Indice

La Storia di Internet

24 gennaio 2018

1962:
- Internet è il risultato dell'evoluzione del concetto di **Galactic Network**, un'infrastruttura basata su un insieme di computer globalmente interconnessi al fine di scambiare dati, discusso da **Licklider** al MIT.
- **Licklider** avvia programma di ricerca **ARPA** (Advanced Research Project Agency) che cambierà il proprio nome nel '71 in **DARPA** (Defense ARPA), poi di nuovo in ARPA nel '93 ed infine in DARPA nel '96.
- Leonard Kleinrock pubblica il primo articolo sulla teoria del **Packet Switching**.

1965:
- Thomas Merrill e G. Roberts riescono a far comunicare due computer (Uno in Massachussets e l'altro in California), creando la prima **wide-area computer network** della storia.

1969:
- Nasce ARPANET (ARPA NETwork), prima rete globale dalla quale poi nacque Internet nel 1983, dalla connessione host-to-host di due nodi, il primo a UCLA e il secondo a SRI.
- **Steve Crocker** invia il primo **Request For Comments (RFC)**.

1973:
- A Standfort, **Vint Cerf** e **Bob Kahn** creano il **TCP/IP**: nasce la posta elettronica.
- Bob Metcalfe inventa l'**Ethernet**.

1980:
- TCP viene adottato come protocollo standard dal DoD (Department of Defense).
- Inizia l'attività di USENET (USEr NETwork) e i relativi primi gruppi di discussione delle NEWS *(prima applicazione client-server su larga scala).*

1983:
- **Jon Postel** sviluppa il **DNS** e la forma *user@host.domain.*
- ARPANET adotta il TCP/IP, mentre l'ISO/OSI è sempre meno usato.

- Vengono introdotte le reti IP di classe A, B, C .
- Vengono introdotti i **TLD** .edu, .com, .net, .org oltre quelli della codifica ISO.
- **NSF** (National Science Foundation) costituisce i supercomputer centers per servire la comunità scientifica americana.

1985:
- L'algoritmo di routing originario viene rimpiazzato dai protocolli **IGP** ed **EGP**.
- NFS lancia **NFSNET** (NFS NETwork), un backbone a 56K basato su TCP/IP, che verrà poi privatizzato nel '95.
- Tra l'85 e l'86 Internet, TCP ed Ethernet hanno un picco di diffusione, e nascono le prime multinazionali operanti nel settore delle reti (IBM, Proteon, Synoptis, CISCO...)

1987:
- NFS passa a linee T1 (1.55Mbps)e con il loro successo si vuole migrare a linee T3 (45Mbps).
- Diventano serie le problematiche di Network Management.

1989:
- Vint Cerf e Bob Kahn organizzano il primo workshop sul Gigabit.
- **Tim Barners-Lee** al CERN propone il concetto di **ipertesto**: nasce il **WWW**.

1990:
- ARPANET chiusa, nascono i primi 130 incidenti relativi a WORMS.

1991:
- Nasce **PGP** (Pretty Good Privacy)

1992:
- Nasce **ISOC** (Internet SOCiety), fondata da Vint Cerf e Bob Kahn, di cui diventa parte l'*Internet Activity Board*.
- Il WWW esplode nella rete. Boom.

1995:
- Federal Networking Council (FNC) formula la
  Definizione di Internet:
  "Internet si riferisce al sistema di informazione globale che:
  - *è logicamente interconnesso attraverso un address space (spazio degli indirizzi) unico e globale, basato sull'IP o le sue successive estensioni e sviluppi;*
  - *è in grado di supportare comunicazioni mediante la suite Transmission control protocol/Internet protocol (TCP/IP) o le sue successive estensioni/sviluppi, e/o altri protocolli compatibili con l'IP;*
  - *fornisce, utilizza e rende accessibili, sia pubblicamente che privatamente, servizi di alto livello che poggiano sui differenti strati di comunicazioni e di infrastrutture a esse correlate"*

0 Internet Governance

IANA (Internet Assigned Numbers Authority): si occupa della gestione dello spazio di indirizzamento IP e dei nomi a dominio, dei numeri di Autonomous System, dei numeri di protocollo IP.
Delega la gestione locale ad alcune entità come **ARIN** (America), **RIPE NCC** (Europa) e **APNIC** (Asia-Pacifico).
Attualmente rimpiazzata da **ICANN** *Internet Corporation for Assigned Name and Numbers*.

IEPG (Internet Engineering Planning Group) coordina le operazioni di Internet, le attività e l'interoperabilità tra i vari Internet Service Operators mondiali (RFC1690).

0.1 Standards

Internet esiste anche grazie allo sviluppo, la verifca e l'implementazione di Standard Internet.

- **IETF**: Internet Engineering Task Force, **sviluppa** gli standards.
  L'**IRTF** (Internet Research Task Force) promuove la ricerca e lo sviluppo di Internet.
- **IESG**: Internet Engineering Steering Group, **verifica** gli standards
- **ISOC**: Internet SOCiety, **promulga** gli standards.

Tipi di standards **de jure** (codificati da organismi nazionali/internazionali) e **de facto** (massiccia adozione da parte degli utenti).

0.1.1 Organizzazioni

IEEE (Institute of ELectrical and Electronic ENgineers): attivo nello sviluppo di standards di comunicazione dati.

- **802**: sottocomitato concentrato sull'interfaccia fisica degli apparati e sulle procedure per gestire connessioni tra dispositivi di rete.

CCIIT (Consultative COmmittee for International Telephone and Telegraph), gruppo dell'**ITU** (International Telecommunications Union) è un'agenzia dell'ONU specializzata in telecomunicazioni.
Emano le *raccomandazioni* ogni 4 anni (ex. *V.21 Duplex 300 bit/s modem modulation*).

ISO (International Standards Organization) organo consulente dell'ONU, con lo scopo di promuovere lo sviluppo di standards nel mondo, favorendo lo scambio internazionale di cose e servizi.
Concepisce il modello a sette livelli **OSI** (Open System Interconnection)

0.2 Internet Aziendali

0.2.1 Intranet

Intranet è il termine che descrive l'uso delle tecnologie Internet **all'interno di una organizzazione** invece che per le connessioni con l'esterno (con l'internet globale).
E' necessaria una **rete locale** per connettere i computer e un'**informatizzazione diffusa** dei vari settori.

Router instrada i dati tra i pc aziendali e internet, controlla gli accessi alla rete locale, computer e servizi.

0.2.2 Extranet

Extranet è l'insieme di risorse hardware e software che realizzano la presenza visibile in Internet di una organizzazione *(data mining, servizi web...)*. Normalmente servizi posti in un'area speciale: la **DMZ** (De-Militarized Zone) in cui i server non sono critici e i servizi replicati da server protetti.

0.3 ISO OSI Reference Model

OSI introduce il concetto di **sistema** *(risorse hw, sw, periferiche, programmi)* e di **applicazione** *(programma che elabora dati ed eroga servizi)*.
7 Livelli, ciascuno dei quali sfrutta i servizi di quello inferiore e li eroga a quello superiore. Questi comunicano attraverso le loro *interfacce*. Le operazioni di un protocollo sono realizzate mediante i **protocolli**.

1. **Liv. Fisico**: Livello più basso, insieme di regole che specificano le connessioni elettriche e fisiche tra i dispositivi fisici.
   Corrisponde agli standard di interfaccia dei vari dispositivi (es. RS232, V.24, V.35 ...)

2. **Liv. Collegamento**: Specifica come un dispositivo accede al mezzo specificato dal livello fisico e come realizza la comunicazione con un nodo adiacente. E' suddiviso nei sottolivelli LLC e MAC. BSC, HDLC

3. **Liv. Rete**: Responsabile della connessione tra due nodi della rete, del routing e dello scambio di informazioni. IP.

4. **Liv. Trasporto**: Garantisce il corretto trasferimento delle informazioni (controllo errori, sequenza frames...) E' il primo livello end-to-end. TCP, UDP.

5. **Liv. Sessione**: Fornisce regole per la gestione dei flussi di dati tra nodi (attivazione/terminazione connessioni, controllo flusso msg, controllo dati).

1. **Liv. Presentazione**: Responsabile della trasformazione dei dati, della loro formattazione, in modo tale da poter essere ricevuti correttamente dal mittente. De/crittografia, de/compressione dei dati.

1. **Liv. Applicazione**: comprende tutti i programmi applicativi relativi alla rete. VT (Virtual Terminal), posta elettronica X.400, accesso a DB X.500.

1 Liv. Fisico

1.1 Mezzi trasmissivi

Le diverse reti utilizzate per l'accesso ad internet sfruttano mezzi trasmissivi differenti.

