Univ. Ferrara – Corso di laurea in Informatica

Computer e network security

Alberto Gianoli

Premessa

Tradizionalmente la sicurezza dei computer era legata a fattori pratici (pochi computer perchè costosi) e amministrativi (difficile accesso a un computer)

L'aumento degli utilizzatori ha richiesto la creazione di meccanismi per tenere separati gli utenti e i loro dati.

L'introduzione della rete e dei collegamenti tra computer richiede misure per proteggere i dati durante la trasmissione

Definizioni

Computer security: è il nome generico con cui ci si riferisce all'insieme di tools usati per proteggere i dati e tenere lontani gli hackers

Network security: sono i meccanismi usati per proteggere le informazioni quando vengono trasmesse da un computer ad un altro all'interno di una rete.

Internet security: estensione del problema precedente quando si ha a che fare con Internet.

Definizioni (2)

Security attack: qualsiasi azione che compromette la sicurezza dei dati (o del computer)

Security mechanism: un meccanismo ideato per rilevare, prevenire o eliminare un security attack. Non esiste un unico meccanismo che protegge da tutto. Però hanno quasi tutti in comune l'uso della crittografia.

Security service: un servizio che migliora la sicurezza di un sistema o della trasmissione di dati. I security services fanno uso di uno o più security mechanisms.

Security attacks

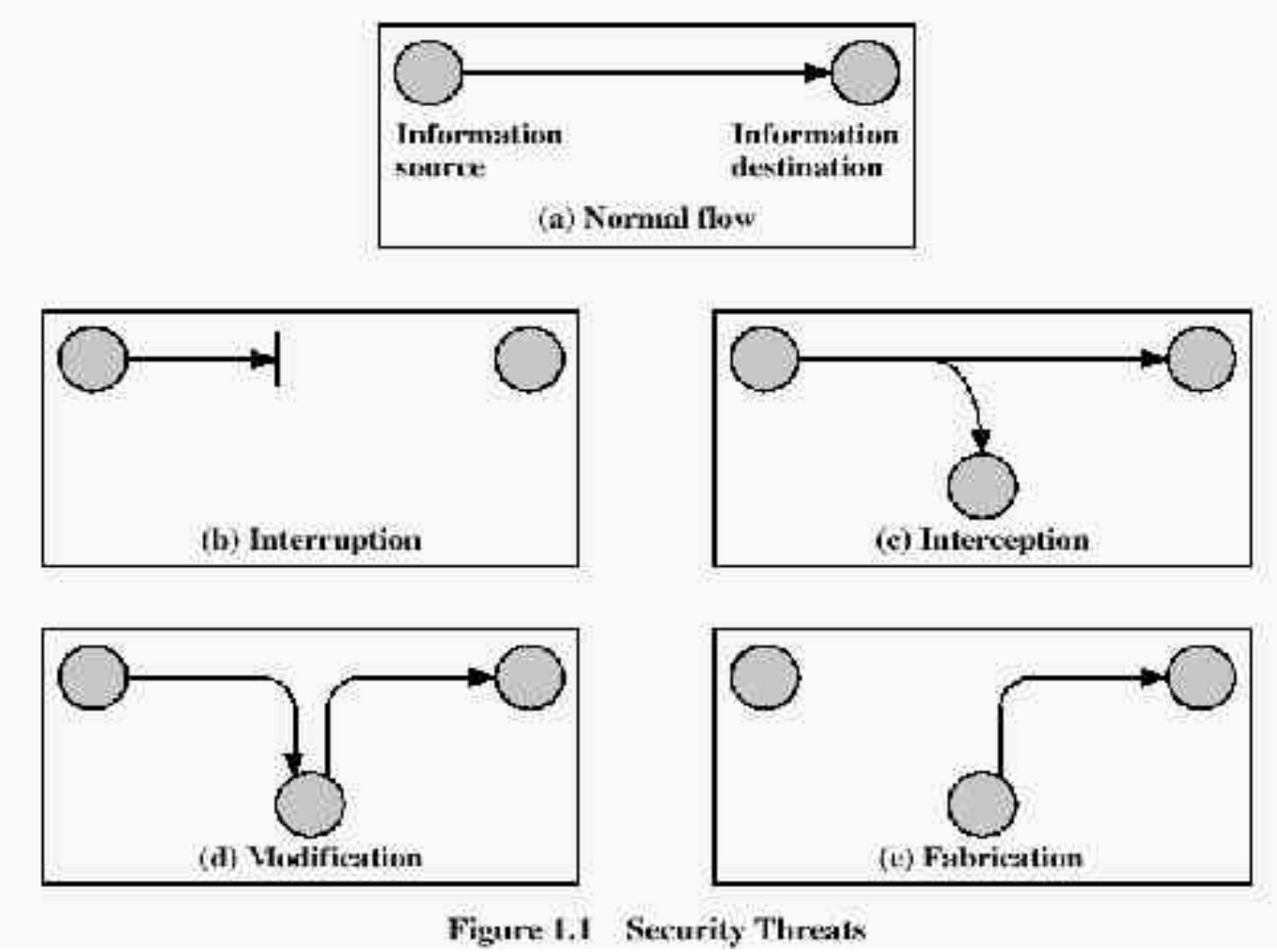


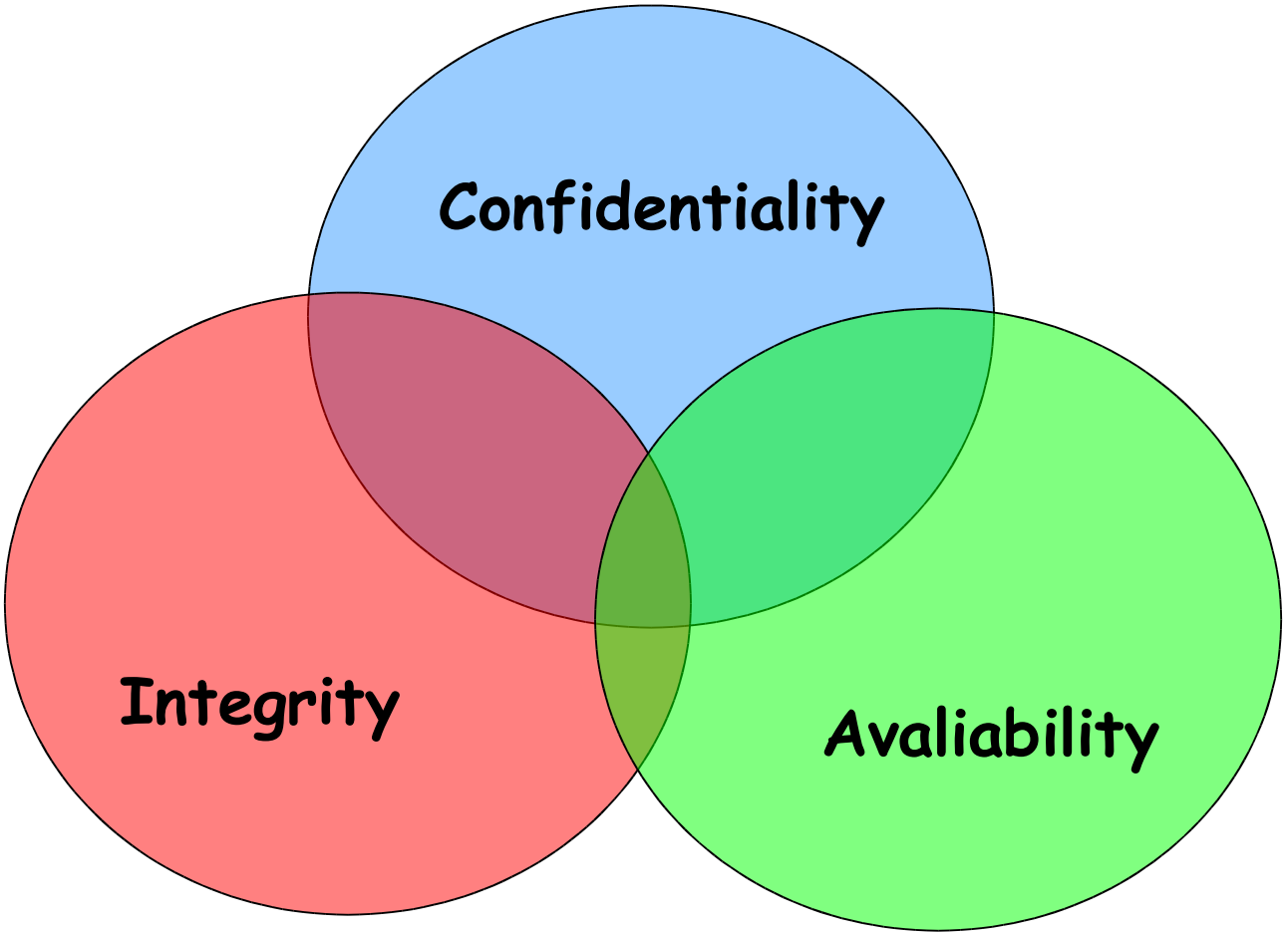
Figure 1.1 Security Threats

Security attacks (2)

- Interruzione: è un attacco che colpisce l'availability di un sistema
- Interception: colpisce la confidentiality
- Modification: altera l'integrità dei dati
- Fabrication: compromette l'autenticità di un messaggio

Il secondo attacco è considerato un attacco passivo (più difficile da scoprire), mentre gli altri sono classificati come attacchi attivi.

