RETI DI CALCOLATORI

ESERCIZI RIEPILOGATIVI
Lezioni 1 - 30 con Risposte

Luca Agostini

Esercizi relativi alle lezioni 1, 2, 3, 4, 5, 6

1) Descrivere le differenti tipologie di canali trasmissivi

Un canale di comunicazione è un collegamento fisico oppure logico per trasportare informazioni tra due entità, ad esempio due calcolatori.

I canali possono essere:

· Punto-punto, in cui l'interconnessione è tra due entità e la trasmissione può essere unidirezionale o bidirezionale.

· Multipunto, in cui la connessione è tra più nodi. Uno dei nodi ha il controllo del canale e tale nodo è detto master mentre gli altri sono detti slave. E' un tipo di canale non più molto diffusa.

· Broadcast, in cui il canale collega più nodi e ogni trasmissione raggiunge tutti i nodi, per cui sono necessari indirizzi, per far capire il destinatario della trasmissione.

2) Quali sono le principali caratteristiche di una Local Area Network (LAN)? Di una Metropolitan Area Network (MAN)? Di una Wide Area Network (WAN)? Discutere le principali differenze tra LAN, MAN e WAN

Le reti si possono distinguere dalla loro distanza, ovvero si distinguono dalle distanze che intercorrono fra i sistemi: si va dai centimetri (network on chip) alle migliaia di chilometri (interconnessione di reti geografiche). Le diverse distanze tra le reti comporta un utilizzo di tecniche diverse. Anche il tipo di informazione scambiata è diversa in funzione del tipo di rete. Ci sono principi fondamentali validi per tutti i tipi di rete. Le tre più importanti tipologie di rete sono:

· LAN - Local Area Network. Elevata velocità di trasmissione, transmission rate 100Mb/s. Ha una copertura limitata, che va da una decina di metri (una stanza) al chilometro (campus). E' conforme allo standard ISO/IEEE.

Le reti locali, solitamente chiamate LAN, sono reti private installate all'interno di un singolo edificio o campus, con dimensione fino a qualche Km. Sono largamente impiegate per collegare persona1 computer e workstation negli uffici delle aziende e nelle fabbriche, allo scopo di condividere risorse (per esempio stampanti) e scambiare informazioni.

Le LAN si distinguono dagli altri tipi di rete per tre caratteristiche: (1) la dimensione, (2) la tecnologia di trasmissione, (3) la topologia.

Le reti LAN hanno dimensioni contenute, il che significa che il tempo di trasmissione più sfavorevole ha un lite, che è noto. Conoscere questo limite permette l'uso di alcune tecniche, che non sarebbero altrimenti applicabili. Semplifica inoltre la gestione della rete.

Le LAN possono usare una tecnologia di trasmissione rappresentata da un cavo a cui sono connesse tutte le macchine, come le linee telefoniche duplex usate in passato in alcune zone rurali. Le LAN tradizionali lavorano a velocità comprese tra 10 Mbps e 100 Mbps, hanno bassi ritardi (microsecondi o nanosecondi) e hanno pochissimi errori.

· MAN – Metropolitan Area Network. Velocità di trasmissione medio alta, >2Mb/s. Utilizzo su estensioni cittadine, dell'ordine di 10 km.. Standard sia ITU-T sia ISO/IEEE.

· WAN – Wide Area Network. Velocità di trasmissione molto bassa. Copertura molto vasta, a livello di nazione e continente, dell'ordine che va da una decina di chilometri a un migliaio di chilometri. Standard ITU-T.

Una Wide Ara Network o WAN copre un'area geograficamente estesa, spesso una nazione o un continente. Racchiude una raccolta di macchine destinate a eseguire programmi utente (applicazioni). Seguendo le consuetudini, queste macchine vengono chiamate host. Gli host sono collegati da una communication subnet, per brevità chiamata semplicemente subnet (sottorete). Gli host sono di

proprietà dei clienti (per esempio i persona1 computer degli utenti), mentre la communication subnet è generalmente posseduta e gestita da una compagnia telefonica o da un Intemet service provider.

Queste reti hanno una gerarchia, con la WAN in alto e la LAN in basso.

Le trasmissioni dati sono attualmente ad alta velocità in quanto a banda larga e questo perché il mezzo trasmissivo installato è una risorsa molto preziosa per cui si cerca di utilizzarla al massimo cercando di trasferire il più possibile informazioni.

3) Qual'è la differenza tra commutazione e multiplazione?

La MULTIPLAZIONE, ovvero multiplexing in inglese, consiste nella condivisione di un canale fisico.

Può essere basata sul tempo, per cui le stazioni sullo stesso canale condiviso trasmettono a turno.

Può essere basata sulla frequenza, quindi usando tecniche di modulazione, ad esempio WDM (wavelength multiplexing), la multiplazione a modulazione di lunghezza d'onda. Modulare il segnale significa usare diverse frequenze di modulazione, questo comporta una non interferenza di più segnali che sono trasmessi sul canale. Le trasmissioni radio e televisive sono gestite in questo modo.

Può essere basata su codici.

La soluzione trasmissiva affinché i dati trasmessi siano riconoscibili è quella di adottare i pacchetti che hanno una intestazione, ovvero informazioni di servizio, che contiene l'informazione di appartenenza ad una data comunicazione da parte di quel pacchetto di bit. Questa operazione può essere fatto nel calcolatore oppure nel multiplexer. Il demultiplexer, in fase di ricezione dati, riuscirà a dividere opportunamente i pacchetti in base a tale informazione. L'informazione aggiuntiva necessaria a tale tipo di soluzione è detta overhead. Il canale può essere usato dinamicamente, ovvero statisticamente, per la comunicazione che ha dati da trasmettere in quel momento.

Per rendere riconoscibili i pacchetti si può usare la tecnica TDM (Time-Division Multiplexing), in cui l'appartenenza è codificata nella posizione temporale. A tale scopo è richiesta una sincronizzazione tra multiplexer e demultiplexer. In questo tipo di soluzione se una trasmissione non contiene nulla, qualcosa deve essere comunque trasmesso, a differenza della soluzione precedente.

Questo tipo di multiplazione, insieme alla commutazione di circuito, di cui parleremo, è quella usata nelle reti telefoniche, tariffate a tempo. A differenza delle trasmissione dati, tariffate a pacchetto e quindi a volume generato.

COMMUTAZIONE [SWITCHING]. Nel momento in cui abbiamo la capacità di multiplare comunicazioni diverse sullo stesso canale possiamo creare dei dispositivi che sono collegati a diversi canali; questi dispositivi sono detti switch (commutatori), router (instradatori), nodi della rete, intermediate systems (sistemi che stanno nel mezzo di una comunicazione). In altre parole, la COMMUTAZIONE è un'operazione all'interno di un nodo, rappresentato da un dispositivo collegato a diversi canali, che tratta l'informazione da trasmettere sotto forma di segnale, affinché sia indirizzata verso la destinazione desiderata.

Un dispositivo del genere ha diversi canali da cui riceve informazioni, di "colore" diverso ed il suo compito è quello di spostare le informazioni da un canale all'altro.

Questo avviene per la loro capacità di multiplexing e demultiplexing.

Ci sono svariati modi in cui i nodi possono realizzare la loro commutazione:

· Nodi a commutazione di pacchetto, packet switching. I più utilizzati, guardano le intestazioni dei pacchetti, le elaborano ed effettuano una operazione di routing ed infine commutano il pacchetto, ovvero lo spostano verso l'uscita opportuna del dispositivo sulla quale quel pacchetto verrà multiplato con altri. Questo è il modo in cui viaggiano i dati a pacchetto in una rete, come ad esempio Internet.

· Commutazione di circuito, circuit switching, che può essere basata sul tempo (usata in telefonia, ma

in disuso) oppure sulla frequenza, come nel caso dell'optical switching (anche quest'ultima tecnica è usata in Internet).

4) Qual'è la principale differenza tra la comunicazione orientata e quella non orientata alla connessione?

Ci sono due grosse famiglie nel contesto di interazione e servizi offerti dai livelli: servizi non connessi e servizi connessi, quindi comunicazioni non orientate oppure orientate alla connessione.

Nelle COMUNICAZIONI NON ORIENTATE ALLA COMUNICAZIONE, CONNECTIONLESS, non è necessario un contatto o un'azione preliminare. Le unità dati, cioè le PDU (Protocol Data Unit) vengono mandate ognuna a se stante (si parla di servizio datagram). Questo tipo di servizio non richiede normalmente informazione di stato, non richiede di mantenere traccia degli scambi precedenti, nè negli End System (ES) nè negli Intermediate System (IS). I servizi non connessi sono di tipo best-effort (non affidabile), cioè si fa il possibile ma non si garantisce nulla, non ci sono conferme e quindi i messaggi possono andare persi. Inoltre non c'è controllo di flusso, quindi i dati possono essere troppi per il destinatario. Non c'è controllo della congestione e quindi i dati possono essere troppi per gli intermediate system (IS). Il servizio connectionless funziona con qualsiasi tipo di canale (sia punto-punto, sia multipunto, multicast o broadcast). E' un servizio più semplice e flessibile, per cui le funzionalità sofisticate sono demandate ad altri livelli, sia sotto che sopra.

Nelle COMUNICAZIONI ORIENTATE ALLA CONNESSIONE, CONNECTION ORIENTED, più sofisticate, richiedono un coordinamento precedente alla comunicazione, quindi occorre un meccanismo (protocollo) di segnalazione e occorrono informazioni di stato negli Intermediate System e negli End System. In questo caso è possibile garantire che i dati siano consegnati correttamente, una sola volta ed in ordine, ma ad un costo e ad una complessità aggiuntiva.

Comunicazioni non orientate alla connessione sono UDP (User Datagram Protocol) e IP (Internet Protocol); connessioni orientate alla connessione sono TCP (Transmission Control Protocol), GPRS (General Packet Radio Service, una delle tecnologie di telefonia mobile cellulare).

5) Nell'ambito delle reti di calcolatori, cosa si intende per "host"?

Per host si intende un sistema collegato ad una rete, una stazione, che riceve ed invia dati sulla rete. E' anche definito End System, a differenza degli Intermediate System che hanno un ruolo di smistamento e reindirizzamento dei dati.

Una rete di calcolatori è una rete do host.

6) Descrivere le caratteristiche principali dei 7 livelli dell'architettura di riferimento OSI

I livelli (layers) dell'architettura di riferimento OSI (Open System Interconnection), sono:

7	Application	Livello applicazione. L'unità dati è specifica dell'applicazione (una pagina HTML, un messaggio di posta elettronica ecc.). Le funzionalità sono specifiche dell'applicazione.
6	Presentation	Livello presentazione. Ha a che fare con l'informazione, cioè il contenuto informativo che va scambiato. Il livello presentazione cerca di adattare la rappresentazione e il formato dei dati che può essere diverso nei due End System in comunicazione. Quindi si occupa di fare una traduzione di sintassi dal formato usato nell'End System mittente a una sintassi di comunicazione, una sintassi usata nel trasferimento. La cifratura è un'altra funzionalità di questo livello.
5	Session	Livello sessione. Ha a che fare con transazioni, che sono comunicazioni più complicate, con procedure che coinvolgono eventuali scambi di dati. Esso ha

a che fare con l'organizzazione della comunicazione. Nasconde interruzioni di servizio che possono avvenire, cioè cerca di mascherare ai livelli superiori una eventuale mancata connettività.

4	Transport	Livello trasporto. L'unità dati può essere un messaggio (un insieme di bit) oppure una sequenza di byte. Il livello trasporto opera End-to-End, cioè le entità in comunicazione si trovano sempre sugli End System e non gli Intermediate System e quindi i punti terminale della comunicazione. Esso è inconsapevole dell'attraversamento di tanti nodi. Una serie di funzionalità che questo livello ha servono per compensare le limitazioni del livello rete o inferiori, quindi, ad esempio fare il controllo degli errori. Può fare controllo di flusso, quindi evitare che il trasmettitore sovraccarichi il ricevitore; può fare il controllo di congestione, cioè evitare che il trasmettitore trasmetta così tanto da sovraccaricare la rete. Il livello trasporto si occupa di adattare i pacchetti, si tratta di un servizio di trasferimento di sequenza di byte, che devono comunque essere organizzati in gruppi per metterli in pacchetti di livello 3.
3	Network	Livello rete. Esso lavora su unità dati che sono pacchetti, un insieme di bit, o PDU (protocol data unit), come da terminologia OSI. Tale livello si occupa della consegna dei dati (dei pacchetti) attraverso nodi intermedi, overo tanti Intermediate System, quindi fa funzionalità di instradamento dei pacchetti ("routing"), di inoltro dei pacchetti ("forwarding"), che sono ricevuti da collegamenti in ingresso e mandati su collegamenti in uscita. Il livello rete si occupa di definire il formato e la modalità di utilizzare degli indirizzi, che identificano gli End System all'interno della intera rete, per poter consegnare i dati all'End System giusto passando attraverso un certo numero di Intermediate System.
2	Data Link, o Link	Livello collegamento. L'unità dati che il livello data link è un frame (o trama), che è un gruppo di bit. Per poter usare sequenze di bit la prima funzionalità realizzata dal livello data link è la cosiddetta frame delineation, cioè capire dove inizia e finisce il frame, data una sequenza di bit. Il livello data link aggiunge i suoi dati in cime ed alla fine dei dati che riceve dall'alto. Un'altra funzionalità di questo livello è il controllo dell'accesso al mezzo. In un canale a bus con molte stazioni che devono trasmettere, deve esser deciso quale di esse deve trasmettere, perché se più di una trasmette allo stesso tempo i segnali interferiscono e si rovinano l'un l'altro. Il controllo dell'accesso al mezzo è detto Medium Access Control, MAC, e viene realizzato dal livello data link. Un'altra importante funzionalità del livello data link è il rilevamento e la correzione degli errori. Quando il livello fisico trasferisce un bit, può succedere che quel bit venga trasferito in modo sbagliato, una qualsiasi ragione. Il livello data link ha anche funzionalità di controllo del flusso, che vuol dire evitare che il trasmettitore trasmetta più di quanto sia in grado di ricevere il ricevitore. Nelle reti moderne non viene fatto nel livello data link, ma è una funzionalità che il livello data link può avere.
1	Physical	Livello fisico, physical layer; ha a che fare, per quanto riguarda le unità dati, con i bit; quindi, poiché i livelli scambiano dati tra di loro, il livello fisico scambia dei bit. Esso definisce come vengono codificati i bit, come vengono rappresentati da un segnale che viene trasmesso sul mezzo. Definisce le caratteristiche fisiche dei mezzi che vengono usati per la trasmissione, le caratteristiche dei segnali elettromagnetici usati su questi mezzi e dei connettori che i mezzi devono avere per collegare le realizzazioni del livello fisico sui vari sistemi. La standardizzazione è molto importante.

7) Come avviene la comunicazione tra i diversi livelli OSI?

L'interazione tra i vari livelli avviene tramite la cosiddetta Protocol Entity (entità protocollare) che realizzala le funzionalità di un livello (N-entity, in riferimento alla realizzazione delle funzionalità del livello N). Una N-entity comunica con una N-entityt remota dello stesso livello ed usa i servizi del livello inferiore (n-1). L a comunicazione tra una N-entity ed una N-1-entity avviene attraverso un Service Access Point (SAP), un punto di accesso ai servizi. Il SAP è anche un modo in cui vengono identificate le entità.

I SAP sono importanti per realizzare il concetto di indirizzo. Un indirizzo deve identificare prima di tutto un sistema su cui sono in esecuzione delle identità protocollari, e deve identificare il processi che stanno generando o ricevendo dati. Quindi gli indirizzi sono alla fine delle sequenze di SAP.

La sequenza di SAP usate identifica la specifica comunicazione tra due entità protocollari.

Le informazioni che vengono mandate da una entità di un certo livello ad un'entità remota dello stesso livello sono dette Protocol Data Unit (PDU), in particolare N-PDU, cioè PDU di livello N. Si noti che quando una N-PDU viene passata al livello inferiore essa viene ribattezzata come /N-1)-SDU, ovvero una Service Data Unit di livello N - 1. Questo vuol dire che essi sono dati di servizio perché il livello N - 1 offre un servizio al livello N superiore, che passa anche delle informazioni di servizio insieme alla (n-1)-SDU. Il livello N - 1 aggiunge informazioni ulteriori che in terminologia OSI sono dette Protocol Control Information (ottenendo un (N-1)-PCI), che sarebbe poi una specie di header del livello. Quindi il livello N - 1 aggiunge informazioni (PCI) all'(N-1)-SDU e così facedno costruisce la sua (N-1)-PDU, che vuole mandare al livello N - 1 remoto. Questo procedimento di prendere le informazioni del livello superiore ed aggiungerci delle informazioni si chiama Encapsulation, incapsulamento o anche imbustamento.

Il generico livello n-esimo fornisce servizi al livello superiore n+1-esimo e interagisce con un livello n-esimo remoto, cioè i livelli n-esimi di stazioni diverse interagiscono tra di loro, tramite un protocollo di livello n-esimo.

Il generico livello n-esimo usa i servizi forniti dal livello sottostante (n-1-esimo).

Due livelli adiacenti interagiscono attraverso interfacce, che definiscono servizi, primitive, offerti al livello superiore. Il livello inferiore fornisce servizi al livello superiore ed il livello superiore usa i servizi forniti dal livello inferiore.

8) Che cosa si intende per round-trip time?

Il Round Trip Time o Round Trip Delay (acronimo RTT) è una misura del tempo impiegato da un pacchetto di dimensione trascurabile per viaggiare da un computer della rete ad un altro e tornare indietro (tipicamente, un'andata client-server ed il ritorno server-client).

Protocolli di trasporto come TCP prevedono la ricezione di un ACK (riscontro) in seguito all'arrivo a destinazione dei dati trasmessi. Essi mantengono la stima del RTT corrente per ogni connessione e usano tale stima per determinare il tempo massimo di attesa di un ACK prima della ritrasmissione dei dati.

9) Cos'è e a cosa serve CSMA/CD? A che livello della pila protocollare OSI si trova?

CSMA/CD è l'algoritmo che decide chi può trasmettere, Carrier Sense Multiple Access withCollision Detection. Gli indirizzi delle stazioni servono per capire chi deve ricevere; questo algoritmo, date tutte le stazioni collegate al cavo, serve per stabilire chi può trasmettere. L'algoritmo si basa sul concetto di "carrier sense": la stazione che vuole trasmettere, prima di trasmettere "ascolta" la rete e guarda se c'è una portante, cioè se c'è qualche altra stazione che sta trasmettendo e quindi sta modulando il suo segnale su una portante, si tratta cioè di capire se c'è un segnale sul canale, visto che il protocollo 802.3 trasmette in banda base. Se c'è un segnale sul canale allora la stazione aspetta e quando ci

si accorge che nessuno sta trasmettendo allora essa trasmette. C'è una collision detection, cioè un controllo di collisione, in cui ci si può comunque imbattere perché i segnali si propagano con una certa velocità finita. In caso di collisione il segnale è intellegibile. Quando la stazione si accorge che c'è stata collisione essa smette di trasmettere, ma non subito, spetta un certo tempo e poi riprova. In realtà la stazione trasmette una "jamming sequence", cioè una sequenza di segnali, non identificabili come bit, ma tali da far capire a tutte le stazioni che c'è stata una collisione e devono pertanto ritrasmettere. La stazione riprova a trasmettere, dopo un tempo casuale, perché le stazioni non devono tutte aspettare lo stesso tempo. L'accesso al mezzo è dunque non deterministico, in quanto la stazione non sa quando potrà trasmettere. Il tempo per accorgersi che è avvenuta una collisione dipende da quanto sono lontane le stazioni, cioè dal diametro della rete, ovvero da quanto è grande la rete.

Per fare in modo che una stazione sia sempre in grado di rilevare collisioni deve essere che il round trip time sia minore o uguale al tempo che la stazione trasmittente impiega a trasmettere il pacchetto minimo, cioè il pacchetto che riesce a trasmettere più velocemente possibile; RTD ≤ min T_{TX}

Il round trip time dipende in sostanza dal diametro della rete diviso la velocità di propagazione del segnale, che dipende dal mezzo trasmissivo, vicino a c, la velocità della luce. Il tempo minimo di trasmissione dipende dalla dimensione minima di un pacchetto e dalla velocità di trasmissione e cioè dal bit rate. Il protocollo 802.3 e Ethernet funzionano a 10Mb/s, che è il bit rate R. Lo standard definisce in 64 byte (512 bit) la dimensione minima dei pacchetti, il tempo minimo di trasmissione è dell'ordine dei 576 tempi di bit, tempi richiesti per trasmettere un bit. Da questi dati possiamo capire quale è la distanza massima tra due stazioni, ovvero il diametro della rete, che risulta essere 5760 m, a 10 Mb/s, quindi quasi 6 km.

Con Ethernet a 100 Mb/s la dimensione diventa 10 volte più piccola e con Ethernet a 1 Gb/s diventa 100 volte più piccolo, ovvero 50 m. In realtà saranno presi degli accorgimenti, ma questo è il calcolo della dimensione minima con l'algoritmo CSMA/CD.

CSMA/CD è un protocollo MAC (Media Access Control), quindi posto al secondo livello del modello ISO/OSI.

10) In che modo è possibile dimensionare correttamente una rete Ethernet?

Dati alcuni parametri, è possibile calcolare l'estensione di una rete, come sopra riportato.

Vale la relazione:

$2 \cdot D / p \leq p_{min} / R$ (2D / p, tempo di percorrenza andata e ritorno), da cui

$D \leq p \cdot P_{min} + R / 2 \approx 2 \cdot 10^8 \cdot 512 / 10 \cdot 10^6 / 2 = 5760$ m, in una rete Ethernet 10 Mb/s

In cui:

D è il diametro della rete; p è la velocità di propagazione del segnale nel mezzo

p_{min} è la dimensione minima del pacchetto (64 byte = 512 bit); R è la velocità di trasmissione, il bit rate

Nel progettare e dimensionare una rete non devono essere superati i limiti massimi degli spezzoni, non devono inoltre essere superati i limiti imposti dal livello MAC e cioè che il tempo massimo dei segnali sulla rete sia tale da permettere alle stazioni i rilevare collisioni. Nella pratica il dimensionamento delle reti Ethernet è molto semplice e basta su due fondamenti: quello di rispettare i limiti fisici di ogni tratta, per cui con l'uso di 10BAE-T essa è l massimo 100m; poi di non avere più di 4 ripetitori in cascata, cioè un pacchetto, per passare da una stazione all'altra, non deve passare per più di 4 ripetitori.

Il numero massimo delle stazioni nella rete dipende da quante devono trasmettere. Le collisioni creano una perdita di efficienza, tipicamente del 30%-40%. Ogni volta che c'è una collisione non si usa il canale. Non ci devono essere troppe stazioni nella rete, per cui essa va spezzata e il dominio di collisione deve contenere poche stazioni. □

Esercizi relativi alle lezioni 7, 8, 9, 10, 11, 12

1) Cos'è, a cosa serve e come funziona uno switch?

Quali sono le differenze tra gli switch e i bridge?

Essi sono apparati che permettono di creare un ponte di collegamento tra domini di collisione diversi, ovvero tra LAN diverse. Le differenze stanno nel fatto che coni switch vengono separati i domini di collisione, mentre con i bridge no, per cui in una rete con molte stazioni si ha perdita di efficienza.

Il switch è un'apparecchiatura che, alla pari di un bridge, collega tra loro diversi segmenti logici di una rete (diversi domini di collisione) e che consente il passaggio di informazioni dall'uno all'altro, impedendo tuttavia che l'intero traffico presente su uno di essi si riversi negli altri, e viceversa, come invece accadrebbe se la LAN Ethernet non disponesse di alcun filtro al proprio interno. Lo switch deve disporre almeno di due porte, anche se nelle configurazioni più comuni ne troviamo almeno 8, mentre nei bridge al massimo ci sono 4 porte.

La primissima tecnica di switching, che eredita in toto la modalità operativa dei bridge, si chiama store-and-forward. Ogni trama che arriva su una delle porte dello switch viene incamerata per intero in una speciale porzione di memoria (buffer) e quindi scartata o trasferita a un altro segmento a seconda dell'indirizzo di destinazione (mac address) indicato al suo interno. L'operazione è velocissima, ma comporta in ogni caso un certo rallentamento perché la trama deve arrivare per intero nel buffer dello switch prima di cominciare a essere ritrasmessa su un'altra porta (a cui corrisponde un altro segmento, appunto). È la tecnica di commutazione più affidabile, poiché prima di rispedire il pacchetto ci si accerta di averlo per intero e se ne verifica la correttezza attraverso il calcolo del crc (Cyclic Redundancy Check), ed è l'unica utilizzabile quando si collegano segmenti funzionanti a velocità diverse, come Ethernet e Fast Ethernet, per esempio. Tuttavia su impianti molto veloci, come nel caso di una dorsale che funziona tutta a 100 Mbps o più, il numero di trame in circolazione è molto elevato e il ritardo che si accumula per la registrazione di ciascuna si fa sentire.

2) Che cos'è e a cosa serve il Filtering Database?

Lo ha come elemento centrale il Filtering Database, o forwarding, che è una struttura dati he usa per fare inoltro selettivo (selecting forwarder) delle trame. Dato un indirizzo MAC di destinazione, al switch serve la porta di inoltro di quella trama, ovvero la porta attraverso la quale quella trama deve essere inviata. Nel Filtering Database abbiamo una serie di righe che contengono un indirizzo MAC e la porta su cui inoltrare pacchetti che siano destinati a quell'indirizzo MAC. La porta viene in qualche modo identificata da un numero o un coppia di numero o in altro modo. Il database permette di non inviare pacchetti su altre porte e questo rappresenta un filtro. Quando un pacchetto ha una destinazione sconosciuta lo switch manda il pacchetto su tutte le porte. Quando lo switch riceve una trama da una certa porta X, esso verificherà se ci sono stati errori. Se non ci sono errori viene acquisito l'indirizzo MAC destinazione e verificato se c'è nel Filtering Database; se non c'è la trama è inoltrata su tutte le porte e tale operazione è detta di Flooding, inondazione. Se la trama ha destinazione la porta X stessa allora viene scartata, ma se la porta di inoltro, secondo quanto dice il database, è diversa da X allora la trama viene inoltrata sulla porta di inoltro. Questo permette l'inoltro delle trame solo dove servono.

Nel caso in cui una stazione venga spostata dalla porta di inoltro le trame non raggiungeranno più la stazione. Per risolvere questo problema si dà un termine di scadenza alle righe del filtering database. Quando le entry del database, ovvero le righe del database, diventano troppo vecchie, esse vengono eliminate.

3) Che cos'è il protocollo Spanning Tree?

E' una soluzione standard al problema con il bridging trasparente che riguarda la presenza di percorsi chiusi nella rete. Poiché il traffico broadcast è inoltrato con il meccanismo del flooding e quindi non viene filtrato, succede che la rete si satura velocemente e si crea il cosiddetto broadcast storm, in cui la rete diventa in una frazione di secondo piena di copie di pacchetti; la stessa cosa succede nel caso di trame unknown. La soluzione al broadcast storm è quella di spengere il bridge, quindi è da evitare.

La soluzione di eliminare i percorsi chiusi consiste nel tagliare (non fisicamente) dei link. Vogliamo comunque avere dei percorsi chiusi, in quanto sono quelli che ci offrono ridondanza e tolleranza ai guasti. Il taglio è ricavato dalla sospensione dell'uso, tramite una soluzione standard che è il protocollo spanning tree che sospende l'uso di alcune porte.

Lo spanning tree trasforma una rete con percorsi chiusi (maglie) in un albero, un grafo.

Operativamente sia ha 1. una selezione del root bridge; 2. una selezione della porta root che sarà quella per raggiungere il root bridge; 3. una selezione di designated port, porte designate a ricevere e inoltrare pacchetti in una LAN.

I bridge devono riuscire a fare questa operazione in modo distribuito. Ogni bridge deve operare queste decisioni per conto proprio scambiando informazioni con gli altri bridge. Occorrono dunque dei parametri di configurazione che determinano quali bridge diventerà il root bridge e quali porte verranno scelte come root porte o come designated port.

Le porte che non sono né root port, né porte designate, non verranno usate, potranno esere ripristinate in caso di guasti, ovvero in caso di cambiamento topologico.

Alla fine una rete con maglie diventa un albero.

Il protocollo spanning tree si basa sullo scambio di pacchetti BPDU (Bridge Protocol Data Unit), che sono pacchetti mandati periodicamente da ogni bridge a un indirizzo multicast predefinito, Esistono due tipi di BPDU, le configuration BPDU, usate nella fase di creazione dell'albero e le Topology Change Notification BPDU, usate quando c'è un cambiamento topologico nella rete.

Il primo passo nella creazione dell'albero è la creazione del root bridge, che è basata sul root identifier, un identificatore della radice, che contiene la root priority. All'inizio ogni bridge assume di essere root bridge.