1.1.1 Cavo coassiale

Cavo originale delle reti Ethernet, costituito da un filo di rame ricoperto da un materiale dielettrico, quindi da una calza in rame e una guaina in polietilene. Sulla base del diametro, se ne distinguono due varianti:

- thick (RG-8), di diametro 0.4 cm;
- thin (RG-58). di diametro 0.25 cm, che usa un connettore a T, tipico delle vecchie LAN.

1.1.2 Doppino telefonico

Un *doppino ritorto* (detto anche *coppia bifilare*) è un tipo di cablaggio composto da una coppia di conduttori in rame isolati ritorti, utilizzato nella **rete di accesso**[1] alla **PSTN** (*Public Switched Telephone Network*). I doppini sono utilizzati anche, in un intreccio di quattro coppie, per trasmettere dati in una rete locale, attraverso il protocollo Ethernet.

1.1.3 Cavo UTP

Unshielded twisted pair: evoluzione del doppino telefonico, è costituito da più doppini intrecciati. É soggetto a disturbi elettrici, in particolare se il segmento è molto lungo, poiché agisce come un'antenna, ma è vantaggioso in termini economici.

1.1.4 Cavo STP

Shielded Twisted Pair: risente dei disturbi elettrici in misura minore del cavo UTP, ma è più costoso e difficile da stendere.

1.1.5 Fibra ottica

Le fibre ottiche sono filamenti realizzati in modo da poter condurre al loro interno la luce (propagazione guidata). Ne esistono due tipologie, le **monomodali**, in cui la sorgente è un laser, e le **multimodali**, in cui la sorgente è un LED.

[1] Termine con cui si indica la parte di rete destinata al collegamento fra la sede dei singoli utenti finali fino alla prima centrale di commutazione, e più in generale al collegamento tra un utente e il suo provider.

Presentano numerosi vantaggi: sono flessibili, immuni ai disturbi elettrici ed alle condizioni atmosferiche più estreme e poco sensibili alle variazioni di temperatura.
Le architetture di rete che usano la fibra ottica sono indicate con **FTTx** (*Fiber To The x*). In particolare, **FTTH** (*Fiber To The Home*) indica il collegamento che raggiunge la singola unità abitativa.

2 Liv. Collegamento

2.1 Ethernet

É una delle tecnologie per il collegamento LAN più utilizzate al mondo, posto a metà fra il primo e il secondo livello (più in particolare nel sottolivello MAC) del modello ISO.

2.2 Sottolivelli

Il DLL, in caso di reti LAN brodcast, è diviso in due sottolivelli:

2.2.1 MAC

Il livello MAC (*Media Access Control*) è diverso per ciascun tipo di LAN e disciplina l'accesso contemporaneo di molti nodi ad un solo canale di comunicazione condiviso, evitando e gestendo le collisioni.

Indirizzo MAC Indirizzo fisico della macchina, non individuabile dall'esterno utilizzato per l'instradamento diretto in reti locali, ovvero per raggiungere un host da una stessa sottorete passando per il solo livello 2.
La conversione degli indirizzi di livello 3 (es. IP) in indirizzi MAC di livello 2 è in genere eseguita dal protocollo ARP, mentre la procedura opposta da protocolli come RARP e DHCP.

2.2.2 LLC

Il livello LLC (*Logical Link Control*), posto tra livello MAC e *Livello di rete*, controlla il flusso di dati e gestisce gli errori, fornendo un'interfaccia unica per tutti i tipi di LAN. I protocolli PPP e HDLC fanno parte di questo sottolivello.

Standard I dati ricevuti dal livello superiore vengono incapsulati sottoforma di *frame LLC* ed inviati a quello inferiore (MAC), che si occuperà di trasmetterli sul mezzo fisico prescelto. I frame LLC sono costituiti dall'indirizzo sorgente, di destinazione, un campo di controllo ed infine i dati.
In base all'implementazione, il LLC prevede 3 diversi servizi fornibili al livello superiore:

- **LLC1**: servizio *Connectionless*, non è prevista alcuna forma di *conferma*, di *correzione errori* ne di *controllo del flusso*.
- **LLC2**: servizio *Connection-oriented* unicast (punto-punto) e simmetrica. Prevede meccanismi di *correzione errori* e di *sequenziamento dei dati*. Analogo ad altri protocolli di livello 2 come l'**??**.

- **LLC3**: servizio alternativo al LLC1 in quanto è *Connectionless*, ma prevede una *conferma di ricezione* (acknowledge - ACK) per i frame inviati e garantisce la *consegna ordinata* dei dati.

2.3 ATM

Asynchronous Transfer Mode, protocollo standard che si è imposto nelle connessioni LAN-LAN e LAN-WAN grazie alla sua capacità di trasportare in contemporanea segnali diversi.
Nelle reti ATM *maggiori sono le stazioni* a comunicare maggiori sono le prestazioni ed i dati vengono inviati *point-to-point*, mentre nelle LAN la *banda è condivisa* (con relativi problemi di affollamento) e i messaggi sono inviati in *broadcast*.
Altro vantaggio è il *routing*: prima di iniziare il trasferimento dei dati, viene costruito il percorso virtuale tra i due nodi. Questo scorpora il problema di determinare il percorso di instradamento dal trasporto effettivo.

Celle ATM L'unità di trasmissione dei dati è un pacchetto detto **Cella**; lunghezza fissa di *53 bytes* di cui:

- **5 byte di Header** che contiene
  - Virtual Path Identifier (VPI)
  - Virtual Channel Identifier (VCI): *all'interno di un path*
  - Payload Type (PT): *tipo di dato della parte user information (payload)*
  - Congestion Loss Priority (CLP)*scarica o meno la cella in caso di congestione*
  - Header Error Control (HEC)
- **48 di User Information** (payload)

Livelli ATM ATM usa un **PRM** (Protocol Reference Model) il quale prevede che il protocollo operi su 3 livelli:

- **Physical Layer**: l'equivalente del Livello Fisico nella gerarchia ISO/OSI
- **ATM Layer**: paragonabile ad una parte del Livello di Collegamento ISO/OSI, si occupa del routing;
- **AAL** (ATM Adaptation Layer): necessario alla connessione ed alla corretta comunicazione tra la rete ATM e reti non ATM. Qui si effettuano le operazioni di segmentazione/riassemblaggio dei dati.

L'architettura di ATM segue il principio del **Core & Edge**: nei nodi interni (**core**) avvengono operazioni sui due livelli più bassi, mentre i terminali utente (*edge*) operano su tutti i livelli (velocizza il trasporto nei nodi interni, relegando ai nodi esterni funzioni più costose)

2.3.1 AAL

Atm Adaption Layer, fornisce una classe di servizi attivabili in base alle esigenze dell'utente ed in base alle reti con cui ATM si interfaccia. Quindi si è deciso di suddividere le reti in quattro **classi di servizio** - A, B, C e D - secondo 3 parametri:

- **Timing** necessario o non necessario;
- **Bit rate** costante o variabile (CBR o VBR);
- **Connessione** di tipo Connection-oriented o Connectionless.

Ad ognuna di queste classi è stato fatto corrispondere uno specifico tipo di ATM Adaptation Layer:

- **AAL 1**: utilizzato per le applicazioni che richiedono emulazioni di circuito (telefonia), dove il *Bit rate* è costante e la *Connessione* è di tipo *connection-oriented* (**Classe A**).
  Su ogni cella viene effettuato l'incapsulamento *SAR* che prevede di riservare dal payload 1 byte in cui inserire i seguenti campi:
  - *SN* (Sequence Number), costituito da:
    - CSI (Convergence Sublayer Indicator): di 1 bit, individua i limiti dei blocchi di correzione nel Convergence Sublayer;
    - SC (Sequence Counter): di 3 bit, indica il numero della cella;
  - *SNP* (Sequence Number Protection), costituito da:
    - CRC: di 3 bit, è usato per il controllo ridondanza ciclica;
    - 1 bit di parità;
- **AAL 2**: utilizzato per quelle applicazione che richiedono una classe di servizio VBR-rt (real time VBR, cioè *Bit rate* variabile e *Timing* necessario) come il trasporto di audio-video compresso (**Classe B**).
  Nelle celle AAL 2 il payload non ha dimensione fissa; per ogni cella, la sua lunghezza è dichiarata in un campo apposito di 6 bit detto LI (Length Indicator);
- **AAL 3/4**: si rivolge ad applicazioni che richiedono una classe di servizio VBR-nrt (non real time VBR, cioè *Bit rate* variabile e *Timing* non necessario). Dal punto di vista della *Connessione*, può supportare sia la tipologia *connection-oriented* (**Classe C**), sia quelle di tipo

connectionless (**Classe D**), sebbene per queste ultime sia ormai stato soppiantato da AAL 5.
Applicazioni di queste classi prediligono l'integrità dei dati piuttosto che la costanza nel *cell delay*. I messaggi ricevuti vengono prima incapsulati tra appositi header e trailer, e solo dopo sono segmentati in celle.
Nelle celle di ALL 3/4 la parte di header deve fornire informazioni sul segmento, numero di sequenza e campo per il multiplexing; il trailer deve contenere l'indicatore di lunghezza ed il campo CRC. Il payload è quindi ridotto a 44 byte;

- **AAL 5**: è stato introdotto data la complessità di AAL 3/4, per:
  - ridurre l'*overhead* di elaborazione del protocollo;
  - ridurre il *sovraccarico* nella trasmissione;
  - garantire l'*adattabilità* ai protocolli di trasporto esistenti.

