<epam>

Network Configuration

Linux Essentials

TRAINING

CENTER

<epam>

Network Configuration Files

Name	Description
/etc/hosts	The main purpose of this file is to resolve hostnames that cannot be resolved any other way.
/etc/resolv.conf	This file specifies the IP addresses of DNS servers and the search domain
/etc/sysconfig/network	This file specifies routing and host information for all network interfaces
/etc/sysconfig/network-scripts/ifcfg-*<interface-name>*	For each network interface, there is a corresponding interface configuration script. Each of this files provide information specific to a particular network interface

Network Configuration Files

/etc/hosts – is used to resolve hostnames if that can't be resolved by means the DNS server.

```
[root@test1 ~]# cat /etc/hosts
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
10.156.0.10 test1.europe-west3-c.c.terraform-293410.internal test1  # Added by Google
169.254.169.254 metadata.google.internal  # Added by Google
192.168.0.1 myserver.example.com
[root@test1 ~]#
```

Network Configuration Files

/etc/resolv.conf – defines parameters of the mechanism for translating network names to IP addresses. Contains the next configuration key/values

- **nameserver <IP address>**
- **search** <list of domains the system should search when trying to resolve an unqualified host name>

```
[root@test1 ~]# cat /etc/resolv.conf
# Generated by NetworkManager
search europe-west3-c.c.terraform-293410.internal c.terraform-293410.internal google.internal
nameserver 169.254.169.254
[root@test1 ~]#
```

Network Configuration Files

/etc/sysconfig/network – is used to specify information about the desired network configuration. The following values may be used:

- **NETWORKING**=<***yes/no***>
  - **yes –** Networking should be configured
  - **no –** Networking should not be configured
- **HOSTNAME**=<***value***>, where <***value***> should be the Fully Qualified Domain Name (FQDN)
- **GATEWAY**=<**value**>, where <***value***> is the IP address of the network's gateway

Network Configuration Files

/etc/sysconfig/network-scripts/ifcfg-*<interface-name>*

- Supply the configuration information for each network interface.
- Contain a series of keywords and values parsed at boot time.
- Values are used to configure network interfaces, net masks, and host names; set default gateways; and perform other tasks required to bring the host up on the network.

The following is a sample ifcfg-eth0 file for a system using static IP address:

```
DEVICE="eth0"                 # The interface name
BOOTPROTO="none"              # Set to "dhcp" to use DHCP
IPADDR="172.16.205.99"        # Host's static IP address
NETWORK="172.16.205.96"       # The network number
BROADCAST="172.16.205.127"    # The broadcast address
NETMASK="255.255.255.240"     # The netmask
ONBOOT="yes"                  # yes to configure at boot
USERCTL="no"                  # Non-root user are not allowed to control this device
```

Configuring Networking in CLI

- **ip** command – is a tool for configuring network interfaces. It is used to:
  - Assign an address to a network interface
  - Configure network interface parameters
  - Bring interface up or down
  - Assign and remove routes
  - Manager ARP cache

  The syntax for **ip** command is as follows:

  ip [OPTIONS] OBJECT { COMMAND | help }

  The most frequently used objects are:

  - **link (l) –** Display and modify network interfaces
  - **address (a)** – Display and modify IP addresses
  - **route (r)** – Display and alter the routing table
  - **neigh (n)** – Display and manipulate neighbor objects (ARP table)

Configuring Networking in CLI

Name	Command
View status of interface	$ip a show eth0
Stop Ethernet interface	$ip link set eth0 down
Start Ethernet interface	$ip link set eth0 up
List all network interfaces	$ip a
Restart network service	$ systemctl restart network
Add an IP address	$ip a add {ip_addr/mask} dev {interface}

Configuring Networking in CLI

- **ip route** command
  - View or configure routing table within kernel
  - Executed at boot time when networking initialized

Add route syntax:
 # ip route add {NETWORK} via {GATEWAY_IP} dev {DEVICE}
For example, add network 192.168.55.0/24 available via 192.168.1.254
 # ip route add 192.168.55.0/24 via 192.168.1.254 dev eth0

Alternatively, we can use old “route” command
 # route add –net 192.168.55.0 netmask 255.255.255.0 gw 192.168.1.254 dev eth0

Display routes:
 # ip route show