A CLOUD GURU

BASIC NETWORK CONFIGURATION

Introduction to Basic Network Configuration

Matthew Pearson
Linux Training Architect

Section Components

1 **Understanding Network Interfaces**

What is a network interface and what is it used for?

2 **Managing Wired Network Interfaces**

Display information and configure wired network interfaces using the `ip` and `ifconfig` commands.

3 **Managing Wireless Network Interfaces**

Display information and configure wireless network interfaces using the `iw`, `iwconfig`, and `iwlist` commands.

4 **Discovering Network Devices**

View and adjust connected network devices using the `ip` and `arp` commands.

A CLOUD GURU

BASIC NETWORK CONFIGURATION

Introduction to Basic Network Configuration

Understanding Network Interfaces

Managing Wired Network Interfaces

Managing Wireless Network Interfaces

Discovering Network Devices

Matthew Pearson
Linux Training Architect

BASIC NETWORK CONFIGURATION

Understanding Network Interfaces

What is a Network Interface?

Wired Network Interfaces

Wireless Network Interfaces

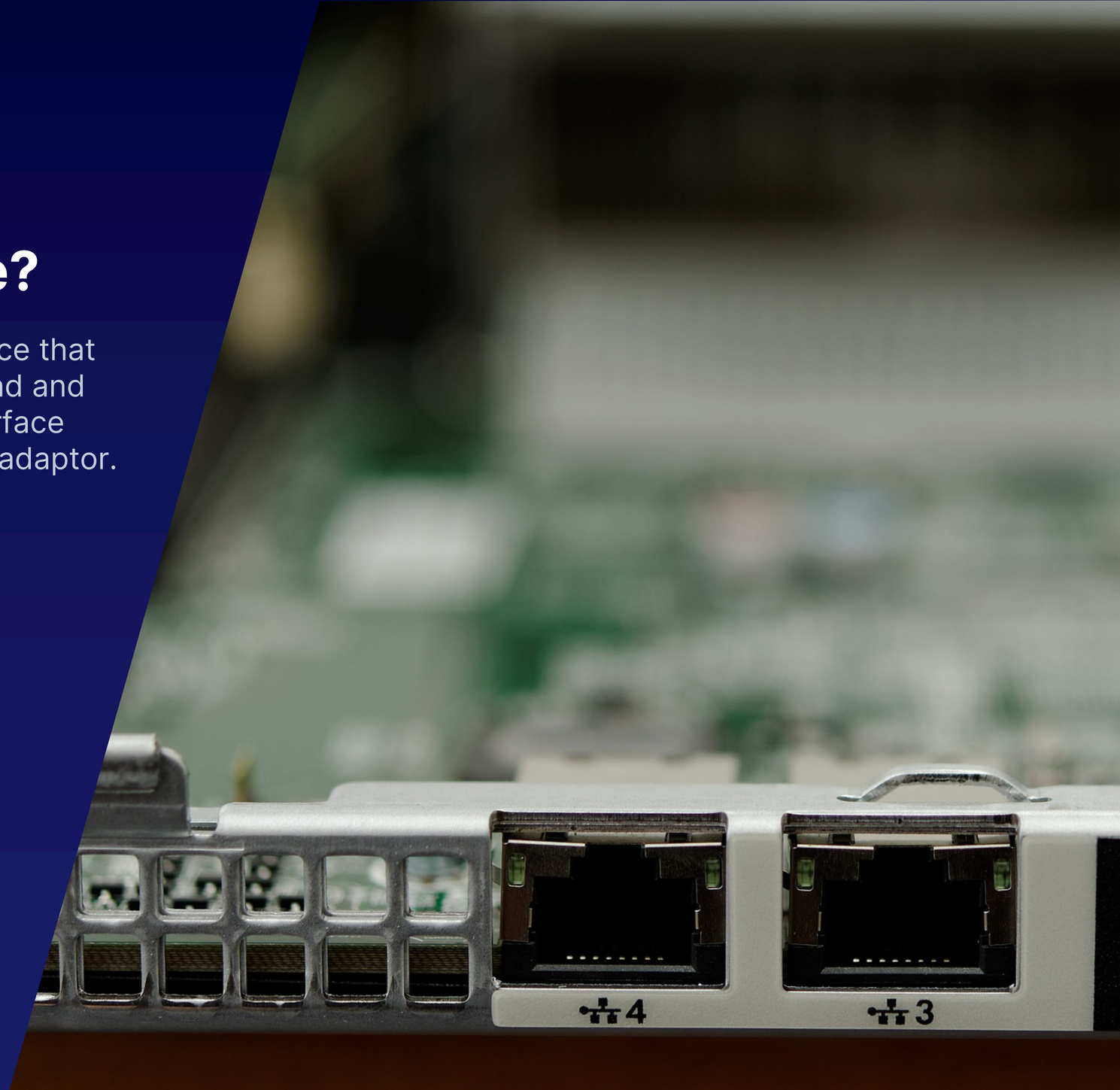
Matthew Pearson
Linux Training Architect

A CLOUD GURU

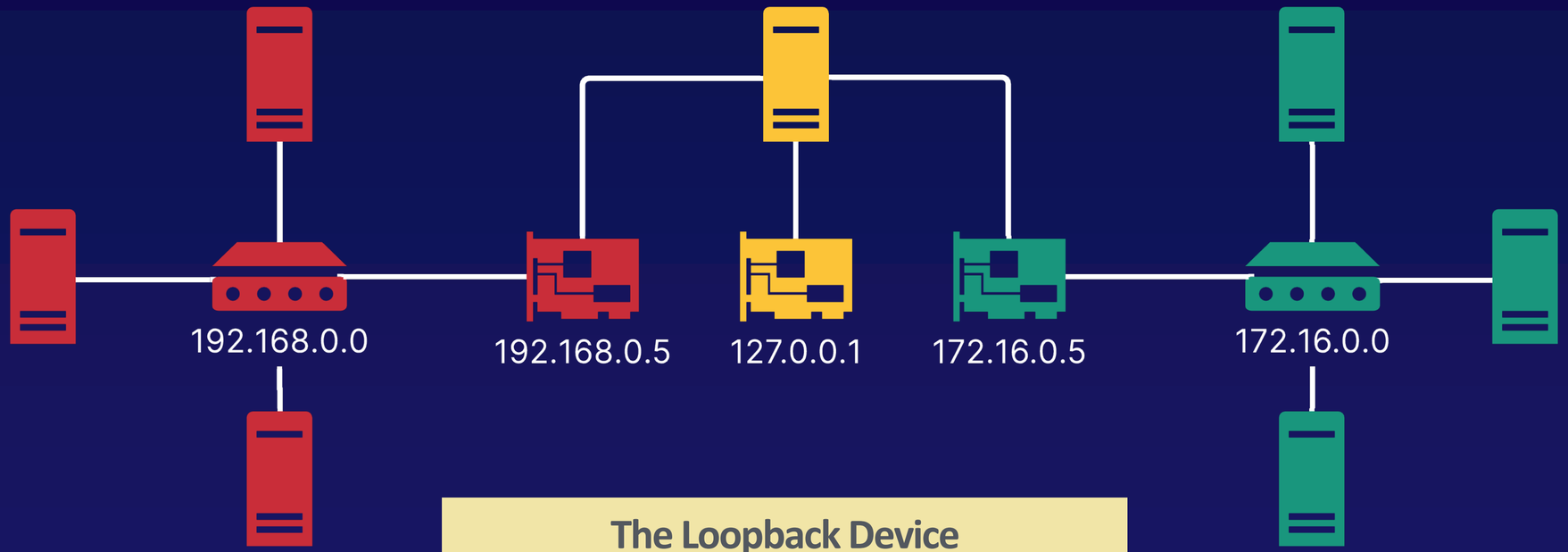
A CLOUD GURU

What is a Network Interface?

