

# Getting started with Linux

## Session Objectives

- **Getting started with Linux**
  - **History of Linux**
  - **Architecture of Linux**
  - **Versions of Linux**
- **Entering Linux**
  - **Login**
  - **Logout**
- **Date and Time**

---

# Introduction

An operating system is a software program that acts as an interface between the user and a computer. The main functions of an operating system are

- Memory management – Allocating memory to various applications running on the system and cleaning up unnecessary data from memory.
- Process management – Rationing out CPU time to enable several programs to run concurrently.
- Command interpretation – Translating commands typed by the user to binary code for the CPU to understand
- Peripheral management – Managing the devices (peripherals) attached to the computer.

Some of the popular operating systems available are:

- Linux
- Unix
- MS DOS
- MS Windows 95
- MS Windows 98
- MS Windows NT Server 4.0

There are basically two categories of operating systems – Single user operating systems and multi-user operating systems.

## Single user operating system

- Designed for use by one person at a time.
- Eg., MS DOS

## Multi user operating system

- More than one user can work on a multi-user system.
- Eg., Linux, Windows NT Terminal Server, Unix

NOTE
Windows 95 is a multi-tasking operating system and <b>not</b> a multi-user operating system. It can handle simultaneous execution of multiple tasks but cannot have multiple users connected to it at one point of time. Linux is a multi-user and multi-tasking operating system.

# History of Linux

Linux was introduced in 1991 by Linus Torvalds, a student at the University of Helsinki, Finland. He wrote the source code for the Linux kernel and made Linux available on the Internet.

Linux is a multitasking, multi-user operating system, which means that many people can run many different applications on one computer at the same time. This differs from MS-DOS, where only one

person can use the system at one point of time. Under Linux, to identify yourself to the system, you must log in, which entails entering your login name (the name the system uses to identify you), and entering your password, which is your personal key for logging in to your account.

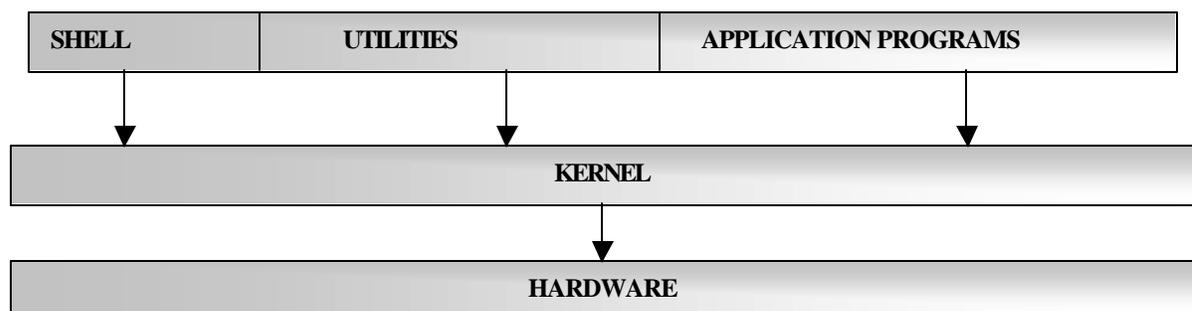
Linux is a Unix clone. The history of Unix is discussed briefly now. Unix started off as a single-user operating system. In 1969, Ken Thompson, with help from Dennis Ritchie and others wrote a small general purpose operating system, for small machines. The operating system attracted a lot of users and was eventually developed to be a multi-user operating system. Many vendors such as Sun, IBM, HP and others purchased the source code for Unix. They developed their own versions of Unix. This resulted in a lot of differences between various versions of Unix. Some standards were laid down to avoid confusion. These were known as Portable Operating System for the UNIX environment (POSIX). POSIX was a set of standards that enabled software to run on different Unix-based operating systems without changes to the source code. Linux is POSIX compliant.

Linux follows the open development model. Torvalds has made the source code of the Linux kernel available for changes over the Internet. This means that the current development version of Linux is always open to everybody, which is called as GPL (General public License). Modifications to the kernel code are also accepted. The result is that whenever a new version of Linux having new functionality is released, people work on the new version to fix bugs, if any.

