**Introduction to Networking**



A network is a group of computers and computing devices connected together through communication channels, such as cables or wireless media. The computers connected over a network may be located in the same geographical area or spread across the world.

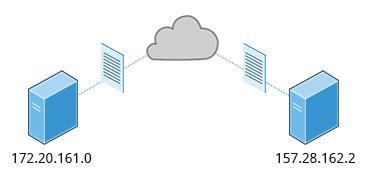
A network is used to:

* Allow the connected devices to communicate with each other.
* Enable multiple users to share devices over the network, such as printers and scanners.
* Share and manage information across computers easily.

Most organizations have both an internal network and an Internet connection for users to communicate with machines and people outside the organization. The Internet is the largest network in the world and is often called "the network of networks".

* [Previous](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)
* [Next](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)

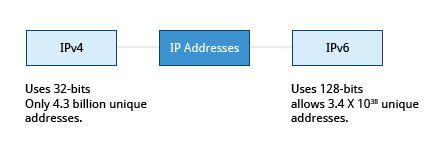
**IP Addresses**



Devices attached to a network must have at least one unique network address identifier known as the IP (**Internet Protocol)** address. The address is essential for routing **packets** of information through the network.

Exchanging information across the network requires using streams of bite-sized packets, each of which contains a piece of the information going from one machine to another. These packets contain **data buffers** together with **headers** which contain information about where the packet is going to and coming from, and where it fits in the sequence of packets that constitute the stream. Networking protocols and software are rather complicated due to the diversity of machines and operating systems they must deal with, as well as the fact that even very old standards must be supported.

**IPv4 and IPv6**



There are two different types of IP addresses available: **IPv4** (version 4) and **IPv6** (version 6). **IPv4** is older and by far the more widely used, while **IPv6** is newer and is designed to get past the limitations of the older standard and furnish many more possible addresses.

**IPv4** uses 32-bits for addresses; there are ***only*** 4.3 billion unique addresses available. Furthermore, many addresses are allotted and reserved but not actually used. **IPv4** is becoming inadequate because the number of devices available on the global network has significantly increased over the past years.

**IPv6** uses 128-bits for addresses; this allows for 3.4 X 1038 unique addresses. If you have a larger network of computers and want to add more, you may want to move to **IPv6**, because it provides more unique addresses. However, it is difficult to move to **IPv6** as the two protocols do not inter-operate. Due to this, migrating equipment and addresses to **IPv6** requires significant effort and hasn't been as fast as was originally intended.

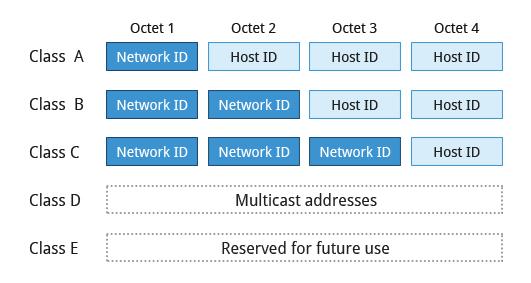
* [Previous](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)
* [Next](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)

**Decoding IPv4 Addresses**

A 32-bit IPv4 address is divided into four 8-bit sections called [octets](http://en.wikipedia.org/wiki/Octet_%28computing%29).

Example:

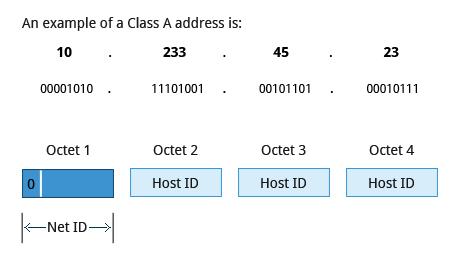
IP address → 172 . 16 . 31 . 46

Bit format → 10101100.00010000.00011111.00101110

Network address are divided into five classes: A, B, C, D, and E. Classes A, B, and C are classified into two parts: **Network addresses (Net ID)** and **Host address (Host ID)**. The Net ID is used to identify the network, while the Host ID is used to identify a host in the network. Class D is used for special multicast applications (information is broadcast to multiple computers simultaneously) and Class E is reserved for future use. In this section you will learn about classes A, B, and C.

* [Previous](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)
* [Next](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)

**Class A Network Addresses**

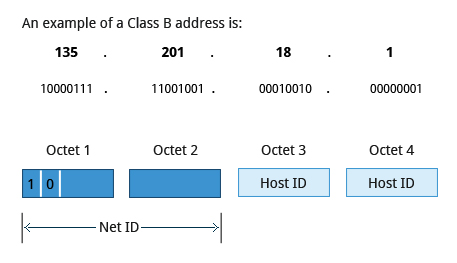


Class A addresses use the first octet of an IP address as their Net ID and use the other three octets as the **Host ID.** The first bit of the first octet is always set to zero. So you can use only 7-bits for unique network numbers. As a result, there are a maximum of 127 Class A networks available. Not surprisingly, this was only feasible when there were very few unique networks with large numbers of hosts. As the use of the Internet expanded, Classes B and C were added in order to accomodate the growing demand for independent networks.