A network interface is a hardware or software device that allows a computer to connect to a network and send and receive data. Can be referred to as a Network Interface Card or Controller (NIC), Network Adaptor, or LAN adaptor.



Wired Network Interfaces



The Loopback Device

A special, virtual network interface that allows internal communication on a computer. It has an address of 127.0.0.1 (IPv4) and ::1 (IPv6) and is mapped to `localhost`

Wireless Network Interfaces

Service Set Identifier (SSID)

A unique identifier that is used to define a Wireless Local Area Network (WLAN) and differentiate it from other similar networks in the area.

BASIC NETWORK CONFIGURATION

Understanding Network Interfaces

What is a Network Interface?

Wired Network Interfaces

Wireless Network Interfaces

Matthew Pearson
Linux Training Architect

A CLOUD GURU

A CLOUD GURU

BASIC NETWORK CONFIGURATION

Managing Wired Network Interfaces

- Understanding the `ip` command
- Understanding the `ifconfig` command
- Managing interfaces with the `ip` command
- Managing interfaces with the `ifconfig` command

Matthew Pearson
Linux Training Architect

The ip Command

Command	Purpose
`ip help`	Display a list of commands and options for the `ip` command
`ip addr help`	Display a list of command and options for the address subcommand
`ip link help`	Display a list of command and options for the link subcommand
`ip addr`	Show information for all address
`ip addr show dev em1`	Show information for a specific device
`ip addr add 192.168.1.1/24 dev em1`	Add an address to a device
`ip addr del 192.168.1.1/24 dev em1`	Remove an address from a device
`ip addr add 192.168.1.1/24 broadcast 192.168.1.255 dev em1`	Add an IP address and specific broadcast address to a device

Note:
The `ip` command is provided by the `iproute2` package

The ip Command

Command	Purpose
`ip link`	Show information for all interfaces
`ip link show dev em1`	Show information for a single device
`ip -s link`	Show interface statistics
`ip link set`	Alter the status of an interface
`ip link set mtu` *`number`*	Set maximum transmission unit for a network interface
`ip link set em1 promisc on`	Set a network interface to promiscuous mode
`ip link set em1 up`	Bring a device online
`ip link set em1 down`	Bring a device offline

The ifconfig Command

Command	Purpose
`ifconfig`	Display information for active network interfaces
`ifconfig -a`	Display information for all network interfaces
`ifconfig eth0`	Display information for a specific network interface
`ifconfig eth0 up`	Bring a device online
`ifconfig eth0 down`	Bring a device offline
`ifconfig eth0 192.168.1.10`	Assign an IP address to a network interface
`ifconfig eth0 netmask 255.255.255.0`	Assign a netmask to a network interface
`ifconfig eth0 broadcast 192.168.1.255`	Assign a broadcast address to a network interface
`ifconfig eth0 0.0.0.0`	Remove an IP address from a network interface
`ifconfig eth0 mtu` *`number`*	Set maximum transmission unit for a network interface
`ifconfig eth0 promisc`	Set a network interface to promiscuous mode

Note:
The `ifconfig` command is provided by the `net-tools` package

A CLOUD GURU

BASIC NETWORK CONFIGURATION

Managing Wired Network Interfaces

- Understanding the `ip` command
- Understanding the `ifconfig` command
- Managing interfaces with the `ip` command
- Managing interfaces with the `ifconfig` command

Matthew Pearson
Linux Training Architect

A CLOUD GURU

BASIC NETWORK CONFIGURATION

Managing Wireless Network Interfaces

Understanding the `iw` Command

Understanding the `iwconfig` Command

Understanding the `iwlist` Command

Matthew Pearson
Linux Training Architect

The iw command


```
# iw dev
phy#0
    Interface wlan0
            ifindex 3
            type managed

# iw dev wlan0 link
Connected to 00:19:e6:8d:55:64 (on wlan0)
    SSID: wsiit
    freq: 2437
    RX: 18444610 bytes (94857 packets) T
    X: 2554688 bytes (17365 packets)
    signal: -60 dBm
    tx bitrate: 54.0 MBit/s
    bss flags: short-preamble short-slot-time
    dtim period: 0
    beacon int: 100
```

Note:
The iw command is provided by the iw package

Command	Purpose
`iw help`	Print all supported commands
`iw help command`	Print help information for specified command
`iw dev`	View available wireless interfaces
`iw list`	List all wireless devices and their capabilities
`iw dev wlan0 link`	Display link information
`iw dev wlan0 info`	Show information for an interface
`iw event`	Monitor events from the kernel
`iw wlan0 scan`	Scan for available SSIDs
`iw dev wlan0 connect <SSID>`	Connect to a wireless network
`iw dev wlan0 disconnect`	Disconnect from a wireless network

The iwconfig and iwlist commands


```
# iwconfig wlan0
Wlan0    IEEE 802.11g ESSID:”MyNetwork"
         Mode:Managed Frequency:2.427 GHz Access Point:
         00:1D:A2:88:A9:41 Bit Rate:54 Mb/s Tx-Power=20 dBm
         ...

# iwlist wlan0 scan
wlan0    Scan completed :
         Cell 01 - Address: 00:12:17:46:E6:AF
         ESSID:”MyNetwork”
         Protocol:IEEE 802.11bg
         Mode:Master
         Channel:1
         Encryption key:off
         Bit Rate:1 Mb/s
         Bit Rate:2 Mb/s B
         ...
```

Note:
The iwconfig and iwlist commands is provided by the wireless tools package

Command	Purpose
`iwconfig`	Display information about all available wireless interfaces
`iwconfig wlan0`	Display information about a wireless interface
`iwconfig --help`	Display a list of commands and options
`iwconfig wlan0 essid "MyNetwork" key my_key`	Connect to a wireless network by providing a key
`iwconfig wlan0 rate 24M`	Set the bitrate for an interface
`iwlist wlan0 scan`	Scan for available wireless networks
`iwlist wlan0 freq`	List available frequencies
`iwlist wlan0 rate`	List available bit rates

