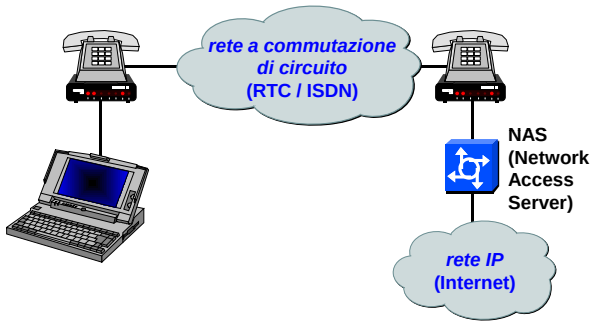
Sicurezza delle reti IP

Antonio Lioy
< lioy @ polito.it >

Politecnico di Torino
Dip. Automatica e Informatica

Accesso remoto via canali dial-up



Autenticazione di canali PPP

- **PPP è un protocollo ...**
  - ... per incapsulare pacchetti di rete (L3, es. IP) ...
  - ... e trasportarli su un collegamento punto-punto
    - reale (es. RTC, ISDN)
    - virtuale L2 (es. xDSL con PPPOE)
    - virtuale L3 (es. L2TP su UDP/IP)
- **tre fasi, svolte in sequenza:**
  - LCP (Link Control Protocol)
  - autenticazione (opzionale; PAP, CHAP o EAP)
  - L3 encapsulation (es. IPCP, IP Control Protocol)

Autenticazione degli accessi remoti

- **per accessi dial-up ma anche wireless o virtuali**
- **PAP**
  - Password Authentication Protocol
  - password in chiaro
- **CHAP**
  - Challenge Handshake Authentication Protocol
  - sfida simmetrica
- **EAP**
  - Extensible Authentication Protocol
  - aggancio a meccanismi esterni (sfide, OTP, TLS)

PAP

- **Password Authentication Protocol**
- **RFC-1334**
- **user-id e password inviate in chiaro**
- **autenticazione solo all'attivazione del collegamento**
- **molto pericoloso!**

CHAP

- **RFC-1994 "PPP Challenge Handshake Authentication Protocol (CHAP)"**
- **meccanismo a sfida simmetrico (basato sulla password)**
- **sfida iniziale obbligatoria, possibile ripetere la richiesta (con sfida diversa) durante la comunicazione a discrezione dell'NAS**
- **chi supporta sia CHAP sia PAP, *deve* prima offrire CHAP**

EAP

- **RFC-2284**
  "PPP Extensible Authentication Protocol (EAP)"
- **un framework flessibile di autenticazione a livello data-link**
- **tipi di autenticazione predefiniti:**
  - MD5-challenge (simile a CHAP)
  - OTP
  - generic token card
- **altri tipi possono essere aggiunti:**
  - RFC-2716 "PPP EAP TLS authentication protocol"
  - RFC-3579 "RADIUS support for EAP"

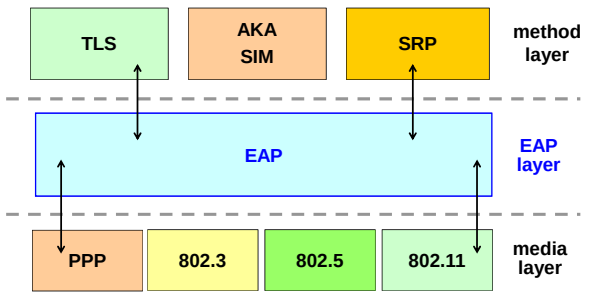
EAP - incapsulamento

- **per trasportare i dati di autenticazione usa un proprio protocollo di incapsulamento (perché il livello 3 non è ancora attivo …)**
- **caratteristiche dell'incapsulamento EAP:**
  - indipendente da IP
    - supporta qualsiasi link layer (es. PPP, 802, …)
  - ACK/NAK esplicito (no windowing)
    - assume no reordering
  - non supporta la frammentazione

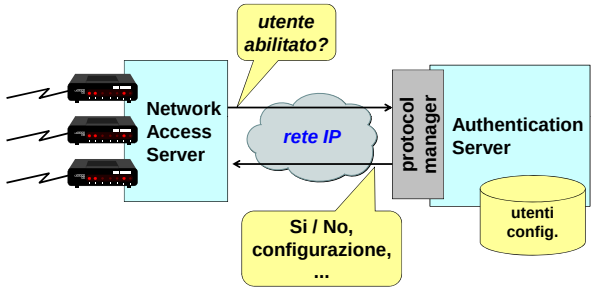
EAP

- **il link non è considerato fisicamente sicuro**
  - i metodi EAP devono provvedere ai necessari servizi di sicurezza
- **metodi EAP:**
  - EAP-TLS
  - EAP-MD5
  - tunnelled TLS (permette di operare qualsiasi metodo EAP protetto da TLS)
  - EAP-SRP (Secure Remote Password)
  - GSS_API (incluso Kerberos)
  - AKA-SIM

EAP - architettura



Autenticazione per collegamenti dial-up



Le tre A

- **i produttori di NAS dicono che la sicurezza richiede tre funzioni:**
  - Autenticazione
  - Autorizzazione
  - Accounting
- **l'AS ricopre proprio queste tre funzioni dialogando con uno o più NAS tramite uno o più protocolli**

Protocolli di autenticazione via rete

- **RADIUS**
  - il più diffuso
  - funzionalità di proxy verso altri AS
- **DIAMETER**
  - evoluzione di RADIUS
  - enfasi su roaming tra ISP diversi
  - elevata attenzione alla sicurezza
- **TACACS+ (TACACS, XTACACS)**
  - in origine tecnicamente migliore di RADIUS ma meno diffuso perché proprietario

RADIUS

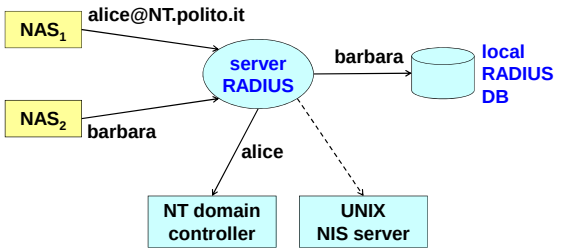
- **Remote Authentication Dial-In User Service**
- **Livingston Technologies**
- **UDP, porta 1812 (errore: UDP/1645)**
- **supporta autenticazione, autorizzazione e accounting per l'accesso alla rete:**
  - porte fisiche (analogiche, ISDN, IEEE 802)
  - porte virtuali (tunnel, accessi wireless)
- **amministrazione e accounting centralizzato**
- **schema client-server tra NAS e AS**
  - timeout e ritrasmissione
  - server secondari