Each Class A network can have up to 16.7 million unique hosts on its network. The range of host address is from 1.0.0.0 to 127.255.255.255.

**Note: The value of an octet, or 8-bits, can range from 0 to 255.**

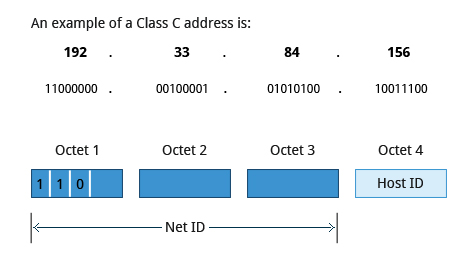
**Class B Network Addresses**



Class B addresses use the first two octets of the IP address as their Net ID and the last two octets as the Host ID. The first two bits of the first octet are always set to binary 10, so there are a maximum of 16,384 (14-bits) Class B networks. The first octet of a Class B address has values from 128 to 191. The introduction of Class B networks expanded the number of networks but it soon became clear that a further level would be needed.

Each Class B network can support a maximum of 65,536 unique hosts on its network. The range of host address is from 128.0.0.0 to 191.255.255.255.

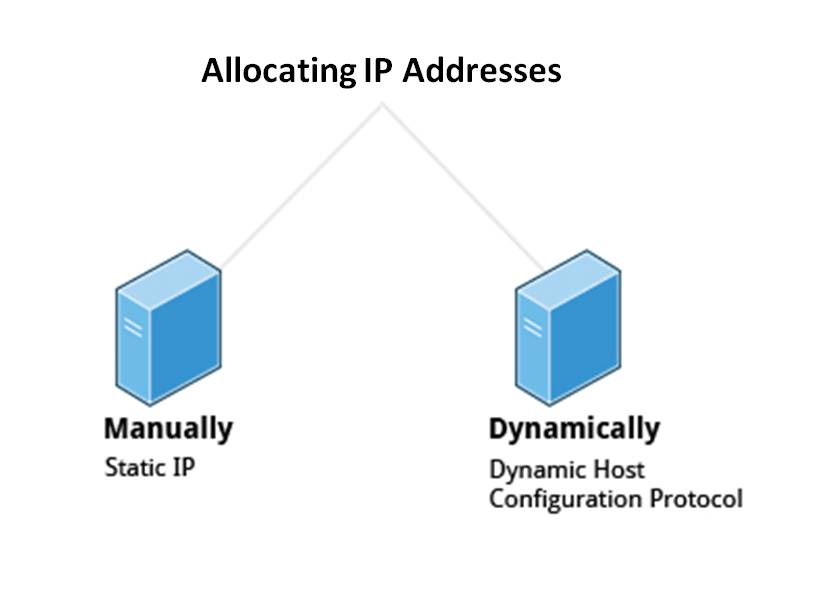
**Class C Network Addresses**



Class C addresses use the first three octets of the IP address as their Net ID and the last octet as their Host ID. The first three bits of the first octet are set to binary 110, so almost 2.1 million (21-bits) Class C networks are available. The first octet of a Class C address has values from 192 to 223. These are most common for smaller networks which don't have many unique hosts.

Each Class C network can support up to 256 (8-bits) unique hosts. The range of host address is from 192.0.0.0 to 223.255.255.255.

**IP Address Allocation**



Typically, a range of IP addresses are requested from your Internet Service Provider (ISP) by your organization's network administrator. Often your choice of which class of IP address you are given depends on the size of your network and expected growth needs.

You can assign IP addresses to computers over a network manually or dynamically. When you assign IP addresses manually, you add **static** (never changing) addresses to the network. When you assign IP addresses dynamically (they can change every time you reboot or even more often), the **Dynamic Host Configuration Protocol (DHCP)** is used to assign IP addresses.

* [Previous](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)
* [Next](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)

**Manually Allocating an IP Address**

Before an IP address can be allocated manually, one must identify the size of the network by determining the host range; this determines which network class (A, B, or C) can be used. The **ipcalc** program can be used to ascertain the host range.