A CLOUD GURU

BASIC NETWORK CONFIGURATION

Managing Wireless Network Interfaces

Understanding the `iw` Command

Understanding the `iwconfig` Command

Understanding the `iwlist` Command

Matthew Pearson
Linux Training Architect

A CLOUD GURU

BASIC NETWORK CONFIGURATION

Discovering Network Devices

Understanding the `ip` Command

Understanding the `arp` Command

Matthew Pearson
Linux Training Architect

Using the ip Command (ip neighbour)


```
# ip neigh
169.254.169.254 dev eth0 lladdr 0e:fc:27:8d:c6:49 STALE
10.0.1.1 dev eth0 lladdr 0e:fc:27:8d:c6:49 REACHABLE

# ip neigh add 192.168.1.10 lladdr 0e:e5:20:c6:c4:75 dev
eth1

# ip neigh show
169.254.169.254 dev eth0 lladdr 0e:fc:27:8d:c6:49 STALE
10.0.1.1 dev eth0 lladdr 0e:fc:27:8d:c6:49 REACHABLE
192.168.1.10 dev eth1 lladdr 0e:e5:20:c6:c4:75 PERMANENT

# ip neigh del 192.168.1.10 dev eth1

# ip neigh
169.254.169.254 dev eth0 lladdr 0e:fc:27:8d:c6:49 STALE
10.0.1.1 dev eth0 lladdr 0e:fc:27:8d:c6:49 REACHABLE
```

Command	Purpose
`ip neigh`	Display neighbor objects
`ip -s neigh`	Display neighbor objects in verbose with statistics
`ip neigh show dev em1`	Show the `arp` cache for a device
`ip neigh add 192.168.1.10 lladdr 1:2:3:4:5:6 dev em1`	Add an entry into the ARP table
`ip neigh del 192.168.1.10 dev em1`	Invalidate an entry in the ARP table
`ip neigh replace 192.168.1.10 lladdr 1:2:3:4:5:6 dev em1`	Replace an entry or add one if not defined

Using the arp Command


```
# arp
Address                  HWtype  HWaddress           Flags Mask            Iface
instance-data.ec2.inter  ether   0e:fc:27:8d:c6:49   C                     eth0
ip-10-0-1-1.ec2.interna  ether   0e:fc:27:8d:c6:49   C                     eth0

# arp -s 192.168.1.11 -i eth2 0e:4a:08:cf:6d:61

# arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.1.11             ether   0e:4a:08:cf:6d:61   CM                    eth2
169.254.169.254          ether   0e:fc:27:8d:c6:49   C                     eth0
10.0.1.1                 ether   0e:fc:27:8d:c6:49   C                     eth

# arp -i eth2 -d 192.168.1.11

# arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
169.254.169.254          ether   0e:fc:27:8d:c6:49   C                     eth0
10.0.1.1                 ether   0e:fc:27:8d:c6:49   C                     eth0
```

Command	Purpose
`arp [-avn]`	Display the contents of the ARP cache
`arp -i eth1`	Display entries for an interface
`arp -a 192.168.1.11`	Display entries for an IP address
`arp -s 192.168.1.11 -i eth2 1:2:3:4:5:6`	Add an entry to the ARP cache
`arp -i eth1 -d 192.168.1.11`	Remove an entry from the ARP cache

A CLOUD GURU

BASIC NETWORK CONFIGURATION

Discovering Network Devices

Understanding the `ip` Command

Understanding the `arp` Command

Matthew Pearson
Linux Training Architect

A CLOUD GURU

BASIC NETWORK CONFIGURATION

Section Conclusion

Matthew Pearson
Linux Training Architect

Section Components

Understanding Network Interfaces

Understand what a network interfaces is used for and the difference between wired and wireless interfaces.

Managing Wireless Network Interfaces

Use the `iw`, `iwconfig`, and `iwlist` commands to display information and configure network interfaces.

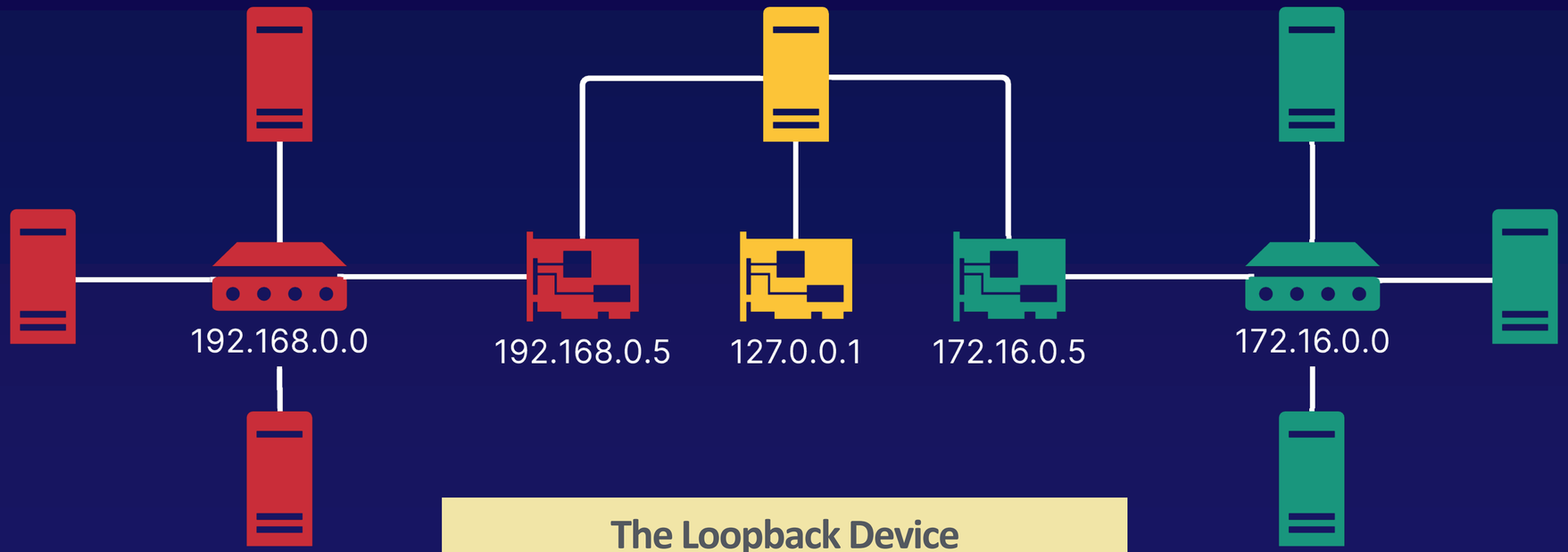
Managing Wired Network Interfaces

Use the `ip` and `ifconfig` commands to display information and configure network interfaces.

Discovering Network Devices

Use the `ip` and `arp` commands to display information about connected devices and alter the `arp` cache.

Wired Network Interfaces



The Loopback Device

A special, virtual network interface that allows internal communication on a computer. It has an address of 127.0.0.1 (IPv4) and ::1 (IPv6) and is mapped to `localhost`

Wireless Network Interfaces

Service Set Identifier (SSID)

A unique identifier that is used to define a Wireless Local Area Network (WLAN) and differentiate it from other similar networks in the area.

