soggetta a giochi algebrici

Protocollo 3: messaggio m cifrato e firmato in hash

FIRMA E CIFRATURA

- il mittente U calcola $h(m)$ e genera la firma

$$f = D(h(m), k_U[\text{priv}])$$

- calcola separatamente il crittogramma

$$c = C(m, k_V[\text{pub}])$$

- spedisce a V la tripla $\langle U, c, f \rangle$

Protocollo 3: messaggio m cifrato e firmato in hash

DECIFRAZIONE E VERIFICA

- V riceve la tripla $\langle U, c, f \rangle$
- decifra il crittogramma c: $m = D(c, k_V[\text{priv}])$
- calcola separatamente $h(m)$ e $C(f, k_U[\text{pub}])$
- se $h(m) = C(f, k_U[\text{pub}])$ conclude che il messaggio è autentico.

Protocollo 3: messaggio m cifrato e firmato in hash

- richiede lo scambio di una maggiore quantità di dati, ma l'incremento è trascurabile
- la firma può essere calcolata più velocemente

Attacchi man in-the-middle

- Debolezza dei protocolli:
 - le chiavi di cifratura sono pubbliche e non richiedono un incontro diretto tra gli utenti per il loro scambio.
- Un crittoanalista attivo può intromettersi proprio in questa fase iniziale del protocollo, pregiudicando il suo corretto svolgimento.
- Attacco attivo chiamato **man in-the-middle**:
 - Il crittoanalista si intromette nella comunicazione tra U e V,
 - si comporta agli occhi di U come se fosse V,
 - si comporta agli occhi di V come se fosse U,
 - intercettando e boccando le comunicazioni di U e V

Attacchi man in-the-middle sulle chiavi pubbliche

Il crittoanalista X devia le comunicazioni che provengono da U e V e le dirige verso se stesso:

- U richiede a V la sua chiave pubblica (attraverso e-mail, pagina web, ...)
- X intercetta la risposta contenente k_V [pub] e la sostituisce con la sua chiave pubblica k_X [pub].
- X si pone in ascolto in attesa dei crittogrammi spediti da U a V, cifrati mediante k_X [pub].
- X rimuove dal canale ciascuno di questi crittogrammi, e lo decifra, lo cifra mediante k_V [pub] e lo rispedisce a V.
- U e V non si accorgeranno della presenza di X se il processo di intercettazione e rispedizione è sufficientemente veloce da rendere il relativo ritardo apparentemente attribuibile alla rete.

Certification Authority

- Un algoritmo crittografico è tanto robusto quanto la sicurezza delle sue chiavi
- lo scambio/generazione della chiave è un passo cruciale
- gli attacchi man-in-the-middle sono i principali problemi di sicurezza da affrontare
- Sono nate le **Certification authority**
sono infrastrutture che garantiscono la validità delle chiavi pubbliche e ne regolano l'uso, gestendo la distribuzione sicura delle chiavi alle due entità che vogliono comunicare

Key Certification Authority (CA)

ente preposto alla certificazione di validità delle chiavi pubbliche

- La CA autentica l'associazione <utente, chiave pubblica> emettendo un certificato digitale.
- Il certificato digitale consiste della chiave pubblica e di una lista di informazioni relative al suo proprietario, opportunamente firmate dalla CA.
- Per svolgere correttamente il suo ruolo, la CA mantiene un archivio di chiavi pubbliche sicuro, accessibile a tutti e protetto da attacchi in scrittura non autorizzati.
- La chiave pubblica della CA è nota agli utenti che la mantengono protetta da attacchi esterni e la utilizzano per verificare la firma della CA stessa sui certificati.

Certificato digitale

Un certificato digitale contiene:

- una indicazione del suo formato (numero di versione)
- il nome della CA che lo ha rilasciato
- un numero seriale che lo individua univocamente all'interno della CA emittente
- la specifica dell'algoritmo usato dalla CA per creare la firma elettronica
- il periodo di validità del certificato (data di inizio e data di fine)
- il nome dell'utente a cui questo certificato si riferisce e una serie di informazioni a lui legate
- un'indicazione del protocollo a chiave pubblica adottato da questo utente per la cifratura e la firma: nome dell'algoritmo, suoi parametri, e chiave pubblica dell'utente
- firma della CA eseguita su tutte le informazioni precedenti.

CA

- Se U vuole comunicare con V può richiedere K_V [pub] alla CA, che risponde inviando a U il certificato digitale cert_V di V
- Poiché U conosce K_{CA} [pub], può controllare l'autenticità del certificato verificandone il periodo di validità e la firma prodotta dalla CA su di esso.
- Se tutti i controlli vanno a buon fine, K_V [pub] nel certificato è corretta e U avvia la comunicazione con V
- Un crittoanalista potrebbe intromettersi solo falsificando la certificazione, ma si assume che la CA sia fidata e il suo archivio di chiavi sia inattaccabile

CA

Esistono in ogni stato diverse CA organizzate ad albero

la verifica di un certificato è in questo caso più complicata e si svolge attraverso una catena di verifiche che vanno dalla CA di U alla CA di V

- V invia a U una sequenza di certificati ordinati in accordo alla CA che li ha firmati, da quella che ha certificato V a quella che è alla radice dell'albero delle CA

Ogni utente mantiene localmente al sistema una copia del proprio certificato cert_U e una copia di K_{CA} [pub]

interagisce con la CA una sola volta, e poi la gestione delle chiavi pubbliche diventa decentralizzata

Protocollo 4: messaggio m cifrato, firmato in hash e certificato

FIRMA, CIFRATURA E CERTIFICAZIONE

Il mittente U:

- si procura il certificato $cert_V$ di V
- calcola $h(m)$ e genera la firma

$$f = D(h(m), k_U[\text{priv}])$$

- calcola il crittogramma

$$c = C(m, k_V[\text{pub}])$$

- spedisce a V la tripla $\langle cert_U, c, f \rangle$
 - $cert_U$ contiene la chiave $k_U[\text{pub}]$ e la specificazione della funzione h utilizzata

Protocollo 4

VERIFICA DEL CERTIFICATO

- il destinatario V riceve la tripla $\langle cert_U, c, f \rangle$ e verifica l'autenticità di $cert_U$ (e dunque di $K_U[\text{pub}]$) utilizzando la propria copia di $K_{CA}[\text{pub}]$

DECIFRAZIONE E VERIFICA DELLA FIRMA

- V decifra il crittogramma c mediante la sua chiave privata e ottiene

$$m = D(c, k_V[\text{priv}])$$

- V verifica l'autenticità della firma apposta da U su m controllando che risulti

$$C(f, k_U[\text{pub}]) = h(m)$$

Conclusioni

- Il punto debole di questo meccanismo è rappresentato dai certificati revocati, i.e., non più validi
- Le CA mettono a disposizione degli utenti un archivio pubblico contenente i certificati non più validi
- La frequenza di controllo di questi certificati e le modalità della loro comunicazione agli utenti sono cruciali per la sicurezza
- Attacchi man-in-the-middle possono essere evitati se si stabilisce un incontro diretto tra utente e CA nel momento in cui si istanzia il sistema asimmetrico dell'utente