Esso comincia a generare delle configuration BPDU, le C-BPDU che sono mandate ad un indirizzo multicast ed arrivano ovunque nella rete, quindi a tutti i bridge. In un campo è scritto che esso è il root bridge, includendo nella Bridge PDU il proprio root identifier che contiene il proprio indirizzo MAC e la propria bridge priority. Ogno bridge riceve le C-BPDU e confronta il proprio identifier con quelli nelle C-BPDU ricevute.

Esiste un criterio per il quale il bridge capisce se ha diritto a diventare root bridge oppure no. Se non deve essere il root bridge allora include l'identifier del root bridge nelle C-BPDU. Cioè se non può essere root bridge, allora inserisce il root identifier che ha appena ricevuto. In questo caso il bridge assume che sia l'altro bridge ad essere root bridge e lo scrive nelle C-BPDU che genera.

Ad un certo punto tutti i bridge riconoscono lo stesso bridge come root e quindi tutte le C-BPDU contengono lo stesso root identifier. Il bridge che a quel root identifier è a tutti gli effetti la radice, ovvero lo sa lui e lo sanno tutti gli altri.

In sostanza, tutti si candidano, esce fuori quello con diritto maggiore e questo aviiene con le Configuration Bridge Protocol Data Unit, che si propagano su tutta la rete.

A questo punto deve essere selezionata la root port.

Ogni C-BPDU contiene il costo del percorso attreaversato, dalla root fino al punto in cui la C-BPDU viene ricevuta, questa informazione è contenuta nel campo root path cost. Un bridge ha diverse porte e quindi riceverà diverse C-BPDU da queste porte. Nelle C-BPDU che arrivano dalle porte c'è scritto il

root path cost. Il bridge confronta tale valore ricevuto dalle sue porte e sceglie come root port quella da cui riceve C-BPDU con costo minimo, tramite criterio univoco. In questa fase ogni bridge ha una porta radice. La porta radice è quella che ha il percorso migliore verso il root bridge.

Quando è stat selezionata una root port, il bridge smette di inviare C-BPDU sulla root port, quindi le C-BPDU vengono generate dal root bridge su tutte le sue porte, gli altri bridge generano C-BPDU su tutte le porte esclusa la radice, per cui le c-BPDU viaggiano dalla radice verso le foglie.

Le trame dati, non le C-BPDU, che sono inviate dalle stazioni, vengono inoltrate dai bridge attraverso le varie porte e raggiungono la radice attraverso la root port.

Le root port fanno si che il traffico vada verso la radice. Quando il bridge radice inoltra i pacchetti ricevuti sulle sue altre porte, questi discendono attraverso la root port fino alle foglie. Il traffico si propaga quindi lungo l'albero. Dalla foglia verso la radice e dalla radice verso la foglia. La root port è quella con minimo percorso.

A questo punto occorre realizzare l'ultimo passo dello spanning tree e cioè la selezione della designated port, ovvero della porta designata a inoltrare traffico su ogni LAN.

Se c'è una LAN con più di una porta allora ci saranno informazioni che sono arrivate dalla radice e quindi esse avranno seguito percorsi diversi. Il costo del percorso dalla radice viene incluso nelle C-BPDU. Dal confronto dei costi i bridge scelgono in modo coerente quale delle porte sarà designated. Quella non designated smette di trasmettere C-BPDU. Tutte le altre sono poste in stato blocking.

Quindi i passaggi tottali sono 3:

• Identificazione del bridge radice. • Identificazione della porta radice. • Identificazione delle porte designated.

Con questi passaggi si crea un albero che permette di evitare il broadcast storm e non ha percorsi chiusi.

C'è però la necessità di reagire a cambiamenti topologici, dovuti ad esempio ad un errore, cioè una porta o un collegamento non sono funzionanti, oppure si verifica il fallimento del Link Integrity Test, oppure una C-BPDU (che vengono generate periodicamente) non viene ricevuta entro il tempo previsto.

Un bridge che si accorge di un cambiamento topologico reagisce generando una TCN BPDU (Topology Change Notification BPDU), che è inviata attaraverso la root port per raggiungere più velocemente possibile la radice; essa è una trama di servizio, diversa.

La radice imposta un bit particolare detto Topology Change Bit nelle Configuration-BPDU che genera. I bridge che ricevono tale C-BPDU dalla root port, a loro volta, impostano un altro bit, detto Topology Change Acknowledgment nella loro C-BPDU, che si diffondono. Il bride che apprende di un cambiamento topologico svuota il filtering database, in quanto esso è stato costruito usando un albero che non è più valido per cui deve essere costruito un albero diverso.

A questo punto viene messo in discussione tutto ciò che era stato scelto: si guarda se il root bridge è sempre lo stesso (nelle C-BPDU), si verifica di nuovo la porta radice e si verificano di nuovo le porte designated o non designated. Il guasto viene recuperato e un nuovo albero viene costruito e includerà anche la LAN oggetto del guasto.

Lo spanning tree ha dei limiti, tra cui quello delle tempistiche per le quali i vari timer che controllano le reazioni non devono essere troppo bassi in quanto i bridge reagirebbero troppo velocemente per cui si potrebbero creare dei loop temporanei e quindi delle maglie con una immediata broadcast storm,

Quindi quello che si fa è far reagire lentamente i bridge, con timer lunghi, ma in questo caso ci possono essere momenti in cui si perde connettività a seguito di un cambiamento topologico.

Inoltre efficienza nei costi e prestazioni non sono punti di forza dello spanning tree, in quanto ci sono collegamenti inutilizzati, che non possono smaltire traffico, mentre altri collegamenti diventano molto carichi creando un collo di bottiglia.

Lo spanning tree crea un albero con una sola strada verso una LAN. Questo è inaccettabile per interconnessioni geografiche. Quindi in questo caso i bridge non vengono usati. Il problema dei bridge è quello di creare un solo albero da usare per tutto il traffico. Questo albero ad un certo punto può diventare congestionato. La soluzione è quella di poter usare alberi diversi a seconda del mittente, ma i bridge non sono in grado di fare questo perché usano un protocollo molto semplice per scegliere l'inoltro dei pacchetti. A differenza degli apparati detti router che usano una soluzione più sofisticata, ottenendo prestazioni migliori.

4) Quali sono le principali differenze tra lo standard Ethernet, Fast-Ethernet e Gigabit Ethernet? Cosa rimane invariato nei tre standard?

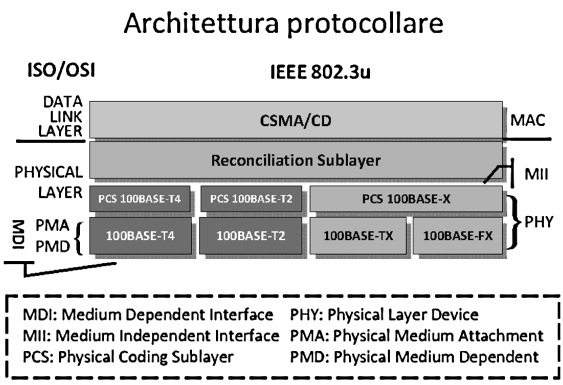
Con il Fast-Ethernet si riduce il diametro massimo della rete; Fast Ethernet conserva tutte le caratteristiche e i parametri di Ethernet:

- Utilizza lo stesso protocollo di accesso multiplo CSMA/CD di Ethernet;
- Utilizza lo stesso meccanismo di gestione delle collisioni;
- Utilizza lo stesso formato e la stessa lunghezza minima.

Con Gigabit Ethernet (IEEE 802.3Z, 802.3AB) si aumenta la durata della trasmissione di una trama minima, detto Carrier Extension, cioè estensione della portante. Si aggiunge in fondo alla trama una parte di bit detta extension bit in modo che i dati (ovvero l'effettiva trama) e la extension bit siano 4096 bit.

Rimane invariato il protocollo MAC CSMA/CD e il formato del pacchetto.

5) Descrivere lo standard Fast Ethernet IEEE 802.3u mostrandone l'architettura protocollare



Lo standard Fast Ethernet 802.3u non va a toccare il livello MAC, che rimane identico a quello dello standard 802.3 (Ethernet), con il CSMA/CD.

Questo protocollo identifica una serie di sotto-livelli, sotto il MAC. Si identifica una modularità, con il sottolivello "Reconciliation Sublayer" comune ai sottostanti sotto-livelli, che avrà funzionalità comuni a tali sotto-livelli, che sono livelli fisici e se ne hanno 4, suddivisi come si nota in figura: quelli a destra hanno una X finale (e vanno sotto il nome comune di 100BASE-X) e quelli a sinistra hanno o T2 o T4 finali.

Il nome inizia per 100 indicando operatività a 100 Mb/s. Base perché si opera in banda base.

I due sotto-standard di livello fisico con la lettera T nella parte finale effettuano una trasmissione su cavo UTP (Unshielded Twisted Pair) di bassa qualità trasmissiva (categoria 3), cavo telefonico.

Il 100BASE-T4 usa tutte e 4 le coppie, usando alcune coppie a turno nella stessa direzione.

Il 100BASE-T2 usa una modulazione molto complessa per riuscire a usare solo due coppie.

Questo permette il funzionamento non solo sugli stessi cablaggi di IEEE 802.3, ma anche su quelli della telefonia, prima ancora.

Purtroppo questi standard, il 100BASE-T4 e 100BASE-T2 sono molto complessi e non sono stati utilizzati, a favore degli standard 100BASE-TX e 100BASE-FX (accomunati nominalmente con 100-BASE-X). Il 100BASE-T4 ed il 100BASE-T2 hanno anche una codifica fisica (PCS, Physical Co-

ding Sublayer) molto complessa.

Gli standard a 100Mb/s più utilizzati sono quelli che vanno collettivamente sotto il nome 100BASE-X.

Hanno due sottolivelli fisici, quello TX che usa cavo tipo telefonico (non telefonico), di categoria più alta con migliori proprietà trasmissive. Poi l'FX che usa la fibra ottica per la trasmissione. abbiamo dunque due standard fisici diversi con funzionalità comuni che sono definiti in un modulo, o sottolivello, comune che si chiama PCS 100BASE-X, in cui PCS sta per Physical Coding Sublayer.

Esso definisce la codifica di linea da usare che è una codifica 4B5B. Vengono presi 4 bit ed essi vengono codificati su 5. Stiamo aumentando il bit rate perché per trasmettere 100Mb/s ne dovremo trasmettere 125.

La ridondanza permette l'utilizzo di simboli di controllo e quello di creare fra essi un codice di IDLE per l'Inter Packed Gap, il momento di silenzio tra due pacchetti.

Questo è importante perchè l'IPG non deve essere fatto spegnendo il trasmettitore e quindi lasciando che il ricevitore si de-sincronizzi dal trasmettitore, ma vengono trasmesse sequenze di bit che identificano un Packed Gap. Il ricevitore può dunque rimanere sincronizzato per ricevere la prossima trama.

I sotto-moduli che sono dipendenti dal mezzo fisico, ovvero i Physical Medium Dependent, si chiamano 100BASE-TX e usano cavi UTP di categoria 5, oppure cavo doppino schermato (Shielded Twisted Pair, STP), oppure 100BASE-FX che usa la fibra ottica.

La codifica di linea in 100BASE-TX è NRZI verso il transceiver, ovvero tra il livello Physical Coding Sublayer e il livello Physical Medium Dependence.

Il trasmettitore sul cavo fisico genera una codifica MLT-3, che ha una occupazione di spettro più ristretta in quanto ha una variabilità minore di una codifica NRZI.

Siccome il Physical Coding Sublayer è comune tra il Physical Medium Dependence 100BASE-TX e quello 100BASE-FX allora viene anche definita dallo standard la codifica da usare tra i due sottolivelli. Quindi un transceiver che opera su fibra ottica è in grado di ricevere lo stesso segnale (quello a codifica NRZI) e trasmetterla sotto forma di segnale ottico.

Lo standard è talmente chiaro che il transceiver può essere pluggable, cioè si possono fare schede su cui si può staccare il transceiver per rame ed infilare quello per fibra ottica, con l'interfaccia verso il Physycal Coding Sublayer ben definita.

La ridondanza nei codici viene usata per assicurare che il segnale (quello sopra) che ha una periodicità più lunga di quella NRZI abbia comunque abbastanza transizione per consentire la sincronizzazione.

Per quanto riguarda il *dimensionamento* della rete usando 100BASE-TX, avendo aumentato 10 volte la velocità della trasmissione il diametro della rete si riduce di 10 volte, con un diametro massimo di 500 m. In realtà poichè dovranno essere usati dei ripetitori, in quanto i cavi rame possono essere massimo 100 m, i ripetitori introducono un ritardo che ha un impatto sul Round Trip Delay e quindi il vincolo posto dal Medium Access Control sulla dimensione massima del dominio di collisione è 205 m.

Avendo usato 100BASE-FX, occorre rispettare i vincoli del livello MAC con la dimensione massima della rete limitata dal CSMA/CD (se non si opera in full-duplex, in cui le stazioni sono collegate direttamente agli switch, con una lunghezza massima di quasi 500 m.).

Usando ripetitori con tratte in fibra, possiamo usare un solo ripetitore e possiamo fare tratte tali per cui il dominio di collisione sia al massimo 300 m. Ovvero tra due stazioni non ci devono essere più di 300 m.

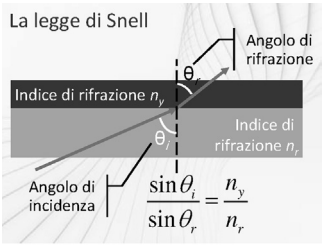
6) Come è strutturata una fibra ottica?

In che modo l'informazione viene trasmessa attraverso una fibra ottica?

I PRINCIPI DI COMUNICAZIONE OTTICA per capire come mai ci sono due standard per fibra ottica, che usato onde di lunghezza diversa.

La comunicazione ottica usa fibre ottiche, Laser e LED, con parametri che influenzano la comunicazione ottica.

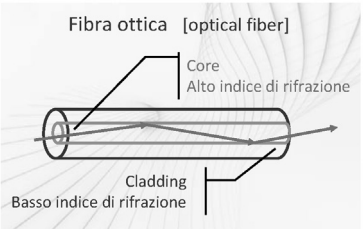
La legge di Snell si applica quando un raggio elettromagnetico incide su una superficie di separazione tra due dielettrici che hanno indice di rifrazione diversi, indice legato alla velocità di propagazione nel dielettrico. Velocità proporzionale alla velocità della luce e di tale indice.



A lato la legge di Snell, con il fenomeno della rifrazione.

Sotto il fenomeno della riflessione, quando gli indici di rifrazione sono molto diversi.

Le fibre ottiche sono basate sul fatto di avere una superficie di separazione tra due materiali dielettrici in cui uno ha un indice di rifrazione molto più alto dell'altro. Viene mandato un segnale all'interno del primo materiale e questo segnale si riflette sulla superficie.



La costruzione della fibra ottica avviene inserendo un materiale dentro l'altro.

Quindi la fibra ottica contiene all'interno il materiale ad alto indice di rifrazione e rappresenta il core della fibra.

L'altro materiale, a basso indice di rifrazione è detto cladding.

I segnali ottici si propagano nella fibra ottica senza uscire e questo diventa un modo per trasferire segnali ottici, anche a lunga distanza in quanto la fibra, se costruita bene ed ha poche impurità, attenua poco il segnale. Il segnale si propaga a lungo senza perdere troppa potenza e quindi è ricevibile.

Ci sono diverse problematiche, ad esempio il fenomeno per cui un segnale nella fibra si propaga in modi diversi, e quando arriva dall'altra parte ha subito una certa distorsione.

Quanto sia accentuato questo fenomeno dipende dagli indici di rifrazione e dalla dimensione del core.

Ci sono due grosse famiglie di fibre ottiche: fibre multi-modali e fibre mono-modali.

Le prime hanno un core, relativamente, più grandi e sono misurati in micron, micrometri.

La produzione delle fibre è un aspetto molto importante, la loro produzione deve essere fatta in modo molto opportuno e devono essere molto pure, in quanto le impurità hanno impatto sulle prestazioni.

Sono fatte in silicio, con produzione anche in plastica, più grandi, usate per applicazioni "automotive".

La trasmissione avviene tramite LED, light emitting diode, oppure con LASER.

La modulazione che si adotta nella trasmissione è di tipo on-off, quindi si accende il segnale per trasmettere un 1, lo si spegne per trasmettere uno 0.

La ricezione avviene mediante un photo detector e la trasmissione avviene sempre in modo sincrono, cioè il ricevitore deve sincronizzarsi sul trasmettitore.

Attenuazione

Le fibre ottiche introducono una attenuazione, che non è costante. Essa è proporzionale alla distanza. A lato immagine dell'attenuazione (in decibel) per chilometro.

Al variare della lunghezza d'onda (l'inverso della frequenza), effettuata oltre il campo visivo, l'attenuazione decresce all'aumentare della lunghezza d'onda ma soprattutto si nota che essa ha alcuni minimi.

Nella trasmissione ottica si cerca dunque di usare segnali che abbiamo attenuazione in questi minimi. Si dice dunque che esistono tre finestre per la trasmissione ottica, incentrate sulle lunghezze d'onda come mostrato in figura, ovvero 850 nm, 1310 nm, 1550 nm.

Dispositivi diversi usano finestre diverse. Nella prima finestra si possono usare LED, nelle altre si usano di norma solo laser. La distanza sarà maggiore per una finestra più alta perchè l'attenuazione è minore.

Wavelenght Division Multiplexing (multiplazione a divisione di lunghezza d'onda)

E' un concetto importante e consiste nell'idea di inserire più segnali in una stessa fibra, segnali che usano frequenze diverse.

Ci sono due grosse famiglie di division multiplexing: coarse WDM che ha granularità poco fine e usa diverse finestre e Dense WDM che ha una granularità molto piccola dei canali, con un numero di centinaia di canali e, sperimentalmente anche migliaia, nella stessa finestra.

7) Cosa si intende per BSS (Basic Service Set)?

Qual'è la relazione tra BSS ed ESS (Extended Service Set)?

SCENARI DI UTILIZZO delle RETI WIRELESS, standard IEEE 802.11

L'elemento base nelle wireless LAN si chiama Basic Service Set (BSS). Due terminali che comunicano nella stessa wireless LAN costituiscono un Basic Service Set.

Ce ne sono due tipi:

. Indipendent BSS, detto anche Ad hoc network, in cui le stazioni comunicano direttamente tra loro e non hanno bisogno di infrastruttura. Si chiama Ad hoc perché le reti Ad hoc sono quelle che si fanno per uno scopo particolare.

. BSS, basato su Access point, che è un dispositivo per comunicare, in cui la stazione non comunica direttamente con un'altra stazione, ma passa per l'Access Point. Le stazioni non ricevono i segnali da altre stazioni, ma dall'Access Point, pur essendo, esse, in grado di farlo. In questo caso serve una infrastruttura. Le stazioni possono essere distanti da qualche metro a qualche centinaio di metri a seconda delle condizioni del canale (influenzato da condizioni atmosferiche, ostacoli ecc.).

Extended Service Set (ESS)

Si ha un Extended Service Set quando si collegano due BSS, attraverso un Distribution System che collega i due Access Point i quali faranno da bridge. Un Access Point prende le trame dalla stazioni del suo BSS e le propagano sul Distribution System e l'altro Access Point le propaga nel suo BSS. Si crea una unica rete, pur avendo due BSS separati.

I Basic Service Set possono essere completamente separati, a causa ad esempio della distanza fra Access Point, ma i Basic Service Set possono anche essere parzialmente sovrapposti e questo ad esempio al fine di supportare lo spostamento delle stazioni senza che queste perdano connettività.

Il terminale nella zona di sovrapposizione può decidere se collegarsi ad uno piuttosto che ad un altro Access Poit e quindi far parte del BSS1 piuttosto che del BSS2.

Il terminale in movimento si accorge che un Access Point si sta allontanando in quanto il segnale è più debole e quindi può decidere di collegarsi ad un altro e continuare a muoversi nel nuovo BSS.

Serve un protocollo tra i due Access Point affinché questo possa avvenire ed è quello di cui parlavamo prima, ratificato dallo standard IEEE 802.11F.

Questo ci permette di avere copertura e servizio ininterrotto anche se la stazione si muove.

BSS colocate

Le BSS possono essere colocate, cioè completamente sovrapposte. Una stazione, in qualsiasi momento, può decidere quale BSS utilizzare, ovvero quale Access Point utilizzare.

Si usa questo, ad esempio, per la tolleranza ai guasti, si rompe un Access Point e quindi la stazione si collega immediatamente all'altro.

Si tenga presente che una scheda di rete (che identifica una stazione) usa sempre un solo Access Point; ci sono applicazioni particolari in cui è possibile avere schede di rete di rete doppie, che hanno due trasmettitori, due ricevitori ecc., che possono usare due Access Point diversi allo stesso tempo.

Le BSS colocate possono essere fatte per migliorare le prestazioni.

("*Questo per quanto riguarda il livello fisico.*"; segue Servizi del Livello MAC)

8) Quali sono i servizi offerti dal livello MAC nelle reti wireless IEEE 802.11?

I SERVIZI DEL LIVELLO MAC sono:

Autenticazione

E' il primo servizio che il livello MAC offre. Esso è il primo passo per comunicare. La stazione deve dimostrare all'Access Point che è abilitata ad usare la rete wireless.

In IEEE 802.11 l'autenticazione viene richiesta dal terminale, a cui segue l'eventuale conferma dall'Access Point.

E' possibile configurare gli Access Point ad accettare un tipo di configurazione che si chiama "Open system authentication" per cui l'Access Point è aperto e non fa nessun tipo di verifica.

Un tipo di autenticazione più restrittiva è la "Shared key authentication" in cui la chiave (segreta) è precedentemente condivisa tramite un canale sicuro.

L'Access point lascerà l'accesso alla rete solo ai terminali che possiedono quella chiave.

L'altro aspetto importante dello standard è quello della riservatezza della rete, la privacy dei dati.

Riservatezza (privacy)

IEEE 802.11 definisce un meccanismo di riservatezza che si chiama "Wired equivalente privacy", WEP. Questo è un fare in modo che il canale wireless dal punto di vista della privacy sia equivalente ad un canale cablato.

Esso è basato sulla cifratura simmetrica, cioè il terminale e l'Access Point hanno una chiave condivisa

segreta che usano per cifrare i dati. Questa soluzione è molto debole, quindi in seguito è stato ratificato lo standard IEEE 802.11i che prende il nome di "WiFi Protected Access", WPA. Esso realizza dei meccanismi di cifratura e di autenticazione più sofisticati e più robusti.

Associazione (disassociazione)

E' un altro servizio importante del livello MAC, l'associazione o la dissociazione di una stazione all'Access Point. La stazione che vuole usare un certo Access Point deve mettersi d'accordo con l'Access Point e quindi creare una associazione tra terminale e access point e di conseguenza anche con il distribution system, per cui gli altri Access Point verranno a saperlo. L'associazione diventa un meccanismo fondamentale per supportare il roaming, cioè il movimento di una stazione che era prima collegata ad un Access Point e poi si collega ad un altro. Esiste una zona in cui due BSS sono sovrapposti ed una stazione è in grado di usare un Access Point piuttosto di un altro e quello che userà dipende da quello a cui si associa. Ci sarà dunque un protocollo per cui un Access Point dice ad una stazione che la sta usando ed un altro Access Point che dice che non la sta usando più. Quindi quest'ultimo non farà più nulla sulle trame della stazione con cui non c'è più associazione, pur potendole ricevere.

Divenire parte di una rete

Per divenire parte di una rete si effettua l'operazione di "Channel scanning".

Lo standard prevede, all'interno della banda, ad esempio quella a 2,4 GHz, diversi canali di comunicazione, la stazione li prova tutti per vedere se c'è qualche altra stazione e li può provare in due modi diversi.

Il primo in modo passivo, semplicemente ascoltando se qualcuno trasmette oppure in modo attivo provando a generare un segnale per vedere se ci sono altre stazioni.

Quindi quando una stazione deve diventare parte di una rete wireless prima di tutto si deve autenticare poi si deve associare ad un Access Point, e a questo punto adotta i vari parametri di livello MAC e di livello fisico che si usano in quella rete e poi può cominciare ad operare.

9) In che modo viene regolamentato l'accesso al mezzo trasmissivo nelle reti Wireless?

MEDIUM ACCESS CONTROL

Il funzionamento dell'algoritmo di accesso al mezzo, avendo un mezzo condiviso e decidere quale stazione può comunicare.

Le modalità di controllo di accesso al mezzo sono due, distribuito o centralizzato.

L'accesso distribuito è detto "Distribution control function (DCF).

L'accesso centralizzato è detto, nello standard, "Point coordination function" (PCF).

Distribution control function

E' basata su un meccanismo di carrier sense multiple access, simile a Ethernet, in cui le stazioni prima di trasmettere ascoltano il mezzo. Diversamente da Ethernet, dove si aveva collision detection, il DCF s usa "Collision avoidance", cioè si cerca di evitare le collisioni attendendo un tempo casuale prima di ritrasmettere (backoff time). La stazione ascolta e sente che non c'è nessuno sul mezza, ma prima di trasmettere aspetta questo tempo detto backoff time, cercando di evitare che, se un'altra stazione nello stesso tempo ha ascoltato e trovato il mezzo libero si metta anch'essa a trasmettere contemporaneamente provocando una collisione.

In alternativa si può usare un meccanismo di richiesta e di attesa di permesso (RTS/CTS), scambiando due messaggi, Request To Send e Clear To Send.

La ragione per cui si fa questo è che non si può verificare se ci sono collisioni, perché il trasmettitore nel momento che trasmette satura il ricevitore che non può sentire collisioni.

Nelle reti cablate trasmettitori e ricevitori sono collegati a canali fisici, doppini, diversi.

Per questa ragione ci vuole un meccanismo di conferma, cioè di Acknowledgment, dopo la trasmissione.

Quando il ricevente riceve una trama MAC conferma sempre dopo l'avvenuta ricezione.

Point coordination function

Essa prevede un coordinamento centrale, normalmente fatto dall'Access Point.

Si hanno tempistiche controllate, con l'Access Point, o il coordinatore, che usa un meccanismo di "poll" per dire chi può trasmettere. Questo meccanismo può coesistere con il DCF.

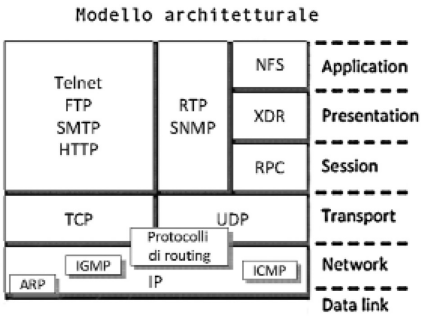
10) Quali sono le caratteristiche della pila protocollare TCP/IP?

L'ARCHITETTURA PROTOCOLLARE TCP/IP

Vengono usati altri protocolli, come UDP (User Datagram Protocol), molto importante; NFS (Network File System), molto usato; ARP (Address Resolution Protocol); ed altri.

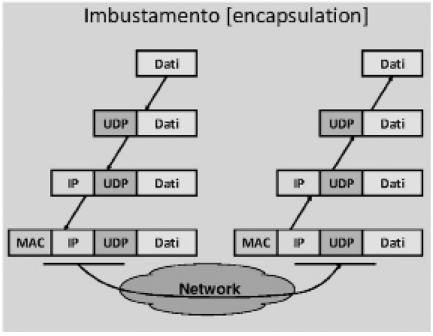
L'architettura TCP/IP è uno "standard" di dominio pubblico, le specifiche sono pubbliche. Essa non è in vero e proprio standard, che lo fa un ente di standardizzazione. Lo è diventato de facto.

E' indipendente da costruttori. I documenti che descrivono i vari protocolli e le loro specifiche si chiamano RFC (Request For Comment).



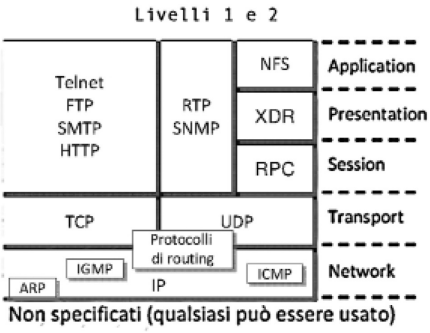
Il modello architetturale a lato mostra come ci siano vari protocolli. Esso è messo in confronto con il modello OSI a destra, in cui non è rappresentato il livello fisico.

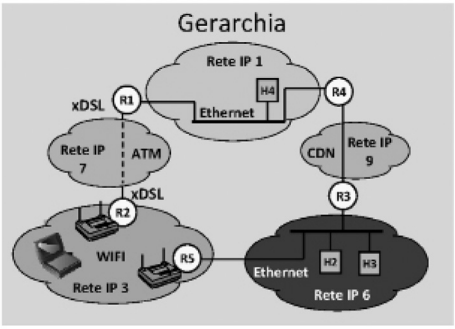
Alcuni protocolli dell'architettura protocollare TCP/IP corrispondono abbastanza fedelmente al modello OSI nelle loro funzionalità, altri no. Questo deriva dal fatto che TCP/IP è una architettura protocollare nata indipendentemente dal modello OSI.



