

Networking

Configurazione essenziale dei servizi su Linux

Marco Prandini

Informazioni

- **La configurazione di un'interfaccia di rete richiede come minimo**
 - indirizzo IP
 - netmask
- **più, se la rete locale è interconnessa ad altre**
 - gateway specifici
 - default gateway
- **e, se è disponibile un sistema di risoluzione di nomi**
 - indirizzi dei server DNS
 - domini di default per costruire i FQDN

Metodi di configurazione

- Le informazioni possono essere assegnate
 - manualmente
 - da un server DHCP
- Attenzione alla dicotomia tipica runtime/persistenza
 - comandi per cambiare istantaneamente la configurazione
 - configurazione da applicare all'avvio
- Attenzione alle interfacce utente di configurazione
 - metodo classico: editing file di testo
 - segue l'approccio standard di qualsiasi servizio
 - modifiche non applicate fino a `systemctl restart networking` (o equivalenti)
 - metodo di default in molte distribuzioni di Linux: **NetworkManager**
 - se presente, può essere pilotato da GUI o `nmcli`
 - può sovrascrivere le modifiche runtime in qualsiasi momento

Configurazione runtime

- suite iproute2
 - comando ip + sottocomandi, sostituisce ifconfig e route
 - controllo completo di tutti gli aspetti più avanzati (link layer, interfacce virtuali, tunnel, VXLAN, policy-based routing, ...)
- sottocomando address (a)
 - visualizzazione `ip a`
 - assegnazione `ip a add <address>/<mask> dev <interface>`
 - rimozione `ip a del <address>/<mask> dev <interface>`
- sottocomando route (r)
 - visualizzazione `ip r`
 - routing via gw `ip r add <dst_net>/<mask> via <gw_addr>`
 - routing via dev `ip r add <dst_net>/<mask> dev <interface>`
 - rimozione `ip a del <address>/<mask>`

Tool di monitoraggio

■ Verifica dello stato delle connessioni

– **ss**

- **-t / -u** TCP/UDP only
- **-l / -a** stato LISTEN (il default è ESTABLISHED) / ALL
- **-n** non risolvere gli indirizzi/porte in nomi simbolici
- **-p** mostra processi che usano la socket

■ Intercettazione del contenuto dei pacchetti

– **tcpdump**

– **wireshark**

→ man pages e documentazione

Risoluzione dei nomi - generalità

■ La mappatura da nomi di host a indirizzi IP e viceversa è uno dei tanti casi in cui il sistema ha bisogno di un dizionario di nomi

■ Il primo accorgimento adottato da GNU/Linux riguarda la *scelta della sorgente di informazioni*

- *Name Service Switch*
- svolta dalla libreria C di sistema
- supporta un set fisso di possibili database
- configurata tramite **/etc/nsswitch.conf**
 - vedi man page omonima

NSS

■ Sintassi di `nsswitch.conf`

- `<entry> ::= <database> ":" [<source> [<criteria>]]*`
- `<criteria> ::= "[" <criteria> + "]"`
- `<criteria> ::= <status> "=" <action>`
- `<status> ::= "success" | "notfound" | "unavail" | "tryagain"`
- `<action> ::= "return" | "continue"`

risposta
ricevuta

la sorgente esiste
ma non sa rispondere

la sorgente esiste
ma è occupata

la sorgente non
è raggiungibile

■ Es.

```
passwd: files nis ldap
group: files ldap
hosts: ldap [NOTFOUND=return] dns files
```

(i colori indicano
l'azione di default)

Risoluzione dei nomi – host e IP

■ Es: `hosts: ldap [NOTFOUND=return] dns files`

■ `files` → la sorgente di informazioni è il file `/etc/hosts`

- formato: `<IP> <FQDN> [<ALIAS> ...]`

esempio: `8.8.8.8 dns.google.com gdns`

■ `dns` → la sorgente di informazioni è il sistema DNS

- l'interrogazione di server DNS è un'ulteriore set di funzioni della libreria C di sistema, il `resolver`

- si configura attraverso `/etc/resolv.conf`

- esempio

```
nameserver 137.204.58.1
```

```
domain disi.unibo.it
```

```
search ing.unibo.it
```

DNS caching

■ Spesso si trova un server DNS locale

- Miglioramento prestazioni
- Maggiore flessibilità per contesti dinamici

```
~$ cat /etc/resolv.conf
```

```
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 127.0.1.1
```

- Tutti gli IP che iniziano per 127 puntano a localhost

```
sudo ss -naup | grep 127.0.1.1:53
```

```
...
UNCONN 0          0          127.0.1.1:53   *:*      users:(("dnsmasq",pid=2154,fd=4))
```

Risoluzione di nomi via NSS

■ Il comando getent permette di interrogare i database del name service switch

```
getent <db name> <keyword>
```

■ Esempi:

```
- $ getent passwd las
```

```
las:x:1000:1000:Lab Amministrazione
Sistemi,,,:/home/las:/bin/bash
```

```
- $ getent hosts www.unibo.it
```

```
137.204.24.35   atrproxy.unibo.it www.unibo.it
```