  Di contro, AAL 5 diventa *meno sicuro*, assumendo un comportamento simile a quello del sottolivello MAC dell'Ethernet: se il pacchetto consegnato non è valido, non si corregge ma viene automaticamente scartato.

Modello 3D Il modello di riferimento di ATM prevede anche 3 piani trasversali che, andando ad intersecarsi con i livelli visti in precedenza, costituiscono il cosidetto modello 3D. Questi piani sono:

- **Control Plane**: piano responsabile della generazione e gestione delle richieste di segnalazione;
- **User Plane**: piano responsabile della gestione del trasferimento dei dati utente;
- **Management Plane**: piano che non si interseca con i tre livelli ATM, ma si compone di due unità:
  - *layer management*: gestisce alcune funzioni come il rilevamento di guasti e problemi di protocollo;
  - *plane management*: gestisce e coordina le funzioni relative alla comunicazione tra i piani.

3 Liv. Rete

3.1 Tipologie di rete cablata

3.1.1 PAN

Personal Area Network: Rete personale che non si estende per più di 10-20 metri. Il termine si riferisce propriamente a reti con connessioni via cavo.

3.1.2 LAN

Local Area Network: Rete con un raggio limitato ad una abitazione o un edificio.

3.1.3 WAN

Wide Area Network: Rete che copre ampie aree geografiche connettendo tra loro più sottoreti locali.

3.1.4 MAN

Metropolitan Area Network: Rete metropolitana caratterizzata da una velocità di trasmissione molto elevata (tipicamente fibra ottica).

3.1.5 GAN

Global Area Network: Internet.

3.2 Tipologie di rete wireless

3.2.1 NFC

Near-Field Communication: Lo scambio di informazioni tramite tag elettromagnetici. Portata: 20cm.

3.2.2 BAN

Body Area Network: Rete che collega dispositivi indossabili, inferiore al metro.

3.2.3 WPAN

Wireless Personal Area Network: Rete di dispositivi personali in un raggio inferiore ai 20 metri.

3.2.4 WLAN

Wireless Local Area Network: LAN ottenuta tramite tegnologie wireless.

3.3 Topologia delle reti

3.3.1 Rete a dorsale

I dispositivi sono connessi tutti ad una via di trasmissione principale detta appunto dorsale. Un'interruzione in qualunque punto della dorsale compromette tutta la rete.

3.3.2 Rete ad albero

La trasmissione avviene in modo gerarchico tra i nodi padre e figlio, fino ad arrivare alla *root* dalla quale dipende tutto il funzionamento della rete.

3.3.3 Rete a stella

Tutti i dispositivi sono connessi ad un *hub* (router o swith di rete), più economico da sostituire in caso di rottura. Inoltre, in caso di danni ad un cavo, viene disconnesso solo un terminale. Questa topologia è tipicamente utilizzata per la realizzazione di reti LAN.

3.3.4 Rete ad anello

Le informazioni vengono passate da un dispositivo all'altro in modo ciclico, la trasmissione è unidirezionale anche se questo si può ovviare con un secondo anello in direzione opposta. Questa topologia era tipica delle LAN TokenRing, ora viene utilizzata principalmente nelle MAN in fibra ottica.

3.3.5 Rete a maglia

In inglese mesh, è una rete in cui ogni dispositivo può essere connesso ad ogni altro dispositivo ottenendo di fatto un grafo connesso. É la topologia di rete meno vulnerabile, ma è poco utilizzata nelle reti cablate a causa dei costi. É diffusa invece nelle WLAN, spesso nella versione *ad hoc*, dove i collegamenti nascono e muoiono dinamicamente. In questa topologia il Routing viene effettuato da ogni nodo.

3.3.6 9

Grid Rete di computer incentrata sulla condivisione dinamica delle risorse, nel contesto di calcolo distribuito e HTC (*High Troughput Computing*).
Rispetto ad un vero e proprio cluster di computer, la grid ha una composizione più eterogenea. La condivisione della capacità di calcolo non si limita al software, ma coinvolge anche l'hardware, grazie a librerie *middleware* (*software glue*) che si collocano tra il S.O. e lo strato fisico della macchina.

3.4 Internet Protocol (IP)

L'IP definisce l'esatto formato dei dati mentre attraversano l'internet TCP/IP. Svolge la funzione di **routing** e definisce regole.
L'unità fondamentale è il **datagram IP**, diviso in *header* (intestazione) e *data* (blocco dati). Nell'**header** sono presenti i seguenti campi:

- **VERS**: 4 bit, indica la versione dell'IP del datagram (IPv4 o IPv6)
- **HLEN**: 4 bit, indica la lunghezza dell'header in parole da 32 bit
- **Lunghezza totale**: 16 bit, indica la lunghezza totale del datagram in ottetti (incluso il blocco data). max=2 alla 16.
- **Servizio**: 8 bit, indica come deve essere gestito il datagram. E' diviso in 5 sottocampi:
  - 3 bit di **precedenza**, per specificare l'importanza del datagram (0-7)
  - 3 bit suddivisi in D, T, R per specificare il tipo di trasporto. D chiede un basso ritardo, T un alto throughput, R alta affidabilità
- **ID, FLAG, OFFSET**:controllano la frammentazione e il riassemblaggio del datagram a seguito dell'incapsulamento degli stessi in frame al liv. fisico.
- **TTL**: Time To Live, durata (s) concessa al datagram di restare in trasporto.
- **PROTOCOL**: indica quale protocollo di più alto livello ha generato la porzione Data
- **CHECKSUM**: garantisce l'integrità dell'header mediante CRC sii bit dell'header
- **SOURCE**: indirizzo IP a 32 bit dell'host che ha generato il datagram
- **DESTINATION**: indirizzo IP a 32 bit dell'host al quale è destinato il datagram
- **DATI**: dati trasportati dal datagram
- **OPTIONS** debugging
- **RIEMPIMENTO** area riempita dibit = 0 per garantire lunghezza multipla di 32 bit.

3.5 IPv4

Un IP su 32 bit (4 byte) identifica univocamente una rete ed un host appartenente alla rete `x.y.z.w`;
L'indirizzo è diviso in due parti: **host** e **rete**.
Esistono 5 classi di IP:

- **Classe A**: 0xxxxxxxx.y.z.w,
  subnet mask: 255.0.0.0 (parte rete=x, host=y.z.w)
- **Classe B**: 10xxxxxxx.y.z.w,
  subnet mask: 255.255.0.0 (parte rete=x.y, host=z.w)
- **Classe C**: 110xxxxxx.y.z.w,
  subnet mask: 255.255.255.0 (parte rete=x.y.z, host=w)
- **Classe D**: 1110xxxxx.y.z.w (multicast)
- **Classe E**: 11110xxxx.y.z.w (riservata).

NB: L'indirizzo IP indica la *connessione di un host alla rete*, non l'host in sè.

3.5.1 Indirizzi speciali

Gli IP possono far riferimento a *reti* o *host*.

- **indirizzo di rete**: IP in cui i bit della parte *host* sono tutti a 0, denota la rete stessa;
- **indirizzo di broadcast**: Se tutti i bit della parte *host* sono a 1, riservato a tutti gli host della rete;
- **default route**: 0.0.0.0
- **loopback address**: 127.0.0.1
- **Broadcast locale**: 255.255.255.255

3.6 Configurazione IP

Per configurare un host IP occorre specificare: Indirizzo IP, Subnet Mask, Default Gateway, IP del Nameserver; O si può utilizzare il protocollo **DHCP** il quale gestirà queste definizioni.

