

# Sicurezza delle applicazioni



Il progetto di sicurezza di  
un sistema di servizi



# Sicurezza di un sistema

- La sicurezza di un sistema di servizi è sempre pari alla sicurezza più bassa posseduta da un suo componente.
- Le componenti di un sistema di servizi hanno una triplice natura: sw, hw e di rete (o connessione).
- Ogni componente deve concorrere affinché il sistema di servizi riesca a garantire:
  - la riservatezza,
  - l'integrità,
  - la disponibilità

delle risorse



# Sicurezza del software

*Criticità con cui convivere*

- **Strutturale debolezza della catena produttiva** (progettazione, stesura specifiche, prototipo, auditing, rilascio, ...)
- **Difficoltà nella misurazione dell'errore nel meccanismo di produzione** (oggettive e di tipo psicologico: *il mio sistema è quello più sicuro*)



**Difficoltà nella  
certificazione del singolo  
prodotto software**

# Certificazione della produzione del software



Il progetto di  
sicurezza di  
sistema di ser

- La certificazione del software induce
  - un aumento del costo di produzione,
  - il rallentamento dell'emissione sul mercato del prodotto (e soprattutto delle nuove versioni del prodotto che devono essere a loro volta certificate).

**Modello di certificazione**

Meccanismo di produzione  
del software

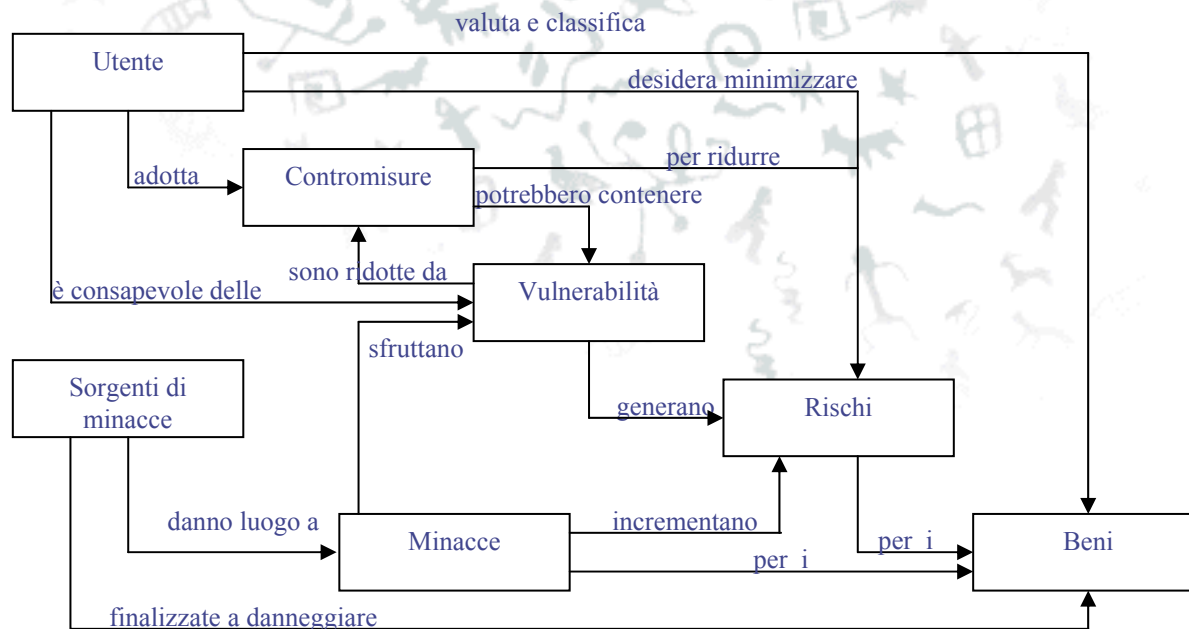
Meccanismo di distribuzione  
del software certificato

# Certificazione del modello di produzione del sw



Il progetto di sicurezza di un sistema di servizi

Lo standard di riferimento è l'ISO/IEC 15408 (1-2-3), che prevede il seguente contesto:



# La valutazione della sicurezza



Il progetto di sicurezza di un sistema di servizi

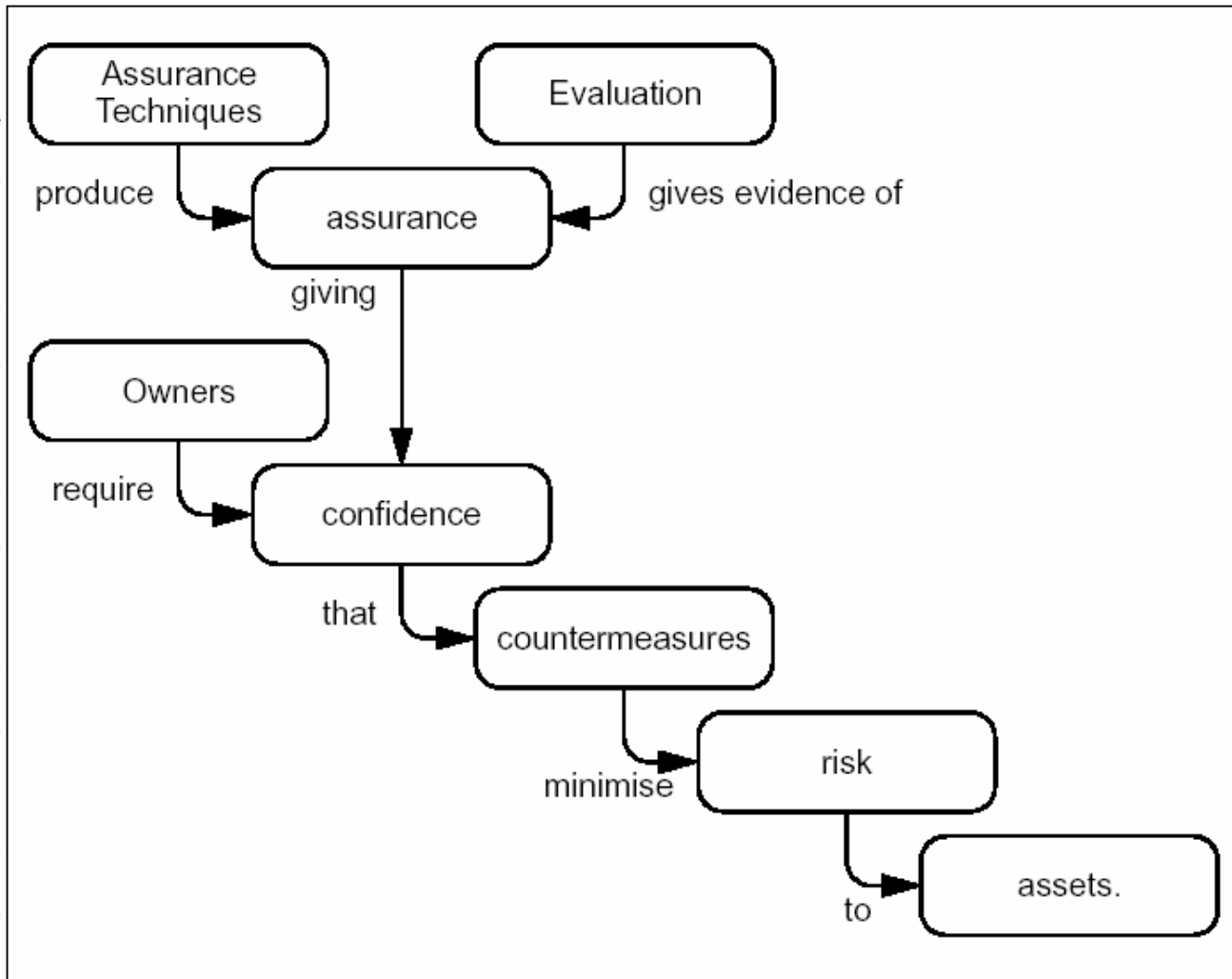
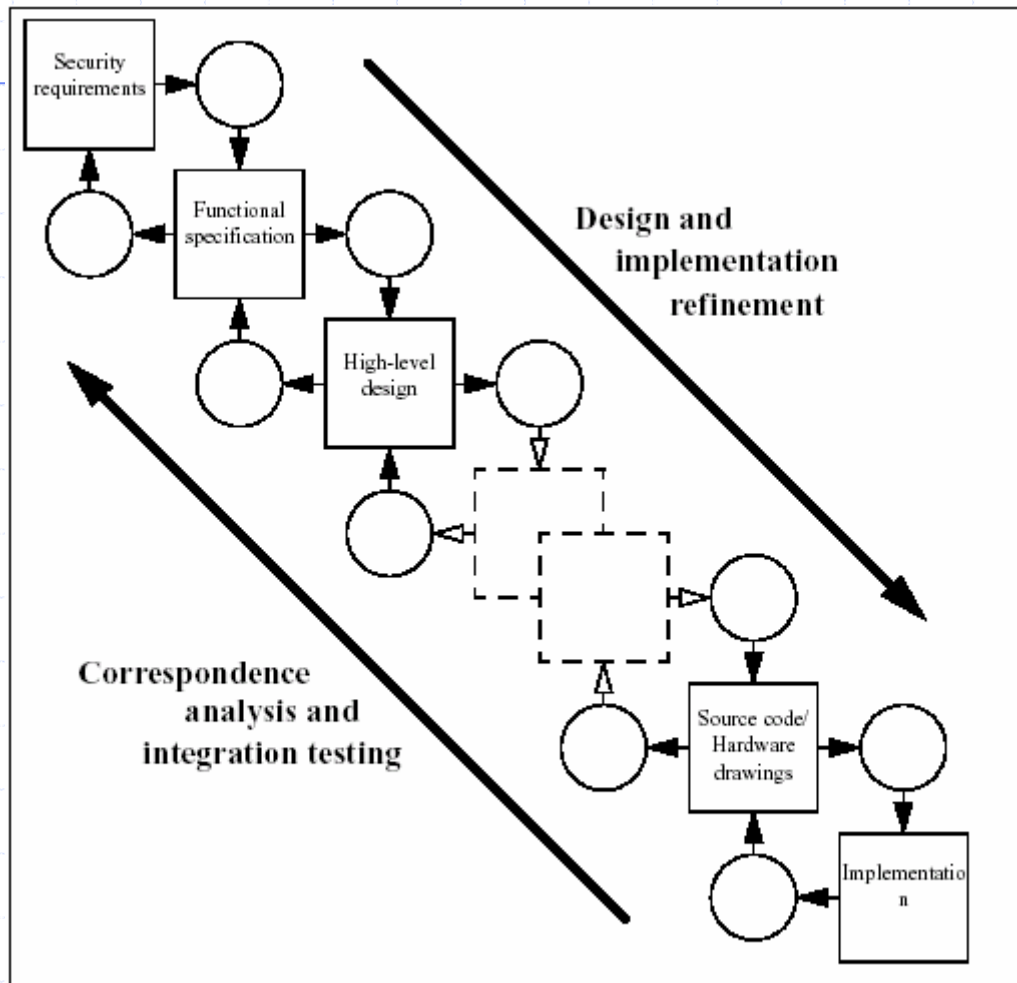


