Università degli Studi di Perugia
Laurea in Informatica

Laboratorio di Reti di computer

Sergio Tasso

 sergio@unipg.it

Laboratorio Reti di Computer

Scopo del corso

- Fornire conoscenze ed esperienze sulla gestione ed il controllo delle reti di computer.

L'argomento

- metodi di gestione con particolare riferimento ad ambienti basati sui sistemi operativi UNIX e WINDOWS/NT
- installazione e configurazione dei principali servizi di rete
- installazione e configurazione di apparati di instradamento (router e gateway)
- panoramica sui prodotti di controllo e monitoraggio
- considerazioni sui sistemi di sicurezza

Laboratorio Reti di Computer

Programma del corso

- Introduzione alla gestione delle reti
- Il modello TCP/IP
  - livello IP (Internet Protocol)
  - livello TCP (Transport Control Protocol)
- Configurazione del TCP/IP
  - Come definire sottoreti
  - Definizione delle interfacce
- La configurazione del Routing:
    - ⋆ minimale
    - ⋆ statico
    - ⋆ dinamico

Laboratorio Reti di Computer

- Servizi di rete (definizione, configurazione ed uso):
  - La gestione dei nomi:
    - la Host table
    - il DNS (Domain Name Service)
  - L'accesso a risorse in rete :
    - comandi remoti
    - il servizio NIS (Network Information Service)
    - il sistema NFS (Network File System)
  - La posta elettronica:
    - gli aliases
    - il servizio sendmail

Laboratorio Reti di Computer

- Servizi di controllo e gestione:
  - SNMP (Simple Network Management Protocol)
    - definizione del protocollo
    - MIB (Management Information Base)
    - applicazioni di gestione
      - monitoraggio
      - definizione degli allarmi
      - trattamento degli eventi
  - Altri comandi e tools di diagnostica
- Sicurezza delle reti:
  - Criteri di valutazione dei rischi
  - Strumenti per il controllo dei rischi
  - Limitazione degli accessi
    - TCP wrapper
    - Firewall
  - Strumenti di monitoraggio e IDS
- Esempi ed esercitazioni su tutti gli argomenti trattati

Libri di testo

Gary Govanus – *TCP/IP* *(Per configurare, implementare e gestire TCP/IP sulla vostra rete)* – Handbook Informatica Professionale McGraw-Hill

Craig Hunt - *TCP/IP Network Administration* - O'Reilly & Associates, Inc.

Dispense del docente

TCP/IP

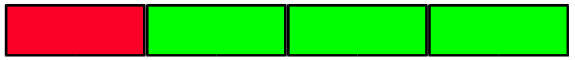
- Concetti di base sul protocollo TCP/IP
- Configurazione del software di gestione reti su computer con sistema operativo Unix
- Configurazione di alcuni dei più importanti servizi di rete
- Diagnostica e sicurezza

IP address

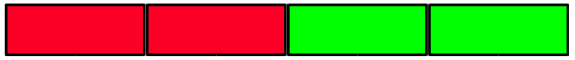
- Un indirizzo IP su 32 bit permette di identificare univocamente una rete ed uno specifico host appartenente alla rete
- L'indirizzo si divide in due parti:
  - rete
  - host
- Esistono 4 tipi di classi di IP address.

Classi di indirizzi IP (x.y.z.q)

- Classe A: con x < 128



- Classe B: con 128 <= x <= 191



- Classe C: con 192<= x <= 223

- Classe D: con x > 223

Indirizzi riservati

- Livello rete
  - Classe A:
    - 0.y.z.q (default route)
    - 127.y.z.q (loopback address)
- Livello host
    - x.0.0.0 e x.255.255.255 per la classe A
    - x.y.0.0 e x.y.255.255 per la classe B
    - x.y.z.0 e x.y.z.255 per la classe C

    che identificano rispettivamente la rete stessa e il "broadcast address"

Nota

Gli indirizzi IP vengono assegnati alle interfacce di rete non agli host!

E’ quindi errato, anche se comune, chiamare gli indirizzi IP “indirizzi degli host”.

Infatti un computer che funziona da gateway ha diversi indirizzi per ogni rete che collega, e quindi viene indirizzato in maniera diversa da singoli dispositivi in base alla rete su cui questi risiedono.

Sottoreti

- un IP address può essere localmente modificato usando parzialmente i bit relativi agli host come bit di indirizzo per sottoreti.
- Una sottorete è definita applicando all'indirizzo IP una maschera di bit chiamata subnet mask.
  - Se un bit su una subnet mask è a 1:
    - l'equivalente indirizzo IP è interpretato come un network bit (cioè appartiene ad un indirizzo di rete)
  - Se un bit nella maschera è a 0:
    - il relativo bit nell'indirizzo IP fa parte di un indirizzo di host (host bit)

Sottoreti (effetti della subnet mask)

Classe	IP address	Subnet mask	Interpretazione
B	128.66.12.1	255.255.255.0	host 1 della sottorete 128.66.12.0
B	130.97.16.132	255.255.255.192	host 4 della sottorete 130.97.16.128
C	192.178.16.66	255.255.255.192	host 2 della sottorete 192.178.16.64
B	132.90.132.5	255.255.240.0	host 4.5 della sottorete 132.90.128.0
A	18.20.16.91	255.255.0.0	host 16.91 della sottorete 18.20.0.0

Sottoreti: altro modo di scrivere la subnet mask (x.y.z.q/s)

S identifica il numero di 1 nella subnetmask

Classe	IP address	Subnet mask	Interpretazione
B	128.66.12.1/24	255.255.255.0	host 1 della sottorete 128.66.12.0
B	130.97.16.132/26	255.255.255.192	host 4 della sottorete 130.97.16.128
C	192.178.16.66/26	255.255.255.192	host 2 della sottorete 192.178.16.64
B	132.90.132.5/28	255.255.240.0	host 4.5 della sottorete 132.90.128.0
A	18.20.16.91/16	255.255.0.0	host 16.91 della sottorete 18.20.0.0

Tabella di routing (1)

- I gateway instradano i dati tra diverse reti
- Gli host prendono decisioni di instradamento nel modo seguente:
  - Se l'host di destinazione è sulla rete locale, i dati vengono spediti all'host di destinazione;
  - Se l'host di destinazione è su una rete remota, i dati vengono mandati al gateway locale.
- Il protocollo IP basa le sue decisioni di instradamento su parte rete dell'indirizzo IP

Tabella di routing (2)

- Analisi dell'indirizzo IP fatta dall'host:
  - determina il tipo di classe dell'indirizzo IP (bit più significativi)
  - controllo rete di destinazione, se è locale (sottorete) applica all'indirizzo di destinazione la subnet mask
  - cerca la rete di destinazione nella routing table (altrimenti detta forwarding table)
  - instrada i pacchetti di dati seguendo l'instradamento dato dalla tabella di routing.

Tabella di routing (3)

- In ambiente Unix il comando per visualizzare la tabella di routing è:

  netstat -nr

  dove -r è l'opzione per la tabella

  n è per vederla in modo numerico

Tabella di routing (4)

Routing tables

```
Destination     Gateway         Flags   Refcnt  Use     Interface
127.0.0.1       127.0.0.1       UH      1       298     lo0
default         128.66.12.1     UG      2       50360   en0
128.66.12.0     128.66.12.2     U       40      98379   en0
128.66.2.0      128.66.12.3     UG      4       1179    en0
128.66.1.0      128.66.12.3     UG      10      1113    en0
128.66.3.0      128.66.12.3     UG      2       1379    en0
128.66.4.0      128.66.12.3     UG      4       1119    en0
```

dove:

Destination = rete od host di destinazione

Gateway = gateway da usare per la specifica destinaz.

Flags:

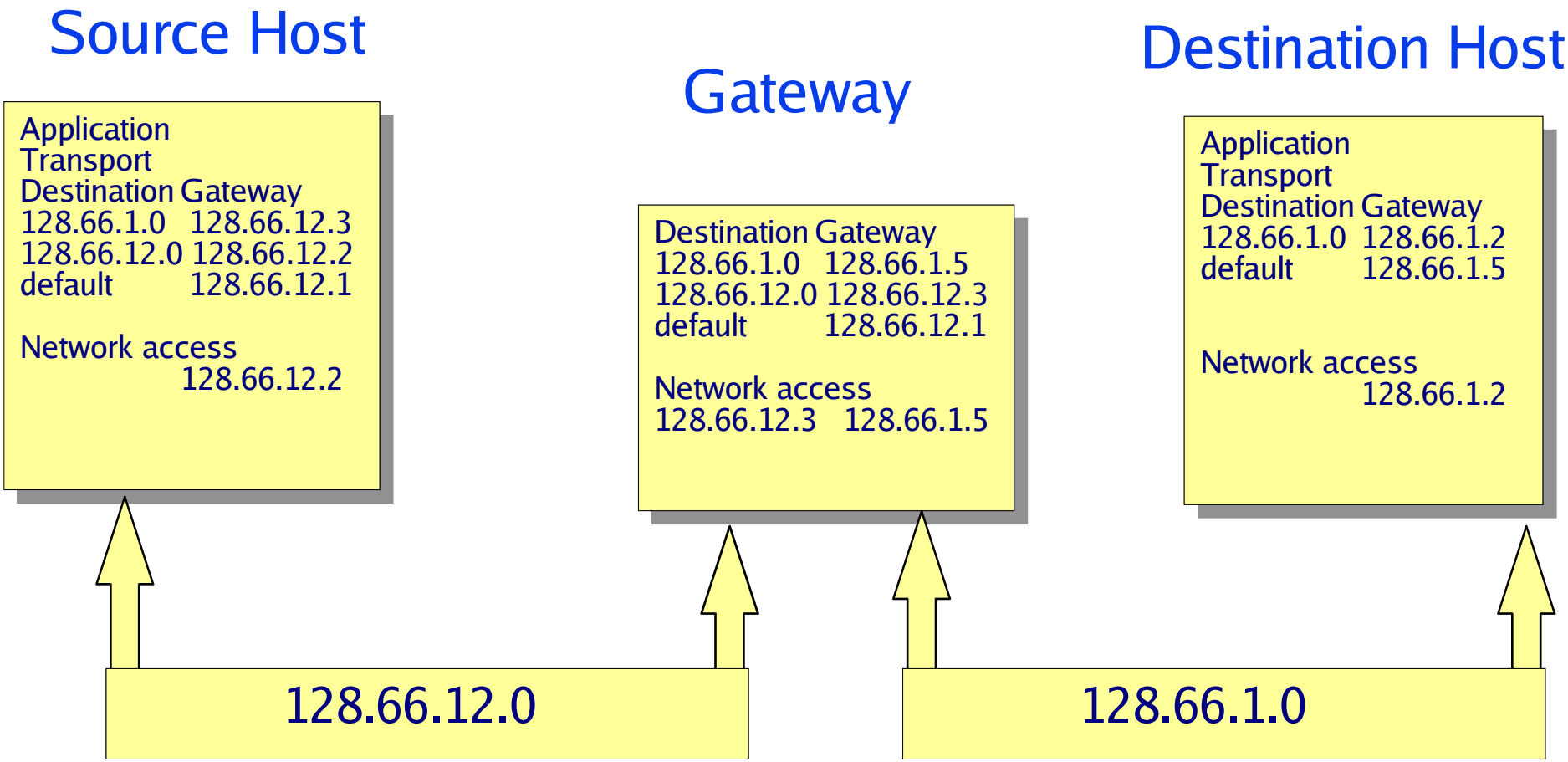
- U = route in linea e attiva
- H = route per host spec. (non per una rete)
- G = route che usa un gateway
- D = route aggiunta da una ICMP redirect

Refcnt= n.ro di volte che la route è stata usata

Use = n.ro pacchetti trasmessi su quella route

Interface= nome dell'interf. di rete usata per la route

Tabella di routing (5)



Address Resolution Protocol (ARP)

- Un host Internet puo' comunicare con un altro host solo se conosce il suo indirizzo fisico di rete (ad es.: l'indirizzo ethernet) che e' anche fondamentale per il funzionamento dei gateway al fine di definire il percorso dei pacchetti.
- I programmi applicativi in genere conoscono solo il ***nome*** dell'host o il suo indirizzo IP
- ***Risoluzione dinamica e a basso livello degli indirizzi***: ARP
- L'host che ha bisogno dell'indirizzo fisico di un altro host della rete, invia un pacchetto broadcast di richiesta dell'indirizzo al destinatario, il quale risponde includendo l'indirizzo di rete
- Esiste in ciascuna macchina una ***cache*** che memorizza gli indirizzi richiesti via ARP
- Il legame tra indirizzo IP e indirizzo fisico del trasmettitore e' inclusa in ogni richiesta ARP trasmessa in broadcasting; i riceventi aggiornano i dati in cache

Address Resolution Protocol (ARP) (II)

- L'ARP e' un protocollo a basso livello che nasconde il tipo di indirizzamento fisico usato nella rete, consentendo all'utente di usare direttamente gli indirizzi IP scelti per le macchine.
- E' composto funzionalmente da:
  1) determinazione degli indirizzi fisici quando si trasmette un pacchetto
  2) risposta a richieste ARP di altre macchine
- Nel primo caso occorre considerare che il meccanismo di richiesta via broadcast puo' generare problemi (metodo best-effort di ethernet, errori hardware, etc)
  1) aggiornamento dei dati in cache allo scadere di un *Time To Live* (TTL)
  2) operativita' degli altri protocolli in presenza di richieste ARP pendenti

Reverse Address Resolution Protocol (RARP)

- Le workstation diskless hanno bisogno di caricare il sistema operativo e la configurazione da uno o piu' server mediante effettuazione di una richiesta broadcast che utilizza il protocollo TCP/IP denominato RARP
  - l'host spedisce la richiesta RARP al *server* ed attende da questo una risposta (IP address)
  - Se la richiesta è ripetuta, rispondono anche i server secondari
- L'identificatore unico e' l'indirizzo fisico della macchina
- Richiesta e risposta ARP differiscono per il campo *tipo* della frame
- Il server nel rispondere scambia gli indirizzi mittente-destinatario, cambia il contenuto del campo tipo e trasmette l'indirizzo IP

Risoluzione degli Indirizzi Address Resolution Protocol (ARP)

- Tabella di traduzione degli indirizzi IP in indirizzi Ethernet
- In ambiente Unix la tabella può essere visualizzata con il comando:
  - arp -a

```
peanut.nuts.com  (128.66.12.2) at 8:0:20:0:e:c8
pecan.nuts.com  (128.66.12.3) at 8:0:20:1:77:fe
walnut.nuts.com  (128.66.12.4) at 0:0:1d:0:bc:bb
```

Numeri di protocollo (1)

- Il numero di protocollo è un byte presente nella 3.a parola del datagram header che specifica a quale tipo di protocollo sono passati i dati dal livello IP
- In Unix i tipi di protocollo sono definiti nel file /etc/protocols, visualizzabile con il comando:
  - `cat /etc/protocols`

Numeri di protocollo (2)

```
# Internet (IP) Protocols #
ip      0       IP      # internet protocol, pseudo protocol number
icmp    1       ICMP    # internet control message protocol
igmp    2       IGMP    # internet group multicast protocol
ggp     3       GGP     # gateway-gateway protocol
tcp     6       TCP     # transmission control protocol
pup     12      PUP     # PARC universal packet protocol
udp     17      UDP     # user datagram protocol
```

dove: la prima colonna riporta il nome ufficiale del protocollo,
la seconda colonna il numero di protocollo,
la terza eventuali aliases (nome maiuscolo)
dopo il carattere # commenti o nomi estesi

Numeri di porta (1)

- Dopo che IP ha passato i dati al protocollo di trasporto, quest'ultimo passa i dati al processo dell'applicazione di rete per una corretta elaborazione.
- I vari processi di applicazioni di rete (chiamati Network Services) vengono identificati da numeri di porta espressi su 16 bit.
- Esistono due tipi di numeri di porta:
  - Source Port Number identifica il processo che invia i dati
  - Destination Port Number identifica i processo che riceve i dati

  Entrambi sono contenuti nel TCP segment (prime due parole)

Numeri di porta (2)

- Per vedere i numeri di porta definiti in un ambiente Unix dare:
  - `cat /etc/services`

