Attacchi/2

Qualche esempio delle tecniche usate per violare le reti, e relative contromisure

Marco Prandini

Sicurezza dei protocolli di rete

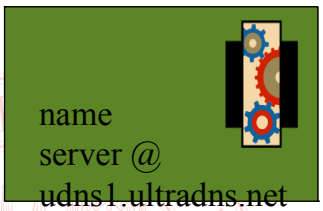
- Alcuni esempi di attacchi portati attraverso i protocolli di rete
  - Classico attacco: hijacking, cioè dirottamento della connessione perché passi attraverso i sistemi dell'attaccante
  - sfruttando i protocolli applicativi
    - direttamente (Es. HTTP attraverso manipolazione del browser)
    - attraverso i protocolli ausiliari (es. DNS)
  - sfruttando la mancanza di sicurezza di IP stesso

DNS spoofing

- Query e risposta

www.amazon.com?

207.171.166.48



DNS spoofing

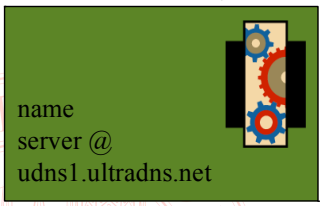
- Risposta falsificata

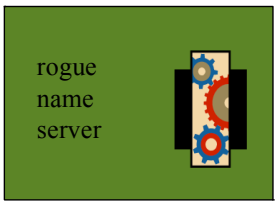
www.amazon.com?

X

207.171.166.48

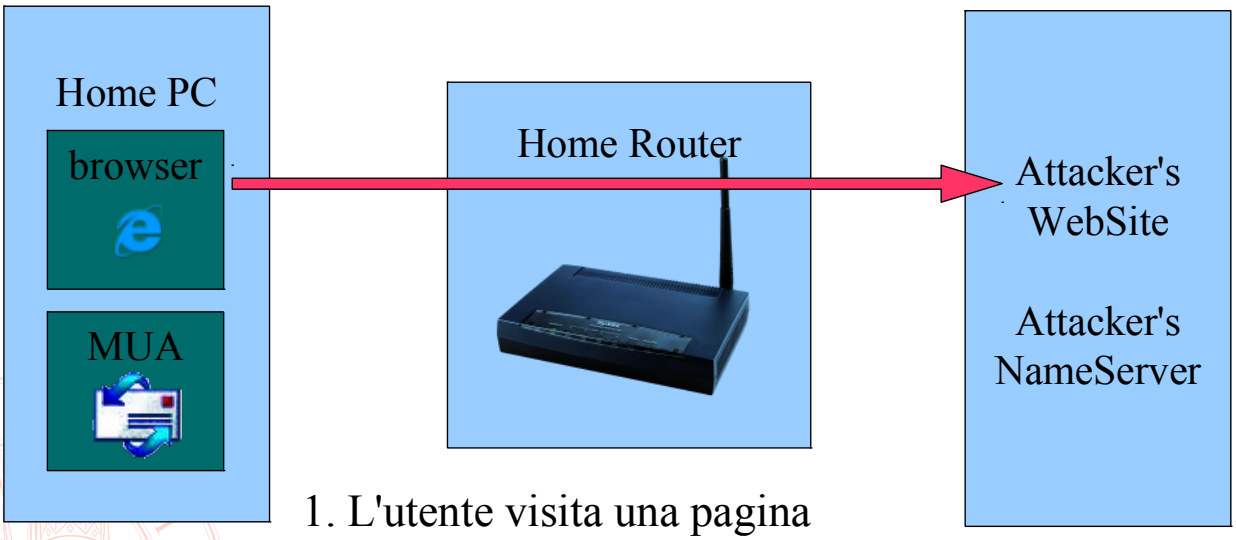
cracker address





DNS spoofing (pharming)

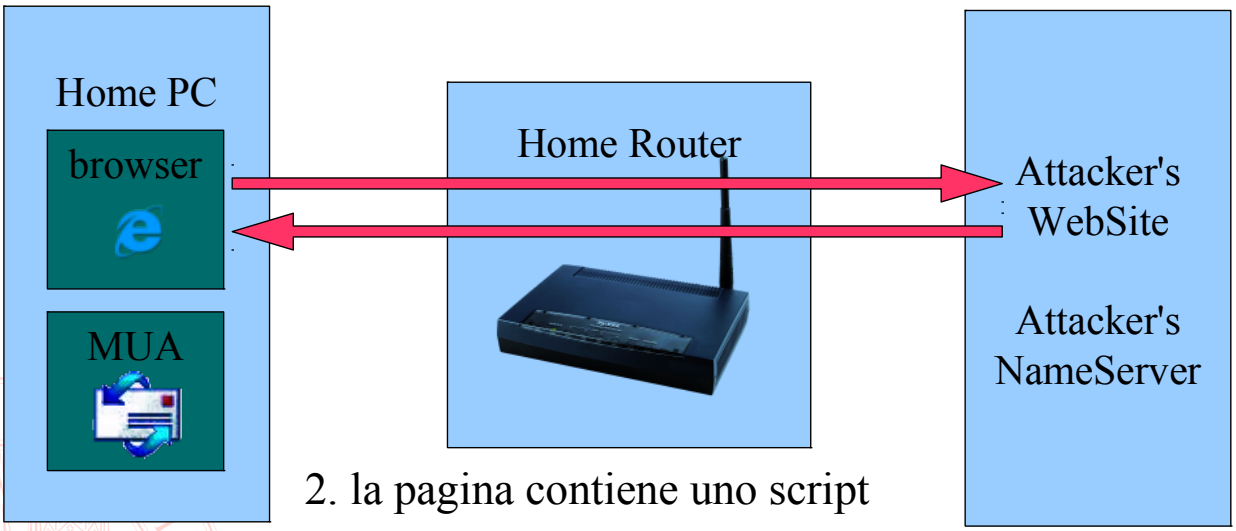
- Sembra difficile falsificare una risposta DNS?



1. L'utente visita una pagina HTML, consapevolmente o no

DNS spoofing (pharming)

- Sembra difficile falsificare una risposta DNS?

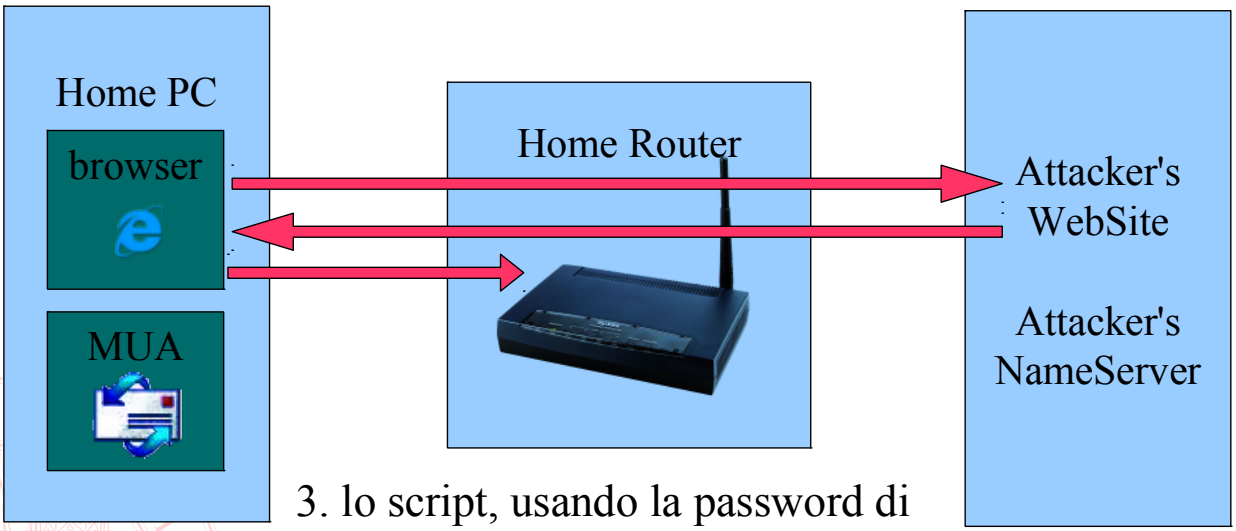


2. la pagina contiene uno script

HTML, consapevolmente o no

DNS spoofing (pharming)

