# Operating System Diagnostics

In this tutorial we're going to use several diagnostic tools on the Linux Operating System according to specifications to identify any issues and modify the system as per the tool recommendations. We'll also carry out preventative maintenance to ensure our system remains in good health. Note that we'll only be examining some of the available tools and your encouraged to explore the OS to discover more administrative/diagnostic tools and their uses.

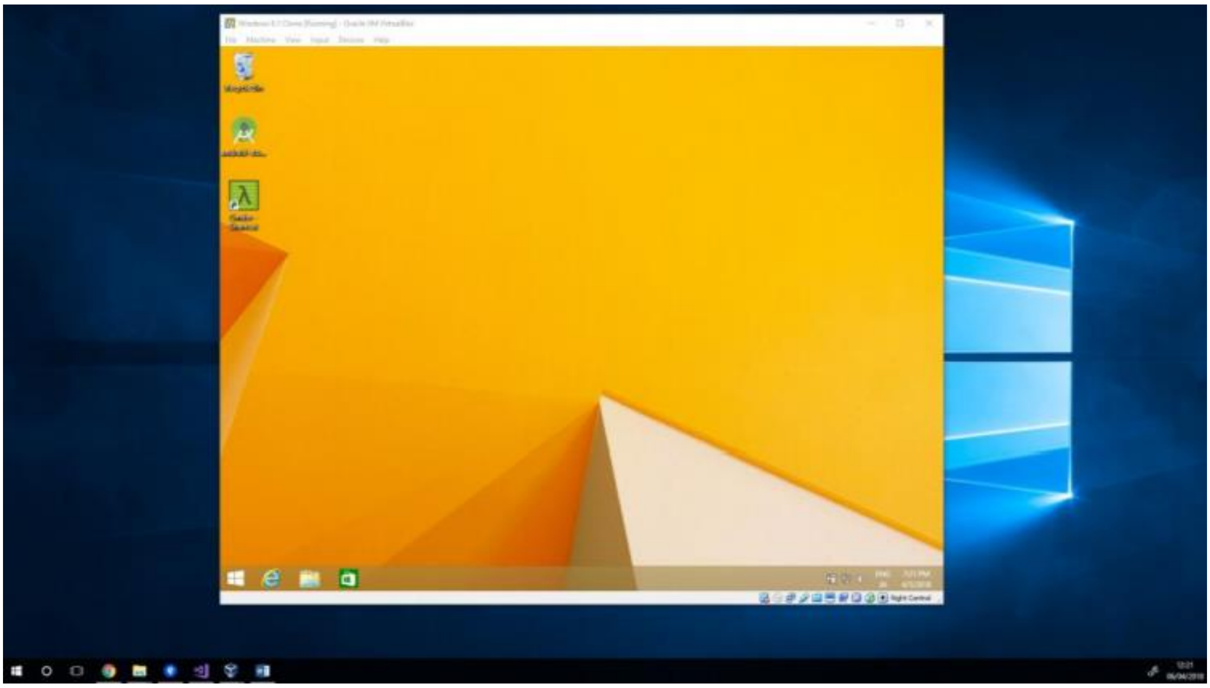
All testing will be performed in a *Virtual Machine (VM)*. A VM can be thought of as a computer inside a computer. Essentially, a VM runs on a host computer inside virtualisation-software. We'll use **Virtual Box** to host an instance of the Linux OS. The power of VMs come from the fact that we can sandbox/isolate the VM. We can also save the VM as a file on the host computer, make copies of the file and thus create copies of the VM, with all its files and settings.

**NOTE:** The document "*Assessment – Operating System Diagnostics.docx*" *is to be completed and submitted as part of your final assessment*. Make sure you take plenty of screenshots as you work through this tutorial as evidence.

## Contents

## Step 1: Installing Virtual Box (Optional)

Note: VirtualBox is already installed on the AIE network.  Skip this step if Virtual Box is already installed on your machine.