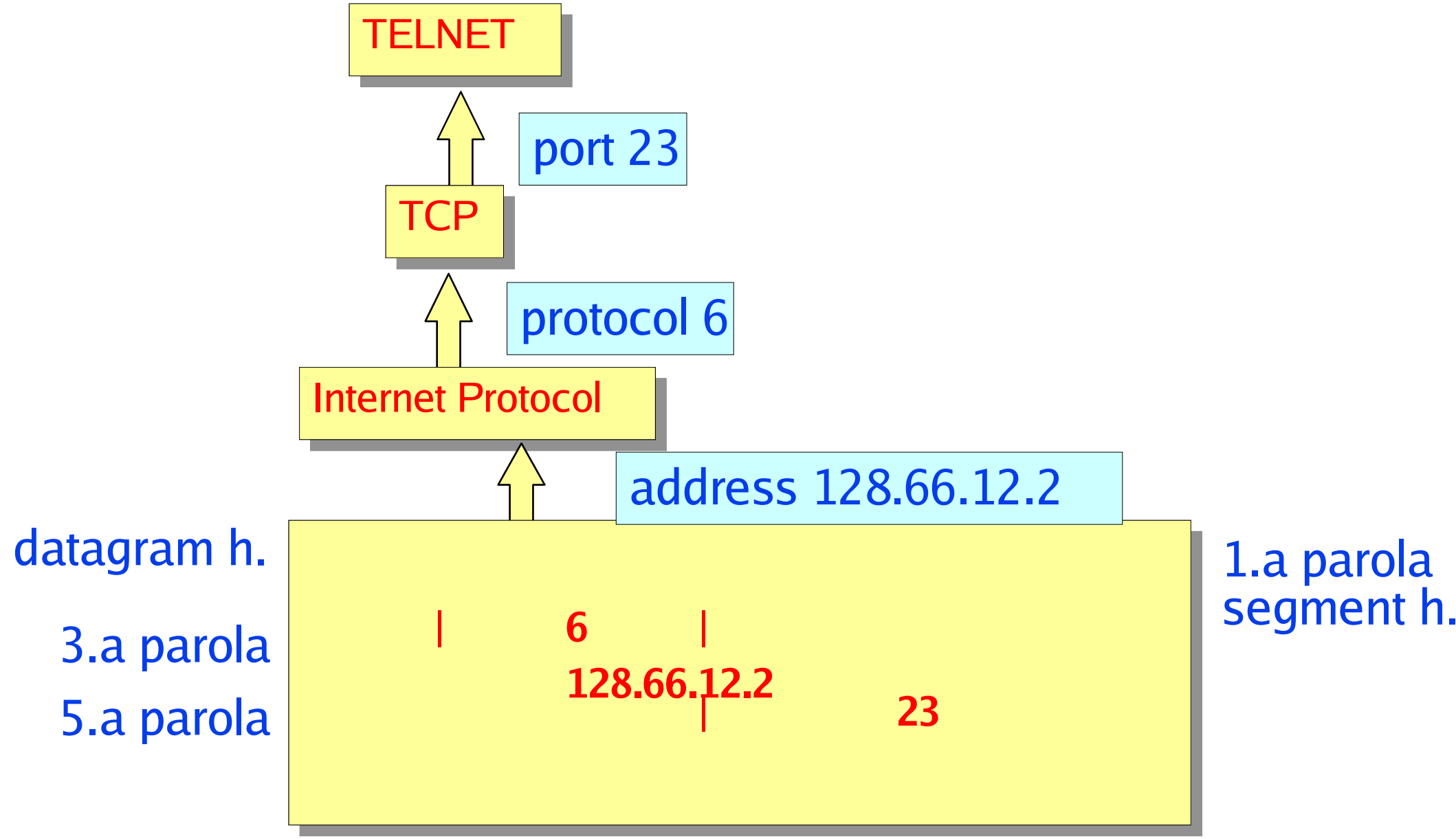
```
# Network services
#
echo            7/udp           ping
echo            7/tcp
systat          11/tcp
netstat         15/tcp
ftp-data        20/tcp
ftp             21/tcp
telnet          23/tcp
smtp            25/tcp          mail

...
```

dove: la prima colonna riporta il nome del processo

la seconda colonna indica in n.ro di porta / tipo di protocollo

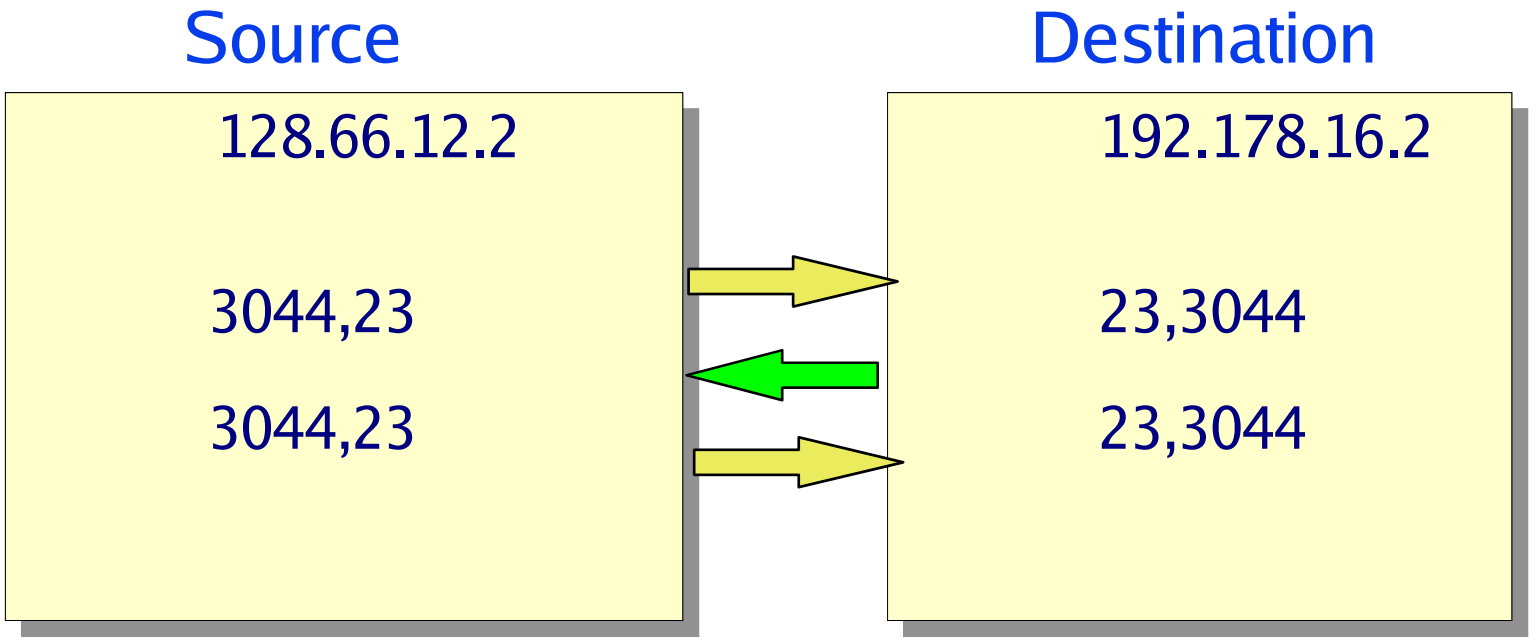
la terza colonna l'applicazione sotto cui gira il processo

Numeri di porta (3)



Sockets

- Un Socket è una combinazione di un indirizzo IP e un numero di porta.
- Un socket identifica univocamente un unico processo di rete sull'intera rete Internet.



Name Service (1)

- E’ possibile associare ad ogni interfaccia di rete (su reti di tipo TCP/IP) un nome (name) associato al relativo IP address.
- I nomi sono più facili da ricordare e da scrivere correttamente rispetto agli indirizzi numerici.
  - Es: 141.250.1.4 oppure egeo.unipg.it

Name Service (2)

- Quasi sempre gli indirizzi numerici e i nomi sono intercambiabili, ma in tutti i casi prima di effettuare una connessione il sistema converte l'hostname in un IP address.
- L'amministratore di rete è responsabile dell'assegnazione dei nomi e degli indirizzi e della loro memorizzazione.
- La traduzione di nomi in indirizzi deve essere nota a tutti gli host della rete

Name Service (3)

- Esistono due metodi diffusi per convertire i nomi in indirizzi:
  - Il metodo più vecchio prevede semplicemente la consultazione di una tabella chiamata host table.
  - Una tecnica più recente utilizza un sistema di database distribuiti chiamato Domain Name Service (DNS)

Host table (1)

- La host table è un semplice file di tipo testo che associa gli indirizzi IP con i rispettivi host name.
- Nei sistemi Unix la tabella si trova nel file /etc/hosts e può essere visualizzata con il comando:
  - `cat /etc/hosts`

Host table (2)

Esempio di /etc/hosts

```
#
# Tabella di IP address e host name
#
141.250.1.4   egeo.unipg.it egeo nsrhost raphost mailhost
127.0.0.1     localhost
141.250.1.7   teseo.unipg.it teseo ftp gopher finger
...
```

dove:

ad ogni IP address (es: 141.250.1.4) viene associato l'host name (es: egeo.unipg.it) e gli "altenate host name" o alias (es: egeo, nsrhost, raphost, mailhost).

L'host name e tutti i suoi alias riconducono allo stesso IP address.

L'indirizzo IP 127.0.0.1 viene associato al nome localhost per reindirizzare il proprio host come se fosse su rete remota

Host table (3)

- Nonostante l'introduzione del DNS la host table resta largamente usata perché:
  - molti utilizzano una piccola h.t. per gli host locali più importanti (proprio host, server locali e gateway locali) per usarla in caso che il DNS non funzioni o durante la fase iniziale di startup del sistema.
  - siti molto piccoli, che non sono connessi in Internet, usano la h.t., specie se gli host locali cambiano raramente le proprie informazioni
  - siti con vecchi sw su cui non "gira" DNS.

Host table (4)

- Unix fornisce una serie di comandi per costruirsi automaticamente /etc/hosts e /etc/networks con i dati messi a disposizione da DDN Network Information Center (DDN NIC). Si sconsiglia di usare questi comandi per costruirsi la h.t. (usare il DNS), ma restano usatissimi per costruirsi la /etc/networks che è un file per la traduzione di indirizzi di rete in nomi di rete.

Host table: la NIC h.t. (1)

- Il DDN NIC mantiene una grande tabella degli host di Internet chiamata NIC host table.
- La NIC h.t. è memorizzata sull'host nic.ddn.mil nel file netinfo/hosts.txt
- Gli host inclusi in questa tabella vengono detti registered hosts.
- La NIC h.t. contiene tre tipi di elementi:
  - record di rete
  - record di gateway
  - record di host

Host table: la NIC h.t. (2)

Formato dei record di hosts.txt:

_____ /etc/networks _____

NET : 26.0.0.0 : MILNET :

keyword network address network name

_____ /etc/hosts _____

HOST : 16.1.0.9 : WSL.DEC.COM : VAX : ULTRIX : TCP/SMTP :

keyword host address host name computer operating system service

_____ non usati_____

GATEWAY : 18.10.0.4, 18.26.0.138 : GW.LCS.MIT.EDU : VAX : CGW : IP/GW, EGP :

keyword gateway address gateway name computer operating system service

_____ non usati_____

Host table: la NIC h.t. (3)

- Per ottenere via rete la h.t. possiamo collegarci al NIC tramite ftp e poi scaricare il file /etc/hosts
- In caso di ambiente UNIX BSD (Berkeley) si hanno a disposizione specifiche funzioni:
  - gettable nic.ddn.mil — per prelevare il file hosts.txt
  - htable host.txt — per convertire host.txt nel formato /etc/hosts

  oppure

  - htable networks.txt — per convertire networks.txt nel formato /etc/networks

Host table: la NIC h.t. (3)

- La htable prende in input il file specificato (es:hosts.txt), i file localhosts, localnetworks e localgateways e crea in output tre file:
  - hosts
  - networks
  - gateways.
- E' a carico dell'amministratore di sistema rimuovere i file non interessanti (gateway, hosts.txt) e spostare in /etc quelli che interessano (hosts e networks)

Domain Name Service (DNS) (1)

- DNS è un sistema di database distribuiti, quindi non relega tutte le sue informazioni in una unica grande tabella (come l'host table) e non si intasa quando il database cresce.
- DNS garantisce l'aggiornamento a tutta la rete anche nell'inserimento di un nuovo piccolo host.

Domain Name Service (DNS) (2)

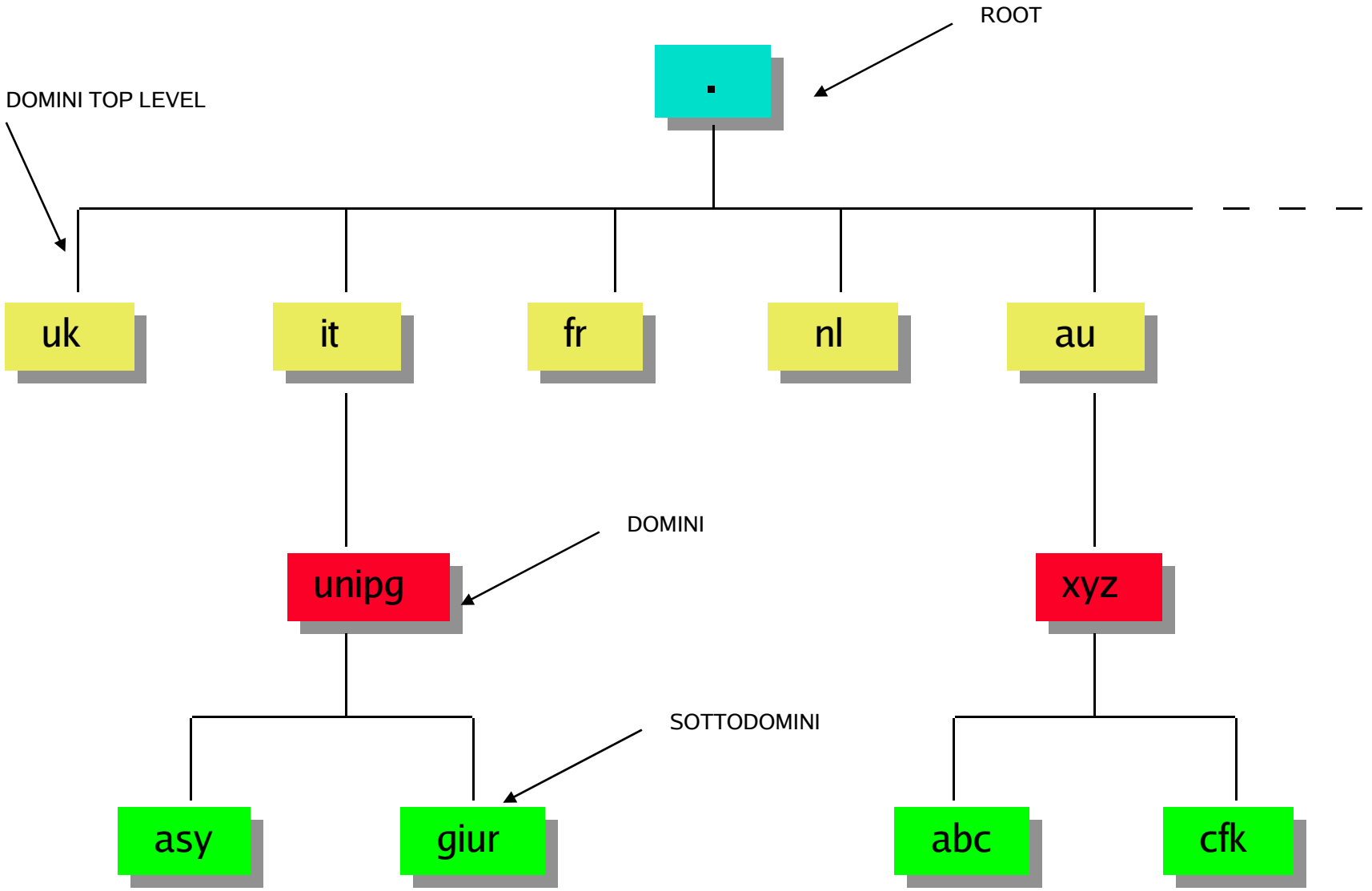
- Come funziona:

  quando un server DNS riceve una richiesta di informazioni su un host che non è sotto al suo diretto controllo, passa la richiesta ad un authoritative server. Un authoritative server è un server responsabile del dominio di cui voglio informazioni. Quando l'authoritative server risponde, il server locale salva (in cache) la risposta per un uso futuro.