Risoluzione nomi DNS diretta

- Per interrogare direttamente il DNS e avere più controllo sulle query si usano tipicamente `host` e `dig`
 - non considerano `nsswitch`
 - usano i `nameserver` di `resolv.conf` di default
 - possono interrogare un server specifico
- `host` (tipicamente per conversioni IP \longleftrightarrow nome)
 - query di un nome: `host www.unibo.it`
 - query a un server specifico: `host www.unibo.it 8.8.8.8`
- `dig` (tipicamente per ottenere informazioni legate a un dominio diverse da nomi `host`)
 - conoscere i Mail eXchanger: `dig mx example.com`
 - conoscere i Name Server: `dig ns example.com`

Sincronizzazione

- L'allineamento dell'ora di un sistema ad un orologio di riferimento è cruciale
 - per la diagnostica dei problemi (timestamp su log)
 - per i protocolli di autenticazione e autorizzazione (i messaggi hanno una vita limitata)
 - per la sincronizzazione di azioni distribuite
 - per il valore legale di azioni compiute attraverso i computer
- È possibile usare un protocollo che compensa i ritardi di rete per ottenere informazioni precise via Internet: *Network Time Protocol (NTP)*
 - sito ufficiale: <http://www.ntp.org/>
 - grande quantità di informazioni su: <http://www.eecis.udel.edu/~mills/ntp.html>

NTP in breve

■ Preciso

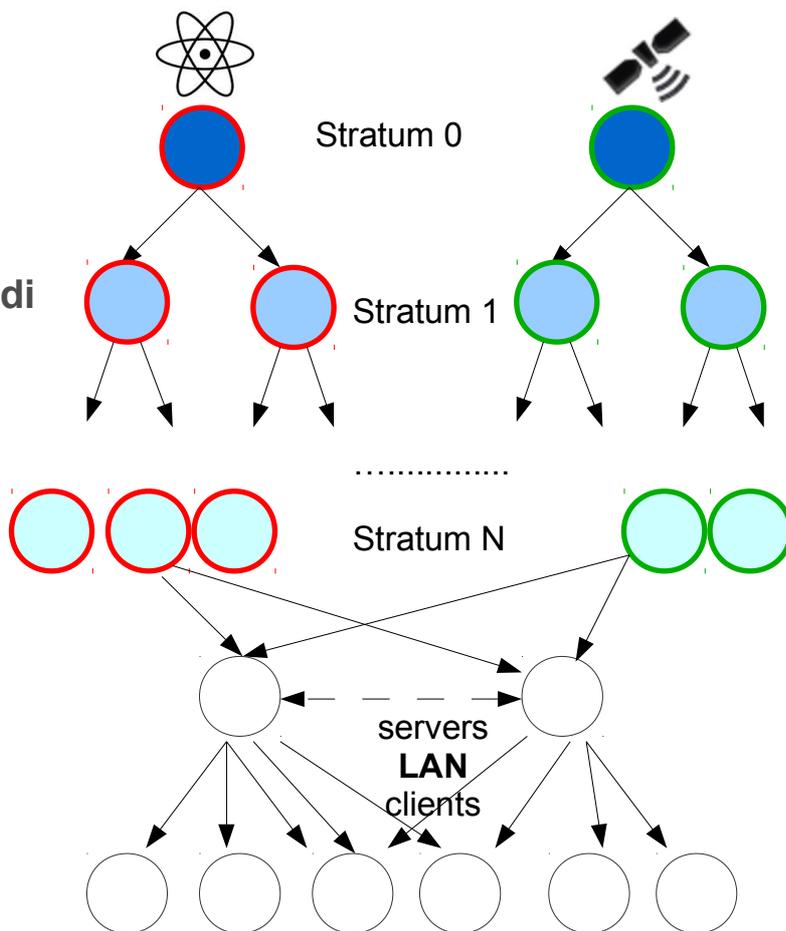
- poche decine di millisecondi di scarto su WAN
- <1 millisecondo su LAN
- supporto di sorgenti HW (oscillatori, GPS, ...)

■ Standard

- portato su ogni architettura nota

■ Scalabile e affidabile

- *multi-server*
- *strata*
- *peering*
- *auto-keying*



NTP su Linux

■ Il demone *ntpd* è client e/o server in funzione della configurazione

■ `/etc/ntp.conf` – esempio

```
server 0.ubuntu.pool.ntp.org
server 1.ubuntu.pool.ntp.org
peer fellow.server.lan
# By default, exchange time with everybody, but don't allow configuration.
restrict -4 default kod notrap nomodify nopeer noquery
restrict -6 default kod notrap nomodify nopeer noquery
# Local users may interrogate the ntp server more closely.
restrict 127.0.0.1
restrict ::1
```

NTP – inizializzazione e uso sporadico

- Il tool **ntpd** permette di sincronizzare l'orologio locale a un server NTP
 - senza parametri usa i server in **ntp.conf**
 - **ntpd** non deve essere attivo
 - accetta come parametro un server specifico

- L'ora viene modificata in due modi
 - se la differenza è più di 0.5 secondi: **step**
 - se la differenza è meno di 0.5 secondi: **slew** con **adjtime()**

- Non rimpiazza **ntpd**, che usa algoritmi sofisticati
 - per compensare errori e ritardi dei pacchetti dai server
 - per profilare il comportamento dell'orologio locale