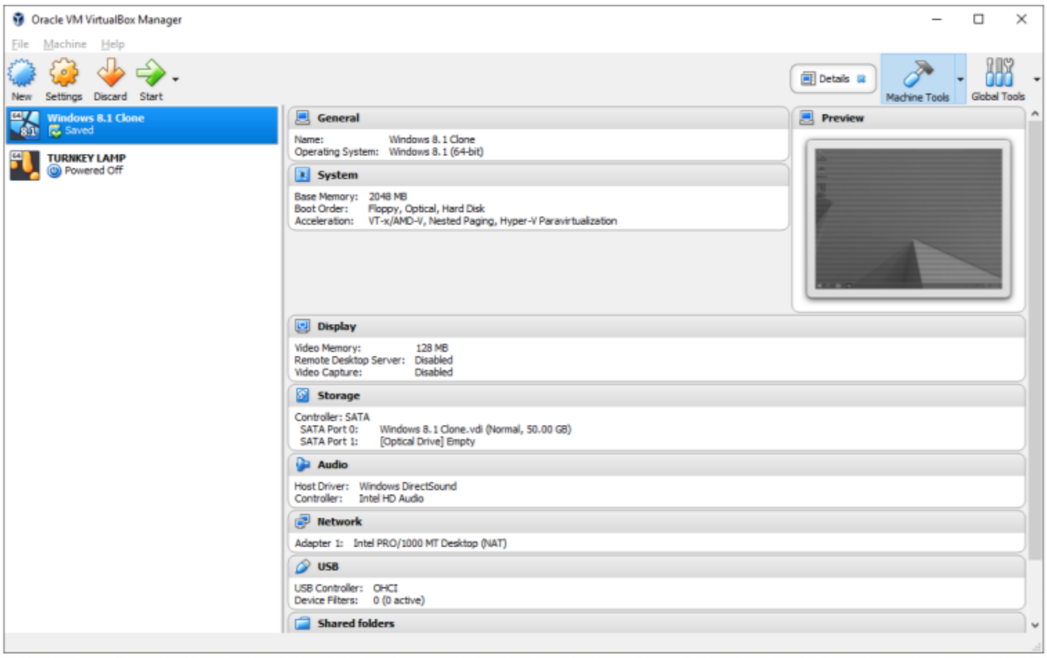


*Windows 8 running within a Virtual Box window on Windows 10*

Visit the VirtualBox website to download the latest version of the software:

[https://www.virtualbox.org/wiki/Downloads](https://www.virtualbox.org/wiki/Downloads)

Once Virtual Box is installed you should see a screen like the following. The list on the left includes your collection of VMs. The right panel shows details for the currently selected VM.



**General**

Name: Windows 8.1 Clone
Operating System: Windows 8.1 (64-bit)

**System**

Base Memory: 2048 MB
Boot Order: Floppy, Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging, Hyper-V Paravirtualization

**Display**

Video Memory: 128 MB
Remote Desktop Server: Disabled
Video Capture: Disabled

**Storage**

Controller: SATA
SATA Port 0: Windows 8.1 Clone.vdi (Normal, 50.00 GB)
SATA Port 1: [Optical Drive] Empty

**Audio**

Host Driver: Windows DirectSound
Controller: Intel HD Audio

**Network**

Adapter 1: Intel PRO/1000 MT Desktop (NAT)

**USB**

USB Controller: OHCI
Device Filters: 0 (0 active)