Linux is an *operating system*, which is loosely defined as a set of programs (often referred to as *processes*) which manage the various resources of a computer for one or more users. These resources include the central processing unit (CPU), the file system, any monitors or terminals, disk drives and other mass storage devices, printers, and facilities for network interaction with other computers. The most common method of interacting with Linux is through a *command-line interface*. Although Linux possesses a fully developed graphical user interface (GUI) (via the X-Window system), which can largely eliminate the need to use the command-line interface for common tasks, the command-line approach is emphasized in this course.

## Architecture of Linux

The Linux operating system consists of three main software components as shown below:



### Kernel

Kernel is the core of the Linux system. The kernel controls the resources of the computer, allotting them to different users and tasks. The kernel interacts directly with the hardware, thus making the programs easy to write and portable across different platforms of hardware. Since the kernel communicates directly with the hardware, the parts of the kernel must be customized to the hardware

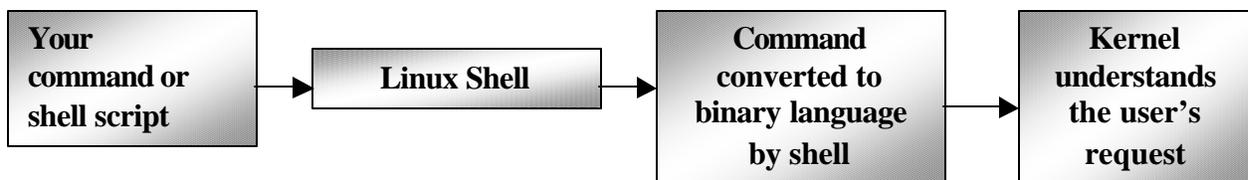
features of each system. However, the kernel does not deal directly with a user. Instead, the login process starts up a separate, interactive program called the shell for each user.

## Shell

Linux has a simple user interface called the shell that has the power to provide the services that a user wants. It protects the user from having to know the intricate hardware details.

Computers understand the language of 0's and 1's called binary language. In the early days of computing, instructions were provided using binary language, which is difficult for all of us, to read and write. So in OS, there is a special program called Shell. Shell accepts the user's instruction or commands in English and translates it into computer's native binary language.

This is what the shell does for us:



When the user types commands into the command-line interface in Linux, the user is actually interacting with the operating system via a special process called a *shell*.

## Linux utilities and Application programs

The Linux utilities or commands are a collection of programs that service day-to-day processing requirements. These programs are invoked through the shell, which is itself another utility. Apart from the utilities provided as part of the Linux operating system, more than a thousand Linux-based application programs, like database management systems, word processors and various other programs are available from independent software vendors.

## Features of Linux

- **Multitasking:** While it is possible for several users to run their respective jobs(also called processes) at the same time, linux also permits a user to run more than one job at a time.For example, say the ls command is issued, we do not have to wait for the \$ prompt to appear to enter the next commmand.We can relegate the ls job to the background, by simply affixing an &(ampersand) to the command and run another command say, the date command in the foreground. It therefore permits a user to perform less urgent jobs in the background,while doing the more important ones in the foreground.
- **Multi-user:** Linux is a multiuser operating system, where there can be up to a few hundred users sharing the system resources, based on the concept of time sharing, with each user allotted a time slice without the user being aware of it.Not more than one user license is required for multiple users using the operating system.
- **Multiplatform:** Linux runs on many different CPUs, not just Intel.
- **Multiprocessor:** Symmetric Multi Processing (SMP) support is available on the Intel and SPARC platforms and Linux is used in several loosely coupled MP applications.
- **Multithreading:** Linux has native kernel support for multiple independent threads of control within a single process memory space.

