LPIC-1 102-500 – Lesson 14

109.2 Persistent network configuration

Network Interfaces

- To connect a system to a network a Network Interface Card (NIC) is needed. A typical protocol used by these cards can Ethernet, Wi-Fi (wireless cards) and in legacy systems modems and RS-232 serial ports.
- Examples of network interfaces:
  - **lo (loopback)**: use to check the TCP/IP stack and for private access to local services. The localhost hostname and the 127.0.0.0/8 (IPv4) and ::1 (IPv6) IP addresses, point here.
  - Legacy Device Naming:
    - **eth0**: The first Ethernet card starts from id 0**.** If there are more, this naming scheme increases the card id by 1 e.g. **eth1**, **eth2** etc.
    - **wlan0**: A **Wi-Fi** example
  - Consistent Network Device Naming:
    - **enp0s25**: an Ethernet example
    - **wlp3s0**: a Wi-Fi example
  - **ppp0**: A serial connection (dial-up, leased line. etc)

Setting hostnames with */etc/hostname* and */etc/hosts*

- The **/etc/hostname** is used on most distros for setting up the hostname, e.g. web01.
- The hostname is also included in the **/etc/hosts** file:
  127.0.0.1 web01
- Often it is combined with the domain name:
  127.0.0.1 web01.example.com web01

Associating hostnames with IP addresses in */etc/hosts*

- The **/etc/hosts** file is used to associate hostnames with IP addresses and it is a simple way to resolve hostnames in networks without DNS or in cases where you do not have access to the DNS server. You can also set easy to remember names for hosts you use often.

- **$ cat /etc/hosts**

```
127.0.0.1       localhost
127.0.1.1       name-lpi
69.64.38.128    web01.example.com web01
```

- **$ ping malena**

```
PING web01 (69.64.38.128) 56(84) bytes of data.
64 bytes from web01 (69.64.38.128): icmp_req=1 ttl=50 time=189 ms
64 bytes from web01 (69.64.38.128): icmp_req=2 ttl=50 time=189 ms
^C
--- web01 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 189.017/189.067/189.117/0.050 ms
```

Set name resolution systems with */etc/nsswitch.conf*

- The **/etc/nsswitch.conf** file configures the name resolution systems to be used and their priority that should be obeyed when resolving names of users, groups, password and hostnames.
- **$ cat /etc/nsswitch.conf**

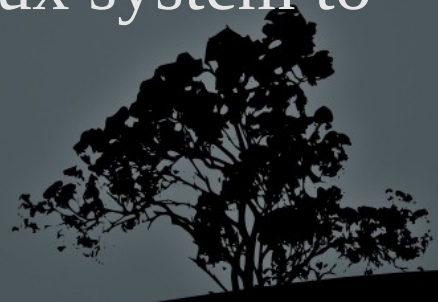
```
passwd:   files # resolve user names from /etc/passwd
shadow:   files # resolve passwords from /etc/shadow
group:    files # resolve group names from /etc/group
hosts:    files dns # resolv hostnames from /etc/hosts and
then DNS
```

Set name resolution systems with */etc/nsswitch.conf*

- **$ cat /etc/nsswitch.conf**

```
passwd:  files ldap # resolve usernames from
                    # /etc/passwd first and then LDAP
shadow:  files ldap # resolve passwords from
                    # /etc/shadow first and then LDAP
group:   files ldap # resolve group from /etc/group
                    # and then LDAP
hosts:   files dns ldap # resolve hostnames from
                        # /etc/hosts first, then
                        # DNS and then LDAP
```

- Other possible authentication systems, instead of **ldap**, could be **nis** and **nisplus** of Sun Microsystems and **winbind** or **sssd** which can join a Linux system to Microsoft Active Directory.



Configure DNS client with */etc/resolv.conf*

- The **/etc/resolv.conf** is used to configure namservers (DNS resolvers) and search domains. Example:

  Search example.com # you can resolve example.com hosts by name
  nameserver 192.168.10.1
  nameserver fd64:d180:8b30::1

Persistent network settings with */etc/network/interfaces* (Debian)

- In Debian systems the **/etc/network/interfaces** file is used for persistent network setup
- **$ cat /etc/network/interfaces**

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static # iface eth0 inet
                       # dhcp for automatic
                       # IP assignment
    address   10.0.2.15
    netmask 255.255.255.0
    network 10.0.2.0
    broadcast 10.0.2.250
    gateway 10.0.2.250
```

} # not needed for dhcp

Persistent network settings with */etc/network/interfaces* (Debian)

- In Debian systems the **/etc/network/interfaces** file is used for persistent network setup
- **$ cat /etc/network/interfaces**

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static # iface eth0 inet
                       # dhcp for automatic
                       # IP assignment
    address   10.0.2.15/24 # alternative way
    gateway    10.0.2.250
    dns-nameservers: 10.1.1.1 10.1.2.3 # DNS
```

Persistent network settings with */etc/sysconfig/network-scripts/ifcfg-ethX* (RedHat)

- In RedHat systems, every network interface has its own configuration file under: **/etc/sysconfig/network-scripts**.
- **$ cat /etc/sysconfig/network-scripts/ifcfg-eth0**

```
DEVICE=eth0
BOOTPROTO=none # =dhcp for automatic assignment
ONBOOT=yes
HWADDR=00:50:56:98:6e:d9
NETMASK=255.255.255.0
IPADDR=172.16.255.14
GATEWAY=172.16.255.250
TYPE=Ethernet
```