Figure 4.2 - Evaluation concepts and relationships

# L'approccio dei CC



Il progetto di sicurezza di un sistema di servizi



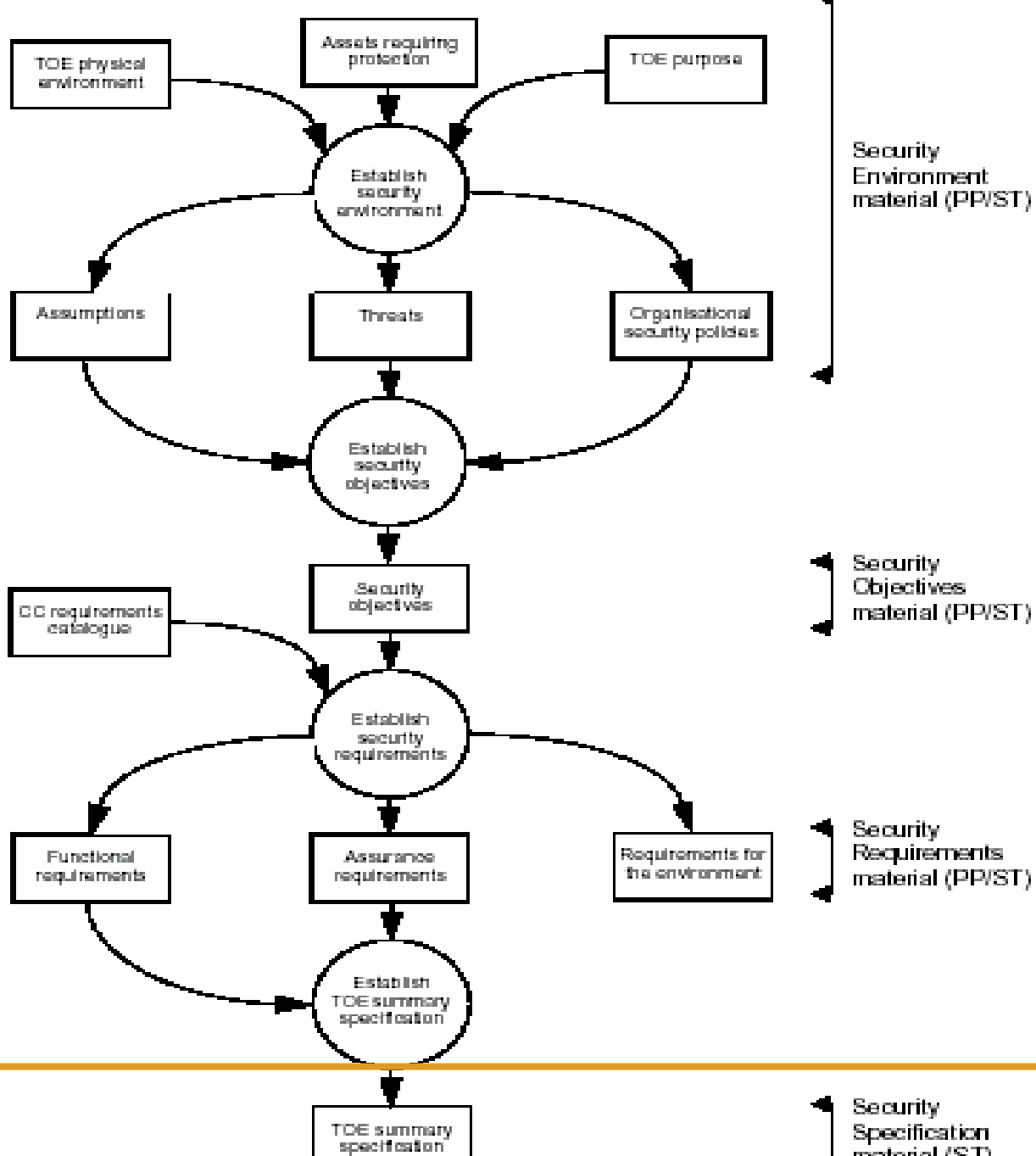
Il concetto di sicurezza deve essere sempre presente durante l'intero ciclo di vita di produzione del sw e deve avere dei continui feedback dalle varie fasi previste.

Il raggiungimento della sicurezza è un processo continuo che si affina ed arricchisce durante il processo di produzione del sw

Figure 4.3 - TOE development model



Il progetto di sicurezza di sistema di servizio



# Il processo di definizione del TOE

# Autenticazione del sw



Il progetto di  
sicurezza di  
sistema di ser

- È il processo che assicura ad un utente che il server erogatore del servizio richiesto è proprio quello autorizzato ad erogarlo
- L'autenticazione del richiedente abbinata a quella del software prende il nome di **mutua autenticazione**.

Se è prevista l'installazione di applet o activeX, accertarsi che tale sw sia digitalmente firmato e che il firmatario sia attendibile.



# Proprietà di una applicazione sicura

Quattro sono le caratteristiche principali da soddisfare affinché una applicazione si possa definire **SICURA**

- **Riservatezza:** *la possibilità di fornire le informazioni solo e soltanto agli utenti che ne hanno diritto*
- **Integrità:** *la garanzia che l'informazione non venga manipolata durante la sua registrazione sulla BD e/o distribuzione*
- **Disponibilità:** *la possibilità di fornire le informazioni solo e soltanto quando servono evitando attacchi di che non consentono l'erogazione del servizio (denial service)*
- **Non ripudio:** *in caso di transazioni in rete, l'originario non può negare la paternità della stessa*



# Tecniche per una applicazione sicura

Sono essenzialmente quattro

- **Identificazione**
- **Autenticazione**
- **Controllo degli accessi**
- **Tracciabilità (user accountability)**



# Identificazione

L'identificazione è il processo attraverso il quale una risorsa dichiara la propria identità nell'ambito di un sistema o di un'applicazione. Tale identità nel caso di risorse di sistema quali host, stampanti, router può essere espressa in diversi modi in funzione del tipo di applicazione di riferimento (indirizzo IP, URL, MAC Address, Fully Qualified Domain Name).

Nel caso persone è espressa attraverso uno USERID e specificata in un campo "login name" o "nome utente". Viene solitamente richiesta ogni volta che l'utente vuole accedere ad una risorsa gestita dal sistema o dall'applicazione, e viene assegnata all'utente quando lo stesso viene per la prima volta registrato nel sistema o tra gli utenti dell'applicazione.