- **Portability:** Linux is an operating system written in high level language. This frees the operating system from depending on a particular computer design and its underlying architecture. Therefore, it is possible to move an entire operating system with its file system, utilities and application programs to a totally different machine.
- **Multiple virtual consoles:** Several independent login sessions through the console, you switch by pressing a hot-key combination. These are dynamically allocated.
- **Memory protection:** Linux has memory protection between processes, so that one program can't bring the whole system down.
- **Shared copy-on-write pages among executables:** This means that multiple process can use the same memory to run in. When one tries to write to that memory, that page (4KB piece of memory) is copied somewhere else. Copy-on-write has two benefits: increasing speed and decreasing memory use.
- **Unified Memory Pool:** A unified memory pool for user programs and disk cache, so that all free memory can be used for caching, and the cache can be reduced when running large programs.
- **Open Systems:** All source code is available, including the whole kernel and all drivers, the development tools and all user programs; also, all of it is freely distributable. Plenty of commercial programs are being provided for Linux without source, but everything that has been free, including the entire base operating system, is still free. Also, one can add to the existing toolkit by simply writing a program and storing the executable in a separate area in the file system.
- **Pseudoterminals (pty's)** are made available in Linux.
- **Internationalization:** Linux provides support for many national or customized keyboards, and it is fairly easy to add new ones dynamically.
- **Support for FileSystems:** Linux supports several common filesystems, including minix, Xenix, and all the common system V filesystems, and has an advanced filesystem of its own, which offers filesystems of up to 4 TB, and names up to 255 characters long.
- **Networking:** TCP/IP networking, including ftp, telnet, NFS, etc are supported by Linux.

## Linux Versions

Linux version 0.02 was introduced in 1991. Torvalds and hundreds of developers from across the world worked on it and in March 1994, version 1.0 of the Linux kernel was released. Red Hat Linux 6.0 uses version 2.2.5 of the Linux kernel.

## Linux Flavors

The various flavors of Linux available are:

**Bastille Linux** - Creates a secure Linux distribution for use on the desktop, the main use being that network administrators can give out Linux CD's to their users without worrying about security. Bastille Linux will be using Red Hat 6.0 as their basis.

**Caldera Systems Open Linux** - The installation procedure of Caldera provides a wizard-like interface. It comes along with application software such as C++ compiler, Java IDE and Netscape Communicator. It is Y2K compliant.

**Debian Linux** - High-quality non-commercial Linux distribution.

**eIT's easyLinux** - Graphical installation in beginner, advanced and professional modes.

**Linux Mandrake** - Integrates the graphical user interface KDE with the RedHat distribution.

**Redhat Linux** - Red Hat is among the most popular distributors of Linux. It supports a wide variety of hardware as it has many drivers for various hardware devices. The Red Hat 6.0 distribution consists of an application CD with a fully functional Java IDE (Integrated Development Environment), a fully functional C++ IDE and a standard C++ compiler. It also includes Netscape Communicator as part of its distribution. It is Y2K compliant.

**Secure Linux** - Creating a secured Linux for (internet) server systems. This distribution will feature something like a package system or at least proper file accounting. Strong cryptography and extra web server functionality will most likely be included as well.

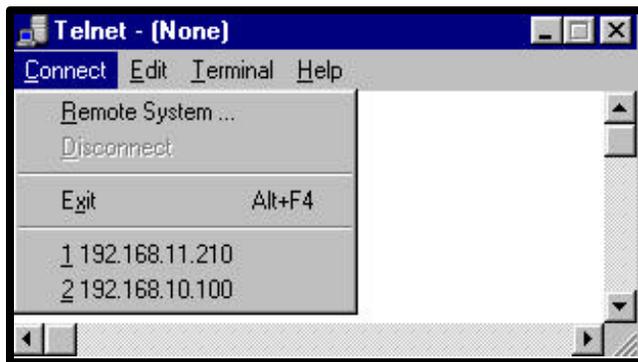
**Slackware Linux** - Provides the latest news about the Slackware Linux distribution, as well as installation help, configuration help, and other general information about Slackware Linux.

**SuSE Linux** - SuSE Linux is the full-featured package, including support and manual. SuSE Linux is the first Linux distribution, which supports installation and applications in Braille and therefore is suitable for partially sighted and blind people.