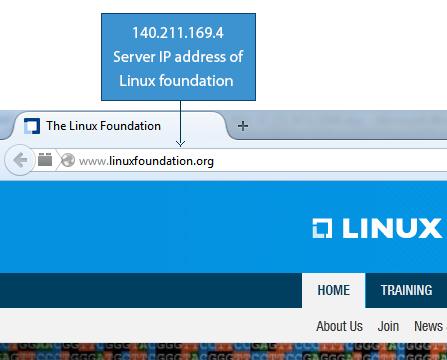
**Note: The version of ipcalc supplied in the Fedora family of distributions does not behave as described below, it is really a different program.**

Assume that you have a Class C network. The first three octets of the IP address are 192.168.0. As it uses 3 octets (i.e. 24 bits) for the network mask, the shorthand for this type of address is 192.168.0.0/24. To determine the host range of the address you can use for this new host, at the command prompt, type: ipcalc 192.168.0.0/24 and press **Enter**.

From the result, you can check the **HostMin** and **HostMax** values to manually assign a static address available from 1 to 254 (192.168.0.1 to 192.168.0.254).

* [Previous](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)
* [Next](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)

**Name Resolution**



**Name Resolution** is used to convert numerical IP address values into a human-readable format known as the **hostname**. For example, 140.211.169.4 is the numerical IP address that refers to the **linuxfoundation.org** hostname. Hostnames are easier to remember.

Given an IP address, you can obtain its corresponding hostname. Accessing the machine over the network becomes easier when you can type the hostname instead of the IP address.

You can view your system’s hostname simply by typing **hostname** with no argument.

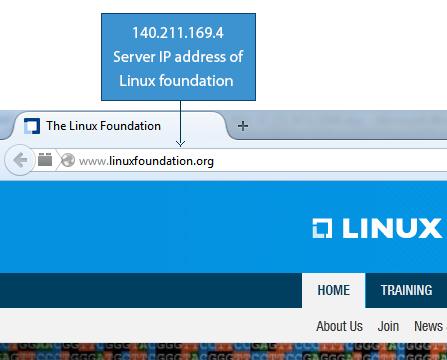
**Note: If you give an argument, the system will try to change its hostname to match it, however, only root users can do that.**

The special hostname **localhost** is associated with the IP address 127.0.0.1**,** and describes the machine you are currently on (which normally has additional network-related IP addresses).

**Note: The next two screens cover the demonstration and Try-It-Yourself activity. You can view a demonstration and practice the procedure through the Try-It-Yourself activity.**

* [Previous](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)
* [Next](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)

**Name Resolution**



**Name Resolution** is used to convert numerical IP address values into a human-readable format known as the **hostname**. For example, 140.211.169.4 is the numerical IP address that refers to the **linuxfoundation.org** hostname. Hostnames are easier to remember.

Given an IP address, you can obtain its corresponding hostname. Accessing the machine over the network becomes easier when you can type the hostname instead of the IP address.

You can view your system’s hostname simply by typing **hostname** with no argument.

**Note: If you give an argument, the system will try to change its hostname to match it, however, only root users can do that.**

The special hostname **localhost** is associated with the IP address 127.0.0.1**,** and describes the machine you are currently on (which normally has additional network-related IP addresses).

**Note: The next two screens cover the demonstration and Try-It-Yourself activity. You can view a demonstration and practice the procedure through the Try-It-Yourself activity.**

* [Previous](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)
* [Next](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)

**Using Domain Name System (DNS) and Name Resolution Tools**

**Domain Name System (DNS)** translates Internet domain and host names to IP addresses.

Click  below to view a demonstration on how to use DNS and name resolution tools.

To view the video in High Definition mode, click HD in the video panel.

[Skip to a navigable version of this video's transcript.](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#before-transcript_i4x-LinuxFoundationX-LFS101x-video-962b126eb66141c3b5c63d70ea5a8b3f)

* 0:00 / 1:17

[1.0x](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)

Maximum Volume.

[Fill browser](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)

[Turn on captions](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)

