Autenticazione

- Garantisce l'identità di una "parte" ad una altra "parte".
  - ATTENZIONE: Autenticazione è diversa dall'identificazione!
- Le parti possono essere utenti o computer.
- Spesso richiesta autenticazione bi-direzionale. Autenticazione di un computer ad un utente per prevenire attacchi di spoofing in cui un computer pretende di essere un altro computer (per acquisire la password degli utenti).
- Spesso necessaria la combinazione di autenticazione "utente-a-computer" e "computer-a-computer".

Autenticazione

- È per molti aspetti il servizio di sicurezza primario.
  - La correttezza del controllo degli accessi si basa su una corretta autenticazione.
  - La correttezza del controllo delle intrusioni degli accessi si basa su una corretta autenticazione.

Autenticazione e crittografia

L'autenticazione può fare uso di misure crittografiche.

Crittografia: dato un testo in chiaro produce una versione non intelligibile, detta testo cifrato, crittografato o, colloquialmente, crittato. È ovviamente possibile anche la trasformazione inversa. Si utilizza una chiave per cifrare e decifrare.

Distinguiamo crittografia a

- chiave privata (o simmetrica) La stessa chiave è utilizzata per cifrare e decifrare
- chiave pubblica (o asimmetrica) Utilizza una coppia di chiavi ($\langle$pubblica,privata$\rangle$): una per cifrare e l'altra per decifrare

Autenticazione utente a computer

Può essere basata su:

- qualcosa che l'utente conosce, *something you know* (es., password)
- qualcosa che l'utente possiede, *something you have* (es., tesserina magnetica)
- qualcosa che l'utente è, stabilito sulla base di caratteristiche biometriche, *something you are* (es., impronte digitali, retina, iride, mano, tratti somatici, digitazione, scrittura, impronta della voce, ...).

o una combinazione di queste.

Autenticazione basata su password

- Basata su coppia:
  - login: l'utente si identifica
  - password: l'utente fornisce prova della sua identità.
- Metodo di autenticazione più antico e più diffuso (e lo sarà ancora per diverso tempo).
  - è il più semplice
  - è il più economico
  - è di facile implementabilità

    però ....
  - è anche il più debole

Vulnerabilità delle password

- Spesso le password possono:
  - essere facilmente indovinate (guessing)
  - osservate da persone che osservano l'utente legittimo mentre la scrive (snooping o shoulder surfing)
  - acquisite da terze parti nella comunicazione lungo la rete (sniffed)
  - acquisite da terze parti che impersonano l'interfaccia di login (spoofing)
  - ottenute tramite ricerca esaustiva (cracking)
- Chiunque riesca a conoscere la password di un utente può impersonare (masquerading) l'utente nell'accesso al sistema.

Vulnerabilità delle password

Una delle maggiori cause della vulnerabilità delle password è rappresentata dagli utenti che non le scelgono o gestiscono propriamente.

Cause di vulnerabilità delle password

Il primo passo per limitare la vulnerabilità delle password è una buona gestione delle password.
Spesso le password sono vulnerabili perchè gli utenti non pongono abbastanza cura:

- non cambiano la password per molto tempo
- condividono la password con colleghi o amici
- scelgono password “deboli” perchè facili da ricordare (e.g., nome di parenti o animali domestici, o date di nascita)
- usano la stessa password su più computer
- scrivono le password su pezzi di carta per essere sicuri di non dimenticarla.

Vulnerabilità di progetto

In alcuni casi il sistema forza all'uso di password deboli, per comodità e facile accettazione

- codice fiscale
- cognome della madre
- nome delle scuole frequentate
- Problemi
  - robusto solo se si usano diverse informazioni contemporaneamente, scelte da un pool abbastanza ampio
  - se voglio usare un valore falso per maggiore robustezza (il cognome di mia madre per la banca è "Godzilla"), posso violare il vincolo legale di correttezza della dichiarazione

Vulnerabilità di progetto

In alcuni casi il cambio di contesto non rende più vere le assunzioni iniziali

- Uso di Bancomat nei normali punti vendita

Altre vulnerabilità:

- password costanti (`scott/tiger`)
- password rappresentate in chiaro (su file o in memoria)
- tracce nei log, per autenticazioni fallite

Le password è bene che siano rappresentate con una funzione hash, protette da un nonce, o *salt* (ancora più importante oggi, a causa delle *rainbow tables*)

- è bene comunque che anche la rappresentazione hash sia protetta, per evitare l'uso di strumenti di *cracking* (nel vecchio Unix tutto stava in `/etc/passwd`)

Scelta delle password

Una buona gestione richiede che gli utenti

- cambino spesso la propria password
- mantengano le loro password private
- scelgano password che non siano facili da indovinare.
  Una buona password dovrebbe
  - essere di almeno 8 caratteri e utilizzare un insieme di caratteri abbastanza largo (sia caratteri alfanumerici sia caratteri speciali).
  - non essere facilmente intuibile e non corrispondere a parole di dizionari (o leggere variazioni di queste).
  - facile da ricordare, altrimenti
    * gli utenti la scrivono
    * gli utenti la dimenticano (denials-of-service)

..... Sfortunatamente spesso questi principi di base non sono seguiti.

Controlli su password

Molti sistemi utilizzano controlli automatici per evitare grosse debolezze nelle password.

- restrizioni su lunghezza e sul minimo numero di caratteri, richiedendo combinazione di caratteri numerici o alfanumerici.
- controllo rispetto a dizionari e rifiuto di parole del linguaggio naturale (per prevenire dictionary attacks).
- massimo tempo di validità di password (gli utenti sono obbligati a cambiare la password quando “scade”).
  - mantenimento della storia recente delle password
  - mantenimento di tempo minimo di validità

Controlli su password

In alternativa

- la password può essere scelta dal sistema.
  - non sempre ben accetto (password difficili da ricordare)
  - problema di distribuzione delle password.
- utilizzo di sequenze one-time-password (spesso gli utenti le scrivono per ricordarsele).

Distribuzione iniziale di password

- L'utente si reca fisicamente dall'amministratore e si autentica tramite metodi tradizionali (es., carta di identità). L'amministratore prepara l'account e l'utente digita la password.
  - scomodità per l'utente
  - pericolo per il sistema (l'utente accede al sistema in modalità superuser, mentre l'amministratore non guarda!).
    A volte la password è digitata da una tastiera speciale (es., ATM)
- L'amministratore prepara l'account con una password iniziale e la comunica all'utente che dovrà cambiarla al primo utilizzo (pre-expired password).
  - la password dovrebbe essere abbastanza forte (e.g., scelta casualmente).
  - spesso password iniziale molto debole per convenienza (es., matricola studente).

Autenticazione basata su possesso

- Basata su possesso da parte degli utenti di token (oggetto spesso della dimensione di una carta di credito).
- Ogni token ha una chiave crittografica (memorizzata nel token) che può essere utilizzata per dimostrare l'identità del token a un computer.
- Token sono più sicuri delle password
  - mantenendo controllo sul token l'utente mantiene controllo sull'utilizzo della sua identità

Vulnerabilità dei token

- L'autenticazione basata sul token dimostra solo l'identità del token, non quella dell'utente
  - i token possono essere persi, rubati o falsificati
  - chiunque acquisisca un token può impersonare l'utente
- spesso autenticazione basata su token è combinata con autenticazione basata su conoscenza.
  - Per compiere violazioni, terze parti necessitano sia del token sia di conoscere la password
  - Es., Bancomat richiede: Tessera bancomat + PIN (Personal Identification Number).

Autenticazione basata su token

Due tipi di token:

- memory card: hanno memoria ma non hanno capacità di processo.
  - non possono controllare il PIN o crittografarlo per la trasmissione.
  - il PIN è trasmesso in chiaro
    - vulnerabile ad attacchi di sniffing
    - richiede fiducia nel server di autenticazione
    - Es: telecomandi per l'apertura di cancelli; tessera con elenco di codici stampati.
- smart token: hanno capacità di processo.
  - Es.: chiavi elettroniche per automobili; CIE.