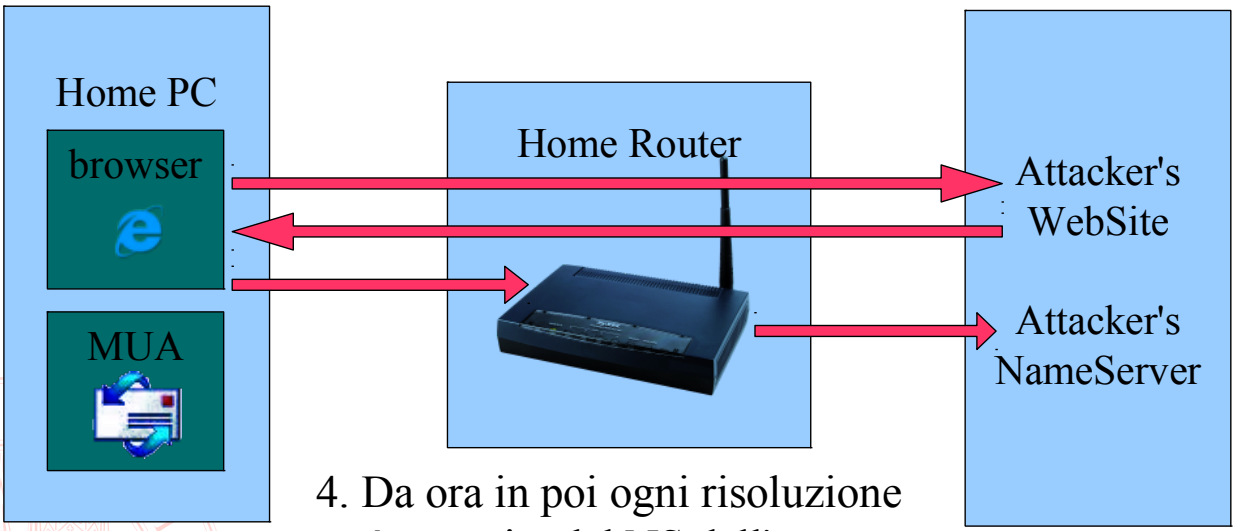
- Sembra difficile falsificare una risposta DNS?



3. lo script, usando la password di default del router, riprogramma il DNS

DNS spoofing (pharming)

- Sembra difficile falsificare una risposta DNS?

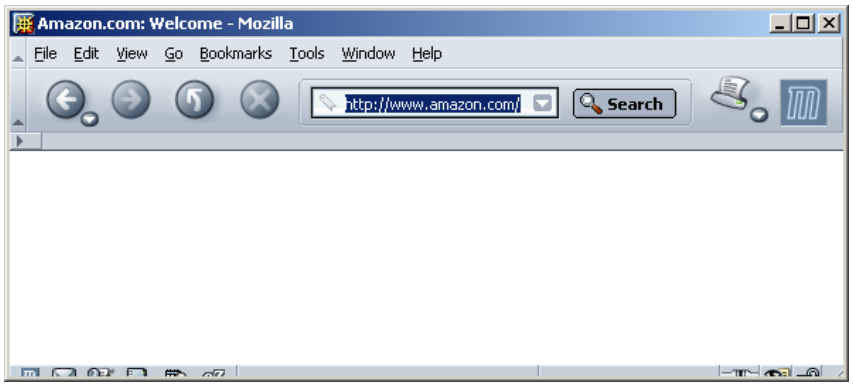


4. Da ora in poi ogni risoluzione sarà eseguita dal NS dell'attaccante

Contromisure e contro-contromisure

- HTTPS permette di bloccare questi attacchi
- … ma esistono modi
  - per evitare che venga visualizzata l'URL effettivamente visitata
  - o per far accettare al browser qualsiasi certificato

HTTPS

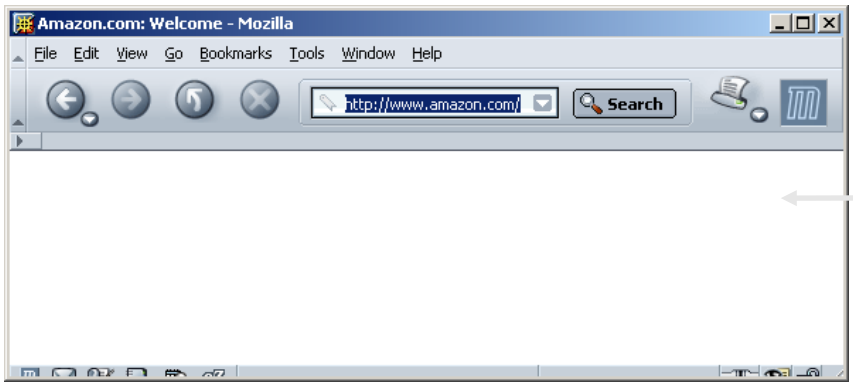


"Provami che hai la chiave privata di amazon"

amazon.com.

Web server @ crackeraddress

HTTPS



"Provami che hai la chiave privata di amazon"

amazon.com.

Web server @ crackeraddress

HTTPS

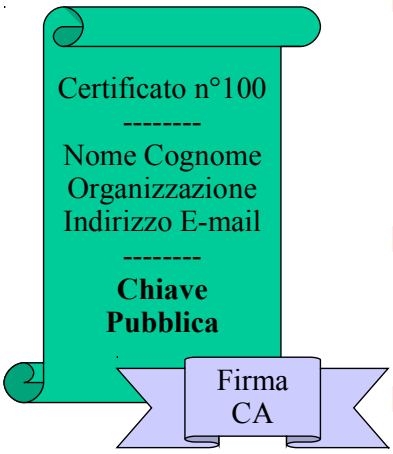
"Provami che hai la chiave privata di amazon"

amazon.com.

Web server @ crackeraddress

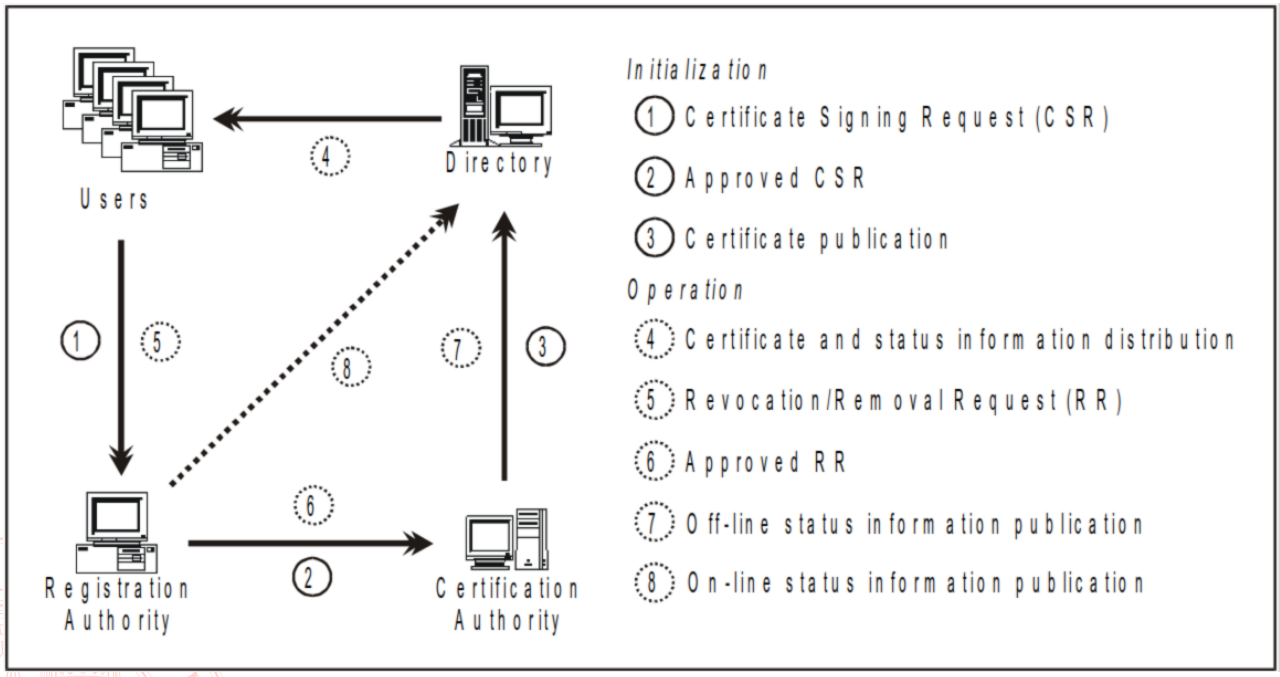
- **Come fa il browser a verificare la prova fornita dal web server?**
  - **Certificate store**
  - **Trusted CAs**