The ip Command

Command	Purpose
`ip help`	Display a list of commands and options for the `ip` command
`ip addr help`	Display a list of command and options for the address subcommand
`ip link help`	Display a list of command and options for the link subcommand
`ip addr`	Show information for all address
`ip addr show dev em1`	Show information for a specific device
`ip addr add 192.168.1.1/24 dev em1`	Add an address to a device
`ip addr del 192.168.1.1/24 dev em1`	Remove an address from a device
`ip addr add 192.168.1.1/24 broadcast 192.168.1.255 dev em1`	Add an IP address and specific broadcast address to a device

The ip Command

Command	Purpose
`ip link`	Show information for all interfaces
`ip link show dev em1`	Show information for a single device
`ip -s link`	Show interface statistics
`ip link set`	Alter the status of an interface
`ip link set mtu` *`number`*	Set maximum transmission unit for a network interface
`ip link set em1 promisc on`	Set a network interface to promiscuous mode
`ip link set em1 up`	Bring a device online
`ip link set em1 down`	Bring a device offline

The ifconfig Command

Command	Purpose
`ifconfig`	Display information for active network interfaces
`ifconfig -a`	Display information for all network interfaces
`ifconfig eth0`	Display information for a specific network interface
`ifconfig eth0 up`	Bring a device online
`ifconfig eth0 down`	Bring a device offline
`ifconfig eth0 192.168.1.10`	Assign an IP address to a network interface
`ifconfig eth0 netmask 255.255.255.0`	Assign a netmask to a network interface
`ifconfig eth0 broadcast 192.168.1.255`	Assign a broadcast address to a network interface
`ifconfig eth0 0.0.0.0`	Remove an IP address from a network interface
`ifconfig eth0 mtu` *`number`*	Set maximum transmission unit for a network interface
`ifconfig eth0 promisc`	Set a network interface to promiscuous mode

The iw command


```
# iw dev
phy#0
    Interface wlan0
            ifindex 3
            type managed

# iw dev wlan0 link
Connected to 00:19:e6:8d:55:64 (on wlan0)
    SSID: wsiit
    freq: 2437
    RX: 18444610 bytes (94857 packets) T
    X: 2554688 bytes (17365 packets)
    signal: -60 dBm
    tx bitrate: 54.0 MBit/s
    bss flags: short-preamble short-slot-time
    dtim period: 0
    beacon int: 100
```

Command	Purpose
`iw help`	Print all supported commands
`iw help command`	Print help information for specified command
`iw dev`	View available wireless interfaces
`iw list`	List all wireless devices and their capabilities
`iw dev wlan0 link`	Display link information
`iw dev wlan0 info`	Show information for an interface
`iw event`	Monitor events from the kernel
`iw wlan0 scan`	Scan for available SSIDs
`iw dev wlan0 connect <SSID>`	Connect to a wireless network
`iw dev wlan0 disconnect`	Disconnect from a wireless network

The iwconfig and iwlist commands


```
# iwconfig wlan0
Wlan0    IEEE 802.11g ESSID:”MyNetwork"
         Mode:Managed Frequency:2.427 GHz Access Point:
         00:1D:A2:88:A9:41 Bit Rate:54 Mb/s Tx-Power=20 dBm
         ...

# iwlist wlan0 scan
wlan0    Scan completed :
         Cell 01 - Address: 00:12:17:46:E6:AF
         ESSID:”MyNetwork”
         Protocol:IEEE 802.11bg
         Mode:Master
         Channel:1
         Encryption key:off
         Bit Rate:1 Mb/s
         Bit Rate:2 Mb/s B
         ...
```

Command	Purpose
`iwconfig`	Display information about all available wireless interfaces
`iwconfig wlan0`	Display information about a wireless interface
`iwconfig --help`	Display a list of commands and options
`iwconfig wlan0 essid "MyNetwork" key my_key`	Connect to a wireless network by providing a key
`iwconfig wlan0 rate 24M`	Set the bitrate for an interface
`iwlist wlan0 scan`	Scan for available wireless networks
`iwlist wlan0 freq`	List available frequencies
`iwlist wlan0 rate`	List available bit rates

Using the ip Command (ip neighbour)


```
# ip neigh
169.254.169.254 dev eth0 lladdr 0e:fc:27:8d:c6:49 STALE
10.0.1.1 dev eth0 lladdr 0e:fc:27:8d:c6:49 REACHABLE

# ip neigh add 192.168.1.10 lladdr 0e:e5:20:c6:c4:75 dev
eth1

# ip neigh show
169.254.169.254 dev eth0 lladdr 0e:fc:27:8d:c6:49 STALE
10.0.1.1 dev eth0 lladdr 0e:fc:27:8d:c6:49 REACHABLE
192.168.1.10 dev eth1 lladdr 0e:e5:20:c6:c4:75 PERMANENT

# ip neigh del 192.168.1.10 dev eth1

# ip neigh
169.254.169.254 dev eth0 lladdr 0e:fc:27:8d:c6:49 STALE
10.0.1.1 dev eth0 lladdr 0e:fc:27:8d:c6:49 REACHABLE
```

Command	Purpose
`ip neigh`	Display neighbor objects
`ip -s neigh`	Display neighbor objects in verbose with statistics
`ip neigh show dev em1`	Show the `arp` cache for a device
`ip neigh add 192.168.1.10 lladdr 1:2:3:4:5:6 dev em1`	Add an entry into the ARP table
`ip neigh del 192.168.1.10 dev em1`	Invalidate an entry in the ARP table
`ip neigh replace 192.168.1.10 lladdr 1:2:3:4:5:6 dev em1`	Replace an entry or add one if not defined

Using the arp Command


```
# arp
Address                  HWtype  HWaddress           Flags Mask            Iface
instance-data.ec2.inter  ether   0e:fc:27:8d:c6:49   C                     eth0
ip-10-0-1-1.ec2.interna  ether   0e:fc:27:8d:c6:49   C                     eth0

# arp -s 192.168.1.11 -i eth2 0e:4a:08:cf:6d:61

# arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.1.11             ether   0e:4a:08:cf:6d:61   CM                    eth2
169.254.169.254          ether   0e:fc:27:8d:c6:49   C                     eth0
10.0.1.1                 ether   0e:fc:27:8d:c6:49   C                     eth

# arp -i eth2 -d 192.168.1.11

# arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
169.254.169.254          ether   0e:fc:27:8d:c6:49   C                     eth0
10.0.1.1                 ether   0e:fc:27:8d:c6:49   C                     eth0
```

Command	Purpose
`arp [-avn]`	Display the contents of the ARP cache
`arp -i eth1`	Display entries for an interface
`arp -a 192.168.1.11`	Display entries for an IP address
`arp -s 192.168.1.11 -i eth2 1:2:3:4:5:6`	Add an entry to the ARP cache
`arp -i eth1 -d 192.168.1.11`	Remove an entry from the ARP cache

A CLOUD GURU

BASIC NETWORK CONFIGURATION

Section Conclusion

Understanding Network Interfaces

Managing Wired Network Interfaces

Managing Wireless Network Interfaces

Discovering Network Devices

Matthew Pearson
Linux Training Architect

A CLOUD GURU

ADVANCED NETWORK CONFIGURATION AND TROUBLESHOOTING

Introduction to Advanced Network Configuration and Troubleshooting

Matthew Pearson
Linux Training Architect

Section Components

1

Adjusting Network Routing

Alter network routing to determine how packets are transferred and received. This lesson will focus on the `ip route` and `route` commands.