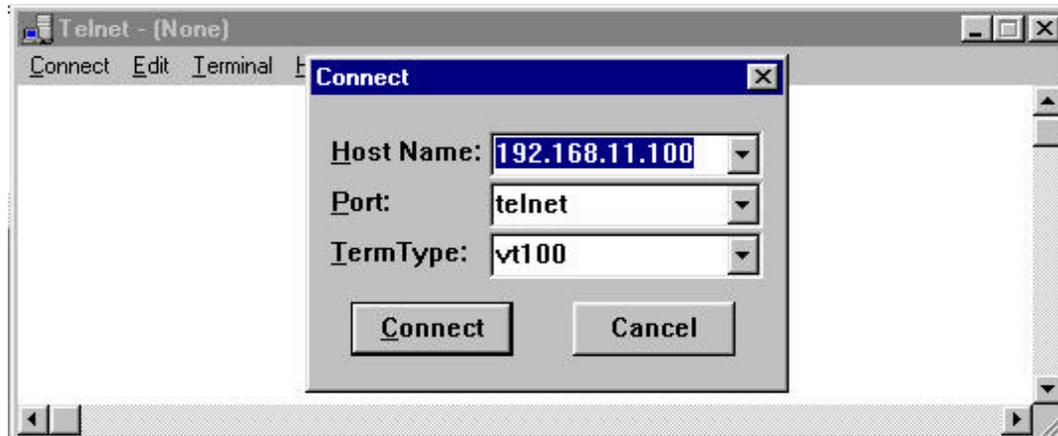
**Trinux Linux** - Trinux is a portable Linux distribution that boots from a single floppy disk, loads its packages from a FAT/Ext2 partition, floppy disks, or HTTP/FTP servers, and runs entirely in RAM.

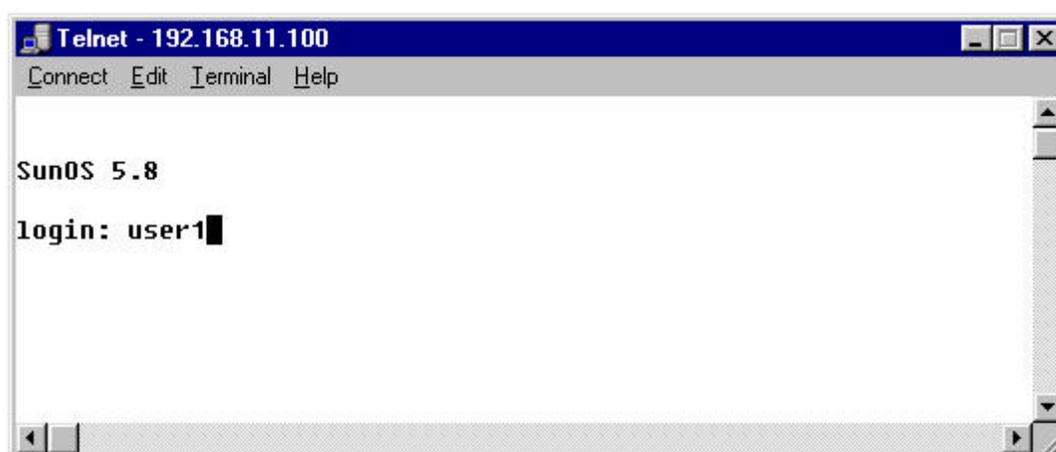
## Login and Logout in Linux

To connect to the Linux system, go to Start→Run and type telnet. The following screen appears:



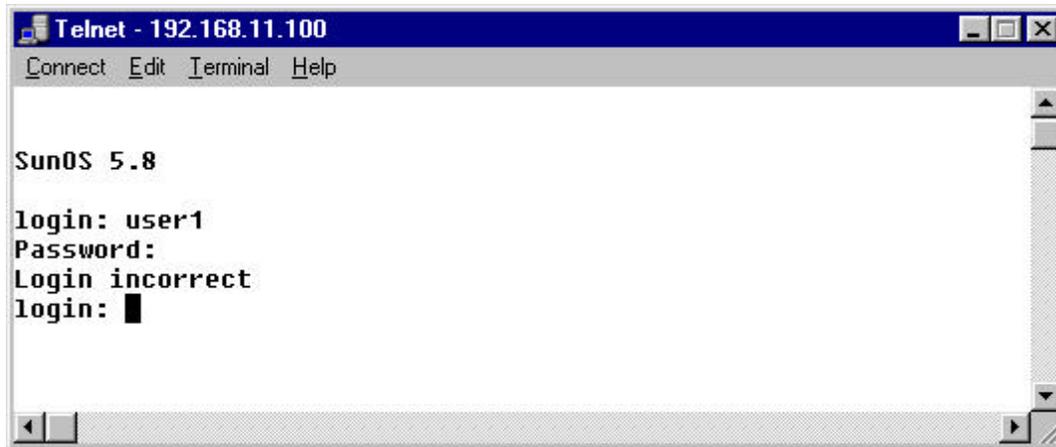
Choose **Connect→Remote System** from the menu. The following screen appears:





Specify the host name and click **Connect**. After connecting to the Linux system, a message similar to the one shown below appears at the terminal

Each user is provided with a user name or login name, which is to be entered at the login prompt. After the login name has been entered, the user has to enter his password. The characters entered for the password won't be echoed to the screen, so the user has to make sure that he has typed the password carefully. Otherwise an error message will be displayed on the screen.

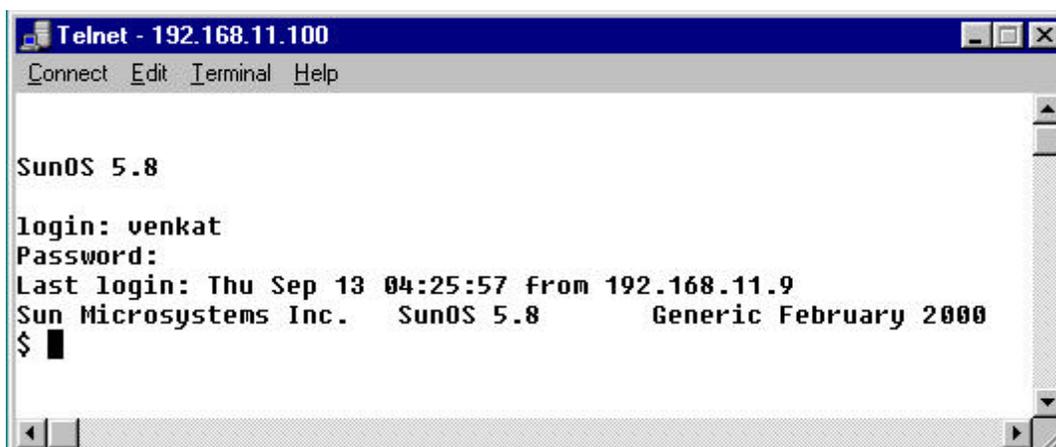


```
Telnet - 192.168.11.100
Connect Edit Terminal Help

SunOS 5.8

login: user1
Password:
Login incorrect
login: █
```

Linux keeps track of user names and information about users in special files (the shadow and passwd files under the /etc directory). When the user enters his login name and password, these are checked in these special files. If the user name does not match any of the names in the file, the login name is displayed again. Thus Linux ensures that only authorized users can access the machine. When a valid user name and password is entered at the terminal, the \$ symbol appears on the screen. This is called the shell prompt.



```
Telnet - 192.168.11.100
Connect Edit Terminal Help

SunOS 5.8

login: venkat
Password:
Last login: Thu Sep 13 04:25:57 from 192.168.11.9
Sun Microsystems Inc. SunOS 5.8 Generic February 2000
$ █
```

Login names and passwords are case sensitive and have to be typed carefully.

#### NOTE

Throughout this book we will be working with the user name **user1** and the password **transed**.

## Virtual consoles

The system's **console** is the monitor and keyboard connected directly to the system. Linux, like some other versions of UNIX, provides access to **virtual consoles** (or VCs), that let the user have more than one login session on the console at one time.