Certificazione della chiave pubblica

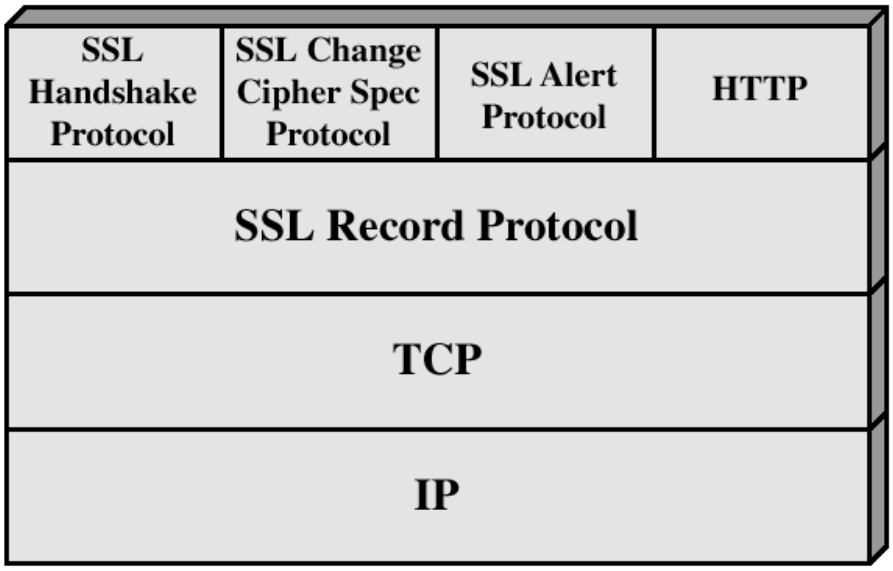


- Certificato X.509
  - Associa chiave e titolare
  - Autenticità e integrità garantite dalla firma digitale di una terza parte fidata (Certification Authority)
- Per verificare la firma serve la chiave pubblica della CA
  - Chi ci garantisce che questa sia autentica?
- Serve una Root of Trust

PKI – ciclo di vita dei certificati

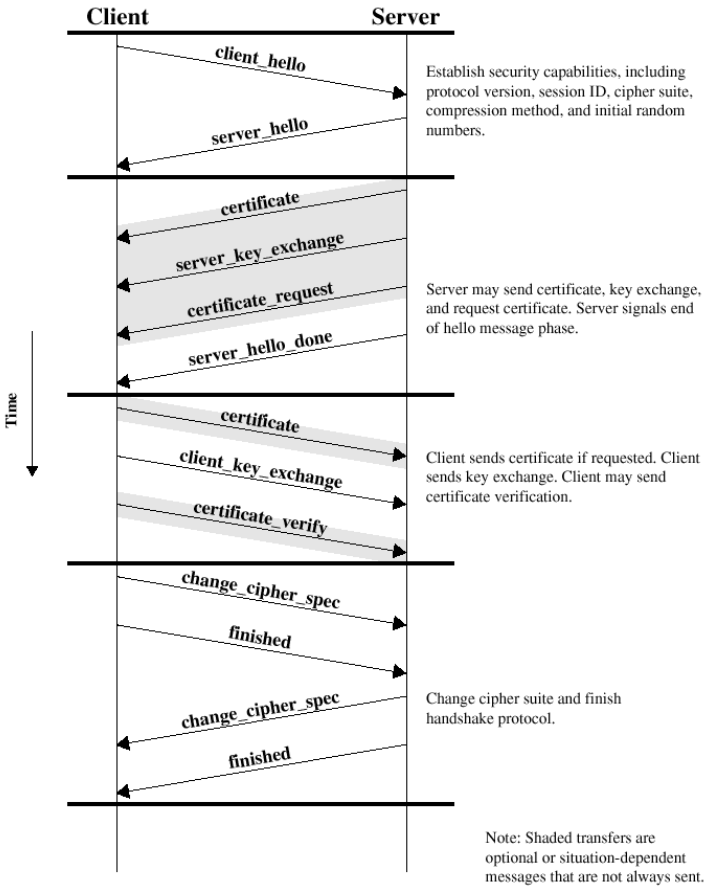


Architettura di SSL



Handshake Protocol

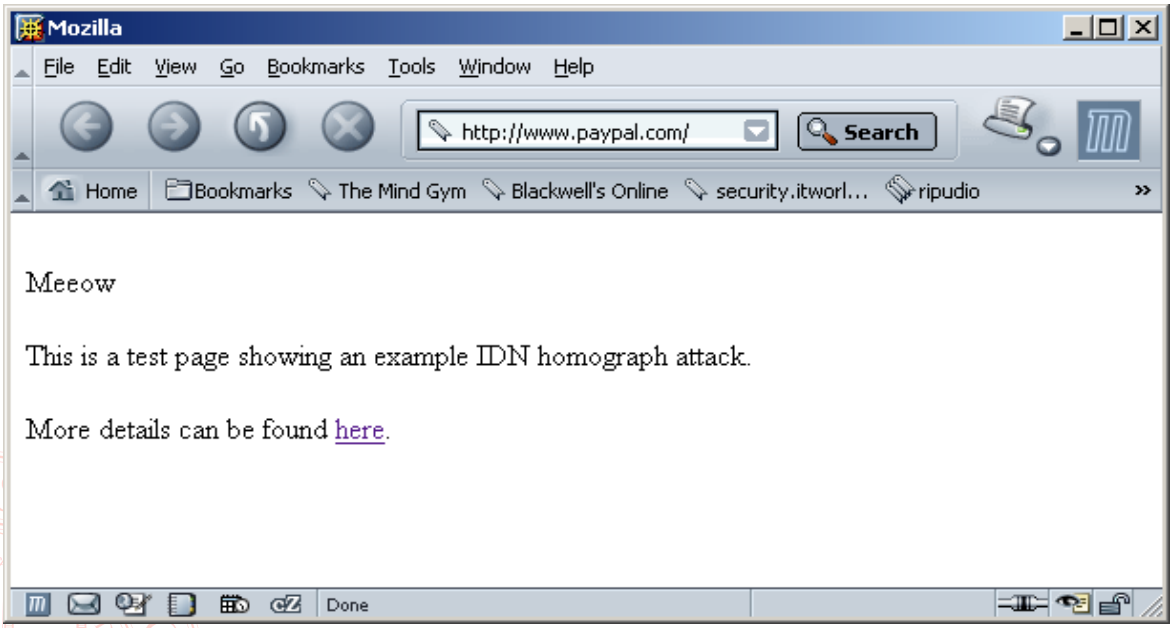
- La parte più complessa di SSL.
- Consente al server ed al client di autenticarsi reciprocamente
  - nelle applicazioni web è comune che il server provi la sua autenticità edntre il client no
- Negozia gli algoritmi e le chiavi per la cifratura ed i controlli di integrità
- Interviene prima che qualsiasi dato sia trasmesso



