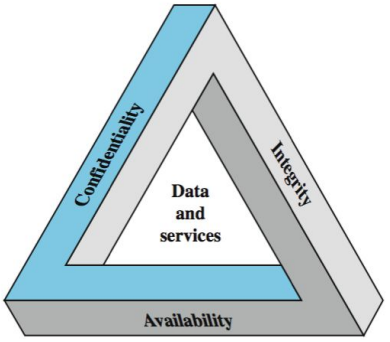
Sicurezza delle reti

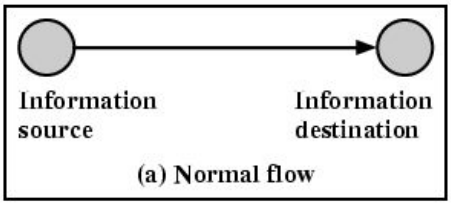
Introduzione

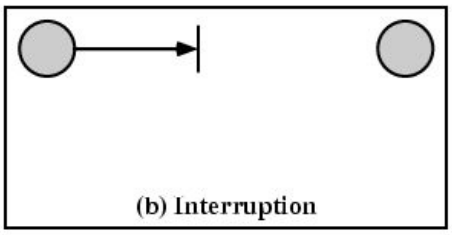
Per capire di cosa significhi sicurezza di rete, dobbiamo porci alcune **domande fondamentali**:

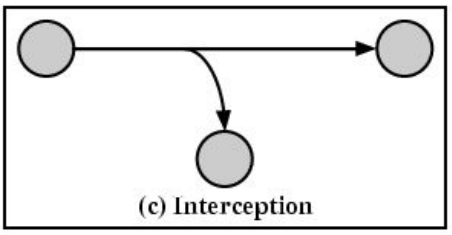
- ➔ Quali sono le **risorse** (o **asset**) che vogliamo proteggere?
  - ◆ **Hardware:** intesi come sistemi, componenti e dischi (parti “fisiche”).
  - ◆ **Software:** sistema operativo e software applicativo.
  - ◆ **Dati:** file e database.
  - ◆ **Rete:** collegamenti e apparati.
- ➔ Che cosa intendiamo per **protezione**? Significa **garantire** le **proprietà** di:
  - ◆ **Confidenzialità:**
    - Nessun utente deve poter ottenere o dedurre dal sistema informazioni che non è autorizzato a conoscere.
    - **Riservatezza dei dati:** Le informazioni confidenziali non devono essere rivelate o rilevabili da utenti non autorizzati.
    - **Privacy:** L’utente controlla o influenza quali informazioni possono essere collezionate e memorizzate.

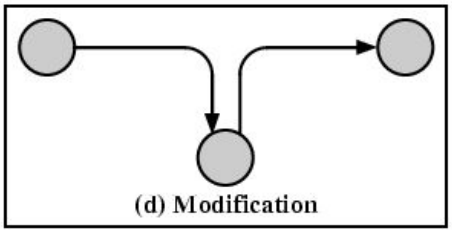


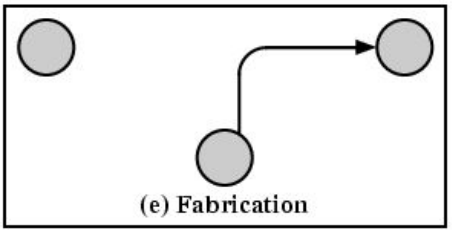
  - ◆ **Integrità:**
    - Impedire l’alterazione diretta o indiretta delle informazioni, sia da parte di utenti e processi non autorizzati, che a seguito di eventi accidentali (necessario poterlo verificare facilmente).
    - **Dei dati:** modifica di informazioni e programmi solo se si dispone delle autorizzazioni necessarie.
    - **Del sistema:** il sistema funziona correttamente e non è stato compromesso.
  - ◆ **Disponibilità:**
    - Rendere disponibile a ogni utente abilitato le informazioni a cui ha diritto di accedere nei modi e nei tempi e nei modi previsti.
    - In informatica sono incluse prestazioni e robustezza.
  - ◆ (Autenticità):
    - Ciascun utente deve poter verificare l’autenticità delle informazioni.
    - Si richiede di poter verificare se un'informazione è stata manipolata.
  - ◆ (Tracciabilità):
    - Le azioni devono essere tracciate in modo univoco, in modo tale da supportare la non-ripudiabilità e l’isolamento delle responsabilità.
- ➔ **In che modo** queste risorse sono **minacciate**?
  - ◆ Le minacce **compromettono** le **proprietà** di confidenzialità, integrità e disponibilità.
  - ◆ Una **minaccia** è una **possibile violazione** della sicurezza.
  - ◆ Un’ **attacco** è una **violazione effettiva** della sicurezza, che possono essere:
    - **Attivi:** tentativi di **alterazioni di funzionamento** e risorse di un sistema.
    - **Passivi:** tentativi di **carpire informazioni** senza alterare risorse.
    - **Interni:** iniziati da un'entità **interna** a un sistema.
    - **Esterni:** iniziati da un’entità **esterna** al sistema.
  - ◆ Alcuni esempi di attacchi:
  - ◆ Classi di minacce e attacchi:
    - **Disclosure:** accesso non autorizzato alle informazioni.
    - **Deception:** accettazione di dati falsi.
    - **Disruption:** interruzione o prevenzione di operazioni corrette.
    - **Usurpation:** controllo non autorizzato di alcune parti del sistema.











- ➔ Che cosa bisogna fare per **contrastare** queste minacce?
  - ◆ La complessità della sicurezza risiede proprio in questa domanda, in quanto non esiste un'unica risposta ed esse possono cambiare nel tempo.
  - ◆ **Sistemi complessi:** Le risorse da proteggere sono sistemi composti da sottosistemi.

La sicurezza pone delle **sfide**:

- ➔ **Attacchi potenziali:** durante la progettazione di un sistema occorre considerare i possibili attacchi allo stesso.
- ➔ **Soluzioni contro-intuitive:** nello sviluppo dei meccanismi di sicurezza stessi.
- ➔ Utilizzo di **sistemi di sicurezza** sia a livello **fisico** sia a livello **logico** (protocollare).
- ➔ La sicurezza **dipende** anche per una buona parte **dagli utenti:**
  - ◆ **Informazioni possedute** e creazione, distribuzione e protezione di tali informazioni.
- ➔ Continua **battaglia** tra **amministratori** e **attaccanti**:
  - ◆ Per un **attaccante** basta sfruttare una **singola vulnerabilità**, mentre un **amministratore** deve prevederle ed **eliminarle tutte**.

Principi fondamentali di progettazione della sicurezza:

- ➔ **Aspetti economici:** la soluzione deve essere il più semplice da implementare e verificare.
- ➔ **Fail-safe default:** i comportamenti non specificati devono prevedere un default sicuro.
- ➔ **Progettazione aperta (open source):** preferibile rispetto alla closed-source.
- ➔ **Tracciabilità delle impostazioni:** Qualsiasi operazione deve poter essere ricostruita e il sistema ripristinato.
- ➔ **Separazione dei privilegi:** differenziazione degli accessi alle risorse critiche e create da ciascun utente.
- ➔ **Separazione delle funzionalità:** distinzione dei ruoli nei diversi punti del sistema fisico e logico.
- ➔ **Isolamento dei sottosistemi:** un sistema compromesso non dovrebbe compromettere gli altri.
- ➔ **Modularità:** meccanismi di sicurezza indipendenti, sostituibili, riusabili.

Politiche di sicurezza: una politica di sicurezza è cosa è e non è permesso:

- ➔ Le regole possono riguardare utenti, operazioni e dati.
- ➔ Sono garantite da meccanismi di sicurezza:
  - ◆ **Prevenzione:** l'attacco deve essere reso impossibile.
    - Spesso pesanti e interferiscono con il sistema.
  - ◆ **Scoperta:** in grado di scoprire un attacco in corso.
    - Utile quando non è possibile prevenire un attacco, applicato usando il monitoraggio delle risorse.
  - ◆ **Recupero:**
    - Fermare l'attacco e ripristinare una situazione precedente ad esso.
    - Far funzionare il sistema correttamente durante l'attacco (fault-tolerant).

Livelli e **Meccanismi** di Sicurezza:

- A che livello inserire un meccanismo?
- **Livelli bassi:** Meccanismi **generali**, semplici, **grossolani**, ma **dimostrabili corretti**.
- **Livelli alti:** Meccanismi **ad hoc** per gli utenti, **sofisticati**, **difficili da dimostrare corretti**.