2

Displaying Statistics on Network Sockets

View network sockets to determine what ports are listening, what services are listening on those ports, and what files have been opened by network sockets. This lesson will focus on the `ss`, `netstat`, and `lsof` commands.

3

Analyzing and Monitoring Network Traffic

The ability to monitor and analyze network packets on a server is a helpful tool in troubleshooting network issues and keeping track of network communication. In this lesson, we will focus on the `tcpdump` and `nmap` commands.

4

Interacting with Remote Hosts

Being able to test connections and interact with remote hosts is a large part of Linux administration. This lesson will focus on the `ping`, `ping6`, and `ncat` commands.

A CLOUD GURU

ADVANCED NETWORK CONFIGURATION AND TROUBLESHOOTING

Introduction to Advanced Network Configuration and Troubleshooting

Adjusting Network Routing

Displaying Statistics on Network Sockets

Analyzing and Monitoring Network Traffic

Interacting with Remote Hosts

Matthew Pearson
Linux Training Architect

ADVANCED NETWORK CONFIGURATION AND TROUBLESHOOTING

Adjusting Network Routing

Understanding the `ip route` Command

Understanding the `route` Command

Using the `ip route` and `route` Commands

Matthew Pearson
Linux Training Architect

A CLOUD GURU

The ip route and route Commands

Command	Purpose
`ip route show`	Display the routing table
`ip route add 10.0.2.0/24 via 10.0.2.10 dev eth1`	Add a route
`ip route del 10.0.2.0/24 via 10.0.2.10 dev eth1`	Remove a route
`ip route add default via 10.0.2.10`	Add a default gateway
`ip route add prohibit 10.0.2.10/24`	Block destination route, send ICMP message
`ip route add blackhole 10.0.2.0/24`	Block destination route, silently discard

Command	Purpose
`route (-n)`	Display the routing table
`route add -net 10.0.2.0/24 gw 10.0.2.10 eth1`	Add a route
`route del -net 10.0.2.0/24 gw 10.0.2.10 eth1`	Remove a route
`route add default gw 10.0.2.10`	Add a default gateway
`route add -host 10.0.2.10 reject`	Block destination route for a host
`route add -net 10.0.2.0 netmask 255.255.255.0 reject`	Block destination route for a network

ADVANCED NETWORK CONFIGURATION AND TROUBLESHOOTING

Adjusting Network Routing

Understanding the `ip route` Command

Understanding the `route` Command

Using the `ip route` and `route` Commands

Matthew Pearson
Linux Training Architect

A CLOUD GURU

ADVANCED NETWORK CONFIGURATION AND TROUBLESHOOTING

Displaying Statistics on Network Sockets

Understanding the `ss` Command

Understanding the `netstat` Command

Using the `lsof` Command

Using the `ss`, `netstat`, and `lsof` Commands

Matthew Pearson
Linux Training Architect

A CLOUD GURU

The ss and netstat Commands

ss

A utility for investigating network sockets

Option	Description
`-l, --listening`	display listening server sockets
`-a, --all`	display all sockets (default: connected)
`–i, --interfaces`	display interface table
`-s, --summary`	show socket usage summary (like SNMP)
`-e, --extended`	show detailed socket information
`-n, --numeric`	don’t resolve names
`-p, --programs`	display PID/Program name for sockets
`-t, --tcp`	display only TCP sockets
`-u, --udp`	display only UDP sockets

netstat

Print network connections, routing tables, interface statistics, masquerade connections, and multicast memberships

Option	Description
`-l, --listening`	display listening server sockets
`-a, --all`	display all sockets (default: connected)
`–i, --interfaces`	display interface table
`-s, --statistics`	display networking statistics
`-e, --extended`	show detailed socket information
`-v, --verbose`	be verbose
`-n, --numeric`	don’t resolve names
`-p, --programs`	display PID/Program name for sockets
`-t, --tcp`	display only TCP sockets
`-u, --udp`	display only UDP sockets
`-r, --route`	display routing table

The lsof Command

`lsof`

List open files. Provided by the `lsof` package.

Option	Description
`-u user_name`	list open files by user
`-u ^user_name`	list open files and exclude a user
`-i [46][protocol][@hostname|hostaddr][:service|port]`	list open files by network connection
`-p PID`	list open files by PID
`-p ^PID`	list open files and exclude a PID
`/directory`	list open files by directory
`/dev/sda1`	list open files by device
`-c`	list open files by process name

ADVANCED NETWORK CONFIGURATION AND TROUBLESHOOTING

Displaying Statistics on Network Sockets

Understanding the `ss` Command

Understanding the `netstat` Command

Using the `lsof` Command

Using the `ss`, `netstat`, and `lsof` Commands

Matthew Pearson
Linux Training Architect

A CLOUD GURU

A CLOUD GURU

ADVANCED NETWORK CONFIGURATION AND TROUBLESHOOTING

Analyzing and Monitoring Network Traffic

Understanding the `tcpdump` Command

Understanding the `nmap` Command

Using `tcpdump` and `nmap` in the Command Line

Matthew Pearson
Linux Training Architect

The tcpdump Command

Option	Description
`-D`	List interfaces available for capture
`-i eth0`	Capture packets on an interface or all interfaces (any)
`-c`	Capture a specified count of packets
`-n`	Disable hostname resolution
`-nn`	Disable protocol, port, and hostname resolution
`-i any protocol`	Capture packets by protocol on all interfaces
`-i any host 10.0.2.10`	Capture packets by a host on all interfaces
`-i any src/dst 10.0.2.10`	Capture packets by source or destination address on all interfaces
`-A`	View packet content in ASCII
`-X`	View packet content in hex and ASCII
`-w file_name.pcap`	Save the output of tcpdump to a file
`-r file_name.pcap`	Read packets from a file

`tcpdump`

A network traffic monitoring tool. Can monitor protocols other than TCP.

Logical operators 'and' and 'or' can be used to combine filters.

`TCP Flags:`

Value	Flag Type	Description
s	SYN	connection start
f	FIN	connection finish
p	PUSH	data push
r	RST	connection reset
.	ACK	acknowledgment

The nmap Command

Option	Description
`hostname`	Scan using a hostname or multiple hostnames
`10.0.2.10`	Scan using IP address or multiple IP addresses
`-v 10.0.2.10`	Increase verbosity
`-iL hosts.txt`	Scan a list of hosts from a file
`-A 10.0.2.10`	Enable OS detection, version detection, script scanning, and traceroute
`-O 10.0.2.10`	Enable OS detection
`-sA 10.0.2.10`	Detect firewall or packet filters
`-Pn 10.0.2.10`	Skip host discovery (formerly -PN)
`-sn 10.0.2.10`	`Perform a "ping scan" do not detect open ports (formerly -sP)`
`-F 10.0.2.10`	Perform fast scan using less ports
`-r 10.0.2.10`	Scan ports consecutively - don't randomize

`nmap`

Network mapper is a network exploration and security scanner.

