

Configuración básica de la red

Departamento de Sistemas Telemáticos y Computación (GSyC)

<http://gsync.urjc.es>

Febrero de 2012



©2012 GSyC
Algunos derechos reservados.
Este trabajo se distribuye bajo la licencia
Creative Commons Attribution Share-Alike 3.0

Nombre de la propia máquina

Hay dos maneras de cambiar el nombre de la máquina

① Usar la orden *hostname*

```
hostname <nuevo_nombre_de_maquina>
```

El efecto es inmediato, pero se pierde al reiniciar

② Modificar un fichero de configuración

- /etc/HOSTNAME (Red Hat)
- /etc/hostname (Debian)
- /etc/config/system (OpenWrt)

El cambio es persistente, pero solo tiene efecto tras reiniciar

- El nombre cambiado de esta manera afecta al *prompt*, a la variable de entorno `HOSTNAME` y poco más
 - (El *prompt* es la línea que muestra la shell para indicar que está a la espera de órdenes)

```
milogin@zeta40:~$
```

- Los servicios de red no usan ni consultan este nombre, sino que resuelven nombres en `/etc/hosts` o en el DNS
- El usuario no suele percibir la diferencia porque lo normal y razonable es usar el mismo nombre en `/etc/hostname` y en `/etc/hosts`
 - Pero en una máquina a medio configurar podrían ser distintos

Nombres de otras máquinas: /etc/hosts

A principios de los años 80, cuando no existía el DNS, el fichero /etc/hosts era la única manera de resolver nombres de máquina

```
127.0.0.1      localhost
82.159.1.186  mi_pc
193.147.49.5   bilo.gsync.es      pantuflo
193.147.49.80  zeta20.aulas.gsync.es  zeta20
```

En la actualidad se sigue usando `/etc/hosts`, es útil para

- Resolver direcciones que no están en ningún servidor de DNS
- Resolver direcciones muy frecuentes sin usar su nombre de dominio completo (FQDN)

Este fichero tiene precedencia sobre el DNS

Algunas aplicaciones exigen que tanto localhost como el nombre de la máquina puedan resolverse

Su ubicación es la misma en prácticamente todas las distribuciones Linux

- Equivale a
 - `%SystemRoot%\system32\drivers\etc\hosts` (MS Windows)
 - `/private/etc/hosts` (Mac OS)

Servidores de DNS

- Dentro de la URJC
193.147.184.2
193.147.184.11
- En las redes domésticas normalmente podremos apuntar al router proporcionado por el ISP, que frecuentemente tiene la dirección
192.168.1.1
- En otro lugar, debemos averiguar la dirección del servidor de nuestro ISP
- También podemos emplear (en cualquier parte) los servidores de google
8.8.8.8
8.8.4.4

Interfaces de red

El Hardware de red puede ser muy variable, pero en TCP/IP se define un interfaz abstracto. Permite enviar y recibir paquetes siempre de la misma forma, con independencia del tipo de dispositivo

- Interfaz *loopback*: `lo`
- Interfaces ethernet: `eth0`, `eth1`, `eth2`,...
- Interfaces wifi: `wlan0`, `wlan1`...
Pueden tener otros nombres, dependiendo del driver
- Interfaces ppp (teléfonos fijos o móviles):
`ppp0`, `ppp1`, `ppp2`,...