Occultamento dell'URL

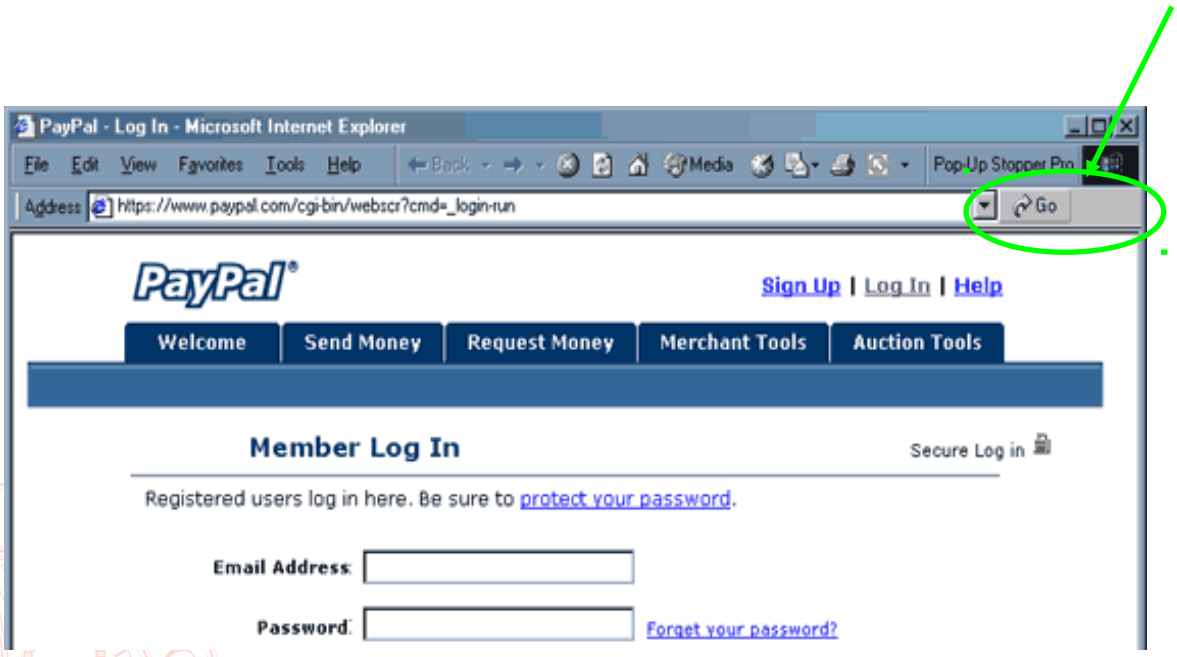
RFC3490/1/2: International Domain Names

ad esempio: http://www.pаypal.com/

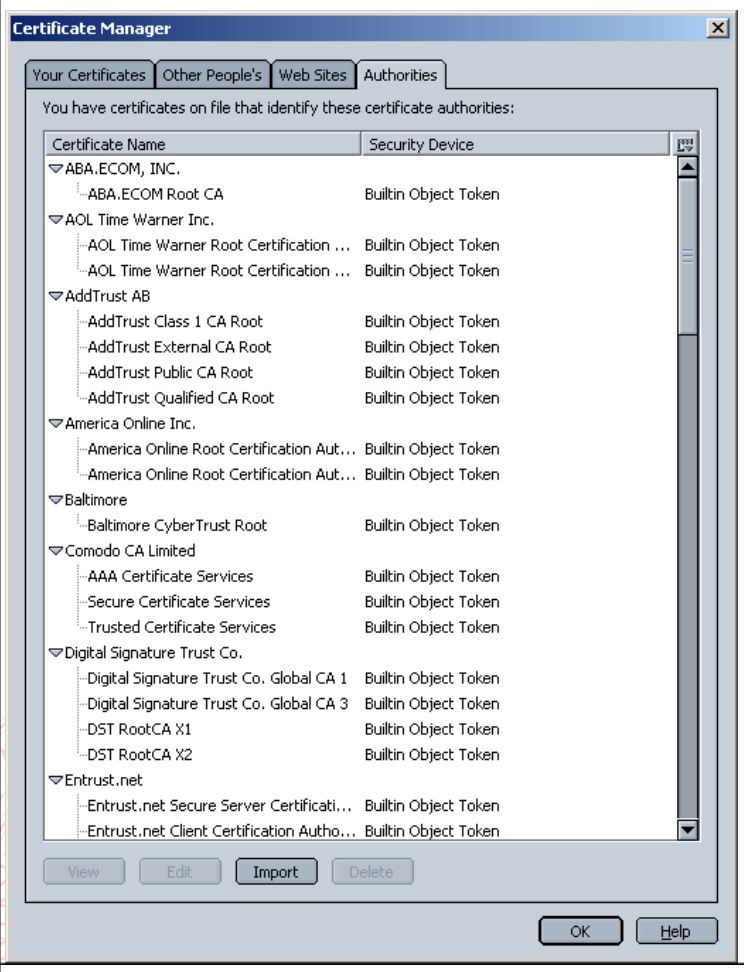


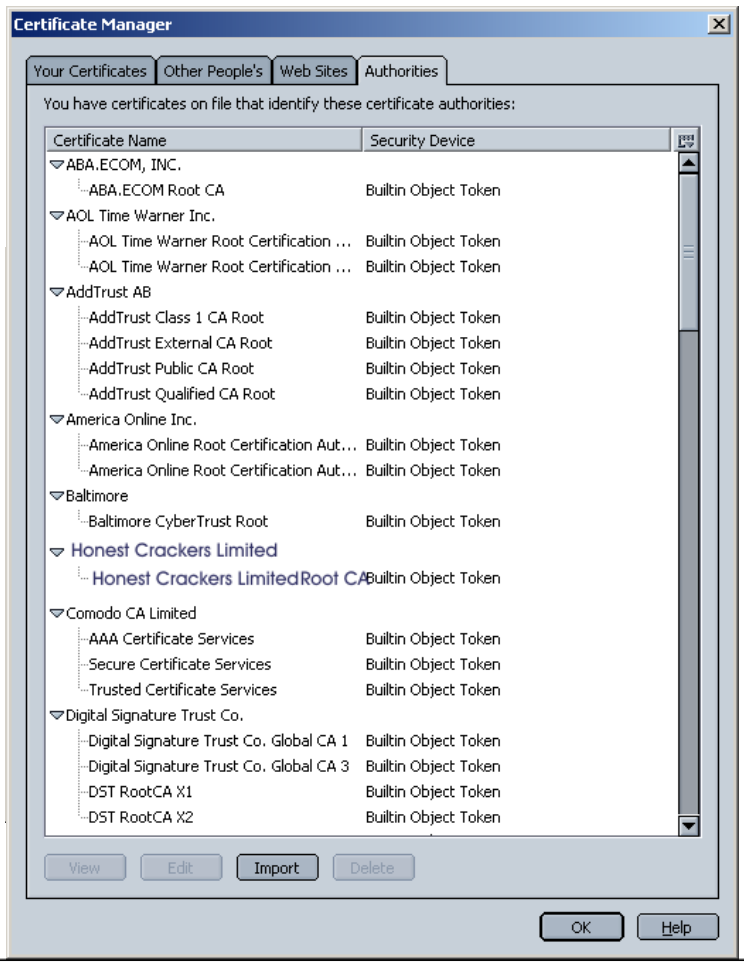
Occultamento della barra degli indirizzi

