



Linux Mint 18

Cinnamon Edition

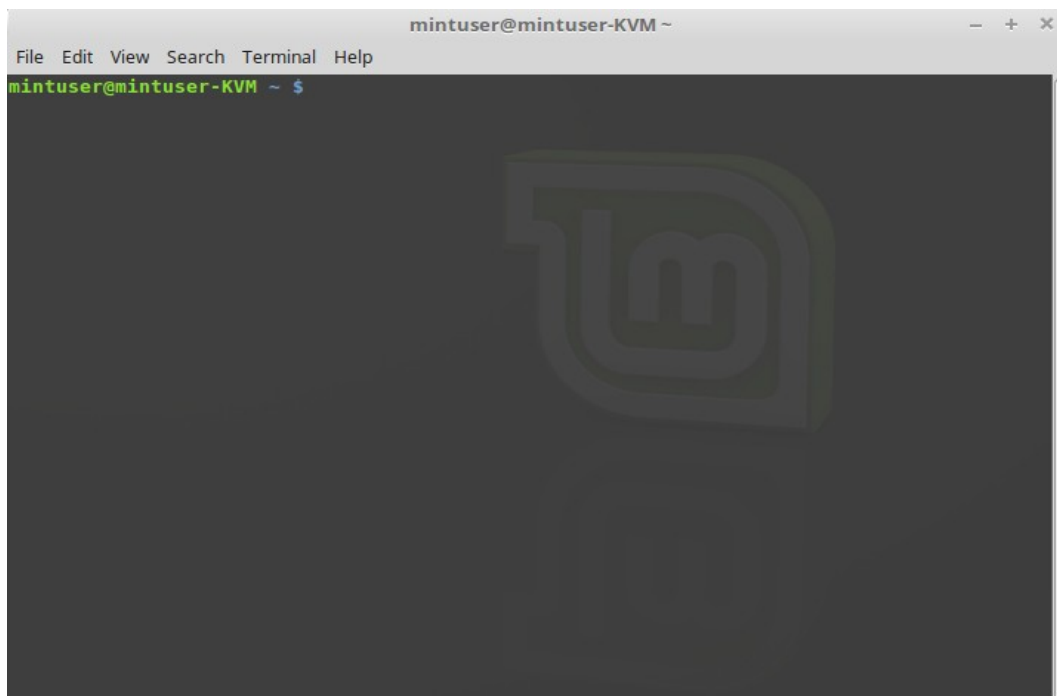
NOTE from mikeB of Code-it Software Solutions:

Some of this content was written by myself BUT I “pirated” some of this content from the original copy from the “Linux Mint” web site and other sources that are noted within this document.

Using the powerful 'Terminal'

You're most likely thinking to yourself “Why in the heck would I want to revert back to the old MS-DOS way of doing things? That's so old school since GUI became popular way back in the 1980's.”

My answer to that would be that I'm 90% positive that if you try some of the sample input commands I'm going to show on this document that you'll spend a lot of time developing your “keyboard ninja” skills. There is a lot of good books that go into this deeper and might be well worth the money to advance your skills – of course if you follow my video tutorials that will be offered in the future – that will work too ;-). It's much more fun then clicking a button that we've grown use to with GUI.



The “Terminal” button launches an application called “Terminal” which lets you enter commands directly using the keyboard. If you’re used to Microsoft Windows you may think that this is old hat, because in Windows the command line hasn’t progressed much further than the “DOS prompt” and is hidden away in an accessories menu.

One of the ways in which Linux systems differ from Windows systems is that the terminal is quite important in Linux, as it is often used as a way of getting more direct control over one's computer. We’d probably agree that the Terminal isn’t the most visually appealing application included in Linux Mint, but it is worth knowing that it is by far the most powerful, and once learned it’s actually not very hard to use.

In fact, it is worthwhile understanding that *every* command that you execute using a graphical desktop environment goes through the Terminal. When you click on an icon on the menu, for example, you are instructing Cinnamon to pass a textual instruction to the Terminal.

As an instructional exercise, if you right click “Menu” on the bottom left of the desktop, choose configure, click “Open the menu editor” and navigate to any of the application menus, pick an application, and click on the properties button, you will see in the “Command” field the text command that is passed to the system when you click on that application's entry in the menu. In other words, you have already been using the Terminal for some time, perhaps without realizing it. All that was different was that, instead of you typing out the commands, your desktop (Cinnamon) was doing it for you (which is a good thing because nobody wants to remember hundreds of application names all the time).

Sooner or later, though, you will probably be in a situation which causes you to use the Terminal directly, either to access commands that are not available through any GUI, or to get a job done more efficiently. Yes, you read that right! It can be more efficient to type a single command, for certain tasks, than to open up lots of windows to achieve the same thing. The more you use it, the more you’ll come to actually like it.