Come **ottenere** un **sistema sicuro**? Necessarie alcune fasi:

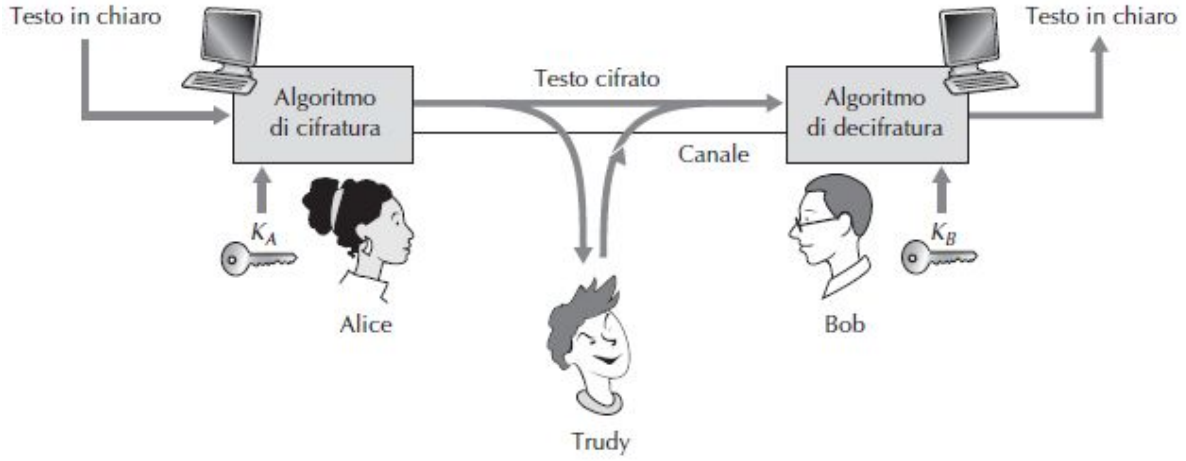
1. **Specifica:** descrizione del funzionamento desiderato del sistema.
2. **Progetto:** traduzione delle specifiche in componenti che le implementeranno.
3. **Implementazione:** creazione del sistema che soddisfa le specifiche.

⇒ Indispensabile verificare continuamente la correttezza dell'implementazione.

Crittografia

La crittografia è la scienza che si occupa di **proteggere l'informazione** rendendola sicura, in modo che un **utente non autorizzato** che ne entri in possesso **non sia in grado di comprenderla**.

Algoritmo crittografico: è una funzione che prende in **input** una **chiave** e produce in **output** un **messaggio trasformato**.



Algoritmo simmetrico	Algoritmo asimmetrico
● **Chiavi** di cifratura e decifratura **uguali**. ● La chiave deve essere **mantenuta segreta**.	● Le **chiavi** di cifratura e decifratura sono **diverse**, di cui una **pubblica**, e una **privata**.

Robustezza crittografica:

- **Non deve essere possibile facilmente:**
  - **Ottenere** il corrispondente **testo in chiaro senza** conoscere la **chiave di decifratura**.
  - **Dato** un **testo cifrato e** il corrispondente **testo in chiaro ottenere** la **chiave di cifratura**.
- Un **algoritmo** di dice **computazionalmente sicuro** se:
  - Il **costo necessario** per violare un' informazione è **superiore** al **costo dell'informazione** stessa.
  - Il **tempo necessario** per **violare** un' informazione è **superiore al tempo di vita utile dell'informazione** cifrata.

Crittoanalisi: tenta di ricostruire il testo in chiaro senza essere a conoscenza della chiave di cifratura.

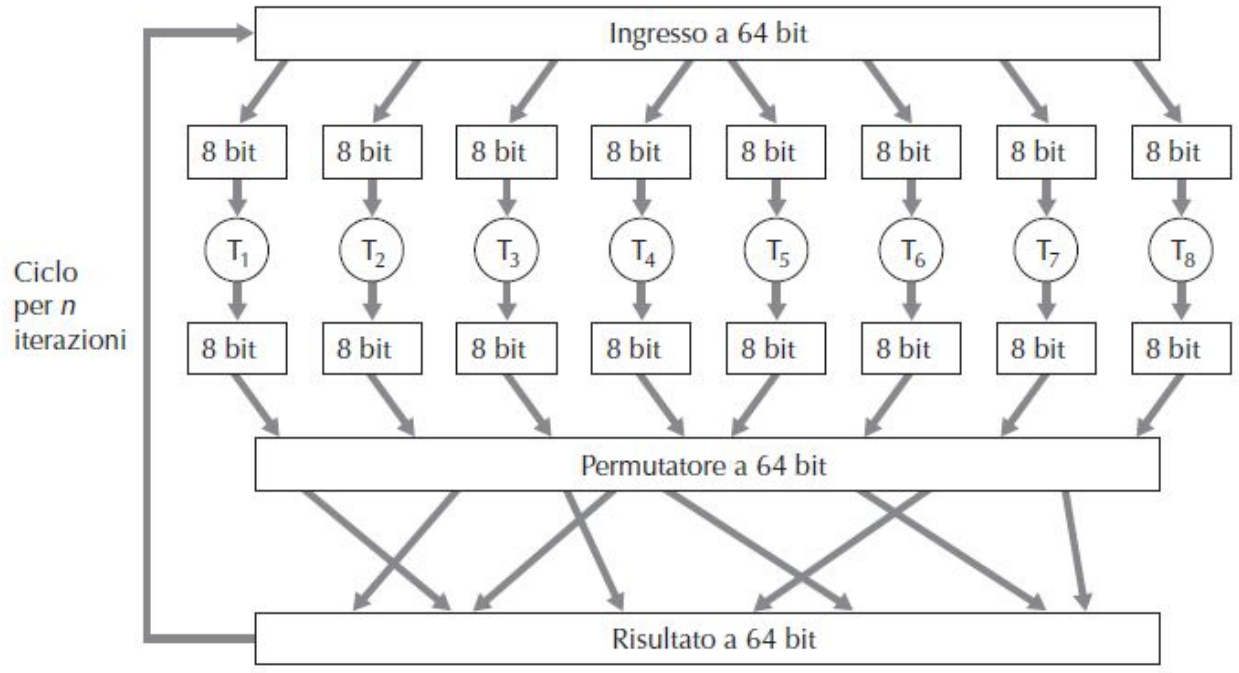
Cifratura Monoalfabetica

- Ogni carattere viene sostituito da un altro (**permutazione**), secondo un certo alfabeto che costituisce la chiave.
- Le **chiavi possibili sono pari al numero di permutazioni** possibili, 21! ovvero circa 5,1 x 10^{19}.

Analisi delle frequenze: in uno spazio di chiavi molto ampio, come l'alfabeto, ogni lettera si ripropone con una certa frequenza, contando il numero delle occorrenze nel testo cifrato è possibile ipotizzare con buona probabilità quale sia la lettera corrispondente.

Cifrari a Blocchi

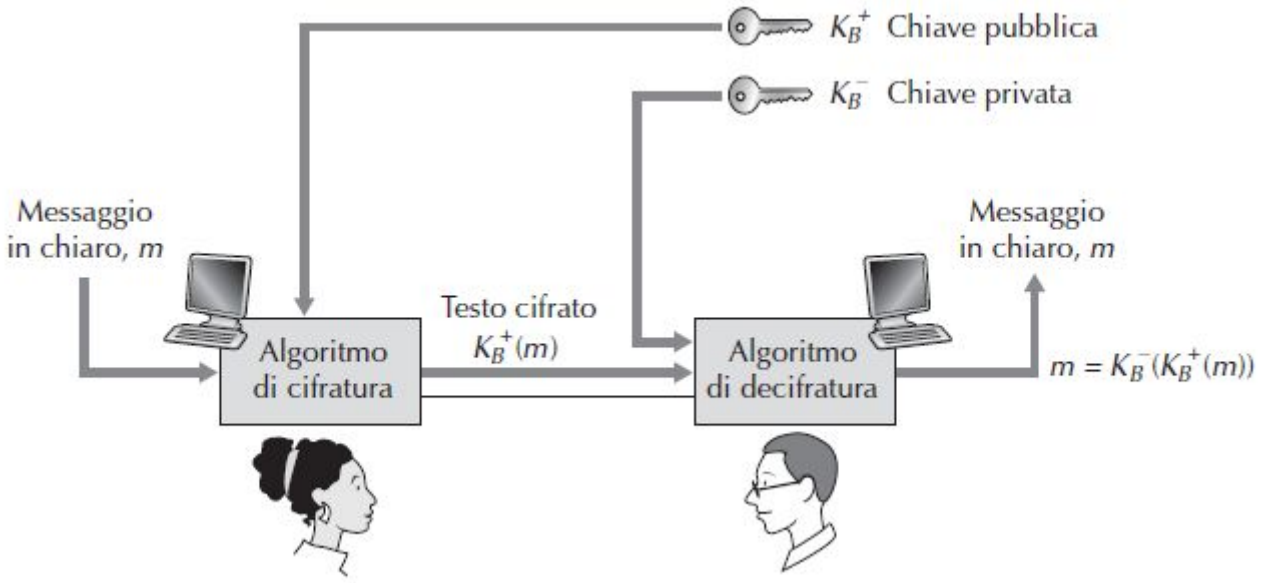
- È una tecnica di **cifratura simmetrica**, utilizzata in molti protocolli sicuri di Internet, come PGP, SSL e IPSec.
- Dati **k bit**, i possibili $\mathbf{2^k}$ ingressi vengono permutati.
    - Le **permutazioni** possono essere **combinate** per creare **schemi più complessi**.