3.6.1 Subnet

Un'IP di una rete può essere gestito come un insieme di sottoreti introducendo una **subnet mask** più restrittiva, che assegna i *bit più significativi* della parte *host*, alla parte di *network*, ottenendo un insieme di sottoreti di classe (e dimensione) inferiore.
Il subnetting è effettuato per

- **ragioni topologiche**: superare i limiti di distanza, differenziare le connessioni di reti fisiche diverse, filtrare il traffico fra reti;
- **ragioni organizzative**: amministrazione, visibilità strutture, isolamento traffico;
- **ragioni tecniche**: ottimizzazione spazio indirizzamento IP, limitare dominio di broadcast, limitare effetti malfunzionamenti.

3.7 Protocolli di Address Resolution

Quando un pacchetto di livello 3 (Network) deve essere incapsulato in un protocollo di livello 2 (Data Link), questo deve inserire nell'header del pacchetto *l'indirizzo Data Link*.
Quindi per comunicare si deve conoscere l'**indirizzo fisico** dell'host di destinazione se questo appartiene *alla stessa rete del mittente*; O l'**indirizzo del Gateway** se l'host destinazione *appartiene ad un'altra rete*.

3.7.1 ARP

Address Resolution Protocoll: associa (risolve) la corrispondenza indirizzo ip (generalmente conosciuto) - indirizzo fisico di un host.
L'host A che vuole conoscere l'indirizzo fisico di B, invia un pacchetto broadcast contenente l'IP dell'host B, il quale rissponderà fornendo il suo indirizzo fisico.
In ogni macchina è presente una **cache** che salva gli indirizzi risolti via ARP per consultazioni successive (cache=*soft state - indirizzi possono diventare vecchi == timer di scadenza)*
L'host che effettua richiesta ARP via broadcast include il proprio indirizzo fisico, cosicchè tutti gli host possono aggiornare la propria cache.

3.7.2 RARP

Reverse Address Resolution Protocol: l'host spedisce la richiesta RARP ad un server mediante pacchetto broadcast con specificato indirizzo fisico, ed attende una risposta (che includerà l'IP relativo a quell'indirizzo fisico trasmesso mediante richiesta RARP).
Usato per workstation diskless (devono caricare il S.O. da un server ad ogni avvio).

3.8 Routing

Il routing è l'azione di scambiare informazioni in una rete da una sorgente ad una destinazione, incontrando almeno un nodo intermedio. Coinvolge due attività: *determinare il percorso ottimale di routing* e *trasportare gruppi di info (pacchetti) attraverso la rete.*
Metric: misura standard che indica il percorso ottimale da calcolare da parte di un protocollo di routing, calcolato considerando il *rapporto* **destination / next-hop**.
I routers comunicano tra loro e mantengono aggiornate le routing tables, medianti msg come *routing update* (aggiorna le tabelle) o *link-state advertisement* (informa che i routers usano il protocollo OSPF).

3.8.1 Algoritmi di routing

Devono essere: ottimali, semplici e con basso overhead, robusti e stabili, rapidi nella convergenza, flessibili.
Classificati in: statici/dinamici, single/multi-path, piatti/gerarchici, Host/Router-intelligent, Intra/inter-domain, **Distance-vector**[2]/**Link-stat**e[3].

3.8.2 Routing Table

I *gateway* instadano i dati tra diverse reti. Gli *host* prendono decisioni di instradamento. Il *protocollo IP* basa queste decisioni sulla parte *rete* dell'indirizzo IP.
L'host: determina la classe dell'IP, controlla la rete di destinazione (se è locale (sottorete) vi applica la *subnet mask*), cerca la rete di destinazione nella *routing table* , ed instrada i pacchetti al *gateway locale* per poi seguire il percorso indicato nella tabella. `netstat -nr`.
Il routing (e quindi la tabella di routing) può essere:

- **minimale**: aggiornamento tab routing effettuato al momento della definizione di una interfaccia
  `route add network subnet gateway`
  `route add 141.250.4.0 255.255.255.0 141.250.9.3`
- **statico**: instradamento gestito mediante info di routing predefinite e costanti
- **dinamico**: instradamento gestito via sw da protocolli di routing.

[2] Ad ogni riga della routing table corrisponde una rete; *periodicamente* i router raggiungibili si scambiano le istruzioni di routing attraverso le reti connesse; *il ricevente sostituisce le proprie istruzioni se quelle ricevute sono più ottimali*, le quali verranno propagata.
[3] Informazioni inviate in *broadcast*, permette ad *ogni* router di calcolare il cammino ottimale. Routers= nodi di un grafo connesso

3.8.3 Autonomous System

Le reti e gli IS (Intermediate System) si suddividono in **interni** ad un dominio di routing ed **esterni** ad un dominio di routing [4]. Il dominio di routing prende il nome di **AS** (Autonomous System), identificando la politica di routing adottata.
Un ISP (Internet Service Provider) può avere più AS, in questo caso distinti da un **ASN** (AS Number) che identificano univocamente le reti ai fini del routing. Vengono assegnati da ICANN.
GLi AS si dividono in :

- **Multihomed AS**: mantiene connessioni con più di un AS, per consentire ad un AS di mantenere la connessione alla rete anche in caso di malfunzionamenti di una delle connessioni
- **Stub AS**: connesso solamente con un altro AS; consente forme di peering privato tra AS.
- **Transit AS**: fornisce attraverso di sè connessioni con altre reti; La rete A può usare B appartenente ad un transit-AS per connettersi alla rete C. (Se un AS è un ISP per un altro AS, allora questi è un transit-AS).

I *router* che instradano messaggi all'interno dello stesso AS sono **Interior Router**, quelli che instradano anche tra AS diversi sono **Exterior Router**. I primi eseguono una famiglia di protocolli detta **IGP** (Interior Gateway Protocol), i secondi **EGP** (Exterior Gateway Protocol).

[4] insieme delle reti soggette all'amministrazione di una stessa organizzazione

3.9 Protocolli di Routing IGP

3.9.1 OSPF

Open Shortes Path First, protocollo standard per routing all'interno di un AS, è open source e basato sull'omonimo algoritmo di Dijkstra.
E' un **link-state routing protocol**, in quanto invia *link-state advertisements* (LSA) a **tutti** i routers di una stessa **area gerarchica**. Un router OSPF accumula LSA e calcola lo Shortest Path.
Quindi il routing può essere Inter-Area o Intra-Area. Il responsabile del routing Inter-Area è l'*OSPF Backbone*, che può essere anche spezzato e ricollegato tramite *virtual-links* (questo per definire topologie logiche diverse da quelle fisiche). OSPF distingue 4 tipi di router:

- **Internal Router**: interni ad un'area
- **Are Border Router**: che connettono 2 o più aree
- **Backbone Router**: appartenenti alla dorsale (area 0)
- **Border AS Router**: router di confine tra AS

OSP si basa sull'invio di pacchetti (router non adiacenti non scambiano informazioni):

- **Hello**: usato per scoprire i neighbors all'avvio del router, il designed router (DR) ed il backup designed router (BDR)
- **LS Update**: fornisce i propri criteri per la selezione del costo del link
- **LS ACK**: conferma un LS update
- **Database Description**: comunica gli aggiornamenti che conosce
- **LS Request**: richiesta di info di stato ai neighboars routers

OSPF Packet format
Header (24 Bytes): version number (1), Type (1), Length (2), Router ID(4), Area ID (4) Checksum (2), Autentication Type (2), Autentication (8);
Data (Variable).

3.9.2 RIP

Routing Information Protocol, implementato nel programma *routed*, si basa sull'algoritmo di Bellman-Ford (vettore-distanza), quindi è un **Distance-vector routing protocol**. Ha un limite di *15 hops*: reti più distanti sono irraggiungibili.
Un router RIP invia *tutta la routing table* o una porzione di essa ai *router vicini* (neighbors distanti 1 hop) ad intervalli di tempo.
Abbiamo due forme di RIP:

- **Attiva**: usata dai *routers*, invia in *broadcast* aggiornamenti periodici di routing ed usa i msg in arrivo per aggiornare la propria routing table.
- **Passiva**: usata dagli *hosts* (ma anche dai routers[5], usa i msg in arrivo per aggiornare la routing table, ma non invia aggiornamenti[6].

