Reti e Linux

Andrea Bontempi

POuL

Corsi Linux 2012

Una breve introduzione: le reti

- Una rete di calcolatori è un mezzo fisico sul quale è possibile inviare e ricevere messaggi o flussi di dati.
- La prima rete a commutazione di pacchetto degna di nota è stata ARPANET (1969) con lo scopo di creare un sistema di telecomunicazione resistente agli attacchi nucleari (Siamo in piena guerra fredda). Ogni pacchetto sarebbe stato instradato fino a destinazione seguendo una delle strade disponibili: se fosse saltato un collegamento un'altro avrebbe immediatamente preso il suo posto.

Una breve introduzione: Internet

- Ad oggi per "Rete" si intende Internet il più grande ed esteso sistema di telecomunicazione esistente.
- Internet non è una rete vera e propria, ma un insieme di reti connesse tra di loro (*Interconnected Networks*).
- La sua struttura non è stata progettata, è nata dall'interazione delle singole reti che compongono Internet.
- Per come è stata concepita, Internet non ha proprietari, è incontrollabile e intrinsecamente resistente. Nonostante molti continuino a tentare (con risultati altalenanti) di cambiare la sua natura.

La pila ISO/OSI

Iproute2

- Iproute2 è un tool di gestione della rete sotto GNU/Linux che sostituisce i vecchi comandi *net-tools*.
- Lavora in *user space* e comunica al kernel la configurazione da applicare secondo il protocollo standard *netlink*.
- Per rendere il passaggio il più possibile indolore l'uso del nuovo tool si concentra su un unico comando "IP" e li comandi sono classificati tramite uno stile ad albero gerarchico evidentemente ispirato a *Cisco IOS*.

Struttura del comando IP

ip *[OPTIONS]* OBJECT *{ COMMAND | help }*

Configurare le schede di rete

Gestione del livello data link

ip link *{ COMMAND | help }*

- *show* Visualizza lo stato delle interfacce di rete.
- *set dev DEVICE {up | down}* Accende o spegne una interfaccia.
- *set dev DEVICE address MAC* Cambia l'indirizzo mac di una interfaccia. (L'interfaccia deve essere spenta)
- *set dev DEVICE arp {on | off}* Abilita o disabilita il servizio ARP su una interfaccia. (Normalmente va tenuta abilitata)

NB: con *ip neigh show* è possibile vedere la tabella ARP del kernel nel momento in cui lanciate il comando.

Connettersi via cavo

Gestione del livello di rete

ip addr *{ COMMAND | help }*

- *show* Visualizza gli indirizzi assegnati alle interfacce di rete.
- *{add | del} ADDRESS/NETMASK dev DEVICE* Aggiunge o rimuove un indirizzo IP ad una interfaccia, la netmask va inserita in "slash notation", una interfaccia può avere più indirizzi ip.

Esempio: Aggiungere e rimuovere un indirizzo IP

1) ip addr add 192.168.0.1/24 dev eth0
2) ip addr del 192.168.0.1/24 dev eth0

Introduzione al routing

- Nel kernel è presenta una tabella di routing che viene consultata per capire dove mandare un pacchetto in base alla sua destinazione.
- Questa è solitamente molto semplice, poiché normalmente un computer ha un unica connessione attiva per volta. Ma non è sempre così.
- Possiamo andare a modificare la tabella per ottenere instradamenti voluti in determinate condizioni. Per esempio possiamo volere una rete isolata con indirizzi 192.168.0.0/24 connessa via cavo ethernet, ma continuare a navigare su internet tramite il Wi-Fi.

Gestione del routing

Gestione del routing di sistema

ip route *{ COMMAND | help }*

- *list* Visualizza le attuali regole nella tabella di routing.
- *get ADDRESS/NETMASK* Visualizza l'attuale indirizzamento di un particolare indirizzo IP.
- *{add | del | change} ADDRESS/NETMASK [via GATEWAY] [dev DEVICE]* Aggiungi, rimuovi o modifica una regola.
  1. L'indirizzo deve indicare una intera rete.
  2. Se non viene specificato il gateway, verrà usato quello di default.
  3. In caso in cui un pacchetto rientri in più regole, vince la regola con la maschera di rete più lunga
  4. Aggiungendo il device possiamo forzare l'uscita su una specifica interfaccia.

Statistiche dei socket

Investigare nelle connessioni attive

ss [OPTIONS]

- -n Non risolve i nomi dei servizi (porte usate)
- -r Risolve gli host name.
- -a Visualizza tutti i socket.
- -l Visualizza solo i socket in ascolto.
- -e Visualizza maggiori informazioni sui socket.
- -i Visualizza informazioni TCP.
- -p Visualizza i processi che usano i socket.
- Filtri: –ipv4, –ipv6, –tcp, –udp, –dccp, –raw, –unix ...

Il coltellino svizzero: netcat

Il programma "nc" accetta diversi parametri per permettere una selezione delle sue funzionalità:

- *-l* Per accettare una connessione
- *-L ADDRESS:PORT* Per ridirezionare una connessione sulla porta dell'indirizzo dato.
- *-p* Per specificare una porta (Da usare solo insieme a -l e -L)
- *-u* Per creare una connessione UDP al posto della TCP.
- *ADDRESS PORT* Per connettersi ad un certo indirizzo ad una certa porta.