# Identificazione: anonymous

In molte situazioni è previsto l'utente "anonymous", per consentire ad utenti non registrati di poter comunque accedere a parte delle risorse del sistema.

Potenzialmente una persona può avere diverse identità in funzione della risorsa a cui accede, anche se, almeno all'interno di un circuito chiuso, sarebbe preferibile, anche per motivi di tracciabilità, assegnare lo stesso USERID allo stesso utente



# Autenticazione

L'autenticazione è il processo attraverso cui, in una comunicazione tra due parti (uomo-macchina, macchina-macchina, uomo-applicazione, applicazione-macchina, ecc...) una parte verifica la veridicità dell'identità conclamata dall'altra parte. In alcune situazioni può essere necessario che tale verifica d'identità sia reciproca. In questi casi si parla di **mutua autenticazione**. Il processo di autenticazione viene svolto basandosi su alcune proprietà che si conoscono essere possedute dalla controparte.

Tipicamente i parametri utilizzati per svolgere tale fase sono:

- un segreto condiviso tra le due parti;
- un particolare oggetto in possesso della parte da autenticare;
- una caratteristica personale in possesso della parte da autenticare.

Sistemi di autenticazione basati sul primo parametro sono chiamati sistemi di autenticazione debole. I sistemi invece che utilizzano almeno uno degli altri parametri sono chiamati sistemi di autenticazione forte.



# Autenticazione debole

Questo è il caso più generale e riconducibile all'uso tecnica something you know -**SYK** come per esempio password o codice di accesso. Tra le due entità viene stabilito un segreto, la password, che si assume essere nota solo alle due parti in causa. Ogni parte potrà quindi facilmente identificare l'altra richiedendo alla stessa l'esibizione del segreto condiviso.

Quando un utente di rete richiede l'accesso ad un calcolatore remoto deve essere preventivamente autenticato e la sua password viene inviata via rete al calcolatore remoto che provvede a verificarne la validità. Questo meccanismo ha incentivato la realizzazione di programmi che consentono ad un intrusore di intercettare tali password e quindi utilizzarle per accedere abusivamente alle risorse di rete sotto false sembianze.



# Autenticazione debole: la pwd

È stato accertato che gli utenti costretti ad utilizzare le password per accedere a risorse di calcolo, scelgono password estremamente facili da indovinare. A tale proposito esistono programmi che forniti di appositi dizionari tentano di indovinare le password degli utenti dei sistemi (riuscita del 20%)

Esistono alcune tecniche per limitare i danni, ad esempio:

- limitare il numero dei tentativi di utilizzo di quella password (bancomat);
- assegnare alla password un periodo temporale di validità;
- imporre una lunghezza minima o comunque dei vincoli nel formato della password (almeno sette cifre non tutte numeriche, etc...);
- one-time password;
- assegnare una password, anziché farla scegliere all'utente;
- crittografare la pwd durante la sua trasmissione e/o registrazione su file.



# Autenticazione forte

I meccanismi di autenticazione forte sono basati principalmente su

- un oggetto in possesso dell'utente ( in genere una smart-card) - tecnica something you have –**SYH** o dei token crittografici,
- una caratteristica personale (come per esempio una impronta biometrica) - tecnica something you are –**SYA**,
- uso di sistemi di **crittografia asimmetrica**



# Autenticazione forte: crittografia asimmetrica

Il processo si basa sulla presenza di una coppia di chiavi (pubblica e privata); le informazioni che l'utente invia all'applicazione consentono di stabilire che solo l'utente poteva generarle (autenticità del mittente) in quanto unico possessore della chiave privata; le informazioni di autenticazione sono utilizzabili una sola volta.

Questa forma di autenticazione garantisce il non ripudio e prevede la presenza di una *trusted part* che garantisce l'abbinamento *identità della persona / chiave pubblica assegnata* attraverso il rilascio di un certificato digitale. La chiave privata deve essere gelosamente custodita dal suo legittimo possessore. Gli strumenti che consentono la custodia sicura della chiave privata sono dei supporti detti token crittografici. Questi dispositivi consentono all'utente di utilizzare la chiave privata attraverso una preventiva identificazione tramite l'introduzione del PIN. Principali implementazioni: PEM, SSL PGP, firma digitale.



# Token crittografici: la smart card

La smart card è uno strumento relativamente recente e tecnologicamente giovane regolato dalla suite ISO/IEC 7816.

Le componenti che consentono l'uso della smart card sono:

- porta seriale o usb (o altra tipologia di porta);
- Il lettore (reader) di smart card;
- Il token fisico (smart card).



# Smart card: le componenti fisiche

Una smart card crittografica è composta dai seguenti elementi:

- CPU: unità centrale;
- OS : sistema operativo che risiede su ROM;
- ROM: memoria a sola lettura ove risiede il sistema operativo;
- EPROM: memoria a lettura/scrittura equiparabile ad una memoria di massa;
- RAM: memoria a lettura/scrittura per i soli dati in uso dalla CPU;
- Hw-SEC: componente hw per la gestione dei meccanismi di sicurezza;
- CRY-PROC: co-processore crittografico per i processi di crittografia simmetrica ed asimmetrica



# Smart card: il file system

I dati sulla smart card sono allocati in un vero e proprio file system gerarchico rappresentato nel seguente modo:

- Master File (MF - root directory);
- Dedicated File (DF - directory);
- Elementary File (EF - file).