RIP è più propenso a generare *loops* rispetto a OSPF, ma ha bisogno di *meno risorse*, è più semplice da implementare ed è disponibile di default su Unix/Linux (**routed**)
RIP deve gestire 3 problemi:

- non rileva loop
- instabilità (risolto usando un numero basso di distanza max)
- problemi di convergenza lenta. Risolto adottando tecniche di:
  - *split horizon update* (un router non propaga info su un'altro che ha ricebuto questo msg)
  - *hold down* (il router ignora aggiornamenti inerenti a una rete dopo che ha ricevuto un msg di rete irraggiungibile),
  - *poison reverse* (se scompare un collegamento, questo gli verrà assegnato distanza infinita) + *triggered update* (i routers aggiornano la scomparsa immediatamente)

RIP1 Packet format: Command (1), Version number (1), Zero (2), Address Family ID (2), Zero (2), Address (4), Zero (4), Zero (4), Metric (4)
RIP2 Packet format: Command (1), Version number (1), Unused (1), Address Format ID (2), Route Tag (2), IP address (4), Subnet Mask (4), Next Hop (4), Metric (4)

[5] Quindi i roter possono essere attivi o passivi, gli host passivi
[6] sono del tipo (indirizzo destinazione, distanza)

3.10 Protocolli di routing EGP

3.10.1 BGP

Border Gateway Protocol, protocollo di inter-domain routing che mette in comunicazione routers appartenenti ad AS differenti (*gateway di confine*). BGP effettua

- **inter AS routing** tra due o più router BGC appartenenti ad AS diversi.
- **intra AS routing** tra due o più router iBGP appartenenti allo stesso AS
- **pass-through AS routing** tra due o più router BGP che scambiano attraverso un AS che non esegue BGP, il cui traffico non è destinato a nessun nodo interno a quell'AS.

Due router BGC formano una connessione TCP *(peer o neighbor routers)* e si scambiano prima di tutto le routing tables e per ogni rete il prossimo hop, dopodichè si scambiano *messaggi di gestione della connessione (nel campo Type)* (Open, Update, Notification, Keepalive, Refresh).
BGP Packet format: Marker[7](16), Length (2), Type (1), Data (variable).

3.11 ICMP

Internet Control Message Protocol, protocollo progettato per riportare anomalie durante il routing dei pacchetti e per verificare lo stato della rete. I vari tipi di messaggi ICMP sono:

codice	Messaggio
0	Echo Reply *(verifica raggiungibilità nodo)*
3	Destination Unreachable *(segnalazione anomalia)*
5	Redirect *(associa un router diverso da quello di default per un migliore instradamento)*
8	Echo Request *(verifica raggiungibilità nodo)*
11	Time Exceeded for a Datagram *(segnalazione anomalia)*
17	Address Mask Request *(Richiedi quale netmask è usata in una rete)*
18	Address Mask Reply *(Invia all'interfaccia che l'ha richiesta, la propria netmask)*

[7] valore riconosciuto da entrambi i peers per contrassegnare l'inizio del msg.

3.12 IP Multicasting

Trasmissione di un datagram ad un **gruppo di host** *(identificato da un IP di destinazione)*, che arriverà a tutti i suoio membri con la stessa affidabilità di un pacchetto *unicast* (best effort).
Un gruppo può essere permanente (IP statico) o transitorio. La gestione di questi ultimi è delegata ai *multicast agents* (entità che girano sui routers o su host dedicati).

3.12.1 IGMP

Internet Group Management Protocol: protocollo che supporta le funzioni di IP multicasting consentendo la gestione di sessioni multicasting, incluso l'invio di datagram a gruppi di host.

3.13 Dispositivi

Nel secondo livello OSI troviamo dispositivi in grado di riconoscere i dati organizzati in **frame**; Agendo su di esso, gestendoli ed instradandoli il tutto in modo trasparente.

3.13.1 Bridge

Un bridge *(ponte)* è un dispositivo di rete munito di porte con cui è collegato a diversi ***segmenti di rete*** (generalmente due o più LAN) da cui riceve dati e li instrada selettivamente verso una porta destinataria. Questi segmenti rappresentano i **domini di collisione**[8] della rete.

Tipologie In base alle tipologie delle due reti si distinguono:

- **Transparent Bridge**: collega 2 LAN Ethernet o IEEE 802.3
- **Source Routing Bridge**: collega 2 LAN Token Ring
- **Translational Bridge**: collega 2 LAN di tipo diverso (deve adattarsi alle diverse regole trasmissive)

3.13.2 Switch

Uno switch *(commutatore)* è un dispositivo molto simile al bridge, ma a differenza di quest'ultimo è collegato direttamente agli ***host*** ed ha un numero di porte nettamente superiore.

Implementazione LAN Una LAN può essere implementata con i seguenti apparati switch (switching LAN):

- 10baseT Switch *(IEEE802.3)*: ogni porta implementa un Dominio di Collisione separato
- 100baseT Switch *(IEEE802.3u)*: ogni porta implementa un segmento di rete FastEthernet usando un cavo *Unshielded twisted Pair (UTP)*
- 1000baseCX Switch *(IEEE802.3z, IEEE802.3ab)*: ogni porta implementa un segmento di rete Gigabit Ethernet usando un cavo in rame *Shilded Twisted Pair (STP)*
- 1GbaseCX4 Switch*(IEEE802.3ae, IEEE802.3ah)*: ogni porta implementa un segmento di rete 10 Gigabit Ethernet usando un cavo in rame *STP*

[8] insieme di nodi che concorrono per accedere allo stesso mezzo trasmissivo per successivamente trasmettere

Virtual LANs Implicito nella switching LAN vi è il concetto di Virtual LAN (**VLAN**), grazie al quale è possibile partizionare una rete locale basata su switch, in più *reti locali logicamente separate* tra di loro (ma che condividono la stessa rete fisica).

4 Liv. Trasporto

Porta : Per poter identificare il processo al quale destinare il datagram, si introduce il **portnumber**, un numero intero positivo che rappresenta diversi punti di destinazione astratti indirizzati dagli host internet per implementare diversi servizi.
Per comunicare con una porta esterna, l'host deve conoscere l'IP del destinatario e il numero di porta del protocollo usato da quell'host.

4.1 UDP

User Datagram Protocol: Fornisce un servizio di consegna *non affidabile e senza connessione*, utilizzando l'IP per trasportare i messaggi e permettendo di *distinguere tra più destinazioni all'interno di un stesso host.*
UDP si occupa quindi della differenziazione tra le varie provenienze e destinazioni all'interno di un singolo host

4.2 TCP

TRansmission Control Protocol: Fornisce un servizio di consegna *affidabile e con connessione*, isolando i programmi applicativi dal networking mediante un'interfaccia uniforme per il trasferimento.
L'interfaccia tra programmi e servizio di consegna affidabile del TCP/IP è basata su:

- **Orientamento allo stream**: il trasferimento dati di due programmi viene immagazzinato come sequenze di bit *(stream)* suddivisi in byte, passando dal mittente al destinatario l'esatta sequenza di **ottetti**.
- **Connessione di circuito virtuale**: trasferimento avviene solo quando mittente e destinatario hanno verificato la sussistenza delle condizioni di trasferimento.
- **Trasferimento bufferizzato**: mentre il programma genera un ottetto alla volta, il trasferimento accorpa insieme questi ultimi per ottimizzare la trasmissione. Se avviene in blocchi più grandi, il trasferimento avverrà in blocchi più piccoli.
- **Stream non strutturato**: il servizio di stream TCP non bada ad eventuali strutture presenti in strutture dati. QUesto lo dovrà fare il programma.
- **Connessione full-duplex**: consente il trasferimento simultaneo in entrambe le direzioni.

Il TCP garantisce *affidabilità* mediante il **riscontro positivo con ritrasmissione**.

Per ottimizzare la trasmissione, usa la **finestra scorrevole**: continua a trasmettere *stream* senza risconto fino a che ci si muove all'interno di una finestra predefinita di stream.
Anche il TCP usa il concetto di *porte*, che però identificano connessioni virtuali di circuiti. Per identificare una connessione usa l'insieme IP-Portnumber (=**Socket)** così da permettere la condivisione simultanea di un portnumber da parte di più hosts.

5 Liv. Sessione

6 Liv. Presentazione

7 Liv. Applicazione

7.1 Servizi di Rete

7.1.1 Telnet

Servizio di emulazione di un terminale a carattere ASCII attraverso la rete (basato su TCP). Effettua l'astrazione del terminale consentendo l'accesso remoto attraverso la rete.
Client: invocato dall'utente, realizza la connessione col *server remoto* e passa i caratteri digitali alla tastiera dall'utente al server, visualizzando l'output nel server.
Server: accetta connessioni di rete, passa i caratteri digitati dall'utente al S.O., come se fossero digitati da taastiera locale. Invia l'output al client.
Basato su 3 aspetti:

- **NTV**: Network Virtual Terminal, terminale virtuale; ogni client e server traduce i comandi nativi in quelli del NTV
- **Opzioni negoziate**: tra client e server aumenta funzionalità di telnet
- **Viste simmetriche**: fanno si che ai lati della comunicazione ci siano programmi invece di una tastiera e un monitor.