Come tutti i modelli architetturali a strati, in Internet si usa il meccanismo di imbustamento (encapsulation) per cui i dati dell'utente vengono imbustati, confezionati, con l'intestazione di un protocollo di un certo livello e imbustati in pacchetti IP che vengono trasmetti inserendoli in pacchetti di livello data link, quindi aggiungendo ad esempio una intestazione MAC per poi trasferirli nella rete ed essere ricevuti dall'altra parte per essere "de-imbustati".

Le buste vengono aperte e i vari livelli protocollari vengono elaborati fino a che si arriva ai dati che vengono passati alle applicazioni che usano i servizi di rete.





Gerarchia rete Internet: tante "piccole" reti di livello 2 collegate da router.

Livelli 1 e 2

Dall'architettura protocollare si nota che i livelli 1 e 2 mancano. Questo perché la rete Internet deve funzionare con calcolatori omogenei, di qualsiasi marca e di qualsiasi tipo, ognuno con la sua scheda di rete.

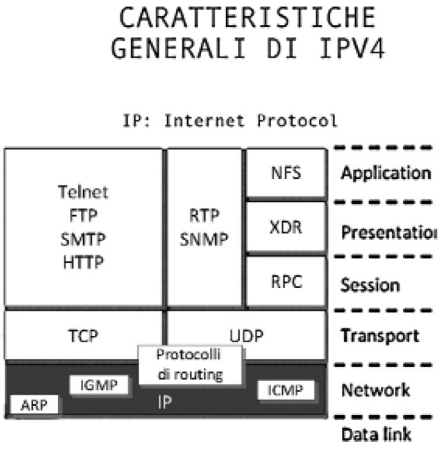
L'idea è che il protocollo IP e la rete Internet devono funzionare indipendentemente dalle specifiche schede di rete usate. Quindi, considerando questo dal punto di vista protocollare, devono funzionare indipendentemente dallo specifico protocollo di livello 2 e di livello fisico che vengono usate.

Nell'architettura protocollare vengono definiti protocolli che funzionano dal livello 3 in su.

Il protocollo principale, quello per trasportare i dati, è il protocollo IP, Internet Protocol, e l'architettura specifica come il protocollo IP specifica come può usare i servizi di moltissimi protocolli di livello 2. Ad oggi i protocolli di livello 2 usati sono pochi: Ethernet, 802.11 (reti wireless), PPP (Point to Point Protocol).

Poiché l'architettura protocollare è indipendente dal livello 2 fa si che la rete sia organizzata in modo gerarchico. La rete Internet è una rete di piccole reti di livello 2, cioè di reti che usano protocolli diversi di livello 2. I dispositivi indicati nei cerchi (R1, R2, R3 e R4) , che sono i router, prendono i pacchetti da una rete di livello e li inoltrano su un'altra rete di livello 2.

I router sono degli intermediate system (in terminologia OSI) che sono responsabili di inoltrare i pacchetti dal mittente al destinatario, da una rete ad un'altra, ad esempio da H2 a H4 in figura.



CARATTERISTICHE GENERALI DI IPv4

Il protocollo principale della rete Internet è IP, di cui analizziamo le caratteristiche generali della versione 4.

Esso è un protocollo di funzionalità di livello network, nella pila OSI, che trasferisce pacchetti attraverso una rete da un intermediate system ad un altro dalla sorgente alla destinazione attraverso, appunto, una serie di intermediate system.

IP: Internet Protocol

Protocollo a pacchetti, la rete è basata sulla commutazione di pacchetto, packet switching.

IP fornisce un servizio non connesso (connectionless) detto anche servizio di tipo datagram. Questo

vuol dire che ogni pacchetto , dal punto di vista del protocollo IP, viaggia per conto proprio ed è indipendente dagli altri.

Il servizio è connectionless perché non richiede che la rete o chi manda un pacchetto e chi lo riceve si mettano d'accordo in precedenza, prima di trasferire i pacchetti. Quando una stazione ha un pacchetto da mandare, lo prende e lo manda nella rete.

La rete a questo punto fa del proprio meglio (concetto di best effort) per portare il pacchetto a destinazione.

Per questa ragione il servizio fornito non è affidabile, in quanto non si può sapere a priori se un pacchetto ce la farà ad arrivare a destinazione o meno.

Saranno i protocolli di livello superiore, il livello 3 o le applicazioni, a preoccuparsi di verificare se i pacchetti arrivano a destinazione ed eventualmente chiedere la ritrasmissione.

IP è un protocollo vecchio, ma non obsoleto. Stiamo usando la versione 4 ma sta subentrando la versione 6.

Datagram rispetto a servizio connesso

Ogni pacchetto attraversa la rete indipendentemente dagli altri pacchetti, quindi due pacchetti che appartengono alla stessa comunicazione possono eventualmente seguire un percorso diverso.

Questo ha delle implicazioni, ad esempio è possibile una consegna fuori ordine dei pacchetti, il che complica la vita al ricevitore.

Inoltre, in un servizio di tipo datagram la gestione delle risorse (p.e. banda) è complessa.

Però tale servizio ha minore complessità, prerogativa delle reti moderne che hanno successo.

Usare un servio datagram lo rende più robusto, cioè si ha una adattamento "naturale" a cambiamenti nel traffico e nella topologia (guasti). In caso di guasti i pacchetti passeranno da un'altra strada. Cosa che non avviene in una soluzione di tipo connesso, in cui le stazioni, i nodi, si mettono d'accordo e il guasto è gestito rimettendosi d'accordo.

Il servizio datagram è adatto al traffico "dati" (bursty), con treni di pacchetti seguito da silenzio, questo tipo di dati bursty è diverso da quello tipo voce.

Il servizio datagram a pacchetti è problematico quando si vogliano realizzare servizi "carrier grade", cioè quei servizi che gli operatori vendono per cui è necessario controllare la qualità del servizio e per cui è necessario un recupero guasti veloce.

In telefonia un guasto è recuperato in tempi dell'ordine di 50 millisecondi, con un servizio di tipo datagram tale tempo è di secondi, decine di secondi se non minuti.

Dunque IP va bene per recuperare guasti a patto che non si debbano recuperare troppo velocemente. Va bene per trasportare traffico dati ma in modo best effort, quando si deve garantire una certa qualità del servizio diventa più complicato perchè la gestione delle risorse è più difficile.

Il protocollo IP, Internet Protocol, specifica:

Il formato dei pacchetti.

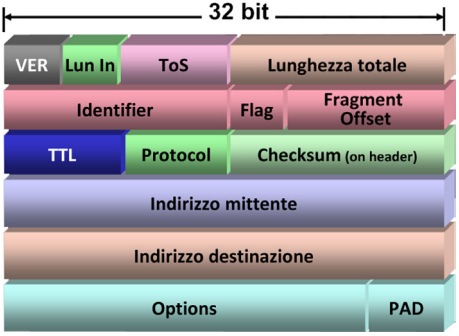
La funzionalità di frammentazione/riassemblaggio [segmentation and reassembly]

Il formato degli indirizzi ["addressing"] e l'assegnazione. Gli indirizzi identificano una stazione.

Specifica il routing [instradamento].

Come si possono realizzare classi di servizio diverse.

11) Descrivere i campi che compongono l'intestazione di un pacchetto IP



FORMATO DEI PACCHETTI

L'intestazione del pacchetto è organizzata su righe di 32 bit (4 byte), come mostrato in figura.

Questo perchè sarebbe difficile rappresentare tutti i campi su una unica riga.

I 32 bit sono poi adattabili all'architettura di un elaboratore che opera a 32 bit, per mezzo dei registri della sua CPU.

Spostamenti dati, ad esempio dalla memoria, per multipli di 4 byte rendono tutto più efficiente.

I campi:

• VER, la versione = 4;

• Lun In, la lunghezza della intestazione; questo perché la lunghezza della intestazione IP è variabile, multiplo di 4 byte; ci sono sempre le prime 5 righe, poi ci sono le Options, campi opzionali e poi riempimento;

• ToS, Type of Service, serve per implementare classi di servizio, per distinguere i tipi di pacchetto, è un campo di 8 bit;

• Lunghezza totale, data dall'intestazione + il campo dati, non riportato in figura, i dati sono dopo l'intestazione;

• Identifier, specifica se effettuare l'operazione di frammentazione o riassemblaggio;

• Indirizzo mittente;

• Indirizzo destinatario;

• Checksum, per il rilevamento errori sull'intestazione, 2 byte;

• Protocol, contiene una indicazione di quale è il protocollo di livello superiore che è contenuto nel pacchetto, questo permette a chi riceve il pacchetto di saperlo interpretare ed elaborare (equivalente al campo Inter Type nell'intestazione Ethernet);

• TTL, Time To Leave, rappresenta il tempo di vita del pacchetto, è un campo molto importante, è un byte, quindi ha valore tra 0 e 255. Ogni router che inoltra il pacchetto verso un altro router decrementa il valore, se il valore è zero, il router non inoltra il pacchetto e lo butta via. Questo perché i router non conoscono la strada del pacchetto e capita che li mandino in percorsi circolari detti loop o loop di routing, senza questo campo i pacchetti girerebbero nella rete per sempre;

• Options, di lunghezza variabile

Formato TLV (type-length-value), cioè il primo byte di Option è il tipo, il secondo è la lunghezza in byte e poi segue il valore;

Esempi di tipo: Source Routing, Route Recording, Timestamp;

PAD: padding [riempimento];

Una stazione che trova un campo options non conosciuto nel pacchetto, lo elabora comunque.

□

Esercizi relativi alle lezioni 13, 14, 15, 16, 17, 18

1) Qual'è il formato degli indirizzi IP?

FORMATO DEGLI INDIRIZZI IP

Indirizzi IP

- Hanno lunghezza 32 bit (4 byte)
- Sono rappresentati in notazione decimale puntata(dotted decimal notation)
- Ogni byte è espresso come numero decimale separato da un punto, ad esempio 12.4.56.38 oppure 193.129.3.215
- Ogni elemento assume un valore tra 0 e 255

L'indirizzo è lungo 32 bit, esso deve essere organizzato come un prefisso che identifica la rete, seguito da una parte di host che identifica l'host all'interno della rete. Come facciamo a sapere quali di questi 32 bit costituiscono l'identificativo di rete (prefisso) e quali l'identificativo di host. Quello che ci chiediamo è quanto è lungo il prefisso: avere una dimensione fissa sarebbe troppo limitativo. Ad esempio con 2 byte per l'uno e due byte per l'altro, identifichiamo 65536 possibilità, ma se la rete fisica ha 100 host sprechiamo tantissimi identificatori di host che non possiamo usare da nessuna altra parte. Se il prefisso è troppo lungo abbiamo pochi identificatori di host. Non potendo decidere a priori, si stabiliscono tre dimensioni di prefisso, come segue:

Class A: 1 byte Indirizzi di Class A

Class B: 2 bytes Indirizzi di Class B

Class C: 3 bytes Indirizzi di Class C

Per identificare se il prefisso è di Classe A, B o C si deve verificare il primo byte. Il valore del primo byte ci permette di capire se l'indirizzo è di Classe A, B o C. In particolare si dovrebbero guardare i primi bit.

Classe A

Il primo bit ha valore 0, con il primo byte che ha un valore tra 0 e 127, per esempio 84.240.20.1;

Max 128 prefissi di rete (Network)

Max 16M indirizzi per host;

Classe B

L'indirizzo inizia per 10, quindi il primo byte ha valore tra 128 e 191, per esempio, 153.240.20.1

Max 16K prefissi di rete (Network)

Max 64K indirizzi per host

Classe C

L'indirizzo inizia per 110, quindi il primo byte ha valore tra 192 e 223, per esempio, 203.240.20.1

Max 2M prefissi di rete (Network)

Max 255 indirizzi per host

Si può continuare

Classe D

L'indirizzo inizia per 1110, con un valore possibile tra 224 e 239, per esempio, 225.240.20.1

Usati per multicast, una stazione manda un pacchetto che ha per destinazione un gruppo di stazioni.

Classe E: anycast

Tutte le classi successive alla C hanno indirizzi non assegnati a interfacce, mentre gli indirizzi delle classi A, B e C identificano le interfacce, con una parte che identifica la rete ed una parte che identifica l'host. La parte che identifica l'host può assumere alcuni valori particolari.

Valori particolari del campo host

Tutti 1: è detto directed broadcast

Per esempio, 203.240.20.255, di classe C, con l'ultimo byte con tutti i bit a 1. Il pacchetto viene inoltrato dai router fino alla rete di destinazione sulla quale il pacchetto è destinato ad essere ricevuto da chi è disposto. Il punto è che il particolare identificativo di host con tutti i bit a 1 è riservato, ed è riservato a questo scopo e quindi non può essere assegnato ad una certa interfaccia.

Tutti 0: si usa per identificare la LIS, ovvero la rete logica.

Per esempio, 203.240.20.0 è il cosiddetto indirizzo della rete. La rete che ha indirizzo 203.240.20 è la rete 203.240.20.0. Non è usato come indirizzo destinazione, quindi, in linea di principio, può essere assegnato ad un'interfaccia.

Identificatori di host disponibili

Data una parte di host di n bit, ci sono 2n - 2 identificatori disponibili.

Eventualmente 2n - 1 se l'indirizzo di rete è assegnato ad un'interfaccia, cioè il .0 finale, anche se normalmente non si fa in quanto esso convenzionalmente è quello che serve per dare un nome alla LIS e, nota bene, non mandare pacchetti.

Indirizzi particolari

Tutti 1: limited broadcast

255.255.255.255

Esso non è ricevuto da tutte le stazioni ed inoltre non è neppure routed [inoltrato, instradato] dai router. Il pacchetto, quando inviato, viene propagato nella rete fisica e viene ricevuto da stazioni "interessate" a riceverlo. Non va a tutte le stazioni di Internet e non va a tutte le stazioni collegate alla stessa rete fisica. E' un pacchetto che può essere ricevuto da un certo numero di stazione della rete fisica.

Altro indirizzo particolare e fatto da tutti 0, ovvero 0.0.0.0 che rappresenta questo host, che non ha un indirizzo IP e, volendo mettere un indirizzo del mittente, metterà tutti 0.

Altro indirizzo particolare, che è un insieme di indirizzi, è quello detto di loopback, nella forma 127.*.*.* ovvero con 127 nel primo byte. Se una stazione manda un pacchetto all'indirizzo che inizia con 127, e normalmente si usa 127.0.0.1. Quello che succede è che il livello IP prepara il pacchetto, mette dentro l'indirizzo destinazione, poi, invece di passare il pacchetto al livello inferiore, il livello data link affinché venga mandato via, auto-riceve il pacchetto e lo elabora. Questo serve ad esempio per ragioni di testing, con due applicazioni sulla stessa stazione che possono simulare una comunicazione attraverso la rete esattamente come avverrebbe con il livello IP.

2) Cosa sono il Classfull e Classless addressing?

Per classfull addressing si intendono i prefissi basati sulla classe, con cui abbiamo dei problemi in quanto sono poco flessibili perché hanno una bassa efficienza nell'uso dello spazio di indirizzi (il numero massimo di identificativi, ovvero di stazioni, può essere minore di quello di cui si ha realmente bisgno); inoltre l'assegnazione degli indirizzi deve essere fatta in modo centralizzato per assicurarsi che non ci siano due organizzazioni nel mondo che usano gli stessi indirizzi.

La netmask permette di fare una identificazione dei prefissi non basata sulle classi, ovvero Classless Addressing, cioè prefissi non basati sulle classi. La netmask è una sequenza di bit associata ad un indirizzo IP e serve a demarcare il confine tra la parte di rete e quella di host nell'indirizzo IP. Vd. *NETMASK*.

3) Cos'è una netmask?

NETMASK

Serve a superare le limitazioni ed i problemi che si hanno con le classi di indirizzo, ovvero i problemi con il Classful Addressing, appunto i prefissi basati sulla classe.

I problemi sono poca flessibilità, che portano ad una bassa efficienza nell'uso dello spazio di indirizzi. La classe C dà 254 identificativi, la classe B ne dà 16K, quindi se ho una rete con 500 host, non posso usare la classe C, devo usare la classe B con un enorme spreco.

La soluzione sarebbe quella di poter avere una parte di host di 9 bit piuttosto che di 8 bit, per cui avrei 512 identificativi di host con 9 bit.

Il secondo problema è che l'assegnazione degli indirizzi deve essere fatta in modo centralizzato per assicurarsi che non ci siano due organizzazioni nel mondo che usano gli stessi indirizzi.

Questo si fa avendo un ente centralizzato che assegna i prefissi e, nell'assegnare gli indirizzi, assegna un prefisso naturale, di una certa lunghezza, cioè di un certo valore.

Dal prefisso naturale, ad esempio 130.192, dovrebbe essere possibile per ogni nuova sottorete che vorremmo fare usare un prefisso derivato da quello, ad esempio il prefisso 130.192.5.0. In un altro 130.192.6.0.

Vorremmo poter creare dei prefissi più lunghi a partire dei prefissi naturali brevi.

Per questo ci viene in aiuto la netmask che ci permette di fare una identificazione dei prefissi non basata sulle classi.

La netmask è una sequenza di bit associata ad un indirizzo IP.

Essa serve per demarcare il confine tra la parte di rete e quella di host nell'indirizzo IP.

Dato un indirizzo IP, ad esempio 192.168.10.69, questo è un indirizzo di classe C, per cui il prefissi è dato dai primi 3 byte, 24 bit. Però vorremmo trovare un modo affinchè il prefisso non sia dato dai primi tre byte, ma dai primi 26 bit. Per questo si una netmask che ha i primi 26 bit ad uno e gli ultimi a 0. Quelli a zero sono i bit che nell'indirizzo identificano l'host. Quindi dove ci sono gli 1, l'indirizzo è la

parte di rete, dove ci sono gli 0 è la parte di host.
La netmask è scritta in notazione decimale puntata.

La parte di rete/host può avere qualunque lunghezza.

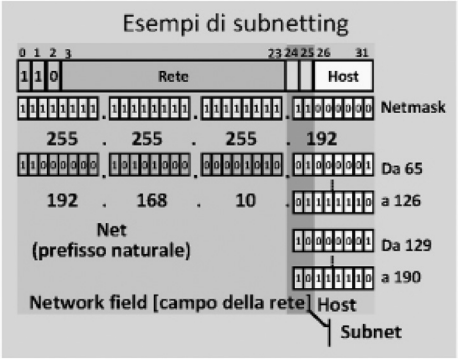
La netmask non può avere valori qualsiasi, di seguito i valori ammissibili per i byte della netmask:

0 0000 0000
128 1000 0000
192 1100 0000
224 1110 0000
240 1111 0000
248 1111 1000
252 1111 1100
254 1111 1110
255 1111 1111

Netmask / Prefissi naturali

I prefissi corrispondenti alla classe si chiamano prefissi naturali e per usarli quando si usano le netmask, si definiscono le netmask cosiddette naturali che hanno un numero di bit a 1 pari alla lunghezza del prefisso specificato dalla classe.

I prefissi naturali sono quelli che si ricavano da una classe. Un indirizzo di classe A ha un prefisso naturale che è un byte. La netmask naturale del prefisso di classe A è 255.0.0.0.



Classe A -> 255.0.0.0

Classe B -> 255.255.0.0

Classe C -> 255.255.255.0

Con la netmask possiamo fare quello che si chiama subnetting o quello che si chiama supernetting.

4) Cosa si intende per subnetting e supernetting?

Subnetting e Supernetting

Subnetting: si prende un certo prefisso naturale per creare un prefisso più lungo di quello naturale, questo tramite la creazione di una netmask con un numero di bit a 1 superiore alla netmask naturale.

Supernetting: si crea un prefisso più corto di quello naturale.

Si fa questo quando si vuole avere un prefisso che riassume in sé una serie di altri prefissi più lunghi utile come informazione ai router per inoltrare i pacchetti. Riduce il numero di informazioni con cui i router hanno a che fare e aumenta la scalability della rete.

Negli esempi di subnetting, si noti come sia evidenziato nell'indirizzo che si ha che vogliamo dividere in una parte di rete e una parte di host. La parte di rete contiene un prefisso naturale (detto anche Net) e l'estensione del prefisso naturale, il campo della rete, detta anche Subnet, come dire che c'è una rete che viene divisa in sottoreti. Dal punto di vista della logica dell'IP abbiamo un identificativo di rete che è il campo della rete e un identificativo di host.

Per dire che il prefisso sarà lungo 26 bit si usa una netmask con 26 bit a 1 e i restanti a 0.

A questo punto, dal prefisso naturale è possibile creare tante subnet, tanti prefissi più lunghi, per esempio il prefisso che ha nella subnet il valore 01 e quindi che ha identificativi di host in cui gli ultimi 6 bit vanno dal valore 000001 al valore 111110, ovvero da 65 a 126 in decimale.

Scrivendo tutto in notazione decimale puntata otteniamo 192.168.10.[da 65 fino 126].

Si può anche definire altre subnet, dove gli identificativi di host hanno sempre lo stesso formato (da 000001 a 111110), ma il prefisso della subnet vale 10 invece di 01 come prima e allora, scrivendo il tutto in notazione decimale puntata abbiamo l'indirizzo 192.168.10.[da 129 a 190].

Dalla notazione decimale non si capisce bene dove finisce il prefisso e dove inizia l'estensione di host, se non si scrive il tutto i binario, per lo meno la parte finale.

Subnetting e assegnazione centralizzata degli indirizzi

Gli indirizzi sono assegnati alle organizzazioni in prefissi naturali, al Politecnico di Torino è stato assegnato il prefisso naturale 130.192. Questo è l'identificativo della Net, della rete del Politecnico di Torino.

Questo viene visto come un "grosso" insieme di indirizzi e la singola organizzazione può usare il subnetting per definire prefissi per ogni rete (ogni subnet all'interno della sua rete aziendale) in modo indipendente dall'ente che assegna gli indirizzi.

5) Cos'è una Routing table? Come è strutturata?

6) Cosa si intende per Prefix Matching?

L'operazione per verificare se due indirizzi hanno lo stesso prefisso o no si chiama prefix matching,

PREFIX MATCHING [confronto dei prefissi]

Stessa LIS: comunicazione diretta

Si suppone di avere un host che deve mandare un pacchetto ad una destinazione che è nella stessa LIS e quindi ha lo stesso prefisso.

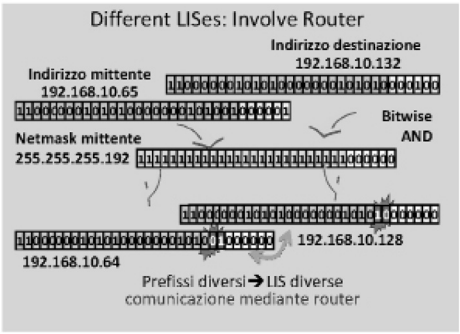
Abbiamo un indirizzo dell'host (192.168.10.65, scritto anche in binario) che deve mandare un pacchetto. L'host ha la propria netmask, fornita alla configurazione ed essa è 255.255.255.192, che indica un prefisso di 26 bit, scritta in binario. Quello che fa l'host è un AND bit a bit tra il suo indirizzo e la sua netmask. Questo avviene in un ciclo di clock. Il risultato è una sequenza di 32 bit, con valore decimale 192.168.10.64, che ha in sotanza il prefisso nei primi 26 bit ed ha una serie di bit a 0 nell'estensione dell'host.

Quindi questa operazione azzera l'estensione dell'host.

Abbiamo poi un indirizzo destinazione, ad esempio 192.168.10.101, l'host dovrebbe in teoria capire

quale è l'indirizzo della destinazione, ma non lo sa perché non ha la netmask della destinazione. Inoltre quello che realmente gli serve è sapere se l'indirizzo di destinazione è uguale al suo. Se quindi estrae dall'indirizzo destinazione un numero di bit pari alla lunghezza del proprio prefisso e lo confrontasse con il proprio prefisso, allora saprebbe se è uguale o no. Se è uguale hanno lo stesso prefisso, se non lo è, esso non è necessariamente il prefisso dell'host che potrebbe essere in realtà più lungo o più corto.

L'host prende l'indirizzo destinazione, ne fa un AND bit a bit con la propria netmask e ottiene una sequenza di bit, che è della stessa lunghezza del proprio prefisso estratta dall'indirizzo originale della destinazione più gli altri bit a 0. A questo punto confronta il risultato ottenuto dalla stessa operazione con il proprio indirizzo e verifica se sono uguali. Se sono uguali i prefissi sono uguali, la sorgente e la destinazione hanno lo stesso prefisso, e dunque la LIS è la stessa e quindi il pacchetto può essere consegnato direttamente.

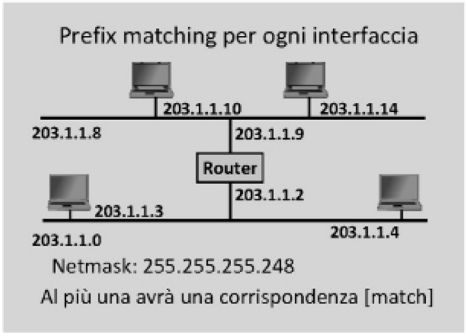


Se non lo sono, vedi "Different LISes: Involve Router", il fatto che i primi 3 byte siano uguali non vuol dire nulla. Alla fine dell'operazione vengono fuori due sequenze di bit diverse, sono uguali i primi 24 bit, man non i successivi due bit. Quindi i prefissi sono diversi, le stazioni appartengono a LIS diverse e sono in reti fisiche diverse, le stazioni non possono mandare il pacchetto direttamente, la comunicazione deve avvenire mediante un router.

7) Cos'è una Default Net Route di una subnet?

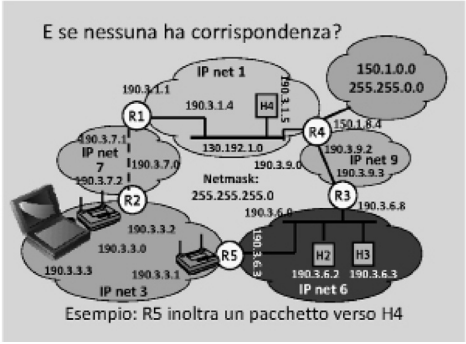
PRINCIPI DI FUNZIONAMENTO DEI ROUTER E SCENARI DI USO DI INDIRIZZI

I router fanno ciò per cui sono "famosi": "ROUTE" I PACCHETTI, ovvero scegliere un percorso per far arrivare alla destinazione i pacchetti.



Prefix matching per ogni interfaccia

Quando un router riceve un pacchetto da un mittente guarda l'indirizzo destinazione del pacchetto e deve fare la stessa operazione fatta dal mittente (prefix matching) per tutte le sue interfacce. In pratica deve essere verificato se la destinazione è in una delle LIS a cui il router è connesso e su quale si trova. Tramite l'operazione di prefix matching il router riesce a capire quale è l'interfaccia su cui deve inoltrare il pacchetto. Il router in figura in alto ha due interfacce e quindi per ogni interfaccia fa una operazione di bitwise AND tra l'indirizzo che ha sull'interfaccia e la netmask che ha sull'interfaccia con l'indirizzo della destinazione e la netmask che ha sull'interfaccia. Facendo questa operazione con l'interfaccia di sopra troverà che i prefissi sono diversi, facendolo con quella di sotto troverà che i prefissi sono uguali e allora sa che può consegnare direttamente il pacchetto usando il servizio Ethernet, se la rete è Ethernet, mettendo il pacchetto IP in una trama Ethernet e mandandolo a destinazione.



Nel fare prefix matching il router troverà al più una corrispondenza, ma ci può non essere corrispondenza, come mostrato nella figura centrale. In questo caso il router si avvarrà della sua routing table, la tabella di routing.

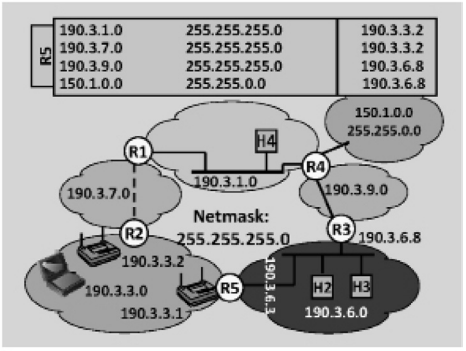
La routing table contiene una serie di righe in cui ogni riga contiene una destinazione e un next hop. Una destinazione è una sottorete logica IP, una Logical IP Subnet, una LIS. Quindi è un prefisso che identifica quella particolare sotto rete logica per cui è una coppia indirizzo/netmask.