- Esempi:
    - **DES** (**D**ata **E**ncryption **S**tandard):
        - È il più noto algoritmo crittografico simmetrico moderno, nato negli anni '70 a seguito di un **progetto di IBM**.
        - Chiavi da **56 bit** (ormai **obsoleto**).
    - **Triple-DES:**
        - Per aumentare la sicurezza del **DES** lo si applica **tre volte con chiavi diverse**.
        - Chiavi da **112 bit** (56*2) e da **168 bit** (56*3).
    - **AES (Advanced Encryption Standard):**
        - Algoritmo scelto come vincitore nel 2000 a un bando del NIST per trovare il **successore del DES**.
        - Può utilizzare **chiavi da 128, 192 e 256 bit**.
        - Sta gradualmente soppiantando il triple-DES.

Cifratura a chiave pubblica/privata

- Per superare il **problema di distribuzione delle chiavi**, è necessario un **canale sicuro** dove trasmetterle.
- Nella crittografia asimmetrica **ogni utente ha una coppia di chiavi**, costituite da una chiave **pubblica** e da una chiave **privata**.
- **Ogni dato cifrato con la chiave pubblica, può essere decifrato solamente con la corrispondente chiave privata, e viceversa.**

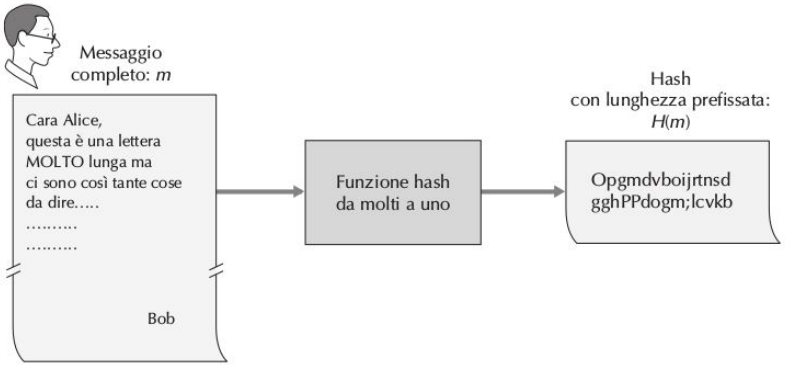


Crittografia asimmetrica

- Vantaggi:
  - **Non** è più **necessario incontrarsi** per **scambiarsi le chiavi**.
  - La **stessa chiave pubblica** può essere usata da **più utenti**.
- Requisiti:
  - Deve essere **semplice la generazione** di una coppia di **chiavi pubblica/privata**.
  - Deve essere **semplice l'operazione di cifratura e decifratura** se si è a conoscenza della relativa chiave.
  - Deve essere **computazionalmente impraticabile ricavare** la **chiave privata da** quella **pubblica**.
  - Deve essere **computazionalmente impraticabile ricavare** il **testo in chiaro avendo** il **testo cifrato** e la chiave pubblica.

Algoritmo RSA

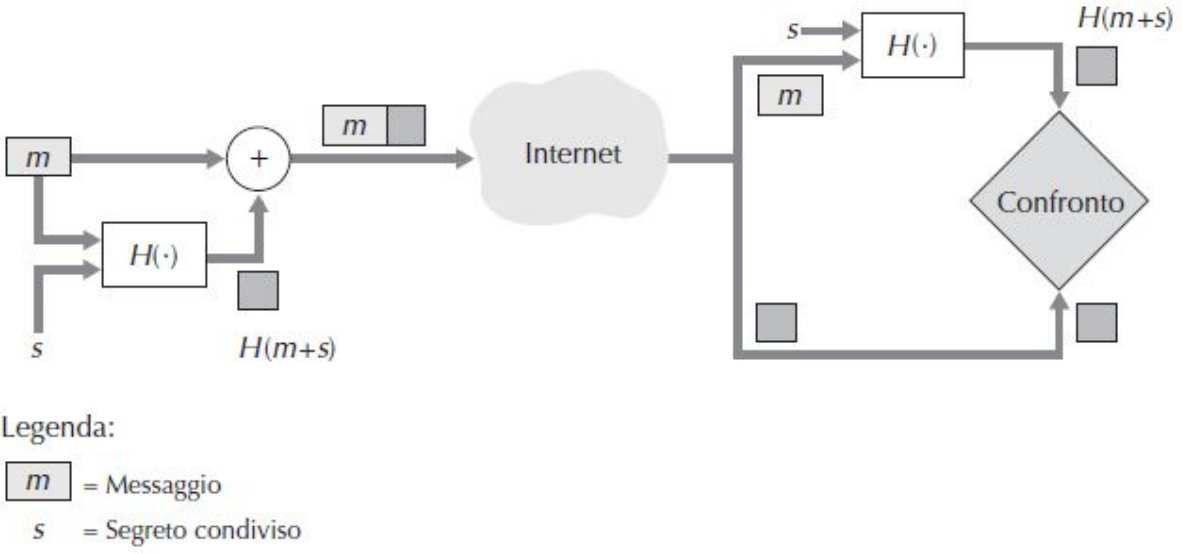
- Prende il nome dalle iniziali dei suoi inventori (**R**ivest, **S**hamir, **A**dleman).
- **Si basa** sulla **difficoltà** di **scomporre** un numero in **fattori primi**.
- La chiave in RSA ha di solito dimensioni di **almeno 2^{10} bit** (300 cifre decimali).
- Un attacco a forza bruta contro RSA non consiste nel provare tutte le chiavi possibili, ma nel fattorizzare il prodotto di due numeri primi.
- Per capire come avviene la cifratura e la decifratura con RSA ci si deve avvalere della matematica a modulo. Una volta scelti due numeri primi, si calcola:
  - $n = p * q$
  - $z = (p-1)*(q-1)$
  - un numero $1 < e < n$, relativamente primo a z
  - un numero d tale che $(e * d-1)$
- Chiave pubblica: (n, e)
  - Per cifrare $m \Rightarrow c = m^e \bmod n$
- Chiave privata: (n, d)
  - Per cifrare $c \Rightarrow m = c^d \bmod n$



Funzioni hash

- Una funzione hash **trasforma** un **messaggio** di lunghezza arbitraria **in output** di **lunghezza fissa.**

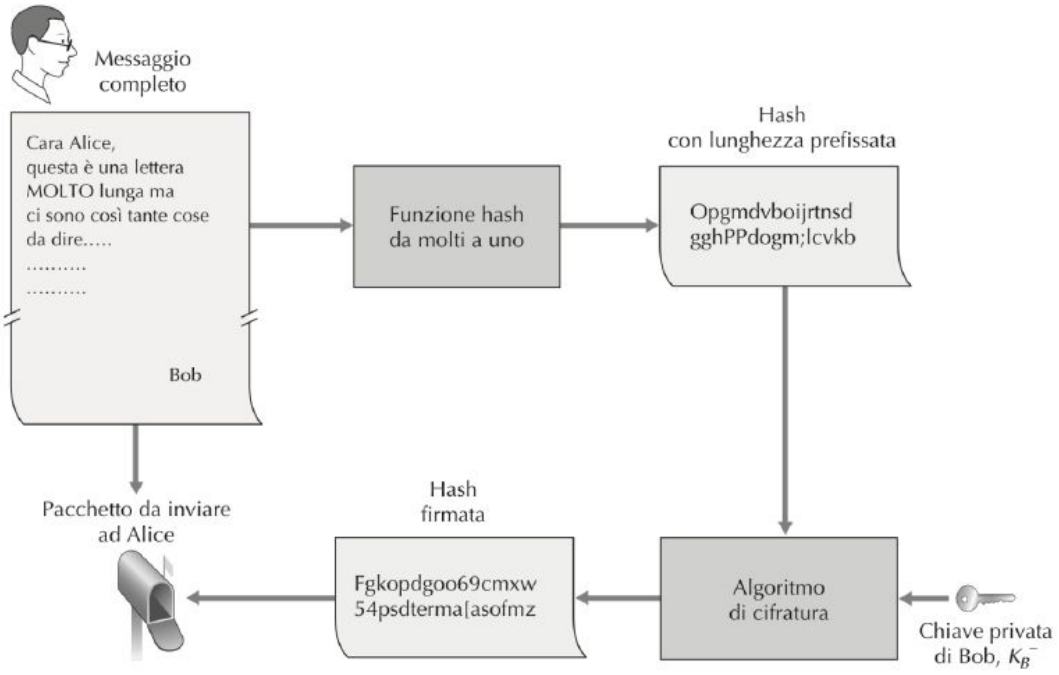
- Per soddisfare le condizioni di sicurezza stabilite per le funzioni hash, gli algoritmi devono essere:
  - **Coerenti:** a input uguali corrispondono output uguali.
  - **Casuali:** per impedire l'interpretazione accidentale del messaggio originale.
  - **Univoci:** la probabilità che due messaggi diversi generino lo stesso hash deve essere virtualmente nulla.
  - **Non invertibili:** risalire al messaggio originale a partire dall'hash deve essere impossibile.
- Le **funzioni hash non invertibili** vengono **normalmente utilizzate** per assegnare un' **impronta digitale** a un messaggio o a un file.
  - Costituisce una **prova di integrità** e **autenticità** del messaggio.
  - Utilizzato per assicurarsi che nessuno sia intervenuto sul contenuto del messaggio.
- **M**essage **A**uthentication **C**ode (**MAC**):
  - Utilizzato quando non si è interessati a occultare il contenuto del messaggio, ma solamente per preservarne l'integrità.