Attualmente sostituito con **SSH** (Secure SHell)

7.1.2 OpenSSH

Supporta i protocolli SSH1 e SHH2. La prima usa l'algoritmo di crittografia *RSA* per la negoziazione delle chiavi, la seconda usa **DSA** (Digital Signature Algorithm) e Diffie-Helmann. OpenSSH usa CRC nel primo caso e HMAC nel secondo.
L'insieme dei programmi di OpenSSH comprende: **ssh** (che sostituisce rlogin e telnet), **scp** (al posto di rcp), **sftp** (ftp), **sshd, ssh-add, ssh-agent, ssh-keygen e sftp-server.**

portforward Implementata da **ssh**, simile al NAT, permette di instradare connessioni TCP in un canale cifrato

7.1.3 Comandi r

rlogin Remote login, permette all'amministratore di configurare una serie di macchine in modo tale da permettere ad un utente con stesso ID di accedervi senza password. Poco sicura ma comoda.

rsh Remote shell, simile ad rlogin permette l'esecuzione remota di un singolo comando. L'esito di tale comando viene visualizzato nel terminale dell'utente.

7.1.4 DNS

Il DNS (*Domain Name System*) è un **servizio di rete** utilizzato per associare i nomi degli host, più semplici da ricordare per l'utente, ai relativi indirizzi IP.

Spazio dei nomi gerarchico basato su un insieme di database distribuiti, garantisce l'aggiornamento di tutta la rete. L'insieme dei nomi viene suddiviso in zone dette domini che possono coprire più host ed essere suddivise in sotto-domini e così via (struttura ad albero):

- I **domini di primo livello** o **TLD** (*Top Level Domain*) sono i figli del nodo radice ".", suddivisi in:
  - **gTLD** (*generic TLD*), ad esempio .com, .edu...
  - **ccTLD** (*country-code TLD*) ad esempio .it, .fr, .uk...),
  - **infrastrutturali**: .arpa, usato per la risoluzione inversa dei nomi.

  Essi sono assegnati da ICANN alle organizzioni o alle autorità responsabili locali .
- I **domini di secondo livello**, in genere, appartengono alle organizzazioni che li hanno registrati e comprendono il loro e il dominio precedente separati da un punto (es. unipg.it).
- I successivi **sotto-domini** vengono creati per rendere la gestione del DNS modulare e seguono la stessa logica dei domini di secondo livello.
- Infine le foglie corrispondono agli **host**.

Quindi, al contrario degli indirizzi IP, in un nome DNS la parte più importante è la prima partendo da destra (appunto, il TLD). Ad ogni dominio è associato un resource record (RR), file ASCII che contengono records del database DNS.

Server DNS per rendere disponibile lo spazio dei nomi e rispondere alle richieste del resolver risolvendo i nomi (name server)

Client DNS una libreria di funzioni per generare e inviare le richieste sui nomi, interrogando i server DNS (resolver)

Risoluzione La conversione di un nome in un indirizzo è detta *risoluzione*, mentre la conversione di un indirizzo in nome è detta *risoluzione inversa*. La risoluzione può essere **statica** (mapping stabilito permanentemente tramite una host table) o **dinamica** (mapping stabilito ad ogni avvio dell'host).

Funzionamento: Per eseguire la risoluzione il client chiama il risolutore, passandone come parametro il nome.
Questo invia un pacchetto UDP a un server DNS locale (*primary server*) che cerca il nome e se è presente nella sua cache lo restituisce; in caso contrario interroga ricorsivamente i server partendo da un root server del TLD fino ad arrivare ai server autorevoli del nome richiesto (*authoritative server*), i quali invieranno la loro risposta al client.

BIND *Berkeley Internet Name Domain* è l'implementazione più comune del DNS su ambiente Unix. BIND è composto da una parte client, il `resolver` (libreria che genera e invia le richieste al server) ed una parte server, `named` (demone che risponde alle richieste del resolver).
BIND può essere configurato come

- ***caching-only*** reindirizza ogni richiesta del resolver ad altri server e memorizza il risultato che ritornano in una cache locale
- **authoritative** contiene info su tutta la zona di sua competenza. Può essere
  - ***secondary***: scaricano gli Zone files dal *primary server* e li memorizzano in appositi file detti *zone file transfer*;
  - ***primary***: gestiscono le informazioni relative a specifici domini, salvate negli Zone files, confgurati dall'amministratore di rete.

Di seguito, le **Configurazioni** del **Resolver**, del **Named** e degli **Zone files** di BIND:

Resolver (/etc/resolv.conf): contiene istruzioni per l'esecuzione delle richieste; Si può usare la configurazione di default, altrimenti è necessario specificare:

- **nameserver <IP-address>**: le richieste saranno inviate all'IP *IP-address*. Si possono specificare al massimo 3 nameserver, nel caso il primo non risponda.
- **domain <name>**: nome del dominio di default che verrà concatenato a sinistra di ogni nome host che non contiene il carattere punto (in caso di fallimento omette i domini meno significativi fino a concatenare solo il TLD).

- **search <domain-1, ..., domain-n>**: come domain ma con la possibilità di avere più domini da provare ad aggiungere al nome host (ma non risale i domini se fallisce).

Named (Server): è necessario configurare più files:

- **/etc/named.conf**: parametri generali di configurazione e puntatori ai file dei domini gestiti dal server (ossia gli zone files)
  - caching-only: si omettono i comandi di configurazione del primary e secondary server tranne il dominio di loopback.
  - primary server
  - secondary server
- **/etc/named.ca**: puntatori ai root domain server. Stabilisce il nome dei root server e i loro indirizzi.
- **/etc/named.local**: zone file per la traduzione del reverse domain 0.0.127.IN-ADDR.ARPA (lookback). Permette quindi la conversione di 127.0.0.1 nel nome "localhost".
- **/etc/named.hosts**: zone file per la risoluzione diretta
- **/etc/named.rev**: zone file per la risoluzione inversa

Zone files File di testo che descrivono un sottoinsieme di domini (spesso un singolo dominio), ogni riga viene detta *Resource Record* (RR) ed è della forma: con:

- **name**: nome di dominio (in genere si usa @ per riferirlo al dominio definito nello zone file)
- **ttl (time to live)**: tempo di permanenza del RR nella cache di un sistema remoto
- **record class**: sempre `IN`, indica che il record è un INternet DNS RR
- **record type**: il tipo di RR (v. standard resource record)
- **record data**: info specifiche del tipo di RR

I principali componenti di uno zone file sono chiamati "standard resource record" e sono:

- **SOA** (Start of authority): segna l'inizio di un zone file (in genere è il primo record usato e ne esiste uno per zone file), definendo parametri specifici per questo zone file [data]

- **NS** (Name Server): nome del server *[record data]* che ha autorità su questo dominio *[name]*
- **A** (Address record): associa l'hostname *[name]* ad un indirizzo IP *[record data]*
- **PTR** (domain name PoinTR): associa gli indirizzi IP *[name]* ad un nome di host *[record data]*
- **MX** (Mail eXchanger): definisce il server *[record data]* che gestisce la posta per un host o un dominio *[name]*
- **CNAME** (Canonical NAME): definisce un alias per il nome di un host

Per avviare un tool di debugging (**dig**) si usa
`dig @server hostname`
e permette di interrogare un nameserver per ottenere informazioni e verificarne la configurazione.

7.1.5 NIS

Network Information Service, servizio che permette di definire risorse comuni ad un insieme di host, così da permettere agli utenti di spostarsi da un host all'altro mantenendo tutti i suoi dati principali *(login, home dir, autorizzazioni).*
Converte i file UNIX in un formato *database* **(NIS map)** che gode di un *controllo centralizzato* dei files amministrativi in un singolo server.

NIS map vengono creati dai seguenti files di sistema:

- **/etc/passwd**: per login, password, shell e home dir
- **/etc/group**: per i gruppi utenti
- **/etc/netgroup**: autorizzazioni host per l'accesso alle risorse locali
- **/etc/auto.home**: posizione assoluta della home directory
- **/etc/ethers**: info usate da RARP per ethernet
- **/etc/hosts**: usato da ARP
- **/etc/networks**: usato per mappare indirizzi di rete in nomi di rete
- **/etc/netmasks**: usato per definire la subnet mask
- **/etc/protocols**: nome protocollo, numero
- **/etc/services**: elenco servizi e relativa porta
- **/etc/aliases**: alias agli indirizzi email

e memorizzati nel **master server** che le rende disponibili ai client tramite il processo **ypserv**. I *client* aggiornano le loro info ricevendo i database con il demone **ypbind**.