Remember how you didn’t like certain things when you were a kid and how you couldn’t do without them now? Terminal is one of these things. In a few weeks, you’ll be completely addicted to it. You’ll begin to feel in complete control of your computer. There will be times when you won’t even have to use the Terminal, but you will anyway, because for some tasks it’s faster, more accurate, more versatile and actually simpler to use than equivalent graphical interfaces. And anyone watching you will think you're a complete pro.

Here's some cool examples that you might want to try:

```
espeak "This is a test"
```

this command will speak the text string you provide

create or copy a txt file into your HOME folder (i.e. test.txt)

```
espeak -f test.txt
```

this command will “speak” the complete file

create wav file from a text file (text 2 speech)

```
espeak -f t1.txt --stdout > t1.wav
```

(“t1.txt being the name of a file within your HOME folder – t1.wav being the name of the created wav file that will reside in the same folder as the txt file)

to play an audio file from the command line (“myfile” = the name and extension of the audio file)...
play myfile.wav
play myfile.mp3

the “figlet” command turns ordinary terminal text into big fancy letters
figlet Do you want to play a game ?

What to get your daily fortune? Try these commands just for the fun of it:

```
fortune | cowsay  
fortune | cowsay -f tux  
cowsay -f tux "This is my text."
```

I suspect that you’d like to see how fast you can download (save as video or mp3) Youtube videos?

This is one of my favorite apps (Clipgrab) that is simple to use and FAST!

To install it from the command line (one command at a time):

```
sudo add-apt-repository ppa:clipgrab-team/ppa  
sudo apt-get update  
sudo apt-get install clipgrab
```

See how fast it downloads and installs using the Terminal? Now give the app a test drive – you should now have an icon to activate it within your Menu > “Internet” folder. Tip: use the “Search” function to show videos that match your input (I.e.”Paul Simon”)

The above samples will give you a “taste” of just some of the things you can accomplish (or waste your time being amazed) with Terminal command lines – if you are anything like me – you’ll get to love’n it ;-)

LINUX COMMAND LINE CHEAT SHEET

SYSTEM INFORMATION

```
uname -a # Display Linux system information  
uname -r # Display kernel release information  
cat /etc/redhat-release # Show which version of redhat installed  
uptime # Show how long the system has been running + load  
hostname # Show system host name  
hostname -I # Display the IP addresses of the host  
last reboot # Show system reboot history  
date # Show the current date and time  
cal # Show this month's calendar  
w # Display who is online  
whoami # Who you are logged in as
```

HARDWARE INFORMATION

```
dmesg # Display messages in kernel ring buffer  
cat /proc/cpuinfo # Display CPU information  
cat /proc/meminfo # Display memory information  
free -h # Display free and used memory ( -h for human readable, -m for MB, -g for GB.)  
lspci -tv # Display PCI devices  
lsusb -tv # Display USB devices
```

dmidecode # Display DMI/SMBIOS (hardware info) from the BIOS
hdparm -i /dev/sda # Show info about disk sda
hdparm -tT /dev/sda # Perform a read speed test on disk sda
badblocks -s /dev/sda # Test for unreadable blocks on disk sda

PERFORMANCE MONITORING AND STATISTICS

top # Display and manage the top processes
htop # Interactive process viewer (top alternative)
mpstat 1 # Display processor related statistics
vmstat 1 # Display virtual memory statistics
iostat 1 # Display I/O statistics
tail 100 /var/log/messages # Display the last 100 syslog messages
(Use /var/log/syslog for Debian based systems.)
tcpdump -i eth0 # Capture and display all packets on interface eth0
tcpdump -i eth0 'port 80' # Monitor all traffic on port 80 (HTTP)
lsof # List all open files on the system
lsof -u user # List files opened by user
free -h # Display free and used memory (-h for human readable, -m for MB, -g for GB.)
watch df -h # Execute "df -h", showing periodic updates

USER INFORMATION AND MANAGEMENT

id # Display the user and group ids of your current user.
last # Display the last users who have logged onto the system.
who # Show who is logged into the system.
w # Show who is logged in and what they are doing.
groupadd test # Create a group named "test".
useradd -c "John Smith" -m john # Create an account named john, with a comment of "John Smith" and create the user's home directory.
userdel john # Delete the john account.
usermod -aG sales john # Add the john account to the sales group