Security goals



Security services

Confidenzialità: privacy

Autenticazione: chi ha creato / mandato dei dati è veramente chi dice di essere

Integrità: i dati non sono stati manomessi

Non ripudiabilità: far finta di non aver mandato un email

Access control: prevenire l'uso improprio delle risorse

Availability: garantiscono l'accesso a un servizio / ai dati

Metodi di difesa 1

- Minimi privilegi: ogni oggetto (utente o programma) deve avere solo i minimi privilegi necessari al suo lavoro

- Difesa in profondità: non fare affidamento su un solo meccanismo di difesa, per forte che possa essere/sembrare

- Minimizzare i punti di accesso: dato che vanno controllati, ridurne il numero aiuta il lavoro

- Cercare l'anello debole: bisogna essere consci di qual'è il punto più debole del sistema, e se necessario rafforzarlo fino ad un livello accettabile

Metodi di difesa 2

- Fail-safe: quando si verifica un errore il sistema non deve permettere l'accesso a intrusi, anche a costo di rifiutare utenti legittimi

- Ciò che non è espressamente permesso è proibito:
  - Fail-safe
  - Di soito l'utente pensa l'opposto
  - Attivare i servizi sulla base delle reali necessità; gli altri sono disattivati

- Partecipazione universale: non è possibile avere una macchina sicura senza la partecipazione degli utenti

Metodi di difesa 3

- Criptare le informazioni
- Controlli software: metodi per ridurre l'accesso ai dati o ai servizi a seconda dell'utente. Si basano sulla conoscenza di una password.
- Controlli hardware: come sopra ma si basano su una pezzo hardware (es. smartcard)
- Policies: per esempio obbligare gli utenti a cambiare le password con una certa frequenza oppure imporre password opportunamente scelte.
- Controlli antropometrici: fa molto film di spionaggio, ma un portatile capace di riconoscere l'impronta digitale è già esistito.....

Crittografia

I metodi di crittografia sono caratterizzati da tre parametri indipendenti:

- Il tipo di operazione usata per passare dal plaintext al ciphertext

- Il numero di chiavi usate
  - → Chiave simmetrica (chiave unica)
  - → Chiave asimmetrica (due chiavi, public key encryption)

- Il modo in cui il plaintext viene processato.

Crittografia (2)

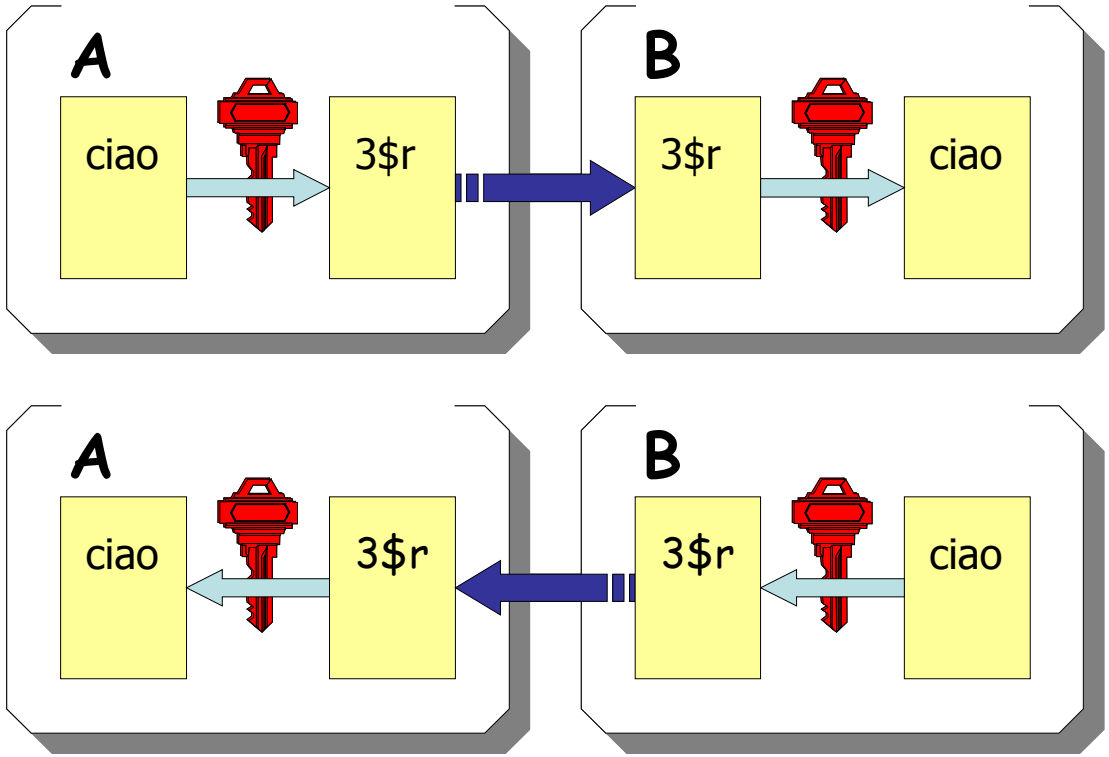
Key Size (bits)	Numero di chiavi alternative	Tempo necessario a 10^6 decritt./µs
32	$2^{32} = 4.3 \times 10^9$	2.15 millisecondi
56	$2^{56} = 7.2 \times 10^{16}$	10 ore
128	$2^{128} = 3.4 \times 10^{38}$	5.4×10^{18} anni
168	$2^{168} = 3.7 \times 10^{50}$	5.9×10^{30} anni

Crittografia (3)

Il sistema più usato è quello a chiave pubblica: facilita la distribuzione della chiave, mantenendo il livello di sicurezza.

Come si fa a scambiarsi la chiave?

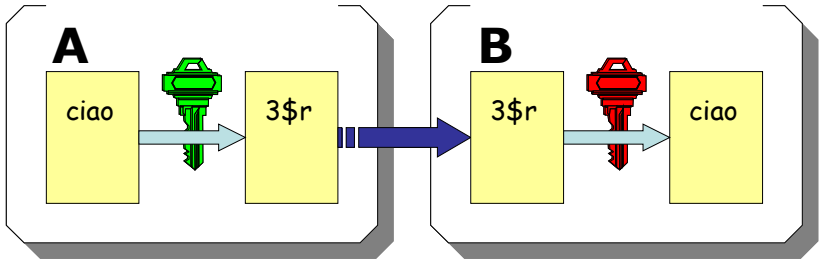
Se ci sono n utenti il numero delle chiavi è $O(n^2)$



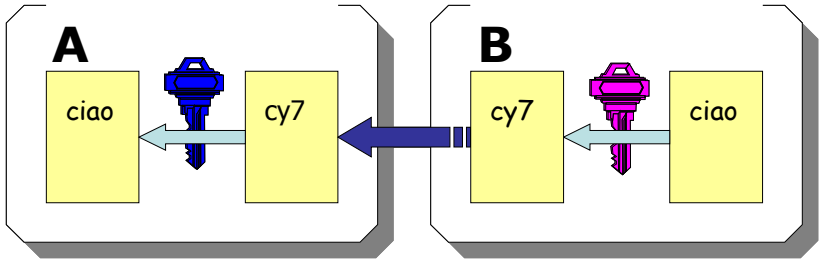
Crittografia (4)

Nel meccanismo detto “a chiave pubblica” ogni utente ha due chiavi (pubblica e privata):

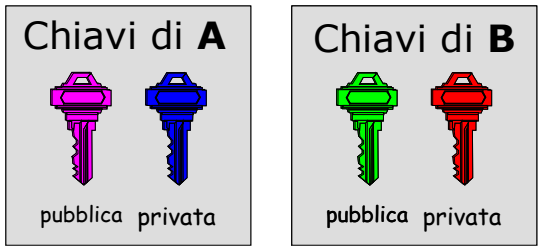
- Da una è impossibile risalire all'altra
- Quello che si cifra con una si decifra con l'altra



Non è necessario lo scambio: il mittente cifra con la pubblica del destinatario. Questi decifra con la sua chiave privata.



Per n utenti il numero di chiavi è O(n)



Funzioni di hashing

Sono funzioni che accettano in input un messaggio di lunghezza variabile e producono una stringa di lunghezza fissa (*hash*)

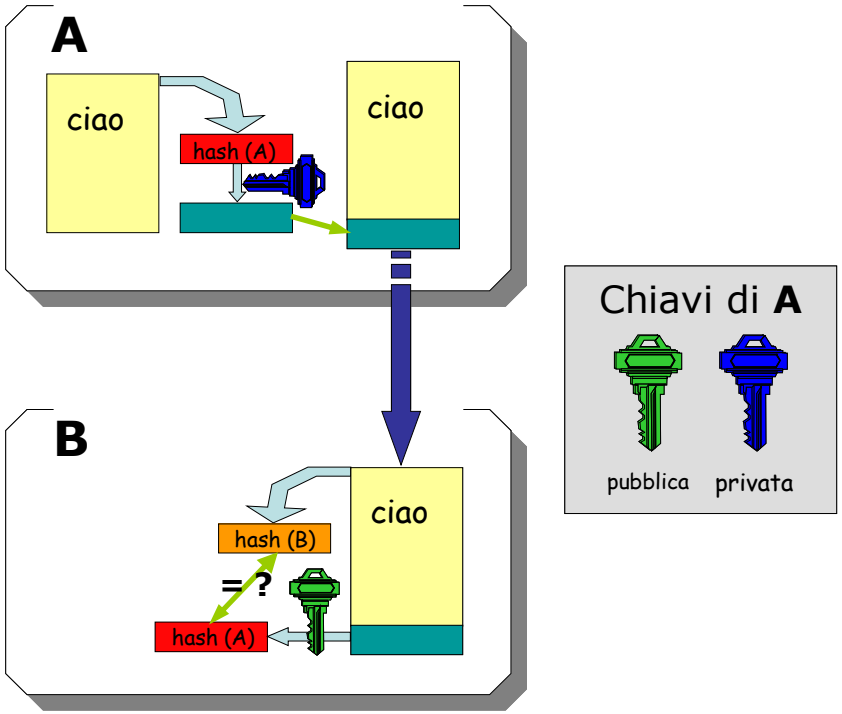
Ovviamente due messaggi diversi possono avere lo stesso hash, ma:

- minime modifiche a un messaggio cambiano radicalmente il suo hash
- la possibilità che due messaggi diversi che abbiano senso e che siano simili/correlati producano lo stesso hash è praticamente nulla

Firma digitale

Possiamo mettere insieme hash e cifratura per produrre una firma digitale:

- A calcola l'hash del messaggio e lo cifra con la sua chiave privata; l'hash cifrato è la firma digitale
- A invia messaggio e hash a B
- B ricalcola l'hash e lo confronta con quello ricevuto da A
- Se i due hash sono uguali, il messaggio non è stato modificato e A non può ripudiarlo



Certificati

Un messaggio dotato di firma digitale è “sicuro” se:
- la chiave di A non è stata compromessa
- B conosce la “vera” chiave pubblica di A

La convalida delle chiavi pubbliche viene effettuata attraverso i certificati: una autorità esterna (Certification Authority, CA) garantisce l'autenticità

Esistono due modelli basati su questi presupposti:
- PGP: “web of trust”
- X.509: organizzazione gerarchica

X.509

Un certificato X.509 contiene sicuramente (altri campi sono opzionali):

- alcune informazioni sul proprietario
- la data di scadenza
- la chiave pubblica del proprietario
- informazioni sul garante (la CA)

Il certificato è firmato dall CA

→ cioè contiene anche un hash del suo contenuto, cifrato con la chiave privata della CA

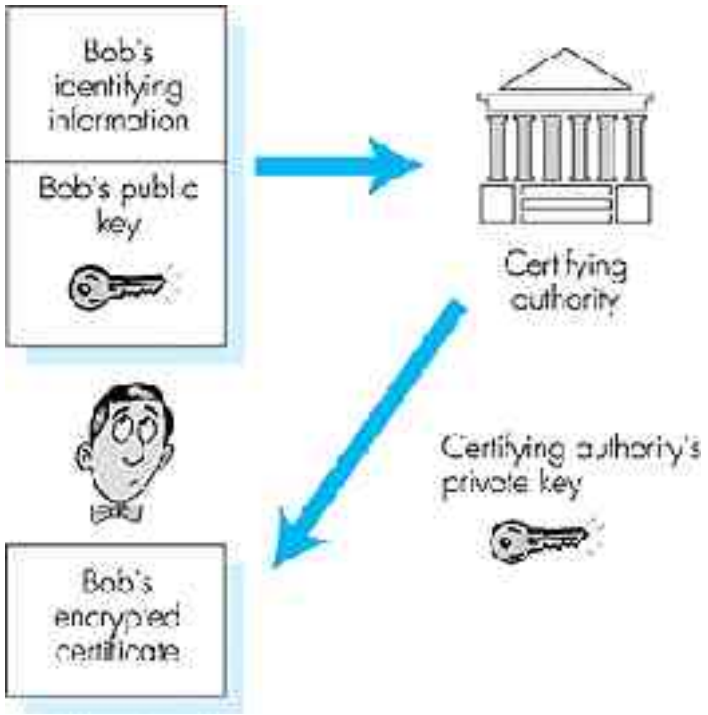
Autorità di certificazione (1)

Autorità di certificazione (CA) garantisce chiavi pubbliche di ogni utente

- Utente conosce la chiave pubblica della CA

Un utente (persona, router, etc.) registra la sua chiave pubblica con CA

- Utente fornisce a CA "garanzia di identità"
- CA crea certificato collega utente a chiave pubblica
- Certificato è firmato dalla CA



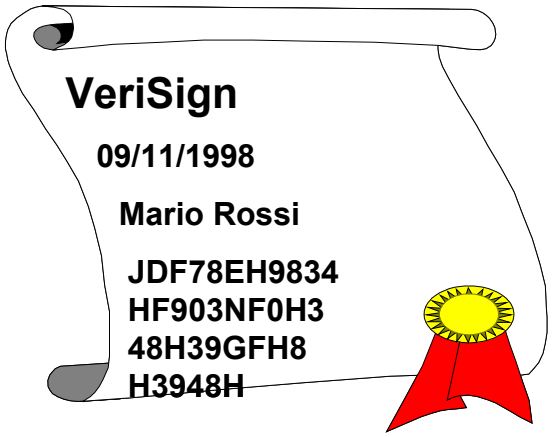
- Alice vuole conoscere la chiave pubblica di Bob:
- Chiede a CA certificato di Bob
- Verifica autenticità del certificato (verifica firma CA)

Autorità di certificazione (2)

L'Autorità di certificazione:

•garantisce la effettiva corrispondenza di una chiave pubblica con il soggetto che la espone

•pubblica, in un apposito registro, certificati firmati con la propria chiave privata che specificano:
- Il nome dell'Autorità
- La data di emissione del certificato
- La data di scadenza del certificato
- Il nominativo del soggetto
- La chiave pubblica del soggetto



Autorità di certificazione (3)

Le chiavi pubbliche possono essere sospese o revocate (ad es. furto o smarrimento)

- L'Autorità di certificazione gestisce un registro storico delle chiavi pubbliche revocate

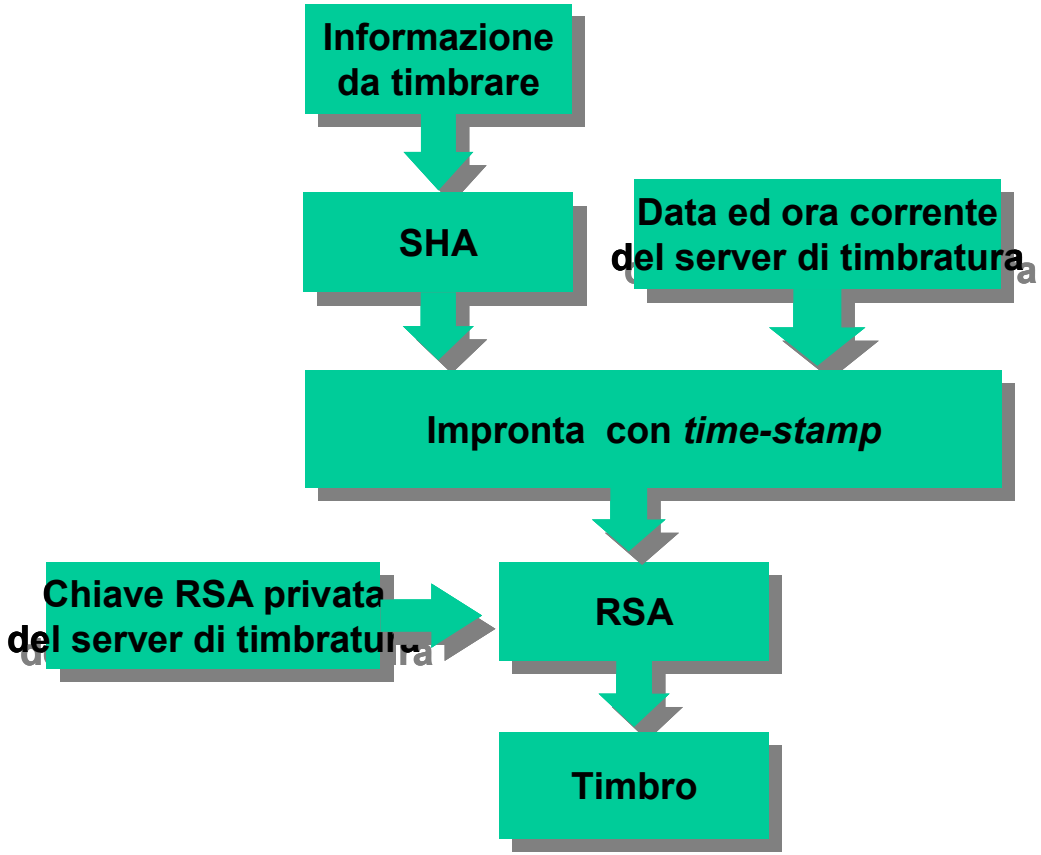
I certificati vanno chiesti al momento della verifica della firma

Esempio: verifica di un firma

- Si chiede alla CA la chiave pubblica del firmatario al momento della firma.
- Tale sequenza di operazioni viene svolta in modo automatico dal software

Autorità di certificazione (4)

Allo stesso modo una CA può anche fornire una certificazione temporale (quando è stato creato un documento)