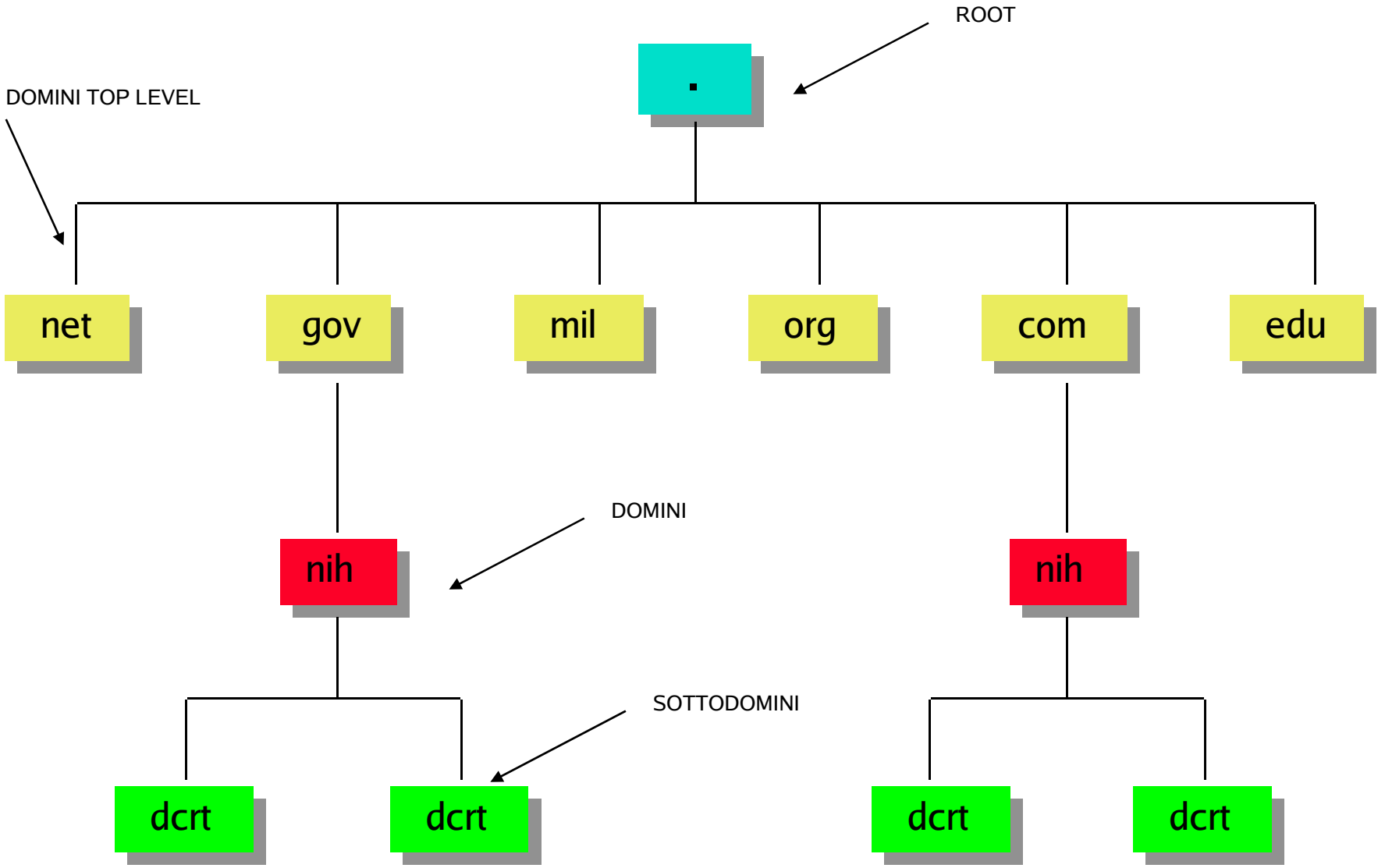
Domain Name Service (DNS) (3)

- Sia il DNS che l'host table vengono entrambi comunemente chiamati name service e listati nel file /etc/services. In tale file la h.t. ha assegnata una porta 42/UDP ed è chiamata nameservice, mentre DNS ha assegnata una porta 53/UDP ed è chiamato domain.

DNS: gerarchia di domini (geografica)



DNS: gerarchia di domini (U.S.A.)



DNS: BIND

- L'implementazione di DNS nella maggior parte dei sistemi UNIX è il sw chiamato Berkeley Internet Name Domain (BIND)
- Il sw è essenzialmente diviso in due componenti:
  - resolver che gestisce le "query"
  - name server che gestisce le risposte

DNS: BIND (name server)

- I name server di BIND girano un processo chiamato named ("name" "d") e vengono così classificati:
  - primary server da cui vengono ricavati tutti i dati di un dominio. Questi server caricano le informazioni sul dominio direttamente da un file creato dall'amministratore di dominio.
  - secondary server trasferiscono l'intero database del dominio (zone file) dal suo primary server.
  - caching-only server prendono informazioni da altri server in occasione di query e le mantengono per le volte successive. Le loro informazioni non sono esaustive di un dominio e quindi sono considerati non-authoritative.

Configurazione del TCP/IP

- Per poter configurare un host a “girare” sotto TCP/IP occorre avere prima queste informazioni:
  - IP address
  - host name
  - default gateway address
  - routing protocol
  - name server address
  - domain name
  - subnet mask
  - broadcast address

Come ottenere un IP address

- Per ottenere un IP address di rete si deve inoltrare domanda ad appositi centri:
  - In U.S.A. al N.I.C.
  - In Europa al R.I.P.E. o ai suoi centri dislocati a livello nazionale, per l'Italia al GARR-NIS

Come assegnare un host address

- Un amministratore di rete ottenuto l'indirizzo di rete (network number) può assegnare gli indirizzi agli host in due modi:
  - un indirizzo alla volta:
    - ad ogni host viene assegnato un indirizzo, in ordine sequenziale nell'ambito del range degli indirizzi disponibili per quella rete (address space)
  - gruppi di indirizzi:
    - vengono delegati ad amministratori di sottoreti blocchi di indirizzi, all'interno della sottorete i vari indirizzi degli host saranno definiti in ambito locale

Come scegliere un host name

- RFC 1178 fornisce ottimi consigli per scegliere il nome degli host, alcune tra le più importanti sono:
  - Usare parole reali che sia corte, facili da pronunciare e da ricordare.
  - Usare nomi tematici, cioè tutti gli host di un gruppo dovrebbero avere nomi riguardanti una certo tema (es: mitologico)
  - Evitare di usare nomi di progetti, nomi di persona, acronimi, ecc., perché per loro natura sono provvisori

Come definire i routing (1)

- Esistono tre modi per definire i routing:
  - Minimale
    - Non è necessario specificare nessun routing, è la modalità con nessun gateway ad altre reti.
  - statico
    - L'instradamento viene gestito da una static routing table costruita dall'amministratore di sistema. Questa modalità è usata per reti con un basso numero di gateway.
  - dinamico
    - L'instradamento viene gestito da una dynamic routing table che si adatta a tutti i cambiamenti della rete. Queste tabelle vengono costruite da appropriati protocolli di routing. Tali protocolli scambiano informazioni che vengono utilizzare per aggiornamenti delle tabelle. La modalità dinamica risulta molto usata per reti con molti gateway, indispensabile per reti in cui più di un gateway può raggiungere la stessa destinazione.

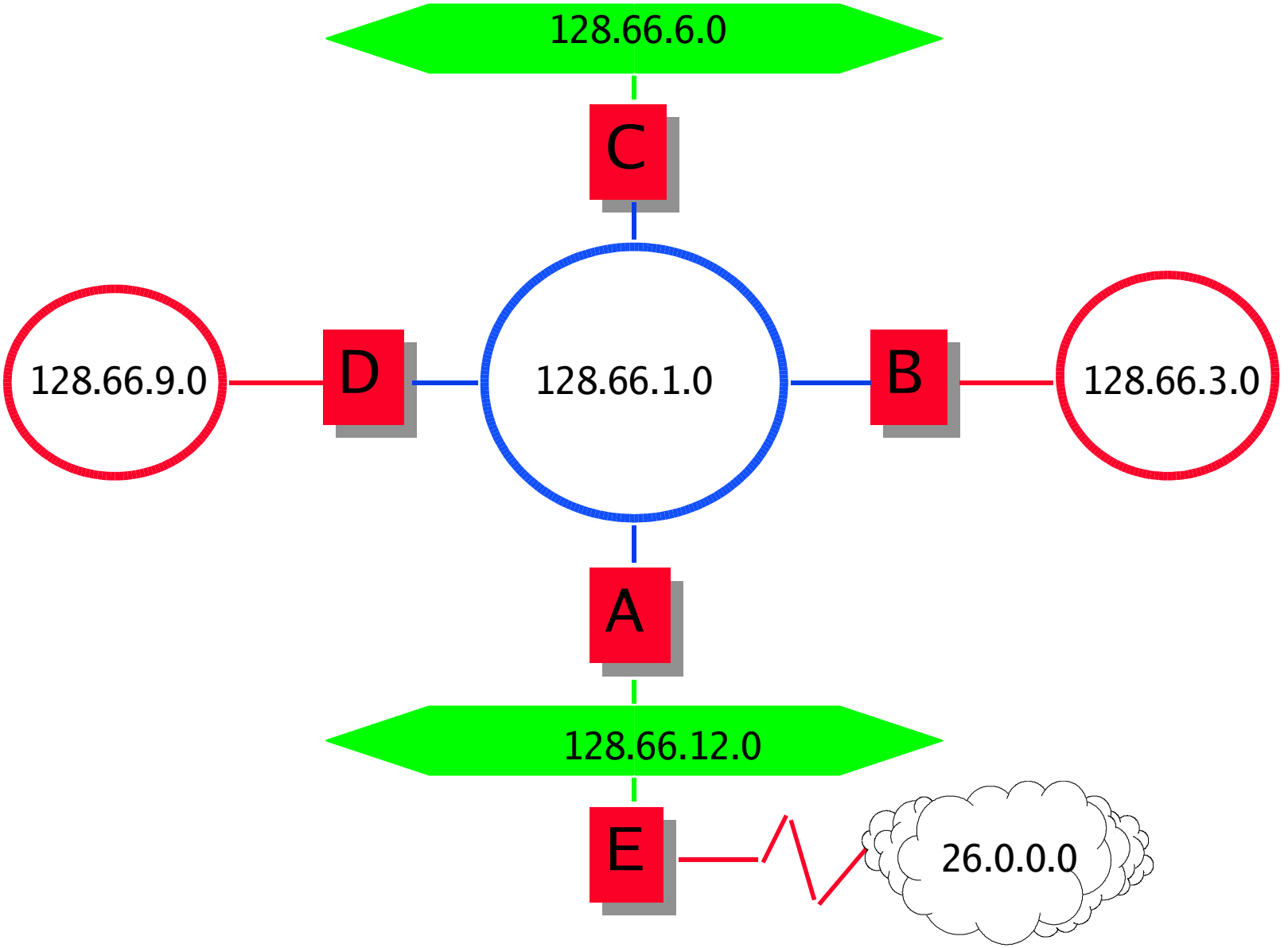
Come definire i routing (2)

- L'amministratore deve decidere quale tipo di routing usare e scegliere il gateway di default per ogni host, per fare ciò è consigliabile seguire le seguenti direttive:
  - Rete con nessun gateway ad altre reti TCP/IP: non è necessario specificare nessun routing né gateway di default.
  - Rete con un solo gateway: non occorre far girare alcun protocollo di routing, basta specificare l'unico gateway come il gateway di default nella tabella di routing statico.

Come definire i routing (3)

- Rete con vari gateway interni per altre sottoreti e un unico gateway per l'esterno: si può o specificare staticamente ogni route della sottorete e definire il gateway con l'esterno come default route oppure far "girare" un protocollo di routing per una gestione dinamica. Dipende dal numero di host e sottoreti, chiaramente il protocollo di routing implica un lieve appesantimento sui vari host e sottoreti.
- Rete con molti gateway verso l'esterno: se si hanno vari gateway che raggiungono la stessa destinazione si deve usare un protocollo di routing per aggiornare ogni cambiamento di rete sui vari gateway.

Come definire i routing (4)



Quando definire una sottorete (1)

- La decisione di creare una sottorete dipende da aspetti topologici ed organizzativi.
  - ragioni topologiche:
    - superamento limiti di distanza: alcuni componenti hw hanno limiti sulla distanza, per es. la rete Ethernet, in cui i cavi “thick” possono avere una lungh. max di 500m e i cavi “thin” di 300m, non appena si superano queste distanze occorre inserire un router IP per collegare una serie di cavi. Nota che la lungh. della rete è data dalla somma di tutti i cavi, non dal cavo più lungo.
    - connessione di reti fisiche diverse: router IP possono essere usati per collegare reti che hanno una diversa tecnologia (token ring a ethernet).
    - filtro del traffico fra reti: il traffico locale rimane nella sottorete locale, solo il traffico verso altre reti è inviato al gateway.

Quando definire una sottorete (2)

- ragioni organizzative:
  - semplificare l'amministrazione di rete: le sottoreti possono essere usate per delegare la gestione degli indirizzi, diagnostica, ecc. a piccole entità.
  - riconoscimento di strutture: la struttura di un ente può richiedere l'indipendenza di qualche divisione (es. dipartimenti universitari)
  - isolamento del traffico di uno specifico ente: per motivi di sicurezza possono esistere enti che vogliono isolare il traffico locale in modo tale da rendere accessibile la sottorete ai soli membri.
  - isolamento di problemi potenziali: un certo segmento di rete può essere meno attendibile del resto della rete, per es. una sottorete può essere riservato ad un gruppo di tecnici per sperimentare nuovi prodotti di rete.

L'Internet Daemon (1)

- Alcuni daemon di protocolli vengono inseriti nei file di boot del sistema e quindi vengono attivati individualmente, e questi sono per esempio:
  - routed per il Routing Information Protocol (RIP)
  - named per il Domain Name Service (DNS)
- Altri daemon non vengono fatti partire individual-mente, ma vengono attivati da un super-server chiamato internet daemon (inetd).
- L' internet daemon analizza le richieste di servizio di rete e fa partire i daemon appropriati per soddisfare le richieste.

L'Internet Daemon (2)

- L'inetd viene fatto partire in fase di boot da un file di inizializzazione es. da /etc/rc.
- Appena partito inetd legge il suo file di configurazione /etc/inetd.conf che contiene i nomi dei servizi per cui inetd fa partire i daemon appropriati.
- Un es. di definizione di servizio in /etc/inetd.conf è:
  - ftp stream tcp nowait root /usr/et/in.ftp in.ftp

L'Internet Daemon (3)

- Genericamente una definizione di servizio in /etc/inetd.conf prevede i seguenti campi:

  name type protocol wait-status uid server arguments

  - name
    - indica il nome del servizio, come scritto in /etc/services
  - type
    - indica il tipo di servizio di consegna dei dati usato, chiamato anche socket type. I tipi più usati sono:
      a - **stream** indica il TCP byte stream
      b - **dgram** indica il servizio di consegna a packet (datagram) di UDP
      c - **raw** servizio di datagram IP

L'Internet Daemon (4)

- protoloc
  - indica il nome del protocollo, come dato in /etc/protocols, i valori più usati sono **tcp** e **udp**.
- wait-status
  - i possibili valori sono: **wait** o **nowait** (generalmente server con type datagram richiedono wait, mentre server con type stream richiedono nowait). Se lo stato è **wait** inetd deve aspettare che il server rilasci il socket prima di associarlo ad un'altra richiesta. Se lo stato è **nowait** inetd può immediatamente cominciare una nuova connessione su quel socket per un'altra richiesta (uso di socket allocati dinamicamente).