NetworkManager

- **NetworkManager** is a daemon that provides another way of configuring the network on a Linux system (usually on Graphical environments).
- It can be configured using a graphical interface but also from the command line too. It has these two CLI tools:
  - **nmtui**: a TUI interactive tool
  - **nmcli**: a CLI utility

The `nmcli` utility

- The **nmcti** command can be used on systems were **NetworkManager** to show and persistently set network details
- **$ nmcti** # show network information for all interfaces.
- **$ nmcli device show wlp2s0** # show information about the **wlp2s0** device.
- **$ nmcli connection** # show a summary of all connections.
- # **nmcli con add type ethernet con-name eth1 ifname enp0s3 ip410.0.2.10/24 gw4 192.168.1.254** # Set connection name (eth1) ip address and gateway.
- # **nmcli con mod enp0s9 ipv4.addresses 192.168.0.8/24** # set an IPv4 address on the **enp0s9** interface.
- # **nmcli con mod enp0s3 ipv4.gateway 192.168.1.1** # set the gateway.
- # nmcli con mod enp0s3 ipv4.method manual # change the method from **DHCP** to **static**.
- # **nmcli con mod enp0s3 ipv4.dns "8.8.8.8"** # Set a DNS resolver.
- # **nmcli con up enp0s3** # Activate the interface and save the changes.

The `hostnamectl` command

- The **hostnamectl** command can also be used on **systemd** enabled systems to set the hostname and display useful information about the running system.
- **$ hostnamectl** # show system information

```
      Static hostname: alita
            Icon name: computer-laptop
              Chassis: laptop
           Machine ID: 5bb6aabf78e346e592f26595840d24e2
              Boot ID: 3eff8dc53ce34dc0b798d2ed66e571ce
    Operating System: Debian GNU/Linux 10 (buster)
              Kernel: Linux 4.19.0-12-amd64
        Architecture: x86-64
```

The `hostnamectl` command

- **# hostnamectl set-hostname mypc** # Set or change the system hostname.
- **# hostnamectl set-location Chassis4B** # Set the location of the system.
- **# hostnamectl set-chassis vm** # Set the Chassis type.

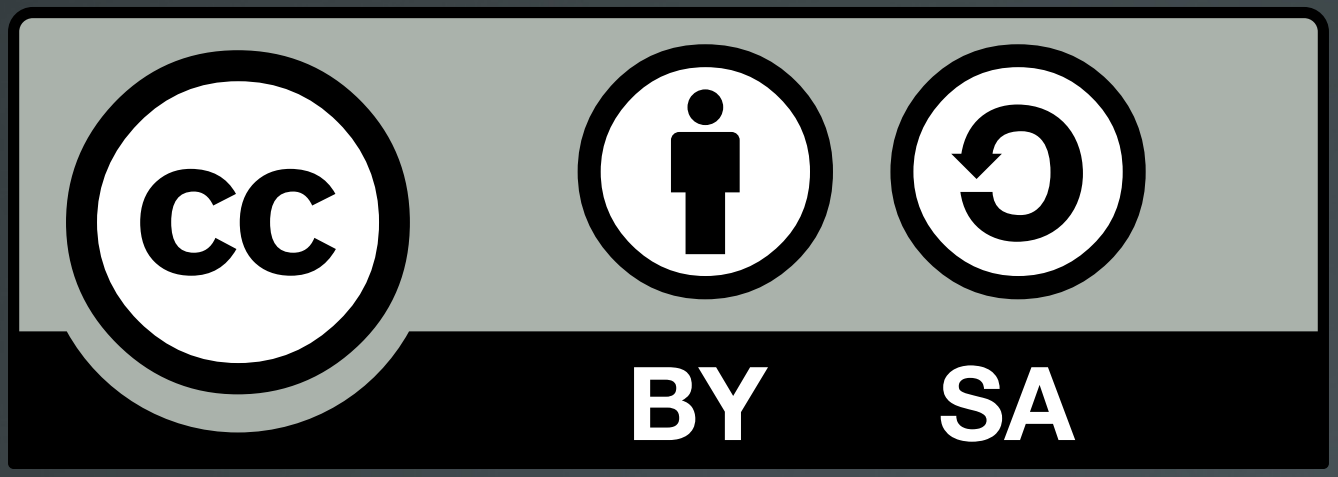
Enable/Disable network interfaces with `ifup` and `ifdown`

- The **ifup** and **ifdown** are used to enable and disable the network interfaces.
- `# ifdown eth0 # disable interface eth0.`
- `# ifup eth0 # enable interface eth0.`

The *DHCP* Protocol

- For the automatic assignment of IP addresses and the rest of the network configuration, the **DHCP** (Dynamic Host Configuration Protocol) protocol is used.
- The DHCP protocols works on the client-server model. The computers set as DHCP clients send a broadcast message in the local network. If there is a DHCP server in the network it will reply with an available IP Address, the netmask, the Default Gateway, the DNS Servers and any other secondary configuration settings.
- `# dhclient eth0 # send a DHCP request from the eth0 interface.`
- `# dhclient -r eth0 # release the current IP from the eth0 interface.`

License



The work titled ”LPIC-1 102-500 – Lesson 14” by Theodotos Andreou is distributed with the Creative Commons Attribution ShareAlike 4.0 International License.