Display All Hosts

**cat /etc/hosts**

Content of Resolve

**cat /etc/resolve.conf**

Display host

**host linuxfoundation.org**

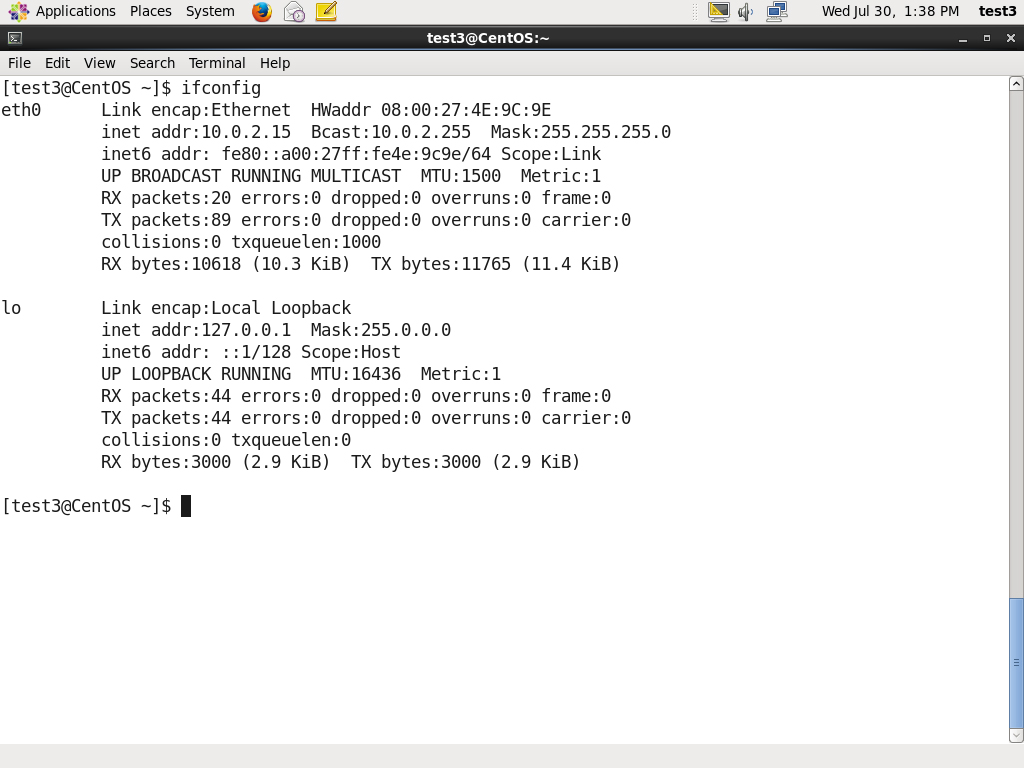
Name Server Interactively

**nslookup google.com**

Information from name server:

dig googel.com

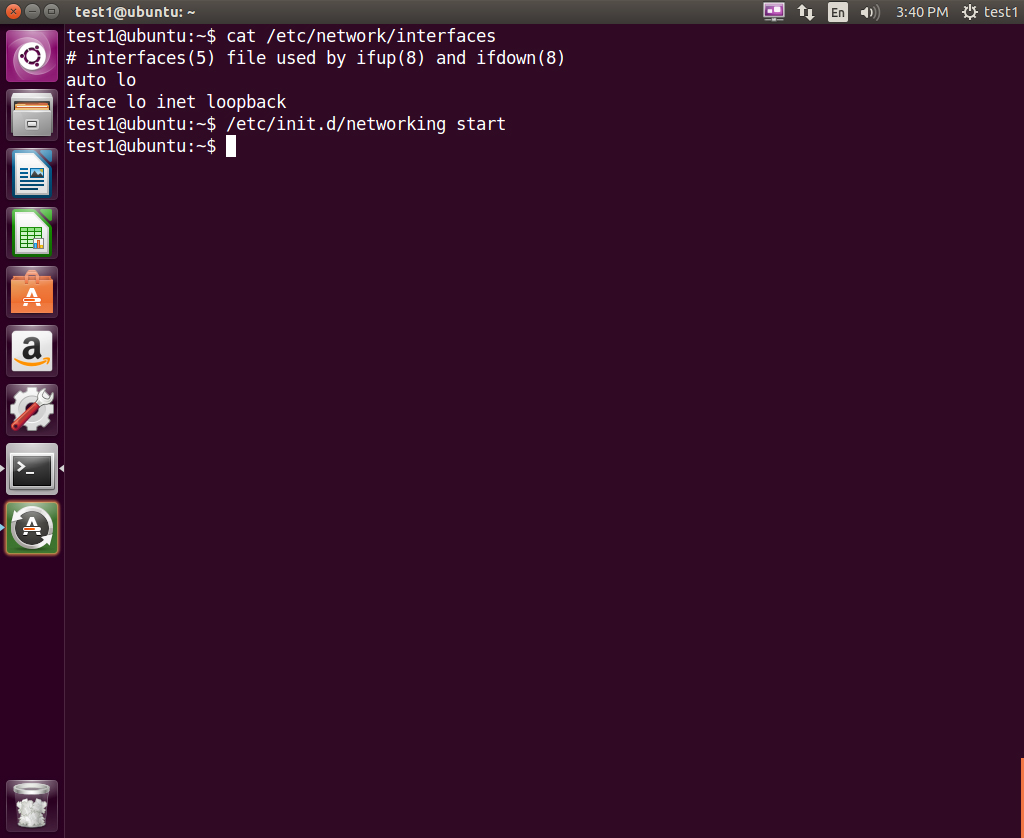
**Network Interfaces**



Network interfaces are a connection channel between a device and a network. Physically, network interfaces can proceed through a **network interface card** (**NIC**) or can be more abstractly implemented as software. You can have multiple network interfaces operating at once. Specific interfaces can be brought up (activated) or brought down (de-activated) at any time.