L'Internet Daemon (5)

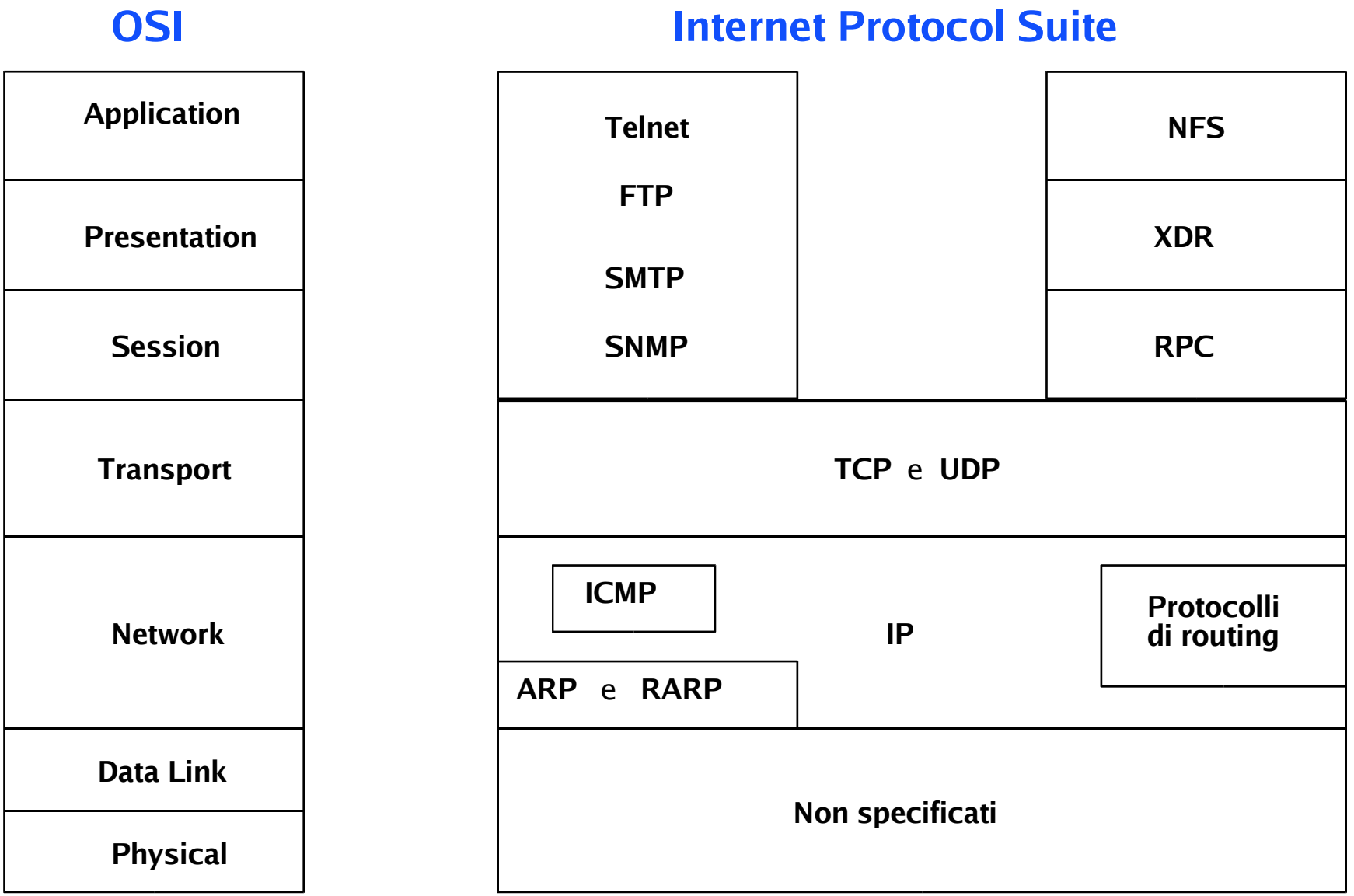
- uid
  - nome dell'utenza sotto cui gira il server, normalmente è **root,** ci sono però due eccezioni: il servizio ***finger*** gira sotto l'utenza *nobody* o *daemon* e il servizio *uucp* gira qualche volta sotto l'utenza *uucp*.
- server
  - è il completo **pathname** del programma del server che inetd deve far partire, può essere usato anche il valore **internal** per indicare che è un servizio gestito direttamente da una funzione di inetd.
- arguments
  - è una lista di argomenti da passare al programma server invocato, per la lista di arg. consultare il *man* del programma.

L'Internet Daemon (6)

- Esempio di un file /etc/inetd.conf

```
ftp         stream tcp    nowait root   /usr/etc/in.ftp     in.ftp
telnet      stream tcp    nowait root   /usr/etc/in.telnetd in.ternetd
shell       stream tcp    nowait root   /usr/etc/in.rshd    in.rshd
login       stream tcp    nowait root   /usr/etc/in.rlogind in.rlogind
exec        stream tcp    nowait root   /usr/etc/in.rexecd  in.rexecd
finger      stream tcp    nowait nobody /usr/etc/in.fingerd in.fingerd
talk        dgram  udp    wait   root   /usr/etc/in.talkd   in.talkd
name        dgram  udp    wait   root   /usr/etc/in.tnamed  in.tnamed
daytime     stream tcp    nowait root   internal
time        stream tcp    nowait root   internal
echo        dgram  udp    wait   root   internal
....
```

Architettura di rete TCP/IP

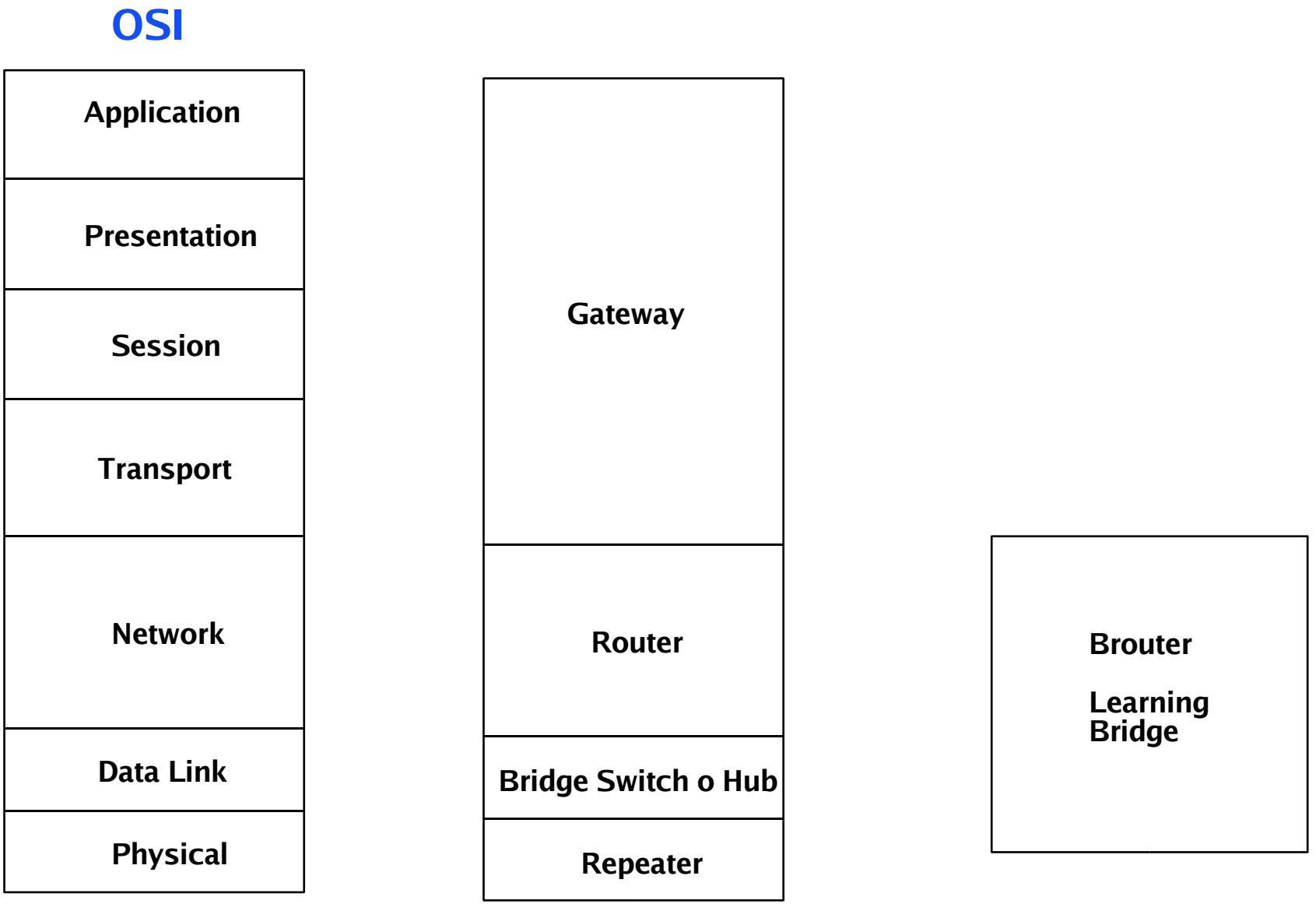


Sotto l'IP

- TCP/IP non specifica i livelli 1 e 2 della rete, ma utilizza quelli disponibili e conformi agli standard.

  Es. su reti locali opera su Ethernet/IEEE802.3, Token-Ring e FDDI, su reti geografiche su HDLC, PPP, SLIP, X.25, Frame Relay, SMDS e ATM.

Dispositivi e livelli OSI



Repeater

- Dispositivo che opera a livello 1. Abbastanza “stupido”, si limita a ripetere ciò che gli arriva.
- Si utilizza quando esiste una limitazione fisica alla lunghezza della LAN. In effetti le 2 LAN possono essere viste, da un punto di vista logico come una sola.

Bridge

- Il Bridge è un dispositivo che opera a livello 2 OSI
- Viene utilizzato per creare una LAN estesa unendo due o più LAN.
- Un bridge ritrasmette selettivamente i pacchetti tra le LAN cui è connesso. Nell'es. il bridge essendo trasparente ai protocolli di livello superiore non ha indirizzi IP.
- Ha prestazioni molto elevate dato il basso carico elaborativo richiesto.
- Dato che opera a livello 2 non si cura di cosa ci sia sulla LAN e può collegare in una sola rete logica spezzoni ethernet, token ring, FDDI,...

Bridge

- In base alla tipologia delle 2 reti si distinguono in:
  - *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 (FDDI, Ethernet,…). Per questo motivo deve essere in grado di adattarsi alle diverse regole trasmissive delle 2 reti (lunghezza della trama, diverse velocità)

Bridge

- Essenzialmente le fasi dell'attività che un bridge deve fare sono 2:
  - Filtering
    - il bridge analizza le trame che gli arrivano da una LAN, cercando di capire se sono indirizzate ad una workstation della seconda LAN
  - Forwarding
    - il bridge passa alla seconda LAN le trame indirizzate ad workstation appartenenti alla stessa

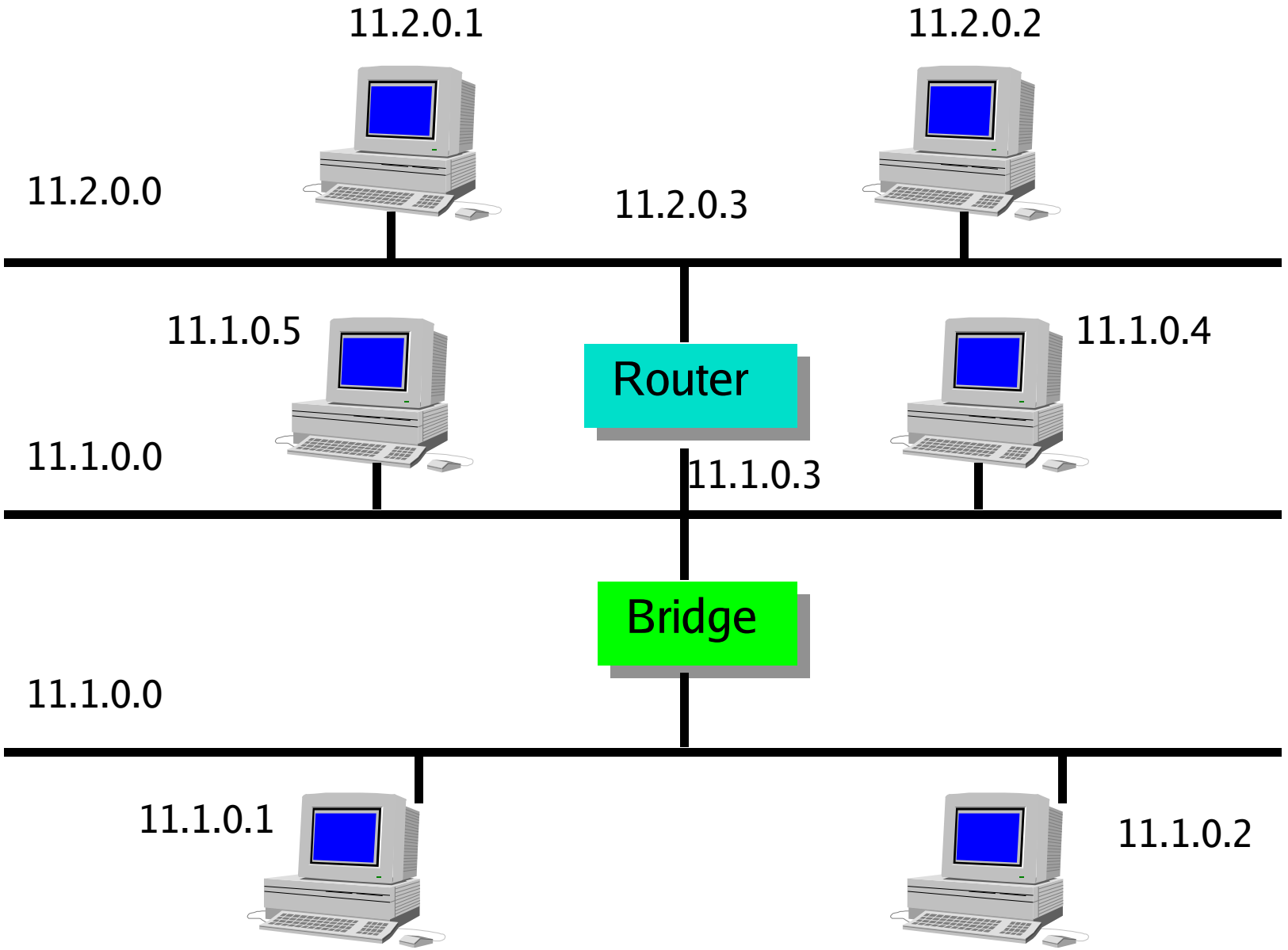
Learning Bridge

- Un bridge riconosce le stazioni dall'indirizzo MAC dall'adattatore di ogni dispositivo presente nelle proprie LAN. Siccome impostare a mano questi indirizzi è una operazione noiosa sono nati i Learning bridge, che, con un breve periodo di auto apprendimento, riconoscono automaticamente le stazioni presenti sulle proprie LAN.
- Limiti dei bridge:
  - I bridge falliscono quando esistono possibilità di path multipli per collegare 2 LAN.
  - Path multipli possono essere necessari per aumentare la sicurezza di una connessione, ad esempio una doppia dorsale con una doppia cascata o con un anello di connessioni remote con link di backup.

Router e Bridge

- Il Router è un dispositivo che opera a livello 3 OSI
- E' in grado di instradare pacchetti da un nodo all'altro e ammette l'esistenza di più cammini fra questi nodi.
- I router collaborano per decidere il cammino ottimale che ogni pacchetto deve percorrere da un nodo sorgente ad uno destinatario.
- Nell'es. il router collega due sottoreti e necessita di due indirizzi IP.
- Il funzionamento di un router è legato al protocollo utilizzato: un router per TCP/IP non instrada pacchetti IPX/SPX e viceversa.

Router e Bridge



Brouter

- Come dice il nome stesso è metà bridge e metà router, infatti si comporta da router per alcuni protocolli e da bridge per altri

Router e gateway

- Il routing tra sottoreti è gestito dagli IP router che originariamente erano stati definiti gateway, tale definizione è però infelice perché gli IP gateway sono quelli che OSI chiama router e i gateway OSI non hanno corrispettivo nel mondo TCP/IP.

  E' comunque uso generale chiamare gateway gli IP router.

  Gli IP router effettuano l'instradamento sulla base di tabelle di routing scritte manualmente o calco-late automaticamente tramite algoritmi ad hoc.

Gateway

- Un gateway è un dispositivo usato per connettere reti con architetture diverse. La funzione di un gateway è quella di convertire i protocolli di una architettura di rete in quelli di un'altra architettura.

  Se nell'es. precedente cambio architettura di rete tra la 11.1 e la 11.2 con protocolli diversi ho bisogno di un gateway.