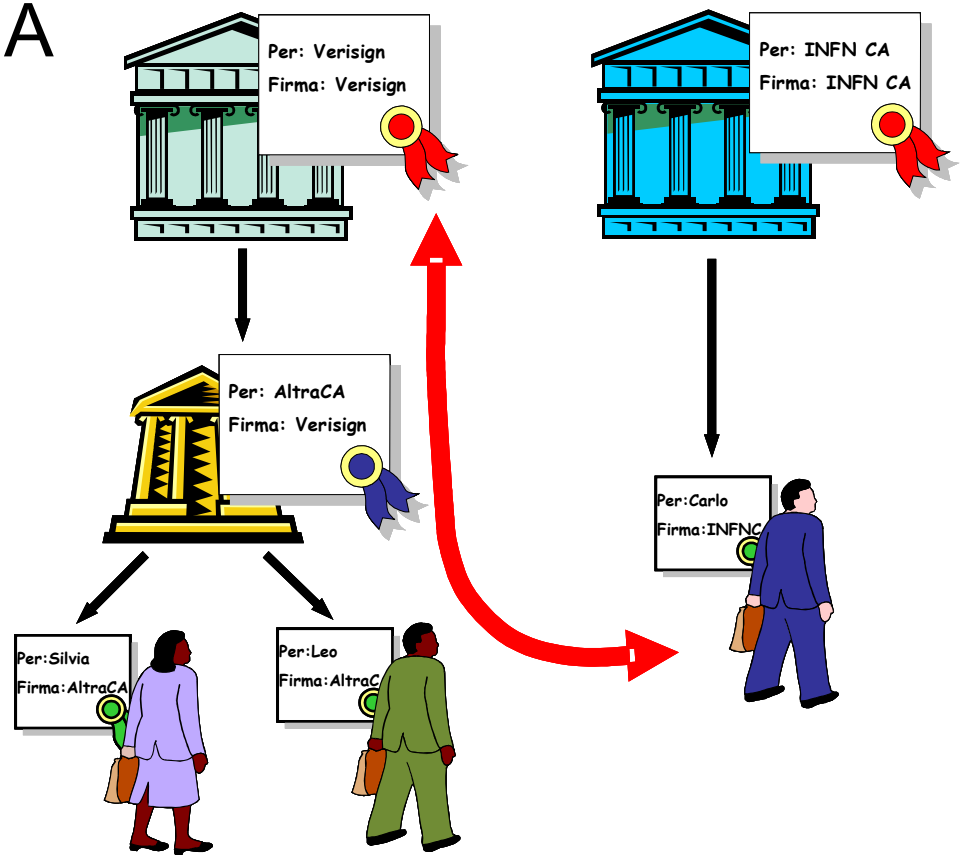


Autorità di certificazione (5)

Anche le CA hanno un proprio certificato

Una CA può garantire un'altra CA (livello inferiore), formando una catena gerarchica di certificati

All'origine della catena c'è una **Root CA** che ha un certificato auto-firmato (root certificate)



Public Key Infrastructure (PKI)

- Una o più CA organizzate gerarchicamente o via “web of trust”
  - Politica di emissione dei certificati: Certificate Practice Statement (CPS)
  - Emette certificati per server e utenti finali (o altre CA)
  - Mantiene una Certificate Revocation List (CRL)
  - Gestisce servizi WWW e LDAP (pubblicazione cert.)
- Gestione delle chiavi private
  - Generazione
    - via browser
    - via hardware: ad es. SmartCard
  - Custodia
    - Hard disk (insicuro!)
    - SmartCard

Richiami di reti (1)

Stack TCP/IP

Application Layer

(SMTP, Telnet, SSH, FTP, ...)

Transport Layer

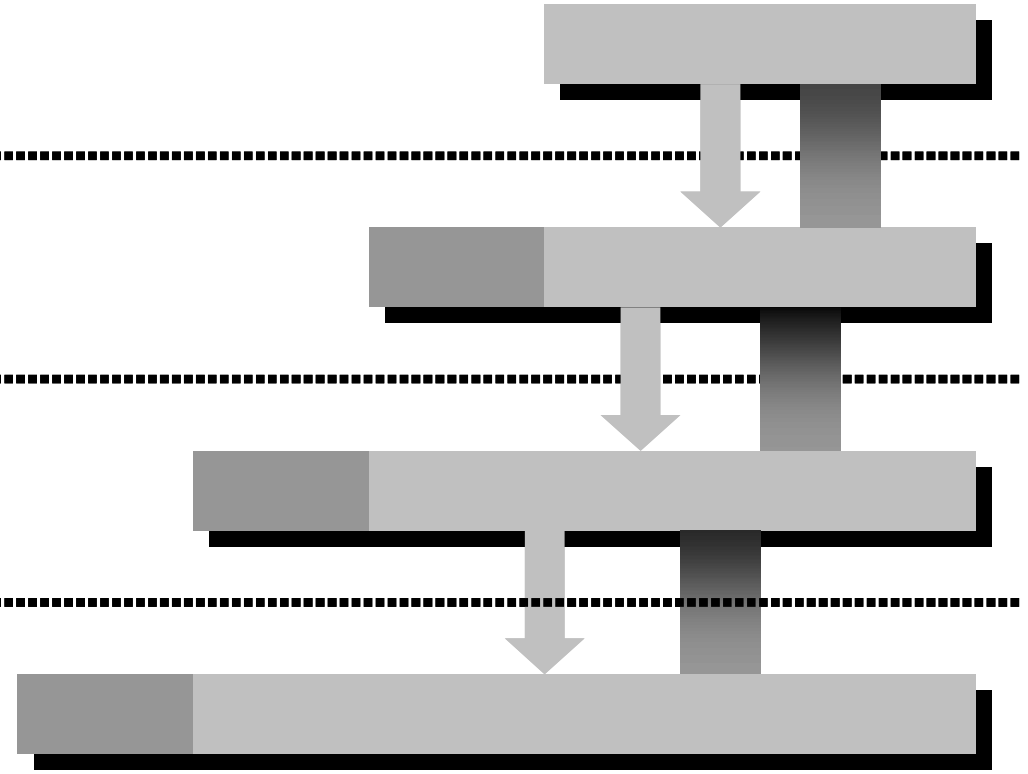
(TCP, UDP, ...)

Internet Layer

(IP, ICMP)

Data Link Layer

(Ethernet, FDDI, ATM, ...)



Richiami di reti (2)

Internet Control Message Protocol

- Protocollo di servizio per IP
  - ➔diagnostica Internet
  - ➔usato da host e router per segnalare condizioni di errore
- Usa datagrammi IP
  - ➔il payload contiene sempre l'header IP e i primi 8 byte del datagramma che ha provocato il messaggio.
  - ➔vari tipi di messaggio (es. Echo Request/Reply, Host Unreachable, Need to Frag, Time Exceeded, etc..)

Richiami di reti (3)

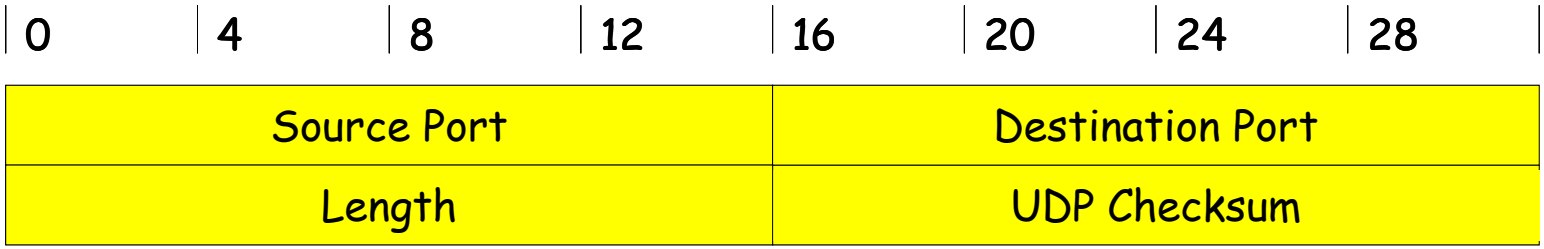
- Concetti per protocolli al livello di trasporto:
  - **Porta**
    - astrazione usata dai protocolli di trasporto per distinguere fra più processi sullo stesso host
    - intero a 16 bit (0-65535)
    - per servizi standard identificativo assegnato dalla IANA
  - **Socket**
    - coppia (indirizzo IP, porta)
    - generalizzazione del meccanismo di accesso a file

Richiami di reti (4)

User Datagram Protocol

Protocollo connectionless (come IP)

- nessuna garanzia di consegna
- assenza controllo di flusso
- assenza di correzione di errore



Richiami di reti (5)