# Smart card: interoperabilità

Data la recente disponibilità di tali strumenti, comparando il funzionamento dei reader e delle smart card di produttori diversi, si osserva che vi sono significative differenze tra:

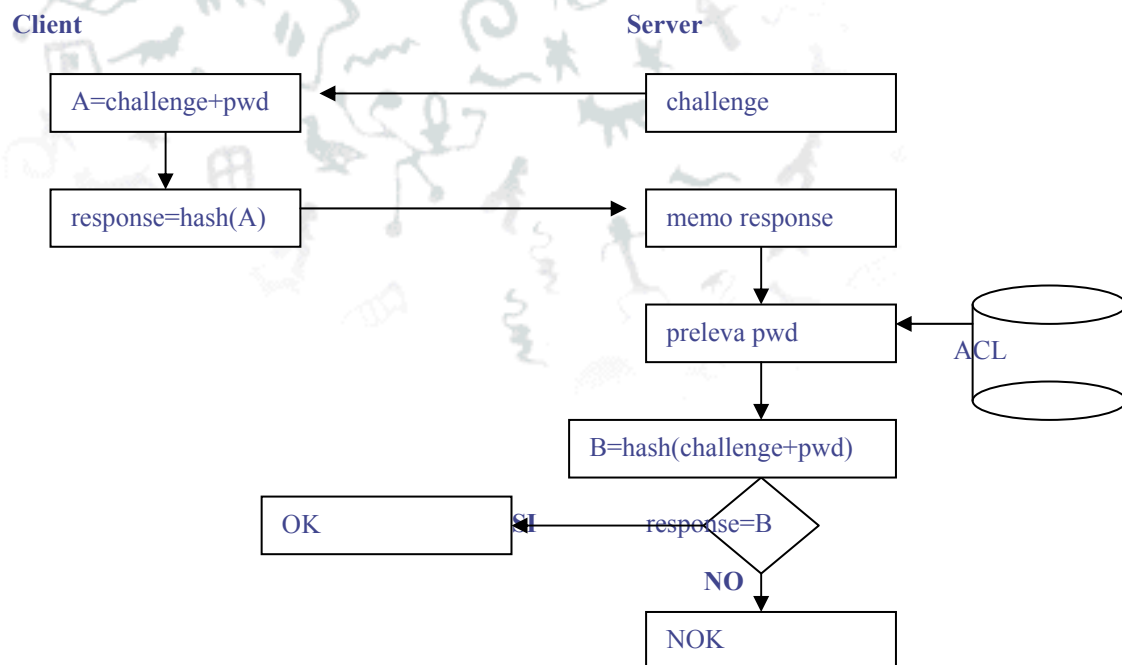
- i firmware dei produttori dei reader di smart card;
- il nome dei file e la forma dei dati contenuti sulla smart card sono fortemente dipendenti dal fornitore della stessa;
- i comandi di gestione della smart card non sempre hanno lo stesso nome ed il medesimo passaggio dei parametri;
- i codici di ritorno che indicano l'esito delle operazioni eseguite sulla smart card non sempre sono coincidenti.



Il progetto di sicurezza di sistema di ser

# Autenticazione digest

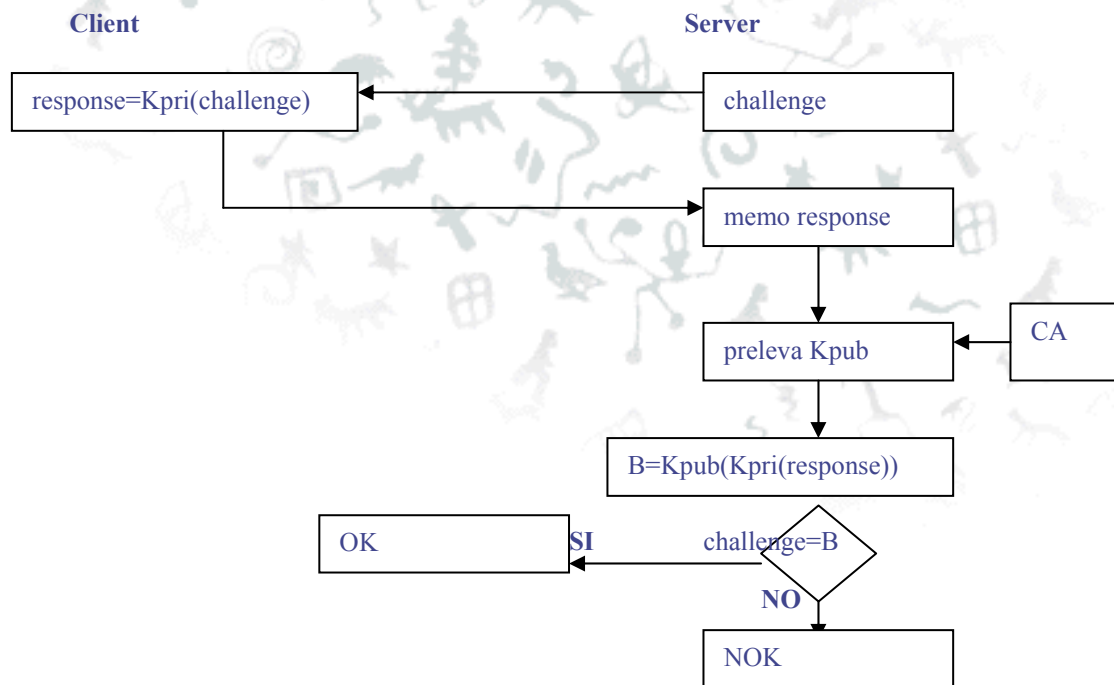
L'idea che sta alla base dell'autenticazione digest o interrogazione-risposta o challenge-response è che il server spedisce una stringa (challenge) al client. Il client unisce alla stringa la sua password e ne calcola l'impronta, rispedito il tutto al server. Il server preleva la password del client dal suo DB ACL e calcola la stessa impronta. Se il confronto tra le due impronte è positivo, allora la password è una password valida.





# Autenticazione digest con CA

Utilizzando la possibilità di una struttura a chiave pubblica, si può ottenere una autenticazione digest che non prevede l'utilizzo di password, ma di chiavi pubbliche.