**Shared folders**

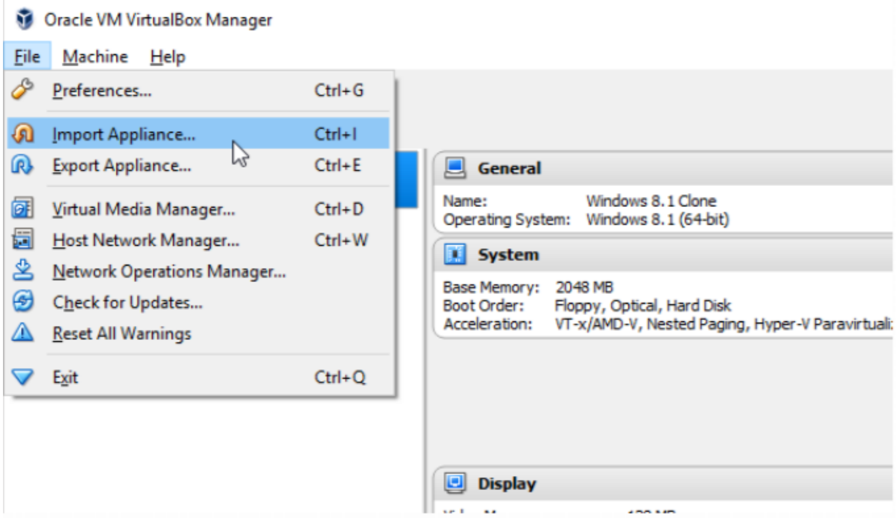
## Step 2: Installing the pre-configured Virtual Machines

We'll use *pre-configured VMs* for this tutorial, with *Linux Mint* already installed in different configurations. This will reduce the amount of work required, but feel free to explore creating your own VM and then installing an operating system from an ISO-file outside of class.

1. Download the Linux Mint VirtualBox Appliance file from *AIE's Canvas Platform*, if not done so already.

    a. **VirtualBox Appliance Filename:** *LinuxMint-AIE-VirusScanningAndBackups.ova*

2. Import the appliance into VirtualBox via the "**File->Import Appliance**" menu



Oracle VM VirtualBox Manager

File
Machine
Help
Preferences... Ctrl+G
Import Appliance... Ctrl+I
Export Appliance... Ctrl+E
Virtual Media Manager... Ctrl+D
Host Network Manager... Ctrl+W
Network Operations Manager...
Check for Updates...
Reset All Warnings
Exit Ctrl+Q
General
System
Display

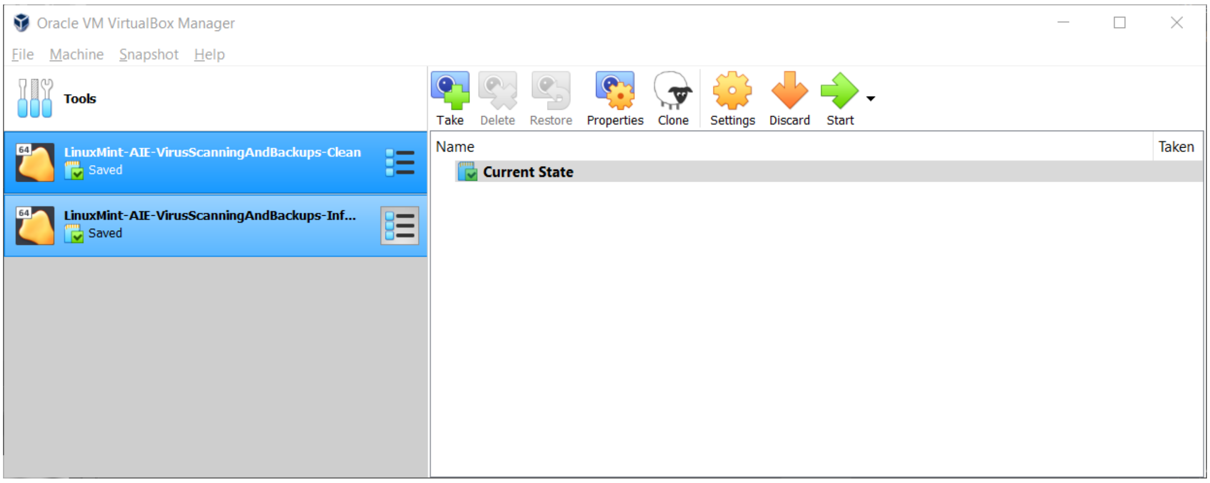
Name: Windows 8.1 Clone
Operating System: Windows 8.1 (64-bit)