RADIUS - RFC

- **RFC-2865 (protocollo)**
- **RFC-2866 (accounting)**
- **RFC-2867/2868 (accounting e attributi per tunnel)**
- **RFC-2869 (estensioni)**
- **RFC-3579 (RADIUS support for EAP)**
- **RFC-3580 (guidelines for 802.1X with RADIUS)**

RADIUS proxy

- **il server può fungere da proxy verso altri server di autenticazione**



RADIUS: protezione dei dati

- **integrità ed autenticazione dei pacchetti tramite keyed-MD5:**
  - key = shared-secret
  - client senza key sono ignorati
- **password trasmessa "crittografata" con MD5 (dopo averne fatto il padding a 128 bit con NUL):**

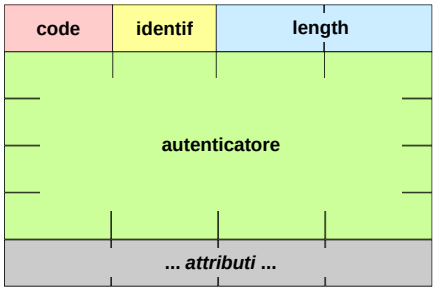
  (password || pad) ⊕ md5(key || autenticatore)

RADIUS

- **autenticazione dell'utente tramite PAP, CHAP, token-card e EAP**
  - CISCO fornisce server per CryptoCard
  - altri offrono SecurID
- **formato degli attributi TLV (facilmente estendibile senza modificare l'installato):**

  attribute Type – Length – Value

RADIUS - formato



RADIUS - pacchetti

- **ACCESS-REQUEST**
- **ACCESS-REJECT**
- **ACCESS-CHALLENGE**
- **ACCESS-ACCEPT (*parametri*):**
  - SLIP/PPP: IPaddr, netmask, MTU, ...
  - terminal: host, port

RADIUS - autenticatore

- **duplice scopo:**
  - autenticare la risposta del server ed evitare replay
  - mascherare la password
- **in Access-Request:**
  - si chiama Request Authenticator
  - sono 16 byte random generati dal NAS
- **in Access-Accept / Reject / Challenge**
  - si chiama Response Authenticator
  - si calcola con un keyed-digest:

md5 (code || ID || length || RequestAuth || attributes || secret)

RADIUS - alcuni attributi

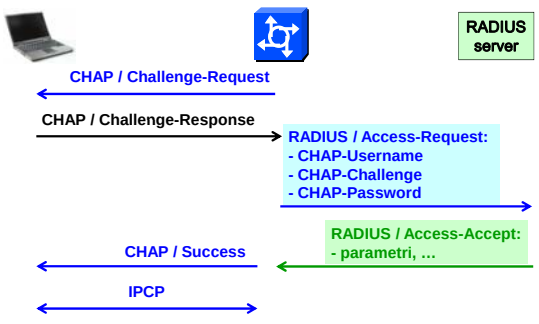
type	length	value

- **type = 1 (User-Name)**
  - value = text, network access identifier (NAI), DN
- **type = 2 (User-Password)**
  - value = password ⊕ md5 (key || RequestAuthent.)
- **type = 3 (Chap-Password)**
  - value = user CHAP response (128 bit)
- **type = 60 (CHAP-Challenge)**
  - value = sfida fatta dal NAS all'utente

NAI (Network Access Identifier)

- **RFC-2486**
- **NAI = username [@ realm]**
- **tutti dispositivi devono supportare almeno NAI lunghi 72 byte**
- **la sintassi esatta di username e realm è descritta nell'RFC (si noti che include solo i caratteri ASCII < 128 ma li include tutti)**

Esempio - CHAP + RADIUS



DIAMETER

- **evoluzione di RADIUS**
- **particolare enfasi sul roaming tra ISP diversi**
- **RFC-3588 "Diameter base protocol"**
- **RFC-3589 "Commands for the 3GPP"**
- **RFC-3539 "AAA transport profile"**
- **RFC-4004 "Diameter mobile IPv4 application"**
- **RFC-4005 "Diameter network access server application"**
- **RFC-4006 "Diameter credit-control application"**
- **RFC-4072 "Diameter EAP application"**

Sicurezza di DIAMETER

- **protezione obbligatoria con IPsec o TLS:**
  - client Diameter DEVE supportare IPsec e PUO' supportare TLS
  - server Diameter DEVE supportare IPsec e TLS
- **configurazioni obbligatorie:**
  - (IPsec) ESP con algo non nulli sia per autenticazione sia per riservatezza
  - (TLS) mutua autenticazione (client DEVE avere un certificato a chiave pubblica)
  - (TLS) DEVE supportare RSA+RC4_128/3DES+ MD5/SHA1 e PUO' usare RSA+AES_128+SHA1

IEEE 802.1x

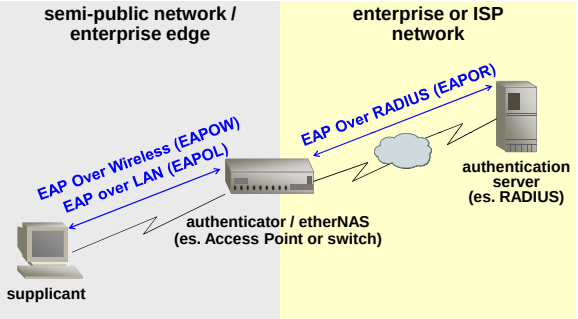
- **Port-Based Network Access Control:**
  - architettura di autenticazione per il livello 2 (MAC - Media Access Control)
  - utile sugli switch di reti wired per bloccare l'accesso alla rete
  - indispensabile nel caso di reti wireless
- **prime implementazioni:**
  - Windows-XP e access-point wireless Cisco

http://standards.ieee.org/getieee802/download/802.1X-2001.pdf

IEEE 802.1x

- **un framework per autenticazione e key-management**
  - 802.1x può derivare chiavi di sessione da usare per autenticazione, integrità e segretezza dei pacchetti
  - sfrutta algoritmi standard per la derivazione delle chiavi (es. TLS, SRP, …)
  - servizi di sicurezza opzionali (autenticazione o autenticazione+cifratura)