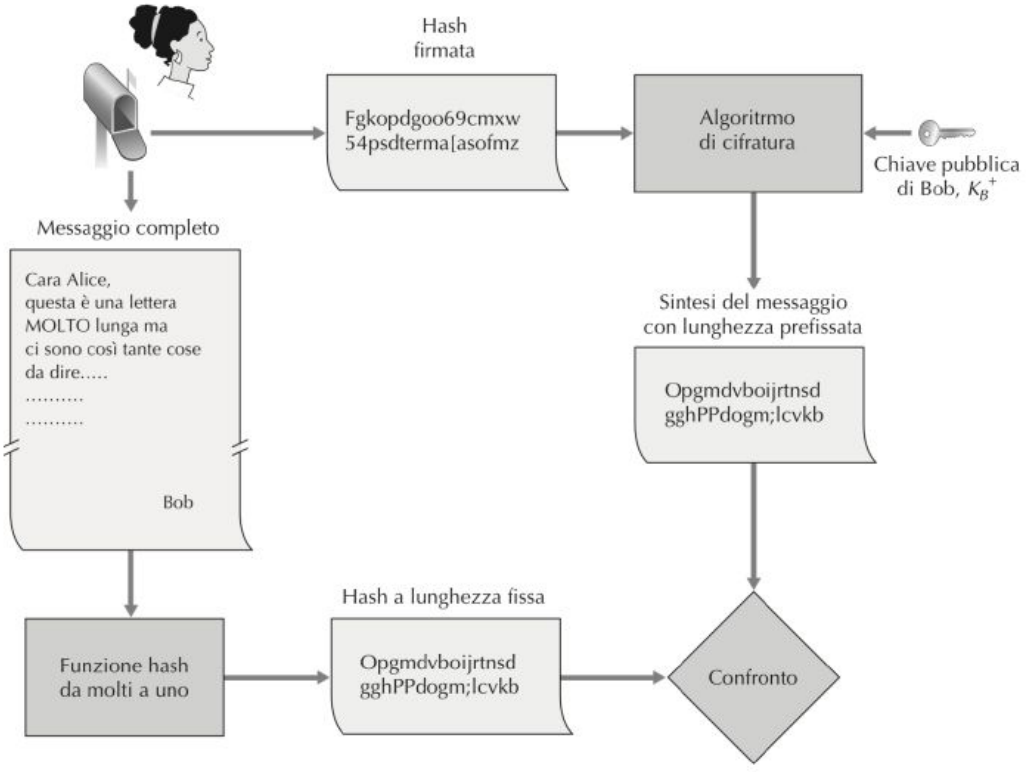
- Alcuni **esempi comuni**:
  - **MD5** (Message Digest 5)
  - **SHA** (Secure Hash Algorithm)

Firma digitale

- **Equivalente** informatico **di una firma convenzionale**, in generale **non** è **ripudiabile**.
- Viene sfruttato l'algoritmo RSA in modo inverso rispetto alla cifratura.
    - **Cifratura ⇒ Verifica**.
    - **Decifratura ⇒ Firma**.
- Viene usata normalmente una **combinazione di RSA e funzioni hash** per evitare di dover firmare l'intero documento, essendo molto oneroso.



Firma di un documento

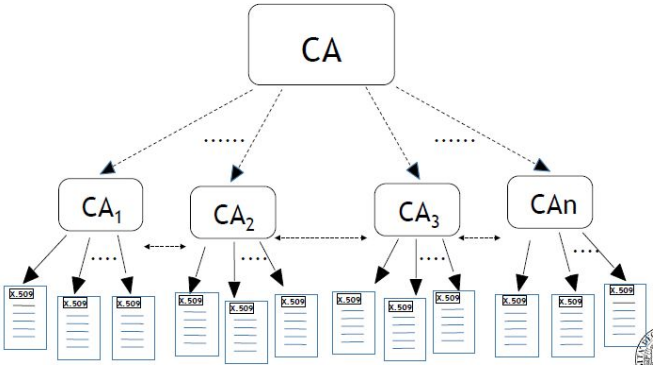


Controllo integrità del messaggio firmato

Autorità di certificazione (Certificate Authority o CA)

Nome campo	Descrizione
Versione	Numero di versione della specifica X.509
Numero seriale	Identificatore unico del certificato fornito dalla CA
Firma	Specifica l'algoritmo utilizzato dalla CA per firmare il certificato
Nome dell'emittente	Identificativo della CA che rilascia il certificato, in formato DN [RFC 4514]
Periodo di validità	Inizio e fine del periodo di validità del certificato
Nome del soggetto	Identificativo dell'entità la cui chiave pubblica è associata al certificato (in formato DN)
Chiave pubblica del soggetto	Chiave pubblica del soggetto e indicazioni dell'algoritmo da utilizzare

- Il problema della firma digitale è che non è garantita l'autenticità, ed è possibile spacciarsi per qualcun altro, qui nasce il concetto di autorità dei certificazione.
- **L'autorità di certificazione** ha il compito di **convalidare l'identità** e **rilasciare i certificati**.
    - La **chiave pubblica** viene così **certificata** e **collegata a un identità**.
- **10 compiti** di una CA:
    1. **Identificare con certezza la persona** che fa richiesta della certificazione della chiave pubblica.
    2. **Rilasciare e rendere pubblico il certificato**.
    3. Garantire **l'accesso telematico al registro** delle chiavi pubbliche.
    4. **Informare** i **richiedenti sulla procedura** di certificazione e sulle tecniche per accedervi.
    5. Dichiarare la propria **politica di sicurezza**.
    6. Attenersi alle norme sul trattamento di dati personali.
    7. **Non rendersi depositario delle chiavi private.**
    8. Procedere alla **revoca o** alla **sospensione** dei **certificati** in caso di richiesta dell'interessato o venendo a conoscenza di abusi o falsificazioni, ecc.
    9. Rendere pubblica la revoca o la sospensione delle chiavi.
    10. **Assicurare** la **corretta manutenzione del sistema** di certificazione.
- **Ottenimento di un certificato digitale:**
    - L'utente genera una coppia di chiavi pubblica e privata, e **invia la chiave pubblica** appena generata **alla CA** con una richiesta di certificazione.
    - La **CA autentica il richiedente**, normalmente chiedendo di recarsi al LVP locale (Local Validation Point).
    - Una volta verificata l'identità la **CA inserisce la chiave nel registro delle chiavi pubbliche**, ed **emette** e invia **il certificato** al richiedente.
- **PKI** (**P**ublic **K**ey **I**nfrastructure):
    - **Struttura gerarchica** per la gestione dei certificati.
    - Alcune **CA certificano altre CA**, ottenendo una "**catena di fiducia**".
        - La CA di primo livello di chiama **Root CA**.
        - Le CA di ultimo livello certificano il singolo utente.