Base Memory: 2048 MB
Boot Order: Floppy, Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging, Hyper-V Paravirtuali

3. The VM Appliance includes separate configurations of the Linux Mint Operating System. After importing the Appliance, you'll have the following VMs ready for use:

    a. *LinuxMint-AIE-VirusScanningAndBackups-Clean*

    b. *LinuxMint-AIE-VirusScanningAndBackups-Infected*



Oracle VM VirtualBox Manager

LinuxMint-AIE-VirusScanningAndBackups-Clean
Saved

LinuxMint-AIE-VirusScanningAndBackups-Inf...
Saved

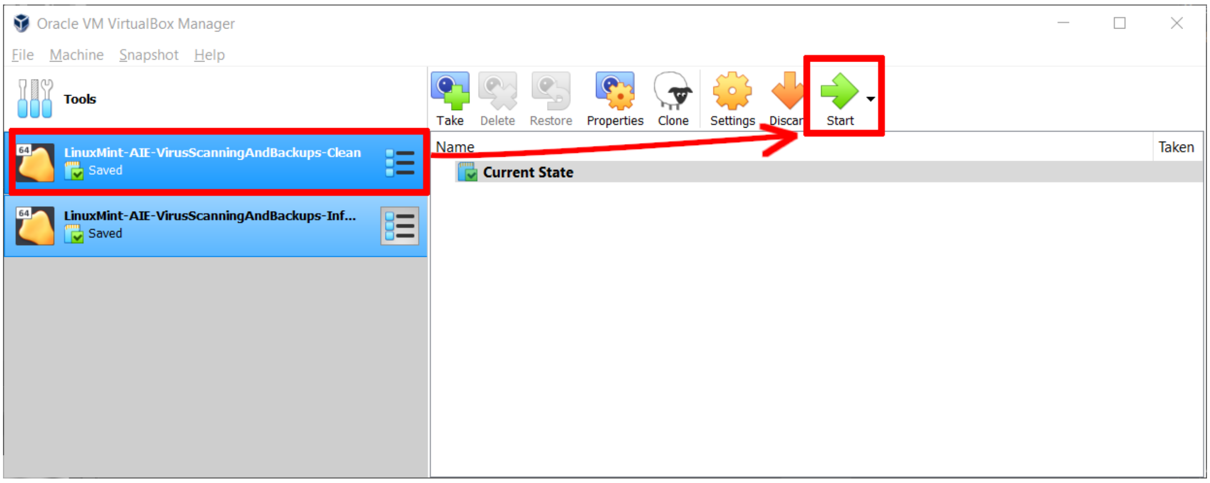
Current State

## Step 3: Starting a Virtual Machine

We're going to start the clean Virtual Machine by following these steps:

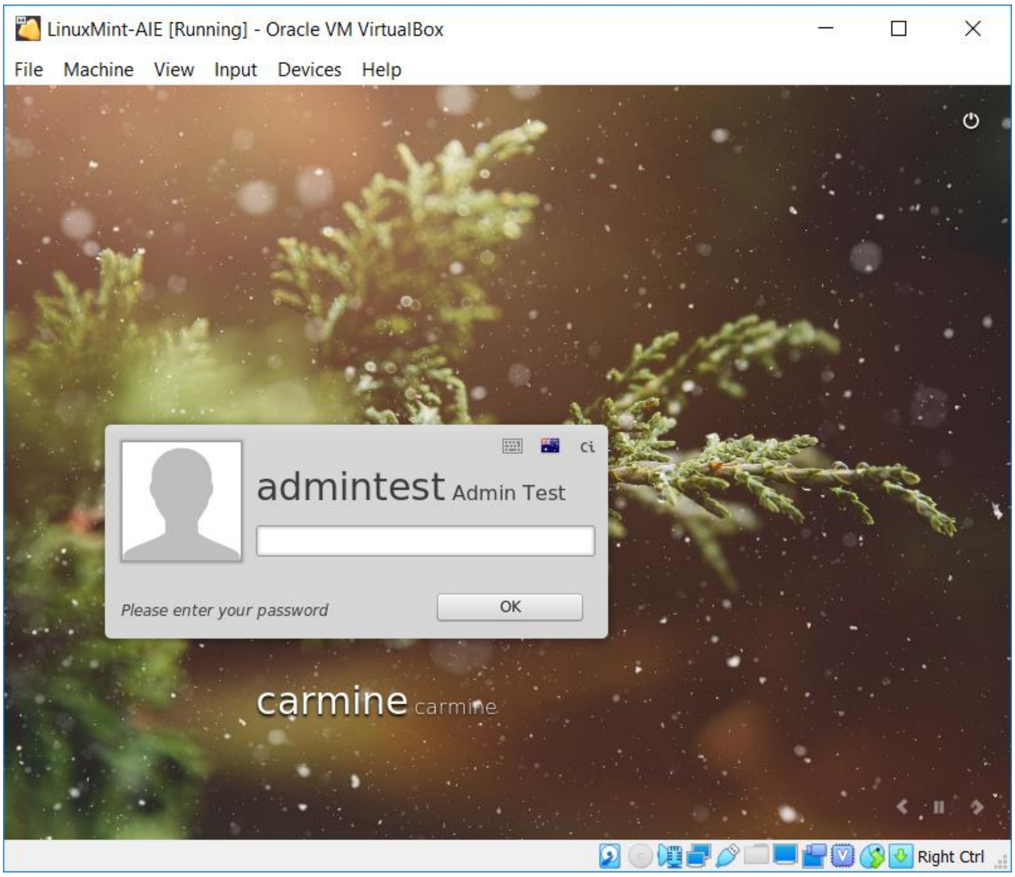
1. Select the "**LinuxMint-AIE-VirusScanningAndBackups-Clean**" Virtual Machine

2. Then press the **Start** button


Oracle VM VirtualBox Manager
File Machine Snapshot Help
Tools
Take Delete Restore Properties Clone Settings Discard Start
LinuxMint-AIE-VirusScanningAndBackups-Clean
Saved
LinuxMint-AIE-VirusScanningAndBackups-Inf...
Saved
Name
Taken
Current State

Once the VM has booted up, you'll be presented with the *Login screen*.

- Select the ***admintest*** account, ***password*** "admin1234"