Routing table [tabella di routing] di R5

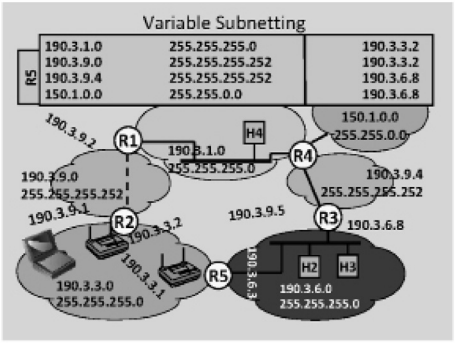
Direttamente collegato

Destinazione		Next hop
190.3.1.0	255.255.255.0	190.3.3.2
190.3.7.0	255.255.255.0	190.3.3.2
190.3.9.0	255.255.255.0	190.3.6.8
150.1.0.0	255.255.0.0	190.3.6.8

Se stiamo usando indirizzamento classless, che è quello che si fa oggi nelle reti IP, allora per capire quanto è lungo il prefisso serve una netmask. Quindi in ogni riga, detta entry, della routing table il router ha una coppia indirizzo/netmask che rappresenta una destinazione e un next hop, che è il prossimo router a cui i pacchetti, per quella particolare destinazione, intesa come LIS a cui la destinazione finale del pacchetto appartiene, devono essere inoltrati. Il next hop è sempre direttamente collegato, in quanto il router deve essere in grado di consegnare il pacchetto. Il next hop ha lo stesso prefisso di una delle interfacce del router.



Ad esempio la destinazione 190.3.1.0 che è su tre byte, è raggiungibile mandando pacchetti ad un next hop che è 190.3.3.2, che è il router R2. Esso è direttamente collegato ed è parte della stessa rete fisica ed ha lo stesso prefisso di R5. Se la destinazione è 190.3.9.0 allora R5 invia pacchetti a 190.3.6.8 che è l'indirizzo (formato sia da 190.3.6.8 sia da 2555.255.255.0) che R3 ha sull'interfaccia collegata alla rete Ethernet 190.3.6.0 che è una rete a cui anche R5 è collegato. R5 guarda con quale delle sue interfacce fa prefix matching e la usa per inoltrare il pacchetto verso R3.



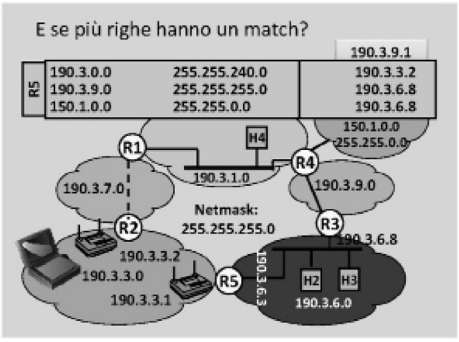
Nel successivo esempio, destinazioni che hanno lo stesso prefisso naturale (190.3, indirizzo di classe B), vediamo che ci sono 3 sottoreti create dallo stesso prefisso naturale con lunghezze di prefissi diversi, determinati dalle netmask. Nella rete dove c'è H4 si è usato un prefisso di 24 bit, sulle due reti che collegano i due router R3 e R4, che hanno molte meno destinazioni e quindi hanno meno necessità di identificatori di host, si usa una netmask di 30 bit, che è la netmask più lunga che si può usare che ci dà due identificativi. L'identificativo della rete

risulta dunque 190.3.9 e 6 bit a zero.

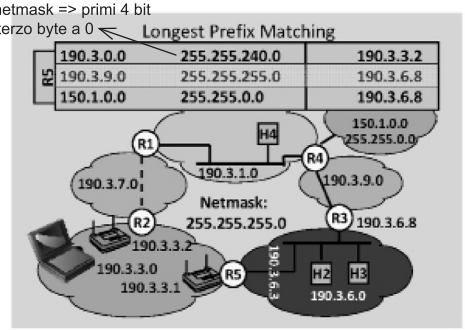
Il router usa la tabella di routing e le entry nella tabella di routing facendo, con un indirizzo di destinazione dato, un AND logico bit a bit tra indirizzo destinazione e la netmask e vede se il risultato è uguale all'indirizzo associato.

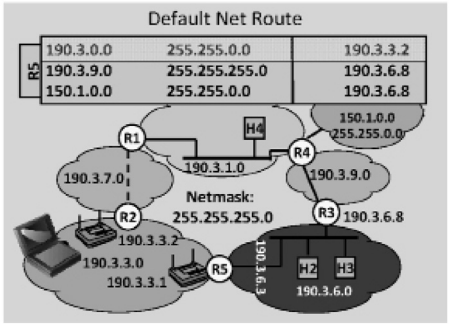
La netmask viene messo in AND bit a bit con l'indirizzo destinazione. Il risultato viene confrontato con l'indirizzo associato: se sono uguali vuol dire che si deve usare il next hop, se non sono uguali, cioè non c'è un matching, allora si passa alla riga successiva. E così via finché il router non trova una riga che fornisca matching.

Se nessuna riga risulta in un matching il pacchetto viene scartato.



La netmask => primi 4 bit del terzo byte a 0





Se più righe fanno un matching, come per le prime due righe dell'esempio in figura, il router deve scegliere un next hop e sceglierà, sempre, quella riga che ha il prefisso più lungo. Il router farà l'operazione detta longest prefix matching. Cerca cioè nella tabella una entry che ha prefisso più lungo possibile che offra un matching e quindi in questo caso di esempio il router sceglierà la seconda riga per cui inoltrerà il pacchetto a 190.3.6.8 (R3), con destinazione 190.3.9.0.

La ragione di questa scelta sta nel fatto che l'informazione più specifica, ovvero la destinazione più specifica, è quella con il prefisso più lungo.

C'è un caso particolare in cui si ha una route (entry nella tabella di routing) più specifica ed è quella che si chiama **DEFAULT NET ROUTE**, di una subnet

Questo avviene quando si ha un prefisso naturale 190.3 con un next hop e poi dei prefissi, delle subnet ricavate dal quel prefisso naturale con un next hop diverso e quindi una route specifica per quelle subnet, come in figura, la prima riga.

C'è anche una route di default più generale, detta appunto default route, che prende la forma mostrata in ultima riga, con una entry in cui la netmask è fatta di tutti 0 e l'indirizzo è fatto di tutti 0.

Questa è una entry molto particolare, per cui qualsiasi indirizzo di destinazione farà matching.

In sostanza questo indica al router di guardare tutte le entry e, nel caso in cui nessuna entry dà un match allora l'entry da usare è l'ultima, la default route.

In questo modo il pacchetto non sarà buttato via e 192.3.6.8 sarà il next hop.

La sequenza di routing, per riassumere, consiste in:

Reti direttamente collegate

Entry [righe] più specifiche

Meno specifiche (aggregate)

Default router

Lezione MOLTO importante

Il piano d'uso degli indirizzi (addressing) ed il routing, in particolare le prestazioni del routing, sono strettamente legati.

Quanto buono sarà il percorso che i pacchetti faranno e quanto grandi saranno le tabelle di routing sono strettamente legati.

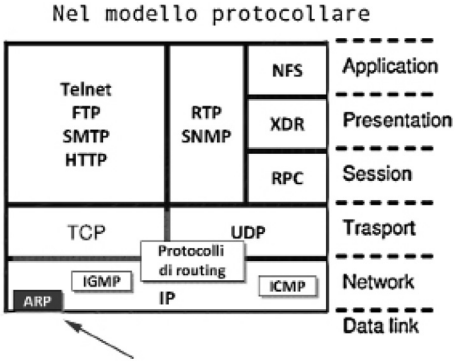
Il routing e le sue prestazioni sono strettamente legati al formato degli indirizzi ed al modo in cui gli indirizzi vengono assegnati alle stazioni e quindi alle varie reti logiche.

8) A cosa servono i protocolli ARP e RARP?

ARP e ICMP sono due protocolli di servizio della architettura di protocolli della rete Internet.

Sono protocolli che non vengono usati per trasportare dati dell'applicazioni dell'utente ma per aiutare il funzionamento della rete dei dispositivi.

ADDRESS RESOLUTION PROTOCOL (ARP) [protocollo per la risoluzione di indirizzi]



A lato è mostrato dove è collocato il protocollo ARP, al livello 3 subito sopra il livello 2. E' un protocollo di servizio e non trasporta dati.

Non ha le tipiche funzionalità del livello trasporto di trasportare dei dati attraverso più hop dal mittente al destinatario ma è un protocollo che aiuta il livello trasporto a funzionare a dovere ed ecco la ragione per cui è collocato a livello trasporto. Nell'immagine è messo subito sopra il livello 2 per indicare che i messaggi vengono imbustati direttamente dentro trame di livello 2.

Caratteristiche generali

→ Protocollo di tipo solicitation basato su broadcast; solicitation vuol dire che una stazione richiede ad altre di fornire informazioni, a differenza di altri protocolli dove una stazione fornisce informazioni direttamente senza che siano esplicitamente richieste. E' basato su broadcast in quanto i messaggi verranno mandati a tutte le stazioni di una rete usando i servizi di broadcast del livello 2.

→ Il protocollo ARP serve per trovare la corrispondenza tra un indirizzo di livello 3 e uno di livello 2

→ Un indirizzo di qualsiasi protocollo di livello 2 (Ethernet o 802.3, in generale indirizzi MAC) e di livello 3 (protocollo IP), ma ARP funziona su ogni tipo di protocollo

→ Il protocollo di livello 3 e di livello 2 a cui si fa riferimento viene specificato in ogni singolo messaggio

Principi di funzionamento

→ I messaggi ARP vengono imbustati direttamente dentro trame di livello 2, nel caso più come in trame Ethernet, ed in questo caso si usa il valore esadecimale sottostante nel campo Ethertype, per specificare che la trama contiene un messaggio ARP

→ Ethertype 0x0806

→ Il protocollo, nel caso di reti TCP/IP, permette di trovare una corrispondenza tra indirizzi MAC e indirizzi IP; tale corrispondenza viene memorizzata nella cosiddetta ARP cache

<MAC address> <IP address> memorizzata in cache

→ ARP cache

→ Quando c'è un pacchetto di livello 3 da inviare, da parte di una stazione

→ Se la corrispondenza tra l'indirizzo della destinazione e il corrispondente indirizzo MAC di quella stazione è nella cache, il pacchetto viene inviato usando quella informazione

→ Altrimenti, si genera una ARP Request, un messaggio ARP request per scoprire quale è l'indirizzo MAC corrispondente all'indirizzo IP di destinazione del pacchetto IP da inviare

REVERSE ARP (RARP) [ARP inverso]

E' un diverso modo di funzionamento dell'ARP.

Caratteristiche generali:

→ Dato l'indirizzo di livello 2 di una stazione, scoprire quello di livello 3;

→ Stesso formato di messaggio del'ARP, con un valore diverso nel campo Operation;

→ Protocollo di tipo solicitation basato sul broadcast;

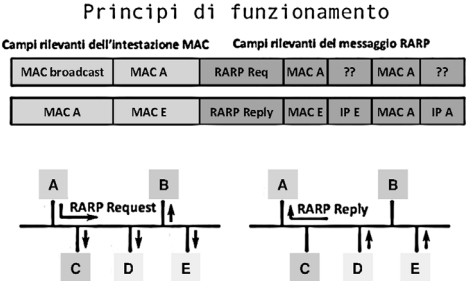
→ Un tempo usato da stazioni senza disco all'atto dell'avviamento [boot] per conoscere il proprio indirizzo IP, conoscendo il proprio indirizzo MAC;

→ EtherType per RARP inoltrato su trame Ethernet: 0x8035

→ Meccanismo rimpiazzato dal DHCP, per la configurazione automatica delle stazioni

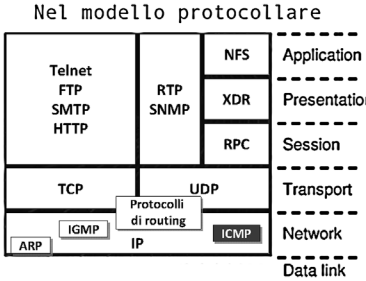
→ Più flessibile

Principio di funzionamento



La stazione A che vuole scoprire quale è la corrispondenza tra un indirizzo MAC e un indirizzo IP, in questo caso il proprio indirizzo MAC, genera una richiesta RARP, la mette in una trama MAC mandata ad un indirizzo broadcast, indicando sia come indirizzo mittente sia quello destinatario il proprio. Non specifica il proprio indirizzo IP perchè non lo conosce. Questa richiesta in broadcast arriva a tutti quanti e se sulla rete c'è qualche stazione configurata a rispondere, come un server configurato a fornire un indirizzo IP alla stazione in base a quello che è il suo indirizzo MAC, può rispondere con una risposta RARP (RARP Reply) che nell'esempio parte dalla stazione E e quindi in una trama MAC va direttamente da E ad A e che contiene l'indirizzo IP di E, ma soprattutto l'indirizzo IP di A per cui A scopre il proprio indirizzo IP.

9) A cosa serve il protocollo ICMP? A che livello della pila protocollare è posizionato?



INTERNET CONTROL MESSAGE PROTOCOL (ICMP)

Nel modello protocollare è posizionato al livello 3, non perchè trasporta i dati (è un protocollo di servizio), ma perchè serve al livello 3, facendolo funzionare meglio verificandone il funzionamento. E' volutamente rappresentato più all'interno del livello 3 rispetto all'ARP per dire che i messaggi ICMP non vengono direttamente imbustate in trame MAC ma dentro pacchetti IP.

Caratteristiche generali

→ Protocollo di servizio (non trasporta dati per le applicazioni)

→ Imbustato in IP

→ Protocol: 0x01

→ Si usa per notifica di errori e anomalie

→ Non specifica reazioni, che è lasciato alla specifica implementazione del protocollo IP

→ L'invio di messaggi ICMP non è obbligatorio

→ I messaggi possono essere ignorati

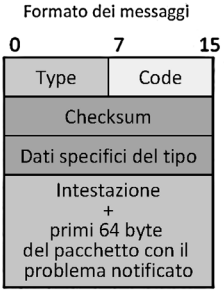
→ Casi d'uso

→ Verificare il funzionamento della rete

→ Notificare anomalie

→ Scoprire la netmask

→ Migliorare il routing



Formato dei messaggi

Messaggio Echo

→ Usato per verificare se un host è raggiungibile

→ Sequence Number è usato per correlare messaggi Reply e Request. Quando una stazione manda un messaggio Echo Request ad un'altra stazione questa risponde con un messaggio Echo Reply.

→ Usato nell'applicazione PING, applicazione molto importante per vedere se una destinazione è raggiungibile.

Questo avviene generando dei messaggi Echo Request e vedendo se arrivano delle risposte Echo Reply. Se arrivano delle risposte la destinazione è raggiungibile, cioè i pacchetti IP possono andare fino a quella destinazione e tornare indietro. Se i pacchetti ICMP non tornano indietro non vuol dire necessariamente che la stazione non è raggiungibile, ma può voler dire anche che la stazione non sta generando risposte Echo Reply perchè non è obbligatorio che la stazione li generi. Nelle reti moderne può anche voler dire che queste risposte siano filtrate da qualche dispositivo intermedio.

Un altro messaggio importante è il messaggio Destination Unreachable, che viene usato da un router quando non riesce ad inoltrare un pacchetto verso la destinazione e nel sotto-campo di Code può specificare perchè non riesce a raggiungere la destinazione.

Messaggio Destination Unreachable

0 Network unreachable (il router non conosce la rete)

1 Host unreachable (il router non riesce a raggiungere l'host)

2 Protocol unreachable

3 Port unreachable

4 Fragmentation needed and DF set

6 Destination network failed

7 Destination host failed

8 Source host isolated

9 Comm. with dest. network administratively prohibited

10 Comm. with dest. host administratively prohibited

11 Network unreachable for type of service

12 Host unreachable for type of service

Un altro importante messaggio è quello di Redirect che serve per suggerire un diverso next hop verso la destinazione. Si usa questo quando una stazione deve mandare pacchetti ad un'altra e usa il suo default gate. Una stazione che non è collegata direttamente manda i suoi pacchetti al default gateway. Il default gateway si accorge che per raggiungere quella destinazione deve inoltrare i pacchetti ad un altro router. A questo punto il default gateway si accorge che la stazione può mandare i pacchetti direttamente a quel router e può mandare un messaggio ICMP di tipo Redirect alla stazione per informare la stazione che i pacchetti li può inviare direttamente al router. Questo tipo di messaggio serve solo per notificare il mittente, non può essere usato tra router.

Messaggio Redirect

→ Non per notificare un router (non è il mittente)

L'importante Messaggio Time Exceed può essere mandato da un router quando questo scarta un pacchetto perchè il Time To Leave è zero.

→ Il TTL in un pacchetto IP è zero

→ Usato nell'applicazione TRACEROUTE, che è usato per verificare il percorso che i pacchetti seguono nella rete. L'applicazione TRACEROUTE mostra quali sono tutti i router attraversati. L'applicazione funziona generando pacchetti con Time To Leave prima uguale a zero, quindi il primo router lo scarterà e manderà una notifica e così l'applicazione impara l'indirizzo del primo router, poi a 1 e si impara l'indirizzo del secondo router e così via.

→ Il reassembly timer arriva a zero

10) In riferimento alle intestazioni dei protocolli di livello trasporto, cosa si intende per "Port"?

Il meccanismo che il livello trasporto ha per capire quale è l'applicazione di livello superiore che deve ricevere i dati e quindi consente di fare la demultiplazione si chiama Port.

La Port è un valore di due byte che si trova nell'intestazione dei messaggi di livello trasporto. Sia i messaggi TCP che UDP cominciano con 4 byte che contengono una Port sorgente (2 byte, valori da 0 a 65536) ed una Port destinazione (2 byte). Il resto dell'intestazione è diverso per TCP e UDP.

Il valore di Source port identifica quale è l'applicazione di livello trasporto che sta generando i dati contenuti nel messaggio di livello trasporto.

Il valore della Destination port identifica a quale delle applicazioni che stanno usando il servizio di livello trasporto i dati vanno consegnati.

→ 0 ... 1023 Porte Statiche, per server, sempre attivi

→ 1024 ... 65535 Porte Dinamiche, per client, ad esempio il browser web, che usano le porte in modo dinamico

11) Cosa sono le "Well-known port"? Fare alcuni esempi di Well-known port

Well Known Port [note], sono porte ad uso dei server che forniscono servizi particolari. Sono definite in varie RFC.

Servizio	port	TCP	UDP	
ftp	21	X		Porta 21, servizio ftp
smtp	25	X		
http	80	X		
pop	110	X		
SNMP	161	X		
DNS	53	X	X	n.b.: 2 servizi di livello trasporto

12) TCP e UDP a confronto: quali sono le differenze tra questi due protocolli e quali sono i possibili scenari applicativi?

TCP e UDP

→ Sono protocolli del livello trasporto e sono in alternativa l'uno all'altro;

→ Servizi diversi perchè hanno caratteristiche molto diverse;

→ TCP fornisce un servizio affidabile, connesso, a byte, in quanto manda delle sequenze di byte a chi offre il servizio, di lunghezza variabile e non manda messaggi;

→ UDP fornisce un servizio di tipo Best-effort, fa del proprio meglio, ma non garantisce nulla per cui non è affidabile, fornisce un servizio di tipo datagram e quindi genera messaggi che vengono portati verso la destinazione. E' non connesso.

UDP (USER DATAGRAM PROTOCOL)

Protocollo di livello trasporto (insieme al TCP, di cui è alternativo). Specifica originale nella RFC 768

Un protocollo datagram

→ Non connesso [connectionless, non serve instaurare una connessione prima della comunicazione, i pacchetti vengono inviati e basta]

→ Non c'è necessità di negoziazione iniziale

→ Ogni messaggio è indipendente dagli altri messaggi che appartengono alla stessa comunicazione, non sono numerati, non ce n'è uno prima di un altro

Servizio best-effort, per cui

→ I messaggi possono
  → Andare persi
  → Essere recapitati fuori ordine

L'applicazione deve tener conto di questo, a fronte della minor complessità a livello trasporto.

Le caratteristiche di UDP sembrano essere quelle del servizio IP. Il protocollo UDP sembra fornire lo stesso servizio del protocollo IP. UDP ha un valore aggiuntivo dato innanzitutto dal multiplexing, per cui molte applicazioni possono usare il servizio.

UDP non aggiunge nulla al servizio offerto da IP in termini di affidabilità e di ordine dei pacchetti, è importante averlo in quanto introduce la possibilità di fare multiplexing e di multiplexing del traffico di applicazioni diverse e quindi di avere le porte.

Valore aggiunto di UDP

→ Multiplexing
  → Port

→ Checksum per verificare l'integrità dei dati
  → Opzionale
  → Non ci sono contromisure

Se la checksum è stata implementata (può non esserlo) ed il pacchetto contiene errori, UDP scarta il pacchetto (messaggio sarebbe più corretto nel contesto UDP, in sostanza sono datagram, un insieme di byte che vengono mandati insieme e hanno una intestazione) e non lo inoltra al livello superiore.

Niente controllo di flusso e congestione

→ Non c'è adattamento alle condizioni di rete
  → Può portare alla congestione dei router (svantaggio), di cui non si accorge; saranno le applicazioni a rallentare il traffico di dati
  → Non si tira indietro, invia sempre (vantaggio, ad uso ad esempio, di applicazioni che devono comunque inviare dati)

Perché UDP?

→ Non stabilisce connessione

→ Non c'è ritardo

→ Non c'è overhead

→ Protocollo semplice → usa meno risorse

→ Non c'è controllo dello stato della connessione

→ Piccola intestazione, per la semplicità del protocollo

→ Non c'è controllo congestione

Formato intestazione

Ci sono 4 campi, ognuno di 2 byte.

Di essi due sono opzionali.

→ Checksum e porta mittente sono opzionali

→ A zero se non usati

Casi d'uso

→ Rete affidabile

→ NFS (Network File System), condivisione dischi, non troppo lontani (Unix, Linux)

→ Affidabilità non richiesta

→ Consegna periodica di dati (un sensore)

→ Media (audio e video, le cui codifiche sono robuste ad un certo livello di perdita)

→ Se il recupero errori può essere problematico

→ SNMP (Simple Network Management Protocol, raccoglie informazioni dagli apparati di rete; serve anche quando la rete è congestionata, serve a maggior ragione in questo caso)

→ Dati in un solo messaggio, contenuti in pochi byte

→ DNS (Domain Name Service, che un protocollo il cui servizio è quello di scoprire un nome associato ad un indirizzo)

→ Il tempo di consegna è fondamentale

→ Per applicazioni real-time [tempo reale]

→ Media (dati audio e video), le applicazioni multimediali sono quelle che hanno portato un incremento notevole del traffico UDP. Le applicazioni tradizionali che usavano UDP prima della multimedialità erano poche e generavano poco traffico.

→ Interattività, i ritardi devono essere bassi

→ Non elasticità

→ Media (dati audio e video), ci può essere tolleranza negli errori, ovvero ci può essere una qualche perdita, ma non un abbassamento del bitrate (tot Mbit/sec).

Traffico elastico, al contrario di quello multimediale, generato da applicazioni elastiche, è ad esempio il trasferimento file, per cui varierà il tempo di consegna se ci sono problemi, ma nulla più, il file alla fine sarà utilizzabile, al contrario di un video che viene trasmesso alla metà di quanto dovrebbe essere trasmesso il quale non sarà utilizzabile.

TCP (transport control protocol)

Protocollo di livello trasporto, alternativo all'UDP.

Caratteristiche generali

→ Protocollo di tipo connesso, si richiede l'apertura di una connessione prima di poter trasferire i dati ed è di tipo full-duplex

  → Full-duplex, una volta che la connessione è aperta, i dati possono essere trasferiti nelle due direzioni

→ Protocollo byte-oriented [a byte], il mittente manda una sequenza di byte, non organizza i dati in messaggi, come nell'UDP; in TCP l'applicazione dice "mandami questa sequenza di byte, poi quest'altra ..." ed il TCP deciderà come organizzare quei byte, che l'applicazione vuole mandare, in messaggi, in quanto il TCP userà, chiaramente, il servizio del protocollo IP e dovrà raccogliere quei byte dentro dei messaggi che vengono messi in pacchetti IP. Quindi la divisione del flusso delle informazioni in messaggi viene fatta dal protocollo TCP stesso, non dall'applicazione che usa il TCP. E questo ha implicazione sulla realizzazione stessa delle applicazioni. Se un'applicazione invia, ad esempio , prima 100 byte, poi 200 byte, il TCP può fare benissimo un unica trasmissione di 100 byte, al che in ricezione non ci si possono aspettare 100 byte e 200 byte spediti all'origine, ma ci si deve aspettare una sequenza di byte di dimensione non conosciuta a priori

→ Il protocollo fornisce un servizio affidabile

  → Byte ricevuti tutti e nel giusto ordine

Casi d'uso

Applicazioni che necessitano di affidabilità

→ FTP: File Transfer Protocol

→ SMTP, POP, IMAP: trasferimento di e-mail

→ HTTP: world wide web, pagine web

Funzionalità

→ Controllo dell'errore

  → ARQ: Automatic Retransmission Request

→ Controllo di flusso, evitare di trasmettere troppo, cioè più di quanto il mittente o il ricevente è in grado di elaborare

→ Controllo di congestione, evitare di trasmettere più di quanto la rete riesce a trasferire

  → TCP solo per applicazioni elastiche

→ Fornisce funzionalità di gestione connessioni

→ (De)multiplexing

→ Segmentazione del flusso dati, è byte oriented

  → No correlazione tra invii e ricezioni

Complesso → Costoso

→ Stato della connessione

→ Memoria

→ Elaborazione

→ Intestazione più grande

→ Overhead di trasmissione

→ Occorrono messaggi di conferma

→ Overhead di trasmissione

→ Ritrasmissioni, anche a sproposito

→ Memoria, per i paccheti ritrasmessi

→ Ritrasmissioni inutili

→ Overhead di trasmissione

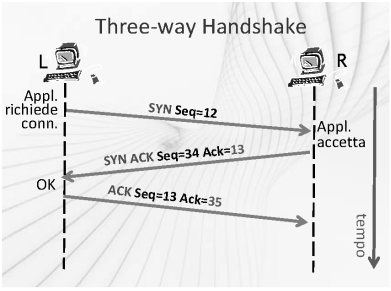
13) Come viene stabilita la connessione dal protocollo TCP?

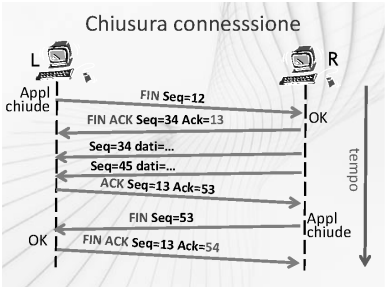
GESTIONE DELLE CONNESSIONI

Cioè l'operazione di apertura e chiusura.

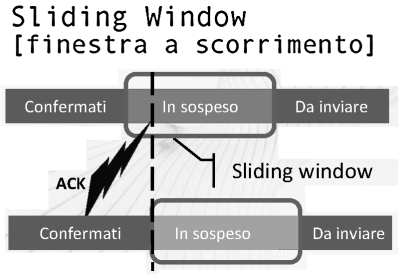
L'apertura della connessione avviene tramite il cosiddetto "Three-way Handshake", detto così perchè richiede lo scambio di tre pacchetti.

La chiusura della connessione comporta quattro messaggi, due in una direzione e due in un'altra, con un significativo overhead.





14) Cosa si intende per "Sliding Window"?



CONTROLLO DI FLUSSO [flow control], è basato su una finestra detta sliding window.

I byte in sospeso sono quelli trasmessi, ma di cui non si sa se sono stati ricevuti o meno.

Il numero di byte trasmessi sono al massimo quelli che stanno in una finestra, la sliding window.

Alla fine della trasmissione dei dati della finestra, il trasmettitore si ferma finchè non riceve degli ACK-nowledgment; se ha meno dati da trasmettere vengono trasmessi tutti quelli da trasmettere.

Alla ricezione di un ACK, che attesta fino a quale byte è stato ricevuto, la finestra viene spostata facendola scorrere in avanti, fino al primo byte confermato. A questo punto ci sono nuovi byte da trasmettere, per poi fermarsi di nuovo. Per una finestra sufficientemente grande il trasmettitore non si fermerà mai perchè gli ACK arriveranno prima che il trasmettitore sia riuscito a trasmettere tutto.

Dimensionamento della finestra

→ Buffer del mittente
  → Byte da confermare
→ Buffer del destinatario
  → Controllo di flusso
→ Rete (buffer dei nodi)
  → Controllo congestione

Il dimensionare la finestra in base al carico della rete, cioè in base allo stato di riempimento dei buffer dei nodi della rete è il cosiddetto controllo di congestione.

Il destinatario, quando riceve i segmenti TCP deve:

→ Consegnare al livello superiore la sequenza di byte completa e ordinata, contenuta in un buffer, di cui c'è bisogno
→ Buffer
  → Riordinare i segmenti
  → Tenere i dati fino a che sono prelevati

Il controllo di flusso è importante perchè il buffer ad un certo punto si può riempire, ed a questo punto eventuali altri byte ricevuti devono essere scartati.

→ Il buffer si può riempire
→ Il campo receiver window serve per comunicare lo spazio disponibile del buffer al trasmettitore
→ La finestra di trasmissione è sempre tenuta più piccola della finestra di ricezione

15) A cosa serve e come funziona il DNS?