- **Problemi:**
    - È comunque necessario ottenere in modo sicuro il certificato della CA.
    - Un certificato può essere revocato, ma la verifica della firma della CA sul certificato revocato va a buon fine!
        - La CA deve pubblicare una lista dei certificati revocati che andrebbe controllata per accertarsi della validità.
    - I**l sistema implica una fiducia nella CA, ma chi lo garantisce?**

Autenticazione

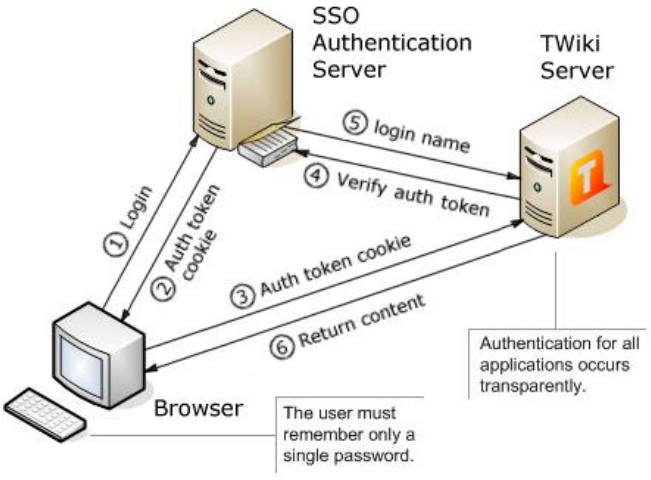
L'autenticazione è il servizio di sicurezza che permette di **garantire l'identità degli interlocutori**, che possono essere computer e utenti abbinati.

Fattori di autenticazione

- Basati su **qualcosa che l'utente**...
  - **Conosce:** password, PIN, etc
    - **Vantaggi: semplice** per l'utente, **economico**, senza immagazzinare segreti lato client.
    - **Svantaggi:** password deboli, vulnerabile a intercettazioni, cracking, e prigrizia degli utenti.
    - **OTP:** una password viene generata a ogni accesso, spesso basate sull'istante temporale, risolve il problema dell'intercettazione.
  - **Possiede:** chiavi, badge, token, etc
    - Autenticazione **basata su possesso**, che fornisce **prova dell'identità dell'utente**.
    - L'autenticazione dimostra solo l'identità del token, e non dell'utente (token rubati, clonati etc).
  - **È:** impronte digitali, etc
    - **Caratteristiche univoche che ne provano l'identità.**
    - Si basa su confronto tra template, le misure possono essere imprecise, dare falsi negativi o positivi.
- Possono essere combinati insieme diversi fattori di autenticazione (multi-factor authentication).

Si possono distinguere **quattro categorie diverse di sistemi di autenticazione:**

- ➔ **Locale:** l'utente accede **in locale al servizio**, che effettua l'autenticazione.
  - ◆ Utilizzabili tutti i fattori di autenticazione descritti senza modifiche.
- ➔ **Diretta:** l'utente accede **da remoto al servizio**, che effettua direttamente l'autenticazione.
  - ◆ Un intruso potrebbe registrare e replicare le informazioni ⇒ **necessario un canale sicuro**.
- ➔ **Indiretta:** l'utente accede **da remoto al servizio**, che si appoggia a un **servizio di autenticazione separato**.
  - ◆ Utilizzato quando molti **sistemi**/applicazioni **condividono** gli **stessi utenti**.
  - ◆ Le **informazioni** degli utenti vengono **centralizzate** su un sistema di autenticazione, utilizzato dagli altri sistemi.
  - ◆ Es: **RADIUS** (**R**emote **A**uthentication **D**ial In **U**ser **S**ervice), **Kerberos** (**SSO**).



- ➔ **Offline:** i **servizi** possono prendere **decisioni autonome** senza dover contattare ogni volta l'autorità di autenticazione.
  - ◆ **Basata su certificati emessi da un'autorità di certificazione.**
  - ◆ I certificati sono distribuiti da un'infrastruttura a chiave pubblica (PKI).

Autorizzazione

Il servizio di **controllo dell'accesso** (detto anche **autorizzazione**) garantisce che l'accesso alle risorse sia **limitato** ai **soli soggetti** che ne hanno **diritto**, che possono avere diritto a diverse modalità di interazione con le risorse stesse.

- ➔ **Soggetti:** utenti, applicazioni, altri sistemi, etc.
- ➔ **Privilegi:** lettura, scrittura, esecuzione, proprietà.
- ➔ **Oggetti:** file, funzioni, applicazioni, altri sistemi.

Controllo degli accessi

- Le **politiche di controllo dell'accesso** definiscono l'attribuzione dei privilegi di accesso dei soggetti sugli oggetti.
- I **meccanismi di controllo dell'accesso** specificano come le relazioni tra i soggetti e gli oggetti (i privilegi) sono rappresentate.
- **Principi utili:**
  - **Privilegio minimo:** ad un soggetto dovrebbero essere concessi solo i privilegi minimi necessari a compiere l'azione che deve compiere.
  - **Separazione dei compiti:** nessun soggetto dovrebbe avere abbastanza potere per sovvertire il sistema.

Meccanismi di controllo dell'accesso

- **Matrice** di controllo dell'accesso:
  - Le **righe** contengono i **soggetti**, le **colonne** gli **oggetti**, nelle caselle sono rappresentati i permessi.

	File 1	File 2	Progr.1	Progr.2
Alice	rwx	rwx,own	x	rwx
Bob	rwx,own	r	x	rwx
Progr.1	rw	rw	–	x

  - **Problemi di scalabilità** nel caso di soggetti e oggetti molto numerosi.
- **Lista** di controllo dell'accesso (**ACL**):
  - Con **matrice** di controllo memorizzata **per colonne**:
    - Ciascuna **risorsa** viene memorizzata **con una lista dei soggetti** che possono interagire con questa e con i relativi permessi.
    - Adatte in contesti in cui la protezione è orientata ai dati.
  - Con la **matrice** di controllo memorizzata **per righe**:
    - A ciascun **soggetto** è **associato l'insieme degli oggetti** con cui può interagire, si memorizza la lista di relazioni che il soggetto ha con gli oggetti e i relativi permessi.
    - Permette di gestire in modo efficiente i permessi associati a un singolo utente.

Politiche di controllo dell'accesso

- **DAC** (**D**iscretionary **A**ccess **C**ontrol)
  - I **singoli utenti possono** a loro discrezione **concedere** e **revocare permessi** su oggetti che sono **sotto il loro controllo**.
  - **Flessibile**, utilizzabile in molti ambiti.
  - Non permette di controllare la diffusione dell'informazione.
- **MAC** (**M**andatory **A**ccess **C**ontrol)
  - A differenza del DAC, la politica di controllo dell'accesso è **determinata centralmente dal sistema** e non dai singoli utenti.
    - **Meno flessibile** di DAC ma **più robusto**.
  - Basato sulla **classificazione degli oggetti e dei soggetti** (es. Top Secret, Secret, Confidential, Classified).

- Gli **approcci** possono essere **combinati** con una suddivisione in utenti e ruoli.
    - **I gruppi** permettono di **gestire insiemi di soggetti/oggetti** in modo omogeneo, dove l'accesso alle risorse è basato sui permessi dei gruppi stessi.
        - Modificare i diritti di un gruppo permette di cambiare direttamente quelli di tutte le entità appartenenti.
    - **I ruoli** definiscono **insiemi di proprietà e responsabilità** solitamente associate alla struttura organizzativa a cui fa capo il sistema (Role Based Access Control).

Firewall

I firewall di rete sono apparecchiature o **sistemi che controllano il flusso del traffico** tra due reti con differenti livelli di sicurezza.

- **Prevenire accessi non autorizzati** alla rete privata.
- **Prevenire esportazioni di dati non autorizzati** dall'interno verso l'esterno.
- **Schermare** alcune **reti** interne e nasconderle agli altri.
- **Bloccare** alcuni **accessi** a servizi o ad utenti.
- Monitoring: **logging** delle azioni.
- **Problematiche:**
    - Si assume che gli attacchi provengono dall'esterno della rete, ma possono iniziare anche dall'interno.
    - Non difende da bug non documentati nei protocolli utilizzati.
    - Filtri difficili da impostare a mantenere perchè è difficile trovare il compromesso tra sicurezza e libertà.
    - Potrebbe degradare le performance della rete.
- **Due politiche principali:**
    - Default **deny**: tutto quello che non è espressamente permesso è negato.
        - I **servizi** sono **abilitati** caso per caso **dopo un analisi**.
        - Utenti molto ristretti e non possono facilmente rompere le policy di sicurezza.
    - Default **permit**: tutto quello che non è espressamente proibito è concesso.
        - I system admin devono reagire limitando o bloccando i protocolli a ogni bug trovato negli stessi.
        - Utenti meno ristretti.