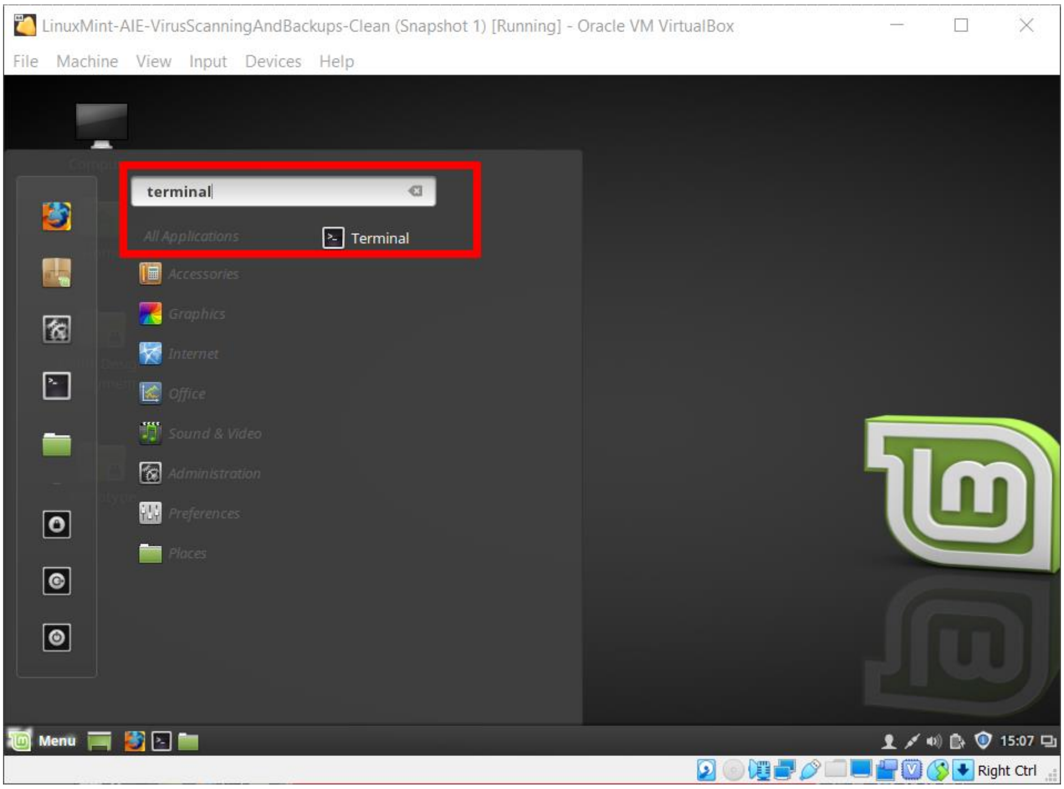

LinuxMint-AIE [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
admintest Admin Test
Please enter your password
OK
carmine carmine
Right Ctrl

## Step 4: Updating the Operating System

A regular preventative maintenance technique for all operating systems, including Linux, ensures that the OS is up-to-date. This avoids issues cropping up from bugs, errors and incompatibility issues in older OS versions.

This section explains how to update the Linux OS and installed packages via the Command Line Interface (CLI). Follow these steps to update the Linux OS packages from the CLI:

1. Open the **Terminal** from the Menu or Application Toolbar



2. Examine the installed packages on the system:

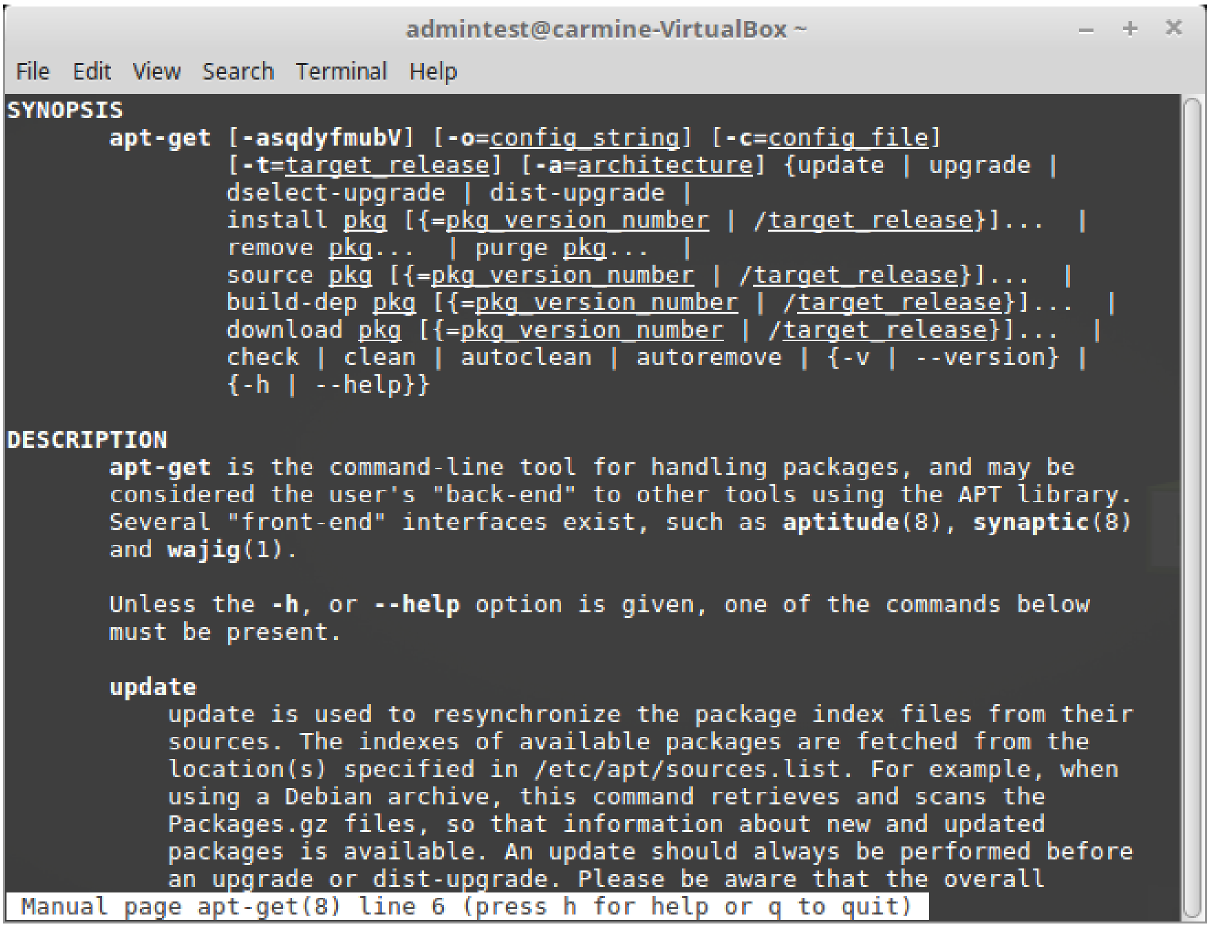
   a. Type "`apt list --installed`" in the terminal