To demonstrate this, log in to your system. Next, press Alt-F2. You should see the login: prompt again. You're looking at the second virtual console. To switch back to the first VC, press Alt-F1. You are now back to your first login session.

A newly-installed Linux system probably lets the user to access only the first half-dozen or so VCs, by pressing Alt-F1 through Alt-F4, or however many VCs are configured on the system. It is possible to enable up to 12 VCs--one for each function key on the keyboard. Use of VCs can be very powerful because the user can work in several different sessions at the same time.

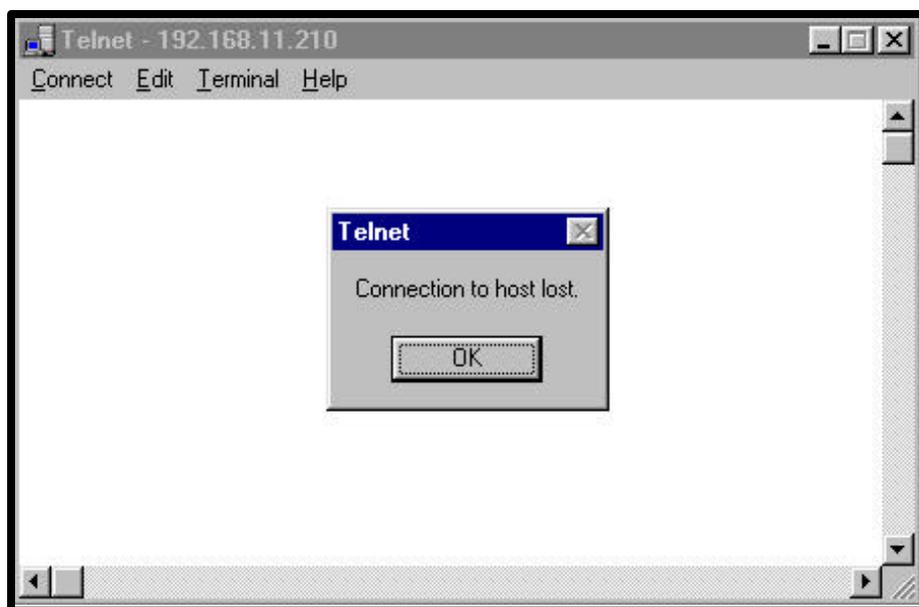
While the use of VCs is somewhat limiting (after all, you can look at only one VC at a time), it should give you a feel for the multiuser capabilities of Linux. While you're working on the first VC, you can switch over to the second VC and work on something else.

### NOTE

Virtual Consoles are available only in the Server.

## Logging out of a Linux Session

Once the user has logged on, the session continues till he instructs the shell to terminate the session. The current Linux session gets terminated on typing `exit` or `logout` at the shell prompt.



---

**NOTE**

Always remember to log out before leaving your terminal in order to maintain the security of files.

## Date and time

The date command can be used to display the current date and time.

```
$ date
Tue May 08 16:17:55 IST 2001
```

There are various options of the date command, which can be used to format the date and time before displaying. The options are given in the table below:

Option	Function
%D	Date as mm/dd/yy
%T	Time as HH:MM:SS
%r	Time in AM/PM representation
%m	Month of the year(in digits)
%d	Day of the month (in digits)
%Y	Year(last two digits)
%H	Hour(00 to 23)
%M	Minutes (00 to 59)
%S	Seconds (00 to 59)

For example, the current time can be displayed by the command

```
$ date +%T
10:38:34
```

```
$ date +%a
Fri
```

```
$ date +%D
05/08/01
```

It is also possible to include messages within the date command.

```
$ date "+Today is : %D"
Today is 05/08/01
```

## Calender

The cal command displays a simple calendar.

```

Telnet - 192.168.11.100
Connect Edit Terminal Help
$ cal
  September 2001
S  M Tu  W Th  F  S
      1
  2  3  4  5  6  7  8
  9 10 11 12 13 14 15
 16 17 18 19 20 21 22
 23 24 25 26 27 28 29
 30
$

```

The options that can be used with the cal command are:

Option	Significance
-m	displays the calendar and keeps Monday as the first day of the week
-j	displays the calendar where the days passed from January 1 are shown
-y	displays the calendar for the entire current year
Yyyy	displays the calendar for the year specified. The year should be between 1 and 9999

To display the calendar for a particular month of a certain year, the following command can be used,

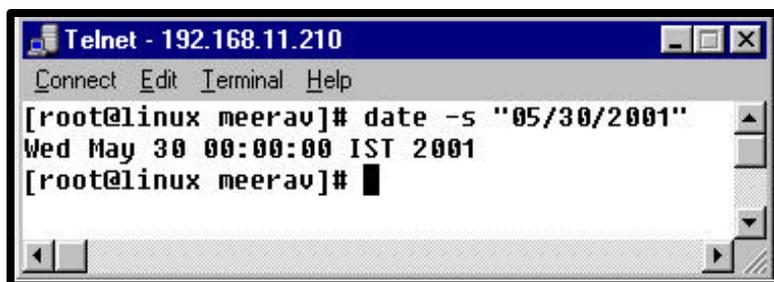
```

Telnet - 192.168.11.100
Connect Edit Terminal Help
$ cal 4 2001
  April 2001
S  M Tu  W Th  F  S
  1  2  3  4  5  6  7
  8  9 10 11 12 13 14
 15 16 17 18 19 20 21
 22 23 24 25 26 27 28
 29 30
$

```

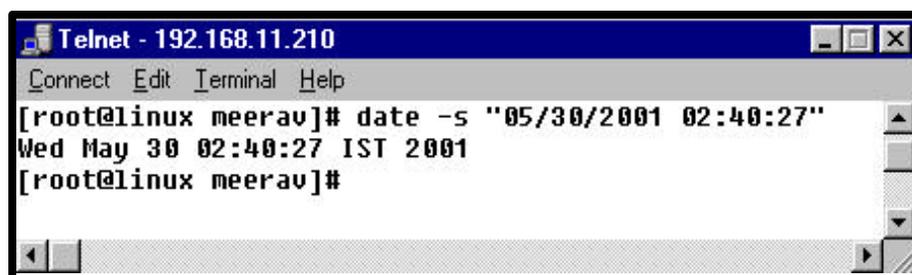
The above command (`$ cal 4 2001`) displays the calendar for the month April in the year 2001.

To change the system date, the following command can be used,



```
Telnet - 192.168.11.210
Connect Edit Terminal Help
[root@linux meerav]# date -s "05/30/2001"
Wed May 30 00:00:00 IST 2001
[root@linux meerav]#
```

To change the system time, use the following command



```
Telnet - 192.168.11.210
Connect Edit Terminal Help
[root@linux meerav]# date -s "05/30/2001 02:40:27"
Wed May 30 02:40:27 IST 2001
[root@linux meerav]#
```

## Other Linux Commands

Let us have a look at a few other Linux commands:

### who

The `who` command displays the details of all the users who are currently logged in.

```
$ who
meerav      tty2  May 11 16:17
root       tty3  May 11 15:46
root       :0    May 11 15:12
```

### who am i

The `who am i` command displays the name of the current user who has logged in.

```
$ who am i
linux!meerav      tty2  May 11 16:17
```

### man

The `man` command is used to display help on the Linux commands. It displays pages from the Linux reference manual that is installed along with the Linux OS.

---

For example to get detailed information about the who command, type as shown below:

```
$ man who
```

## clear

The clear command is used to clear the screen of its contents.

```
$ clear
```

### TOTAL RECALL

1. Differentiate single user and multi user operating systems.
2. What is the function of a kernel and a shell?
3. Explore virtual consoles.

### REVIEW QUESTIONS

#### Fill in the blanks

1. \_\_\_\_\_ is an example of a single user operating system.
2. The \_\_\_\_\_ command is used to display help on common Linux commands.
3. The date command option %r displays \_\_\_\_\_.
4. The \_\_\_\_\_ command is used to display details of users who have logged in.
5. The shell prompt is denoted by \_\_\_\_\_.