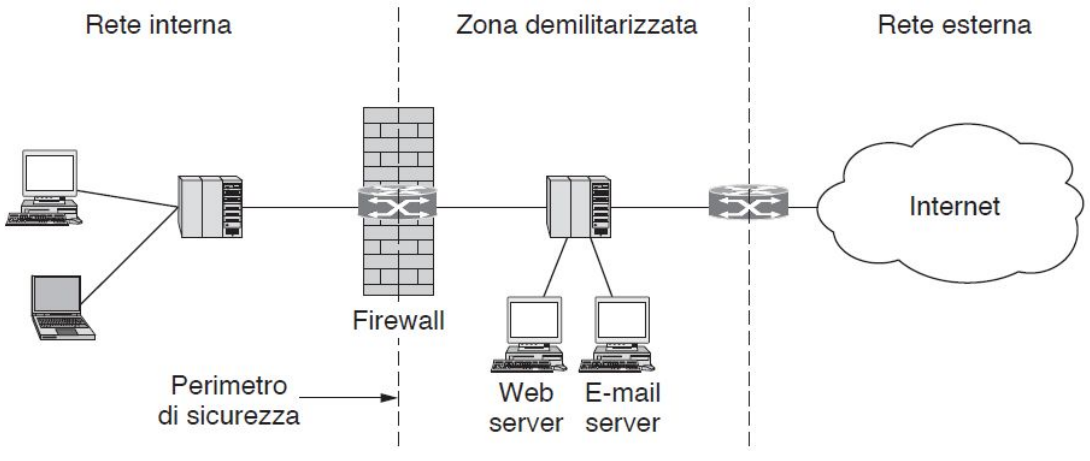
Tre diverse tipologie di firewall:

1. **Packet-Filtering** (1° gen).
    a. **Lavorano a livello 3** e alcune caratteristiche del livello 4.
    b. Possono **filtrare** in base a indirizzo di sorgente/destinazione, tipo di traffico, porte, o informazioni interne al router (come interfacce di destinazione o sorgente).
    c. Vantaggi:
        i. Disponibili in molti router.
        ii. Costo contenuto.
        iii. Trasparenza.
        iv. Velocità.
    d. Svantaggi:
        i. Regole difficili da regolare e possono avere bug.

2. **Stateful-Inspection** (2° gen).
    a. Le informazioni di una **connessione** vengono **salvate** in una **tabella di stato** se non bloccata da nessuna regola.
    b. I **pacchetti** vengono **valutati** in base alla entry delle **connessioni consentite** nella tabella di stato.
    c. Quando una connessione viene chiusa le entry della tabella associata vengono cancellate.
    d. Vantaggi:
        i. -> Tutti i vantaggi del Packet Filtering.
        ii. Buon rapporto tra prestazioni e sicurezza (performance più alte, meno controlli sulla connessione)
        iii. Protezione da IP Spoofing.
    e. Svantaggi:
        i. Mancanza di servizi aggiuntivi.
        ii. Testing complesso.

Source address	Source port	Dest. Address	Dest. Port	Connection state
192.168.0.199	1051	192.168.1.10	80	Handshaking
192.168.0.212	1109	192.168.1.23	25	Closing
192.168.3.105	1212	192.168.0.111	80	Established

3. **Gateway a livello di applicazione** (o **Proxy Server**) (3° gen).
    a. **Livello 7:** il **routing** tra le due interfacce viene fatto **a livello applicazione** dal software del firewall.
    b. Possibilità di autenticazione.
    c. Filtri su comandi specifici.
    d. Vantaggi:
        i. Più sicuro del packet filtering.
        ii. Deve controllare solamente un numero limitato di applicazioni (HTTP, FTP, posta, etc).
        iii. Logging e controllo del traffico semplificato.
    e. Svantaggi:
        i. Overhead su ogni connessione.
        ii. Possibilità di controllare solo un numero limitato di applicazioni.
4. **Proxy server dedicati**
    a. Specifici per ogni applicazione.
    b. Aiutano l'application proxy gateway nel lavoro di contents-inspection.
5. **Personal firewalls**
    a. Proteggono la macchina dove è installato.
    b. Necessario nei mobile users.



Intrusion Detection System (IDS)

Strumento, software o hardware, che **automatizza il processo di monitoraggio** impiegato per individuare eventi che rappresentano **intrusioni non autorizzate**, i computer o reti locali, attuabile **a livello di host (HIDS)** o **rete (NIDS)**.

Un **IDS ideale dovrebbe** riuscire ad **individuare tutte le reali intrusioni**.

- Falsi positivi: IDS rileva un'anomalia, ma non è successo niente.
- Falsi negativi: IDS non rileva un'intrusione avvenuta.

Requisiti di un IDS ideale:

- Scoprire un'ampia gamma di intrusioni, sia già note che non note.
- Scoprirle velocemente, non necessariamente in tempo reale.
- Presentare i report delle analisi in formato semplice e facilmente comprensibile.
- Accuratezza.

Principi di base

- Deve **distinguere** le situazioni **normali** dalle **anomalie**.
- Un **utente normale** si comporta in maniera più o meno **prevedibile**.
    - Non compie azioni atte a violare la sicurezza.
    - I processi da lui eseguiti compiono azioni permesse.

Modelli utilizzati da un IDS (possono essere **statici** o **adattivi**):

- **Scoperta di anomalie:** alcune sequenze di azioni inusuali possono essere tentativi di intrusione.
    - Si **analizzano insiemi di caratteristiche** del sistema confrontando i valori con quelli attesi e segnalando quando le statistiche non sono paragonabili a quelle attese.
        - **Metriche a soglia**
            - Contare il numero di volte che un evento si presenta.
            - Es: Windows: blocco dopo k tentativi di login falliti. Il range è (0, k–1).
        - **Momenti statistici**
            - L'analizzatore calcola la **deviazione standard** (i primi due momenti) o altre misure di correlazione (momenti di ordine superiore).
            - Se i valori misurati di un certo momento cadono al di fuori di un certo intervallo vi è un'anomalia.
        - **Modelli di Markov**
            - La **storia passata influenza la prossima transizione di stato**.
            - Le anomalie sono riconosciute da sequenze di eventi, e non sulle occorrenze di singoli eventi.
            - Il sistema deve essere addestrato per riconoscere sequenze valide.
- **Scoperta di azioni malevole:** nel caso si conosca quali azioni sono considerate malevole.
    - Si **controlla** se una **sequenza di istruzioni** da eseguire è **già nota per essere** potenzialmente **dannosa** per la sicurezza del sistema.
    - La conoscenza è rappresentata mediante regole e il sistema controlla se la sequenza soddisfa una di queste regole.
    - Non si possono scoprire intrusioni non note precedentemente.
- **Scoperta in base a specifiche:** quando si conoscono le **situazioni derivanti da intrusioni**.
    - Si determina se una sequenza di azioni viola una specifica di come un programma o un sistema dovrebbe funzionare.

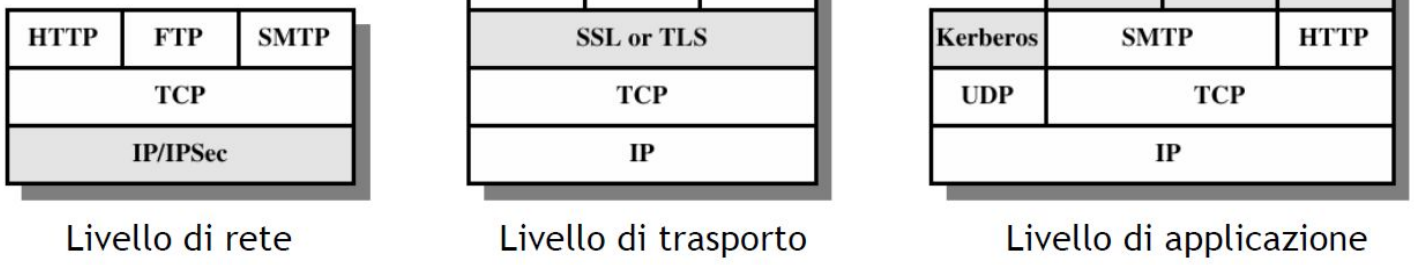
Architettura di un IDS

- È essenzialmente un **sistema di auditing sofisticato**.
- **Tre attori** principali
    - **Agente** (una sorta di **logger**)
        - **Gestisce e rileva le informazioni** e le invia al direttore.
        - Il direttore può richiedere informazioni aggiuntive all'agente.
        - Suddivisi in agenti network e host.
    - **Direttore** (**analizzatore**)
        - **Colleziona** i **dati** inviati **dagli agenti**.
        - Analizza le informazioni rilevanti per determinare attacchi.
        - Gira su un sistema separato rispetto gli agenti.
    - **Notificatore** (**esecutore**)
        - Ottiene i risultati e le informazioni dal direttore e **prende decisione appropriate**.
            - Notificare messaggi agli amministratori.
            - Riconfigurare gli agenti.
            - Rispondere all'attacco.
- **DIDS:** combina agenti sui singoli host e un monitor di rete.