A list of currently active network interfaces is reported by the **ifconfig** utility which you may have to run as the superuser, or at least, give the full path, i.e., /sbin/ifconfig, on some distributions.

**Network Configuration Files**



Network configuration files are essential to ensure that interfaces function correctly.

For **Debian** family configuration, the basic network configuration file is /etc/network/interfaces. You can type /etc/init.d/networking start to start the networking configuration.

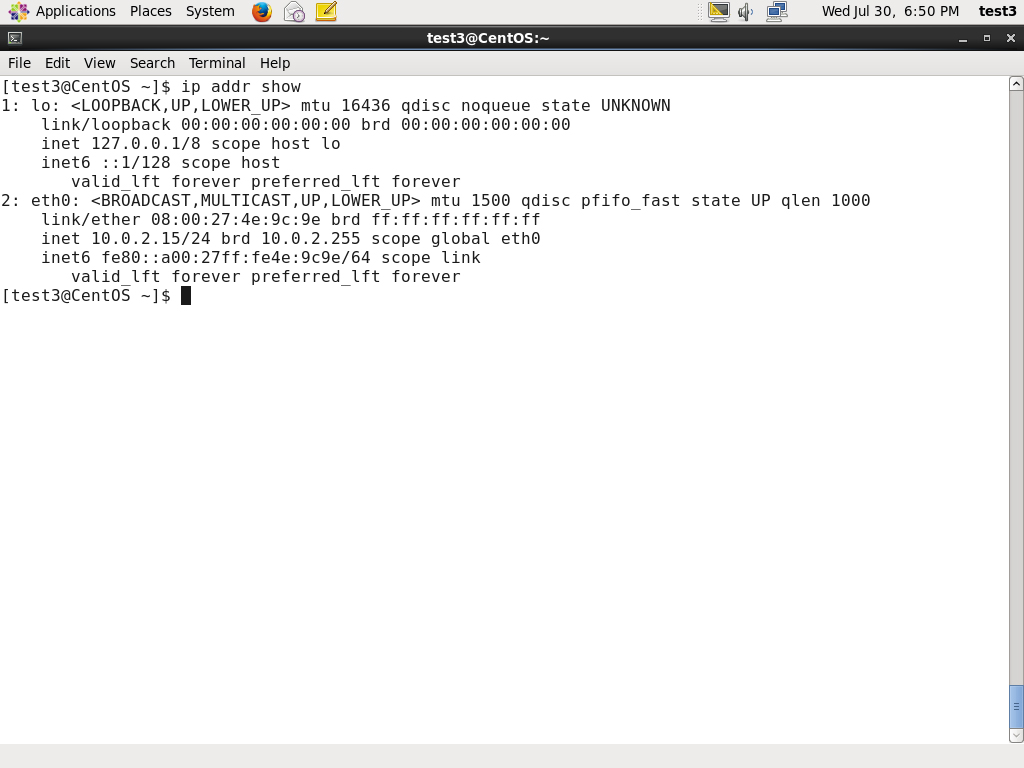
For **Fedora** family system configuration, the routing and host information is contained in /etc/sysconfig/network. The network interface configuration script is located at /etc/sysconfig/network-scripts/ifcfg-eth0.

For **SUSE** family system configuration, the routing and host information and network interface configuration scripts are contained in the /etc/sysconfig/network directory.

You can type /etc/init.d/network start to start the networking configuration for **Fedora** and **SUSE** families.

Click the image to view an enlarged version.

**Network Configuration Commands**

To view the IP address:

$ /sbin/ip addr show

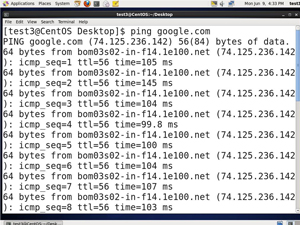
To view the routing information:

$ /sbin/ip route show

**ip** is a very powerful program that can do many things. Older (and more specific) utilities such as **ifconfig** and **route** are often used to accomplish similar tasks. A look at the relevant **man pages** can tell you much more about these utilities.

Click the image to view an enlarged version.