Smart token

Smart token possono utilizzare differenti protocolli di autenticazione

- scambio statico di password: l'utente si autentica al token e il token si autentica al sistema.
- generazione dinamica di password: il token genera periodicamente nuove chiavi.
- challenge-response: basato su un challenge-response handshake.

In tutti i casi, la comunicazione fra token e sistema può avvenire direttamente o tramite utente.

Autenticazione con scambio statico di password

- Il token invia un semplice ID
- Problemi:
  - Se il canale di comunicazione è aperto (radio), è immediatamente vulnerabile a *replay attacks*.
  - Anche se il canale non è aperto, diventa difficile evitare che un malintenzionato acceda alle informazioni di autenticazione

Autenticazione con generazione dinamica di password

- Token che invia un ID e un valore numerico cifrato con una chiave
- Il dialogo non è bidirezionale ed è molto più economico
- Varie tecniche: random, numero progressivo, SKEY
- Random
  - Client invia al server $(ID, r, E_{k_{ID}}(r))$
- Numero progressivo
  - Setup: Client e server mantengono ciascuno un contatore c_c e c_s; all'inizio, $c_c := 0$ e $c_s := 0$
  - Uso
    - Client invia al server $(ID, [c_c,]E_{k_{ID}}(c_c))$;
    - Server controlla se c_c è coerente con c_s e se la cifratura è corretta
    - Client: $c_c = c_c + 1$; Server: $c_s = c_c + 1$;

Tecniche di autenticazione 1-way: SKEY

- Setup
  - random x_0
  - $x_1 = h(x_0); \ldots; x_n = h(x_{n-1})$
  - L'utente riceve la lista $x_0, \ldots, x_{n-1}$; il server conserva solo x_n
- Uso
  - Client invia x_i, ultimo valore del suo elenco (la prima volta, x_{n-1})
  - Server verifica se $h(x_i)$ è pari al valore conservato (la prima volta, x_n)
  - Il client butta via il valore x_i
  - Il server conserva solo il valore x_i, per la verifica successiva

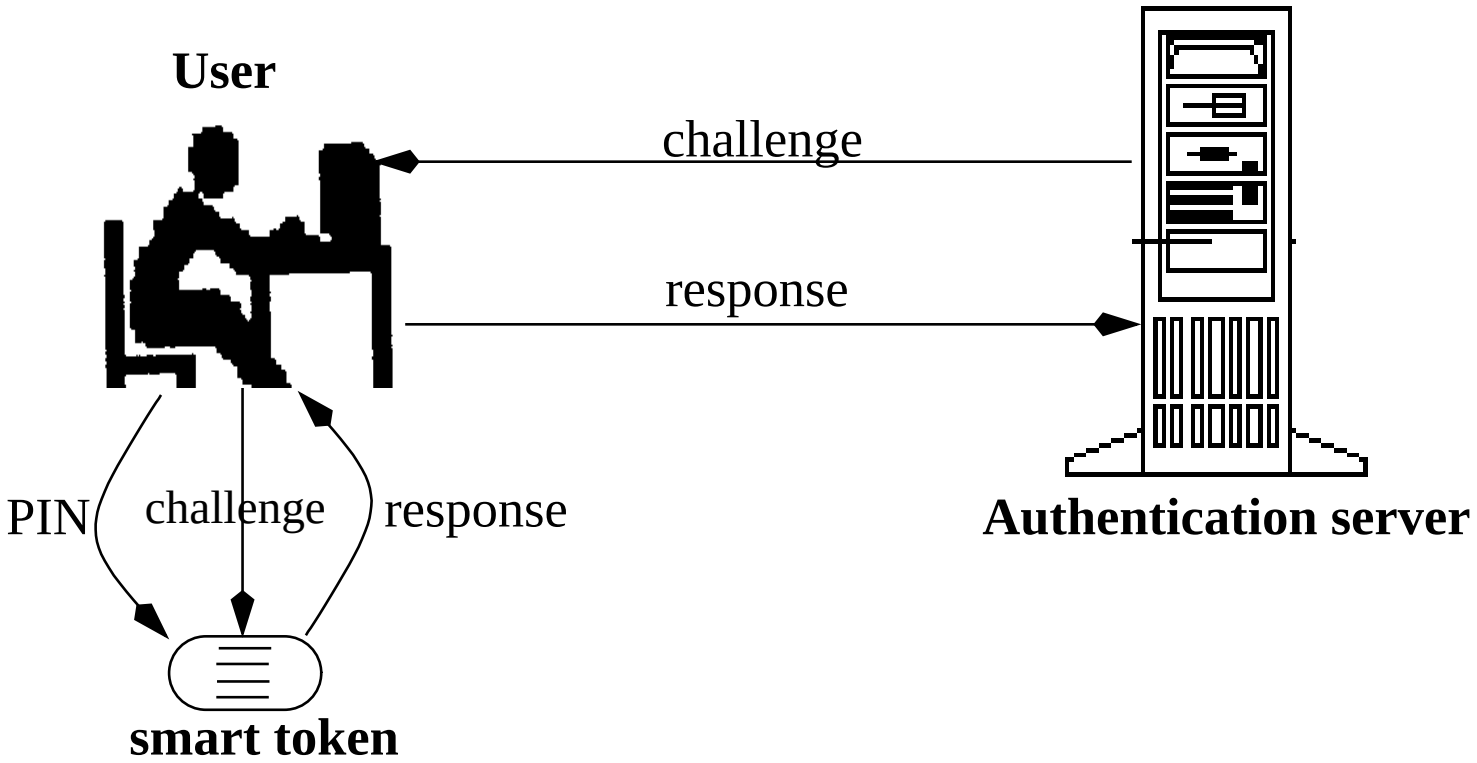
Problemi delle tecniche di autenticazione 1-way