Risposte alle intrusioni

- **Prevenzione:** l'attacco deve essere scoperto prima del completamento.
- **Jailing:** far credere all'attaccante che l'attacco è andato a buon fine, ma confinare le azioni dove non può arrecare danni.

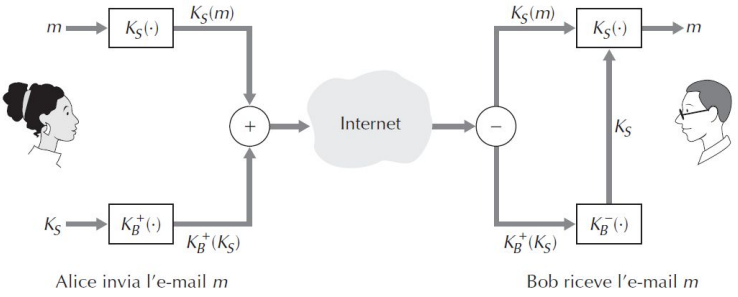
La sicurezza nello stack protocollare TCP/IP



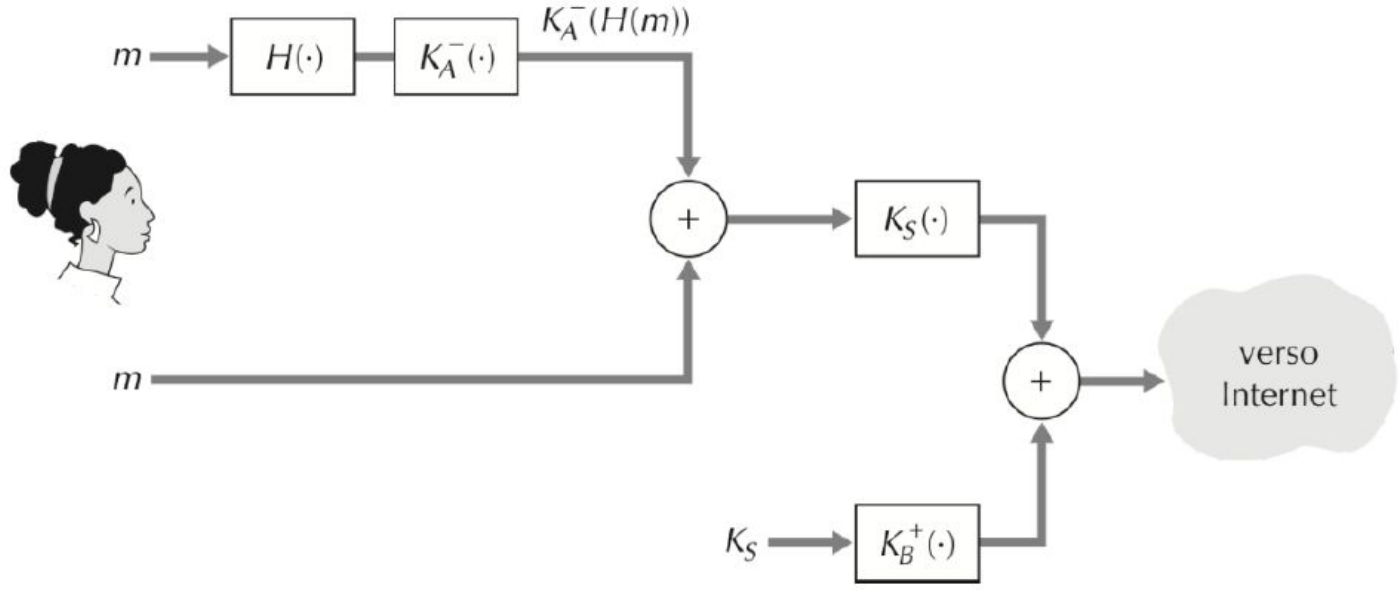
Sicurezza delle email

Caratteristiche di sicurezza necessarie:

- **Riservatezza.**
    - **Cifrando il messaggio** con algoritmi a chiave simmetrica (DES/AES) o a chiave asimmetrica (RSA).
- **Integrità.**
- **Autenticazione** del mittente e del ricevente.

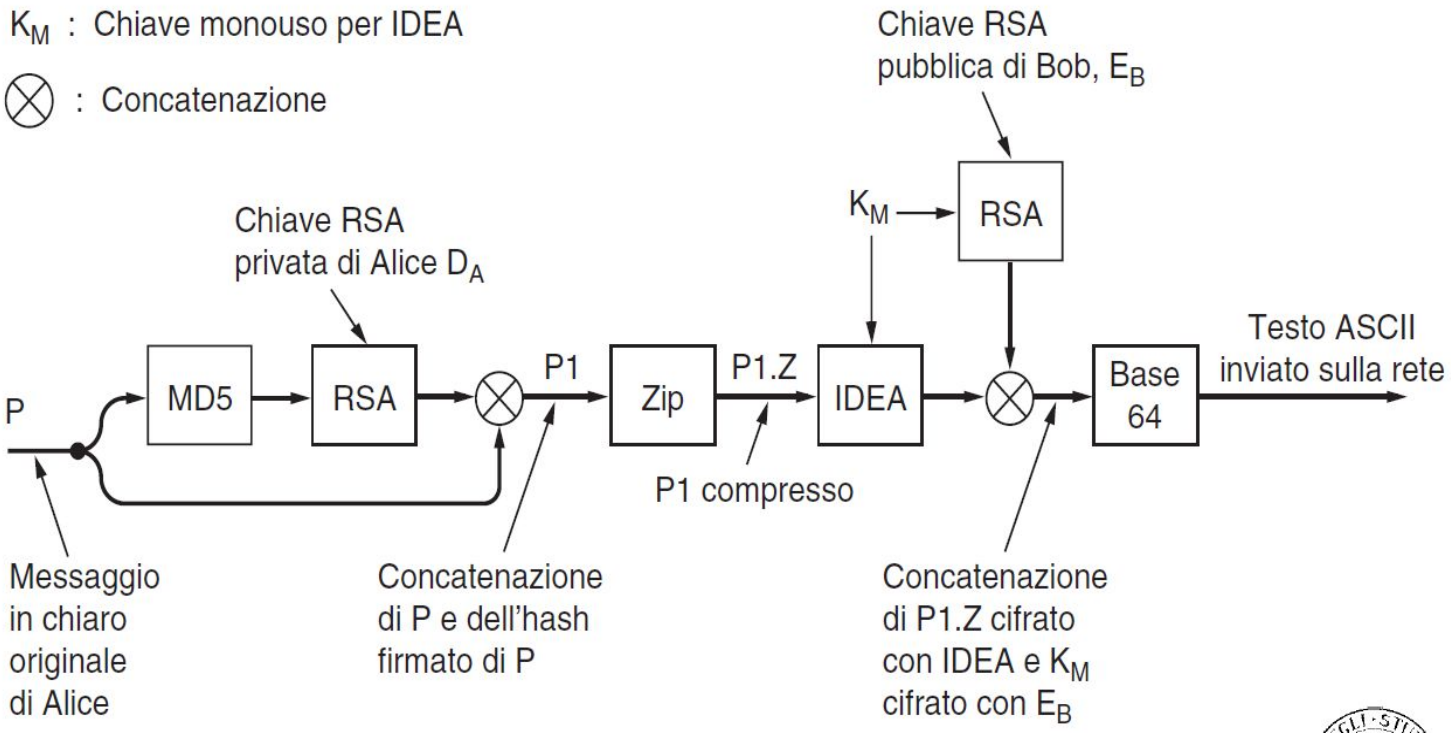


ES: Alice utilizza la crittografia a chiave simmetrica, quella a chiave pubblica, una funzione hash e la firma digitale per ottenere segretezza, autenticazione del mittente e integrità del messaggio.