Il protocollo ICMP (1)

- Il protocollo Internet Control Message Protocol (ICMP) è stato progettato per riportare anomalie che accadono nel routing dei pacchetti IP e per verificare lo stato della rete.
- I vari tipi di messaggi ICMP sono:

Codice	Messaggio	Codice	Messaggio
0	Echo Reply §	13	Timestamp Request
3	Destination Unreachable *	14	Timestamp Replay
4	Source Quence	15	Information Request
5	Redirect	16	Information Replay
8	Echo Request §	17	Address Mask Request
11	Time Exceeded for a Datagram *	18	Address Mask Replay
12	Parameter Problem on a Datagram *		

* messaggi che riportano anomalie

§ messaggi di verifica della raggiungibilità di un nodo

Il protocollo ICMP (2)

- Il messaggio Redirect indica una condizione di stimolo ad un routing migliore, in quanto un router è stato attraversato inutilmente (ha dovuto ritrasmettere il messaggio sulla stessa rete da cui lo ha ricevuto). Quando un host riceve un pacchetto di routing redirect associa un router diverso da quello di default a quella destinazione.
- I messaggi Mask Request e Address Mask Reply sono stati introdotti per permettere ad una interfaccia di scoprire automaticamente la netmask usata in quella rete

I protocolli ARP e RARP (1)

- I protocolli Address Resolution Protocol (ARP) e Reverse Address Resolution Protocol (RARP) sono utilizzati per scoprire in modo automatico le corrispondenze tra gli indirizzi di livello 3 e gli indirizzi di livello 2 e viceversa. Risultano indispensabili in LAN in cui occorre creare una relazione tra indirizzi IP e indirizzi MAC.
- Il protocollo ARP viene usato tutte le volte che una stazione collegata ad una LAN deve inviare un messaggio ad un nodo sulla stessa LAN di cui conosce unicamente l'indirizzo di livello 3

I protocolli ARP e RARP (2)

- Il protocollo RARP viene utilizzato dalle stazioni non dotate di memoria (diskless) per scoprire il loro indirizzo IP in fase di bootstrap.
- I protocolli ARP e RARP operano direttamente sulle reti locali e non su IP (come fa ICMP); infatti essi inviano le loro richieste in broadcast a tutte le stazioni della rete, anche quelle che non utilizzano TCP/IP.
  - La richiesta in broadcast ARP contiene l'indirizzo IP del nodo di cui si vuole scoprire l'indirizzo di livello 2; il nodo avente l'indirizzo IP specificato risponde alla richiesta fornendo il sui indirizzo di livello 2.
  - Il protocollo RARP funziona in modo simile, ma viene fornito un indirizzo di livello 2 e richiesto un indirizzo IP.

Gli Autonomous System (1)

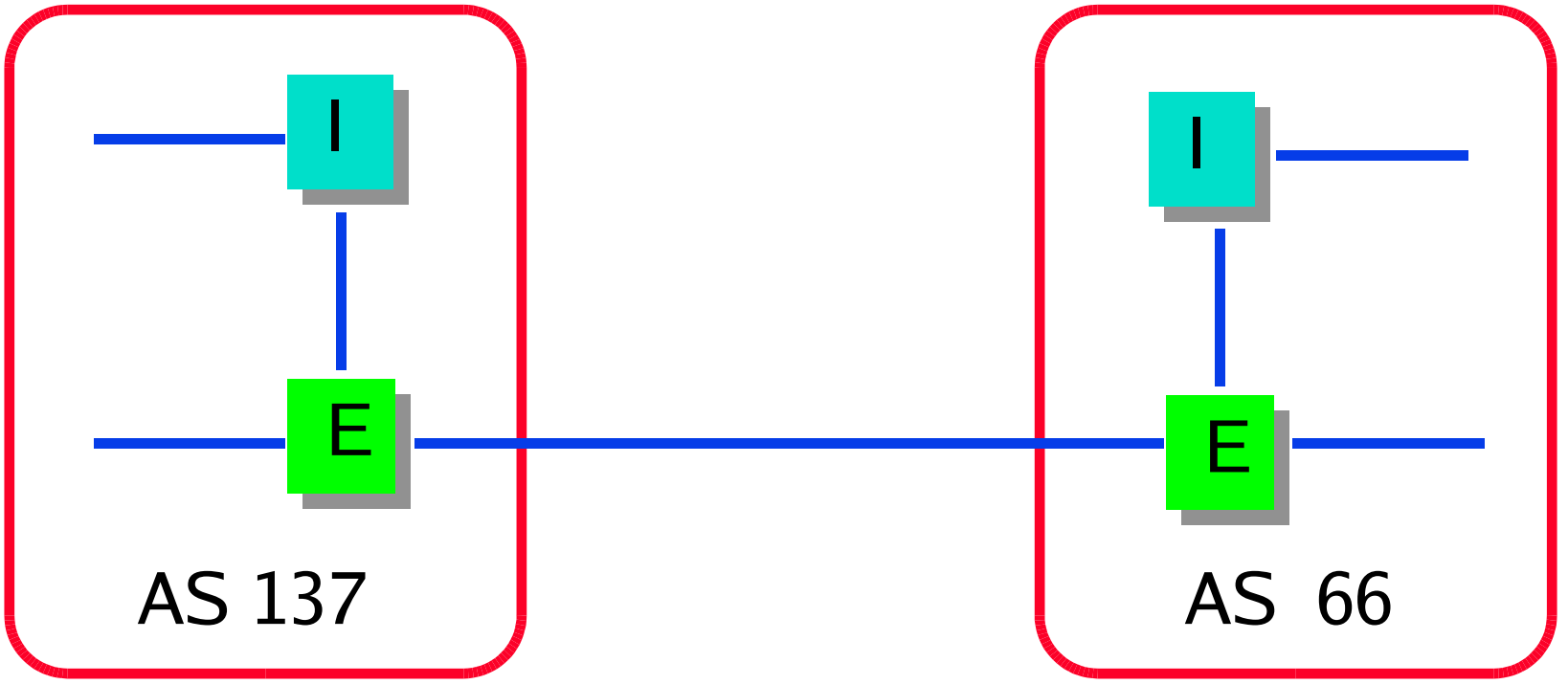
- Il routing TCP/IP è un routing gerarchico su più livelli:
  - un primo livello, all'interno della sottorete, è implicito in quanto è gestito dalla rete fisica
  - un secondo livello, tra le varie reti e sottoreti, viene gestito dagli IP router tramite tabelle di routing
  - un terzo livello raggruppa le reti in Autonomous System (AS), cioè in gruppi di reti controllate e gestite da un'unica autorità.

Gli Autonomous System (2)

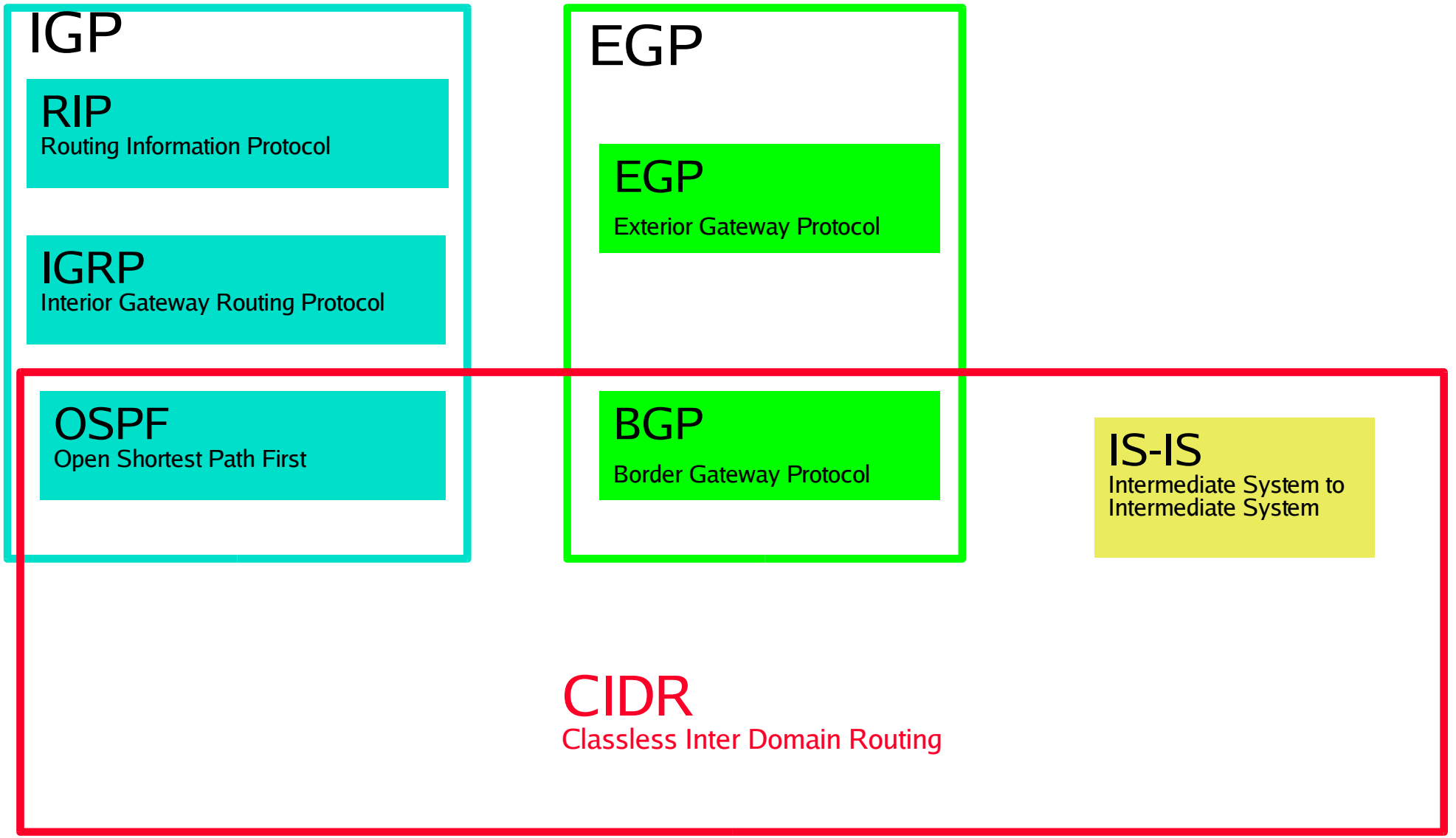
- Gli AS sono identificati da un numero intero, univoco a livello mondiale, assegnato dalla stessa autorità che rilascia gli indirizzi Internet.
- I router che instradano messaggi all'interno dello stesso AS sono detti Interior Router, mentre quelli che instradano messaggi anche tra AS diversi sono detti Exterior Router.
  - Gli Interior Router scambiano informazioni tramite un IGP (Interior Gateway Protocol)
  - Gli Exterior Router comunicano invece tramite un EGP (Exterior Gateway Protocol)

Gli Autonomous System (3)

Esempio di connessione di due AS



Protocolli di routing (1)



Configurazione di una interfaccia di rete

- Quando i protocolli di rete lavorano solo con tipo di rete fisica non al software non è necessario identificare l'interfaccia di rete.
- Poiché però il TCP/IP permette un uso flessibile di differenti reti a livello fisico, l'amministratore di rete deve configurare tali interfacce

Il comando ifconfig

- Il comando ifconfig imposta o controlla i valori di configurazione per le interfacce di rete
- Il comando ifconfig permette al TCP/IP di identificare l'interfaccia di rete e di assegnarle:
  - IP Address
  - subnet mask
  - broadcast address
- ifconfig ha molti argomenti e parole chiave, di seguito vedremo i più importanti per ottenere quelle informazioni base necessarie al TCP/IP per riconoscere una intefaccia di rete

Configurare con ifconfig (1)

- Una sintassi possibile per il comando ifconfig è:

  ifconfig <*interface*> <*address*> **netmask** <*mask*> **broadcast** <*address*>

  es: ifconfig le0 128.66.12.2 netmask 255.255.255.0 broadcast 128.66.12.255

  In cui gli argomenti sono:

  - <*interface*> indica il nome dell'interfaccia da configurare
  - <*address*> è l'indirizzo IP assegnato all'interfaccia: si può scrivere come indirizzo decimale separato da punti oppure come un host name. Se si usa l'host name occorre inserire l'host name e il suo IP address nel file /etc/hosts, perché il sistema deve essere in grado di risolvere i nomi anche senza DNS.

Configurare con ifconfig (2)

- **netmask** <*mask*> specifica la subnet mask per quella interfaccia. Il parametro è da impostare solo se si divide la rete in sottoreti. Se si stanno impostando le sottoreti occorre ricordare che tutti i sistemi della rete devono avere la stessa subnet mask.
- **broadcast** <*address*> specifica l'indirizzo di broadcast per la rete. In genere è quell'indirizzo della rete con impostati a 1 tutti i bit relativi all'host.

Determinare l'interfaccia con netstat (1)

- Per determinare tutte le interfacce di rete disponibili nel sistema si usa il comando netstat.
- Per esempio per controllare lo stato di tutte le interfacce di rete disponibili si da:
  - netstat -ain

    dove:

    -i indica che il display richiesto riguarda le interfacce di rete configurate

    -a indica che le intefacce prese in considerazione sono tutte non solo quelle già configurate

    -n visualizza l'output in forma numerica

Determinare l'interfaccia con netstat (2)

Esempio di output di netstat -ain :

Name	Mtu	Net/Dest	Address	Ipkts	Ierrs	Opkts	Oerrs	Collis	Queue
le0	1500	128.66.0.0	128.66.12.2	1547	1	1127	0	135	0
lo0	1536	127.0.0.0	127.0.0.1	133	0	133	0	0	0

I campi visualizzati da netstat -ain sono:

- Name: nome dell'interfaccia, così come dato dal comando ifconfig. Se appare un * significa che l'interfaccia non è abilitata
- Mtu: La Maximum Transmission Unit indica la lungh. max di un pacchetto che può essere trasmesso senza frammentazioni
- Net/Dest: Indica la rete o l'host a cui quella interfaccia da accesso. In genere contiene indirizzi di reti, può contenere un indirizzo di host se l'interfaccia è configurata come un link point-to-point (host-specific), questo tipo di connessione è un collegamento diretto tra due computer.
- Address: indica l'indirizzo IP assegnato all'interfaccia
- Ipkts: è il campo Input Packets che indica quanti pacchetti ha ricevuto l'interfaccia

Determinare l'interfaccia con netstat (3)

- Ierrs: è il campo Input Errors che indica quanti pacchetti danneggiati ha ricevuto l'interfaccia
- Opkts: è il campo Output Packets che indica quanti pacchetti sono stati mandati fuori da questa interfaccia
- Oerrs: è il campo Output Errors che dice quanti pacchetti inviati hanno causato condizioni di errore.
- Collis: il campo Collisions mostra quante collisioni di tipo Ethernet si sono verificate per quell'interfaccia. (solo per reti Ethernet)
- Queue:il campo Packets Queued riporta quanti pacchetti sono in attesa di utilizzare l'interfaccia, normalmente questo campo è a 0.

Controllare l'interfaccia con ifconfig (1)

- Si può controllare lo stato di una interfaccia con il comando:
  - ifconfig *le0*

che produce in output:

```
le0: flags=63<UP,BROADCAST,NOTRAILERS,RUNNING>
        inet 128.66.12.2 netmask ffff0000 broadcast 128.66.0.0
```

dove il parametro flags indica in forma numerica e accanto per esteso le caratteristiche di interfaccia:

UP	interfaccia abilitata all'uso.
BROADCAST	interfaccia connessa a una rete che supporta i broadcast (es: ethernet)

Controllare l'interfaccia con ifconfig (2)

NOTRAILERS	interfaccia connessa a una rete che non supporta la trailer encapsulation (specifica per Ethernet)
RUNNING	L'interfaccia è operativa.

La seconda linea riporta tutte le caratteristiche dell'interfaccia così come impostate con un precedente comando di configura-zione ifconfig.