- Problemi
  - Random: chi ascolta deve conservare uno o più codici, per verificare che sia un valore nuovo e contrastare attacchi *replay*; ma quanti bastano?
  - Counter: cosa capita quando il dispositivo viene usato "separatamente" e si perde il sincronismo tra client e server?
  - SKEY: il numero di valori per l'accesso è limitato
- Le soluzioni reali, per offrire una buona usabilità, spesso risultano relativamente fragili.
- Si ottengono soluzioni molto interessanti se si dispone di un orologio affidabile interno al token
  - Client invia HMAC(k_{ID}, t)

Autenticazione challenge-response

- il server di autenticazione stabilisce una sfida (challenge). Es., numero casuale.
- il token genera la risposta alla sfida utilizzando K_{ID} del token.
- l'autenticazione ha successo se la risposta è corretta.
- si possono usare certificati e una PKI per garantire robustezza rispetto a Man-In-The-Middle (MITM)

Esempio di challenge-response handshake



Challenge response talvolta non basta

- Esempio di applicazione: Identify Friend or Foe (IFF)
  - challenge response tradizionale è vulnerabile ad attacchi relay
  - per impedire un relay attack, si deve legare la response a caratteristiche autenticabili di chi risponde
  - in ambito militare anche la challenge deve essere autenticata (altrimenti, non serve a nulla essere stealth ai radar)
  - è critico il buon comportamento del generatore di valori random (`/dev/random` e `/dev/urandom`)

Autenticazione basata su caratteristiche dell'utente

- Basata su caratteristiche biometriche dell'utente:
  - caratteristiche fisiche: impronte digitali, forma della mano, impronta della retina o del viso, ...
  - caratteristiche comportamentali: firma, timbro di voce, scrittura, "keystroke dynamic"....
- Richiede una fase iniziale (enrollment phase)
  - esecuzione di più misurazioni sulla caratteristica di interesse
  - definizione di un template

Autenticazione basata su caratteristiche dell'utente

- Autenticazione: confronto fra caratteristica misurata per l'utente sotto controllo rispetto al template memorizzato
- L'autenticazione ha successo se la misura e il template corrispondono (a meno di un intervallo di tolleranza)
- Non si può richiedere perfetta uguaglianza (i sistemi biologici evolvono)
- È importante definire la soglia di tolleranza in modo da massimizzare i successi (autenticazione corretta di utenti legittimi e rifiuto di intrusi) e minimizzare gli insuccessi (falsi positivi e falsi negativi).