Qualche riga di codice js o activeX



Iniezione di CA nel certificate store



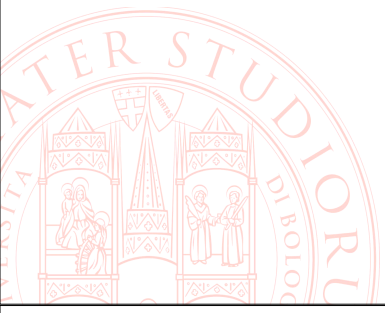


Vulnerabilità di SSL – a livello di protocollo

- DROWN (2016) - https://drownattack.com/
  - Gravi vulnerabilità note nella vecchia versione SSLv2, originata dalle restrizioni imposte dal governo USA all'esportazione di crittografia forte
    - Possibile inviare probe che limitano lo spazio di ricerca delle chiavi a 40 bit
  - Se tale versione è supportata su un server con una certa chiave privata, tutti i server che usano tale chiave sono vulnerabili
  - Impatto: controllo completo, impersonamento del server
- POODLE (2014)
  - Un attaccante in grado di posizionarsi *in the middle* (ad esempio contro gli utenti di un hotspot pubblico) può forzare il downgrade delle connessioni verso SSLv3
  - SSLv3 ha varie vulnerabilità sfruttabili
  - Impatto: controllo completo della connessione

Vulnerabilità di SSL – a livello di implementazione

- Heartbleed (2014) - http://heartbleed.com/
  - Implementazione errata della rinegoziazione delle chiavi
  - Consente di leggere pezzi di memoria del sistema target
  - Impatto: possibile leak di materiale sensibile, come le chiavi



Attacchi a livello IP

- IP non può garantire nessuna proprietà di sicurezza ...
  - autenticità
  - integrità
  - riservatezza
- … di nessuna parte del pacchetto
  - header
  - payload
- Esistono varianti che conferiscono queste proprietà, ma richiedono uno stack modificato

IP Hijacking

- Vari modi di *informare internet che la rotta verso una data subnet passa dal proprio AS*, attraverso il protocollo BGP
  - Autorità apparente di annunciare
  - Annuncio spontaneo (nessuno filtra!)
- Usi differenti:
  - Non malevolo: più veloce che chiedere IP al RIR :-)
  - Spamma e fuggi
  - DoS attivo o passivo
  - Impersonare un bersaglio
  - Man In The Middle
- Dirottamenti accidentali avvengono spesso: quindi basse probabilità di essere notati
- Qualche esempio storico disponibile su completewhois.com

Un esempio recente: Youtube & Pakistan Telecom

Dalla presentazione di Pilosov e Kapela a DEFCON16 (Las Vegas 2008)

- YouTube announces 5 prefixes:
- A /19, /20, /22, and two /24s
- The /22 is 208.65.152.0/22
- Pakistan's government decides to block YouTube
- Pakistan Telecom internally nails up a more specific route (208.65.153.0/24) out of YouTube's /22 to null0 (the routers discard interface)
- Somehow redists from static ✉ bgp, then to PCCW
- Upstream provider sends routes to everyone else…
- Most of the net now goes to Pakistan for YouTube, gets nothing!
- YouTube responds by announcing both the /24 and two more specific /25s, with partial success
- PCCW turns off Pakistan Telecom peering two hours later
- 3 to 5 minutes afterward, global bgp table is clean again

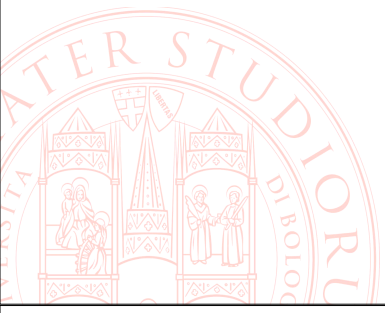
Link interessanti:

http://news.cnet.com/8301-10784_3-9878655-7.html

http://www.defcon.org/images/defcon-16/dc16-presentations/defcon-16-pilosov-kapela.pdf

Come risolvere il problema?

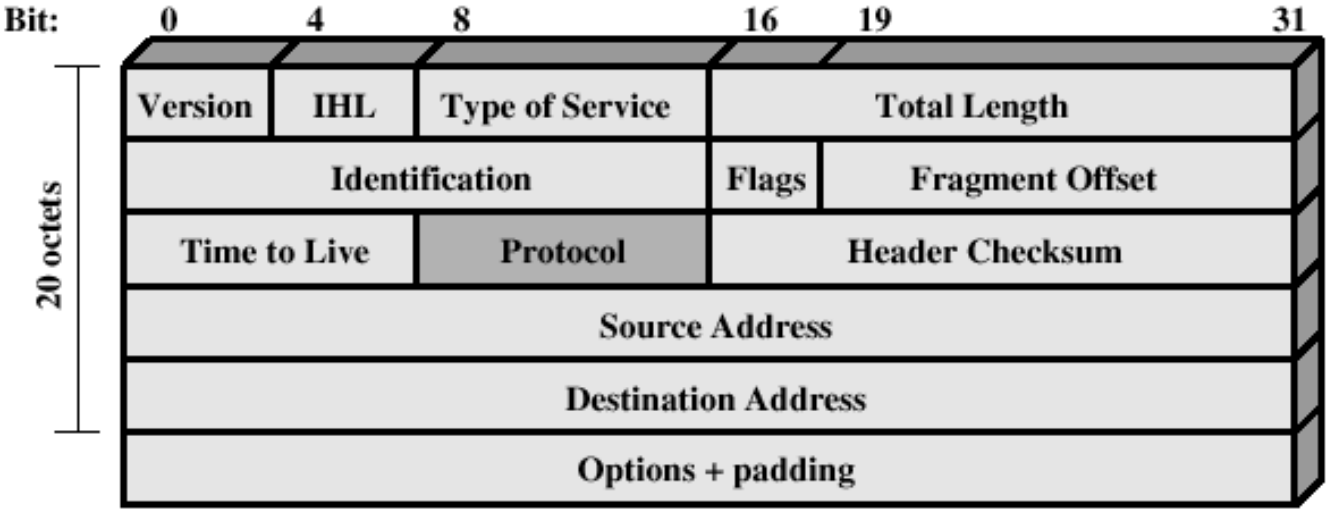
- Reagendo:
  - Per avere l'attenzione dei grossi provider upstream possono volerci giorni, se non siete Youtube
- Prevenendo:
  - Filtrando gli annunci sulla base del contenuto (come essere certi della ragionevolezza?)
  - Autenticando i singoli pacchetti: ad esempio con IPSec



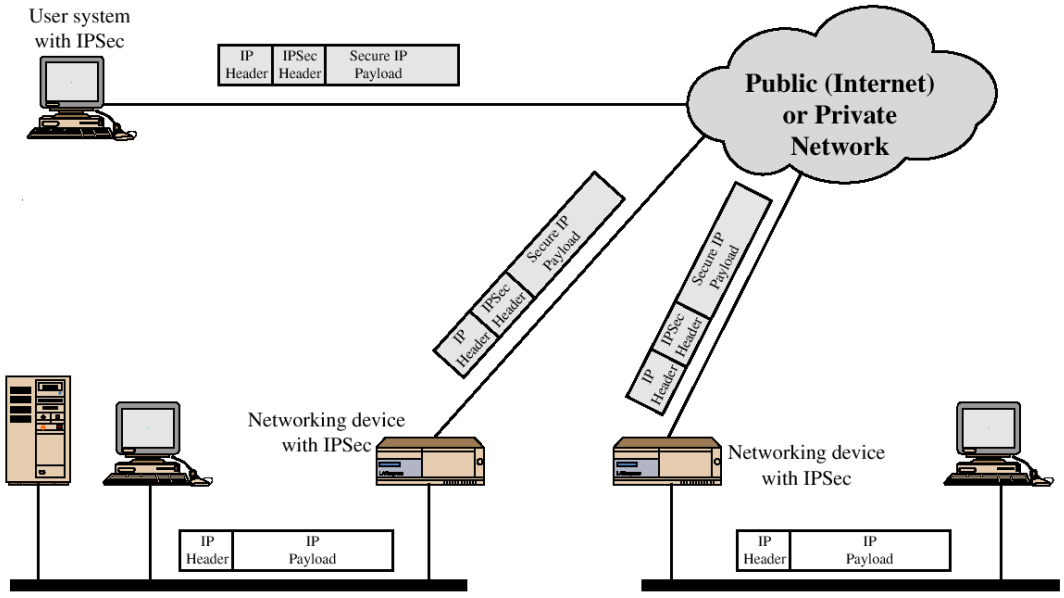
IP Security Overview

- IPSec non è un protocollo singolo
  - set di algoritmi di sicurezza
  - framework per la negoziazione degli algoritmi
  - specifiche per la gestione delle chiavi
- Applicazioni di IPSec
  - Interconnessione di sedi remote attraverso Internet
  - Accesso di client alla rete aziendale attraverso Internet
  - Creazione di reti complesse con criteri di protezione differenziati
- Vantaggi di IPSec
  - Trasparente alle applicazioni
  - Applicabile al traffico infrastrutturale di Internet, come i messaggi che i router si scambiano per aggiornare le tabelle di instradamento