Assegnare una subnet mask con ifconfig

- Esistono diversi metodi per assegnare una netmask con ifconfig:
  - il più semplice e più diffuso è di assegnare direttamente il valore numerico con:
    - ifconfig le0 128.66.12.2 netmask 255.255.255.0
  - mettere in /etc/network la seg. linea:
    - my-mask 255.255.255.0

    e poi:
    - ifconfig le0 128.66.12.2 netmask my_mask
  - su sistemi SunOS si può dichiarare in /etc/netmasks:
    - 128.66.12.2 255.255.255.0

    e poi:
    - ifconfig le0 128.66.12.2 netmask +
      dove il segno + indica che il valore deve essere letto in /etc/netmasks

Impostare il Broadcast Address con ifconfig

- Per impostare l'indirizzo di broadcast con il comando ifconfig basta esplicitare l'indirizzo dopo la parola chiave broadcast:
  - ifconfig le0 128.66.12.2 netmask 255.255.255.0 broadcast 128.66.12.255

N.B.:

L'indirizzo di broadcast è relativo alla sottorete locale, cioè l'interfaccia dichiarata è vista come connessa alla rete 128.66.12.0 che ha come indirizzo di broadcast 128.66.12.255 .

L'indirizzo 128.66.255.255 verrebbe interpretato come l'indirizzo dell'host 255 della sottorete 255 della rete 128.66.0.0 non come un indirizzo di broadcast.

Assegnare un indirizzo ad una interfaccia di rete con ifconfig

Per assegnare l'indirizzo ad una interfaccia di rete basta scrivere il nome dell'interfaccia e l'indirizzo ad essa associato:

```
ifconfig le0 128.66.12.2
```

in alternativa si può definire in /etc/hosts:

```
128.66.12.2              my-gw.nuts.com   my-gw
```

e poi configurare con:

```
ifconfig le0 my-gw.nuts.com
```

Dove inserire i comandi ifconfig

- I comandi ifconfig generalmente vengono eseguiti in fase di boot del sistema da uno dei file di startup.
- Su UNIX BSD i comandi ifconfig vengono di solito inseriti nei file /etc/rc.boot o /etc/rc.local.
- Su UNIX System V i file di startup sono molti e più complessi, ma in generale i comandi ifconfig vengono inseriti in file che hanno nomi tipo /etc/tcp o /etc/init.d/tcp.

Altre opzioni del comando ifconfig (1)

- up e down abilitano e disabilitano l'interfaccia di rete:
  - ifconfig le0 down
  - ifconfig le0 129.66.1.2 up

    una sequenza di down e up può essere utilizzata per riconfigurare l'interfaccia di rete.

Altre opzioni del comando ifconfig (2)

- arp e trailers vengono usate solo per interfacce di tipo Ethernet.
- L'opzione trailers abilita o disabilita (-trailers) la modalità di trailer encapsulation di pacchetti IP.
  - La tecnica di trailer encapsulation permette di ridurre le copie memory-to-memory richieste dal sistema ricevente.
- L'opzione arp abilita o disabilita (-arp) la mappatura degli indirizzi IP con indirizzi fisici Ethernet, svolta appunto dall'Address Resolution Protocol (ARP).

Altre opzioni del comando ifconfig (3)

- metric viene usata solo in sistemi UNIX che utilizzano il Routing Information Protocol (RIP).
  - Il RIP distribuisce informazioni di routing agli altri host e costruisce dinamicamente tabelle di routing. RIP prende le proprie decisioni di instradamento in base ai costi delle varie route. Il costo di una route viene determinato da un routing metric associato alla route. Un routing metring è un numero (più alto è il numero più alto è il costo). RIP favorisce costi bassi! Interfacce direttamente attaccate alla rete hanno di solito il default metric a 0.
- l'opzione metric viene utilizzata per impostare il costo di una interfaccia ad un certo valore.
  - ifconfig le0 26.104.0.19 metric 3
- Usare l'opzione metric solo se esiste un'altra route per la stessa destinazione e si vuole usare quest'ultima come route primaria.

RIPASSO

Distinzione fra routing e protocolli di routing:

- Routing è l'atto di inviare datagram basato su informazioni contenute nella routing table.
- I protocolli di routing sono programmi che scambiano le informazioni usate per costruire le routing table.

Configurazione del routing

Esistono tre tipi generali di configurazioni:

- routing minimale

  Una rete completamente isolata da tutte le altre reti TCP/IP richiede solo un routing minimale. Una routing table minima viene costruita tramite comandi ifconfig quando viene configurata l'interfaccia alla rete.

- routing statico

  Una rete con un numero limitato di gateway ad altre reti TCP/IP può essere configurata con un routing statico. Una tabella di routing statico viene costruita manualmente dall'amministratore di sistema usando il comando route. I cambiamenti sulla rete non vengono aggiornati automaticamente, quindi è adatta per reti che non cambiano spesso.

- routing dinamico

  Una rete con più di una via possibile per una stessa destinazione dovrebbe usare un routing dinamico. Una tabella di routing dinamico viene costruita dalle informazioni scambiate dai protocolli di routing. I protocolli sono designati a distribuire informazioni che dinamicamente aggiustano le route per riflettere cambiamenti delle condizioni di rete.

La tabella di routing minima (1)

- Esempio del contenuto della tabella di routing costruita da un comando ifconfig:

netstat -rn

Routing tables

Destination	Gateway	Flags	Refcnt	Use	Interface
127.0.0.1	127.0.0.1	UH	1	298	lo0
128.66.12.0	128.66.12.2	U	26	50360	le0

la prima riga mostra il loopback al localhost creato quando è stato configurata lo0, la seconda riga è la route alla rete 128.66.12.0 attraverso l'interfaccia le0: l'indirizzo 128.66.12.2 non è un indirizzo di un gateway remoto ma è l'indirizzo assegnato all'interfaccia le0 su quel host (infatti nella colonna Flags non appare G, U=up, H=host).

La tabella di routing minima (2)

- Le limitate capacità di una tale tabella possono essere facilmente verificate con un comando ping. Ping usa l'Echo message dell'ICMP per forzare un host remoto a rispedire indietro all'host locale un pacchetto.

  ping -s almond

  PING almond.nuts.com 56 data byte
  64 byte from almond.nuts.com (128.66.12.1):: icmp_seq=0. time=11. ms
  64 byte from almond.nuts.com (128.66.12.1):: icmp_seq=1. time=10. ms
  ^C
  -------- almond.nuts.com PING Statistic ----------
  2 packets trasmitted, 2 packets received, 0% packet loss
  round-trip (ms) min/avg/max = 10/10/11

- Se però si controlla un host che non appartiene alla rete 128.66.12.0:

  ping 26.40.0.17

  sendto: Network is unreachable

- Così come per host che appartengono a sottoreti (**ping 128.66.1.2**)
- I test di ping mostrano che la tabella di routing creata con ifconfig permette solo comunicazioni con altri host sulla rete locale.

Costruzione della tabella di routing statico (1)

- Per aggiungere o cancellare righe nella tabella di routing si usa il comando UNIX route.

  # route add 26.0.0.0 128.66.12.1 1

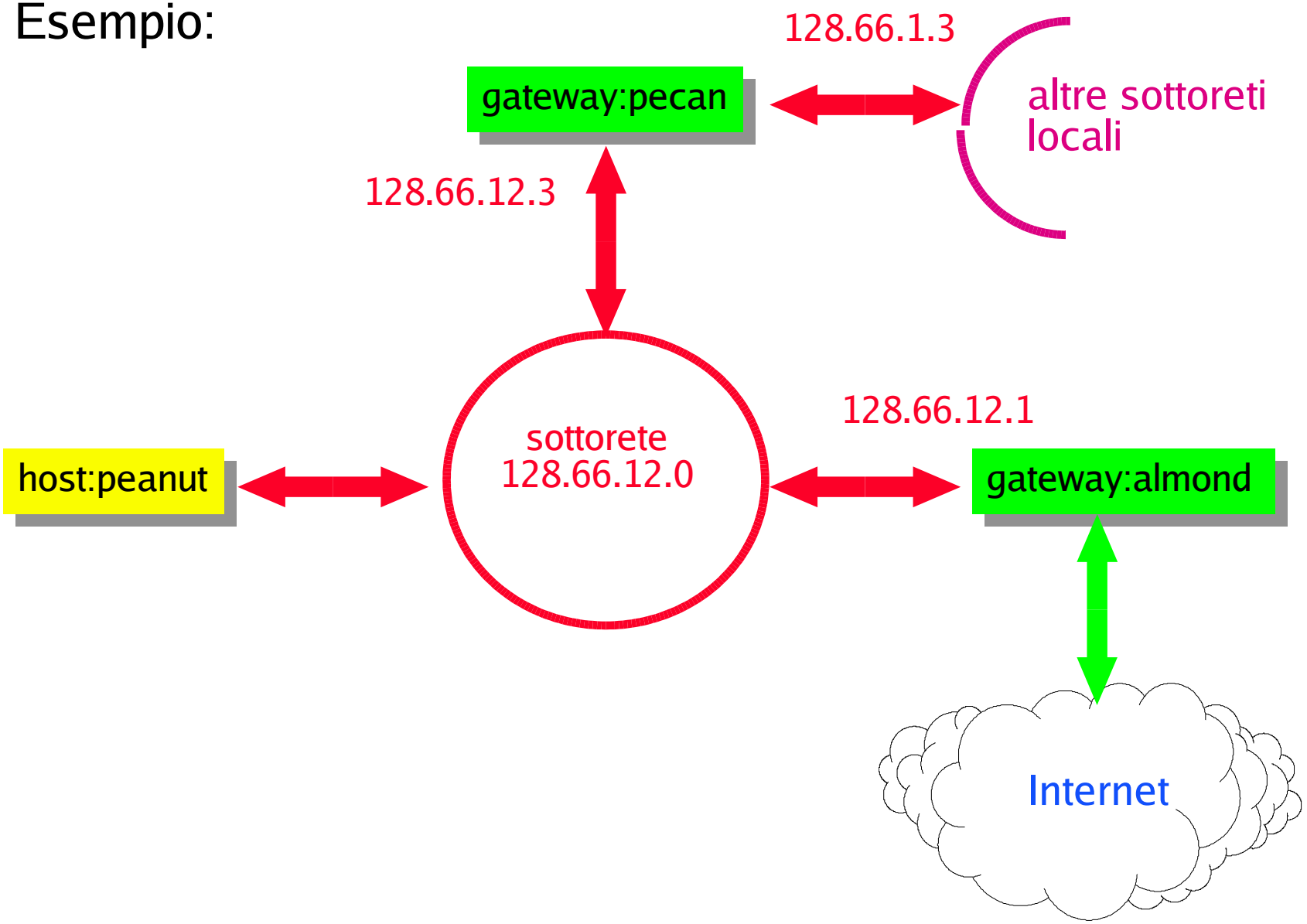
  add net 26.0.0.0: gateway almond

  aggiunge una route sulla tabella di routing di almond per la rete 26.0.0.0

  - il primo argomento dopo il comando route può essere o add o delete per aggiungere o togliere una route
  - il secondo valore è l'indirizzo di destinazione: può essere un indirizzo IP o un nome di rete preso dal file /etc/networks o un nome di host preso da /etc/hosts o la parola riservata **default**.
  - il terzo valore è l'indirizzo del gateway, cioè è un indirizzo IP del gateway esterno tramite il quale i dati vengono inviati a destinazione, deve essere l'indirizzo di un gateway su una rete connessa direttamente
  - l'ultimo parametro è il routing metric (non è usato con delete): se metric è 0, la route è installata come attraverso una interfaccia locale (non c'è il flag G), mentre se è >0 l'indirizzo del gateway è esterno (c'è il flag G)

Costruzione della tabella di routing statico (2)

- Esempio:



Costruzione della tabella di routing statico (2)

Per aggiungere la default route su peanut:

 # route -n add default 128.66.12.1 1

add net default: gateway 128.66.12.1

Dopo aver configurato la default route se si fa:

 # netstat -rn

Routing tables

Destination	Gateway	Flags	Refcnt	Use	Interface
127.0.0.1	127.0.0.1	UH	1	298	lo0
default	128.66.12.1	UG	0	0	le0
128.66.12.0	128.66.12.2	U	26	50360	le0

Se si prova un **ping 26.40.0.17** : funziona regolarmente per comunicare con host remoti.

Analogamente un **ping 128.66.1.2** : funziona per comunicare con host su altre sottoreti.

Ogni dato destinato ad altre sottoreti viene prima inviato ad almond, poi eventualmente instradato su pecan, nel qual caso almond invia a peanut un ICMP Redirect per l'utilizzo di pecan aggiornando così la tabella:

Destination	Gateway	Flags	Refcnt	Use	Interface
127.0.0.1	127.0.0.1	UH	1	298	lo0
128.66.1.3	128.66.12.3	UGHD	0	514	le0
default	128.66.12.1	UG	0	0	le0
128.66.12.0	128.66.12.2	U	26	50360	le0

Costruzione della tabella di routing statico (3)

Per evitare i ripetuti ICMP Redirect occorre specificare le route per ogni sottorete installata per es. usando i comandi:

 # route -n add 128.66.1.0 128.66.12.3 1
add net 128.66.1.0: gateway 128.66.12.3
 # route -n add 128.66.6.0 128.66.12.3 1
add net 128.66.6.0: gateway 128.66.12.3
 # route -n add 128.66.3.0 128.66.12.3 1
add net 128.66.3.0: gateway 128.66.12.3
 # route -n add 128.66.9.0 128.66.12.3 1
add net 128.66.9.0: gateway 128.66.12.3

aggiungendo quindi le righe alla tabella di routing:

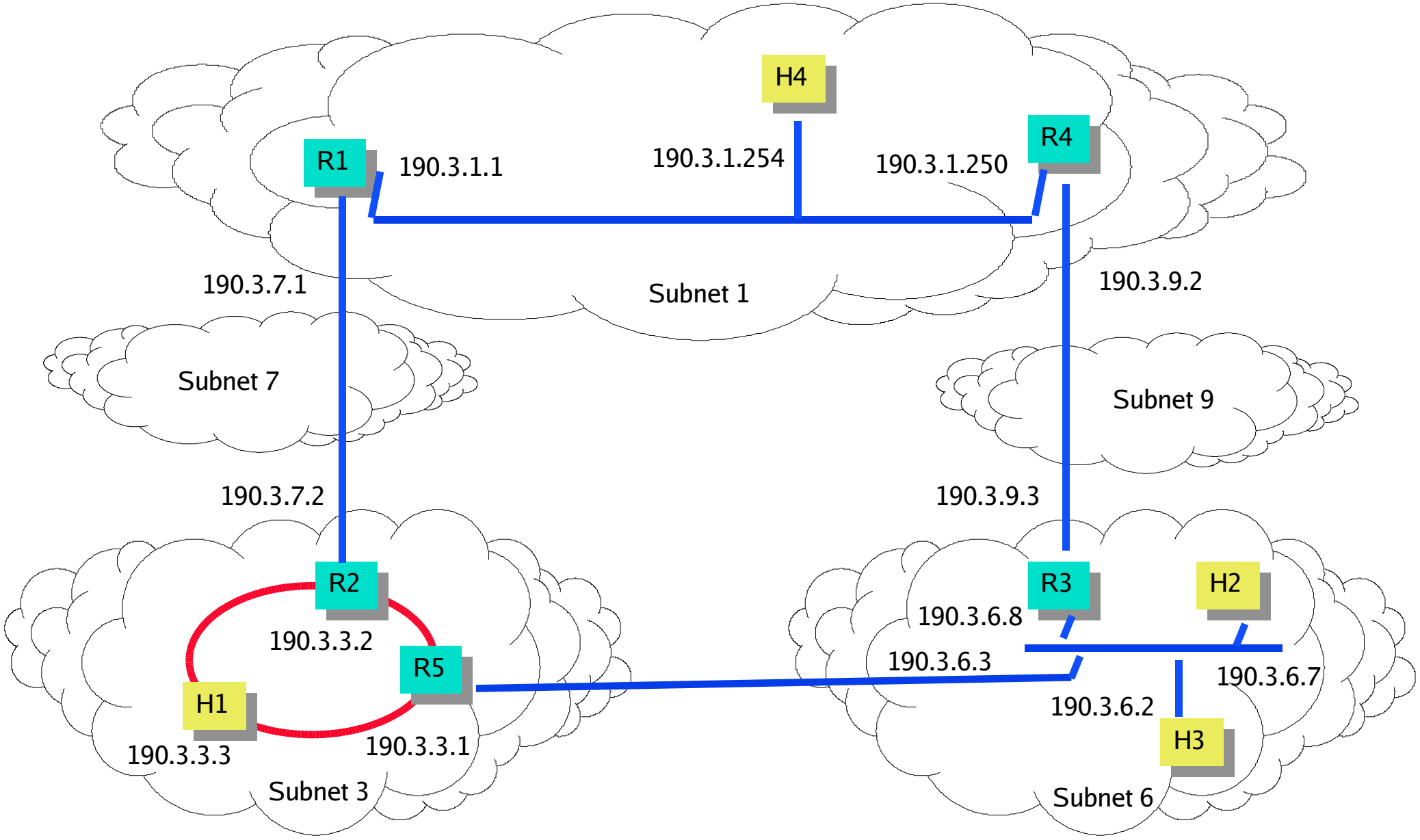
Destination	Gateway	Flags	Refcnt	Use	Interface
127.0.0.1	127.0.0.1	UH	1	298	lo0
128.66.1.3	128.66.12.3	UGHD	0	514	le0
default	128.66.12.1	UG	0	0	le0
128.66.12.0	128.66.12.2	U	26	50360	le0
128.66.1.0	128.66.12.3	UG	1	444	le0
128.66.3.0	128.66.12.3	UG	0	0	le0
128.66.6.0	128.66.12.3	UG	0	0	le0
128.66.9.0	128.66.12.3	UG	0	0	le0