Autenticazione biometrica

- Anche se meno accurate sono una forma di autenticazione indispensabile in certi contesti
  - eliminano vulnerabilità dovute a impersonificazioni.
- Oggi poco utilizzate
  - troppo costose (necessitano di hardware costoso)
  - intrusive (non sempre accettate dagli utenti)
    - Gli scanner della retina sono una tra le misure più accurate, ma gli utenti temono danni agli occhi causati dal laser.
  - dibattiti politici e sociali per potenziale mancanza di privacy.

Considerazioni di fondo sull'autenticazione biometrica

- Sono soluzioni che verranno sempre più utilizzate in certi ambiti
  - es.: controlli di identità alla frontiera
- È opportuno monitorare rispetto a certi usi (es., archivio DNA)
- Negli ambiti informatici tradizionali si osservano dei problemi di fondo, che riducono il potenziale di queste tecniche:
  - Per qualunque indicatore biometrico, esiste qualche elemento della popolazione per cui l'indicatore non è applicabile
  - Il criterio di autenticazione è spesso facilmente osservabile e poi falsificabile, di norma con risorse limitate; l'uso "non supervisionato" presenta un modesto livello di robustezza
  - Una volta che un ID è compromesso, è difficile da rinnovare
- Un approccio che risolve questi problemi: smart tag impiantato nel corpo (ma quanti utenti sono disponibili?)

Quale tecnica di autenticazione?

- Esiste metodo di autenticazione “più forte”, non “migliore”
  - Trade-off fra costi e benefici: metodi più deboli possono andare bene in certi casi
  - Le password sono (e saranno ancora nel breve futuro) il meccanismo di autenticazione più utilizzato.
  - Tecniche di autenticazione biometrica “soft” troveranno diverse applicazioni in contesti nei quali si vuole concedere un privilegio di accesso flessibile alla persona
- Da un punto di vista prettamente tecnico il miglior metodo di autenticazione è costituito dalla combinazione di:
  - autenticazione biometrica e basata su password fra utente e token
  - mutua autenticazione basata su crittografia forte fra token e sistema.
- Attenzione a non consentire aggiramenti

Architetture software per l'autenticazione

- Remote Authentication Dial-In User Service - RADIUS
  - Protocollo per AAA definito in vari RFC (inizialmente RFC 2058 e RFC 2059; ora RFC 2865 e RFC 2866)
    - Authentication: raccoglie le credenziali dell'utente (tipicamente, mediante un digest)
    - Authorization: permette di specificare quali risorse può usare ciascun utente
    - Accounting: tiene traccia delle richieste effettuate (uso tipico, archiviazione delle richieste di inizio e fine sessione)
  - Spesso un componente centrale per la costruzione di soluzioni di autenticazione in rete
  - Integrato con SNMP, VoIP, 802.1x; sono disponibili implementazioni per tanti sistemi (apparati di rete, vari OS)

Architetture software per l'autenticazione

- Lightweight Directory Access Protocol - LDAP
  - protocollo per la memorizzazione e il recupero di dati in formato gerarchico
  - progettato per la realizzazione di directory services, a partire dall'iniziativa X.500
  - viene spesso utilizzato per conservare le credenziali degli utenti e la descrizione delle risorse del sistema, organizzate ad albero

Architetture software per l'autenticazione

- Kerberos
  - Protocollo sviluppato all'MIT (fine anni 70) per l'autenticazione in un sistema distribuito
  - Usa solo tecniche di cifratura simmetrica
  - Windows usa una versione modificata (non compatibile) per l'autenticazione in un ambito distribuito
  - Esiste una Trusted Third Party, organizzata in un Authentication Server e un Ticket Granting Server

Le architetture specifiche spesso usano diversi componenti (abbastanza comune RADIUS+LDAP). Un obiettivo diffuso è il Single Sign On