Cada máquina normalmente tendrá varios interfaces

- Según el diseño original de IPv4, cada interfaz tiene exactamente 1 dirección
- En rigor, las máquinas no tienen dirección IP. (Pertenece a los interfaces)

El interfaz lo

En las máquinas tipo Unix al interfaz *loopback* se le suele llamar *lo*

- Es un interfaz especial, virtual. Se usa para establecer conexiones dentro de una misma máquina,
- En `/etc/hosts` tiene que haber una entrada para la dirección `127.0.0.1` asociada al nombre *localhost*
 - Que no debe mezclarse con la entrada para el nombre y dirección *ordinario*

Ejemplo correcto:

```
127.0.0.1      localhost
193.147.73.136 zeta40.aulas.gsync.es      zeta40
```

- **Ejemplo INCORRECTO**

```
127.0.0.1      localhost zeta40 # ESTO ES UN ERROR!!
```

- Las tramas con destino a *lo* no se envían a ningún dispositivo de red físico, permanecen dentro de la red interna de la máquina
- Es necesario que *lo* siempre esté activo, incluso cuando se identifique a la máquina con una dirección de un rango distinto
 - Ejemplo: Si desactivamos *lo* y desde una máquina con dirección 193.147.73.136 enviamos un *ping* a la propia 193.147.73.136 , no habrá respuesta

Nombre y estado de los interfaces

Para conocer los nombres de los interfaces reconocidos por el núcleo, podemos hacer búsquedas sobre los mensajes del proceso de arranque

```
dmesg | grep eth
```

```
dmesg | grep ath
```

```
dmesg | grep wlan
```

Para conocer el estado actual de los interfaces *activos* (solo de los activos) podemos usar la orden *ifconfig*, sin argumentos

```
koji@mazinger:~$ ifconfig
eth0  Link encap:Ethernet  direcciónHW 00:23:8b:26:b5:81
      Direc. inet:193.147.71.119  Difus.:193.147.71.127  Másc:255.255.
      Dirección inet6: fe80::223:8bff:fe26:b581/64 Alcance:Enlace
      ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST  MTU:1500  Métrica:1
      Paquetes RX:5321 errores:0 perdidos:0 overruns:0 frame:5
      Paquetes TX:82 errores:0 perdidos:0 overruns:0 carrier:0
      colisiones:0 long.colaTX:1000
      Bytes RX:438333 (438.3 KB)  TX bytes:13002 (13.0 KB)

eth1  Link encap:Ethernet  direcciónHW 00:21:00:8b:77:2a
      Dirección inet6: fe80::221:ff:fe8b:772a/64 Alcance:Enlace
      ACTIVO DIFUSIÓN MULTICAST  MTU:1500  Métrica:1
      Paquetes RX:0 errores:0 perdidos:0 overruns:0 frame:0
      Paquetes TX:0 errores:0 perdidos:0 overruns:0 carrier:0
      colisiones:0 long.colaTX:1000
      Bytes RX:0 (0.0 B)  TX bytes:0 (0.0 B)

lo    Link encap:Bucle local
      Direc. inet:127.0.0.1  Másc:255.0.0.0
      [...]
```

iwconfig

En el caso de interfaces inalámbricos, podemos tener información adicional con la orden `iwconfig`

```
koji@mazinger:~$ iwconfig
lo          no wireless extensions.

eth0       no wireless extensions.

eth1       IEEE 802.11  Access Point: Not-Associated
           Link Quality:5  Signal level:0  Noise level:0
           Rx invalid nwid:0  invalid crypt:0  invalid misc:0
```

Configuración de los interfaces de red

Hay dos maneras de configurar los interfaces de red: con herramientas de bajo nivel o con herramientas de alto nivel. No deben usarse ambos métodos simultáneamente

- Con herramientas de bajo nivel:
El administrador invoca a `ifconfig`, `route` pasando parámetros directamente. Esta configuración se pierde al reiniciar la máquina.

Configuración de alto nivel

El administrador (o algunos demonios) editan ciertos ficheros de configuración, especificando todos los parámetros de los interfaces. La ubicación y el formato de estos ficheros es variable.