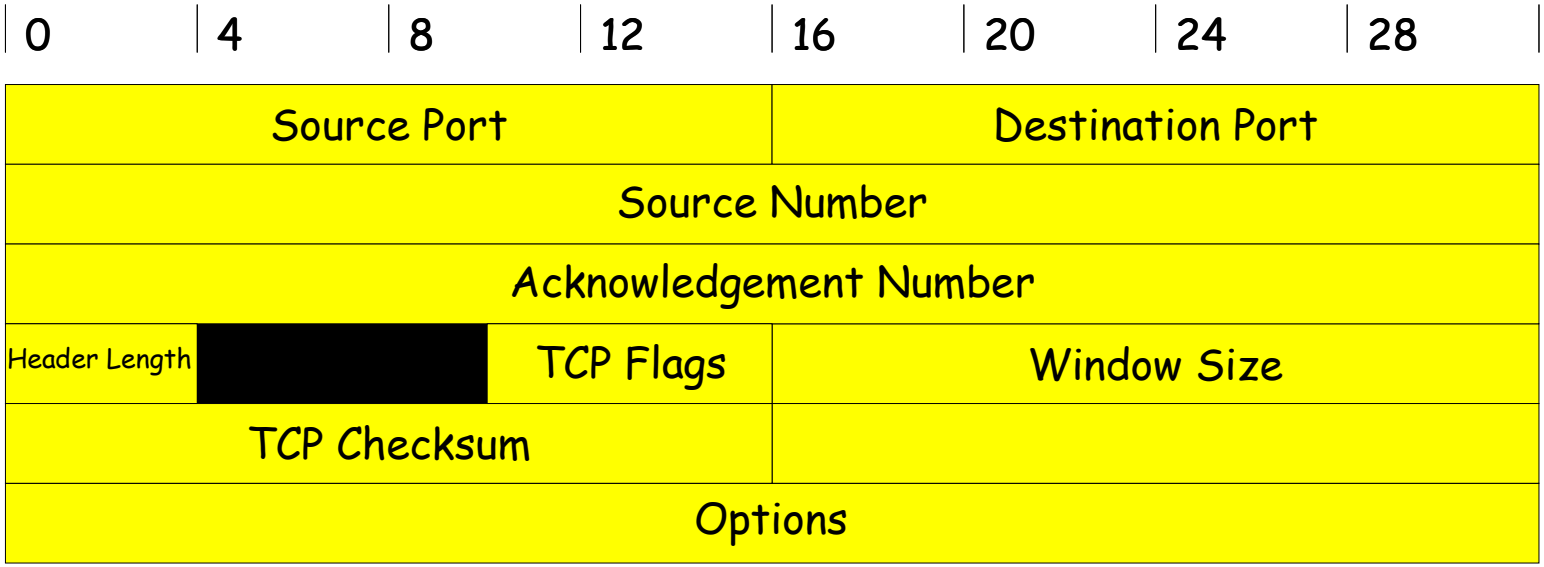
UDP: i problemi

- Assenza di correzione di errore
  - Tutti i controlli sul flusso sono a carico del livello applicativo
- Assenza controllo di flusso
  - Implicazione di sicurezza: IP spoofing
  - Semplice per attaccante sostituirsi ad host
    - Host A effettua query a host B
    - Cattivo (C) risponde ad A ed effettua DoS contro B
- Consigliabile filtrare servizi non essenziali

Richiami di reti (6)

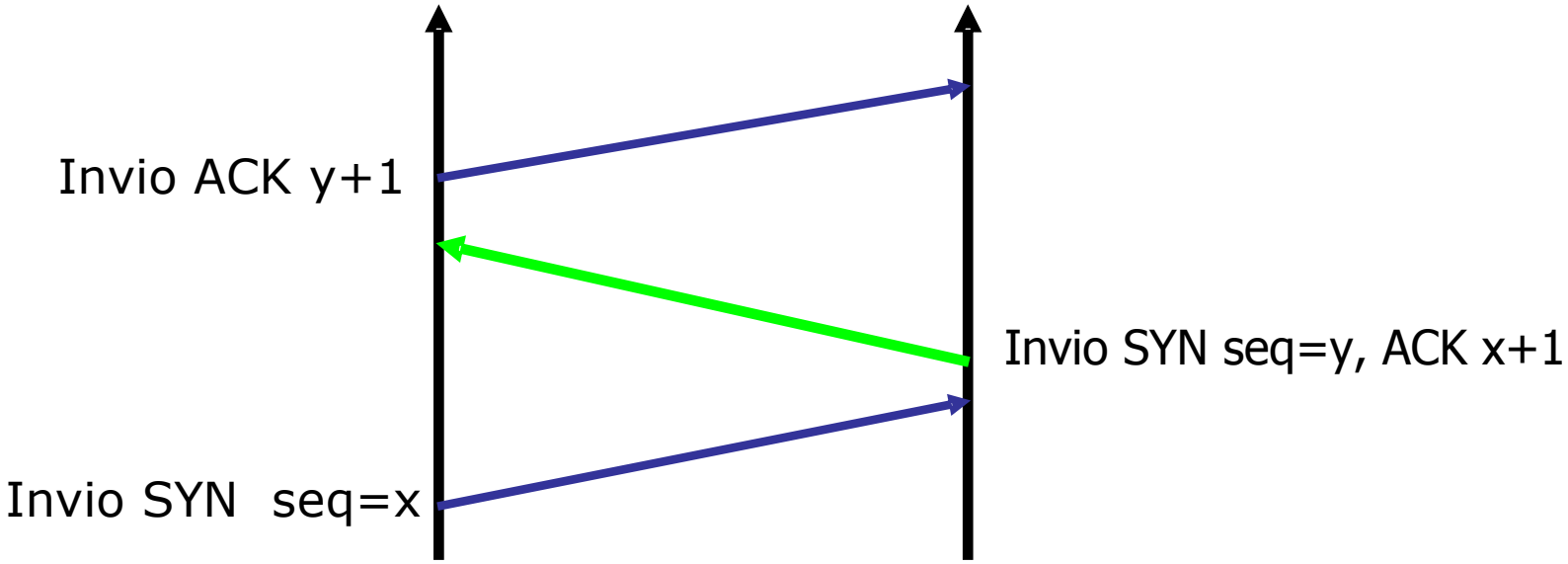
Transmission Control Protocol

- Protocollo connection-oriented
  - controllo del flusso
  - affidabilità della consegna
  - spoofing più difficile (ma non impossibile!)
- Connessione virtuale tra due socket
  - connessione in chiaro



Richiami di reti (7)

Three way handshake



TCP sequence guessing

- Le connessioni TCP utilizzano un numero di sequenza per riordinare i pacchetti.
- Ad ogni nuova connessione viene utilizzato un numero di sequenza (semi-)casuale.
- Se l'attaccante riesce a predire il numero di sequenza, può generare dei pacchetti con mittente falsificato formalmente corretti (anche se, ovviamente, non riesce a vedere le risposte).
- Perché l'attacco vada a buon fine è necessario che il mittente (vero) non riceva i pacchetti di risposta (o non sia in grado di reagire).
- Come protezione, i router non devono far entrare traffico che risulti proveniente dalla rete interna (*ingress filtering*).

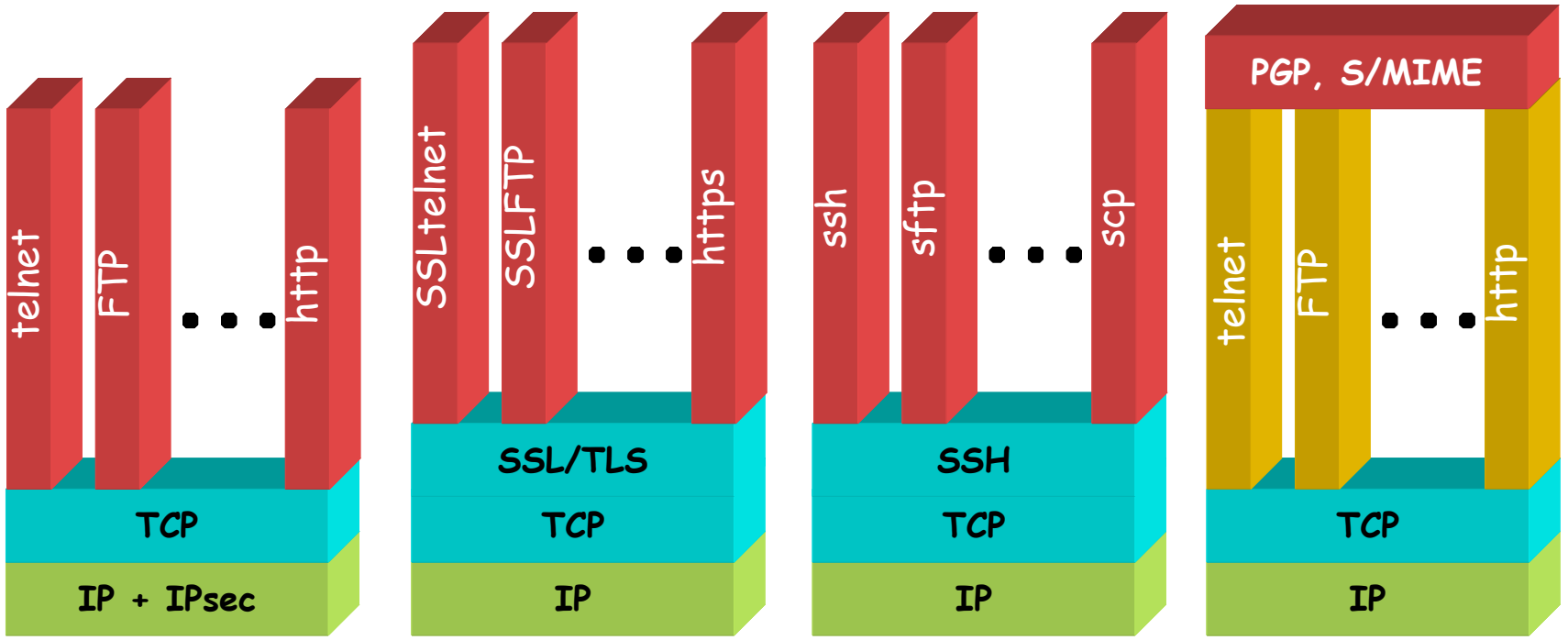
TCP session hijacking

Session Hijacking è l'inserimento in una sessione TCP attiva.

- Spiando una connessione attiva è possibile sostituirsi ad uno dei due interlocutori.
  - **C** spia la connessione tra **A** e **B** e registra i numeri di sequenza dei pacchetti
  - **C** blocca **B** (ad es. via SYN Flood): l'utente in **B** vede interrompersi la sua sessione interattiva
  - **C** invia pacchetti con il corretto numero di sequenza, *con mittente B*, in modo che **A** non si accorga di nulla.
- *hunt* o *dsniff* automatizzano il processo
- I router non devono far entrare traffico che risulti proveniente dalla rete interna (*ingress filtering*).
- Per attaccanti sulla rete interna l'unica difesa è la cifratura dei pacchetti

Protocolli (1)

La criptazione vista finora (PGP, X.509) fornisce sicurezza per una specifica applicazione. Come generalizzare?