Il DNS (Domain Name System) è un meccanismo per la risoluzione dei nomi, da usare al posto degli indirizzi. E' una meccanismo che facilita l'utente.

PRINCIPI DI BASE

In Internet si usano Nomi e Indirizzi

→ I nomi sono più facili da usare per gli utenti
→ Gli indirizzi sono usati per "instradare" i pacchetti
  → Inteso per essere usati dai calcolatori, dai mittenti e dai router par far arrivare il pacchetto alla destinazione. Il DNS fornisce un meccanismo per capire quale è l'associazione tra il nome che si da ad una stazione e il suo indirizzo, che verrà usato dalla rete
→ Un indirizzo ↔ più nomi, indirizzo associato a più nomi
  → Più servizi su un server (FTP, WWW)

→ Un nome ↔ più indirizzi, un nome associato a più indirizzi
  → Bilanciamento di carico [load balancing]
  → Content caching, i contenuti di un server vengono duplicati e tenuti vicini a dove si trova l'utente

La corrispondenza tra nomi ed indirizzi può essere mantenuta in locale, ad esempio in un file.

→ File /etc/hosts
  stazioni di tipo unix
  → 127.0.0.1 localhost
  → 223.1.2.1 alpha
  → 223.1.2.2 beta
  → 223.1.2.3 gamma delta
→ Metodo non pratico su reti di grandi dimensioni
  → Non è scalare, cioè non è in grado di operare su larga scala

DNS fornisce una soluzione gerarchica, che è in grado di operare in situazioni di grosse dimensioni, come TCP.

Quindi DNS funziona su larga scala

→ E' gerarchica dal punto di vista della sintassi dei nomi
→ Gerarchica nell'assegnazione dei nomi, delegata ad autorità responsabili, per cui si crea una gerarchia di autorità a livello territoriale che assegnano i nomi
→ Gerarchica nella risoluzione dei nomi: i server che si usano per risolvere i nomi sono organizzati in una gerarchia. I server costituiscono un database distribuito

Un database distribuito

→ La gerarchia di nomi è usata per trovare l'informazione nella gerarchia dei server
→ L'organizzazione gerarchica dei server e dei nomi è slegata dalla gerarchia di rete (indirizzi e routing). La rete ha una gerarchia di indirizzi, nel modo in cui i pacchetti vengono portati in giro per la rete, il modo in cui viene fatto il routing. La gerarchia del routing nella rete è indipendente da quella del DNS. In altre parole i server in un certo livello gerarchico del DNS non devono essere collocati con lo stesso livello gerarchico degli indirizzi.

GERARCHIA DEI NOMI DI DOMINIO E DEI SERVER

C'è una gerarchia nei nomi di dominio, così come in quella dei server DNS.

Nella sintassi dei nomi di dominio, basata su codifica ASCII, il Top Level Domain (TLD) è la parte che sta più a destra del nome di dominio. Ci sono due tipi di Top Level Domain, il Country Code TLD ed il Generic TLD (ad esempio .it nel primo caso, .com nel secondo). Il dominio di secondo livello è la successiva parte del nome di dominio, dopo il Top Level Domain, può rappresentare organizzazioni o aziende in alcuni TLD, come in polito.it oppure apple.com; può fornire una caratterizzazione (di tipo organizzativo) ulteriore per altri TLDs, come .co in bt.co.uk.

Gerarchia dei server

→ C'è un server per ogni dominio di secondo livello ed in qualche modo c'è una relazione tra il server

del dominio di primo livello corrispondente e quelli di secondo livello

→ Ogni server di dominio di secondo livello conosce gli indirizzi per gli host con nomi nel dominio

→ Ogni server di dominio di secondo livello conosce gli indirizzi dei server che sono responsabili dei domini di livello inferiore

  → Quindi i server sono organizzati nella stessa gerarchia con cui sono organizzati i nomi

→ I server sanno quali sono queste corrispondenze grazie ad una configurazione manuale. Il dNS, cioè, è fortemente basato sulla configurazione manuale

Oltre ai vari server dei singoli domini, esiste un server che si chiama il Root Server, che, essendo di particolare importanza, non è unico, ma ce ne sono diversi, con nomi tipo a.root-server.net, b.root-server.net, c.root-server.net, fino a m.root-server.net.

Root Server, sono a livello più alto della gerachia DNS e conoscono gli indirizzi di tutti quei server che sono responsabili dei domini di livello Top.

→ I root server hanno nome [a-m].root-server.net

→ Sono gestiti da IANA

→ Conoscono gli indirizzi dei server dei TLD

  → ccTLDs: it fr uk

  → gTLDs: com gov aero

Server dei TLD

→ Conosce l'indirizzo dei server del prossimo livello

→ rai.it co.uk nwu.edu

Server del secondo livello, server che conoscono le stazioni che hanno un nome nel dominio di secondo livello; inoltre, siccome alcune organizzazioni potranno avere ulteriori livelli di dominio, i server di secondo livello contengono anche gli indirizzi dei server di livello successivo

→ Nomi delle stazioni

  → www.rice.edu, ftp.nasa.com

→ Server del livello successivo

  → cs.rice.edu, technion.ac.il

Dunque, come spiegato finora, i server sono organizzati in una gerarchia, di cui viene mostrato un esempio nella figura a lato.

La configurazione è fatta a mano, aggiungendo le informazioni nei server quando c'è un nuovo nome di dominio.

La gerarchia è una gerarchia logica, infatti

→ Lo stesso server può essere ospitato su più calcolatori, ci possono essere più copie di uno stesso server, c'è una sola copia logica; in pratica occorre ridondanza dei dati

→ Un calcolatore può ospitare più server DNS, ad esempio del dominio .com e del dominio .net.

→ Sincronizzazione con il server primario

DNS Hosting

→ Il calcolatore che ospita un server DNS non deve essere “vicino” agli host del dominio. Il calcolatore che ospita il server responsabile di rai.it, ad esempio, non deve necessariamente trovarsi sulla rete della Rai, dove c’è l’host www.rai.it, ftp.rai.it eccetera. Esso può essere ovunque. E questo da la possibilità di realizzare servizi di DNS Hosting, per cui un service provider può fornire supporto per la registrazione dei domini

→ Service provider fornisce supporto per la registrazione dei domini. Volendo registrare il dominio pippo.it, se non lo usa nessuno, occorre creare un server, installarlo, configurarlo e dire quale è il suo indirizzo IP affinché sia messo nel database del dominio .it. Tale server con pippo.it deve sempre essere raggiungibile affinchè il nome possa essere risolto, il che può essere difficoltoso in alcune situazioni, ad esempio “in casa”. Quello che si fa è quindi andare da un service provider che permette la registrazione dei nomi; il service provider si fa carico di notificare il gestore del dominio .it e a questo punto il service provider mi fornisce un servizio di DNS Hosting e installa il server per il dominio pippo.it., di cui fornisce l’indirizzo al gestore del dominio .it in modo che quando serve trovare un indirizzo corrispondente ad un nome che finisce per pippo.it il server è pronto a fornire l’indirizzo.

→ Un service provider offre DNS hosting

RISOLUZIONE DEI NOMI

E’ di tipo ricorsivo, ma esiste anche una versione iterativa, dove il primo server chiede quale è l’indirizzo dell’altro server, si ha dunque una specie di rosa di richieste che partono tutte dallo stesso server.

Nella risoluzione dei nomi, la stazione (client) deve avere le seguenti informazioni:

→ Indirizzo di uno o più server DNS

  → Possono essere ovunque

  → Normalmente sono vicini

  → Normalmente il server è responsabile del dominio della stazione, ovvero del dominio in cui la stazione si trova

→ Dominio di default (opz.)

La modalità ricorsiva si usa perchè tutti i vari server che ricevono le risposte possono memorizzare temporaneamente le informazioni che hanno ricevuto e quindi fare una operazione di caching della rcorrispondenza tra nome ed indirizzo. L’operazione di caching viene fatta anche nelle stazioni, in modo da avere già l’indirizzo in caso di ulteriore richiesta da parte dell’utente.

Caching

→ Memorizzazione temporanea [caching] di nomi/indirizzi

→ Anche nelle stazioni

→ Velocizza le interrogazioni da parte dei server, che partono dalla cache

→ Risposte non-authoritative, cioè il server che da la risposta non ha autorità per quella risposta, ma viene fornito l’indirizzo di un server che ce l’ha

  → Indirizzo di un server authoritative

→ Risoluzione iterativa, è una modalità che non permette di sfruttare il meccanismo del caching

Tipi di record DNS (il DNS è un database)

→ A → il tipo di record A (Address) contengono la corrispondenza tra nome e indirizzo

→ MX → da dominio di posta a indirizzo di mail server, occorre dunque conoscere quale è l'indirizzo del server di posta responsabile di quel dominio di posta. E' una problematica diversa che può comunque essere gestita dal DNS, in quanto database contiene anche questo tipo di informazioni

→ CNAME → da alias a nome canonico, canonic name

→ NS → da nome di dominio a indirizzo del suo server

→ Le richieste (query) specificano il tipo di record voluto, ad esempio di può fare una query per record di tipo A oppure di tipo MX ecc.

Inverse Resolution [risoluzione inversa]

→ Dato un indirizzo IP, trovare il nome canonico

→ Stessa procedura e gerarchia di server

→ Si usa un Record PTR (pointer), legato ad un nome fittizio CHE HA UN NOME CANONICO

→ Si costruisce un Nome di dominio fittizio

x.y.z.t.IN-ADDR.ARPA

→ Per esempio, 130.192.3.24 query (PTR)

24.3.192.130.in-addr.arpa

→ Registrazione di polito.it

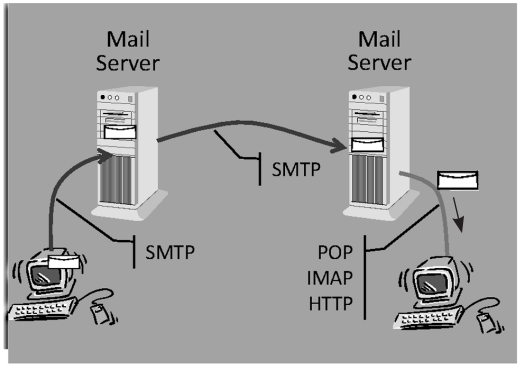
→ DNS server (130.192.3.21)

→ Address range (130.192.0.0)

Esercizi relativi alle lezioni 19, 20, 21, 22, 23, 24

1) Come funziona l'invio e la ricezione di email?

ARCHITETTURA PER IL RECAPITO DEI MESSAGGI di posta elettronica



La posta elettronica è basata sul fatto che un utente riceve i propri messaggi di posta elettronica su un server, detto "Mail server" o anche "Post office". I messaggi permangono sul server finché l'utente, tramite un programma di lettura della posta elettronica (un client), non va a contattare il server, per recuperare il messaggio (a destra nell'immagine).

Serviranno dunque dei protocolli per permettere ad un utente di recuperare i propri messaggi di posta elettronica sul server.

Quando un utente vuol mandare un messaggio di posta elettronica (a sinistra nell'immagine), il programma di posta elettronica contiene un client di posta che va a collegarsi ad un Mail server per trasferire il messaggio di posta al server, il quale aiuterà l'utente a distribuire opportunamente i suoi messaggi di posta elettronica, andando a cercare quale è il mail server che mantiene la casella postale del destinatario del messaggio di posta e trasferire tale messaggio.

Il protocollo per spostare il messaggio dall'utente al server si chiama SMTP, Simple Mail Trasnfer Protocol.

I protocolli per recuperare i messaggi dal server sono diversi, uno è detto POP, Post Office Protocol, IMAP, Internet Message Access Protocol, HTTP, Hyper Text Transfer Protocol, che è anche il protocollo che si usa per il web, ma è anche rilevante per la posta elettronica.

PROTOCOLLI PER IL TRASFERIMENTO DI MESSAGGI

Si tratta di trasferire messaggi dalla stazione dell'utente ad un server e da questo verso il server di destinazione, il protocollo è il protocollo SMTP.

Si basa sul protocollo di livello trasporto TCP sull'uso porta 25, per cui il client apre una connessione TCP sul server che è in attesa sulla porta 25. Il client deve solo sapere quale è l'indirizzo del server, informazione fornita al client in fase di configurazione della posta elettronica. Il server per la posta in uscita è detto anche outgoing mail server.

SMTP: Simple Mail Transfer Protocol

→ Testuale

→ Client-server

→ TCP - porta 25, di default

  → Aperta dal client

→ Command-response (il client manda dei messaggi ed il server risponde)

  → Status code, che dice al client se il server sà soddisfare la richiesta oppure no.

2) Cos'è e a cosa serve MIME?

Lo standard MIME serve a gestire informazioni di un certo tipo, contenute in una email, ad esempio le immagini

→ Le immagini possono essere inviate, ma devono essere codificate come sequenze di caratteri, per esempio base64

→ Il destinatario deve sapere che il messaggio contiene una immagine codificata, deve leggere le informazioni sapendo la codifica, il tipo e decodificare il tutto

Per gestire tutto questo, si usa lo standard MIME.

Multipurpose Internet Mail Extensions: MIME

→ Esso prevede intestazioni aggiuntive, la versione, il tipo, il nome immagine, la codifica ed il contenuto

MIME Version: 1.0

Content Type: image/png; name="image001.png"

Content Description: image001.png

Content Transfer Encoding: base64

iVBORw0KGgoAAAANSUhEUgAAAKgAAABDCA

xAAADsQBlSsOGwAAABl0RVh0U29mdHdhcm

EEQXBBxR19URQF3OEzRo0rKvoeCYm4xJen

Content-Type, contenuti di tipo diverso

→ text
  → plain, html
→ image
  → jpeg, gif, png
→ audio
→ video

3) A cosa serve il protocollo POP?

PROTOCOLLI PER ACCEDERE AI MESSAGGI

Per accedere ai propri messaggi si può usare Webmail, a esempio google mail.

Per fare questo quello che serve è

→ Web server in esecuzione sul calcolatore che ospita il mail server

→ Fornisce accesso ai messaggi tramite interfaccia web

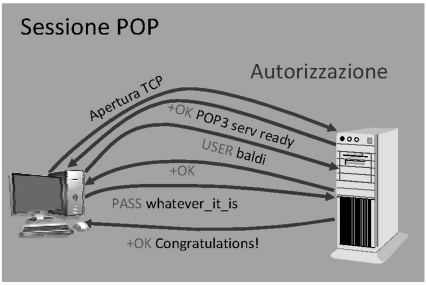
→ I messaggi restano sul server

Pro e contro di webmail

→ Ideale quando non si usi un proprio PC

→ Disponibile ovunque

→ Utilizzabile solo se si ha una connessione Internet



Post Office Protocol: POP

→ Utenti di un singolo PC

→ I messaggi sono spostati sul client

→ Disponibile off-line

→ Protocollo testuale

→ TCP alla porta 110

Sessione POP

+OK implica qualcosa che va a buon fine

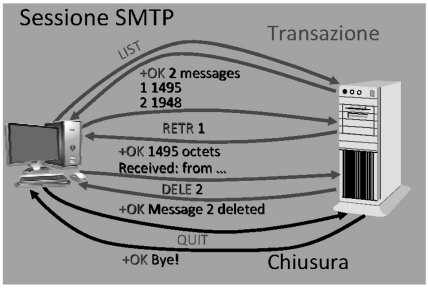
-ERR implica un avvenuto errore

Ai messaggi segue sempre qualcosa di leggibile.

Essendo in una situazione command response, si ha una serie di comandi, di cui USER serve ad autenticarsi, insieme al comando PASS di password

A questo punto, ad autenticazione avventuta, parte una sessione SMTP, a cui seguirà una fase di chiusura.

Sessione SMTP



Internet Message Access Protocol: IMAP

→ Utilizzatori di più PC

→ Per esempio 1 PC al lavoro, 1 PC a casa

→ Protocollo testuale

→ TCP alla porta 143

Unisce il meglio dei due mondi, POP e webmail

→ Disponibile off-line

→ I messaggi rimangono sul server

→ In gerarchia di cartelle

→ Sincronizzazione con copia locale

4) Descrivere i tre standard alla base di Internet: HTTP, HTML e URL

LA RICETTA DEL SUCCESSO del WWW, o Web

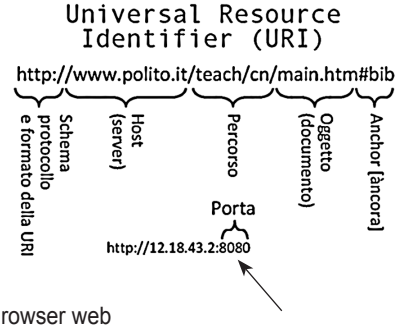
"Ragnatela"su scala mondiale di documenti, in cui le pagine web contengono riferimenti ad altre (links). Il riferimento può essere ovunque in Internet. Il web può essere utilizzato da chiunque.

Gli ingredienti

→ Server
→ Client (browser)
→ Formato dei documenti (Linguaggio HTML)
→ Identificatori (URI, Universal Resource Identifier)
→ Un protocollo (HTTP)

Pagine web

→ Sono oggetti vari composti in una struttura
→ Sono pagine multimediali
→ Scritte nel linguaggio Hyper-Text Markup Language (HTML), che ci permette quali oggetti sono sensibili e come la pagina deve essere visualizzata; il browser fa il rendering della pagina
→ Alcuni oggetti sono "sensibili"



Universal Resource Identifier (URI), è un identificatore di oggetti

→ Identifica ogni oggetto (risorsa), nel web
→ Dice anche dove trovare l'oggetto
→ URL: Universal Resource Locator [localizzatore]
→ Come recuperare la risorsa tramite il sup protocollo
→ Protocollo da utilizzare

Il protocollo http usa come standard la porta 80.

Browser web

→ "Visualizza" pagine web
→ Ne scarica una nuova a seguito di un click
→ Può usare vari protocolli: HTTP, FTP, SIP (per fare una telefonata)
→ Oggetti di vario tipo: Images, video, sound, plug-in

Fattori del successo

→ Intuitivo, "Colorato", Multimediale

HYPER-TEXT TRANSFER PROTOCOL (HTTP)

E' il protocollo usato per tarsferire gli ipertesti delle pagine web, le descrizioni scritte in HTML.

Caratteristiche

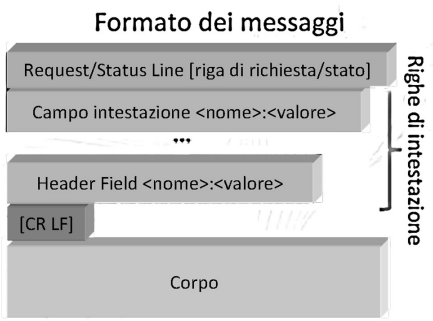
→ Testuale, sequenze di caratteri
→ Basato sul paradigma Client-Server
→ Basato su TCP
→ Apertura da parte client
→ Normalmente porta 80 (server), specificata nella URI

Il protocollo HTTP è Client-Server, in particolare è di tipo Request (Client) - Response (Server)

Il client prima di tutto apre una connessione TCP, poi fa una richiesta HTTP, usando il protocollo HTTP, e il server fornisce una risposta. Poi il client può fare una nuova richiesta, riceve una nuova risposta e così via. Dopo una serie di interazioni può visualizzare la pagina web.

Il protocollo HTTP è stateless, cioè è senza memoria; il server risponde ad ogni richiesta indipendentemente dalle richieste precedenti. Quando al server arriva la seconda richiesta non sa se è legata alla precedente, anche se è la stessa connessione TCP. Ogni richiesta è una storia a parte, la richiesta chiede un oggetto, il server prende l'oggetto e lo passa, l'oggetto è specificato dalla URI.

Questo approccio ha degli svantaggi se si vuole tenere traccia di quello che l'utente ha fatto prima, ad esempio in caso di shopping on-line. L'HTTP nella sua versione base non permette di fare questo, lo fa in versioni successive, con l'aggiunta di qualche meccanismo.



Il formato dei messaggi HTTP

Ci sono delle richieste e delle risposte, in sequenze di caratteri, organizzate in righe. La prima riga si chiama request line nelle richieste oppure status line nelle risposte. Le righe sono terminate da un a capo, dato dai caratteri CR LF.

Dopo la prima riga, abbiamo una serie di campi intestazione, nel formato <nome>:< valore> e terminato con un a capo. L'intestazione termina con una riga vuota in cui c'è solo un a capo, CR LF.

Segue il corpo del messaggio, ad esempio in una richiesta di una pagina web il corpo è vuoto, in una risposta ci sono dei contenuti, che sono la pagina web.

In dettaglio la request line:

Request Line

<metodo> <URL> <versione>

→ Metodi

→ GET, POST, PUT, HEAD

→ Versione

→ HTTP/1.0

→ HTTP/1.1

In particolare GET per prendere un oggetto, POST per dare al server un oggetto nuovo, PUT per modificare un oggetto esistente sul server.

Mettere oggetti sul server significa ad esempio mandare i dati al server di un form di una pagina web.

Si può prelevare solo l'intestazione di un oggetto, con il metodo HEAD, per acquisire informazioni sull'oggetto.

Dopo il metodo abbiamo la URL (o URI) che identifica l'oggetto che vogliamo chiedere al server o che vogliamo dare al server, e poi la versione del protocollo.

Ci sono attualmente due versioni HTTP, una differenza è quella che nella versione 1.0 ogni richiesta e successiva risposta usano una connessione TCP che veniva chiusa alla risposta. Nella versione 1.1

permette di mantenere la connessione aperta, detta connessione persistente.

Esempio di messaggio Request

GET /baldi/pubs/index.htm HTTP/1.1
Host: staff.polito.it
User-Agent: Mozilla/5.0
Accept: text/html,application/xhtml
Accept-Language: en-us
Accept-Encoding: gzip, deflate
Connection: keep-alive

La GET chiede un oggetto al server.

“Connection: keep-alive” significa di chiedere al server di mantenere la connessione aperta. Il server deciderà se è il caso o no.

Status Line, del server

<versione> <status code>

→ 200 OK
→ 301 Moved Permanently
→ 400 Bad Request
→ 404 Not Found
→ 500 Internal Server Error

Esempio di messaggio Response, dal server

HTTP/1.1 200 OK

Date: Sat, 15 Jun 2013 21:17:27 GMT
Server: Apache
Accept-Ranges: bytes
Content-Length: 93589
Keep-Alive: timeout=15, max=97
Connection: Keep-Alive
Content-Type: text/html
<html><head>

“Content-Type: text/html “ è un campo molto importante, perchè dice al client come gestire il contenuto che sta arrivando nel body.

Categorie di status code

→ 2xx successo
→ 3xx redirezione
→ 4xx problema con il client
→ 5xx problema con il server

5) Che funzionalità avanzate offre il protocollo HTTP?

CARATTERISTICHE AVANZATE del PROTOCOLLO HTTP

Autenticazione

Il browser chiede di inserire utente e password; questo succede quando il server richiede una autorizzazione a seguito di una richiesta del client per utilizzare una certa risorsa.

Il client manda un messaggio GET al quale il server risponde con uno status code “401 Unauthorize WWW-Authenticate: <chig>”. In questo caso il server ha incluso anche una autenticatione “WWW-Authenticate“ con una “Challenge”, che è una sequenza di caratteri casuale che il client può usare per autenticarsi.

A questo punto il client fa apparire la mascherina di autenticazione e poi ripete la richiesta GET con le sue credenziali di autenticazione, messe in un campo che si chiama “Authorization“. Per rendere intellegibile le credenziali si usano algoritmi crittografici che operano usando la password sulla “Challenge”. Se l’autorizzazione ha successo il server risponde “200 OK” e invia l’oggetto richiesto.

Da quel momento il client quando comunica con il server manda sempre le credenziali di autenticazione.

Questo tipo di autenticazione è debole, con la variante https di http si usano meccanismi di autenticazione più robusti, più difficili da attaccare.

Cookie

→ Meccanismo per consentire al server di identificare un client prima servito

→ Rende interazione stateful [con memoria] possibile (l’http è di per sè stateless, senza memoria)

→ Si può realizzare il carrello della spesa

→ Preferenze dell’utente

A lato si nota che il cliente chiede al server l’oggetto abc, a cui il server risponde OK (e lo invia), ma invia anche un cookie di valore “xyz”, una sequenza di caratteri, che viene memorizzato dal client.

Ad ogni nuova richiesta (GET) da parte del client viene aggiunta una intestazione con la sequenza del cookie.

Il server correla la richiesta a quella di prima e risponde con l’oggetto. Se il cliente facesse una richiesta senza il cookie, probabilmente otterrebbe una risposta con un nuovo cookie.

MIGLIORAMENTO DELLE PRESTAZIONI

Caching [memorizzazione temporanea]

→ Oggetti memorizzati sul client

→ E se cambiano?

→ Metodo HEAD

→ Campo dell’instazione

If-modified-since: <data>

→ 304 Not Modified

Server proxy

La caching ha vantaggio solo se l'utente accede più volte allo stesso oggetto, invece, usando dei server proxy, il proxy può fare il caching e quindi se un utente chiede un oggetto, il proxy lo va a chiedere al server, quando lo ottiene ne memorizza una copia e se un client chiede lo stesso oggetto viene restituito direttamente.

Questa idea può essere ampliata creando copie dei contenuti vicino agli utenti, grazie al Content Delivery Network (CDN).

Content Delivery Network (CDN)

Quando un utente fa una richiesta, riceve una risposta da un server vicino a lui che ne ha una copia.

6) Come avviene il processo di assegnazione degli indirizzi Internet?

ASSEGNAZIONE DI INDIRIZZI IP alle organizzazioni

Principi di base

→ L'indirizzo IP di ogni stazione deve essere unico

→ Il coordinamento deve essere centralizzato, consiste nel decidere sull'utilizzo degli indirizzi, chi può usare e quale indirizo

→ IANA: Internet Assigned Numbers Authority

→ Meccanismo di delega, affinchè tutti non debbano contattare lo IANA

Anche se abbiamo un ente che ha responsabilità su un solo continente, questi enti delegano gli ISP, attaverso una ulteriore delega:

→ Internet Service Provider (ISP)

→ Centro sistemi informativi

→ Gestore informatico di dipartimento/laboratorio

Tutto questo risulta pesante per ogni nuova sottorete (LAN), per cui lo IANA ha definito l'esistenza di indirizzi privati.

INDIRIZZI PRIVATI

Qual è l'idea?

→ Chiunque li può utilizzare senza chiederne conto
→ Ci saranno duplicati
→ Non possono essere usati "su" Internet
→ Solo dove si è sicuri che siano univoci

I dispositivi comunicano tra di loro in un ambito privato.

Qual è il problema nel comunicare con il resto del mondo? Il problema è che i router inoltrano i pacchetti sul percorso più breve, quindi, avendo due stazioni B, come in figura, che hanno lo stesso indirizzo IP, nel momento in cui A vuole mandare un pacchetto a B, i router inoltreranno il pacchetto sul

percorso più breve, per cui sarà raggiunta solo la stazione B più vicina.

Indirizzi privati

→ 10.0.0.0/8
  → 1 prefisso di classe A
→ 172.16.0.0/16 - 172.31.0.0/16
  → 16 prefissi di classe B
→ 192.168.0.0/24 - 192.168.255.0/24
  → 256 prefissi di classe C

Volendo usare un indirizzo in ambito ristretto, in ambito locale, perchè non ne utilizziamo uno qualsiasi? Cioè, per collegare le stazioni di una rete casalinga, perchè non posso usare un indirizzo qualsiasi? La ragione sta nel fatto che, se per qualche ragione, da qualche parte, in Internet, c'è un host pubblico che ha lo stesso indirizzo di un host privato, i pacchetti spediti da un host locale all'host locale con stesso indirizzo di una qualche host pubblico nella rete Interne, quest'ultimo non potrà mai essere raggiunto.; questo è il concetto di occultamento della destinazione. E' quindi importante aver definito degli indirizzi da usare a livello locale che nessuno cercherà di usare per dei server pubblici.

La struttura di reti private è definita come intranet, che può essere sia pubblica che privata.

7) Cosa si intende per Network Address Translation (NAT)?

NETWORK ADDRESS TRANSLATION (NAT)

Una stazione con indirizzo privato può comunicare su Internet tramite NAT, Network Address Translation [traduzione di indirizzi], che una funzionalità che sta in un router di accesso tra la intranet pubblica e la Internet. Pur non essendo obbligatoria una tale posizione è comunque importante che stia sul percorso che i pacchetti faranno per andare verso il server pubblico che deve essere raggiunto.