The network Mapper services file is located at `/usr/share/nmap/nmap-services`.

The nmap Command Continued

Option	Description
`--iflist`	View host interface and route information
`-p 22,443 10.0.2.10`	Specify ports to scan
`-sU 58 10.0.2.10`	Scan for a UDP port
`-sV 10.0.2.10`	Determine service/version information
`-sS 10.0.2.10`	Perform TCP SYN scan (stealthy scan)
`-sT 10.0.2.10`	Perform TCP connect scan

`nmap`

Network mapper is a network exploration and security scanner.

The network Mapper services file is located at `/usr/share/nmap/nmap-services`.

A CLOUD GURU

ADVANCED NETWORK CONFIGURATION AND TROUBLESHOOTING

Analyzing and Monitoring Network Traffic

Understanding the `tcpdump` Command

Understanding the `nmap` Command

Using `tcpdump` and `nmap` in the Command Line

Matthew Pearson
Linux Training Architect

A CLOUD GURU

ADVANCED NETWORK CONFIGURATION
AND TROUBLESHOOTING

Interacting with Remote Hosts

Understanding the `ping` and `ping6` Commands

Understanding the `ncat (nc)` Command

Using `ping` and `nc` in the Command Line

Matthew Pearson
Linux Training Architect

The ping and ping6 Commands

Option	Description
`hostname`	Send a stream of ICMP packets to a hostname
`10.0.2.10`	Send a stream of ICMP packets to an IP address
`-c 5 10.0.2.10`	Send a specified amount of packets
`-s 100 10.0.2.10`	Alter the size of the packets
`-i 3 10.0.2.10`	Change the interval for sending packets
`-q 10.0.2.10`	Only show the summary information
`-w 5 10.0.2.10`	Set a timeout of when to stop sending packets
`-f 10.0.2.10`	Flood ping. Send packets as soon as possible.
`-p ff 10.0.2.10`	Fill a packet with data. ff fills the packet with ones
`-b 10.0.2.10`	Send packets to a broadcast address
`-t 10 10.0.2.10`	Limit the number of network hops
`-v 10.0.2.10`	Increase verbosity

`ping and ping6`

Utilities used to send ICMP ECHO_REQUEST to network hosts.

Provided by the `iputils` package.

All options listed can be used by both the `ping` and `ping6` commands.

The ncat (nc) Command

Option	Description
`-l port`	Listen for inbound connections on a port
`10.0.2.10 port`	Connect to remote system on a specific port
`-u udp_port`	Specify a UDP port (TCP is the default)
`-w time_count`	Terminate connection after specified time
`-l -k port`	Accept multiple connections in listen mode
`-v`	Increase verbosity
`-z`	Report connection status only
`-i`	Set an idle timeout
`-v -z 10.0.2.10 22 80`	Scan multiple ports
`-v -z 10.0.2.10 20-80`	Scan a range of ports
`-c command`	Executes given command via /bin/sh
`-e command`	Executes the given command

`ncat (nc)`

A networking utility which reads and writes data across networks from the command line.

Provided by the `nmap-ncat` package.

A CLOUD GURU

ADVANCED NETWORK CONFIGURATION
AND TROUBLESHOOTING

Interacting with Remote Hosts

Understanding the `ping` and `ping6` Commands

Understanding the `ncat (nc)` Command

Using `ping` and `nc` in the Command Line

Matthew Pearson
Linux Training Architect

A CLOUD GURU

ADVANCED NETWORK CONFIGURATION AND TROUBLESHOOTING

Section Conclusion

Matthew Pearson
Linux Training Architect

Section Components

Adjusting Network Routing

Alter network routes in order to enable and disable network communication using the `ip route` and `route` commands.

Analyzing and Monitoring Network Traffic

View and analyze network network packets using the `tcpdump` command, and scan hosts and ports using the `nmap` command.

Displaying Statistics on Network Sockets

View information about network sockets and determine the services and files associated using the `ss`, `netstat`, and `lsof` commands.

Interacting with Remote Hosts

Determine the availability of remote hosts and interact with them using the `ping`, `ping6`, and `ncat (nc)` commands.

A CLOUD GURU

ADVANCED NETWORK CONFIGURATION AND TROUBLESHOOTING

Section Conclusion

Adjusting Network Routing

Displaying Statistics on Network Sockets

Analyzing and Monitoring Network Traffic

Interacting with Remote Hosts

Matthew Pearson
Linux Training Architect

A CLOUD GURU

TROUBLESHOOTING NETWORK ISSUES

Introduction to Troubleshooting Network Issues

Matthew Pearson
Linux Training Architect

Section Components

1

Understanding Network Configuration Files and Locations

On older systems, networking was configured by modifying scripts by hand. This lesson will cover the various files and syntax necessary to configure networking.

2

Understanding Network Manager

On many modern distributions, Network Manager has become default networking service. This lesson will cover the command line utilities (`nmcli` and `nmtui`) that are used to interact with the Network Manager service.

3

Analyzing Network Diagnostics

Tracking the path of network packets is helpful in determining points of failure on a network. This lesson will cover the `traceroute` command which is used to map network and the `mtr` command which combines the functionality of `traceroute` and ping.

4

Logs and Utilities for Troubleshooting Network Issues

When troubleshooting issues on a Linux host, the first place to look is the log files. This lesson will cover important files and utilities for troubleshooting network issues.

5

Configuration Files and Utilities for Adjusting Hostnames and IP Addresses

Hostnames and IP address are used to differentiate one host from another. This lesson will cover the configuration files and utilities needed to adjust hostnames and access.

A CLOUD GURU

TROUBLESHOOTING NETWORK ISSUES

Introduction to Troubleshooting Network Issues

Understanding Network Configuration Files and Locations

Understanding Network Manager

Analyzing Network Diagnostics

Logs and Utilities for Troubleshooting Network Issues

Configuration Files and Utilities for Adjusting Hostnames and IP Addresses

Matthew Pearson
Linux Training Architect

TROUBLESHOOTING NETWORK ISSUES

Understanding Network Configuration Files and Locations

Understanding RHEL-Based Network Configuration Files

Understanding Debian-Based Network Configuration Files

Configuring an Interface Using Network Scripts

Matthew Pearson
Linux Training Architect

A CLOUD GURU

ifcfg Configuration Files in /etc/sysconfig/network-scripts

Option	Description
`TYPE=Ethernet`	The type of network interface device
`BOOTPROTO=none`	Specify boot protocol (none|dhcp|bootp)
`DEFROUTE=yes`	Specify default route for IPv4 traffic (yes|no)
`IPV4_DEFROUTE=yes`	Specify default route for IPv6 traffic (yes|no)
`IPV4_FAILURE_FATAL=no`	Disable the device if the configuration fails (yes|no)
`IPV6_FAILURE_FATAL=no`	Disable the device if the configuration fails (yes|no)
`IPV6INIT=yes`	Enable or disable IPv6 on the interface (yes|no)
`IPV6_AUTOCONF=yes`	Enable or disable autoconf configuration (yes|no)
`NAME=eth0`	Specify a name for the connection
`UUID=...`	Specify the unique identifier for the device
`ONBOOT=yes`	Activate interface on boot (yes|no)
`HWADDR=0e:a5:1a:b9:fc:89`	Specify the MAC address for the interface