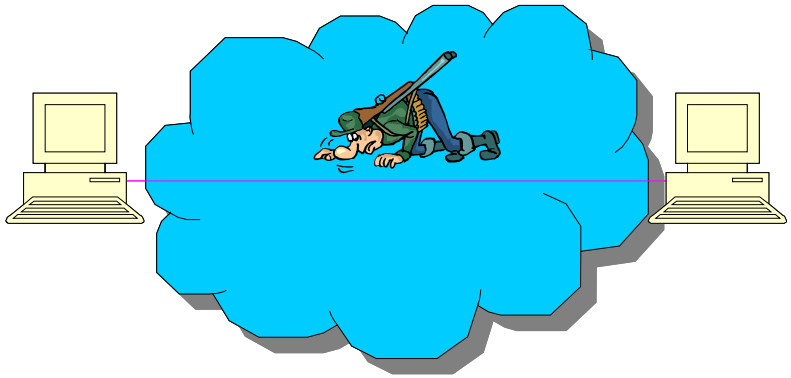
Protocolli (2)

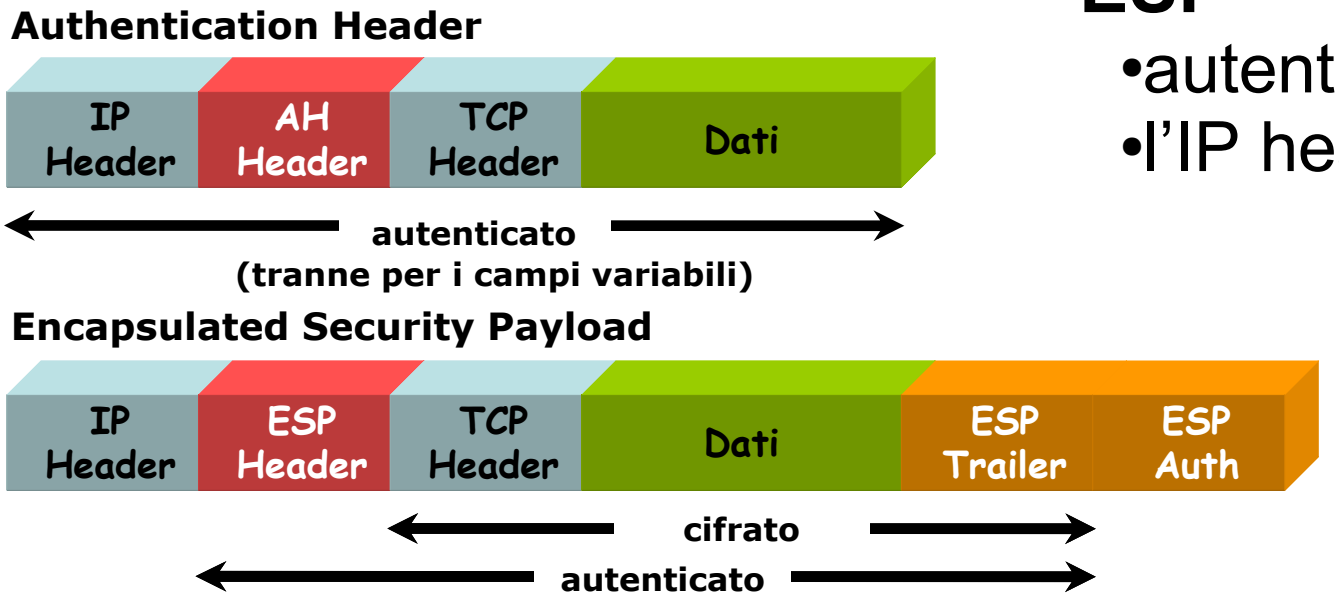
- **Network level**
  - trasparenti per le applicazioni, che non devono essere modificate
  - il network layer deve essere modificato
- **Transport level**
  - viene fornita una libreria di funzioni che può essere utilizzata dai progammi applicativi
  - necessaria la modifica dei programmi applicativi
- **Application level**
  - trasparenti per la rete
  - i servizi di sicurezza devono essere individualmente inclusi in ogni applicazione
  - necessaria la modifica dei programmi applicativi

Ipsec (1)

- Autenticazione e cifratura al livello Network
  - Authentication Header (AH)
- autentifica ogni pacchetto;
  - Encapsulating Security Payload (ESP)
- cifra i dati in ogni pacchetto;
  - Internet Key Exchange (IKE)
- protocollo di negoziazione
  - metodi di autentificazione
  - metodi di cifratura
- consente scambio sicuro di chiavi

Ipsec (2)





AH

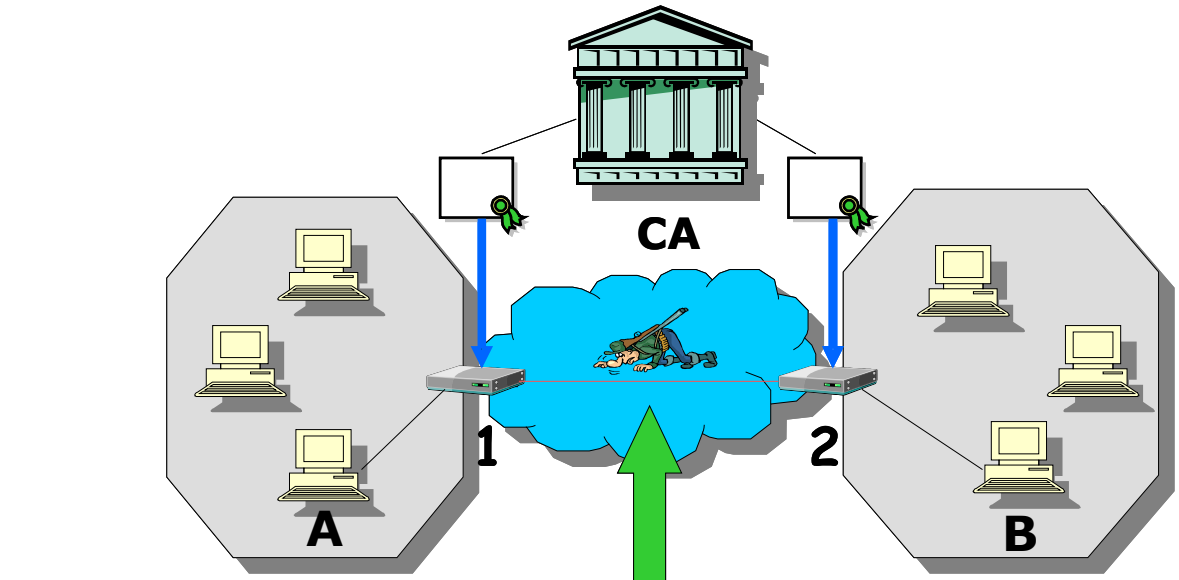
- campi variabili: TOS, Frg offset, TTL, …

ESP

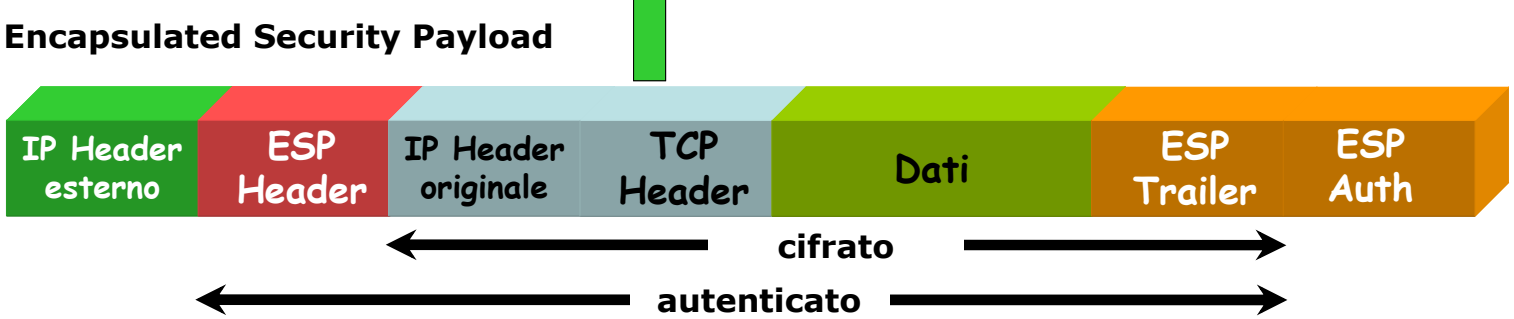
- autentificazione facoltativa
- l'IP header non è protetto

Ipsec (3)

Ipsec Tunnel



Gli indirizzi di A e B non sono visibili all'esterno dei gateway 1 e 2



SSH

- Rimpiazza telnet e i comandi “r” (rlogin, rshell)
  - Versione 1 e Versione 2 (IETF SECSH Working Group);
    - nella versione 1 richiede una distribuzione “manuale” delle chiavi di host e utenti;
  - connessione crittografata: protegge da:
    - IP spoofing;
    - IP source routing;
    - DNS spoofing;
    - sniffing;
    - man-in-the-middle;
  - tunneling traffico tcp e X11;
  - compressione dei dati (facoltativa, utile per connessioni lente).
    - Molto apprezzato anche dagli hacker, che spesso lo installano sulle macchine compromesse, perché impedisce agli amministratori di capire quali dati vengano trasferiti e per il meccanismo di port forwarding.