Questa funzionalità modificherà gli indirizzi che si trovano nei pacchetti

Funzionamento

Dato un host A con indirizzo privato, esso vuole comunicare con un server B di indirizzo pubblico, quindi l'host A genera un pacchetto IP che vuole andare da A a B. Il pacchetto è inviato sulla intranet privata e attraversa la intranet privata e la intranet pubblica verso l'host con indirizzo B, con i vari router che indirizzano in modo opportuno il pacchetto. Sulla strada il pacchetto incontra la funzionalità di NAT che modifica gli indirizzi presenti nel pacchetto. In figura l'indirizzo mittente è modificato in X; l'indirizzo X, che può essere un pool di indirizzi pubblici, è un indirizzo associato al dispositivo che ha la funzionalità di NAT. Quando il pacchetto arriva a destinazione e B risponde, B risponderà mandando la risposta all'indirizzo X, al dispositivo con la funzionalità NAT che, ricordandosi dell'operazione di aver modificato l'indirizzo A in X, farà l'inverso, modificando il pacchetto che arriva da B con destinazione X mettendo destinazione A, inoltrando il pacchetto sulla intranet privata per cui il pacchetto arriverà ad A, completando la richiesta.

Il risultato è che A, con indirizzo privato, ha comunicato con B ad indirizzo pubblico.

8) A cosa serve il protocollo DHCP? Come funziona?

CONFIGURAZIONE DINAMICA DEGLI INDIRIZZI - DHCP: Dynamic Host Configuration Protocol

Ci sono anche altri metodi per la configurazione dinamica, di cui uno, in disuso, utilizzato per il boot dalla rete di stazioni senza disco fisso, è il metodo con richieste multiple, ovvero richieste con protocolli diversi; in questo non c'è la netmask

→ Richiesta RARP per ottenere un indirizzo IP
→ Messaggio del protocollo ICMP, detto Address Mask Request, a mezzo di un router sulla rete
→ Messaggio ICMP, Gateway Discovery
    → Eventualmente più risposte

DHCP: Dynamic Host Configuration Protocol

Un server DCHP sulla rete fisica su cui si trova la stazione che ha bisogno della configurazione. La stazione effettua una richiesta di configurazione a cui il server risponde. Il server ha un database di indirizzi IP, precisamente configurazioni complete da assegnare alle stazioni. Il server sceglie una stazione e la offre alla stazione, quindi passa alla stazione indirizzo, netmask, indirizzo del default gateway, indirizzo dei server DNS. Questo in un unico protocollo, con i suoi messaggi, progettati apposta per questo scopo, con la presenza di un server, sul quale si basa DHCP.

Le caratteristiche di DHCP

→ Imbustato in UDP
    → Porta 67, alla quale vengono mandati i messaggi UDP, quindi il server DHCP è in attesa sulla porta 67 usando l'UDP e questo vuol dire che i messaggi DHCP vengono imbustati dentro pacchetti IP, per cui il client deve specificare il proprio indirizzo per metterlo nel campo indirizzo mittente e l'indirizzo del server. Il client non conosce né il proprio indirizzo, né quella del server per cui. La soluzione è quella di mandare i messaggi in broadcast, sia a livello IP che a livello MAC; l'indirizzo IP di destinazione è un indirizzo IP broadcast, il pacchetto IP è imbustato in una trama MAC mandata in broadcast, il client usa come indirizzo sorgente 0.0.0.0 e per identificare il server, il client usa l'indirizzo broadcast locale 255.255.255.255
→ Messaggi in broadcast
    → A livello MAC e IP
        → Client usa 0.0.0.0
        → Il server usa 255.255.255.255

Questo pacchetto IP, che contiene un messaggio UDP, che contiene la richiesta DHCP viene mandato a questo indirizzo IP, messo in una trama MAC e mandato all'indirizzo MAC broadcast. Questo si propaga su tutta la rete locale e quindi arriva a tutte le stazioni della rete locale. Le varie stazioni scarteranno il pacchetto perchè le stazioni locali vedono un pacchetto mandato all'indirizzo broadcast locale e lo ignorano oppure ci guardano dentro e vedono che è un pacchetto UDP alla porta 67; non avendo, la stazione locale un processo in attesa di pacchetti sulla porta 67, il pacchetto sarà scartato.

Il server DHCP può invece rispondere proponendo una configurazione, così come altri server. Il client ne sceglie una e la richiede.

Negoziazione

→ Il server propone una configurazione IP
    → Ci potrebbero essere più server
    → Più proposte
→ Il client ne sceglie una e la richiede

Allocazione degli indirizzi che il server offre al client, ci sono tre tipi di allocazione

→ Allocazione dinamica, che va bene per la situazione descritta avanti
  → Lo stesso indirizzo IP è assegnato a stazioni diverse in diversi momenti, ad esempio negli hot spot di aeroporti, ecc., con stazioni che usano un indirizzo per un tempo relativamente breve e, quando la stazione non ne ha più bisogno, tale indirizzo è riassegnabile dal server ad un'altra stazione che farà richiesta
  → Una stazione può ricevere indirizzi diversi nel tempo
→ Allocazione automatica, valida in un ambito più controllato
  → Una stazione riceve sempre lo stesso indirizzo IP dal server DHCP, grazie all'indirizzo MAC, detto anche indirizzo hardware, della stazione
  → Non è noto/deciso in precedenza, il primo è scelto a caso dal server DHCP
→ Allocazione manuale, situazione diversa dalla configurazione manuale degli indirizzi; l'amministratore di rete deve indicare per ogni indirizzo MAC quale indirizzo IP usare; la configurazione avviene sul server DHCP ed è più semplice della configurazione manuale sulla stazione
  → Una stazione riceve sempre lo stesso indirizzo
  → Indirizzo assegnato manualmente dall'amministratore di rete

Principali campi del messaggio DHCP

→ op: op code/tipo di mess., tipo di operazione
  → 1 = BOOTREQUEST
  → 2 = BOOTREPLY
→ htype: HW type, tipo di indirizzo di livello 2, che sarà scritto nel messaggio
→ hlen: HW address len, lunghezza di indirizzo di livello 2
→ chaddr: client HW address
→ xid: Transaction ID, identificatore di transazione
→ yiaddr: indirizzo IP assegnato dal server al client

Opzioni, è il campo che contiene le varie informazioni che serviranno per la configurazione. La codifica è di tipo Code - Length - Value, con un primo campo di un byte che identifica il codice del campo opzionale, un campo lunghezza (che indica il numero di byte) ed un campo valore. Nel caso di un codice non conosciuto, l'opzione, che ha un valore che non può essere compreso, viene scartata sapendo che tale campo ha una lunghezza specificata. Questo rende il protocollo facilemente estensibile. Alcuni dei codici più comuni sono

→ Tipo di messaggio (codice 53)
→ Subnet mask (Codice 1)
→ Router (3)
→ Nome di dominio (15)
→ Server DNS (5)

In particolare, per il tipo di messaggio, con codice 53

Value	Message Type
1	DHCPDISCOVER
2	DHCPOFFER
3	DHCPREQUEST
4	DHCPDECLINE
5	DHCPACK

6 DHCPNAK
7 DHCPRELEASE
8 DHCPINFORM

Scambio di messaggi, avviene quando il client vuole una configurazione. Il cliente richiede una configurazione inviando un messaggio “DHCPdiscover”, quindi il campo 53 di valore 1. Questo messaggio serve per scoprire i server DHCP. Sulla rete locale ci sarà un server DHCP, con un suo indirizzo IP ed un suo indirizzo MAC. Il serve, alla ricezione del messaggio “DHCPdiscover” risponderà con un messaggio “DHCPoffer“, in cui è inclusa una configurazione che vuole offrire al client; questo messaggio del server andrà all’indirizzo 0.0.0.0 e sarà mandato in broadcast. Il client riceve l’offerta dal server, anche da altri server, fa una scelta e deve comunicare quale scelta ha fatto affinché quelli non scelti usino la configurazione offerta, ma rifiutata, ad altri client.

Il client a questo punto manda un messaggio “DHCPrequest” sempre dall’indirizzo 0.0.0.0 all’indirizzo broadcast 255.255.255.255 e questo perchè il client non può ancora usare la configurazione in quanto il server ha per ora solo fatto l’offerta e deve essere il server a confermare tale configurazione. Inoltre, mandando questo messaggio in broadcast, anche gli altri server sapranno quale scelta ha fatto il client. A questo punto il server manda un messaggio acknowledgment, il messaggio “DHCPack”, per cui il client mette sulla sua interfaccia la configurazione.

La configurazione fornita ha una durata limitata che si chiama lease (affitto).

Lease [affitto]

→ L’allocazione di indirizzo IP ha durata limitata
→ Il client può chiederne il rinnovo prima che scada, quindi manda di nuovo un messaggio DHCP Request
  → DHCP Request–DHCP Ack
→ Può esserne offerta una nuova, con un messaggio DHCP Offer dal server
  → DHCP Request–DHCP Offer

Rinnovo della lease

→ Se il rinnovo fallisce, si deve rifare l’allocazione
→ Per esempio, il server non risponde a DHCP Request
→ Da DHCP Discover in poi

Quando una stazione fa il reboot

→ Rinnovo lease (DHCP Request)
→ Nuova configurazione (DHCP Discover)

Limitazioni del DHCP

→ Client e server devono essere nella stessa rete fisica, perchè il client manda la richiesta in broadcast ed il server la può ricevere solo se è nella stessa rete fisica
→ Non praticabile in reti con tante sottoreti, cioè in reti molto grandi, con tante piccole sottoreti in quanto, reti con molti host hanno prestazioni che decadono

DHCP Relay, è un meccanismo, implementato nei router, che serve ad implementare la funzionalità DHCP.

→ Normalmente realizzato nei router
→ Il DHCP Relay inoltra messaggi DHCP Request, che sono inviati sulla rete locale, a un server DHCP remoto
  → L'indirizzo del server remoto è fornito manualmente, da opportuna configurazione
→ In questa richiesta il DHCP Relay include il proprio indirizzo IP. Specificatamente l'indirizzo IP che il router ha sull'interfaccia da cui ha ricevuto la richiesta DHCP. L'indirizzo del DHCP Relay sulla rete del client è incluso nel campo giaddr, gateway IP address, del messaggio DHCP
→ Indirizzo assegnato in base alla LIS del client, deducibile dal campo giaddr, che contiene il prefisso dell'indirizzo
  → Nel campo giaddr
→ A questo punto il server manda il messaggio DHCP Reply al DHCP Relay
→ Il DHCP Relay lo inoltra nella rete del client

DHCP e DDNS

Quando si usa DHCP risulta conveniente usare il DDNS, il DNS dinamico. Questo perchè l'host riceve un indirizzo dinamico per cui occorre che il suo nome venga associato all'indirizzo giusto che riceve ogni volta.

Nei vari sistemi operativi questo funziona in modo diverso, a lato un client DHCP in Windows 2000 e altri clients DHCP (Win9x e NT, in cui tutto è fatto dal server).

Si noti come siano aggiornati i record DNS di tipo A e di tipo PTR.

9) Descrivere e confrontare il funzionamento degli algoritmi Distance Vector e Link State

ROUTING DINAMICO

Basato su algoritmi adattativi, di tre categorie

→ Routing centralizzato

→ Routing isolato

→ Routing distribuito

  → Distance Vector

  → Link State

Routing centralizzato, vuol dire che da qualche parte della rete c'è un Routing Control Center che è un nodo di rete che calcola e distribuisce le tabelle di routing per tutti gli altri nodi della rete (si tratta di routing proattivo per preparare le tabelle che saranno usate nel forwarding dei pacchetti per fare il routing al volo). Tutte le tabelle vengono calcolate da un unico centro di calcolo

→ Routing Control Center (RCC)
→ Calcola e distribuisce le tabelle di routing
→ Ha bisogno di informazioni da tutti i nodi
→ Ottimizza le prestazioni
→ Semplifica troubleshooting, la ricerca guasti
→ Traffico di rete sostenuto nelle vicinanze dell'RCC
→ L'RCC è un singolo punto di fallimento del sistema
→ L'RCC è un collo di bottiglia
→ Non adatto per reti altamente dinamiche

Routing isolato, alternativa al routing centralizzato

→ Ogni nodo decide i percorsi indipendentemente
→ Non c'è scambio di informazioni con gli altri nodi; questa soluzione non genera traffico, non c'è point of failure, non c'è un collo di bottiglia nella rete
→ Per esempio, algoritmo Backward Learning
→ Usato dai bridge del protocollo IEEE 802.1D

Routing distribuito

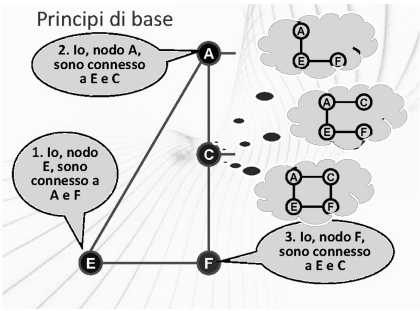
Unisce i vantaggi di routing isolato e centralizzato

→ Router collaborano nello scambiarsi informazioni sulla connettività nella rete, sui collegamenti, sul funzionamento dei router

→ Ogni router decide in modo indipendente, ma in modo coerente, il che non è banale perchè la decisione viene fatta in modo distribuito; a tale scopo sono stati progettati opportuni algoritmi per scambiare informazioni e permettere ai router di decidere in modo distribuito ma coerente

ALGORITMO DI ROUTING DISTANCE VECTOR

I nodi imparano per mezzo dei distance vector, che sono elenchi di distanze, quali sono i nodi raggiungibili attraverso le loro interfacce e a che distanza.



Principi di base, come a lato riportato

Distance Vector

→ Lista di destinazioni raggiungibili (tutte!), generata da ogni nodo; in una rete grande questa lista sarà enorme
→ La lista contiene la distanza dal router che manda l'annuncio, secondo una misura detta metrica
→ Ogni router genera il suo distance vector
→ Ogni router manda il suo distance vector a tutti i vicini

Scenario esemplificativo, in cui A riceve i distance vector dai nodi B e D; con queste informazioni, A fa una operazione di fusione e genera il proprio distance vector.

La fusione permette ad A di generare la routing table.

Fusione e generazione dei distance vector

La prima riga si legge come A avente l'informazione che può raggiungere A a distanza 0, oppure a distanza 1, ricevuta da B e D; quindi nella routing table ci viene scritta l'informazione migliore e viene generata l'informazione di distance vector per questa informazione migliore.

In seconda riga si hanno due informazioni di quale sia la distanza per raggiungere B e viene presa l'informazione migliore, notare come sia modificata la distanza in quanto A deve mandare i suoi pacchetti a B. Nel caso di E, A sceglie una informazione a caso, essendo entrambe valide.

Analogamente per le altre righe, per le quali A guarda tutte le destinazioni annunciate nei distance vector e fa l'operazione di scegliere il distance vector del vicino con un costo minore. Questa è l'ope-

razione di fusione dei distance vector. Da questa operazione nasce la tabella di routing, dalla quale nasce la distance vector di A, prendendo le destinazioni e le destinazioni alle quali A le sa raggiungere. La distance vector viene mandata a tutti i router vicini, che a loro volta faranno la fusione. Quando, dalla operazione di fusione viene generata un tabella di routing uguale a quella esistente allora il router capisce che può smettere di inviare distance vector poichè è stata determinata la topologia della rete.

Il router, in sostanza, manda il suo distance vector, solo quando esso cambia.

Cambiamento topologico, ad esempio si rompe il link tra A e B.

Quindi tutto quello che A aveva appreso dal link B è inservibile, quindi A deve fare nuovamente l'operazione di fusione (merge), per ricostruire la tabella di routing andando a prendere l'informazione migliore che è soltanto quella di D. Poi A genera il distance vector che sarà ricevuto dai nodi che, a loro volta faranno le operazione viste.

Questo mostra l'aspetto dinamico, adattativo, dell'algoritmo.

Il cambiamento topologico causa parecchi problemi

→ Black Hole [buco nero]
→ Count to infinity [conteggio a infinito]
→ Rimbalzi (loop)

Sono problemi di instabilità causati da cambiamenti topologici, per i quali i router cominciano a fare pasticci e non raggiungono una configurazione stabile delle loro tabelle di routing e a capire come inoltrare i pacchetti sulla rete.

Esistono soluzioni, che sono solo parziali

→ Split Horizon
→ Path Hold Down
→ Route Poisoning

In sostanza

Il problema di base è che i router non conoscono la topologia della rete. Ad esempio, in base ai distance vector che B riceve, B non può distinguere i casi, cioè B non distingue se la rete è fatta in un modo o nell'altro.

Vantaggi del distance vector

→ Facile da implementare
→ I protocolli sono facili da utilizzare
→ Richiedono minima configurazione

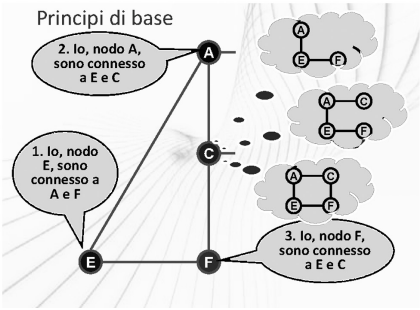
Limitazioni

→ Complessità e tempo di convergenza esponenziali (nel caso peggiore)
→ Da O(n2) a O(n3), con n il numero di nodi
→ I router e collegamenti più lenti determinano il tempo di convergenza di tutti i router in rete
→ Ottimizzazione complicata
→ Ricerca guasti complicata
→ Molto traffico di routing (e dati memorizzati)

Il distance vector non è adatto a reti grandi e complesse.

L'alternativa al distance vector sono gli algoritmi di routing link state.

ALGORITMI DI ROUTING LINK STATE



Principi di base , come a lato riportato, ovvero il nodo E manda in giro una informazione del tipo "io, nodo E, sono connesso a A e F", cioè E dice quale è la topologia intorno a lui, ovvero lo stato dei suoi link (link state). Questa informazione viene mandata a tutti i router della rete, compreso C. I nodi che ricevono tale informazione "capiscono" la topologia della rete. Quindi i router si costruiscono una mappa della rete.

Link state

→ Vengono create informazioni sullo stato dei collegamenti (link), dette Link state

→ Le informazioni devono essere mandate da ogni nodo a tutti gli altri nodi, ovvero viene fatta una operazione di Selective flooding [inondo selettivo], che è un punto critico, di non facile realizzazione

→ Ogni nodo si crea una mappa della rete

→ La stessa mappa su ogni nodo (importante!!!)

→ Ogni nodo calcola le "route" sulla mappa, per mezzo di un algoritmo, detto di Dijkstra

→ Algoritmo di Dijkstra, un matematico, (shortest path first), che cerca il percorso più breve

Perchè tutto funzioni, i link state devono arrivare a tutti i router, affinchè la mappa che i router creano sia quella corretta. Se i link state arrivano a tutti i router, essi possono creare percorsi ottimali e coerenti.

Link state database, che ogni nodo crea.

Ad esempio quello di A dice che A è collegato a B e a D, con un link di collegamento, quindi in questo caso la metrica è il numero di link, ma può essere la distanza o altro, la capacità dei link.

Il router genera la sua tabella di routing.

Convergenza rapida

→ L'algoritmo di Dijkstra ha una bassa complessità di calcolo

→ L • log (N)

→ L: numero di link

→ N: numero di nodi

→ I link state si propagano velocemente

→ I link state non richiedono nessuna elaborazione prima dell'inoltro

L'algoritmo lavora anche su reti grandi.

Traffico di routing e uso di memoria limitati

→ I link state sono piccoli
→ Neighbor greeting veloce ed efficiente
    → Protocollo per scoprire i vicini

Altri vantaggi

→ Raramente genera loop
    → Percorsi di inoltro circolari
→ Semplice da capire e per fare ricerca guasti
    → Tutti i nodi hanno basi dati identiche

Limitazioni

→ Alta complessità di implementazione
    → Selective flooding
    → Prima implementazione ha richiesto diversi anni, 5
→ Protocolli con complessa configurazione

10) Cosa si intende per Autonomous System?

AUTONOMOUS SYSTEM [SISTEMI AUTONOMI]

Di cosa si tratta

→ Un insieme di sottoreti IP (vicine) raggruppate per
    → Topologia e per criteri organizzativi; amministrate da una stessa amministrazione
→ Per esempio le sottoreti di un grosso internet service provider

Quindi si dice che un insieme di sottoreti del service provider X sono un autonomous system

Perché si fa questo?

→ Gestione indirizzi e routing strettamente coordinati
    → Eventualmente ci saranno più domini di routing
→ Interfacce tra autonomous system sono tenute sotto controllo
    → Dati, non tutti i dati non sono scambiati
    → Informazioni di routing, non sono tutte scambiate

Cioè un router di un autonomous system non dice necessariamente tutto ad un router di un altro autonomous system, questo perchè possono benissimo appartenere ad organizzazioni diverse. Dunque il flusso di dati ed il flusso di informazioni di routing sarà controllato.

Gli AS si creano quindi per ragione di amministrazione, in modo che all'interno dell'autonomous system le scelte di routing possano essere autonome (il service provider deciderà di usare i protocolli che vuole nel modo in cui vuole e con la configurazione che vuole). Le scelte del routing esterno, cioè tra un autonomous system ed un altro dovranno essere negoziate.

Gli autonomous system si creano per ragioni di scalability, affinchè la rete possa crescere e possa funzionare come rete di grande dimensioni, questo perchè le informazioni di routing non sono propagate ovunque, cioè la frontiera tra un autonomous system ed un altro è il posto giusto dove ridurre la quantità di informazione di routing che viene propagata. Si riduce filtrando le informazioni oppure aggregando delle informazioni, prendendo un certo numero di annunci diversi ed aggregandoli in un

annuncio unico come se esso ne fosse il riassunto. La perdita di informazione bilancia la minor propagazione dell'informazione.

Da questo si può notare che pur essendo la rete Internet enorme, tra i vari autonomous system vengono propagate informazioni di routing con tutti i dettagli, ma tra autonomous system si eliminano dettagli per cui i router non vengono sopraffatti da informazioni.

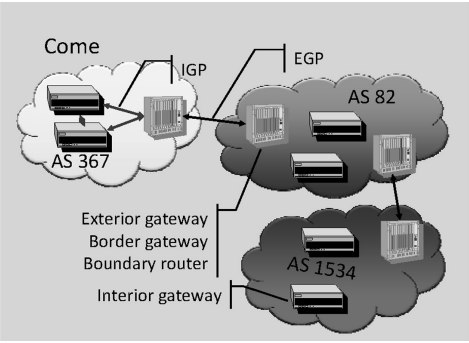
→ Amministrazione

→ Scelte sul routing interno autonome

→ Scelte sul routing esterno negoziate

→Scalability

→ Informazioni non sono tutte propagate ovunque



Si deve in qualche modo riuscire a controllare lo scambio di informazioni tra autonomous system, rendendo ogni autonomous system autonomo in qualche modo.

I router interni agli autonomous system sono detti Interior gateway, questi sono router collegati solo ad altri router interni all'autonomous system e scambiano informazioni tra di loro con dei protocolli detti IGP, Interior Gateway Protocol.

I router collegati a router di un autonomous system diverso (gestiti quindi in modo diverso dal punto di vista amministrativo) sono detti Exterior gateway, Border gateway o Boundry router. Questi scambiano informazioni di routing con protocolli che chiamiamo EGP, Exterior Gateway Protocol. La famiglia di protocolli EGP avrà caratteristiche diverse dalle caratteristiche della famiglia di protocolli IGP.

Gli autonomous system sono identificati da un numero, come si nota in figura.

L'identificazione di un autonomous system

→ Tramite un numero di 2 o 4 byte

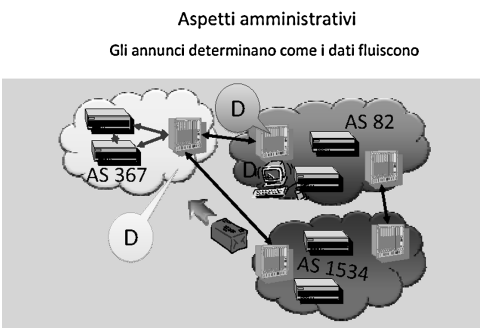
→ Assegnato da IANA (Internet Assigned Numbers Authority)

→ Numeri privati di autonomous system

→ 64512-65534 (originali, su due byte)

→ Per scambi controllati di informazioni di routing

I numeri privati degli autonomous system sono analoghi agli indirizzi privati. Chunque può usare tali numeri di autonomous system senza dover chiedere una autorizzazione allo IANA, ma non ci sarà garanzia che siano univoci. Si usano in contesti di reti private, per creare zone in cui lo scambio di informazioni di routing sia controllato.



Aspetti amministrativi

Gli autonomous system sono importanti anche per aspetti di tipo amministrativo tipo quello di decidere che tipo di percorso faranno i pacchetti nella rete tra autonomous system.

In figura a lato è supposto che nell'AS85 ci sia una destinazione D. In una rete i router troveranno la strada migliore per inoltrare le informazioni nella stessa rete; in uno scenario di più autonomous system è importante il modo in cui i dati fluiscono, dipendente dagli annunci.

Gli annunci che vengono generati in un certo modo hanno un impatto sui percorsi, quindi annunci fatti in un certo modo permetteranno il percorso migliore.

Exterior Routing, routing tra autonomous system

→ Routing non necessariamente sui percorsi più brevi

→ Le scelte sono basate su politiche, politiche configurabili che riflettono gli accordi tra i gestori degli autonomous system

Il router cerca sempre il percorso più breve, il percorso migliore secondo la metrica del particolare protocollo. Nel caso di protocolli da usare per l'exterior routing, routing tra autonomous system, invece la scelta deve essere basata su delle politiche configurabili dall'amministratore della rete, quindi il protocollo di routing e la sua implementazione devono prevedere questo.

Gli autonomous system si fanno anche per ragioni di scalability, per poter operare su grosse reti.

Per ottenere scalability con l'uso di autonomous system si cerca di aggregare le destinazioni

Scalability [capacità di operare su grosse reti]

→ Le destinazioni possono essere aggregate

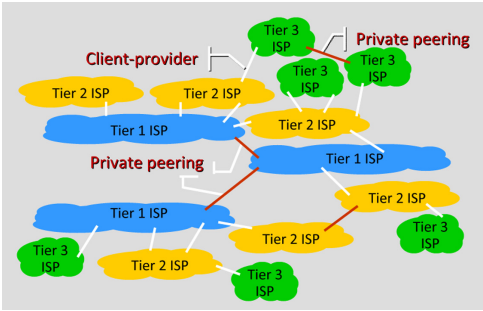
→ 195.1.2.0/24 e 195.1.3.0/24 annunciate come 195.1.2.0/23 dal border gateway, cioè il router di bordo (router che sta al bordo). Le due destinazioni vengono annunciate in modo aggregato e quindi esso annuncia una sola destinazione. Nell'annuncio il /23 indica il prefisso di 23 bit che le annuncia tutte e due per cui viene usato il supernetting

I router esterni all'autonomous system vedranno un solo annuncio e quindi dovranno elaborare, memorizzare, propagare un solo annuncio e faranno meno lavoro, perdendo un pò di informazione, come ad esempi quanto possono essere lontane le due destinazioni.

Questo riprende il concetto iniziale per cui vogliamo avere gli autonomous system per far operare i router in una rete gigantesca riducendo la quantità di informazioni e questo consiste nel creare i pnti di aggregazione all'uscita dagli autonomous system ed il confine, il perimetro, dell'autonomous system ci fornisce il punto ideale dove fare questa operazione.

11) Illustrare l'architettura di routing in Internet

ARCHITETTURA DI ROUTING IN INTERNET



Come viene organizzato il routing nella rete Internet, che è tutta organizzata in Internet Service Provider. Gli ISP non sono tutti uguali, alcuni hanno reti molto grandi che raggiungono diversi continenti e sono interconnessi tra di loro per mezzo di border gateway. Questi ISP (Tier 1 ISP in celeste, Internet Service Provider di Livello 1) scambiano informazioni di routing e scambiano traffico dati; all'interno avranno i propri router che useranno protocollo IGP per capire come raggiungere ogni destinazione. A questi ISP si collegheranno dei Service Provider più piccoli (Tier 2 ISP in giallo, Internet Service Provider di Livello 2), che sono collegati a quelli di livello 1 tramite router boorder gateway ed anche ogni ISP di livello 2 costituisce un autonomous system ed usa un protocollo IGP per scambiare informazioni di routing e di dati. Se un ISP di Tier 2 vuole mandare un pacchetto ad un altro ISP di Tier 2, il pacchetto passerà da un ISP di Tier 1, per cui ci sarà una operazione di transito attraverso un service provider di tier 1. I service provider di tier 2 possono essere collegati a più service provider di tier 1. Poi abbiamo anche Service Provider di Livello 3, cioè Tier 3 ISP (in verde), che sono collegati a quelli di livello 2. Gli ISP Tier 3 possono avere anche più collegamenti con gli ISP Tier 2, per ragioni di load balance e di affidabilità. Si noti come i collegamenti possano essere di tipo diverso: quelli di colore rosso sono i collegamenti detti di private peering, perchè collegano Service Provider dello stesso livello e di norma questi Service Provider hanno un interesse mutuo ad essere collegati tra di loro in quanto vogliono che i propri clienti possano mandare pacchetti ai clienti dell'altro Service Provider. Entrambi hanno una rete molto estesa che dà raggiungibilità su grosse aree geografiche e vogliono essere collegati. Il collegamento di private peering è sicuramente fra i Service Provider di livelli 1, ma può anche essere fra quelli di livello 2 e anche fra quelli di livello 3. Questo può avvenire per non far fare ai pacchetti strade molto lunghe, in quanto la copertura di service provider, ad esempio quelli di livello 3, può essere molto vicina, nella stessa città. La ragione del peering è quello di aver interesse mutuo nel far passare traffico che, in quest'ultimo caso sarà più veloce e gli utenti vedranno prestazioni migliori.