7.1.6 NFS

Network File System, servizio che permette di condividere *directory e files* su una rete. Utenti e programmi possono accedervi da remoto come se fossero locali.
Client: l'inserimento di una directory di un *host remoto* nel proprio *filesystem locale* è detta **mounting**, realizzata con il comando `mount`.
Server: la condivisione di una directory *locale* ad host specifici per l'accesso *remoto* è detta **sharing**, realizzata con il comando `export`. Configurare il file `/etc/exports` sul server *nfsserver*, file contenente le dir da esportare e 'elenco degli host

Programmi NFS :

- **nfsd**[nservers]: demone che gestisce le *richieste NFS* (lato Server); *nserver*= numero di processi da eseguire
- **biod** [nservers]: demone che gestisce I/O dal lato Client
- **rpc.lockd**: gestisce i lock files (server e client)
- **rpc.statd**: controlla lo stato della rete (server e client)
- **rpc.mountd**: esegue le richieste mount del client (server)

NFS è implementato in tre parti indipendenti: NFS, RPC e XDR, per consentire l'uso di questi ultimi due anche da parte di protocolli e programmi.

RPC Remote Procedure Call.
Lato client viene incorporato in fase di compilazione nel codice delle procedure remote; *Lato Server* implementa le funzioni volute e incorpora le funzioni RPC.
Quando il client esegue una procedura remota, RPC invia il messaggio al server e memorizza i valori restituiti. Questo nasconde tutti i dettagli dei protocolli sottostanti, gestendoli in automatico.

XDR eXternal Data Representation.
Consente di scambiare dati tra macchine con architettura *eterogenea*, senza preoccuparsi di conversioni tra le diverse rappresentazioni dei dati. Usata su entrambi i lati della comunicazione, rende indipendente la rappresentazione hw.

7.1.7 NAT

Network Address Translation, servizio per la conservazione degli indirizzi IP. Viene eseguito da un router che connette due reti (*una privata e una pubblica)*, traducendo gli IP della *rete privata* in un indirizzo **IP pubblico** prima che il pacchetto venga inoltrato.
NAT svolge quindi la doppia funzione di *implementare la sicurezza della rete locale* e *preservare gli indirizzi IP*.
L'ordine delle operazioni di routing e traduzione dipende dal tipo di flusso:

- **da Internet a LAN**: viene eseguita prima la traduzione degli IP e poi il routing del pacchetto
- **da LAN a Internet**: viene prima effettuato il routing del pacchetto e poi la traduzione degli indirizzi

7.2 Protocolli livello Applicazione

7.2.1 FTP

File Transfer Protocol, protocollo per il trasferimento di file tra host in una rete TCP/IP. Essendo basato sul TCP è *orientato alla connessione* ed è *affidabile*.
Ogni trasferimento FTP è composta da due processi:

- **Protocol Interpreter** che si occupa di trasmettere comandi fra client e server FTP e *dà inizio al processo FTP sulla porta 21*
- **Data Transfer Process** (DTP) che si occupa del trasferimento vero e proprio tra client e server FTP

Client: contatta il server, specifica i files e la direzione del trasferimento (up/down-load)
Server: Mantiene i files nel disco locale e rimane in attesa di richieste e le serve. Per il DTP il client ed il server si scambiano di ruolo,in quanto il *client* crea il processo di gestione dei files, alloca la porta ed invia il suo numero al server, ed attende richieste. Il *server* riceve le richieste, crea il processo per gestire il trasferimento dati e contatta il client.
Il traffico attraversa la rete in chiaro *(come in telnet)*, si usa **scp** (Secure Copy).
Per LAN si usa **TFTP** (Trivial FTP), copia file interi, eseguito usando UDP, ha minori funzionalità ma è più leggero.

7.2.2 SNMP

Simple Network Management Protocol[9], protocollo di gestione di reti, applicazioni e sistemi, permettendo agli amministratori di *indirizzare richieste e comandi ai nodi della rete e monitorare le risorse.* Basato su *TCP/IP.* Composto da:

- **Agent**: Nodi gestiti, dispositivo che risponde alle richieste del Manager
- **Manager**: Stazione di gestione,programma che interroga e invia comandi all'Agent
- **MIB** (Management Information Base): Informazioni di gestione, archivio di info di gestione immesse dagli agent, detti *oggetti.*
- **SNMP**: Protocollo di gestione, definisce le modalità di interazione tra Manager e Agent
- **SMI**: definisce la struttura degli *oggetti.* Specificato con l'**ASN.1** (Abstract Syntax Notation, Standard ISO). *Object IDentifier* contiene i criteri per definire un oggetto, seguendo una *struttura ad albero.*

[9]SNMPv1, v2 non sono sicuri, v3 sì (quella attuale)

7.2.3 DHCP

Dynamic Host Configuration Protocol[10], protocollo che fornisce supporto per lo scambio di *informazioni di configurazione* tra host di una rete TCP/IP.
Composto da *un protocollo* per la trasmissione dei parametri di configurazione da un DHCP server all'host, e di un *meccanismo* per assegnare gli indirizzi di rete agli host (**BOOTP**, usa UDP).
L'assegnamento degli IP prevede 4 fasi:

1 **Discovering**: il client invia un msg *broadcast* **DHCP Discover** per richiedere l'assegnamento ad un server

2 **Offering**: Un qualunque server invia un msg **DHCP Offer**, con un indirizzo disponibile.

3 **Requesting**: il client invia *broadcast* un msg **DHCP Request** per comunicare quale offerta ha accettato

4 **Acknowledgment**: il server invia un msg **DHCP Acknowledgment** per conferma dell'assegnamento dell'indirizzo. In caso di errori invia **DHCP NACK**

L'assegnamento può essere allocato: **automaticamente** (ip permanente), **dinamicamente** (ip valido per un certo periodo di tempo, permettendo il riuso di ip non più usati), **manualmente** (dall'amministratore, il DHCP è usato solo come mezzo di comunicazione).

[10]NON è utilizzato per configurare i router

7.3 Posta Elettronica

La posta elettronica è un servizio che coinvolge due programmi: **Mail User Agent** (interfaccia utente verso l'applicazione che svolge funzioni di *Composizione, Visualizzazione d Eliminazione*) ed il **programma di trasporto** (es. **Sendmail**, svolge le restanti funzioni di *Trasferimento, Notifica utente*) **Sendimail** riceve e spedisce posta con **SMTP** e fornisce alias di posta.
Il programma **popper** (**POP3**) consente di interagire con il *programma di trasporto* direttamente da un pcc in rete. Anche **IMAP**, ma è più sicuro e veloce.
Gli *indirizzi di posta elettronica* sono del tipo **user@host.domain** o **user@domain**.
È impoortante definire anche un *dominio secondario* in caso di malfunzionamenti (**Mail Relay**[11]).

MIME Multipurpose Internet Mail Extentions, standard di codifica introdotto per le nuove esigenze; Usa una codifica **base 64**, procude linee lunghe max 76 caratteri

7.4 Protocolli Posta Elettronica

7.4.1 SMTP

Simple Mail Transfer Protocol, protocollo per il trasporto[12] affidabile ed efficiente dei messaggi di posta elettronica.
Una richiesta di posta del client causa l'attivazione di un *canale bidirezionale* tra server SMTP del trasmettitore e quello del ricevente (il quale può essere destinatario o intermediario). Il canale serve per lo scambio di comandi SMTP.

Comandi SMTP in ordine di utilizzo:

1 **HELO**: primo comando da inviare, può anche essere riimmesso dopo.

2 **NOOP, HELP, EXPN, VRFY**: usati ovunque

3 **MAIL, SEND, SOML, SAML**: iniziano una transazione di mail. Seguiti da uno o più comandi **RCPT** (destinatario mail) e dal comando **DATA** (in ordine). Dopo l'immisiione del *messaggio* deve seguire la sequenza **¡CRLF¿.¡CRLF¿**. Connessione abortita con **RSET**

4 **QUIT**: ultimo comando, di chiusura.

[11] funzione di SendMail, usata però abusivamente dagli spammers per inviare posta spam
[12] un servizio di trasporto fornisce un **inter processo communication environment IPCE**; l'invio di mail consiste nello scambio di dati tra due IPCE.