FILE AND DIRECTORY COMMANDS

ls -al # List all files in a long listing (detailed) format
pwd # Display the present working directory
mkdir directory # Create a directory
rm file # Remove (delete) file
rm -r directory # Remove the directory and its contents recursively
rm -f file # Force removal of file without prompting for confirmation
rm -rf directory # Forcefully remove directory recursively
cp file1 file2 # Copy file1 to file2
cp -r source_directory destination
Copy source_directory recursively to destination . If destination exists, copy source_directory into destination ,otherwise create destination with the contents of source_directory .
mv file1 file2 # Rename or move file1 to file2 . If file2 is an existing directory, move file1 into directory file2
ln -s /path/to/file linkname # Create symbolic link to linkname
touch file # Create an empty file or update the access and modification times of file.
cat file # View the contents of file
less file # Browse through a text file
head file # Display the first 10 lines of file

tail file # Display the last 10 lines of file
tail -f file # Display the last 10 lines of file and "follow" the file as it grows.

PROCESS MANAGEMENT

ps # Display your currently running processes
ps -ef # Display all the currently running processes on the system.
ps -ef | grep processname # Display process information for processname
top # Display and manage the top processes
htop # Interactive process viewer (top alternative)
kill pid # Kill process with process ID of p id
killall processname # Kill all processes named processname
program & # Start program in the background
bg # Display stopped or background jobs
fg # Brings the most recent background job to foreground
fg n # Brings job n to the foreground

FILE PERMISSIONS

PERMISSION EXAMPLE

U G W
rwx rwx rwx chmod 777 filename # Use sparingly!
rwx rwx r-x chmod 775 filename
rwx r-x r-x chmod 755 filename
rw- rw- r-- chmod 664 filename
rw- r-- r-- chmod 644 filename

LEGEND

U = User
G = Group
W = World
r = Read
w = write
x = execute
- = no access

NETWORKING

ifconfig -a # Display all network interfaces and ip address
ifconfig eth0 # Display eth0 address and details
ethtool eth0 # Query or control network driver and hardware settings
ping host # Send ICMP echo request to host
whois domain # Display whois information for domain
dig domain # Display DNS information for domain
dig -x IP_ADDRESS # Reverse lookup of IP_ADDRESS
host domain # Display DNS ip address for domain
hostname -i # Display the network address of the host name.
hostname -l # Display all local ip addresses
wget http://domain.com/file # Download http://domain.com/file
netstat -nutlp # Display listening tcp and udp ports and corresponding programs

ARCHIVES (TAR FILES)

```
tar cf archive.tar directory # Create tar named archive.tar containing
directory .
tar xf archive.tar # Extract the contents from archive.tar .
tar czf archive.tar.gz directory # Create a gzip compressed tar file name
archive.tar.gz .
tar xzf archive.tar.gz # Extract a gzip compressed tar file.
tar cjf archive.tar.bz2 directory # Create a tar file with bzip2 compression
tar xjf archive.tar.bz2 # Extract a bzip2 compressed tar file.
```

INSTALLING PACKAGES

```
yum search keyword # Search for a package by keyword .
yum install package # Install package .
yum info package # Display description and summary information
about package .
rpm -i package.rpm # Install package from local file named
package.rpm
yum remove package # Remove/uninstall package .
tar zxvf sourcecode.tar.gz
cd sourcecode
./configure
make
make install
# Install software from source code.
```

SEARCH

```
grep pattern file # Search for pattern in file
grep -r pattern directory # Search recursively for pattern in directory
locate name # Find files and directories by name
find /home/john -name
'prefix*'
# Find files in /home/john that start with "prefix".
find /home -size +100M # Find files larger than 100MB in /home
```

SSH LOGINS

```
ssh host # Connect to host as your local username.
ssh user@host # Connect to host as user
ssh -p port user@host # Connect to host using port
```

FILE TRANSFERS

```
scp file.txt server:/tmp # Secure copy file.txt to the /tmp folder on
server
scp server:/var/www/*.html /tmp # Copy *.html files from server to the local
/tmp folder.
scp -r server:/var/www /tmp # Copy all files and directories recursively from
server to the current system's /tmp folder.
rsync -a /home /backups/ # Synchronize /home to /backups/home
```

```
rsync -avz /home  
server:/backups/  
# Synchronize files/directories between the local  
and remote system with compression enabled
```

DISK USAGE

```
df -h # Show free and used space on mounted filesystems  
df -i # Show free and used inodes on mounted filesystems  
fdisk -l # Display disks partitions sizes and types  
du -ah # Display disk usage for all files and directories in  
human readable format  
du -sh # Display total disk usage off the current directory
```

DIRECTORY NAVIGATION

```
cd .. # To go up one level of the directory tree. (Change into  
the parent directory.)  
cd # Go to the $HOME directory  
cd /etc # Change to the /etc directory
```

LINUX

Hack'n the Linux - mikeB