I collegamenti di colore bianco sono collegamenti di tipo Client-provider, in cui un provider compra un servizio dall'altro. Ad esempio un Service provider di livello 2 può comprare un servizio (in pratica chiedere un collegamento ed il servizio comprato è quello di connettività, oppure detto di transito) a quello di livello 1 in quanto quest'ultimo ha pacchetti che possono andare negli Stati Uniti, od un altro per raggiungere la zona Asia-Pacifica.

I router prenderanno decisioni che riflettono gli accordi. I router di bordo, usando i protocolli di routing per i router esterni (External Gateway Protocol) devono essere in grado di implementare gli accordi commerciali tra i Service Provider.

Saranno quindi regolati i vari annunci che i router di bordo mandano verso l'altro router di bordo e, ad esempio, l'ISP Tier 1 manderà annunci solo per destinazioni americane all'ISP Tier 2 con cui ha preso accordi in tale senso. Queste sono quindi politiche di routing e riguardano quali annunci mandare e quali non mandare, ed in quale direzione. Il routing tra anonimous system è tutto basato su questi criteri. E tutto questo vale anche per collegamenti di tipo private peering, ad esempio fra service provider di livello 3, in alto a destra nella figura. Questo si traduce nel non utilizzare un collegamento per far passare traffico oltre quello stabilito dall'accordo. Quindi anche sui collegamenti di peering si devono implementare le politiche di routing che riflettono gli accordi commerciali.

I collegamentii sono detti Private peering perchè i service provider per avere quel collegamento devo-

no avere un collegamento tra loro due router, che non saranno necessariamente vicini, per cui sarà necessario acquistare, o affittare, l'uso di un mezzo trasmissivo (fibra ottica, ad esempio) , quindi questo sarà un collegamento privato, tra due service provider.

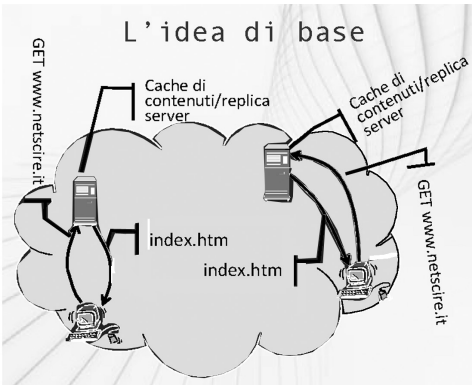
L'alternativa al dover creare dei collegamenti dedicati, cioè privati, tra due router è quello di creare i cosiddetti NAP o IXP, Neutral Access Point (NAP), Internet eXchange Point (IXP).

Si tratta di un locale a cui i service provider possono collegarsi per mezzo, ad esempio, di una fibra che va dal loro router in un loro locale a questo locale che si può dire essere pubblico, gestito da un gestore di terza parte. I service provider metteranno nel locale del NAP/IXP un loro router. Avendo nel NAP/IPX un punto di accentramento, i service provider possono creare un loro collegamento di peering (tratto rosso tratteggiato), per cui possono scambiare dati ed informazioni di routing non attraverso un link dedicato a loro due, ma attraverso il punto pubblico, pubblico perchè ci possono essere altri service provider che hanno un collegamento verso il NAP/IXP. Questo elimina la necessità di collegamenti privati dedicati, così come collegamenti client-provider e questo aumenta la connettività.

Neutral Access Point (NAP), Internet eXchange Point (IXP) è una LAN che collega router di vari AS (ISP), con coppie di router scambiano informazioni di routing, eventualmente usando BGP.

Esercizi relativi alle lezioni 25, 26, 27, 28, 29, 30

1) Cosa si intende per Content Delivery Network?



CONTENT DELIVERY NETWORK

E' un primo tipo di consegna particolare, l'altro è il Multicasting IP.

Si applica a reti molto usate, per fornire servizi web, quindi per assicurarsi che l'accesso al web da parte degli utenti sia veloce ed abbia buone prestazioni, ma soprattutto per servizi tipo il video on demand.

L'idea di base è che, dovendo portare contenuti all'utente, saranno create copie dei contenuti vicino all'utente, con server che hanno delle cache, ovvero copie temporanee dei contenuti, oppure con repliche dei contenuti. Il client scaricherà i contenuti da una replica locale (il server prende il nome di "replica server"). Le aziende (ad esempio CAMAI) quindi installano in giro per il mondo repliche di server sulle quali vengono fatte copie di contenuti di qualsiasi genere a cui l'utente può accedere senza accedere al server originale, con prestazioni migliore, cioè con maggiore velocità. L'azienda vende questo come un servizio.

Quando il client effettua una richiesta (GET www.netscire.it) esso deve accedere al server a lui più vicino, con i server che hanno indirizzi diversi e si troveranno in posti diversi. Gli approcci sono diversi, alcuni non standard. Uno è il DNS, oppure riscrivere le URL, oppure usare anycast.

Come funziona?

→ DNS, che fornisce un indirizzo di una replica locale

→ Riscrittura delle URL, che funziona per le pagine web

→ Anycast, un tipo di consegna pacchetti particolare per cui i pacchetti non vengono mandati all'indirizzo specifico di una stazione ma ad un indirizzo anycast e poi la rete, per mezzo dei router, si occupa di fare la consegna; l'indirizzo corrisponde ad un gruppo di server ed i router si fanno carico di consegnare il pacchetto al server che appartiene a quel gruppo e che è più vicino al mittente. L'anycast non è molto usato.

2) Cos'è e a cosa serve un Interior Gateway Protocol? E un Exterior Gateway Protocol?

DUE FAMIGLIE DI PROTOCOLLI nella rete Internet

Due tipi di protocolli

→ Interior Gateway Protocol (IGP)

→ Usato per routing intra-dominio, cioè all'interno di un autonomous system, da non confondersi col fatto che un protocollo di routing funziona sempre nel dominio di routing viene utilizzato, per definizione

→ Exterior Gateway Protocol (EGP)

→ Inter-domain routing, cioè usato nel routing tra domini

Ci sono due famiglie di protocolli perchè essi hanno obiettivi diversi.

Obiettivi diversi → Diversi criteri di progettazione

Caratteristiche degli IGP

→ Obiettivo è distribuire informazioni sulla topologia di rete

→ Scegliere route [percorsi per l'inoltro di pacchetti] in base a tali informazioni topologiche. Nei protocolli IGP il router cerca di trovare la route "migliore", che dipende dalla definizione che vogliamo dare. I vari protocolli definiscono delle metriche, ovvero dei modi per misurare i percorsi ed un criterio per dire quale percorso è migliore dell'altro. Il router userà il protocollo per raccogliere informazioni e poi per calcolare i percorsi migliori per tutte le destinazioni in base a queste informazioni. Una volta calcolati i percorsi il router può costruirsi la tabella di routing che dice quale è il next hop a cui mandare i pacchetti per una certa destinazione

Caratteristiche dei protocolli della famiglia EGP

→ Servono per distribuire informazioni su Autonomous System

→ Servono a distribuire costi amministrativi, cioè dei costi che rappresentano le scelte degli amministratori di rete su quali siano i percorsi migliori o preferibili per il traffico, tra gli anonimous system

→ Questo permette ai router di decidere in base a politiche configurate dagli amministratori. L'obiettivo dei router è dunque trovare la route "preferita" (non la "migliore"), in base alle indicazioni di chi configura il router, in base agli accordi fatti con i gestori degli altri autonomous system

Vediamo dunque quali sono oggi i protocolli di routing di queste due famiglie di protocolli, iniziando dall'IGP, dai protocolli che usano l'algoritmo del distance vector e poi da quelli che utilizzano l'algoritmo link state e poi passando a quelli della famiglia EGP.

IGP – Distance Vector

→ RIP: Routing Information Protocol

→ IGRP: Interior Gateway Routing Protocol

  → E-IGRP: Enhanced IGRP, versione successiva

IGP – Link State

→ OSPF: Open Shortest Path First, più famoso

→ Integrated IS-IS, molto usato

EGP

→ BGP: Border Gateway Protocol

→ IDRP: Inter Domain Routing Protocol, utilizzato pochissimo

→ Il routing statico è anche una possibile opzione, nel routing inter-dominio, per cui è l'amministratore che definisce il percorso che i pacchetti faranno. Questo per la ragione di scelta del percorso "preferibile" e non "migliore". Per ha il problema che non è adattativo, cioè non si adatta a cambiamenti topologici

INTERIOR GATEWAY PROTOCOL

RIP

→ Originariamente sviluppato per un'architettura diversa

→ RFC 1058 (1988), che ha adattato il protocollo all'uso di Internet e RFC 1388 (1993), che fornisce una prima nuova versione. E' basato sull'algoritmo distance vector

→ Implementato anche in stazioni Unix/Linux, per apprendere gateway; una stazione Unix è funzionante anche come gateway

Caratteristiche

→ Hop count [numero di hop]

→ Al più 15 hop, per non entrare in modalità count to infinity

→ Messaggi di aggiornamento sono periodici

→ Distance vector

→ Ogni 30 s, questo vuol dire che un router manda, ogni 30 secondi, l'elenco di tutte le destinazioni che sa raggiungere, che è molto traffico; questo serve a dimostrare ai vicini che è attivo; questa soluzione è inefficienza

→ Convergenza: in 3 min, nei casi peggiori; è il tempo che ci mette la rete a riconfigurarsi, per cui in tale tempo ci potrebbero essere stazioni non raggiungibili, il che è problematico, ma meglio di una rete rotta

IGRP, protocollo molto usato

→ Protocollo proprietario della Cisco Systems, sempre basato sul distance vector

→ Supera alcuni dei limiti del RIP

→ Per un po' era l'unica alternativa a RIP

Caratteristiche

→ Metriche articolate

→ Ritardo sui link

→ Banda dei link

→ Affidabilità dei link

→ Carico dei link

→ Massima lunghezza dei pacchetti, che i link lasciano trasmettere, detta MTU, Maximum Transmission Unit

→ Multipath routing, mentre di norma i router IP scelgono un percorso e mandano tutti i pacchetti su quel percorso. In questo caso, se ci sono due o più percorsi alternativi, il protocollo dice di usarli tutti, in modo inversamente proporzionale al loro costo, che è calcolato sul peso dei pacchetti. Il carico è dunque diviso sui percorsi

Il percorso migliore è scelto in base ad una combinazione pesata delle informazioni sopra. Il protocollo ha una configurazione di default, ma l'amministratore può specificare quale metrica è più importante, ad esempio tra basso ritardo ed alta affidabilità

EXTERIOR GATEWAY PROTOCOL

BGP, in sostanza l'unico protocollo usato nella famiglia di protocolli Exterior Gateway Protocol (EGP)

→ Attualmente alla versione 4

→ RFC 1654 (1994)

→ Path vector

→ Sequenza di AS fino alla destinazione, protocollo di tipo path vector; la sequenza dice come raggiungere la destinazione per attraversamento di AS

→ Politica di scelta delle route configurabile

InterDomain Routing Protocol (IDRP)

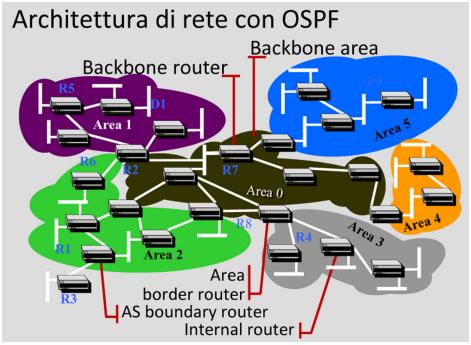
→ Evoluzione di BGP per OSI

→ Adattato a TCP/IP

→ Sarebbe dovuto essere "la" scelta di EGP per IPv6

→ Non molto usato, causa nuove versioni di BGP

3) Descrivere un'architettura di rete basata su OSPF



OSPF, basato sull'algoritmo link state, più complicato da implementare, in particolare la distribuzione dei link state. E' stato utilizzato dopo a tutti gli altri protocolli. E' un protocollo molto ben fatto e molto potente, più difficile da configurare

→ RFC 1247 (1991) e RFC 1583 (1994)

→ Routing gerarchico

→ Il dominio di routing è diviso in aree

→ Aggregazione di informazione tra aree

I router di un'area avranno tutte le informazioni, quelli esterni avranno informazioni meno dettagliate, in quanto sono state aggregate, ma in questo modo i router non si trovano ad avere a che fare con troppe informazioni.

L'area 0, o backbone area, è un'area particolare, che deve collegare tutte le altre come da figura a lato, architettura di rete con OSPF. In questo protocollo bisogna dire ai router quali sono le aree; i router possono avere funzionalità particolari: quelli interni alle aree si chiamano internal router, quelli interni alla backbone area si chiamano backbone router, che sono "normali", poi ci sono gli area border router che sono configurati ad essere parti di più aree che fanno una funzione particolare, quella di vedere tutti i dettagli delle aree per cui sono configurati (e collegati), poi ne fanno un riassunto e lo propagano nelle varie aree. Ci sono dunque dei link state particolari che servono per riassumere ciò che c'è in un'area e dirlo in un'altra area. Questo riassunto fa in modo che i router non debbano avere a che fare con una mappa dettagliata della rete: essi hanno una mappa dettagliata della loro area con informazioni sull'esterno. Questo rende il protocollo gerarchico e fa in modo che funzioni su reti molto grandi, cioè fornisce al protocollo una grossa scalability.

4) Descrivere e commentare le differenze tra la crittografia a chiave segreta e quella a chiave pubblica

CRITTOGRAFIA

→ Offre soluzioni per tutti i casi sopraelencati

→ Letteralmente: scritto nascosto

→ Ha a che fare con tecniche e protocolli per proteggere le informazioni e per rendere sicuro lo scambio di informazioni

Criptazione [encryption], su cui è basata la crittografia

→ L'informazione è rappresentata da un codice

→ La versione codificata non svela l'informazione

→ Sono necessari specifici tecniche e parametri (segreti, più che altro i parametri, che vedremo saranno le cosiddette chiavi) sono necessari per rivelarla

Il funzionamento di base, come rappresentato in figura a lato, è che, data l'informazione, ad esempio un testo in chiaro, si applica un algoritmo normalmente noto che usa un parametro (chiave) che è normalmente segreto. Quello che si ottiene è un messaggio criptato, cioè codificato in un modo che non è intellegibile.

Per rendere il sistema di criptazione più robusto devono essere usate chiavi lunghe, in termini di sequenza di bit.

Quando si è ottenuto un messaggio criptato, ci sarà la possibilità di decriptarlo, applicando un algoritmo di decriptazione, che usa la stessa chiave (che deve rimanere segreta tra chi vuole scambiare tali dati), che permetterà di ottenere il messaggio originale in chiaro.

Decriptazione [decryption]

Chiave condivisa/segreta

→ La chiave deve essere condivisa in modo sicuro

→ Per esempio off-line

→ Richiede una relazione preesistente

→ Se la chiave è compromessa da uno dei partecipanti, nessuno la può più usare

Questo tipo di criptazione che usa la stessa chiave per criptare e per decriptare è detta crittografia a chiave simmetrica, perchè si usa la stessa chiave; è detta anche a chiave segreta perchè essa deve rimanere segreta e deve essere conosciuta solo tra le persone, o le stazioni, che devono scambiare informazioni.

Crittografia a chiave asimmetrica

Nel voler trasferire un messaggio in modo sicuro con questa tecnica, cifriamo il messaggio con un algoritmo e una chiave ottenendo un messaggio cifrato che può essere trasferito attraverso la rete. Il ricevente userà un algoritmo di decriptazione e una chiave diversa da quella usata per criptare il messaggio; il fatto di usare chiavi diverse sta alla base del nome crittografia a chiave asimmetrica. Un algoritmo molto famoso è l' RSA (chiave di 2048 bit).

Questo tipo di crittografia è detto anche a chiave pubblica, in quanto una delle due chiavi può essere

distribuita pubblicamente e usata per criptare, questa è la chiave pubblica. C'è poi la seconda chiave, detta chiave privata, che serve per decriptare (ad esempio la smart card è un piccolo processore con un po di memoria che contiene una chiave privata ed il processore è in grado di eseguire operazioni di crittografia asimmetrica).

Crittografia a chiave pubblica

→ Una delle due chiavi può essere distribuita pubblicamente

→ Usata per criptare, essa è la chiave pubblica

→ Solo chi ha la chiave privata corrispondente alla chiave pubblica può decifrare il messaggio

→ La chiave privata non deve mai essere condivisa

→ Più facile tenerla al sicuro

→ Le chiavi sono complementari

→ Si può fare un deposito (pubblico, un repository) per le chiavi pubbliche

→ Associate ai loro possessori

5) Definire i concetti di segretezza, integrità, autenticazione e non ripudiabilità di un messaggio

SFIDE NELLA SICUREZZA DELLE INFORMAZIONI

La prima riguarda la segretezza, o privacy; la seconda è l'integrità; la terza è l'autenticazione e l'ultima è la ripudiabilità.

Segretezza/privacy

Un osservatore non può accedere all'informazione

→ Per esempio qualcuno che intercetti i pacchetti mentre transitano nella rete

→ Particolarmente semplice violare la privacy in comunicazioni wireless perchè consiste nel sintonizzarsi alle giuste frequenze

Integrità

Assicurarsi che i dati non siano stati manipolati

→ Per esempio pacchetti modificati (sia intestazione, sia contenuto) mentre transitano attraverso la rete

Autenticazione

L'autore è quello che ci si aspetta

→ Un utente o un sistema invia dati facendo finta di essere un altro

→ In alcuni casi l'autenticazione include anche integrità

Non ripudiabilità

L'autore non può negarlo

→ Dopo aver fatto un'operazione un utente possa dichiarare che sia stato qualcun altro

→ Per esempio firma on-line di un contratto

6) Cos'è un certificato digitale?

CERTIFICATI DIGITALI

Detti anche certificati a chiave pubblica, perchè basati sulla crittografia a chiave pubblica.

Di cosa si tratta? Una chiave con una etichetta, firmate.

Quindi il certificato digitale contiene sostanzialmente una chiave, il possessore della chiave, una firma che garantisce l'integrità del tutto.

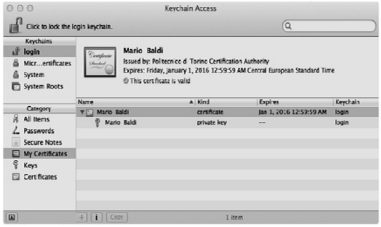
Si tratta di verificare chi firma i certificati digitali: essi sono firmati da una autorità di certificazione, CA.

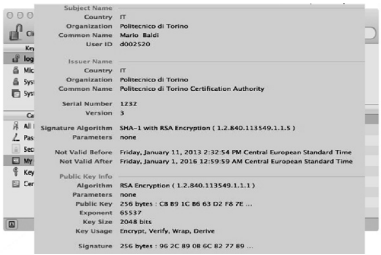
Certification Authority [autorità di certificazione]

→ La CA verifica l'identità del possessore della chiave prima di firmare

→ Per esempio il possessore della chiave deve presentarsi di persona con un documento; questa è una operazione che deve essere fatta off-line e non può essere fatta on-line.

→ Il certificato digitale è utilizzabile per assicurare il non ripudio delle informazioni. Questo perchè quando qualcuno firma le informazioni con la chiave che è nel certificato, l'identità legale è confermata in quanto la CA ne ha effettuato la verifica in modo fisico.





A lato un esempio di certificato, con dettagli vari.

La firma è sotto, in fondo, di 256 byte. La firma garantisce che nessuna delle informazioni riportate sopra può essere cambiata.

Se viene cambiata una sola informazione, allora la firma non è più verificabile.

Si noti l'algoritmo usato per la firma, algoritmo usato dalla CA del Politecnico di Torino per fare la firma. C'è scritto chi ha fatto la firma, la validità e una chiave pubblica , su 256 byte, che si certifica essere posseduta da Mario Baldi.

La chiave pubblica può essere usata con un particolare algoritmo di decriptazione, che è indicato.

Il punto chiave è che è legata una identità ad una chiave pubblica, per mezzo di una Certification Authority che ha firmato il tutto.

Il certificato deve anche poter essere certificato. Si dovrebbe prendere la chiave pubblica della CA e verificare il certificato. Ma il problema è che non c'è una sola CA, per cui quando sono stati definiti i certificati digitali è stato definito anche il Public Key Infrastructure, che risolve il problema del non poter avere una sola CA, così come lo sarebbe se ci fossero più CA indipendenti, con la difficoltà che deriva tra il confrontare i certificati. La soluzione è che ci possono essere più CA, ma esse sono organizzate in una gerarchia di Ca.

PKI: Public Key Infrastructure

→ Sarebbe irrealistico avere

→ Una singola CA

→ Più CA indipendenti

Gerarchia di certification authority

Il che porta al problema di come verificare un certificato.

Verifica dei certificati

Occorre verificare la firma della CA. Quindi occorre avere la chiave pubblica della CA, per cui c'è bisogno del certificato della CA. Tale certificato sarà firmato da un'altra CA, e quindi si va a recuperare il certificato di quest'ultima Ca, e così via lungo la gerarchia, fino al livello più alto che è la Root Certification Authority ed essa firma i certificati da sola. Poichè questa non è verificabile allora quando si richiede un certificato si ottiene, in modo sicuro, anche per la propria chiave pubblica, il certificato della Root CA. Questo ci permette di verificare i certificati di tantissime CA che appartengono alla stessa gerarchia.

Rilascio dei certificati

→ La CA verifica l'identità legale del possessore

→ Eventualmente attraverso una Registration Authority

→ La CA crea la coppia di chiavi

→ La CA firma il certificato, che valida la chiave pubblica

→ Il certificato con la chiave pubblica viene messo in un deposito [repository]

→ La chiave privata è consegnata al possessore

→ Il certificato della root CA è dato al possessore del certificato

Tutto il mondo si fida di una stessa Root CA?

→ Sfortunatamente no

→ Ci sono diverse Root CA

Come otteniamo i loro certificati? Sono Inclusi nei sistemi operativi e browser → Ci fidiamo tutti dei produttori di software!?! (forse solo perché non lo sappiamo)

7) Quali sono le tipologie di attacchi informatici più diffuse?

TIPI DI ATTACCO

Snooping [spiare]

→ Sui collegamenti

→ Wireless

→ Sonde, su collegamenti fisici

→ Può essere fatto anche nei nodi

→ Il traffico può essere dirottato verso un punto di osservazione

Perturbazione del servizio

→Perturbazione del routing

→ Messaggi di routing fasulli immessi nella rete

→Informazioni DNS fasulle, immesse nel DNS

→ DNS poisoning
→Sovraccarico router o host
→ (Distributed) Denial of service

Exploit [debolezze sfruttate]

→ Normalmente software bug
→ Accesso non autorizzato ad un nodo
→ Furto di informazioni
→ Perturbazione servizio
→ Causare “crash” del nodo
→ Attivare percorsi di esecuzione poco comuni, che sono quelli testati di meno, per mezzo di pacchetti inusuali
→ Invio di pacchetti inusuali
→ Per esempio frammenti IP sovrapposti
→ Bug nella pila di rete, implica “crash” nel sistema
→ Richieste inusuali alle applicazioni

Furto di “identificatore” di rete

→ Address spoofing [appropriazione di indirizzo]
→ Server falsi
→ Modifica di pacchetti “al volo”

Esecuzione inconscia di codice dannoso

→ Computer virus
→ Eventualmente in e-mail
→ Trojan horse [cavallo di Troia]
→ Nascosto in un programma
→ Worm [verme]
→ Si propaga tramite rete, quando vanno in esecuzione

Obiettivi

→ Accesso (backdoor), al sistema tramite apertura di porte
→ Furto di dati
→ “Crash” del sistema
→ Uso dell’host (botnet)
→ Attacchi, anonimizzazione
→ Spiare (video, tastiera)

Inoltre

→ “indovinare” password
→ Per esempio usando dizionari
→ Phishing
→ Frodi tramite e-mail

Soluzione: educazione utente

8) Come funziona IPSec?

IPsec - IP SECURITY

Esso crea un framework (una infrastruttura) per permettere a due entità in comunicazione di mettersi d'accordo su quali protocolli usare, per autenticarsi, per scambiarsi le chiavi, e per capire quali sono gli algoritmi da usare, sia per la criptazione, sia per l'autenticazione.

Caratteristiche

→ Criptazione e autenticazione
→ Un paio di chiavi di sessione per ogni direzione, quindi in totale ci sono quattro chiavi
    → Una per criptazione
    → Una per autenticazione
→ Cambiate periodicamente

IPsec usa uno schema detto IKE, Internet Key Exchange.

Internet Key Exchange (IKE)

Usato per accordarsi su

→ Protocolli da usare per scambiare le chiavi
→ Algoritmi che si usano per la criptazione e per l'autenticazione
    → Criptazione (DES, 3DES, RC5)
    → Autenticazione (MD5, SHA1)
→ Chiavi, cioè IKE è usato per i protocolli che servono per accordarsi sulle chiavi

IKE

→ Include svariati protocolli, esso è un framework
    → Per esempio ISAKMP: Internet Security Association and Key Management Protocol
→ Autenticazione dei comunicanti
→ Scambio chiavi tra i comunicanti

Chiavi iniziali, possono essere basate

→ su un segreto condiviso, devono avere una relazione preesistente
→ su certificati digitali

Scambio di chiavi

→ Diffie-Hellman, è un algoritmo
→ Crittografia a chiave pubblica per autenticare i comunicanti da un lato e poi dall'altro la criptazione delle chiavi scambiate
→ Criptazione delle chiavi scambiate
    → DES

Una volta che le due entità di comunicazione hanno le chiavi per cifrare, i dati dovranno essere cifrati e comunicati.

IPsec lavora a livello IP e ha due modalità di funzionamento differenti: transport mode encapsulation e IPsec tunneling

Transport Mode Encapsulation

Prevede la cifratura delle informazioni di livello trasporto ed è basata su un formato di pacchetto detto ESP, per il quale, se abbiamo un pacchetto che deve andare da S a B, ed il pacchetto ha l'intestazione IP, contiene un messaggio TCP, con l'intestazione ed i dati, viene aggiunto tra l'intestazione IP e l'intestazione TCP una intestazione aggiuntiva ESP; ESP definisce dunque una intestazione, ma anche una coda al messaggio, come da figura. L'intestazione ESP permette di autenticare tutto il campo payload del pacchetto IP originale, e quindi in questo caso il messaggio TCP. Da questo il nome Transport Mode Encapsulation, in quanto va ad incapsulare il livello trasporto e a proteggere e ad autenticare il livello trasporto e a cifrare il livello trasporto. Qualcuno che guarda il pacchetto (campo dati) non riesce a vedere cosa contiene il pacchetto. Quello che riesce a vedere è che esso è un pacchetto IP che va da S a D, vede che c'è una intestazione ESP, in cui ci saranno informazioni per il destinatario su quali algoritmi usare per decifrare e verificare l'autenticità. Si noti che l'autenticazione prevede anche l'intestazione ESP ed una parte di coda ESP, questo affinche nessuno possa modificare tale informazione senza che il ricevente se ne accorga.

Authentication header

C'è poi un'altro tipo di header, detto authentication header, che non fornisce una soluzione di cifratura, ma solo di autenticazione, diversa da quella dell'ESP, in quanto questa copre anche l'intestazione IP. La conseguenza è che usando l'authentication header, nessuno può modificare l'intestazione IP, ma può solo vederne i contenuti. Si ha dunque una funzionalità di integrità, e anche di autenticazione in quanto esse sono strettamente legate.

L' authentication header non fornisce funzionalità di cifratura e, per questo, si può usare insieme all'ESP.

Da notare che con una soluzione del genere si avranno problemi con il NAT, in quanto l'indirizzo mittente e/o quello destinazione potrebbero essere cambiati e, vedendo che il pacchetto non è integro, la destinazione scarterà il pacchetto.

Riassumendo il Transport Mode Encapsulation

→ Usata per comunicazioni tra host
→ ESP: Encapsulation Security Payload
→ Authentication header

Esiste un'altra modalità dell'IPsec, detta IPsec Tunneling

IPsec Tunneling

→ VPN: Virtual Private Network, in cui si cerca di collegare reti aziendali attraverso una rete pubblica come Internet. Si hanno dei dispositivi detti VPN gateway o, IPsec gateway se si usa IPsec. Questi dispositivi prendono un pacchetto, ad esempio che transita da A a B e lo mettono in un altro pacchetto che va da X a Y. Quando il pacchetto arriva a Y, Y vede che il pacchetto è per lui e vede che dentro c'è un altro pacchetto, toglie il pacchetto interno e lo immette nella rete aziendale. Questa operazione è detta, in generale, tunneling. IPsec prevede di funzionare in questa modalità, quindi fare il tunneling, ed inoltre cifrare il contenuto del pacchetto.