# Autenticazione Kerberos



Il progetto di sicurezza di sistema di ser

Si tratta di una autorizzazione basata su una terza parte di fiducia: la **Key Distribution Center -KDC** - che controlla le identità di chi richiede il servizio (client) e di chi lo eroga (server). Le risorse, utenti o applicazioni, che condividono con il KDC la chiave di crittografia vengono detti principali. Si possono avere KDC che si basano su crittografia a chiave privata o pubblica.

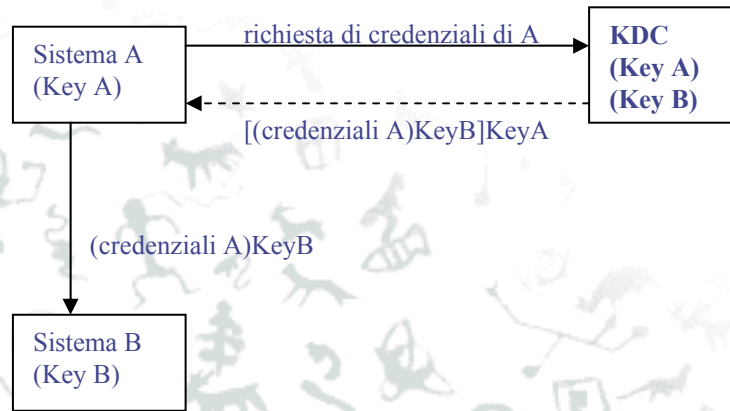
Il colloquio tra un sistema A ed un sistema B basato sulla tecnica KDC si può schematizzare come segue:



# Autenticazione Kerberos



Il progetto di sicurezza di sistema di ser



Il sistema A che vuole colloquiare con il sistema B, chiede al KDC le proprie credenziali (*ticket*). Il KDC invia ad A una stringa crittografata prima con la chiave di B e quindi con quella di A ((credenziali A)keyB)keyA). Il sistema A, che possiede la propria chiave, decripta il messaggio ottenendo il suo ticket criptato + la chiave di B ((credenziali A)keyB) e lo spedisce al sistema B. Il sistema B, che possiede la propria chiave, decripta la stringa arrivata, ottenendo le credenziali di A.



# Autenticazione Kerberos

Il server che fornisce i servizi di KDC è detto **Kerberos**, dal personaggio della mitologia greca Cerberus, cane a tre teste a guardia dell'ingresso degli inferi. In verità un server Kerberos ingloba due funzioni: la Authentication server e quella di Ticket granting server (TGS) preposte rispettivamente all'autenticazione vera e propria e al calcolo del ticket da assegnare. La struttura del ticket è:

$$\text{ticket}_{\text{client,server}} = \text{client, server, address, timestamp, lifetime, key}_{\text{client,server}}, \text{key}_{\text{server}}$$

- **client** = il principal name del client,
- **server** = il primary name del server,
- **address** = indirizzo della rete da cui proviene la richiesta di ticket,
- **timestamp** = ora in cui Kerberos emette il ticket,
- **lifetime** = il periodo di validità del ticket,
- **key**<sub>client,server</sub> = session key crittografica casuale generata da Kerberos, che il client e il server possono utilizzare per crittografare (simmetricamente) i dati durante la validità del ticket,
- **key**<sub>server</sub> = chiave di crittografia del server assegnata da Kerberos. Il fatto che il ticket sia crittografato con tale chiave, assicura che solo il server può leggere le credenziali del client contenute al suo interno.



# Controllo degli accessi

Per controllo degli accessi, o più precisamente sistema per il controllo degli accessi, si intende l'insieme di meccanismi che garantiscono che le entità che accedono a delle risorse in un sistema lo facciano nel rispetto di una serie di regole predefinite.

Il controllo degli accessi può essere svolto a livello di sistema operativo oppure a livello di applicazione per regolamentare l'accesso ai dati.



# Controllo degli accessi: componenti

Nella definizione di un sistema di controllo degli accessi distinguiamo diverse componenti:

- i soggetti cioè le entità attive del sistema, quelle che svolgono le operazioni, tipicamente utenti e processi
- gli oggetti cioè le entità su cui vengono svolte le operazioni, tipicamente file, data base, stampanti, ecc...
- le operazioni che i soggetti possono svolgere sugli oggetti come “read”, “write”, “execute”, “run”, ecc...
- per ogni oggetto esiste un soggetto speciale, il proprietario, che gode di particolari diritti sull’oggetto stesso (administrator).



# Controllo degli accessi: gli std di accesso

I controlli che il sistema deve effettuare vengono espressi attraverso opportune regole di controllo degli accessi che definiscono per ogni coppia soggetto-oggetto le possibili operazioni che possono essere svolte.

Esistono essenzialmente due diversi approcci per la definizione di tali regole:

- DAC (Discretionary Access Control);
- MAC (Mandatory Access Control).

DAC e MAC sono stati definiti per la prima volta nell'Orange Book (TCSEC). DAC è sostanzialmente il sistema di riferimento per la definizione di regole per il controllo degli accessi da utilizzare in applicazioni i cui requisiti di sicurezza sono quelli dettati dal mondo commerciale o civile; mentre MAC trova la sua applicazione essenzialmente in applicazioni di tipo militare.



# Controllo degli accessi: orange book

Nel 1985 il Department Of Defense degli Stati Uniti pubblica lo standard noto come Orange Book o Trusted Computer Security Evaluation Criteria - TCSEC - avente l'obiettivo di stabilire i criteri per la valutazione del livello di sicurezza dei sistemi protetti. DOD (USA Dept. Of Defense) Trusted Computing System Evaluation Criteria, DOD5200.28-std, 1985. Il TCSEC si basa su un insieme di elementi di sicurezza e di elementi rilevanti ai fini della sicurezza. Questo insieme è noto come **Trusted Computer Based - TCB** .