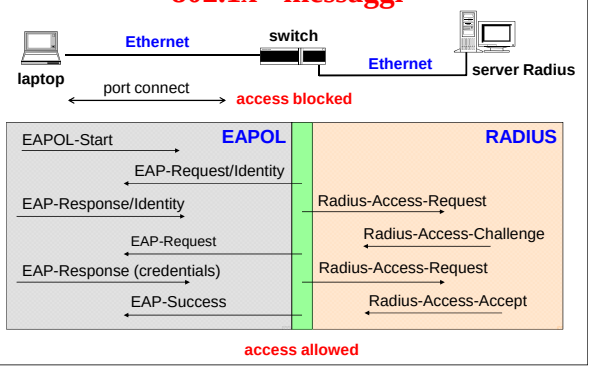
802.1x - architettura



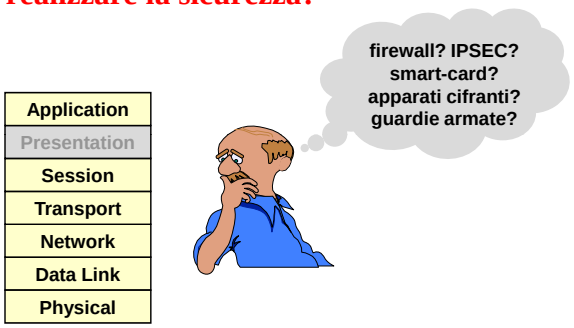
802.1x - vantaggi

- **sfrutta il livello applicativo per l'effettiva implementazione dei meccanismi di sicurezza**
  - conversazione diretta tra supplicant e AS
    - NIC e NAS agiscono come "pass-through device"
  - nessun cambiamento su NIC e NAS per implementare nuovi meccanismi di autenticazione
  - perfetta integrazione con AAA

802.1x - messaggi



A quale livello di rete è meglio realizzare la sicurezza?

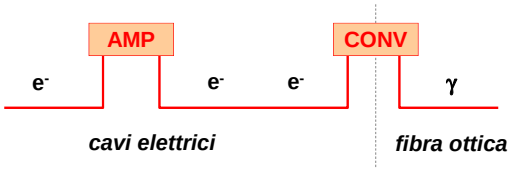


Livello ottimale?

- **più si sale nello stack e più le funzioni saranno specifiche (es. possibile identificare l'utente, i comandi, i dati) ed indipendenti dalle reti sottostanti ma più saranno possibili attacchi di tipo DoS**
- **più si resta in basso nello stack e più sarà possibile "espellere" in fretta gli intrusi ma i dati su cui basare questa decisione saranno più scarsi (es. solo indirizzo MAC o IP, no utenti, no comandi)**

Sicurezza a livello fisico

- **protezione fisica:**
  - del supporto trasmissivo
  - degli amplificatori / ripetitori / convertitori
- **tipicamente solo in reti chiuse (es. militari, governo, alta finanza)**

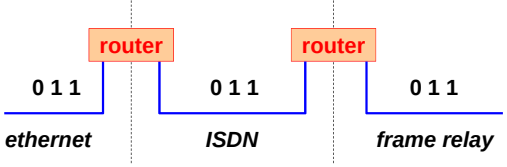


Misure di sicurezza a livello fisico

- **usare reti switched (ossia 10baseT o 100baseT) per eliminare lo sniffing:**
  - evitare gli hub
  - evitare derivazioni multiple sulla stessa porta dello switch
- **proteggere gli armadi / locali che contengono le apparecchiature di rete**
- **proteggere i cavedi / pozzetti**

Sicurezza a livello data-link

- **apparati cifranti per proteggere i dati a livello 2 (MAC)**
- **solo per segmenti con tecnologia omogenea**
  - LAN
  - spezzoni di WAN



Sicurezza a livello data-link

- **sebbene esistano schede cifranti da installare sui client, normalmente non si protegge il livello 2 sulle stazioni ma solo su tratte geografiche punto-punto**
- **si comincia a pensare la gestione della LAN associata a quella della sicurezza:**
  - VLAN
  - switch con porte protette (es. 802.1x)
  - allarmi automatici al comparire di un nuovo MAC
  - assegnazione statica degli indirizzi IP
  - no a DHCP completamente dinamico

Sicurezza del DHCP

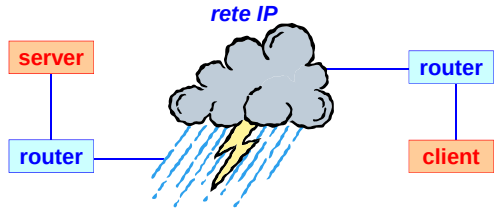
- **protocollo non autenticato**
- **facilissimo attivare shadow server**
- **attacchi possibili da parte del falso server:**
  - denial-of-service
    - fornisco configurazione di rete sbagliata
  - MITM logico
    - fornisco configurazione con subnet da 2 bit + default gateway uguale alla macchina che vuole essere MITM
    - facendo NAT si intercettano anche le risposte

Protezione del DHCP

- **alcuni switch (es. Cisco) offrono:**
  - DHCPsnooping = solo risposte da “trusted port”
  - IP guard = solo IP ottenuti da DHCP (ma ci sono limitazioni sul numero di indirizzi che si riesce a trattare)
- **RFC-3118 “Authentication for DHCP messages”**
  - usa HMAC-MD5 per autenticare i messaggi
  - scarsamente adottato

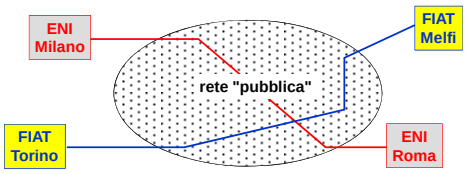
Sicurezza a livello network

- protezione end-to-end per reti omogenee a livello logico (es. IP)
- possibile anche creare VPN (Virtual Private Network) per proteggere solo una parte del path



Che cos'è una VPN?

- una tecnica (hardware e/o software) per realizzare una rete privata ...
- ... utilizzando canali e apparati di trasmissione condivisi o comunque non fidati



Dove si applica una VPN?

- quando si attraversa una rete pubblica e/o non fidata ...
- ... per comunicazioni intra-aziendali tra sedi remote (Intranet)
- ... per comunicazioni inter-aziendali chiuse tra aziende che si sono previamente accordate (Extranet)

Dove NON si applica una VPN?

- **quando si attraversa una rete pubblica e/o non fidata ...**
- **... per comunicazioni inter-aziendali senza accordi precedenti**
- **... per comunicazioni con clienti non noti a priori (commercio elettronico di tipo business-to-consumer)**

Tecniche di realizzazione di una VPN

- **mediante reti nascoste**
- **mediante routing protetto (tunnel IP)**
- **mediante protezione crittografica dei pacchetti rete (tunnel IP sicuro)**

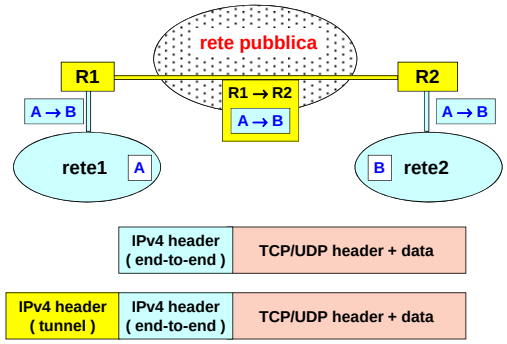
1. VPN mediante rete nascosta

- **le reti da collegare utilizzano un indirizzamento non standard per non essere raggiungibili da altre reti (es. reti nascoste IANA secondo RFC-1918)**
- **è una protezione facilmente superabile se qualcuno:**
  - scopre gli indirizzi usati
  - può leggere i pacchetti in transito
  - ha accesso all'infrastruttura di comunicazione