Quindi, così facendo, si parla di tunnel mode encapsulation

Tunnel Mode Encapsulation

Nel tunnel mode incapsulation si ha il pacchetto originale di prima, che va da S a D, imbustato in un pacchetto che va da X a Y, protetto con le varie tecniche viste prima:

1. con una sola intestazione ESP

2. con una intestazione AH che proteggerà sia l'intestazione interna che quella esterna

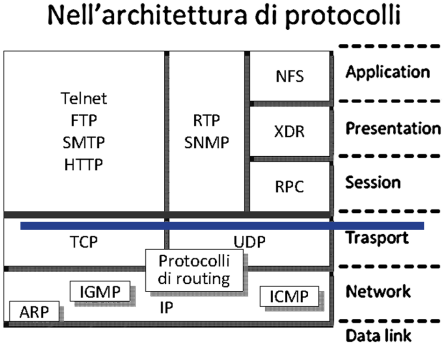
3. con un ESP che nasconde completamente il pacchetto interno ed una intestazione AH che protegge tutto quanto.

Nell'usare il tunnel mode, un osservatore non riesce a leggere gli indirizzi del mittente e del destinatario, cioè il pacchetto originale è completamente nascosto all'osservatore, mentre con il transport mode encapsulation gli indirizzi originali sono sempre visibili.

IPsec è una soluzione abbastanza robusta, ma complicata da utilizzare, in quanto ha tanti parametri di configurazione e tante modalità di funzionamento diverse, esiste dunque un'altra soluzione, molto più utilizzata, che si chiama Secure Socket Layer, SSL.

9) Che cos'è SSL?

SSL SECURE SOCKET LAYER



Nell'architettura di protocolli, i socket sono una interfaccia di programmazione che i livelli superiori possono usare per accedere ai servizi del livello trasporto. Normalmente le applicazioni, i protocolli di livello applicativo, usano i socket per chiedere l'apertura di connessioni TCP, l'invio di messaggi TCP, l'invio di messaggi UDP. L'SSL si preoccupa di cifrare e di autenticare le informazioni che le applicazioni cercano di mandare su una connessione TCP o in un messaggio UDP prima che questi vengano inseriti nel messaggio

Caratteristiche

→ Autenticazione delle due entità in comunicazione

→ Permette la creazione di una sessione di trasporto sicura, che è criptata ed autenticata

→ TLS: Transport Layer Security, che è la versione standard di SSL, che nasce nella comunità Internet, ma non è standard

SSL e TSL sono attualmente implementati in librerie che sono utilizzate dalle applicazioni. Le due soluzioni sono molto simili ed hanno differenze molto piccole. Esse sono largamente utilizzate, per rendere sicuri protocolli che normalmente non lo sono e che, per mezzo di SSL lo diventano, subendo anche una modifica nel nome:

→ POPS, Secure IMAP, Secure SMTP, HTTPS, SFTP

→ Normalmente diversa porta

→ HTTP: 8O, HTTPS: 443

→ Eventualmente la stessa: STARTTLS

SSL assicura uno "strato sicuro" al di sopra del livello trasporto.

Normalmente, quando il server usa il protocollo sicuro, aspetta su una porta diversa.

La porta può essere la stessa se si usa il TLS, in quanto esso prevede un protocollo che si chiama STARTTLS che consente al server ed al client di accordarsi sull'usare la modalità sicura o no, mentre quando il client usa l'SSL si aspetta che il server usi l'SSL, ecco perchè usa una porta diversa.

Di seguito vediamo un meccanismo generale dell'SSL, di come inizia la comunicazione (SSL Handshake).

SSL Handshake [presentazione]

Quando il client apre una connessione, ad esempio TCP, al server, e la connessione è stata aperta, il client manda un messaggio ClientHello al server, che è un messaggio dell'SSL. Il server risponde fornendo al client il proprio certificato digitale che contiene la chiave pubblica del server. Il client a questo punto può verificare l'autenticità del certificato e questo lo fa avendo una serie di certificati di certification authority messi sul sistema dal costruttore del sistema operativo del client; se non ce l'ha segue la procedura vista a suo tempo. Dopo la verifica del certificato il client genera una chiave simmetrica, quindi una chiave da usare con crittografia simmetrica, e la manda al server criptandola usando la chiave pubblica del server. In questo modo un osservatore non può capire quale è la chiave. A quasto punto il client ed il server hanno una chiave privata che possono usare per comunicare.

In realtà, in questa fase, il client ed il server negoziano tutta una serie di parametri.

Negoziazione parametri

→ Il client offre

→ Lista di "cypher", che è una lista di algoritmi di criptografia che è in grado di usare con relativi parametri

→ Parametri

→ Il server

→ Sceglie i cypher

→ Può richiedere il certificato del client, per verificare l'identità del client

Caratteristiche legate alla sicurezza, esse dimostrano che questo protocollo è parecchio sicuro

→ Solo hello e cert del server sono "in chiaro", tutto il resto viene cfrato

→ Il client ed il server usano una coppia di chiavi di sessione per direzione

→ Criptazione

→ Autenticazione

→ Cambiate periodicamente

10) Cosa si intende per "zona demilitarizzata"?

FIREWALL

Un filtro di pacchetti

Regole basate sui campi dei protocolli

→ Regole del tipo: indirizzi mittenti A-F possono comunicare con server S

→ Porta P usata su server Q

→ HTTP ammesso su server R

→ Scarta ogni altro pacchetto

Diversi tipi di firewall

→ Regole a livello di singolo pacchetto (stateless)

→ Regole a livello flusso, cioè del tipo “ammetti le connessioni TCP aperte da un cliente interno alla rete aziendale”

→ Stateful, osservazioni di un certo periodo per mettere in correlazione informazioni diverse

→ Application firewall, vanno a guardare i dati a livello applicativo

→ Firewall in grado di identificare malware e virus

I firewall devono essere in posizione strategica.

Dal punto di vista fisico si realizzano sottoreti diverse, quella interna privata ed una pubblica, detta de-militarized zone, DMZ, in cui l’accesso è permesso dall’esterno, ma in modo controllato, solo verso server predefiniti.

In quella interna l’accesso è estremamente controllato ed è permesso solo dai server che si trovano nella ZONA DEMILITARIZZATA.

11) Quali sono le differenze tra IPv4 e IPv6?

Il formato degli indirizzi, inoltre gli Aggregatable Global Unicast [aggregabili]

Sono indirizzi che identificano le stazioni globali su tutta la rete, quindi non locali. Poter aggregare significa avere un prefisso comune a tutte le reti che stanno, ad esempio in Europa in modo da poter avere nei router una unica informazione di routing: per raggiungere l’Europa “vai da quella parte”. Affinché questo sia possibile occorre che in Europa tutti gli indirizzi abbiamo lo stesso prefisso, cioè che siano assegnati in modo da avere un prefisso comune, cosa che in IPv4 non si faceva.

→ Tali indirizzi iniziano per bx001

→ Prima cifra in esadecimale è 2 o 3

→ Assegnazione topologica, con la gerarchia che esiste tra i service provider; facendo così otterremo una efficace aggregazione

Altra differenza il prefisso, è identificato da una coppia indirizzo/netmask in IPv4, mentre in IPv6 la coppia indirizzo/netmask è sostituita da un prefisso indirizzo/N, dove N è la lunghezza di prefisso (in bit), come sotto riportato

→ FEDC:0123:8700::/36

→ 1111111011011100
0000000010010001110000

In IPv6 non ci sono classi di indirizzo, per cui la lunghezza del prefisso non si capisce guardando i primi bit. Il prefisso è esplicitamente specificato nella notazione *indirizzo/lunghezza del prefisso*.

Altre differenze:

Assegnazione di indirizzi

Come vengono assegnati gli indirizzi ha un impatto anche su come viene organizzato il prefisso.

Negli indirizzi agregatable global unicast è deciso che il prefisso abbia lunghezza 64 bit e l'identificatore di interfaccia è di 64 bit, che è un numero di bit elevato, ma la ragione per cui si scelgono 64 bit per l'identificatore dell'interfaccia è che, se si vuole, è possibile usare l'indirizzo Ethernet in versione estesa come identificatore di interfaccia.

Questa è comunque solo una parte dello spazio di indirizzamento, con gli indirizzi che iniziano per 001 in binario.

L'assegnazione deve essere fatta in dipendenza della gerarchia che c'è nella topologia di rete, ad esempio la gerarchia dei service provider.

Plug and Play, non è necessario configurare le stazioni

PROTOCOLLI MODIFICATI

Ci sono cambiamenti nell'architettura di protocolli in IPv6 rispetto a Iv4.

→ IP
→ ICMP
→ ARP, che scompare
    → Funzionalità di ARP integrate in ICMP
→ IGMP, che scompare
    → Integrato in ICMP

Altri protocolli sono solo aggiornati

→ DNS (nuovo tipo di record di tipo AAAA)
→ RIP e OSPF
→ BGP e IDRP , cambia il modo in cui si chiamano le destinazioni
→ TCP e UDP, in TCP cambiano gli indirizzi per cui bisogna cambiare l'implementazione del protocollo, non il formato dei messaggi TCP o UDP, ma la implementazione dei protocolli
→ Interfaccia socket, che si usa per accedere ai servizi

12) Qual'è il formato di un indirizzo IPv6?

INDIRIZZI IPv6

Come li scriviamo e come li usiamo

Notazione: 8 numeri esadecimali separati da ":"

Gruppi di 2 byte

FEDC:BA98:0876:45FA:0562:CDAF:3DAF:BB01

1080:0000:0000:0007:0200:A00C:3423:A089

Scorciatoie

Per visualizzare indirizzi in modo user-friendly

Gli 0 iniziali di ogni gruppo di cifre possono essere omessi

→ 1080:0:0:7:200:A00C:3423

Gruppi di 0 possono essere sostituiti da "::"

→ 1080::7:200:A00C:3423

Organizzazione dello spazio degli indirizzi (2128)

→ Indirizzi Multicast

→ 1111 1111, il primo byte è fatto da tutti 1

→ FFxx:... è un indirizzo multicast

→ Indirizzi Link local/site local

→ Equivalenti ad indirizzi IPv4 privati

→ 1111 1110 1

→ Link local: FE80::/64, cioè i successivi 64 bit hanno un qualsiasi valore; con questa notazione si indica un prefisso in uso; gli indirizzi Link Local si usano solo su una rete locale e sono gli equivalenti degli indirizzi automatici IPv4, quelli che iniziano per 169.254

→ Site local: FEC::/10, sono l'equivalente degli indirizzi privati in IPv4 come 10.0.0.0

Gli indirizzi rimanenti prendono il nome di Global Unicast, cioè indirizzi unicast globali e servono per dare indirizzi alle interfacce delle stazioni. Di seguito come è organizzato lo spazio global unicast.

Indirizzi rimanenti Global Unicast [unicast globali]

Organizzazione dello spazio Global Unicast

→ Indirizzi per IPv4 interoperability, sono indirizzi dedicati alla possibilità di avere stazioni IPv4 e IPv6 che coesistono ed operano sulla stessa rete; sono indirizzi che hanno i primi 80 bit a zero; di questi indirizzi ce ne sono due tipi: IPv4-apped e IPv4-compatible. Gli indirizzi IPv4 interoperabilty includono un indirizzo IPv4 in un indirizzo IPv6

→ 0...0 (80 bit) → 0::/80, tipica notazione IPv6

→ Per la fase di transizione da IP4 a IPv6

→ Indirizzi IPv4-mapped, gli 80 bit a 0 iniziali sono seguiti da 16 bit a 1, per cui sono "impegnati" 96 bit, con 32 bit finali in cui metteremo un indirizzo IPv4

→ 16 bit a 1 → 0:0:0:0:0:FFFF::/96

→ Indirizzi IPv4-compatible, gli 80 bit iniziali a 0 sono seguiti da ulteriori 16 bit a 0

→ Altri 16 bit a 0 → 0::/96

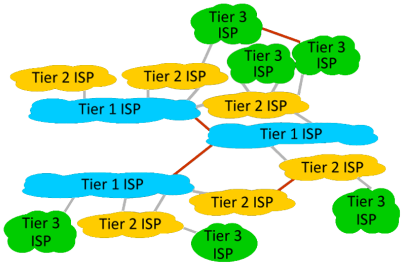
Per esempio 0::0::0::0::0::0::A00::1

→ Notazione compatta, degli indirizzi IPv4-compatible

→ ::A00:1

→ Notazione speciale, adottata per indirizzi IPv4-compatible

→ ::10.0.0.1



Gerarchia dei service provider

Aggregatable Global Unicast [aggregabili]

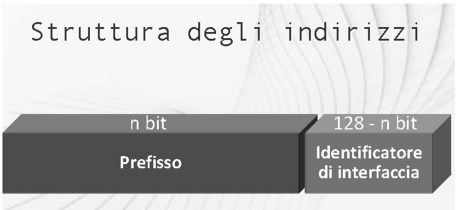
Sono indirizzi che identificano le stazioni globali su tutta la rete, quindi non locali. Poter aggregare significa avere un prefisso comune a tutte le reti che stanno, ad esempio in Europa in modo da poter avere nei router una unica informazione di routing: per raggiungere l'Europa "vai da quella parte". Affinché questo sia possibile occorre che in Europa tutti gli indirizzi abbiamo lo stesso prefisso, cioè che siano assegnati in modo da avere un prefisso comune, cosa che in IPv4 non si faceva.

→ Tali indirizzi iniziano per 0x00 1

→ Prima cifra in esadecimale è 2 o 3

→ Assegnazione topologica, con la gerarchia che esiste tra i service provider; facendo così otterremo una efficace aggregazione

I service provider di Tier 1 riceveranno un prefisso, quelli di Tier 2, che ad essi sono collegati, riceveranno un prefisso ricavato da quello dei service provider di Tier 1, un pochino più lungo, e così via.



Struttura degli indirizzi

Gli indirizzi sono divisi in due parti, una è il prefisso e l'altra è l'identificatore dell'interfaccia.

13) Quali sono le principali problematiche relative alla mobilità nelle reti IP?

SFIDE DELLA MOBILITÀ

Movimenti trasparenti (ad IP)

Quando una stazione si muove, può fare dei movimenti trasparenti dal punto di vista del protocollo IP.

→ All'interno della stessa rete fisica

→ Nella cella o tra celle di una rete cellulare

→ Tra BSS di un ESS WiFi, cioè tra un Basic Service Set ed Un Extended Service Set WiFi

→ Tra le porte di uno switch

→ In tutti questi casi la mobilità è gestita dal livello 2 , cioè i protocolli e i dispositivi di livello 2 (i bridge, i switch, gli acces point, le base station) di una rete mobile si occuperanno di accorgersi che la stazione non è più, ad esempio, collegata ad una porta di uno switch, ma ad un'altra, per cui manderanno le trame di livello 2 dall'altra. E così anche gli access point di un Extended Service Set

Dal punto di vista del protocollo IP non è cambiato nulla , la stazione è sempre parte della stessa rete fisica.

La mobilità non è trasparente quando la stazione cambia rete fisica, in quanto il prefisso dell'indirizzo IP dipende dalla posizione nella rete e questo perchè la logical IP subnet (LIS), cioè tutte le stazioni

con lo stesso prefisso corrisponde ad una rete fisica.

Quando si cambia rete fisica si ha uno spostamento

Il prefisso dell'indirizzo IP dipende dalla "posizione"

→ La logical IP subnet (LIS) corrisponde ad una rete fisica

→ Se una stazione cambia rete fisica

→ Deve cambiare LIS (prefisso)

→ Cambiamento di indirizzo

Dare un nuovo indirizzo alla stazione è un problema, perché tutte le connessioni TCP e sessioni UDP attive vengono interrotte e quindi ogni servizio in corso si interrompe, ad esempio trasferire un file, controllare la posta elettronica, fare una telefonata ecc.

→ Gli identificativi di sessione/connessione includono l'indirizzo IP

→ La quintupla "magica" (indirizzo sorgente e destinazione, protocollo di trasporto, porta sorgente e porta destinazione) identifica univocamente i pacchetti di un certo flusso, cioè di una certa connessione TCP o di una certa sessione UDP

→ Il cambio di indirizzo IP crea problemi anche sui meccanismi di autorizzazione basati sull'indirizzo che quindi rifiuteranno la stazione

14) Cos'è e a cosa serve Mobile IP? Descriverne le caratteristiche e la struttura

MOBILE IP

Una soluzione, fra altre, che risolve la sfida di poter cambiare rete fisica senza dover, in qualche modo, cambiare indirizzo e scambiare dati. Non usatissima.

Caratteristiche

→ RFC 3344 (2002)

→ Trasparente a livello trasporto e applicazioni

→ Interoperabilità con stazioni che non hanno supporto per mobile IP

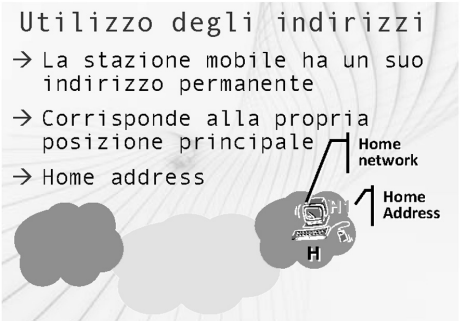
→ Scalability

→ Sicurezza

→ Autenticazione per evitare impersonificazione di stazione mobile [spoofing]

→ Mobilità limitata

→ Al più un "movimento" al secondo



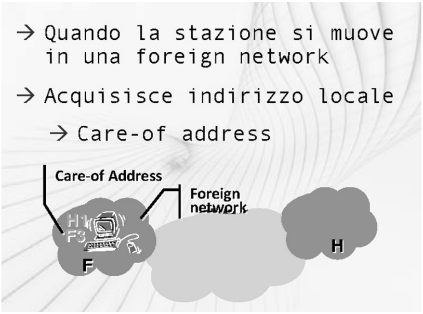
Utilizzo degli indirizzi

→ La stazione mobile ha un suo indirizzo permanente

→ Corrisponde alla propria posizione principale, quando cioè si trova nella home network

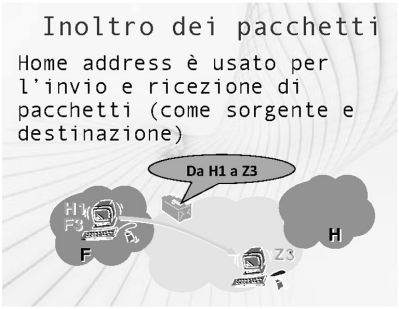
→ Home address, l'indirizzo permanente

→ Quando la stazione si muove in una foreign network, ovvero una rete diversa dalla rete home



→ Acquisisce indirizzo locale di nome care-of address, che è traducibile come "presso, c/o"

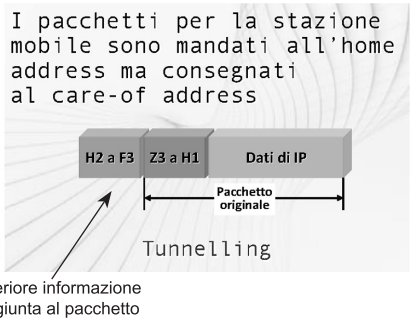
→ Care-of address, indirizzo locale in foreign network



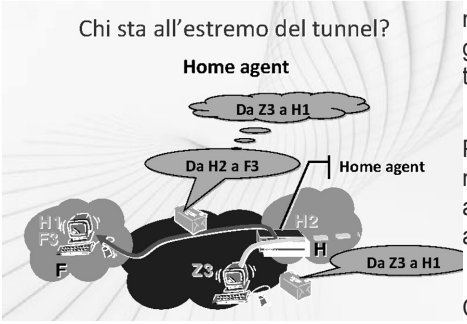
Inoltro dei pacchetti

Home address è usato per l'invio e ricezione di pacchetti (come indirizzo sorgente e indirizzo destinazione), quindi nell'esempio si può notare che quando una stazione manda un pacchetto da H1 a Z3, essa costruisce un pacchetto che va da H1 a Z3.

La stazione dovrà ricevere le risposte, per cui Z3 risponderà ad H1, ma in realtà i pacchetti per H1 che sono mandati all'home address, verranno consegnati al care-of address, ovvero i pacchetti per la stazione mobile sono mandati all'home address ma consegnati al care-of address, come figura a lato



Chi sta all'estremo del tunnell? Cioè chi mette il pacchetto mandato dal server dentro un altro pacchetto all'indirizzo F3 per consegnarlo alla stazione mentre essa è mobile, ovvero è sulla foreign network? Esso sarà il cosiddetto home agent, che garantisce che i pacchetti siano alla fine consegnati alla stazione.



Per il care-of address abbiamo un paio di opzioni, di cui una, appena vista è la co-located care-of address, poi c'è il cosiddetto foreign agent care-of address

Co-located care-of address

→ Ottenuto dalla stazione sia in modo permanente che dinamico (per esempio tramite DHCP)

→ Più indirizzi IP necessari, uno per ogni stazione mobile nella foreign network e questo è limitativo

→ Host termina il tunnel, cioè l'host deve estrarre il pacchetto interno, il suo indirizzo home, dal pacchetto esterno inviato al care-of address e questo vuol dire che l'host ha un carico elaborativo maggiore

→ Non serve un foreign agent

Foreign Agent

Foreign Agent Care-of Address

→ Il care-of addrees è un indirizzo del foreign agent

→ Indirizzo può essere condiviso

→ Niente carico elaborativo nella stazione, in quanto il tunnel è terminato nel foreign agent

A prescindere dall'uso di una soluzione co-located o foreign agent care-of address, il foreign address deve essere registrato.

Registrazione

La stazione che opera su una foreign network notifica il proprio home agent del care-of address che sta usando, perchè l'home agent deve inoltrare il pacchetto. La registrazione serve per comunicare il care-of address

→ Comunica il care-of address

La registrazione può essere fatta dal foreign agent, se il care-of agent è configurato su un foreign agent.

La stazione deve in qualche modo scoprire che c'è un foreign agent, far sapere al foreign agent che ha bisogno di supporto per la mobilità, comunicare il proprio home address e, a quel punto, il foreign agent può notificare l'home agent.

Messaggi di registrazione

Il protocollo Mobile IP definisce, oltre all'architettura, anche protocolli che servono per fare questi scambi di informazione, queste registrazioni. Ci saranno anche funzionalità di registrazione al fine di evitare che un host maligno faccia finta di essere parte della home network per accedervi.

→ Messaggi del mobile IP protocol

Funzionalità di autenticazione

→ Per evitare che un host maligno faccia finta di essere parte della home network per accedervi ed acquisire un indirizzo di una rete

E' dunque fondamentale che nella fase di registrazione ci siano meccanismi di autenticazione.

E' anche importante avere dei meccanismi per gli annunci degli agenti, cioè gli agenti Mobile IP devono rendersi noti alle stazioni.

Annuncio dell'agente

→ Gli agenti Mobile IP devono rendersi noti alle stazioni, questo si fa con una estensione del messaggio ICMP router advertisement

→ Estensione del messaggio ICMP router advertisement

→ Grazie a questi messaggi, la stazione mobile può capire "dov'è", se si trova nella home network o nella foreign network; la stazione mobile può rimanere in ascolto e vedere se c'è un home agent o un foreign agent che si annuncia e quindi capire che c'è supporto per la mobilità e capire se si trova nella home network o nella foreign network. Se non riceve degli annunci entro un certo tempo, la stazione li può anche sollecitare

→ Una stazione mobile può sollecitare l'annuncio del mobile agent, usando il messaggio ICMP router solicitation

La soluzione Mobile IP è stata progettata per supportare la mobilità, inizialmente quando si è cominciato a lavorare su IPv6, ma poi si è estesa anche a IPv4.

Essa ha però delle limitazioni, quindi sono sia state progettate soluzioni diverse, sia, a supporto della mobilità, soluzioni progettate per altri ambiti, di cui una è la cosiddetta Proxy Mobile IPv6

15) Cos'è e a cosa serve Proxy Mobile IPv6? Elencarne le caratteristiche e descriverne l'architettura

PROXY MOBILE IPv6 (PMIPv6 O PMIP)

Nasce per supportare la mobilità in IPv6, ma in realtà può essere usata anche in situazioni in cui IPv6 non è utlizzzato.

Caratteristiche

→ Non richiede supporto negli host, cioè proxy mobile IP richiede che gli host siano in grado di capire i protocolli del mobile IP, che gli host siano a conoscenza della presenza di home agent, un foreign agent. In proxy mobile IPv6 l'host non deve avere nessuna modifica, ma ci sarà un elemento della rete che segue i movimenti degli host

→ Un elemento di rete segue i movimenti degli host

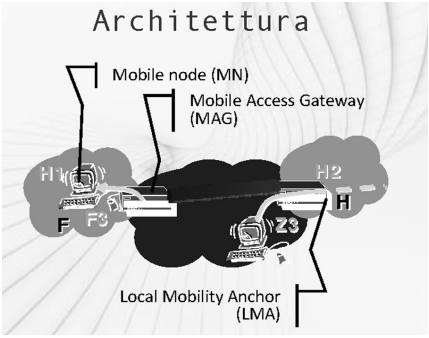
→ Elemento basato su protocolli standard e comunemente utilizzati

→ Un elemento di rete si occupa delle azioni legate alla mobilità

→ Segnalazione (dell'home agent o del foreign agent)

→ Tunneling

→ Protocolli specifici del proxy mobile IPv6 usati dall'elemento di rete



Architettura

la stazione ha il suo indirizzo e lo mantiene ovunque vada; sarà il MAG che si accorge che nella rete c'è una stazione in mobilità e a fare le cose che servono; il MAG (o il foreign agent nel caso precedente) e la stazione devono essere sulla stessa rete fisica; la stazione che si muove non sà nulla del supporto alla mobilità, che è supportata completamente dai dispositivi di rete. In questo tipo di architettura ci dobbiamo spostare dove c'è un MAG, nell'altro ci possiamo spo-

stare nella rete anche in assenza di foreign agent.

In sostanza questa architettura ha il MAG, che deve avere funzionalità diverse da un foreign agent.

Architettura non necessariamente IPv6

→ Mobile node: IPv4 o IPv6 (o dual stack)

→ La rete tra il MAG (Mobile Access Gateway) e il LMA (Local Mobility Anchor, equivalente dell'home agent): IPv4 o IPv6;

→ Segnalazione basata su IPv6, ma potrebbe essere IPv4

→ LMA è un Home Agent Mobile IPv6

16) Cos'è e a cosa server LISP?

LOCATOR/IDENTIFIER SEPARATION

PROTOCOL (LISP)

[PROTOCOLLO PER LA SEPARAZIONE

DI LOCALIZZATORE E IDENTIFICATORE]

E' una ulteriore soluzione per gestire la mobilità.

Gli indirizzi IP hanno due funzioni

→ Identificare le stazioni

→ Localizzare le stazioni

→ Cioè assistono i router nel trovare un percorso verso gli host

LISP le separa, mentre nell'indirizzo IP tradizionale le due funzioni sono fuse.

Quindi identifichiamo da un lato un identifier a e dall'altro un route locator.

Identifier e Route Locator

→ Possono essere entrambe un indirizzo IP

→ Oppure possono essere qualcos'altro

→ Per esempio coordinate GPS, indirizzo MAC

Quello di cui abbiamo bisogno è quello di avere un valore univoco su tutta la rete che permette di identificare la stazione e poi di un qualche altro valore che in qualche modo aiuta a trovare la stazione nella rete.

LISP non ha nessun interesse a sapere cosa siano questi due valori, ma fornirà tutti gli strumenti per fare la corrispondenza, ovvero dato un identificatore trovare il localizzatore e quindi poter raggiungere la stazione.

Le stazioni sono sempre identificate con l'identificatore e, se si spostano, avranno bisogno di un localizzatore diverso.

I campi di applicazione del LISP

→ Mobilità (LISP non era stato pensato per questo)

→ Scalability del routing

→ Attraversamento di zone IPv4 a pacchetti IPv6 e viceversa

→ Network virtualization, cloud computing; si identificano reti virtuali in un datacenter. Locazione identica per identità diverse

→ Multihoming, quando un organizzazione è cliente di due service provider

LISP è applicabile in quei casi in cui abbiamo un identificatore che non dice ai router come inoltrare i pacchetti. I campi di applicazioni sono molti.

Principi di funzionamento del LISP

→ Mapping system: identifier ↔ locator, LISP fornisce una corrispondenza tra identifier e locator

→ Dapprima basato su BGP

→ Poi ispirato al DNS

→ Uno qualsiasi va bene

→ LISP è usato dai router, dagli apparati di rete

→ Quindi gli host non sono consci

LISP e mobilità

→ Gli host mantengono l'identificatore muovendosi

→ Un nuovo locator è acquisito muovendosi

→ LISP è usato per trovare la corrispondenza tra l'identificatore, cioè l'home address ed il care-of address e anche per assicurare la consegna

□

7 Novembre 2015