Costruzione della tabella di routing statico (4)

- Per ogni sottorete aggiunta alla rete occorre aggiungere manualmente le route per la sottorete nella tabella di routing.
- Ad ogni reboot del sistema tutte le informazioni messe nella tabella di routing statico vengono perse, occorre quindi provvedere ad installarle nuovamente. Per esempio su un sistema BSD UNIX inserendo in rc.local i comandi di route:

```
route -n add default  128.66.12.1 1 > /dev/console
route -n add 128.66.1.0  128.66.12.3 1 > /dev/console
route -n add 128.66.3.0  128.66.12.3 1 > /dev/console
route -n add 128.66.6.0  128.66.12.3 1 > /dev/console
route -n add 128.66.9.0  128.66.12.3 1 > /dev/console
```

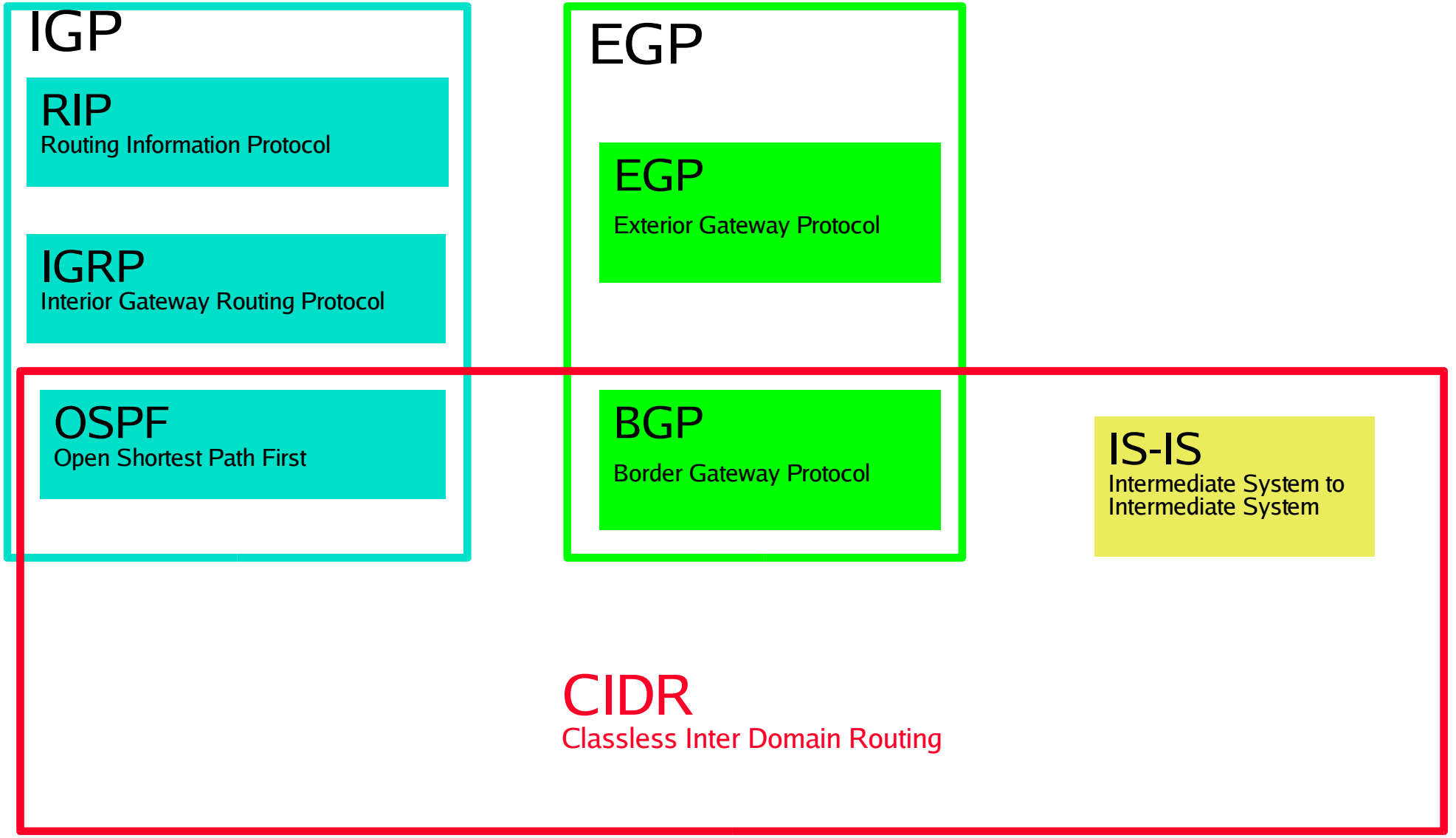
Esempio: Predisporre un file di tipo rc.local per la definizione della tabella di routing statico per il router R5 della rete in figura



I protocolli di routing (1)

- I router che instradano messaggi all'interno dello stesso AS sono detti Interior Router, mentre quelli che instradano messaggi anche tra AS diversi sono detti Exterior Router.
  - Gli Interior Router scambiano informazioni tramite un IGP (Interior Gateway Protocol)
  - Gli Exterior Router comunicano invece tramite un EGP (Exterior Gateway Protocol)

Protocolli di routing (2)



Routing Information Protocol (1)

- In molti sistemi UNIX il RIP funziona facendo "girare" il daemon routed.
- Per far girare il demone routed basta dare:

  # routed

  il comando può avere anche l'opzione -q quando il computer su cui gira non è un gateway, tale opzione previene infatti dal consigliare route, prende solo in considerazione route consigliate da altri sistemi
- Per far partire il demone in fase di startup basta includerlo in uno nei noti file di startup.

Routing Information Protocol (2)

- Routed legge alla partenza /etc/gateways e aggiunge le informazioni ivi contenute alla tabella di routing, quindi può essere utile per definire una route di default attiva. Se riprendiamo l'esempio precedente basterà specificare per l'host peanut almond come gateway di default inserendo in /etc/gateways:

  net 0.0.0.0 gateway 128.66.12.1 metric 1 active

  dove:

  il primo parametro può essere o **net** o **host**,

  il secondo indirizzo è usato per indicare la **default route**

  la terza è una parola chiave **gateway** seguita come quarto dall'**indirizzo del gateway**

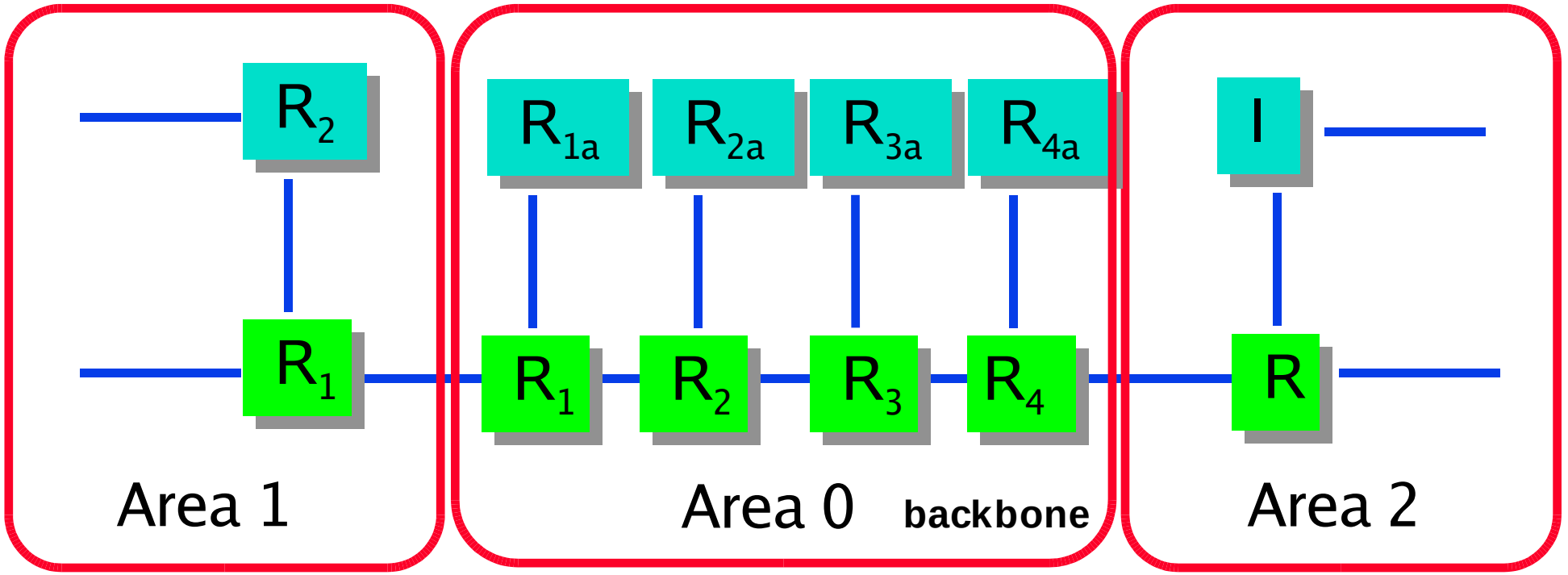
  la quinta è la parola chiave **metric** seguita dal sesto valore che è un numero che rappresenta il numero di gateway attraversati per giungere alla destinazione, ma è comunque un valore arbitrario che l'amministratore può assegnare

  il settimo parametro può essere o **active** o **passive** ad indicare se questa definizione può (active) o no (passive) essere cambiata da RIP in base ad aggiornamenti di routing esterni. Nel caso sia active e non si ricevano per tale gateway nessun aggiornamento la route può essere automaticamente cancellata, per prevenire ciò accorre definirla passive.

Open Shortest Path First (OSPF)

- Sviluppato nel 1988 da IETF, standard nel 1990 (RFC 1247) per routing all'interno di un AS
- OSPF:
  - **è aperto**
  - **supporta subnet variabili**
  - **implementa routing dinamico**
  - **supporta routing in base al tipo di servizio**
  - **esegue il bilanciamento del carico**
  - **supporta sistemi gerarchici (aree)**

Open Shortest Path First (OSPF) (2)



Exterior Gateway Protocol (1)

- EGP è un protocollo per lo scambio di informazioni con gateway di altri autonomous systems (AS).
- Messaggi di tipo Hello e I-Heard-You (I-H-U) sono speciali pacchetti usati per stabilire un dialogo tra due gateway che parlano EGP.
- Computer che parlano in EGP sono chiamati EGP neighbors (vicini) e lo scambio di messaggi Hello e I-H-U è detto acquiring a neighbor (procurarsi un vicino)

Exterior Gateway Protocol (2)

- EGP "gira"
  - o come processo separato usando EGP User Process (egpup)
  - o come parte del Gateway Routing Daemon (gated).
- Usate gated per far girare EGP non usate egpup.

Exterior Gateway Protocol (3)

- Quando egpup è partito legge un file in genere chiamato /etc/egp.init o /etc/egp.conf, in tale file possono apparire i seguenti comandi di configurazione:
  - **autonomoussystem** asn
    dove asn è il numero ufficiale dell'AS
  - **egpneighbor** neighbor
    dove neighbor è un host name o un IP address del gateway remoto che è da considerarsi come EGP neighbor
  - **egpmaxacquire** number
    dove number specifica il numero massimo di vicini che si possono procurare.
  - **egpnetsreachable** net1 net2 net3 ...
    che permette di definire le reti che si consigliano come raggiungibili ai propri vicini EGP
  - **net** destination **gateway** address **metric** number
    è il comando per installare una static route (la route è sempre passive)
  - **defaultgateway** address
    è il comando per installare una default route di tipo active.

Border gateway Protocol (BGP)

- E' il protocollo di routing usato per la comunicazione tra **AS (RFC 1654)**
  - Si basa su un algoritmo vettore-distanza evoluto
  - Si occupa del transito di dati di terze parti su una certa rete. Le reti vengono suddivise in:
    - reti stub (unica connessione al grafo BGP)
    - reti multiconnesse (usate per il traffico in transito)
    - reti di transito (sono disponibili al transito di traffico di terze parti, sono spesso reti di tipo backbone)

Gateway Routing Daemon (1)

- gated è un unico pacchetto che combina RIP, Hello, BGP ed EGP.
- I protocolli di routing in gated sono compatibili con gli stessi protocolli forniti in altre implementazioni (RIP in gated è uguale a RIP in routed, EGP di gated è equivalente a EGP di egpup)

Gateway Routing Daemon (2)

- Tra le ragioni che ci fanno scegliere gated ricordiamo:
  - in sistemi su cui gira più di un protocollo di routing, gated elabora le informazioni date dai vari protocolli e seleziona le route migliori
  - le route apprese da protocolli di routing di tipo interior possono essere annunciate tramite protocolli di tipo exterior
  - gated semplifica la configurazione del routing (tutti i protocolli sono configurati in un solo file /etc/gated.conf usando un'unica sintassi)
  - gated è costantemente aggiornato. Se si hanno vari protocolli l'uso di gated ci assicura che stiamo sempre utilizzando l'ultima versione software.