7.4.2 POP3

Post Office Protocol, protocollo che gestisce la comunicazione client-server; In particolare permette al *client* di ricevere e cancellare messaggi sul server SMTP, mentre l'invio avviene via SMTP.
Il *client* interagisce con il *server POP3*, il quale dialoga con il *server SMTP*.
La sessione attraversa 3 *stadi*:

0 il client chiede il servizio, apre una connessione TCP/IP e il server invia un msg di benvenuto

1 fase di **AUTHORIZATION**, in cui il client fa il login in chiaro.

2 fase di **TRANSACTION**, in cui il client richiede azioni al server

- **Comandi**: **STAT** (numero di msg e dimensione in byte), **LIST [msg]** (elenca msgID e dimensione dei msg), **RETR msg** (riceve msg), **DELE msg** (marca msg come cancellat), **NOOP** (non fa nulla), **LAST** (indica il più alto msgID ricevuto), **RSET** (rimuovi la marcatura cancellati dei msg).

3 fase di **UPDATE** in cui vengono rilasciate le risorse acquisite e si chiude la sessione.

- **Comandi**: **QUIT** (chiude la connessione)

7.4.3 IMAP

Internet Mail Access Protocol, protocollo per l'accesso alle mailbox ed alle news di server centrali da client.
IMAP associa ai messaggi dei *System Flag* che ne definiscono alcuni attributi:

- **\Seen**: messaggio letto
- **\Answered**: messaggio risposto
- **\Flagged**: messaggio marcato come importante/urgente
- **\Deleted**: messaggio marcato come cancellato, per successiva rimozione con **EXPUNGE**
- **\Draft**: messaggio marcato come bozza
- **\Recent**: messaggio recentemente arrivato nella mailbox

7.4.4 HTTP

HyperText Transfer Protocol, protocollo che permette la trasmissione di informazioni sul Web, al fine di realizzare sistemi informativi distribuiti, collaborativi ed *ipermediali* (ossia composti da *multimedialità* distribuita nella rete ed acceduta mediante *hyperlinks* [13]).
Utilizza il protocollo TCP sulla porta 80.

Versioni HTTP è utilizzato dal WWW dal 1990, e da allora è stato aggiornato a diverse versioni:

- **HTTP/0.9**: semplice protocollo per il trasferimento di dati grezzi sulla rete Internet, gestito dal W3C;
- **HTTP/1.0**: pur consentendo il trasferimento di messaggi *MIME*, non era adatto a supportre la crescita esponenziale del WWW;
- **HTTP/1.1**: versione consolidata del protocollo, usato per 15 anni;
- **HTTP/2**: nuovo standard basato sul protocollo SPDY/2 di Google.

Funzionamento L'HTTP ha un'architettura di tipo client/server *stateless*[14] e comprende due tipi di messaggi: di richiesta e di risposta.

- **HTTP Request** inviato dal client verso un server, è composto da:
  - **request line**: riga di richiesta composta da
    - metodo di richiesta *(GET, POST, HEAD, PUT, DELETE, TRACE, OPTIONS, CONNECT)*
    - l'*URI* (indica la risorsa richiesta, es pagina web)
    - la *versione del protocollo.*
  - **header**: informazioni aggiuntive, tra cui l'Host *(nameserver a cui si riferisce l'URL)* e l'User-Agent *(tipo di client)*
  - riga vuota
  - **body**: corpo del messaggio
- **HTTP Response** inviato dal server, è di tipo testuale ed è composto da:
  - **status line**: riga di stato che riporta un codice a tre cifre per identificare lo stato della risposta *(1xx Informational, 2xx Successful, 3xx Redirection, 4xx Client error, 5xx Server error)*

[13] Collegamenti ipertestuali.
[14] Un protocollo di comunicazione stateless non salva informazioni sulla sessione e quindi interpreta ogni messaggio indipendentemente dagli altri.

- **header**: Server *(indica il tipo di server)*, Content-Type *(il tipo di contenuto restituito in codifica MIME, es. text/html, text/plain...)*
- riga vuota
- **body**: contenuto della risposta

HTTPS Adattamento dell'HTTP per comunicazioni sicure, utilizza un protocollo crittografico a doppia chiave. Esiste in due varianti che differiscono per il protocollo crittografico utilizzato:

- **SSL** (Secure Sockets Layer): ogni server deve avere un certificato X.509, emesso da una *certificate authority* (CA) e contenente:
  - Chiave pubblica;
  - Il *distinguished name* del server (nome ed indirizzo);
  - Numero di serie o data di pubblicazione dei certificato;
  - La data di fine validità del certificato.
- **TLS** (Transport Layer): è l'evoluzione di SSL. Si articola in tre fasi:
  1. Negoziazione degli algoritmi da utilizzare;
  2. Scambio delle chiavi;
  3. Autenticazione;
  4. Cifratura simmetrica.

7.5 Sicurezza di Rete

La sicurezza di rete è una branca della sicurezza informatica che nasce nel momento in cui si hanno più computer interconnessi fra loro in quanto presentano diverse vulnerabilità sfruttabili da intrusori per avviare diversi tipi di attacchi.

7.5.1 Minacce

Ogni host connesso ad una rete è sottoposto ad un lungo numero di minacce, quali **Accessi non autorizzati**, **Accesso ad informazioni** o **Negazione di servizi**('utente autorizzato non riesce più ad accedere alle risorse della rete).
Esistono diversi modi per proteggere una rete da questi ed altri attacchi, come l'oscuramento, l'hardening e il firewalling.

7.5.2 Oscuramento

Per oscuramento si intende garantire la sicurezza nascondendo le risorse di rete mediante strumenti come NAT, IP Masquerading, SSH (per le connessioni), GPG (per la validazione di utenti e servizi) e trasmissioni crittografate.

Encryption limita gli accessi ai dati trasmessi sulle reti: il contenuto viene crittografato, spedito e in ricezione de-crittografato.
Si usano i comandi UNIX `crypt` e `des`.

IP Masquerating grazie al NAT è possibile utilizzare un singolo IP privato per affacciare una intera sottorete all'esterno (internet).

7.5.3 Hardening

Per hardening si intende la gestione della sicurezza *a livello del singolo host*;

TCP-wrapper consentono di limitare l'accesso ai servizi basandosi sull'IP e l'hostname del client.
Le richieste verso l'host vengono elaborate dal wrapper, che verifica che l'indirizzo del chiamante sia incluso nell'elenco di ***/etc/hosts.allow***: se è così permette l'accesso, altrimenti verifica non sia incluso in ***/etc/hosts.deny***.

xinetd demone che estende le funzionalità di *inetd*, infatti non gestisce solo l'accesso ai servizi di rete ma controlla anche i servizi stessi a Livello Applicazione.

7.5.4 Firewall

Sistema hardware e/o software che costituisce un intermediario tra la rete locale (o singolo host) ed una o più reti esterne (tipicamente internet), filtrando il traffico.
Nella rete interna ci saranno i servizi di base (come NFS, NIS etc.) rivolti agli utenti della sottorete interna; nella rete esterna i servizi di networking (ad esempio DNS, SMPT, FTP etc.), rivolti sia ad utenti interni che esterni e dunque esposti a rischi.
Nonostante permetta di isolare la rete dal mondo esterno, *non è in grado di proteggerla da attacchi* interni o condotti da linee da lui non controllate. Si distinguono due tipi di firewall:

- firewall **stateless**, che prendono decisioni basate esclusivamente sulla specifica connessione. `ipchains`;
- firewall **stateful**, tengono traccia delle connessioni e prendono dunque decisioni intelligenti. `iptables`, che sostituisce ipchains.

Packet filtering Metodo con il quale un firewall limita il traffico di rete. In Linux, ***NetFilter*** è il filtro di pacchetti implementato nel kernel, la cui interfaccia è rappresentata da `iptables`.
`iptables` contiene alcune **TABLES**, che rappresentano le liste di regole di filtraggio. Queste contengono le **CHAINS** (catene in cui i pacchetti vengono controllati), che a loro volta sono formate da molte **Rules** (composte da un campo **match** e uno **target**).
Ogni pacchetto attraversa almeno una catena, di cui ogni regola controlla (proceduralmente) se il pacchetto soddisfa il campo *match*: se così accade, gli viene applicato il **target**. I principali sono **ACCEPT** viene accettato ed instradato, **REJECT** se lo rifiuta ed avvisa il mittente con un errore o **DROP** se lo blocca e lo scarta senza notifica.
NetFilter è quindi in grado di stabilire il **contesto** del pacchetto: **NEW** (appartiene ad una nuova connessione), **ESTABLISHED** (ad una connessione esistente), **RELATED** (legato ad una connessione esistente) o **INVALID** (pacchetto sospetto, non legato ad alcuna connessione stabilita).