2. VPN mediante tunnel

- **i router provvedono ad incapsulare i pacchetti di rete all'interno di altri pacchetti**
  - IP in IP
  - IP over MPLS
  - altro
- **i router controllano l'accesso alle reti mediante ACL (Access Control List)**
- **protezione superabile da chi gestisce i router o da chi può comunque leggere i pacchetti in transito**

VPN mediante tunnel IP



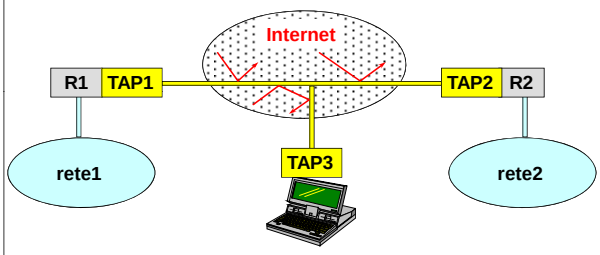
Tunnel IP: frammentazione

- **se il pacchetto da trasmettere ha la massima dimensione consentita, allora deve essere frammentato**
- **massimo degrado = 50%**
- **soffrono maggiormente gli applicativi con pacchetti grandi (tipicamente non interattivi)**

3. VPN mediante tunnel IP sicuro

- **prima di essere incapsulati i pacchetti di rete vengono protetti con:**
  - digest (per integrità ed autenticazione)
  - cifratura (per riservatezza)
  - numerazione (per evitare replay)
- **se gli algoritmi crittografici sono forti, allora l'unico attacco possibile è impedire le comunicazioni**
- **anche detta S-VPN (Secure VPN)**

VPN mediante tunnel IP sicuro



IPsec

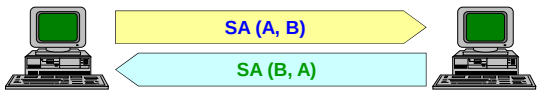
- **architettura IETF per fare sicurezza al livello 3 sia in IPv4 sia in IPv6:**
  - creare VPN su reti non fidate
  - fare sicurezza end-to-end
- **definisce due formati particolari:**
  - AH (Authentication Header)
    per integrità, autenticazione, no replay
  - ESP (Encapsulating Security Payload)
    per riservatezza (+AH)
- **usa un protocollo per scambio chiavi:**
  - IKE (Internet Key Exchange)

Servizi di sicurezza IPsec

- **autenticazione dei pacchetti IP:**
  - integrità dei dati
  - identificazione del mittente
  - protezione (parziale) da attacchi di tipo “replay”
- **riservatezza dei pacchetti IP:**
  - cifratura dei dati

IPsec Security Association (SA)

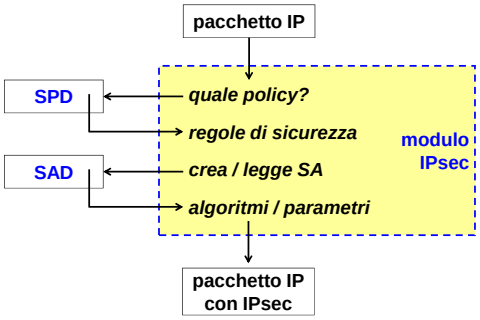
- **connessione logica unidirezionale protetta tra due sistemi IPsec**
- **ad ogni SA sono associabili caratteristiche di sicurezza diverse**
- **occorrono due SA per avere protezione completa di un canale bidirezionale**



Database locali IPsec

- **SPD (Security Policy Database)**
  - contiene le security policy da applicare ai diversi tipi di comunicazione
  - configurato a priori (es. manualmente) oppure agganciato ad un sistema automatico (es. ISPS, Internet Security Policy System)
- **SAD (SA Database)**
  - elenco delle SA attive e delle loro caratteristiche (algoritmi, chiavi, parametri)

Come funziona IPsec (spedizione)



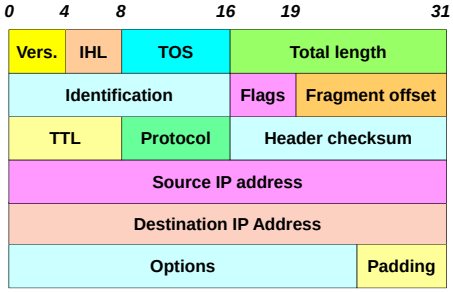
IPsec - seconda versione

- novembre 1998
- RFC-2411 = IPsec document roadmap
- RFC-2401 = architecture
- RFC-2402 = AH
- RFC-2403 = HMAC-MD5-96 in ESP e AH
- RFC-2404 = HMAC-SHA-1-96 in ESP e AH
- RFC-2405 = ESP DES-CBC con IV esplicito
- RFC-2406 = ESP
- RFC-2410 = cifratura nulla in IPsec
- RFC-2451 = algoritmi per ESP CBC

IPsec - scambio chiavi

- RFC-2407 = interpretazione IPsec di ISAKMP
- RFC-2408 = ISAKMP
- RFC-2409 = IKE
- RFC-2412 = OAKLEY

Header IPv4

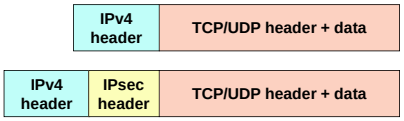


Header IPv4

- **indirizzi IP (32 bit) del mittente e del destinatario**
- **IHL (Internet Header Length) in 32-bit word**
- **TOS (Type Of Service): mai usato (!)**
- **Length: n. di byte del pacchetto IP**
- **Identification: ID del pacchetto (per i frammenti)**
- **Flags: may/don't fragment, last/more fragments**
- **TTL (Time To Live): numero massimo di hop**
- **protocol: protocollo usato dal payload**

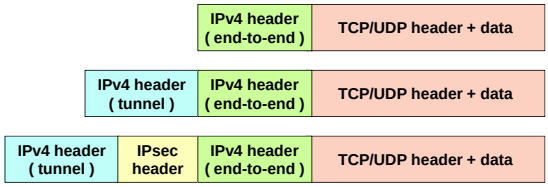
IPsec in transport mode

- **usato per fare sicurezza end-to-end, ossia usato dagli host, non dai gateway (eccezione: traffico destinato al gateway, es. SNMP, ICMP)**
- **vantaggio: computazionalmente leggero**
- **svantaggio: non protegge i campi variabili**



IPsec in tunnel mode

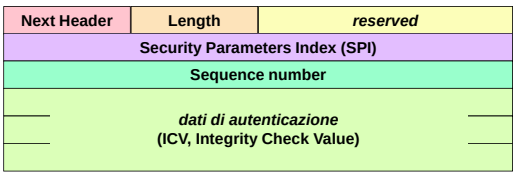
- **usato per fare VPN, solitamente dai gateway**
- **vantaggio: protegge anche i campi variabili**
- **svantaggio: computazionalmente pesante**