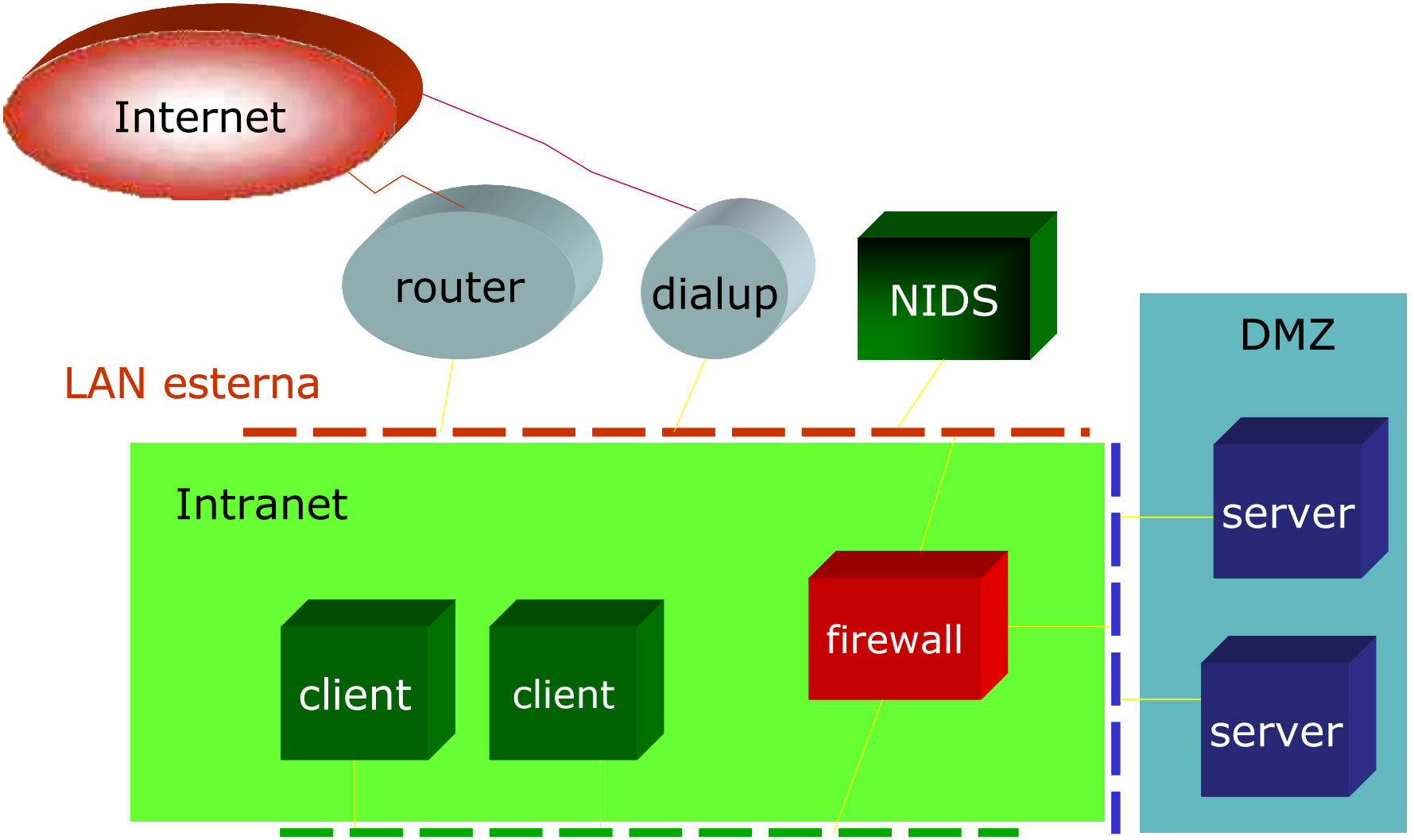
SSL/TLS

- Secure Sockets Layer (SSL) sviluppato da Netscape Communications
  - L'ultima versione (V3.0) è del Marzo 1996;
  - Netscape Communicator, Internet Explorer.
- Transport Layer Security (TSL) Working Group (IETF)
  - versione 1.0 del Gennaio 1999 (RFC 2246).
- Utilizza certificati X.509
- Può essere usato per ogni applicazione TCP (ad es. HHTP, Telnet, FTP, POP3, IMAP)
  - usato da praticamente tutti i server web "sicuri": https://…..
  - le vecchie applicazioni "insicure" possono essere usate in modalità tunnel (ad es. stunnel: http://www.stunnel.org)
- Non interagisce bene con firewall/proxy (man-in-the-middle)

S/MIME

- Permette di inviare e-mail MIME firmati e crittografati
- Supportato (tra gli altri) da *Communicator* e *Outlook*
- Utilizza certificati X.509