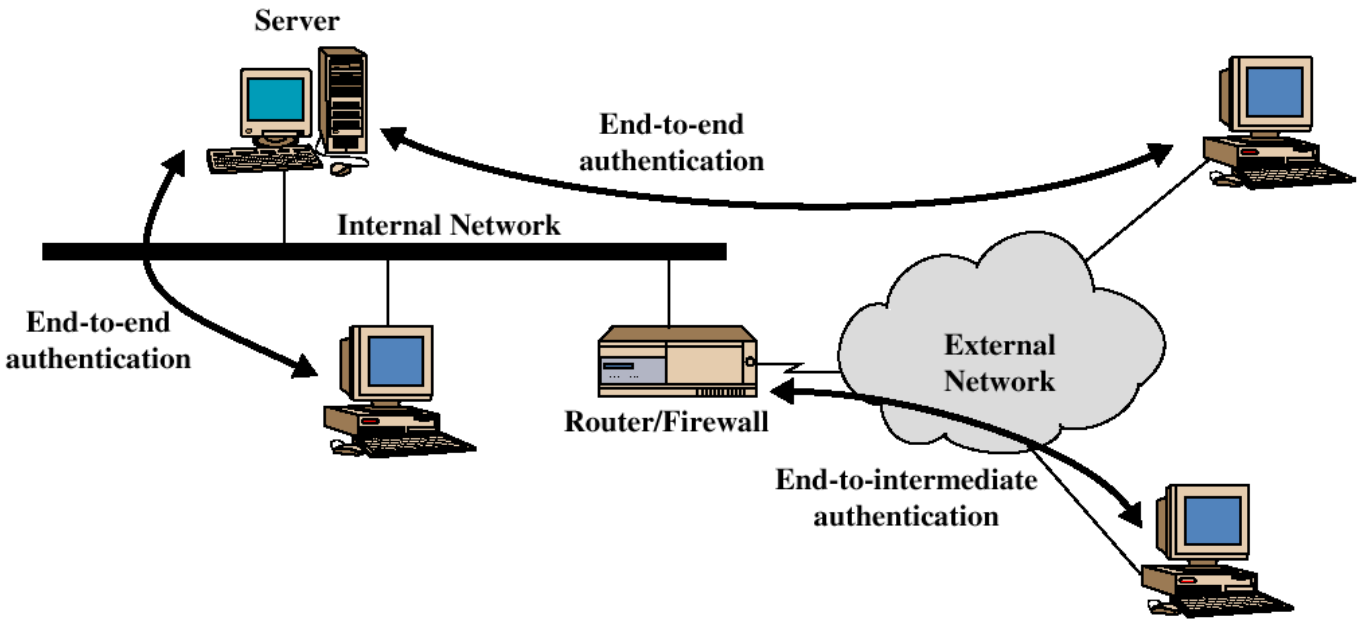
IPv4 Header



IPSec Scenario

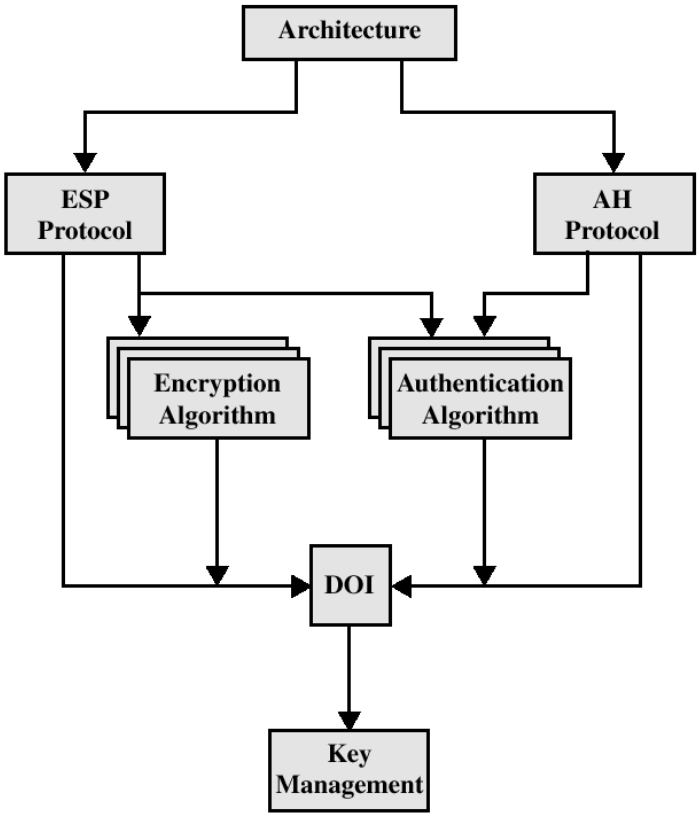


Utilizzo End-to-end / End-to-Intermediate



Gli standard di IPSec

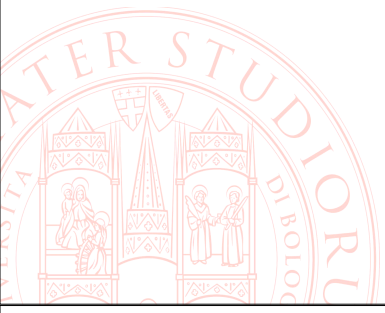
- IPSec documents:
  - RFC 2401: An overview of security architecture
  - RFC 2402: Description of a packet encryption extension to IPv4/IPv6
  - RFC 2406: Description of a packet emcryption extension to IPv4/IPv6
  - RFC 2408: Specification of key managament capabilities



Servizi offerti ed algoritmi utilizzati

- **Controllo dell'accesso**
- **Integrità anche senza connessione**
- **Autenticazione dell'origine dei dati**
- **Rilevazione dei replay**
- **Riservatezza dei dati**
- **Parziale riservatezza dei flussi di traffico**

- Cifratura:
  - Three-key triple DES
  - RC5
  - IDEA
  - Three-key triple IDEA
  - CAST
  - Blowfish
- Autenticazione:
  - HMAC-MD5-96
  - HMAC-SHA-1-96
- Gestione chiavi:
  - Manuale
  - Automatizzata
    - Oakley Key Determination Protocol
    - Internet Security Association and Key Management Protocol (ISAKMP)