Note:

The interface configuration files are named after the interface that they reference (i.e., ifcfg-eth0). They are located in the `/etc/sysconfig/network-scripts` directory.

ifcfg Configuration Files in /etc/sysconfig/network-scripts

Option	Description
`IPADDR=10.0.1.10`	Specify the IPv4 address
`PREFIX=24`	Specify the network prefix
`NETMASK=255.255.255`	Specify the netmask
`GATEWAY=10.0.10.1`	Specify the gateway
`DNS1=192.168.154.3`	Specify a DNS SERVER
`DNS2=10.216.106.3`	Specify another DNS SERVER
`PEERDNS=yes`	Modify the `/etc/resolv.conf` file (yes|no)

```
# cat ifcfg-eth0

BOOTPROTO=dhcp
DEVICE=eth0
DHCPV6C=yes
HWADDR=02:be:5a:69:69:0f
IPV6INIT=yes
ONBOOT=yes
TYPE=Ethernet
USERCTL=no
```

Additional Network Configuration Files


```
# cat /etc/hosts
127.0.0.1 localhost.localdomain localhost
10.0.1.10 acg.example.com acg

# cat /etc/resolv.conf
search example.com
nameserver 192.168.20.4
nameserver 172.168.100.3

# cat /etc/sysconfig/network
NETWORKING=yes
HOSTNAME=example.mylabserver.com

# cat /etc/nsswitch.conf
...
hosts: files dns
...
```

/etc/hosts

Associate hostnames with IP address.

/etc/resolv.conf

The resolver configuration file specifies DNS servers and searches domains for the host.

/etc/sysconfig/network

Used to specify global network settings.

/etc/nsswitch

The Name Service Switch (NSS) configuration file is used to determine which sources to obtain name-service information and in what order.

The /etc/network/interfaces File (Debian Based Systems)


```
# cat /etc/network/interfaces

# An example ethernet card setup: (broadcast and gateway are
optional)
#
# auto eth0
# iface eth0 inet static
# address 192.168.0.42
# network 192.168.0.0
# netmask 255.255.255.0
# broadcast 192.168.0.255
# gateway 192.168.0.1
```

Note:

Lines beginning with the word “auto” are used to identify the physical interfaces to be brought up when `ifup` is run with the `-a` option. (This option is used by the system boot scripts.)

TROUBLESHOOTING NETWORK ISSUES

Understanding Network Configuration Files and Locations

Understanding RHEL-Based Network Configuration Files

Understanding Debian-Based Network Configuration Files

Configuring an Interface Using Network Scripts

Matthew Pearson
Linux Training Architect

A CLOUD GURU

TROUBLESHOOTING NETWORK ISSUES

Understanding Network Manager

Comparing `nmcli` and `ifcfg-*` Options

Understanding Common `nmcli` Commands

Configuring a Network Interface Using `nmcli`

Matthew Pearson
Linux Training Architect

A CLOUD GURU

Comparing nmcli and ifcfg-* Options

nmcli con mod	ifcfg-* file	Purpose
`ipv4.method manual`	`BOOTPROTO=none`	Set a static IPv4 address
`ipv4.method auto`	`BOOTPROTO=dhcp`	Automatically set IPv4 address using DHCP
`ip4 | ipv4.address "192.168.0.10/24"`	`IPADDR=192.168.0.10` `PREFIX=24`	Set static IPv4 address and network prefix
`gw4 | ipv4.gateway 192.168.0.1`	`GATEWAY=192.168.0.1`	Set IPv4 Gateway
`ipv4.dns 8.8.8.8`	`DNS1=8.8.8.8`	Specify DNS server
`autoconnect yes`	`ONBOOT=yes`	Automatically activate this connection on boot
`con-name eth0`	`NAME=eth0`	Specify the name of the connection
`ifname eth0`	`DEVICE=eth0`	Specify the interface for the connection
`802-3-ethernet.mac-address ADDR`	`HWADDR=...`	Specify the MAC address of the interface for the connection

Common nmcli Commands

Command	Purpose
`nmcli dev status`	Show the status of all network interfaces
`nmcli con show`	List all connections
`nmcli con show name`	List the current settings for the connection name
`nmcli con add con-name name ...`	Add a new connection named name
`nmcli con mod name ...`	Modify a connection
`nmcli con reload`	Reload the network configuration files
`nmcli con up name | nmcli con down name`	Activate or deactivate a connection
`nmcli dev dis dev`	Deactivate and disconnect the current connection
`nmcli con del name`	Delete the connection and its configuration file

TROUBLESHOOTING NETWORK ISSUES

Understanding Network Manager

Comparing `nmcli` and `ifcfg-*` Options

Understanding Common `nmcli` Commands

Configuring a Network Interface Using `nmcli`

Matthew Pearson
Linux Training Architect

A CLOUD GURU

TROUBLESHOOTING NETWORK ISSUES

Analyzing Network Diagnostics

Understanding the `traceroute` Command

Understanding the `mtr` Command

Using the `traceroute` and `mtr` Commands

Matthew Pearson
Linux Training Architect

A CLOUD GURU

The traceroute Command

Option	Description
`-I`	Use ICMP ECHO for probes
`-T`	Use TCP SYN for probes
`-f first_ttl`	Specifies what TTL to start (default is 1)
`-g gateway`	Specify a gateway to route the packets
`-i interface`	Specify an interface to send packets through
`-m max_ttl`	Specify the maximum number of hops (default is 30)
`-n`	Do not attempt to resolve host names
`-q`	Set the number of probe packets per hop (default is 3)
`-w`	Set the time to wait, in seconds, for a response (default is 5)
`-4 | -6`	Use IPv4 or IPv6 only
`hostname packet_len`	Set the size of the probing packet (default is 60 bytes)

```
traceroute [options] hostname [packet_len]
```

`traceroute` tracks the route packets taken from an IP network on their way to a given host.

`traceroute6` is identical to `traceroute` with the `-6` option.

The mtr Command

Option	Description
`-r -c 5`	Run `mtr` in report mode and print out statistics based on the number of cycles
`-w`	Run `mtr` in wide report mode and print out statistics
`-c 5`	Specify the number of pings
`-n`	Do not resolve hostnames
`-b`	Show hostnames and IP addresses
`-o "LSD NBAW"`	Specify the fields and order of fields
`-a 10.0.2.20`	Send outgoing packets through a specific interface
`-i seconds`	Specify the interval for sending packets (default is 1)
`-m NUM`	Specify the maximum number of hops (default is 30)
`-f NUM`	Specify the hop (TTL) to start (default is 1)
`-u`	Use UDP datagrams instead of ICMP ECHO
`-T`	Use TCP SYN packets instead of ICMP ECHO
`-4 | -6`	Use IPv4 or IPv6 only

```
mtr [options] hostname
[packet_size]
```

A network diagnostic utility that combines the functionality of the `traceroute` and `ping` commands.