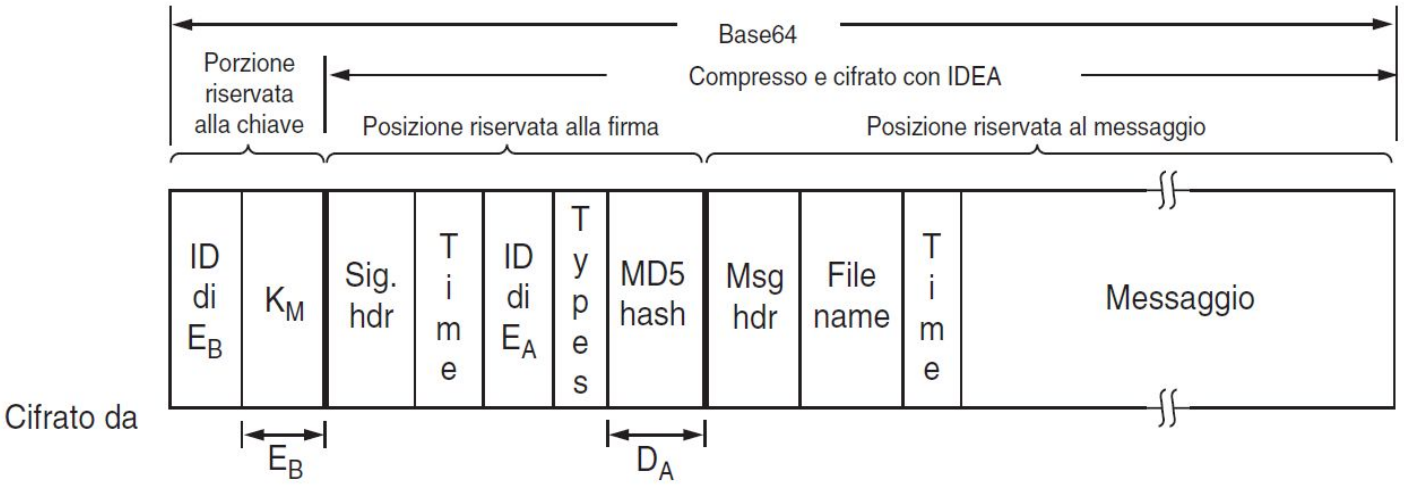


PGP (Pretty Good Privacy)

- ➔ Sviluppato da P. Zimmerman, simbolo del diritto alla privacy elettronica.
- ➔ Programma per lo **scambio sicuro di messaggi testuali** (confidenzialità, autenticazione).
- ➔ **Integra algoritmi di crittografia consolidati**.
- ➔ **Open source** e indipendente dal sistema operativo utilizzato.
- ➔ **Servizi offerti:**
  - ◆ **Autenticazione:** basata su **RSA** o **SHA1**, supporta anche firme distaccate.
  - ◆ **Confidenzialità:** con algoritmo CAST-128 o IDEA o Triplo DES, utilizza chiave di sessione one-time.
  - ◆ **Compressione.**
  - ◆ **Codifica per la compatibilità.**
  - ◆ **Segmentazione.**
- ➔ Invio di un messaggio con PGP:



➔ Formato dei messaggi:



➔ È compito dell'**utente assegnare un livello di fiducia ad ogni conoscente** ed intermediario:
- Il campo **owner trust** esprime il **grado di fiducia nel proprietario** come certificatore; è assegnato dall'utente (unknown, untrusted, marginally trusted, completely trusted).
- Il campo **signature trust** esprime il **grado di fiducia nel firmatario** come certificatore; è uguale a owner trust se il firmatario è tra i conoscenti, altrimenti vale unknown.

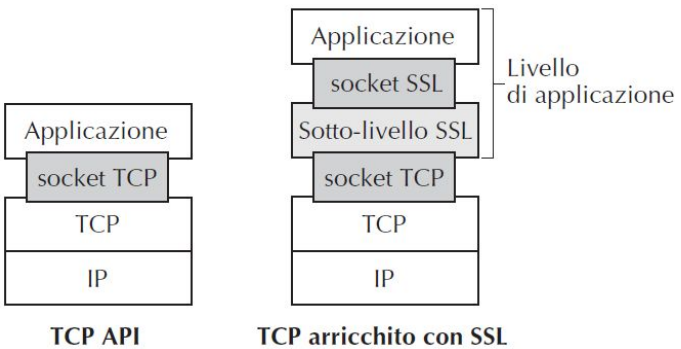
➔ **PGP assegna** il livello di **fiducia nell'abbinamento chiave pubblica ⇒ utente.**

Sicurezza del livello di trasporto

Secure **S**ockets **L**ayer (**SSL**)

➔ Versione di **TCP** arricchita **con servizi di sicurezza**, comprese riservatezza, integrità dei dati e autenticazione del client e del server.

➔ Protocollo progettato inizialmente da NETSCAPE con il nome di SSL, specificatamente per la protezione delle transazioni web.

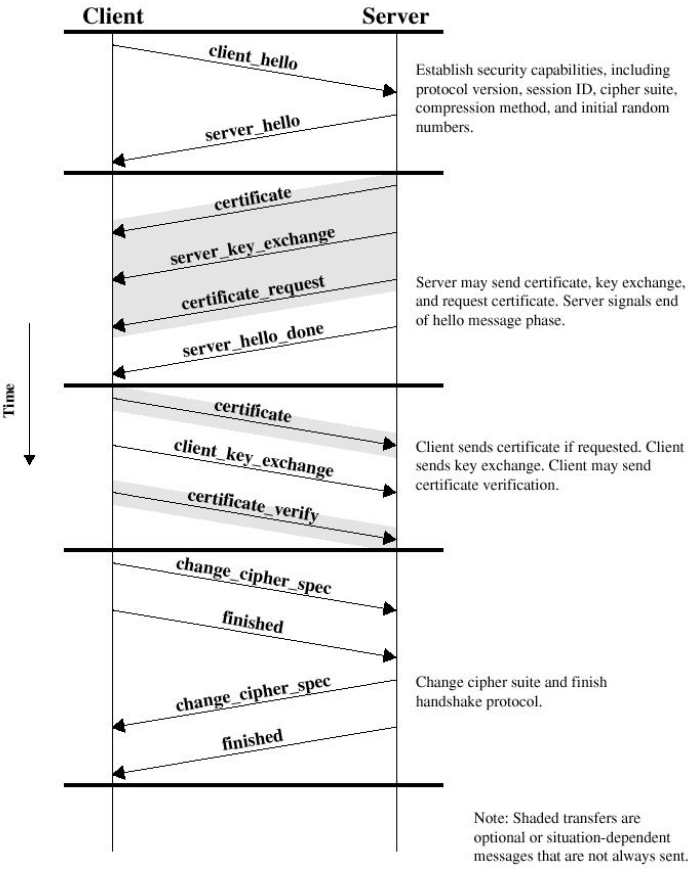


➔ Divenuto **standard IETF**, a partire dalla versione 3.0, (RFC 2246) **con il nome TLS**.

➔ Principalmente **focalizzato** sulle proprietà di **Confidenzialità** e **Integrità** del traffico di rete.

➔ **Handshake:**

- Accorda le due parti sugli algoritmi crittografici da utilizzare.
- Genera le chiavi di sessione per la cifratura dei dati.
- [Opzionale] Autenticazione delle parti.



Sicurezza delle wireless LAN

Wired **E**quivalent **P**rivacy (**WEP**)

- ➔ Protocollo 802.11 progettato per dare **sicurezza** ai **dati** in transito **su reti wireless**.
- ➔ Fornisce autenticazione e **codifica** dei dati **tra terminale e access point** wireless con un approccio a chiave simmetrica condivisa.
- ➔ Pensato per assicurare un livello di sicurezza simile a quello delle reti cablate.
- ➔ Lavora a **livello di datalink**.
- ➔ **Richiede la stessa secret key** condivisa tra **tutti i sistemi in comunicazione** (host e access point).
- ➔ Fornisce autenticazione (per device, cifratura Challenge/Response) e crittografia.
- ➔ Metodi di **autenticazione:**
    - ◆ Shared KEY:
        - L'AP invia il testo in chiaro.
        - Il device cripta il testo in chiaro e lo invia all'AP.
        - Se il testo in chiaro è criptato correttamente, l'AP ritiene autenticato il device.
        - **<!>** Testo in chiaro e testo criptato a disposizione degli attaccanti, possibili gli attacchi bruteforce **<!>**
- ➔ Protocollo **vulnerabile** ("eavesdropping" & "tampering") ⇒ (intercettazioni e manomissioni).
- ➔ Possibili **compromissione** di confidentiality e data integrity.
- ➔ **Scarso controllo di accesso.**

Wi-Fi **P**rotected **A**ccess (**WPA** e **WPA2**)

- ➔ **Crittazione dei dati**, usati con gli standard di autenticazione 802.1X.
- ➔ **Integrità** dei dati.
- ➔ Protezione da attacchi di tipo "replay".
- ➔ Operano a **livello MAC** (Media Access Control).
- ➔ Contiene un sottoinsieme delle feature di sicurezza che sono nello standard 802.11i.
- ➔ **Autenticazione** con EAP.
- ➔ Cifratura e data integrity.
- ➔ **Risolve molte debolezze di WEP.**
- ➔ Attualmente il più sicuro.