Prevede che la sicurezza venga affrontata in termini di obiettivi da prefiggersi e di strumenti da utilizzare per il conseguimento degli obiettivi. Più specificatamente, si prevedono:

- **politiche di sicurezza,**
- **gestione della sicurezza,**
- **documentazione a corredo della sicurezza.**



# Controllo degli accessi: TCSEC

## Politiche di sicurezza

1. Discretionary Access Control -DAC
2. Mandatory Access Control -MAC
3. Security level
4. Riutilizzo oggetti

## Gestione (meccanismi) di sicurezza

1. Identificazione
2. Autenticazione
3. Controllo degli accessi
4. Percorso protetto

## Documentazione di sicurezza

1. Manuale dell'operatore
2. Manuale dell'utente
3. Documentazione di testing
4. Documentazione di progettazione



# TCSEC: DAC

Con il Discretionary Access Control il proprietario di un oggetto determina chi vi può accedere e quali operazioni può svolgere sull'oggetto stesso. Il DAC viene usato in tutti quei sistemi in cui i dati possono essere facilmente ricondotti ad un proprietario, ed esiste la necessità di condividere o scambiarsi tali dati con altri utenti, senza dover chiedere un'autorizzazione all'amministratore del sistema o della base di dati. Questo approccio, non sempre risponde completamente alle esigenze di un'organizzazione, che si trova sovente nella situazione di poter applicare un controllo degli accessi DAC solo ad un sottoinsieme dei propri dati. In questi casi si ricorre a sistemi misti.

Un sistema di controllo degli accessi basato sull'approccio DAC può essere anche gestito centralmente in questo caso l'amministratore del sistema deve diventare proprietario di tutti i dati presenti nel sistema e determina quali operazioni assegnare a ciascun utente.



# DAC: le ACL

L'approccio più diffuso per l'implementazione di un sistema DAC è quello che usa le ACL (Access Control List). Esiste una ACL, per ogni oggetto di un sistema, che specifica tutti i soggetti abilitati all'accesso a questo oggetto e le operazioni che vi possono svolgere.

La maggior parte dei DBMS commerciali consente oggi di definire, associate con i vari schemi della base di dati, delle tabelle per il controllo degli accessi che si rifanno all'approccio DAC.



# TCSEC: MAC

Nell'approccio MAC al controllo degli accessi, le decisioni su chi accede ad un oggetto e sul tipo di operazioni che può compiere, non vengono prese dal proprietario dell'oggetto stesso ma da un'autorità centrale che supervisiona l'intero funzionamento del sistema. Ovviamente tale autorità è anche l'unica che può modificare i diritti di accesso dei vari utenti.

I diritti di accesso assegnati ad un utente vengono solitamente dettati da due parametri:

- il livello di affidabilità dell'utente nell'ambito dell'organizzazione;
- il livello di sensibilità dei dati coinvolti.



# MAC: la classificazione

È necessario aver definito una gerarchia tra gli utenti di un'organizzazione e soprattutto una classificazione dei dati. Le classificazioni più utilizzate in quest'ultimo caso sono a 5 livelli per sistemi militari e a 3 livelli per sistemi civili. Nel primo caso i dati vengono suddivisi in: *non classificati*, *classificati*, *confidenziali*, *segreti*, *top-secret*. Nel secondo: *pubblici*, *proprietary* e *interni*.

Il sistema MAC trova la sua naturale applicazione negli ambienti militari dove nessun proprietario di dati può permettersi di classificare gli stessi top-secret o modificare la classificazione di un dato da segreto a confidenziale. Non è comunque da escludere una sua adozione anche in particolari applicazioni civili. Va comunque sottolineato che DAC e MAC non sono tra loro mutuamente esclusivi e possono essere combinati. In questo caso MAC ha la precedenza. Più precisamente quando un utente vuole accedere ad un oggetto si verifica in prima istanza se l'utente soddisfa i requisiti imposti dal MAC e successivamente vengono applicate le regole imposte dal DAC.



# MAC: i livelli di sicurezza

Livello	Sub	Descrizione
<b>D</b>		<i>Protezione minima.</i> Non sono previste misure per la garanzia della sicurezza
<b>C</b>		<i>Protezione facoltativa.</i> Supporta la politica di protezione DAC (autorizzazione facoltativa) e le funzioni di riutilizzo degli oggetti. Sono gestite le funzioni di identificazione, autenticazione e controllo degli accessi.
	<b>C1</b>	La politica DAC e le funzioni di identificazione, autenticazione e controllo degli accessi sono risolte a livello di gruppo e non di singolo utente. Nulla è previsto a livello di singolo utente.
	<b>C2</b>	Si estende il livello C1 al singolo utente. È il livello minimo per un sistema informativo che gestisce informazioni importanti.
<b>B</b>		<i>Protezione mandatory.</i> Supporta la politica di protezione MAC basata sui livelli di segretezza.
	<b>B1</b>	= C2 + MAC + livelli di segretezza. È possibile trasformare in B1 un qualsiasi sistema informatico esistente .
	<b>B2</b>	Un sistema B2 deve essere progettato come tale dall'inizio, quindi non sono possibili trasformazioni di sistemi appartenenti a fasce più basse. Esso prevede la separazione tra gli utenti e gli amministratori della sicurezza, l'utilizzo di percorsi sicuri e l'analisi di canali occulti
	<b>B3</b>	Introduce il concetto di controllo della sicurezza attraverso strumenti che in tempo reale intercettano il tentativo di attacco (per esempio attraverso l'emissione di un suono di allarme).
<b>A</b>		Verifica dei controlli di sicurezza tramite il modello formale previsto dalla politica di sicurezza adottata.
	<b>A1</b>	prevede l'uso di tecniche formali di controllo tra le specifiche e la politica di sicurezza adottata.