TROUBLESHOOTING NETWORK ISSUES

Analyzing Network Diagnostics

Understanding the `traceroute` Command

Understanding the `mtr` Command

Using the `traceroute` and `mtr` Commands

Matthew Pearson
Linux Training Architect

A CLOUD GURU

A CLOUD GURU

TROUBLESHOOTING NETWORK ISSUES

Logs and Utilities for Troubleshooting Network Issues

Understanding `/var/log/syslog`

Understanding `/var/log/messages`

Understanding Systemd Journal

Understanding the `dmesg` command

Viewing System Logging and Messaging

Matthew Pearson
Linux Training Architect

/var/log/syslog and /var/log/messages


```
# cat /var/log/syslog
Nov 11 16:25:45 2305ef99e21c systemd-networkd[699]: ens5:
DHCPv6 address 2600:1f16:fb6:4403:3998:dd07:153:b736/128
timeout preferred 150 valid 450
Nov 11 16:26:00 2305ef99e21c amazon-ssm-agent.amazon-ssm-
agent[910]: 2020-11-11 16:26:00 INFO [HealthCheck] HealthCheck
reporting agent health.

# cat /var/log/messages
Nov  5 03:37:32 f4a6fcf4871c dhclient[1423]: PRC: Renewing
lease on ens5.
Nov  5 03:37:32 f4a6fcf4871c dhclient[1423]: XMT: Renew on
ens5, interval 9980ms.
```

/var/log/syslog

The main system log for Debian-based hosts. Stores all global system activity and startup messages.

Options are controlled by `/etc/syslog.conf` or `/etc/rsyslog.conf` in newer versions. Additional configuration files can be added to `/etc/rsyslog.d/`.

/var/log/messages

The main system log on RHEL-based hosts. Stores all global system activity and startup messages.

Options are controlled by `/etc/rsyslog.conf`. Additional configurations can be added to `/etc/rsyslog.d/`.

The Journalctl and dmesg

`journalctl`

A logging system introduce by Systemd. Implemented by the `journald` daemon which stores logs in a binary format that can viewed by using the `journalctl` utility.

Settings for the Systemd journal can be updated by modifying `/etc/systemd/journald.conf` or by adding configuration files to `/etc/systemd/journald.conf.d/`.

Option	Description
`-u unit`	View messages for a particular Systemd unit
`-f`	Follow the journal for the latest messages
`-e`	Jump to the end of the journal
`-o format`	Change the format of the messages displayed
`-x`	Add explanation texts from the message catalogue
`-p`	Filter messages based on priority specified
`-S, -U`	Show entries from a specified date (since and until)

`dmesg`

A utility used to examine or control the kernel ring buffer. By default it reads all messages from the kernel ring buffer.

Option	Description
`-C`	Clear the ring buffer
`-c`	Clear the ring buffer contents after printing
`-D`	Disable printing messages to the console
`-E`	Enable printing messages to the console
`-e`	Display local time and delta in human readable format
`-H`	Enable human readable format
`-F file`	Read log from a file

A CLOUD GURU

TROUBLESHOOTING NETWORK ISSUES

Logs and Utilities for Troubleshooting Network Issues

Understanding `/var/log/syslog`

Understanding `/var/log/messages`

Understanding Systemd Journal

Understanding the `dmesg` command

Viewing System Logging and Messaging

Matthew Pearson
Linux Training Architect

A CLOUD GURU

TROUBLESHOOTING NETWORK ISSUES

Configuration Files and Utilities for Adjusting Hostnames and IP Addresses

Updating and Viewing a System Hostname

Understanding the hosts.allow and host.deny Files

Adjusting System Hostnames and Access

Matthew Pearson
Linux Training Architect

Updating and Viewing a System Hostname


```
# cat /etc/hostname
example.mylabserver.com

# hostname
example.mylabserver.com

# hostnamectl status
   Static hostname: example.mylabserver.com
         Icon name: computer-vm
           Chassis: vm
        Machine ID: 9ef2866073d1434aa3bdbde3f2f26eb1
           Boot ID: aac86d591adc49ec8b39ebb4bbea6308
    Virtualization: kvm
  Operating System: Ubuntu 18.04.5 LTS
            Kernel: Linux 5.4.0-1029-aws
      Architecture: x86-64
```

/etc/hostname and /etc/HOSTNAME

The `/etc/hostname` file is used to store the hostname of the system. On some distributions, the `/etc/HOSTNAME` file is used for this purpose but is often aliased to `/etc/hostname`.

hostname and hostnamectl

The `hostname` command is used to show or set the system's hostname (i.e., hostname HOSTNAME). On Systemd systems, the `hostnamectl` command has replaced the `hostname` command (i.e., `hostnamectl set-hostname HOSTNAME`).

/etc/hosts

This file is used to map hostnames and aliases to IP addresses.

/etc/hosts.allow and /etc/hosts.deny


```
# cat /etc/hosts.deny
sshd : ALL


# cat /etc/hosts.allow
sshd : 10.0.3.*


# cat /etc/hosts.deny
vsftpd : .mylabserver.com


# cat /etc/hosts.allow
vsftpd : example.mylabserver.com
```

/etc/hosts.allow and /etc/hosts.deny

These files are used to determine whether or not a client has permission to connect to a network service on a remote host.

The format of both files is as follows: daemon_list : client_list [: command]. The daemon list is a comma-separated list of service daemons, the client list is a comma separated list of clients, and command is an optional command that is executed when a client tries to access a server daemon.

The key word "ALL" may be used for the daemon and client lists in order to allow or deny access to all clients.

A CLOUD GURU

TROUBLESHOOTING NETWORK ISSUES

Configuration Files and Utilities for Adjusting Hostnames and IP Addresses

Updating and Viewing a System Hostname

Understanding the hosts.allow and host.deny Files

Adjusting System Hostnames and Access

Matthew Pearson
Linux Training Architect

A CLOUD GURU

TROUBLESHOOTING
NETWORK ISSUES

Section Conclusion

Matthew Pearson
Linux Training Architect

Section Components

Analyzing Network Diagnostics

Use the `traceroute` and `mtr` utilities in order to map network traffic and analyze diagnostic information.

Understanding Network Configuration Files and Locations

View and modify network configuration scripts, including `/etc/sysconfig/network-scripts` and `/etc/network/interfaces`.

Logs and Utilities for Troubleshooting Network Issues

View and analyze Linux system Logs including `systlog`, messages, Systemd journal, and output from the `dmesg` command.

Understanding Network Manager

View and modify network configurations using the utilities provided by Network Manager.

Configuration Files and Utilities for Adjusting Hostnames and IP Addresses

View and modify a system's hostname using the `/etc/hostname` file and the `hostname` and `hostnamectl` commands. Allow or deny network access with the `hosts.allow` and `hosts.deny` files.

A CLOUD GURU

TROUBLESHOOTING NETWORK ISSUES

Section Conclusion

Understanding Network Configuration Files and Locations

Understanding Network Manager

Analyzing Network Diagnostics

Logs and Utilities for Troubleshooting Network Issues

Configuration Files and Utilities for Adjusting Hostnames and IP Addresses

Matthew Pearson
Linux Training Architect