Gateway Routing Daemon (3)

- Per i quattro protocolli attualmente implementati in gated esiste una tabella di routing metric per decidere la bontà di una route:

Protocollo	Significato di metric	Range	Introvabile
RIP	distanza (hop-count)	0-15	16
Hello	ritardo in millisecondi	0-29999	30000
BGP	non specificato	0-65534	65535
EGP	distanza (non usata)	0-254	255

- gated usa questi metric quando comunica una route attraverso uno specifico protocollo (con RIP userà un metric RIP, con Hello un metrico Hello, ecc)

Gateway Routing Daemon (4)

- al contrario quando gated riceve route consigliate da altri, elabora questi consigli in base al protocollo di routing che ha inviato la route.
- I valori associati ai vari protocolli sono chiamati preference e i valori di default sono:

Tipo di route	**Preference di default**	
direct route	0	(maggiore preferenza)
ICMP redirect	20	
static route	50	
Hello	90	
RIP	100	
BGP	150	
EGP	200	
	255	(minore preferenza)

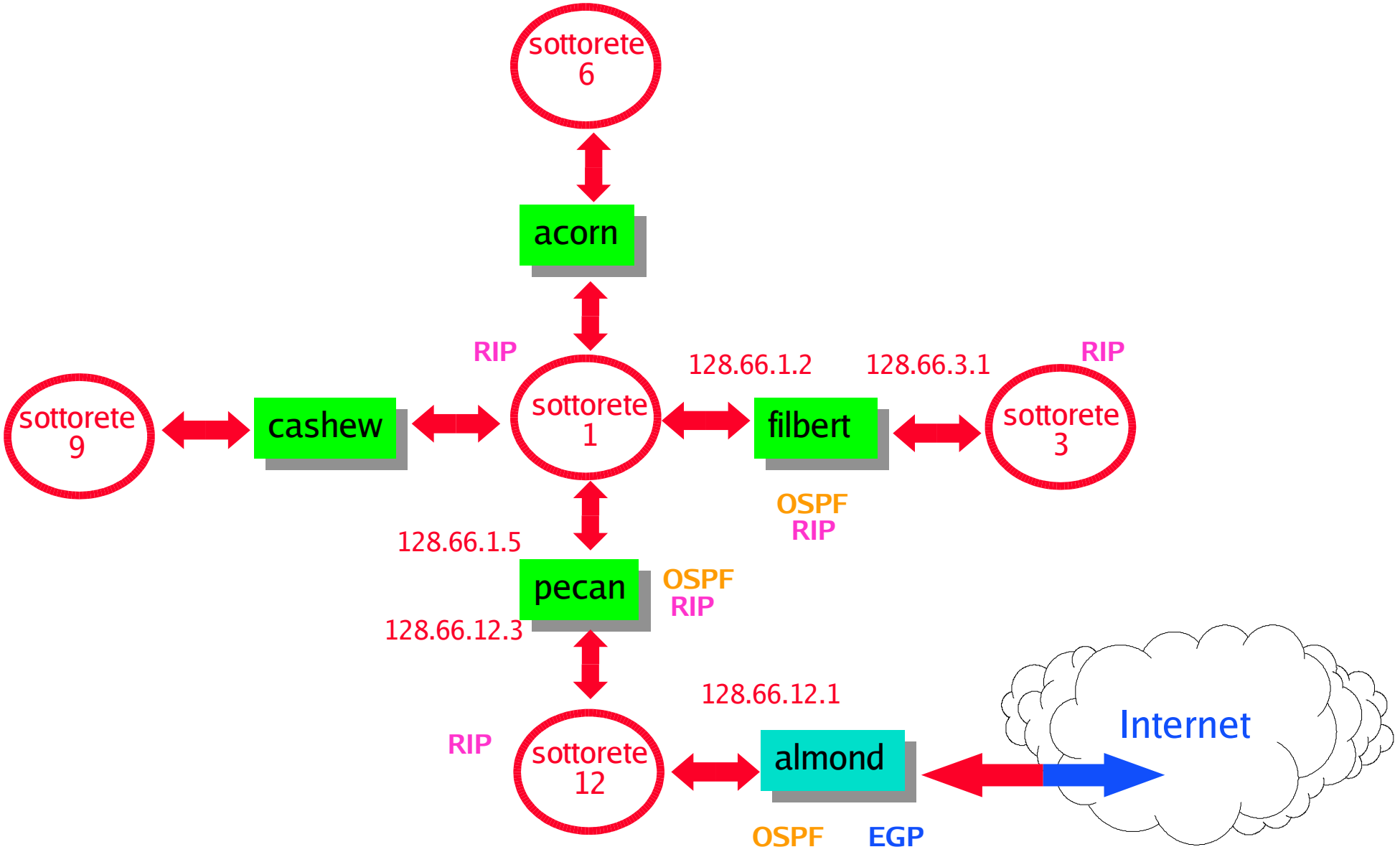
Configurazione di gated (1)

- La sintassi del linguaggio di configurazione di gated è molto simile a quella del linguaggio C. Tutti i comandi finiscono con ";" e comandi associati sono raggruppati insieme da parentesi graffe.
- Il file di configurazione gated.conf viene diviso in quattro sezioni:
  - comandi di definizione
  - comandi di protocollo
  - comandi statici
  - comandi di controllo
- Esistono altri due tipi di comandi che possono essere presenti in /etc/gated.conf, ma che non sono indispensabili per la configurazione di un protocollo, e sono:
  - comandi di direttive
  - comandi di trace

Configurazione di gated (2)

Comando	Tipo	Funzione
%directory	directive	imposta la directory per i file in include
%include	directive	include un file in gate.conf
traceoptions	trace	spec. quali eventi deve consid. il trace
options	definition	definisce le opzioni di gated
autonomoussystem	definition	definisce il numero dell'AS
interface	definition	definisce le opzioni dell'interfaccia
martians	definition	definisce gli indirizzi di dest. non validi
snmp	protocol	abilita il reporting a SNMP
Rip	protocol	abilita RIP
ospf	protocol	abilita OSPF
hello	protocol	abilita il protocollo Hello
redirect	protocol	rimuove route installate da ICMP
EGP	protocol	abilita EGP
bgp	protocol	abilita BGP
static	static	definisce route statiche
import	control	definisce quali route sono importate
export	control	definisce quali route vengono esportate
aggregate	control	definisce le route da aggregare
generate	control	definisce le route da generare

Esempio di conf. di routing: Schema della rete



Esempio di conf. di routing: conf. di peanut

- Configurazione di un host (peanut):

```
rip yes{
    nobroadcast;
    interface 128.66.12.2
            version 2
            multicast
            authentication simple “RIPauth”;
};
```

- `nobroadcast` indica a gated di non esportare in broadcast gli update RIP
- `interface` … gated si pone in ascolto sulla interfaccia 128.66.12.2 sull’indirizzo di multicast per aggiornamenti della tab. di routing di RIP versione 2 e provvede a verificarne l’autenticità con un stringa passata in chiaro (simple).

Esempio di conf. di routing: conf. di filbert (1)

- Configurazione dell'Interior Gateway filbert:

```
# evita di chiudere per timeout l'accesso alla sottorete 3
interfaces {
     interface 128.66.3.1 passive;
}
#imposta il proprio id per il protocollo OSPF
routerid 128.66.1.2;

#abilita RIP-2 su tutte le sue interfaccie
rip yes {
     broadcast ;
     interface all {
            version 2
            multicast
            authentication simple "RIPauth";
     };
};
...
```

Esempio di conf. di routing:
conf. di filbert (2)

```
...
# attiviamo OSPF
ospf yes{
     #definizione della stub-area dall'interf. 128.66.1.2
     area 1 {
             stub ;
             authtype simple;
             interface 128.66.1.2 {
                     authkey "OSPFauth" ;
                     priority 5;
             };
             # network della Stub area 1
             networks {
                     128.66.1.0 ;
                     128.66.3.0 ;
                     128.66.6.0 ;
                     128.66.6.0 ;
             };
     };
};
...
```

Esempio di conf. di routing:
conf. di filbert (3)

```
...
# esportazione delle route OSPF al protocollo RIP
export proto rip {
   interface all;
   proto direct;
   proto ospf ;
};

# esportazione della rete 3 connessa direttamente
export proto ospf metric 0 {
   proto direct interface 128.66.3.1 {
        network 128.66.3.0 };
};
```

Esempio di conf. di routing:
conf. di filbert (4)

- Il router filbert utilizza sia il protocollo OSPF per reperire routes dagli altri router OSPF, sia RIP-2 per propagare le informazioni agli host della sottorete 3
- L'interfaccia cui si fa riferimento per gli statement OSPF è quella definita dalla direttiva `routerid` 122.68.1.2
- Nella sezione OSPF viene dichiarata l'area 1 come area stub: tutti i router della medesima area la devono definire come stub.
- Gli statement `export` migrano le informazioni di routing tra i vari protocolli attraverso le diverse interfacce a disposizione specificando dove occorre una *route-filter* per esportare solo le informazioni di routing riguardanti la sottorete 3

Esempio di conf. di routing:
conf. di pecan (1)

- **Configurazione dell'Interior Gateway pecan:**

```
# interfaccia predefinita per router OSPF
   routerid 128.66.12.3 ;

   #attivare RIP
   rip yes{
        broadcast ;
        version 2;
        # pecan esporta in rip i routing OSPF
        interface 128.66.12.3
                noripin
                authentication simple "RIPauth";
        interface 128.66.1.5
                noripin
                authentication simple "RIPauth";
};
…
```

Esempio di conf. di routing:
conf. di pecan (2)

```
   # attiviamo OSPF su entrambe le interfacce verso la 12 e la 1
   ospf yes{
        backbone{# definizione della BB 12
                authtype simple ;
                interface 128.66.12.3 {
                        atuthkey "OSPFauth"; priority 5;
        };
   };
   #definizione della stub-area  raggiungibile dall'interf. 128.66.1.5
   area 1 {
        stub ;
        authtype simple;
        interface 128.66.1.5 {authkey "OSPFauth" ;priority 5;};
        # network della Stub area 1
        networks {128.66.1.0 ; 128.66.3.0 ; 128.66.6.0 ; 128.66.6.0 ;};
   };
};
```

Esempio di conf. di routing:
conf. di pecan (3)

```
# esportazione delle route OSPF al protocollo RIP
export proto rip {
   interface all;
   proto direct;
   proto ospf ;
};
```

- La direttiva norip in sulle due interfacce serve a non aggiornare le routing tables in base agli aggiornamenti provenienti dal protocollo rip
- La sottorete 12 viene definita come backbone con accesso tramite l'inerfaccia 128.66.12.3
- Deve essere definita anche la stub-area delle reti 1,3,…

Esempio di conf. di routing:
conf. di almond (1)

- **Configurazione dell'exterior gateway almond:**

```
# definizione del proprio AS number
autonomoussystem 249;
# opzioni per la generazione del default
options gendefault;

#definizione del router ID OSPF
routerid 128.66.12.1 ;

# disabilitare RIP
rip no ;

# abilitare OSPF
ospf yes {
      # dich. della BB-area verso la rete interna
      backbone {
               authtype simple ;
               interface 128.66.12.1 {atuthkey "OSPFauth";priority 5;};
      };
};
...
```

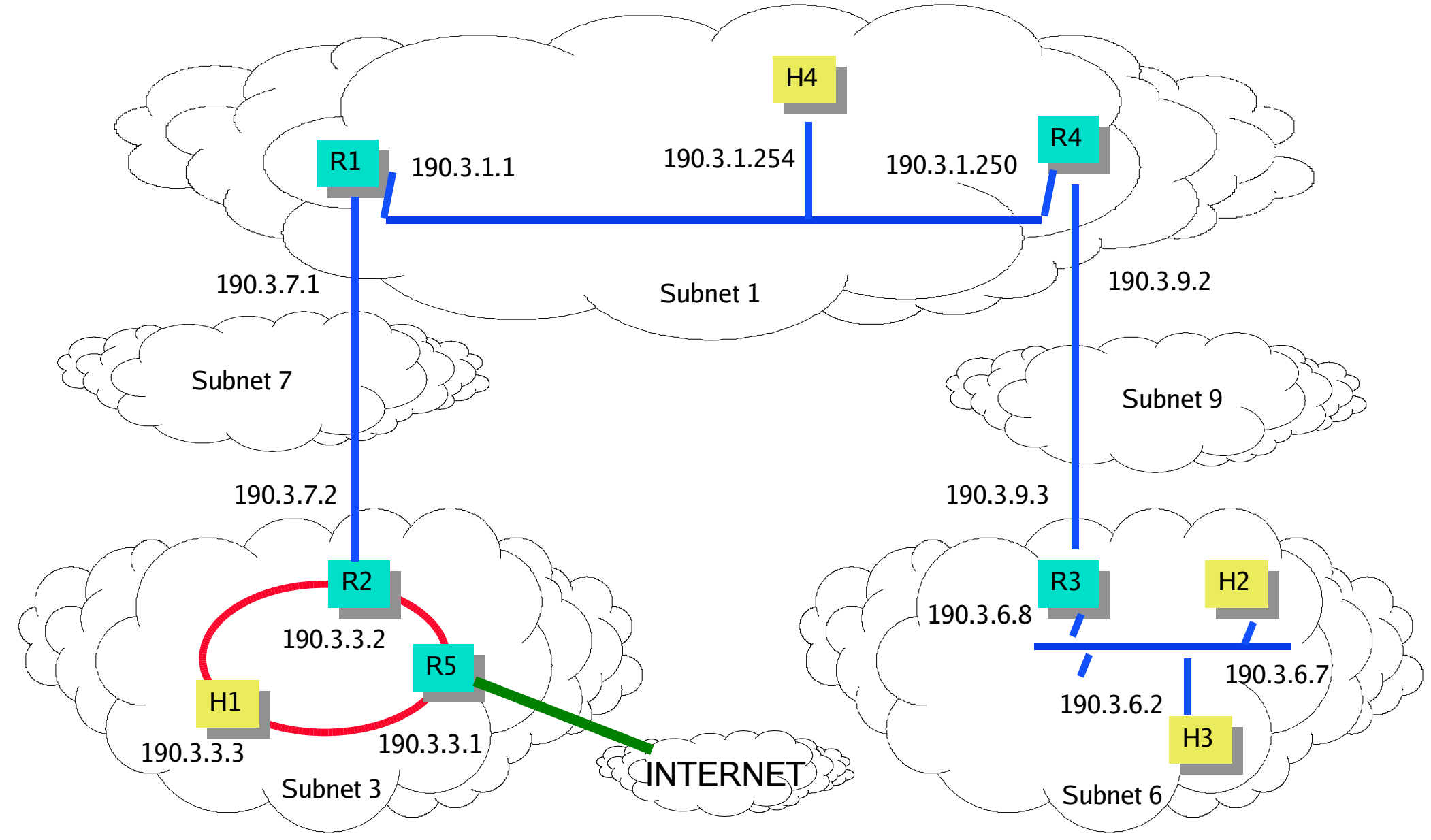
Esempio di conf. di routing:
conf. di almond (2)

```
...
# abilitare egp
egp yes {
     paketsize 12288 ;
     # dichiarazione di un gruppo di router EGP vicini
     group minhello 2:30 minpoll 10:00 {
             neighbor 26.6.0.103 ;
             neighbor 26.20.0.72 ;
     };
};

# esportare le route apprese da altri AS all'interno
export proto ospf {
     proto egp as 249 {all};
};
# esportare le route da OSPF a EGP
export proto egp as 164 {
     proto direct;
     proto ospf {network 128.66.0.0 ; };
};
```

Esempio di conf. di routing: conf. di almond (3)

- option gendefault invita gated a generare una route di default quando il sistema acquisisce un EGP neighbor. La route di default viene usata poi anche nell'ultimo comando propagate per mandarla alla rete 128.66.12.0
- autonomoussystem è necessario perché si usa un protocollo EGP e in particolare fornisce il numero dell'AS.
- egp yes abilita il protocollo EGP con però alcuni parametri addizionali:
  - packetsize 12288 abilita EGP ad accettare pacchetti di aggiornamento lunghi fino a 12288 byte
  - group permette di impostare i parametri per tutti gli EGP neighbor del gruppo, tra i parametri abbiamo:
    - minhello per impostare l'intervallo tra le trasmissioni di pacchetti di Hello
    - minpoll per impostare l'intervallo tra i poll
- Il primo export dice a gated di usare EGP per annunciare la rete 128.66.0.0 (nuts-net) ad Internet. Da notare che non è un indirizzo di sottorete, ma è l'indirizzo di una intera rete di classe B. as 164 è AS di Milnet non di nuts-net.

Esempio (2):


H4
R1
190.3.1.1
190.3.1.254
190.3.1.250
R4
190.3.7.1
Subnet 1
190.3.9.2
Subnet 7
Subnet 9
190.3.7.2
190.3.9.3
R2
R3
H2
190.3.6.8
190.3.3.2
R5
190.3.6.7
H1
190.3.6.2
H3
190.3.3.3
190.3.3.1
INTERNET
Subnet 3
Subnet 6

Esercizio

- Nella rete mostrata in figura, configurare gated per i seguenti host/router utilizzando gli adeguati protocolli di routing dinamico (RIP,OSPF,EGP):
  - H1
  - R3
  - R1
  - R5

Il comando gated (1)

- Il software gated viene fornito con alcuni sistemi UNIX, ma è comunque possibile ottenere l'ultima versione via anonymous ftp da gated.cornell.edu . gated è memorizzato nella directory /pub/gated in un file di tipo tar compresso che contiene il codice sorgente C e il Makefile necessario per compilarlo.
- Una volta preso e compilato è possibile utilizzare il comando gated

Il comando gated (2)

- Il comando gated ha molte opzioni tra le più diffuse ricordiamo:
  - -t serve per riportare su un file di trace errori interni, esterni e cambiamenti della routing table.
  - -c obbliga gated a leggere il file di configurazione e a controllare se ci sono errori di sintassi.
  - -n dice a gated di non aggiornare la tabella di routing del kernel.
  - -f config_file dice a gated dove leggere il file di configurazione: è da specificare se è diverso da /etc/gated.conf . Se associata all'opzione -c permette di testare una nuova configurazione senza interferire con la configurazione corrente.
    - es: gated -c -f test.conf trace.test

Il comando gated (3)

- Per far partire il demone gated in fase di startup basta inserire il comando in uno dei file di startup del sistema, con l'avvertenza di controllare che in detti file non ci siano comandi di tipo routed: gated e routed non possono girare contemporaneamente, eventualmente commentare i comandi di start di routed.
- Il codice gated è di solito installato nella directory /etc quindi un tipico comando per far partire gated da /etc/rc.local può essere:

```
if [ -f /etc/gated -a -f /etc/gated.conf ] then
    gated;    echo -n 'gated' > /dev/console
fi
```