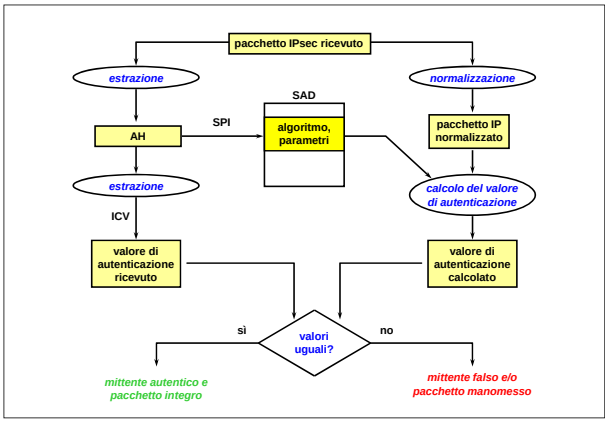


AH

- **Authentication Header**
- **meccanismo (prima versione, RFC-1826):**
  - integrità dei dati ed autenticazione del mittente
  - obbligatorio keyed-MD5 (RFC-1828)
  - opzionale keyed-SHA-1 (RFC-1852)
- **meccanismo (seconda versione, RFC-2402):**
  - integrità dei dati, autenticazione del mittente e protezione da replay attack
  - HMAC-MD5-96
  - HMAC-SHA-1-96

AH - formato (RFC-2402)





Normalizzazione per AH

- **azzerare il campo TTL / Hop Limit**
- **se il pacchetto contiene un Routing Header, allora:**
  - fissare il campo destinazione all'indirizzo del destinatario finale
  - fissare il contenuto del routing header al valore che avrà a destinazione
  - fissare il campo Address Index al valore che avrà a destinazione
- **azzerare tutte le opzioni che hanno il bit C (*change en route*) attivo**

Keyed-MD5 in AH

- **dato M normalizzarlo generando M'**
- **allineare a 128 bit M' (aggiungendo byte a zero) generando così M'p**
- **allineare a 128 bit la chiave K (aggiungendo byte a zero) generando così Kp**
- **calcolare il valore di autenticazione:**

ICV = md5 (Kp || M'p || Kp)

HMAC-MD5-96

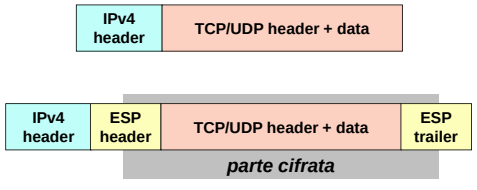
- **dato M normalizzarlo generando M'**
- **allineare a 128 bit M' (aggiungendo byte a zero) generando così M'p**
- **allineare a 128 bit la chiave K (aggiungendo byte a zero) generando così Kp**
- **dati ip = 00110110 e op = 01011010 (ripetuti a formare 128 bit) calcolare la base di autenticazione:**
  B = md5 ((Kp ⊕ op) || md5 ((Kp ⊕ ip) || M'p))
- **ICV = 96 leftmost bit di B**

ESP

- **Encapsulating Security Payload**
- **prima versione (RFC-1827), solo riservatezza**
- **meccanismo base: DES-CBC (RFC-1829)**
- **possibili anche altri meccanismi**
- **seconda versione (RFC-2406):**
  - anche autenticazione (ma esclude l'header IP, quindi non dà la stessa copertura di AH)
  - riduce la dimensione del pacchetto e risparmia una SA

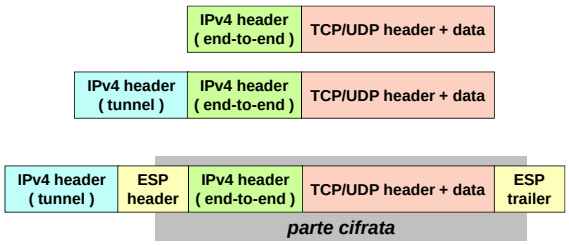
ESP in transport mode

- **usato dagli host, non dai gateway (eccezione: traffico destinato al gateway, es. SNMP, ICMP)**
- **svantaggio: non nasconde l'header**

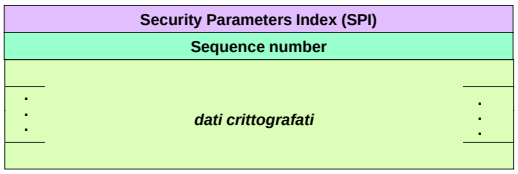


ESP in tunnel mode

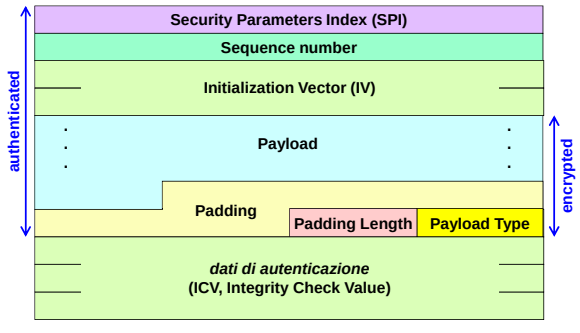
- usato solitamente dai gateway
- vantaggio: nasconde anche gli header



ESP - formato (RFC-2406)



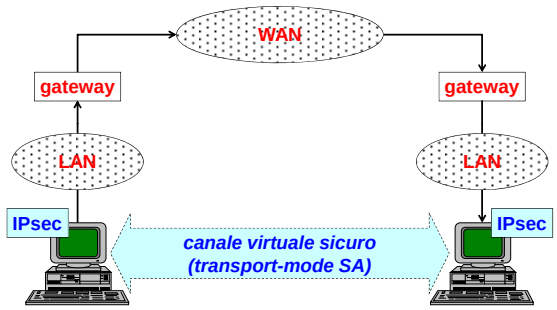
ESP-DES-CBC - formato (RFC-2406)



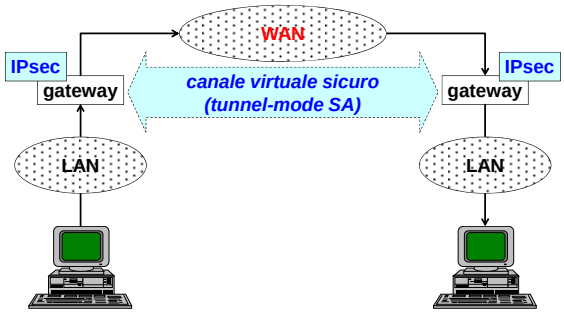
Dettagli implementativi

- **sequence number:**
  - non deve essere strettamente sequenziale (protezione solo da replay)
  - finestra minima di 32 pacchetti (consigliati 64)
- **algoritmi NULL:**
  - per autenticazione
  - per crittografia (RFC-2410)
  - offrono trade-off protezione - prestazioni