**ping**

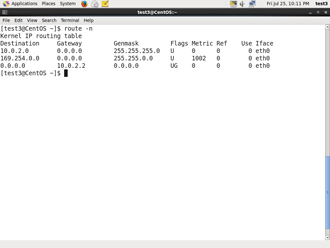
**ping is used to check whether or not a machine attached to the network can receive and send data; i.e., it confirms that the remote host is online and is responding.**

**To check the status of the remote host, at the command prompt, type ping <hostname>.**

**ping is frequently used for network testing and management; however, its usage can increase network load unacceptably. Hence, you can abort the execution of ping by typing CTRL-C, or by using the -c option, which limits the number of packets that ping will send before it quits. When execution stops, a summary is displayed.**

**Click the image to view an enlarged version.**

**route**



**A network requires the connection of many nodes. Data moves from source to destination by passing through a series of routers and potentially across multiple networks. Servers maintain routing tables containing the addresses of each node in the network. The IP Routing protocols enable routers to build up a forwarding table that correlates final destinations with the next hop addresses.**

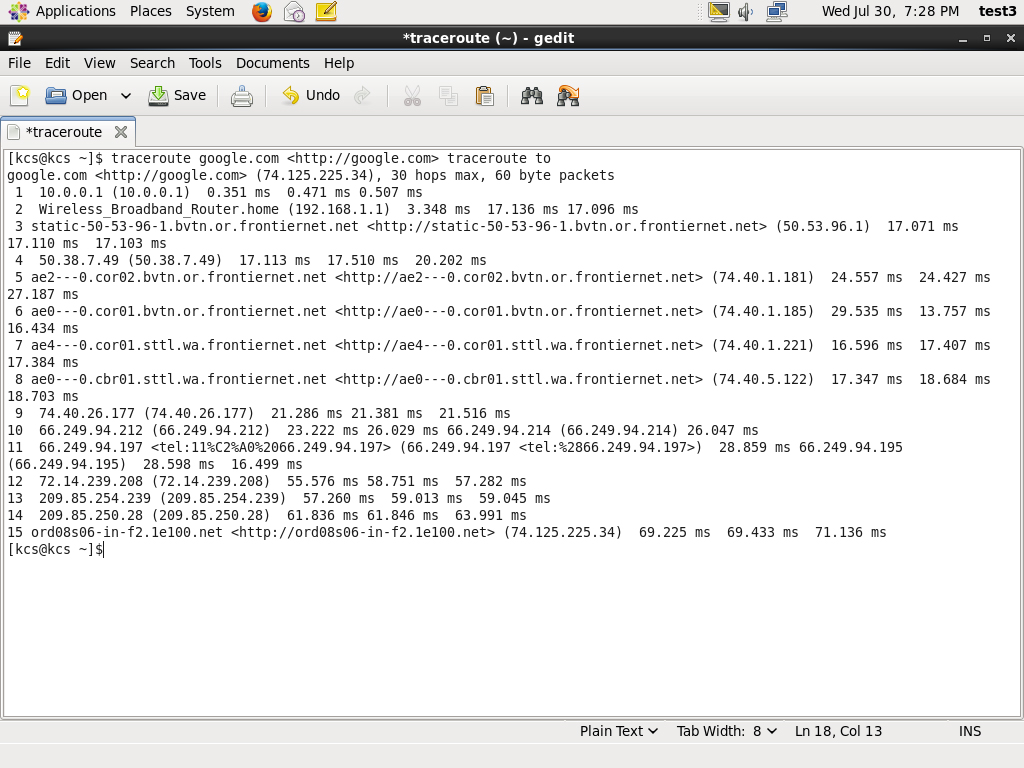
**route is used to view or change the IP routing table. You may want to change the IP routing table to add, delete or modify specific (static ) routes to specific hosts or networks. The table explains some commands that can be used to manage IP routing.**

|  |  |
| --- | --- |
| **Task** | **Command** |
| **Show current routing table** | **$ route –n** |
| **Add static route** | **$ route add -net address** |
| **Delete static route** | **$ route del -net address** |

**Click the image to view an enlarged version.**

* [**Previous**](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)
* [**Next**](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)

**traceroute**



**traceroute is used to inspect the route which the data packet takes to reach the destination host which makes it quite useful for troubleshooting network delays and errors. By using traceroute you can isolate connectivity issues between hops, which helps resolve them faster.**

**To print the route taken by the packet to reach the network host, at the command prompt, type traceroute <domain>.**

**Click the image to view an enlarged version.**

**More Networking Tools**

**Now, let’s learn about some additional networking tools. Networking tools are very useful for monitoring and debugging network problems, such as network connectivity and network traffic.**

|  |  |
| --- | --- |
| **Networking Tools** | **Description** |
| **ethtool** | **Queries network interfaces and can also set various parameters such as the speed.** |
| **netstat** | **Displays all active connections and routing tables. Useful for monitoring performance and troubleshooting.** |
| **nmap** | **Scans open ports on a network. Important for security analysis** |
| **tcpdump** | **Dumps network traffic for analysis.** |
| **iptraf** | **Monitors network traffic in text mode.** |