Esempio: creare una connessione TCP in loopback

1) nc -l -p *9999*
2) nc *127.0.0.1 9999*

Connettersi con SSH

ssh *[OPTIONS] USERNAME@ADDRESS*

- *-C* Comprime il flusso dati, risparmiando banda.
- *-X* X forwarding (Entrambi devono usare X11)
- *-N* Impedisce l'apertura della shell remota.
- *-f* Manda in background il processo ssh.
- *-N -L PORTX:ADDRESS:PORT* Redirige tutte le connessioni dirette alla PORTX (sul proprio computer) alla porta dell'indirizzo specificato, passando attraverso il computer a cui si è connessi via ssh.
- *-Ng -R *:PORTX:ADDRESS:PORT* Redirige tutte le connessioni in entrata sulla PORTX (del computer a cui ci connettiamo) sulla porta dell'indirizzo specificato.

Analisi del traffico con tcpdump

tcpdump *[OPTIONS] [FILTER BPF]*

- *-i DEVICE* Specifica l'interfaccia di rete su cui effettuare l'analisi del traffico.
- -w FILE Scrive i pacchetti letti su un file in formato .pcap
- -n Visualizza gli IP al posto dei nomi di dominio (velocizza l'esecuzione)
- *-Z USER* Specifica quale utente usare dopo l'inizializzazione (effettuata da root)

Esempio: Analizzare solo i pacchetti ICMP

tcpdump -i eth0 icmp

La suite IW

- Il comando "iw" permette la gestione delle schede di rete IEEE 802.11
- Ha la stessa filosofia e modalità d'uso di "ip", e vuole sostituire il vecchio comando "iwconfig".
- Non prende il posto di "ip", che rimane fondametale per la gestione di tutte le schede di rete.
- Lo scopo di "iw" è unicamente la configurazione del livello fisico del Wi-Fi.
- Così come "ip" utilizza una classificazione dei comandi ad albero gerarchico simile a *Cisco IOS.*

Struttura del comando IW

iw *[OPTIONS]* OBJECT *{ COMMAND | help }*

Connettersi al WiFi

Per connettersi ad una rete IEEE 802.11 é necessario conoscere alcune cose fondamentali:

- Il nome della rete a cui andremo a connetterci (che chiameremo *essid*)
- Se presente, il protocollo di cifratura e autenticazione che questa rete richiede.
- Una password o un certificato, in base al protocollo usato.

Effettuare una ricerca delle reti disponibili

iw dev DEVICE scan

Connettersi ad una rete aperta (senza cifratura e autenticazione)

iw DEVICE connect ESSID

Autenticazione WEP e WPA/WPA2

Esistono due tipi diversi di cifratura e autenticazione:

1. WEP Il più semplice ma anche il più facile da aggirare. Si basa su una sola password condivisa fra tutti. *E' fortemente sconsigliato.*
2. WPA/WPA2 Permette l'uso di diversi tipi di cigratura e autenticazione. E'possibile usare una password condivisa tra tutti gli utenti oppure sfruttare un sistema di autenticazione tramiti certificati personali. *La versione con WPA2-AES è considerata sicura.*

Collegarsi ad una rete protetta dal protocollo WEP

iw DEVICE connect ESSID keys PASSWORD

Collegarsi ad una rete protetta dal protocollo WPA/WPA2

wpa_passphrase "ESSID" "PASSWORD" > file.conf
wpa_supplicant -D wext -i DEVICE -c file.conf

DHCP

Il DHCP (Dynamic Host Configuration Protocol) è un protocollo ideato per l'auto-configurazione di un computer in una rete dopo che questo ha effettuato la connessione, per evitare che l'inserimento manuale di queste configurazioni crei degli errori. Il computer riceve da un server DHCP le seguenti informazioni:

- L'indirizzo IP che gli è stato assegnato.
- La maschera di rete usata.
- Il gateway di default.
- I server DNS consigliati.
- Altre informazioni usate solo in determinati casi specifici.

Usare un client DHCP per ricevere la configurazione

```
dhclient DEVICE
```

I server DNS

- I DNS (Domain Name System) sono dei server fondamentali per Internet, hanno il compito di tradurre le stringhe di testo chiamate "nomi di dominio" in indirizzi IP che possono essere usati per la connessione effettiva.
- Sono solitamente gestiti da un ISP (Internet Service Provider), ma esistono server indipendenti.
- La struttura di questi server è fortemente gerarchica, e insieme formano un database distribuito che contiene la traduzione di tutti i nomi di dominio nei relativi indirizzi IP.
- Per sapere a chi appartiene un particolare dominio si può fare una ricerca su uno dei tanti servizi WHOIS presenti in rete.

Per risolvere manualmente un nome di dominio

```
nslookup HOSTNAME
```

Gestione dei DNS

- Il primo livello dei nomi di dominio è riservato ai paesi o a particolari "gruppi", e sono i comuni: *.it*, *.com*, *.org*, *ecc...*
- I domini di secondo livello possono essere assegnati pagando il "gestore" del dominio di livello superiore. Ad esempio se volessi comprare il dominio *pippo.it* dovrei richiedere al gestore del dominio .it di assegnarmelo (se non lo è già) pagando una somma annuale. (DI solito si usano degli intermediari).
- Una volta "proprietari" di un determinato nome di dominio di livello 2 possiamo creare nomi di dominio di livello 3. Nel nostro esempio possiamo quindi creare: *www.pippo.it, ftp.pippo.it*, *ecc...*
- Abbiamo la possibilità di creare nomi di livello superiore, senza limite, ad esempio *ciao.come.stai.pippo.it* è valido come tutti gli altri.

FIN

GRAZIE