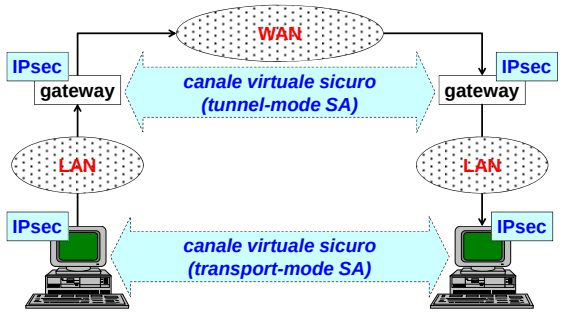
End-to-end security



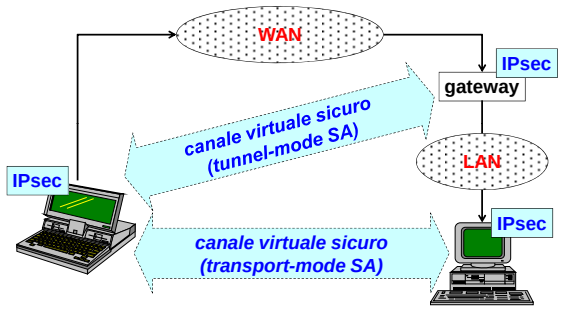
Basic VPN



End-to-end security with basic VPN



Secure remote access



IPsec key management

- **componente fondamentale di IPsec**
- **fornisce ai sistemi IPsec comunicanti le chiavi simmetriche necessarie per l'autenticazione e/o la cifratura dei pacchetti**
- **distribuzione delle chiavi?**
  - OOB (es. manuale)
  - automatica

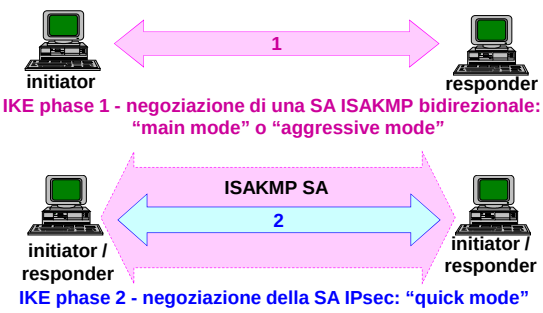
ISAKMP

- **Internet Security Association and Key Management Protocol**
- **RFC-2408**
- **definisce le procedure necessarie per negoziare, stabilire, modificare e cancellare le SA**
- **non indica il metodo da usare per lo scambio delle chiavi**
  - OAKLEY (RFC-2412): protocollo che realizza lo scambio autenticato delle chiavi simmetriche tra sistemi IPsec

IKE

- **Internet Key Exchange (RFC-2409)**
- **= ISAKMP + OAKLEY**
- **creazione di una SA per proteggere lo scambio ISAKMP**
- **con questa SA protegge la negoziazione della SA richiesta da IPsec**
- **la stessa SA ISAKMP può essere riusata più volte per negoziare altre SA IPsec**

IKE - funzionamento



IKE: "modi" di funzionamento

- **Main Mode:**
  - 6 messaggi
  - protegge l'identità delle parti
- **Aggressive Mode:**
  - 3 messaggi (ma non protegge l'identità delle parti)
- **Quick Mode:**
  - 3 messaggi
  - negoziazione solo della SA IPsec
- **New Group Mode:**
  - 2 messaggi

IKE: metodi di autenticazione

- **Digital Signature**
  - non-repudiation della negoziazione IKE
- **Public Key Encryption**
  - protezione dell'identità nell'aggressive mode
- **Revised Public Key Encryption**
  - meno costoso, solo 2 operazioni a chiave pubblica
- **Pre-Shared Key**
  - l'ID della controparte può essere solo il suo indirizzo IP (problema per gli utenti mobili)

IPsec nei sistemi operativi

- **IPsec è implementato in tutti gli Unix più recenti**
  - implementazione SUN tramite SKIP in Solaris<8
- **Linux:**
  - IPsec nativo nel kernel 2.6 (derivato da Kame)
  - FreeS/WAN (www.freeswan.org) e successori:
    - openswan (www.openswan.org)
    - strongswan (www.strongswan.org)
- **Microsoft lo ha introdotto nei suoi prodotti a partire da Windows-2000**

IPsec nei router

- **tutti i principali fornitori di router (Cisco, 3COM, Nortel, ...) hanno IPsec sui router**
- **tipicamente è usato solo per creare canali protetti tra i router ma non con gli end-node**
- **Cisco ha anche l'autenticazione a chiave pubblica con certificati X.509**

IPsec nei firewall

- **alcuni produttori di firewall (es. IBM, Checkpoint) offrono IPsec all'interno dei loro prodotti di tunnel sicuro**
- **forniscono gratuitamente il client per Windows che però può creare un canale IPsec solo col proprio firewall**

VPN concentrator

- **apparecchiature special-purpose che fungono da terminatori di tunnel IPsec:**
  - per accesso remoto di singoli client
  - per creare VPN site-to-site
- **prestazioni molto elevate in relazione ai costi (bassi)**

Requisiti di sistema per IPsec

- **su router:**
  - CPU potente o acceleratore crittografico
  - non gestito in outsourcing
- **su firewall:**
  - CPU potente
- **su VPN concentrator:**
  - massima indipendenza dalle altre misure di sicurezza

Influenza di IPsec sulle prestazioni

- **diminuizione del throughput di rete:**
  - maggiore dimensione dei pacchetti
    - transport mode AH: +24 byte
    - transport mode ESP-DES-CBC: >= 32 byte
  - maggior numero di pacchetti (per attivare la SA)
- **di solito diminuizione contenuta**
- **eccezione: link punto-punto in cui si usava compressione a livello 2 che diventa inutile o dannosa se associata a pacchetti ESP**
- **possibile compensazione tramite IPComp (RFC-3173) o compressione applicativa**

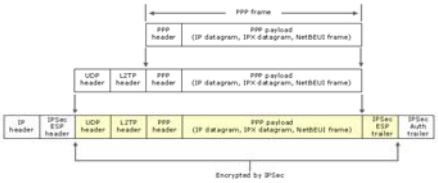
IPsec tunnel mode/L2TP

- **Windows rende sicuro l'accesso remoto dal client al gateway usando L2TP sicurizzato da IPsec**
- **MS dichiara di aver fatto questa scelta perché l'IPsec tunnel mode:**
  - non permette l'autenticazione dell'utente
  - non supporta il multiprotocollo
  - non supporta il multicast
- **la scelta di L2TP causa:**
  - un calo di prestazioni
  - problemi di interoperabilità tra sistemi diversi

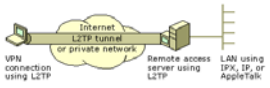
Che cos'è L2TP?