**Note: The next two screens cover the demonstration and Try-It-Yourself activity. You can view a demonstration and practice the procedure through the Try-It-Yourself activity.**

* [**Previous**](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)
* [**Next**](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/7079eecfff7642b7857d4c3d5ae010a5/1#)

**Graphical and Non-Graphical Browsers**

**Browsers are used to retrieve, transmit, and explore information resources, usually on the World Wide Web. Linux users commonly use both graphical and non-graphical browser applications.**

**The common graphical browsers used in Linux are:**

* **Firefox**
* **Google Chrome**
* **Chromium**
* **Epiphany**
* **Opera**

**Sometimes you either do not have a graphical environment to work in (or have reasons not to use it) but still need to access web resources. In such a case, you can use non-graphical browsers such as the following:**

|  |  |
| --- | --- |
| **Non-Graphical Browsers** | **Description** |
| [**lynx**](http://lynx.browser.org/) | **Configurable text-based web browser; the earliest such browser and still in use.** |
| [**links or elinks**](http://elinks.or.cz/) | **Based on lynx. It can display tables and frames.** |
| [**w3m**](http://w3m.sourceforge.net/) | **Newer text-based web browser with many features.** |

* [**Previous**](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/1a4e72ebb0b046bd8e59993a501983f4/#)
* [**Next**](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/1a4e72ebb0b046bd8e59993a501983f4/#)

**wget**



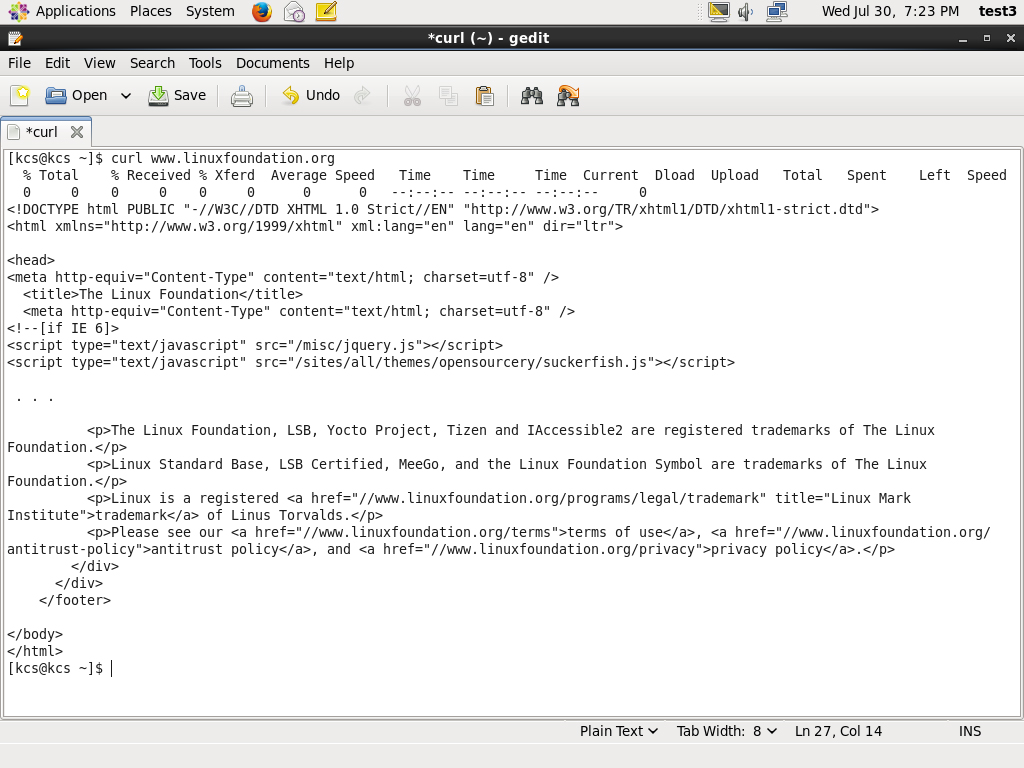
**Sometimes you need to download files and information but a browser is not the best choice, either because you want to download multiple files and/or directories, or you want to perform the action from a command line or a script. wget is a command line utility that can capably handle the following types of downloads:**

* **Large file downloads**
* **Recursive downloads, where a web page refers to other web pages and all are downloaded at once**
* **Password-required downloads**
* **Multiple file downloads**

**To download a webpage, you can simply type wget <url>, and then you can read the downloaded page as a local file using a graphical or non-graphical browser.**