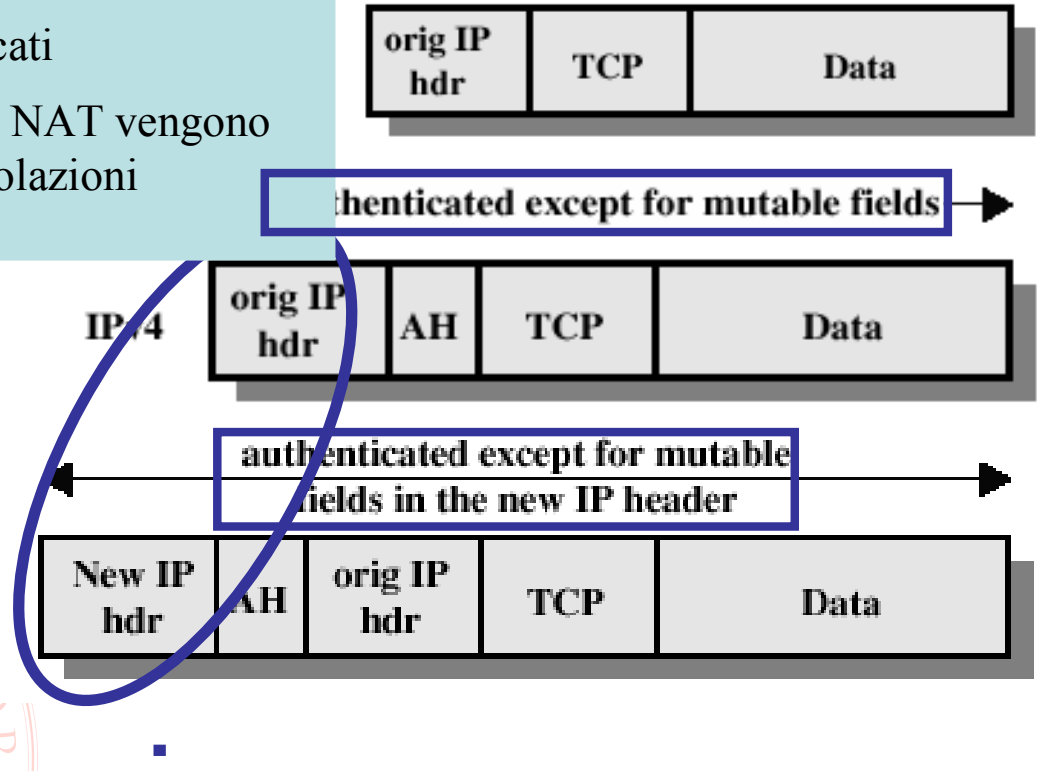
Terminologia di base

- **SA (Security Association)**
  - **relazione unidirezionale tra mittente e destinatario, definita da**
    - **Security Parameter Index (SPI)**
    - **IP Destination address**
    - **Security Protocol Identifier**
  - **due modalità possibili di SA**
    - **Transport Mode**
    - **Tunnel Mode**
- **Protocolli di sicurezza**
  - **AH (Authentication Header)**
  - **ESP (Encapsulating Security Payload)**

Autentication Header

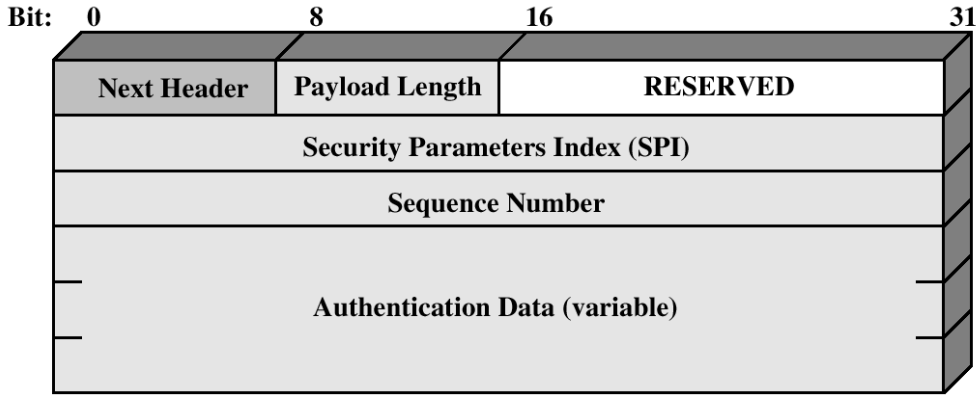
Gli indirizzi, giustamente, non sono considerati campi variabili

- vengono autenticati
- le alterazioni del NAT vengono percepite come violazioni dell'integrità



Authentication Header

- Garantisce l'autenticazione e l'integrità dei pacchetti IP
- Protegge dai replay attacks