- **Layer-2 Tunnel Protocol (RFC-2661)**
- **incapsula i pacchetti PPP in IP**
- **vantaggio:**
  - possibilità di usufruire del supporto PPP per il multi-protocollo (es. anche per IPX, Netbeui e Appletalk)
  - autenticazione dell'utente (PAP / CHAP)
- **svantaggio: overhead**
- **con L2TP ciascun end-point mantiene una PPP state machine come se i due fossero connessi da una linea seriale**

IPsec over L2tp



I pacchetti PPP sono incapsulati 2 volte e poi trasmessi come pacchetti IPsec



I pacchetti viaggiano normalmente come pacchetti IP anche se sono IPX o AppleTalk

Applicabilità di IPsec

- **solo pacchetti unicast (no broadcast, no multicast, no anycast)**
- **tra parti che hanno attivato una SA:**
  - tramite chiavi condivise
  - tramite certificati X.509
- **… quindi in gruppi "chiusi"**

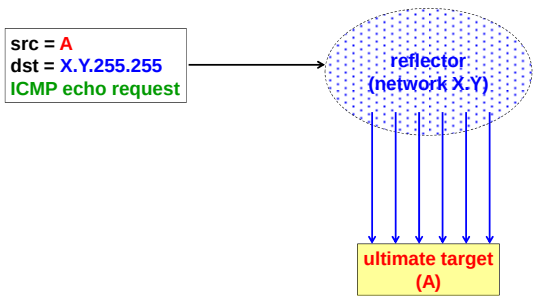
Sicurezza di IP

- **indirizzi non autenticati**
- **pacchetti non protetti:**
  - integrità
  - autenticazione
  - riservatezza
  - replay
- **sono quindi attaccabili tutti i protocolli che usano IP come trasporto, soprattutto quelli di "servizio" ossia non di livello applicativo (ICMP, IGMP, DNS, RIP, …)**

Sicurezza di ICMP

- **Internet Control and Management Protocol**
- **vitale per la gestione della rete**
- **possibili moltissimi attacchi perché completamente privo di autenticazione**
- **funzioni ICMP:**
  - echo request / reply
  - destination unreachable (network / host / protocol / port unreachable)
  - source quence
  - redirect
  - time exceeded for a datagram

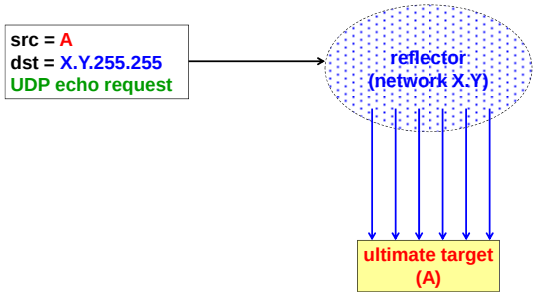
Smurfing attack



Contromisure anti-smurfing

- **per attacchi dall'esterno: rifiutare il broadcast IP**
  interface serial0
  no ip directed-broadcast
- **per attacchi dall'interno: identificare il responsabile tramite strumenti di network management**

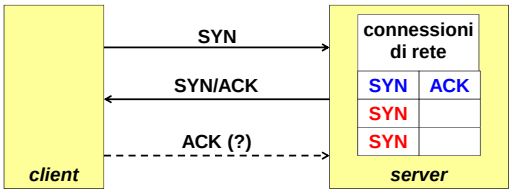
Fraggle attack



ARP poisoning

- **ARP = Address resolution Protocol (RFC-826)**
  - usato per scoprire l'indirizzo L2 di un nodo di cui è noto l'indirizzo L3
  - risultato memorizzato in ARP table
- **ARP poisoning = inserire dati falsi in ARP table:**
  - nodi accettano ARP reply senza ARP request
  - nodi sovrascrivono entry ARP statiche con quelle dinamiche (ottenute da ARP reply)
  - il campo "ar$sha" (sender hw address) di ARP può differire dal campo src nel pacchetto 802.3
  - usato da strumenti di attacco (es. Ettercap)

TCP SYN flooding



- **richieste multiple con IP spoofing**
- **si satura la tabella delle connessioni sino a quando non vanno in timeout (valore tipico: 75")**

Difesa da SYN flooding

- **abbassare il timeout**
  - rischio di eliminare client validi ma lenti
- **aumentare le dimensioni della tabella**
  - aggirabile con invio di più richieste
- **usare un router come "SYN interceptor":**
  - sostituisce il server nella prima fase
  - se l'handshake ha successo, trasferisce il canale al server
  - timeout "aggressivi" (rischio!)
- **usare un router come "SYN monitor":**
  - uccide i collegamenti pendenti (RST)

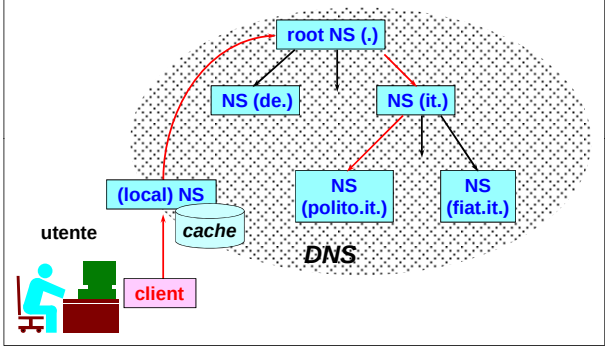
SYN cookie

- **idea di D.J.Bernstein (http://cr.yp.to)**
- **unico sistema veramente efficace per evitare completamente il SYN flooding**
- **usa il sequence number del pacchetto SYN-ACK per trasmettere un cookie al client e riconoscere così i client che hanno già inviato il SYN senza memorizzare niente sul server**
- **disponibile su Linux e Solaris**

Sicurezza del DNS

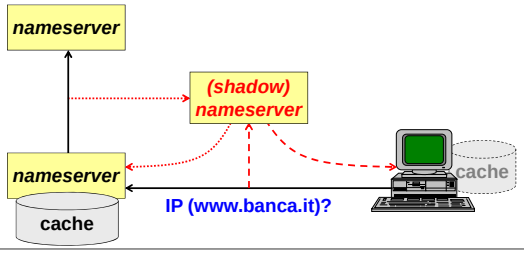
- **DNS (Domain Name System)**
- **traduzione:**
  - da nomi ad indirizzi IP
  - da indirizzi IP a nomi
- **servizio indispensabile**
- **UDP/53 per le query**
- **TCP/53 per zone transfer**
- **nessun tipo di sicurezza**
- **in corso di sviluppo DNS-SEC**