# ITSEC: la versione europea

L'approccio ITSEC prevede che sistemi e prodotti siano valutati secondo gli stessi criteri. In ITSEC, l'oggetto (sistema o prodotto) della valutazione è detto **Target Of Evaluation (TOE)** mentre chi richiede la valutazione (persona od organizzazione) è detto **sponsor** o committente. La valutazione di un TOE viene fatta rispetto al **security target (ST)** che è un documento costituito dalle seguenti specifiche:

1. una **System Security Policy -SSP-** (nel caso dei sistemi) o un **Product Rationale** (nel caso dei prodotti). La SSP costituisce l'insieme delle leggi regole e pratiche che stabiliscono come le informazioni e le risorse critiche per la sicurezza devono essere gestite, protette e distribuite all'interno del sistema. La system security policy deve coprire tutti gli aspetti di sicurezza del sistema incluse le misure di sicurezza di tipo fisico, procedurale e sul personale. Il product rationale, invece, deve includere una descrizione delle caratteristiche di sicurezza del prodotto e deve precisare le assunzioni sull'ambiente, sulle minacce e sulle modalità d'uso. Inoltre, dovrà indicare tutte le misure di sicurezza tecniche e non tecniche necessarie al corretto funzionamento del prodotto e le sue dipendenze dall'hardware, dal software e dal firmware di sistema non forniti dal prodotto stesso;
2. una specifica delle **funzioni** di sicurezza (**Security Enforcing Functions** o **SEF**) che permettono il conseguimento degli obiettivi individuati;
3. la definizione dei **meccanismi** di sicurezza richiesti (opzionale);
4. il livello minimo dichiarato di **robustezza dei meccanismi** (strength of mechanisms);
5. il livello di valutazione desiderato.

# Evoluzione: i CC



Il progetto di sicurezza di sistema di ser

È lo standard europeo che intende unificare ed omogeneizzare i precedenti standard (americani TCSEC; europei, ITSEC e canadesi CTCPEC). Anche qui esistono concetti di TOE, ST e security function e di *grado di fiducia* che viene detto **Evaluation Assurance Level (EAL)**. Viene quindi introdotto il concetto di **Protection Profile (PP)** come un insieme di requisiti di sicurezza, indipendenti dal tipo di realizzazione, e che caratterizza particolari TOE definiti dal mercato o da esigenze dell'utenza.

Le funzioni di sicurezza, raggruppabili in classi, previste sono:

- identificazione ed autenticazione,
- controllo degli accessi,
- imputabilità,
- verifica,
- riutilizzo dell'oggetto,
- fedeltà (accuracy),
- affidabilità del servizio,
- trasmissione dei dati.

Gli EAL sono sette:

- EAL1: functionally tested,
- EAL2: structurally tested,
- EAL3: methodically tested and checked,
- EAL4: methodically designed, tested and reviewed,
- EAL5: semiformally designed, and tested,
- EAL6: semiformally verified design, and tested,
- EAL7: formally designed, and tested,

# Logging



Il progetto di sicurezza di sistema di ser

Per tracciabilità degli utenti s'intende l'insieme di meccanismi adottati per poter ricondurre inequivocabilmente ad tempo ben individuato ed a un utente l'esecuzione di una certa azione, e quindi poter attribuire ad ogni singolo utente le proprie responsabilità.

Adottare un sistema che garantisce la tracciabilità di utente consente di raggiungere due obiettivi molto importanti:

- rilevare in tempo debito un'intrusione;
- prevenire azioni non autorizzate da parte di interni grazie all'effetto scoraggiante che un tale sistema può provocare sugli stessi.

# Logging



Il progetto di  
sicurezza di un  
sistema di ser

Le misure da intraprendere per poter disporre di un sistema per la tracciabilità degli utenti sono:

- adottare sistemi di identificazione/autenticazione che garantiscano l'inequivocabile identità dell'utente e l'impossibilità di praticare tecniche di impersonificazione;
- adottare adeguate politiche di controllo d'accessi;
- generare per ogni evento critico riconducibile a problemi di sicurezza, una registrazione (log) su un opportuno database, che deve essere messo al riparo da possibili danneggiamenti o manomissioni. Il data base conterrà, quindi, una storia (chiamata anche audit trail) di tutti gli eventi critici avvenuti nel sistema;
- verificare periodicamente il contenuto del database al fine di individuarne eventi anomali. Tale fase può essere attualmente svolta automaticamente con l'ausilio di tool per l'analisi dei log.

# Logging: le criticità



Il progetto di  
sicurezza di un  
sistema di ser

Un grosso limite delle tecniche attualmente utilizzate per effettuare la tracciabilità degli utenti è legato alla loro scarsa efficacia quando devono operare su applicazioni o ambienti distribuiti. In cui cioè le componenti utilizzate operano su più postazioni server geograficamente distribuite. In questo caso non è ancora chiaro quali siano effettivamente gli eventi critici da registrare, quale tipo di informazione collezionare e come confrontare e rendere omogenei i diversi tipi di informazione collezionati sui diversi sistemi. Tale processo è reso ulteriormente difficile dalla mancanza di un formato standard per la rappresentazione dei log e dalla difficoltà di poter rendere interoperabili file di log raccolti su piattaforme diverse.



# Sistemi distribuiti

Lo scenario cambia radicalmente se l'utente non vuole più autenticarsi ad un singolo sistema ma vuole autenticarsi ad una rete di calcolatori, o anche se l'applicazione che l'utente sta eseguendo deve accedere ad altri server. In questi casi i principali problemi che devono essere considerati sono:

- quando un utente possiede certi diritti di accesso ad un sistema quali credenziali deve presentare per poter essere “accettato” su un altro sistema collegato al primo
- la richiesta di accesso di un'applicazione ad un host remoto deve essere eseguita con i diritti di accesso dell'applicazione o dell'utente che sta eseguendo l'applicazione;
- sistemi diversi possono avere meccanismi diversi per la classificazione e protezione dei dati, come devono essere coordinate le attività di interscambio di dati fra gli stessi.



Il progetto di  
sicurezza di  
sistema di ser

# Virus e protezione

## Virus



## Glossario