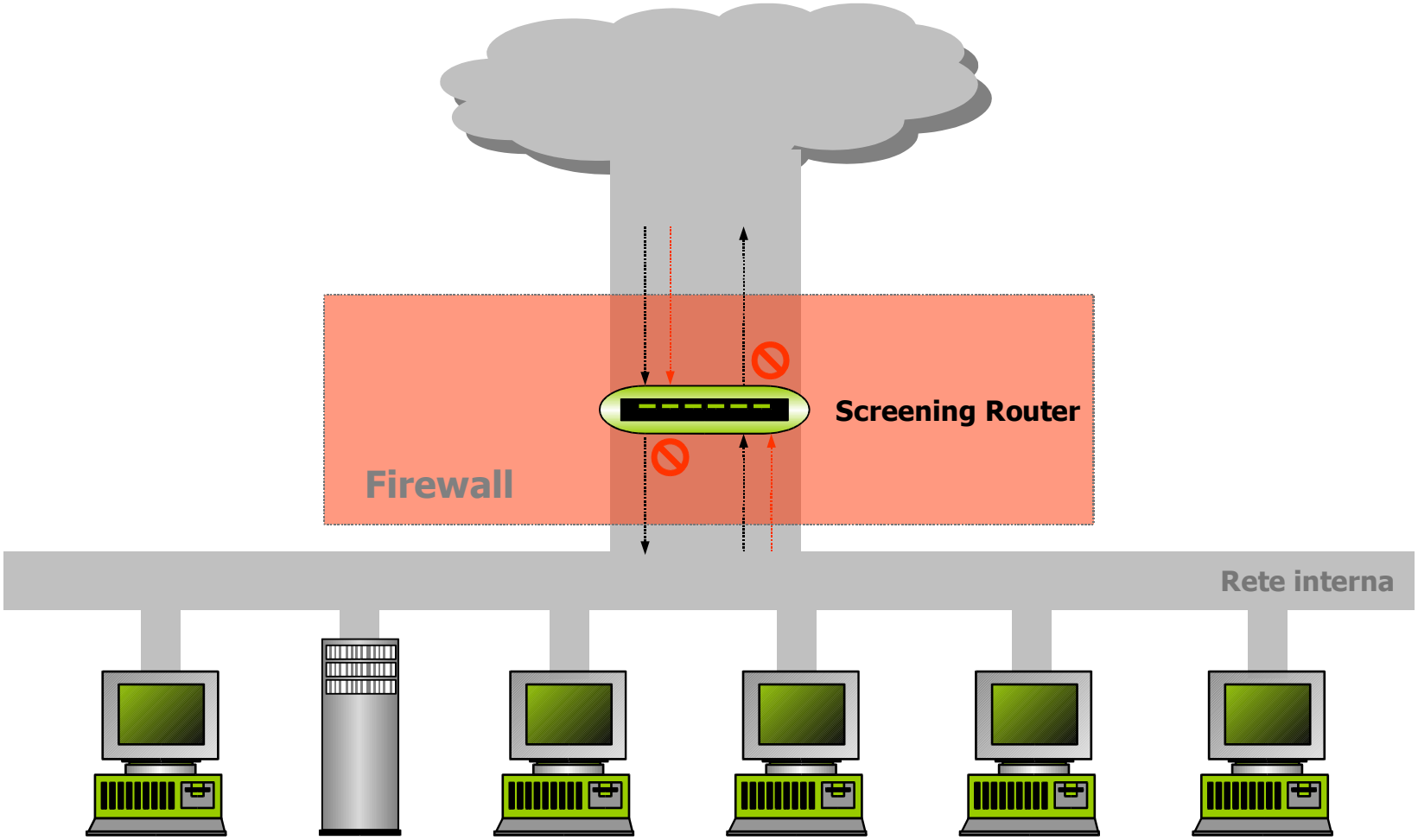
Internet e intranet



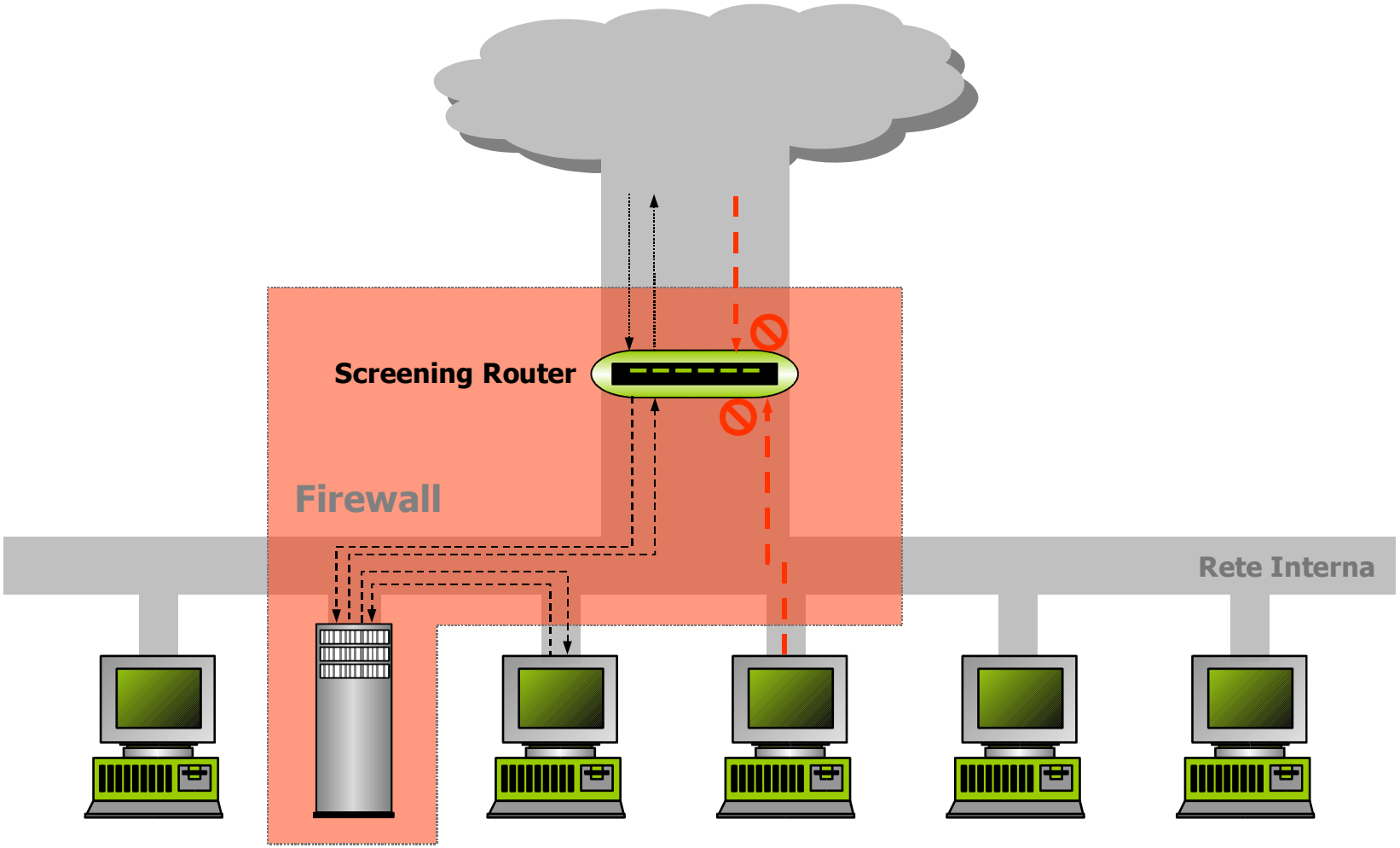
Firewall

- Network device con almeno 2 interfacce di rete
  - Router o hardware dedicato
- Separa zone amministrativamente diverse
  - LAN esterna (insicura)
  - DMZ
  - Intranet
- Effettua routing fra le diverse zone
- Può effettuare mascheramento indirizzi (NAT)
- Filtra traffico fra le diverse zone tramite regole predefinite
  - Router con filtri è un firewall!
- Può mediare accessi a specifiche applicazioni
  - Proxy server

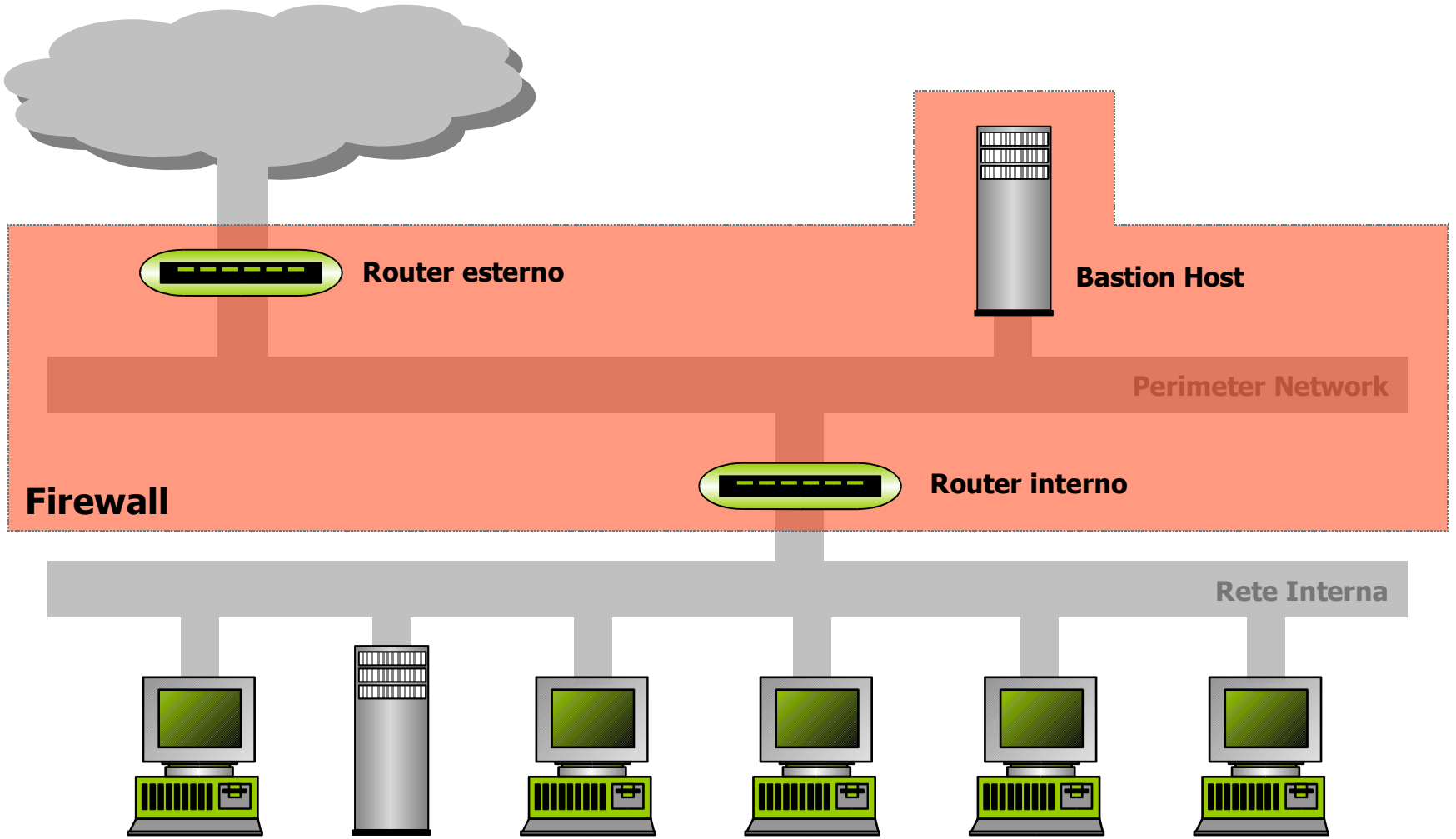
Architetture: screening router



Architetture: screened host



Architetture: screened subnet



Firewall: pro e contro

1. Sono troppo complessi
2. I nuovi servizi sono incompatibili con i FW
3. Sono di difficile gestione, troppo costosi e rapidamente obsoleti
4. Bloccano servizi utili e riducono l'utilità di Internet
5. Ad ogni nuovo servizio il FW deve essere riconfigurato
6. Il nuovo sw ha incluso il supporto per la sicurezza (ad es. SSL)
7. Riducono le prestazioni in modo eccessivo
8. I servizi che vengono fatti passare possono avere problemi di sicurezza
9. Non è in grado di bloccare i virus o applet ostili
10. IPV6 renderà i FW obsoleti

1. Il sw disponibile non è ancora di qualità adeguata
2. In molti casi è l'unica protezione possibile
3. Il costo di un incidente di sicurezza è ben maggiore
4. Se le applicazioni sono scritte bene i FW non sono di intralcio
5. In questo modo ogni novità rimane sotto controllo
6. Il nuovo sw spesso è meno sicuro di quello vecchio
7. È solo questione di comprare hw più veloce
8. Anche le macchine interne devono essere ben configurate
9. I FW non sono lo strumento adatto per bloccare virus e applet
10. IPV6 è ancora lontano.

Firewall: limitazioni

- Scarsa protezione dagli attacchi dall'interno (e dagli utenti...)
- È sempre possibile un accesso non autorizzato alla rete esterna (ad es. via modem)
- Scarsa o nessuna protezione da virus, applet Java, e controlli ActiveX
- Si possono utilizzare tunnel per evitare il blocco di alcuni protocolli

Packet filter

Un device (ad es. un router) che controlla ogni pacchetto IP in arrivo per decidere se inoltrarlo o no.

I filtri si basano sulle seguenti informazioni

- semplici
  - protocollo;
  - porte;
  - indirizzi sorgente e destinazione;
  - flag e opzioni tcp;
  - non prendono in considerazione la parte dati;
  - prendono le decisioni pacchetto per pacchetto;
- *stateful* (dinamici)
  - tengono traccia delle connessioni in corso e possono avere conoscenza dei protocolli più comuni.
  - la necessità di tenere traccia delle connessioni aperte, ovviamente, aumenta il lavoro del filtro

Bastion host

È il sistema a cui si devono connettere gli utenti esterni per accedere alle macchine e ai servizi locali.

È particolarmente esposto agli attacchi e quindi deve essere molto curato dal punto di vista della sicurezza

- va previsto il caso che il Bastion Host venga compromesso: in altri termini, le macchine interne non devono fidarsene più del minimo indispensabile.
  - la rete su cui si trova non dovrebbe trasportare traffico confidenziale (potrebbero installare uno sniffer): dovrebbe essere isolata dalla rete interna da un packet filter.
  - in alternativa potrebbe essere su di uno switch (ma attenzione, anche gli switch permettono snooping! e anche il traffico multicast potrebbe essere di interesse per un intruso)