Architettura del DNS



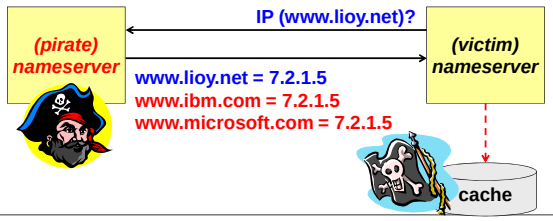
DNS shadow server

- **sniffing per intercettare le query**
- **spoofing per generare risposte false (DoS o ridirezione del traffico su siti falsi)**



DNS cache poisoning

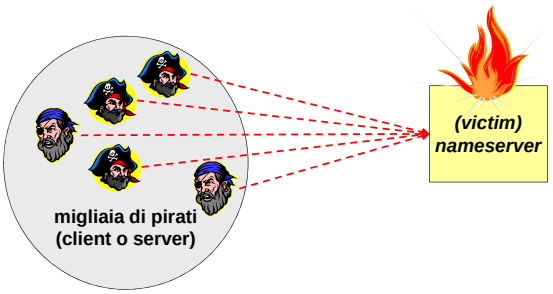
- attirare la vittima a fare una query sul proprio NS
- fornire risposta anche a query non effettuate per forzare / sovrascrivere la cache della vittima



DNS cache poisoning (2a versione)

- fare una query e fornire subito la risposta (falsa) per inserirla nella cache della vittima

(DNS) flash crowd



BIND

- **per la sicurezza del DNS si raccomanda l'uso (e l'aggiornamento periodico!) di BIND**
- **Berkeley Internet Name Domain server**
- **gratis**
- **per Unix e Win-32**
- **http://www.isc.org**
- **iscriversi alla mailing list di sicurezza di BIND perché - essendo un programma enorme - ha talvolta bachi di sicurezza**

DJBDNS

- **server DNS di D.J.Bernstein, progettato per essere sicuro:**
  - semplice e modulare
  - sviluppo con tecniche di programmazione sicura
- **http://cr.yp.to/djbdns.html**
- **tre servizi distinti:**
  - tiny DNS (nameserver autoritativo per un dominio)
  - dnscache (gestore della cache)
  - walldns (un reverse DNS wall)

Caratteristiche di sicurezza di DJBDNS

- **processi inoffensivi:**
  - l'UID non è root
  - girano in chroot
- **dnscache scarta:**
  - le richieste non provenienti da IP fidati
  - le risposte da IP diversi da quello a cui ha fatto richiesta
- **dnscache è immune al cache 'poisoning'**
- **tinydns e walldns non fanno caching di informazioni**

walldns

- **maschera i veri nomi di una rete**
- **serve nel caso di un server che chiede il record PTR prima di fornire i suoi servizi**
- **i nomi veri non vengono mai rivelati, walldns fornisce solo nomi fasulli (per compiacere il richiedente)**
- **non soddisfa i "server paranoici", cioè quelli che fanno un doppio lookup incrociato:**
  - N = dns_query (client_IP, PTR_record)
  - A = dns_query (N, A_record)
  - A è uguale a client_IP?

DNSsec

- **firma digitale dei record DNS**
  - chi è "authoritative" per un certo dominio?
  - quale PKI? (certificati, trusted root CA)
- **gestione più complessa dell'infrastruttura DNS**
  - firme gerarchiche e delegate
  - firme distribuite
- **come trattare nomi inesistenti?**
  - firmare anche l'ASSENZA di un record
  - richiede ordinamento dei record

Alcuni problemi di DNSsec

- **nessuna firma delle query DNS**
- **nessuna sicurezza nel dialogo tra DNS client e DNS (local) server**
  - usare IPsec o TSIG
- **crittografia sui server DNS**
  - sovraccarico computazionale
  - sovraccarico gestionale (on-line secure crypto host)
- **maggior dimensione dei record**
- **scarsa sperimentazione**
  - configurazione? prestazioni?

Sicurezza del routing

- **bassa sicurezza nell'accesso sistemistico ai router per la gestione (telnet, SNMP)**
- **bassa sicurezza nello scambio delle tabelle di routing**
  - autenticazione basata sull'indirizzo IP
  - possibile attivare protezione con keyed-digest
    - richiede chiave condivisa!
    - richiede key-management!
- **variazioni dinamiche del routing anche sugli end-node tramite ICMP**

Protezione fisica dei router

- **limitare l'accesso fisico ai router solo alle persone autorizzate**
- **porta di console:**
  - aggancio diretto di un terminale o PC
  - permette accesso diretto coi massimi privilegi
  - proteggerla tramite una password (per default non c'è!)

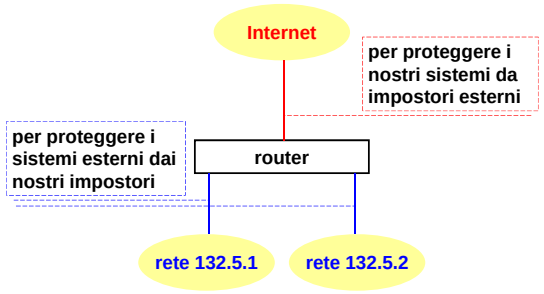
Protezione logica dei router

- **attivare le ACL più comuni**
- **proteggere l'accesso al file di configurazione (ovunque conservato) perché esso contiene:**
  - le password (spesso in chiaro!)
  - le ACL basate su indirizzi IP

Protezione da IP spoofing

- per proteggerci dagli impostori esterni
- ma anche per proteggere l'esterno dai nostri impostori interni (=misura di net-etiquette)
- RFC-2827 "Network ingress filtering: defeating Denial of Service attacks which employ IP source address spoofing"
- RFC-3704 "Ingress filtering for multihomed networks"
- RFC-3013 "Recommended Internet Service Provider security services and procedures"

Filtri per protezione da IP spoofing



Esempio protezione da IP spoofing

```
access-list 101 deny ip
  132.5.0.0 0.0.255.255 0.0.0.0 255.255.255.255
interface serial 0
ip access-group 101 in

access-list 102 permit ip
  132.5.1.0 0.0.0.255 0.0.0.0 255.255.255.255
interface ethernet 0
ip access-group 102 in

access-list 103 permit ip
  132.5.2.0 0.0.0.255 0.0.0.0 255.255.255.255
interface ethernet 1
ip access-group 103 in
```

Sicurezza di SNMP

- **pacchetti UDP/161**
- **SNMP (v1, v2, v3):**
  - protezione di default tramite segreto condiviso trasmesso in chiaro (stringa di "community")
  - nessuna autenticazione dei client
  - nessuna protezione dei messaggi
- **SNMPv3 presta più attenzione alla sicurezza ma è raramente implementato e spesso senza la parte di sicurezza**

Esempio protezione accessi SNMP

```
access-list 10 permit 132.5.1.1
access-list 10 permit 132.5.1.2
snmp-server community public RO 1
snmp-server community private RW 1
```