- Sistemas basados en Red Hat
`/etc/sysconfig/network-scripts`
- Sistemas basados en Debian
`/etc/network`
- OpenWrt
`/etc/config`

Modificar los ficheros de configuración, sin más, no provoca ningún efecto. Es necesario desactivar el interfaz (si estaba activo) y volverlo a activar

- `ifdown eth0`
Desactiva el interfaz eth0
- `ifup eth0`
Activa el interfaz eth0, atendiendo a lo indicado en el fichero de configuración

Otra manera de que los cambios en el fichero de configuración pasen a tener efecto es reiniciar el demonio de red

- `/etc/init.d/networking stop`
`/etc/init.d/networking start`
- O lo que es lo mismo
`/etc/init.d/networking restart`

Interfaces de red en Debian/Ubuntu

Fichero `/etc/network/interfaces`

- Interfaz `lo`

```
auto lo
iface lo inet loopback
```

- Configuración estática (fijada a mano por el administrador)

```
auto eth0
iface eth0 inet static
    address 192.168.1.5
    netmask 255.255.255.0
    gateway 192.168.1.1
    dns-nameservers 192.168.1.1 8.8.8.8 8.8.4.4
```

- Configuración dinámica (DHCP)

```
auto eth1
iface eth1 inet dhcp
```

- Se pueden poner comentarios con #, pero solo a comienzo de línea
- `auto eth0`
Todos los interfaces precedidos por *auto* se activan automáticamente al hacer `ifup -a`
(El sistema hace `ifup -a` cada vez que se inicia)

- El lugar tradicional para especificar la dirección de los servidores de DNS es el fichero `/etc/resolv.conf`

```
nameserver 193.147.184.2  
nameserver 8.8.8.8
```

- Las distribuciones actuales suelen usar la aplicación *resolvconf*, que modifica dinámicamente `/etc/resolv.conf` a partir de la opción `dns-nameservers` de `/etc/network/interfaces` o del resultado de las consultas DHCP
- En un ubuntu actual administrado desde la shell, es recomendable especificar los servidores de DNS en ambos sitios, en `/etc/resolv.conf` y en `/etc/network/interfaces`

Interfaces Inalámbricas

Para configurar una red WiFi (abierta o WEP) los parámetros son:

- wireless-essid mi_identificador_de_red
- wireless-mode [Ad-Hoc|Managed]
 - Ad-Hoc: Sin Access Point
 - Managed: Con Access Point
- wireless-key mi_clave
- wireless-keymode [restricted|open]
 - restricted: El interfaz solo acepta conexiones encriptadas
 - open: El interfaz acepta conexiones de cualquier tipo
- wireless-channel mi_canal

Ejemplo:

```
iface wlan0 inet dhcp
    wireless-essid WLAN_01
    wireless-mode Managed
    wireless-key 58303030313338345630443031
    wireless-keymode open
    wireless-channel 6
```

El essid es sensible a mayúsculas/minúsculas

Si no indicamos canal, usamos el indicando por el AP

Red con seguridad WPA/WPA2

- Es necesario tener instalado el paquete `wpa_supplicant`

```
auto wlan0
iface wlan0 inet dhcp
    wpa-ssid mi_identificador_de_red
    wpa-psk mi_clave
```


Interfaces de red en OpenWrt

Los interfaces de red en OpenWrt se configuran en `/etc/config/network`:

Ejemplos:

- Dirección estática

```
config interface lan
    option ifname eth0
    option proto static
    option ipaddr 192.168.1.1
    option netmask 255.255.255.0
    option gateway 192.168.1.254
    option dns 192.168.1.254
```

- Dirección dinámica (DHCP)

```
config interface lan
    option ifname eth0
    option proto dhcp
```

En OpenWRT es habitual agrupar los interfaces físicos en puentes (*bridges*)

- Un puente es un interfaz virtual formado al agregar uno o más interfaces físicos
- Usando puentes, ya no se puede aplicar `ifup`, `ifdown` al interfaz físico, sino al puente completo
- El nombre del interfaz físico que conocemos p.e. con `dmesg|grep eth` ya no sirve. Tenemos que fijarnos en la definición del puente (en `/etc/config/network`)

```
ifdown lan  
ifup lan
```