**Click the image to view an enlarged version.**

**curl**



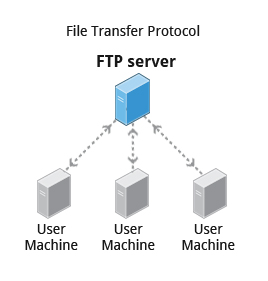
**Besides downloading you may want to obtain information about a URL, such as the source code being used. curl can be used from the command line or a script to read such information. curl also allows you to save the contents of a web page to a file as does wget.**

**You can read a URL using curl <URL>. For example, if you want to read** [**http://www.linuxfoundation.org**](http://www.linuxfoundation.org/) **, type curl** [**http://www.linuxfoundation.org**](http://www.linuxfoundation.org)**.**

**To get the contents of a web page and store it to a file, type curl -o saved.html** [**http://www.mysite.com**](http://www.mysite.com)**. The contents of the main index file at the website will be saved in saved.html.**

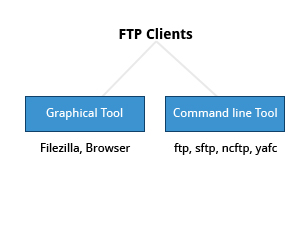
**Click the image to view an enlarged version.**

**FTP (File Transfer Protocol)**



**When you are connected to a network, you may need to transfer files from one machine to another. File Transfer Protocol (FTP) is a well-known and popular method for transferring files between computers using the Internet. This method is built on a client-server model. FTP can be used within a browser or with standalone client programs.**

**FTP Clients**



**FTP clients enable you to transfer files with remote computers using the FTP protocol. These clients can be either graphical or command line tools. Filezilla, for example, allows use of the drag-and-drop approach to transfer files between hosts. All web browsers support FTP, all you have to do is give a URL like : ftp://ftp.kernel.org where the usual http:// becomes ftp://.**

**Some command line FTP clients are:**

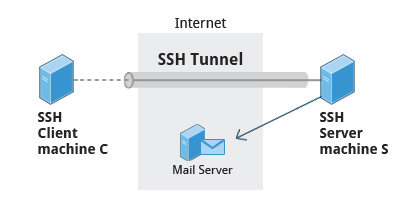
* **ftp**
* **sftp**
* **ncftp**
* **yafc (Yet Another FTP Client)**

**sftp is a very secure mode of connection, which uses the Secure Shell (ssh) protocol, which we will discuss shortly. sftp encrypts its data and thus sensitive information is transmitted more securely. However, it does not work with so-called anonymous FTP (guest user credentials). Both ncftp and yafc are also powerful FTP clients which work on a wide variety of operating systems including Windows and Linux.**

**Note: The next two screens cover the demonstration and Try-It-Yourself activity. You can view a demonstration and practice the procedure through the Try-It-Yourself activity.**

* [**Previous**](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/4a50105a454540ffa046813c5cac0036/1#)
* [**Next**](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/4a50105a454540ffa046813c5cac0036/1#)

**SSH: Executing Commands Remotely**

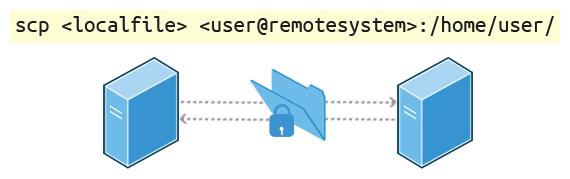


**Secure Shell (SSH) is a cryptographic network protocol used for secure data communication. It is also used for remote services and other secure services between two devices on the network and is very useful for administering systems which are not easily available to physically work on but to which you have remote access.**

**To run my\_command on a remote system via SSH, at the command prompt, type, ssh <remotesystem> my\_command and press Enter. ssh then prompts you for the remote password. You can also configure ssh to securely allow your remote access without typing a password each time.**

* [**Previous**](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/4a50105a454540ffa046813c5cac0036/1#)
* [**Next**](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/4a50105a454540ffa046813c5cac0036/1#)

**Copying Files Securely with scp**



**We can also move files securely using Secure Copy (scp) between two networked hosts. scp uses the SSH protocol for transferring data.**

**To copy a local file to a remote system, at the command prompt, type scp <localfile> <user@remotesystem>:/home/user/ and press Enter.**

**You will receive a prompt for the remote password. You can also configure scp so that it does not prompt for a password for each transfer.**

* [**Previous**](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/4a50105a454540ffa046813c5cac0036/1#)
* [**Next**](https://courses.edx.org/courses/LinuxFoundationX/LFS101x/2T2014/courseware/18780407cf8946c389bed38c4748418c/4a50105a454540ffa046813c5cac0036/1#)