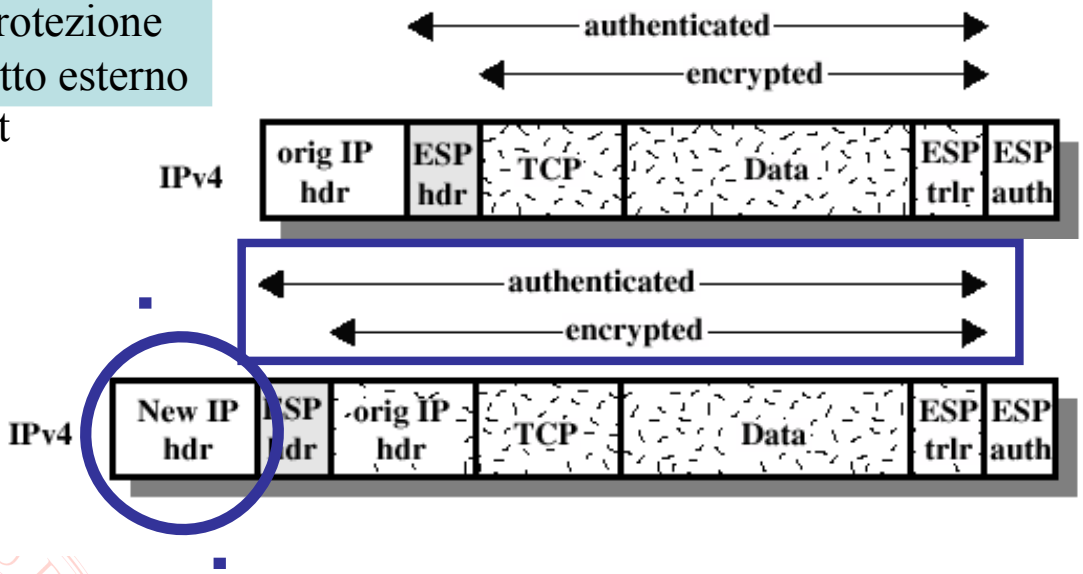


ESP con cifratura ed autenticazione

Nessuna protezione del pacchetto esterno

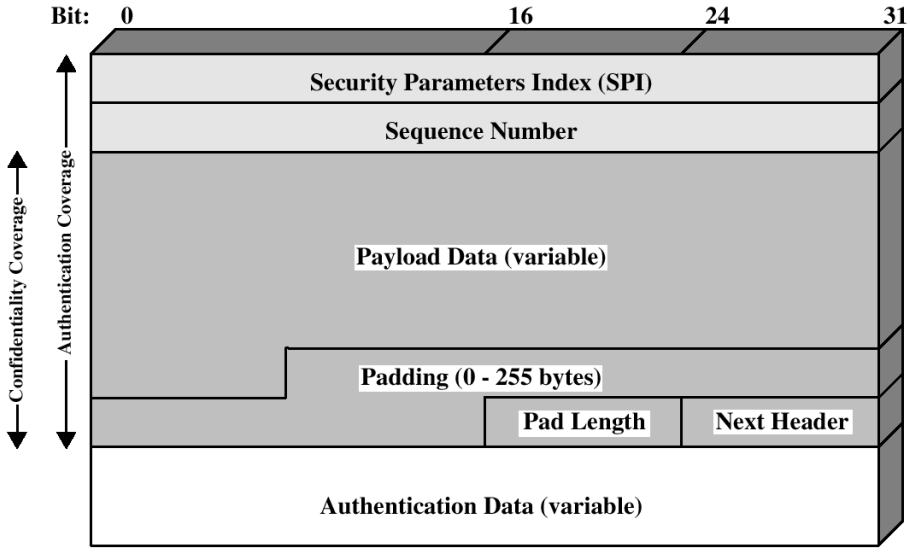
Transport Mode

Tunnel Mode



Encapsulating Security Payload

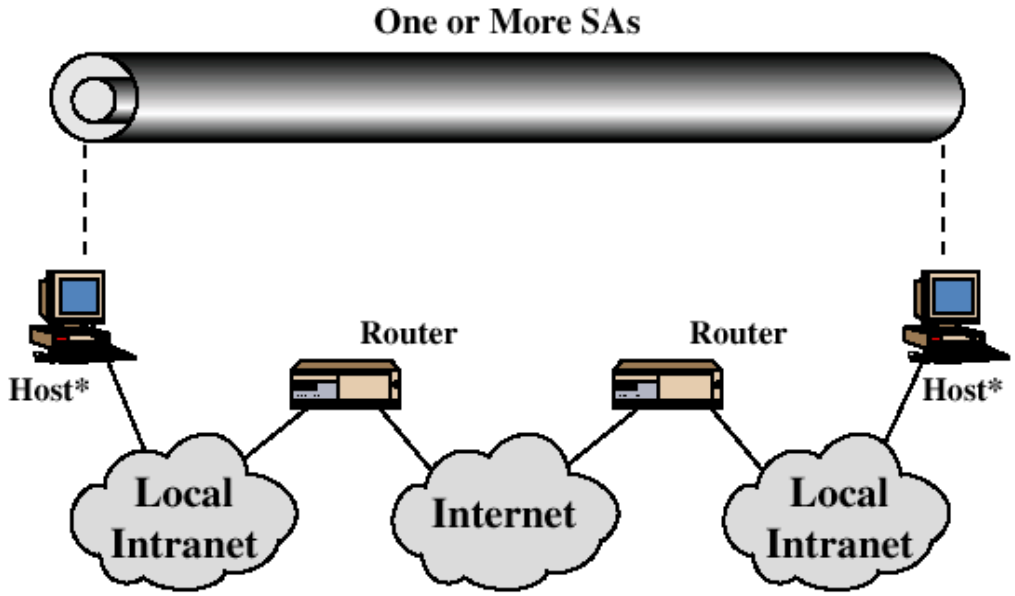
- ESP offre essenzialmente servizi per la riservatezza



Riassunto delle combinazioni dei modi di protezione

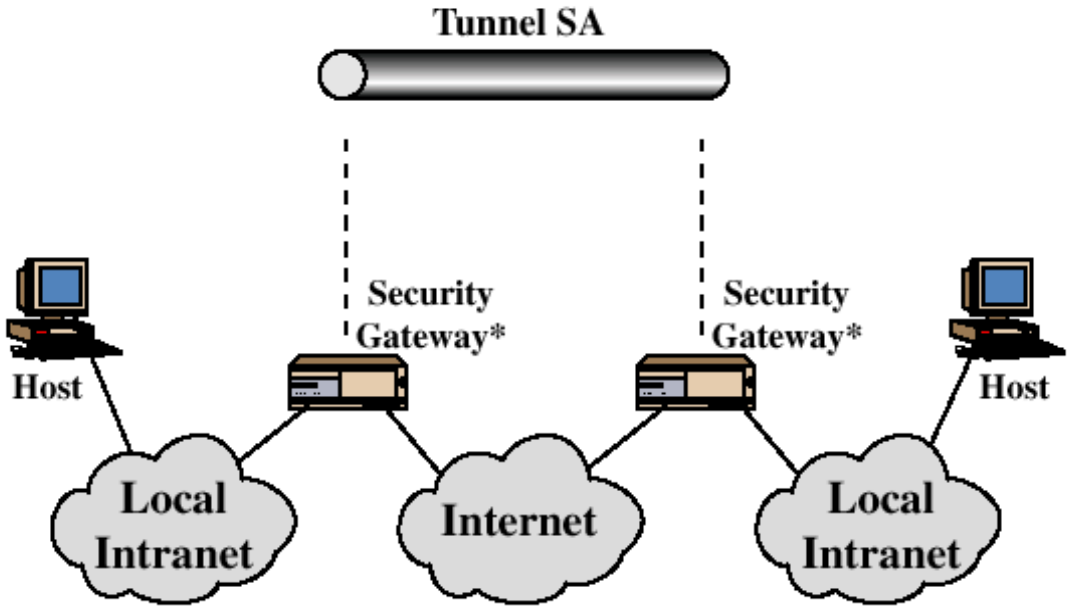
	Transport Mode SA	Tunnel Mode SA
AH	Autentica il payload del pacchetto IP ed alcuni campi dell'header IP	Autentica l'intero pacchetto IP interno ed alcuni campi del pacchetto IP esterno
ESP	Cifra il contenuto del pacchetto	Cifra l'intero pacchetto IP interno
ESP with authentication	Cifra il contenuto del pacchetto. Autentica il payload del pacchetto ma non l'header IP	Cifra ed autentica l'intero pacchetto IP interno.

Combinazione di Security Associations



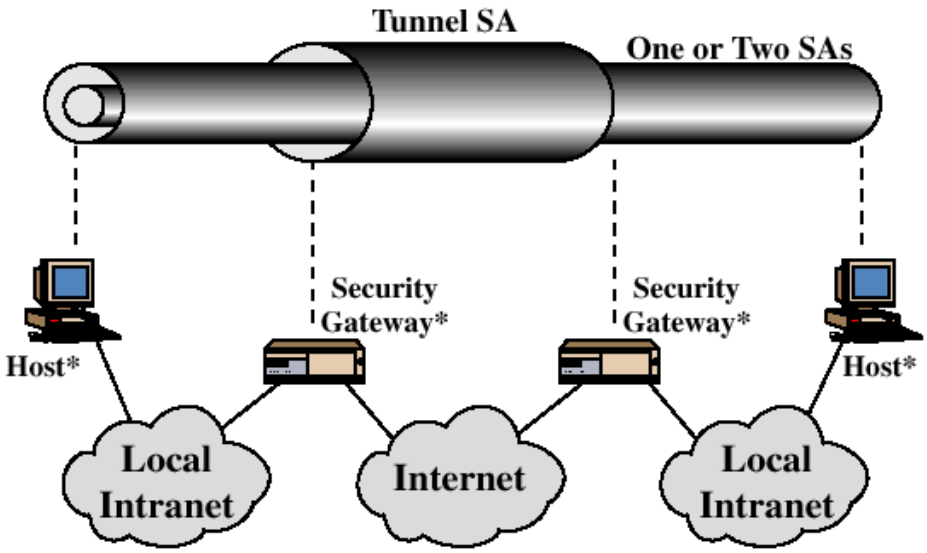
(a) Case 1

Combinazione di Security Associations



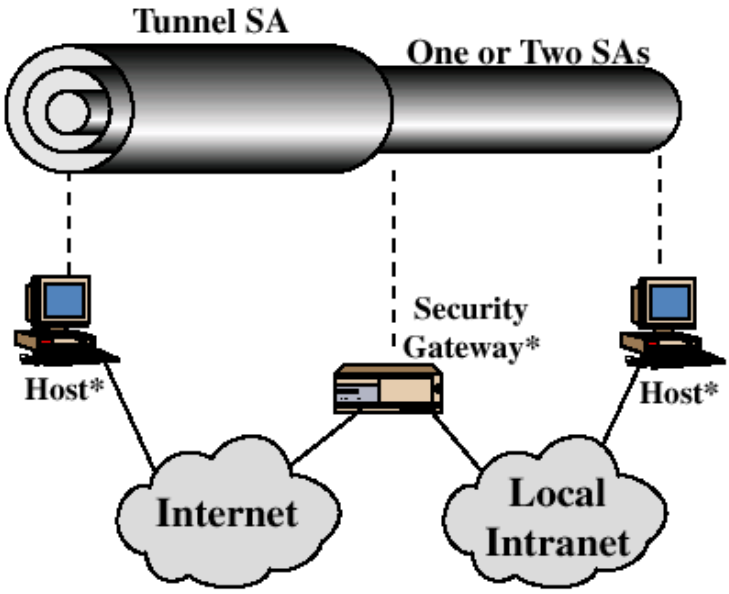
(b) Case 2

Combinazione di Security Associations



(c) Case 3

Combinazione di Security Associations



(d) Case 4

Considerazioni comparative

- SSL/TLS
  - è specifico di un dominio applicativo ☹
  - è semplice e realmente standard ☺
- IPSec
  - è generale e trasparente alle applicazioni ☺
  - è tipicamente implementato nello stack TCP/IP del sistema operativo, con variazioni che rendono difficile l'interoperabilità ☹
- Soluzioni "ibride"
  - utilizzo di varianti di SSL per il trasporto di pacchetti IP analogo al tunnel mode di IPSec
  - implementazione user space, indipendente dal S.O.
    - Es: OpenVPN