   b. Look through the list to see which installed packages are out-of-date

3. The **`apt-get`** command-line utility is used by Linux to manage installed & downloaded packages on your Linux system. Technically, APT stands for Advanced Package Tool. A package can be considered a downloadable chunk of software, such as an application. Use the `apt-get` command, including command-line-options and specifications, by following these steps:

4. Type "**`man apt-get`**" in the terminal to read the help manual and specifications

   a. Pay attention to the "`update`", "`upgrade`" and "`dist-upgrade`" options


admintest@carmine-VirtualBox ~
File Edit View Search Terminal Help

```
SYNOPSIS
       apt-get [-asqdyfmubV] [-o=config_string] [-c=config_file]
               [-t=target_release] [-a=architecture] {update | upgrade |
               dselect-upgrade | dist-upgrade |
               install pkg [{=pkg_version_number | /target_release}]...  |
               remove pkg...  | purge pkg...  |
               source pkg [{=pkg_version_number | /target_release}]...  |
               build-dep pkg [{=pkg_version_number | /target_release}]...  |
               download pkg [{=pkg_version_number | /target_release}]...  |
               check | clean | autoclean | autoremove | {-v | --version} |
               {-h | --help}}

DESCRIPTION
       apt-get is the command-line tool for handling packages, and may be
       considered the user's "back-end" to other tools using the APT library.
       Several "front-end" interfaces exist, such as aptitude(8), synaptic(8)
       and wajig(1).

       Unless the -h, or --help option is given, one of the commands below
       must be present.

       update
           update is used to resynchronize the package index files from their
           sources. The indexes of available packages are fetched from the
           location(s) specified in /etc/apt/sources.list. For example, when
           using a Debian archive, this command retrieves and scans the
           Packages.gz files, so that information about new and updated
           packages is available. An update should always be performed before
           an upgrade or dist-upgrade. Please be aware that the overall
 Manual page apt-get(8) line 6 (press h for help or q to quit)
```

5. We'll now do an update, followed by a dist-upgrade to ensure our installed software packages are all up-to-date. It's very important to update before trying to install anything to avoid outdated files from the database

   a. Type "`sudo apt-get update`" in the terminal

   b. Once the update has finished, type "`sudo apt-get dist-upgrade`" in the terminal

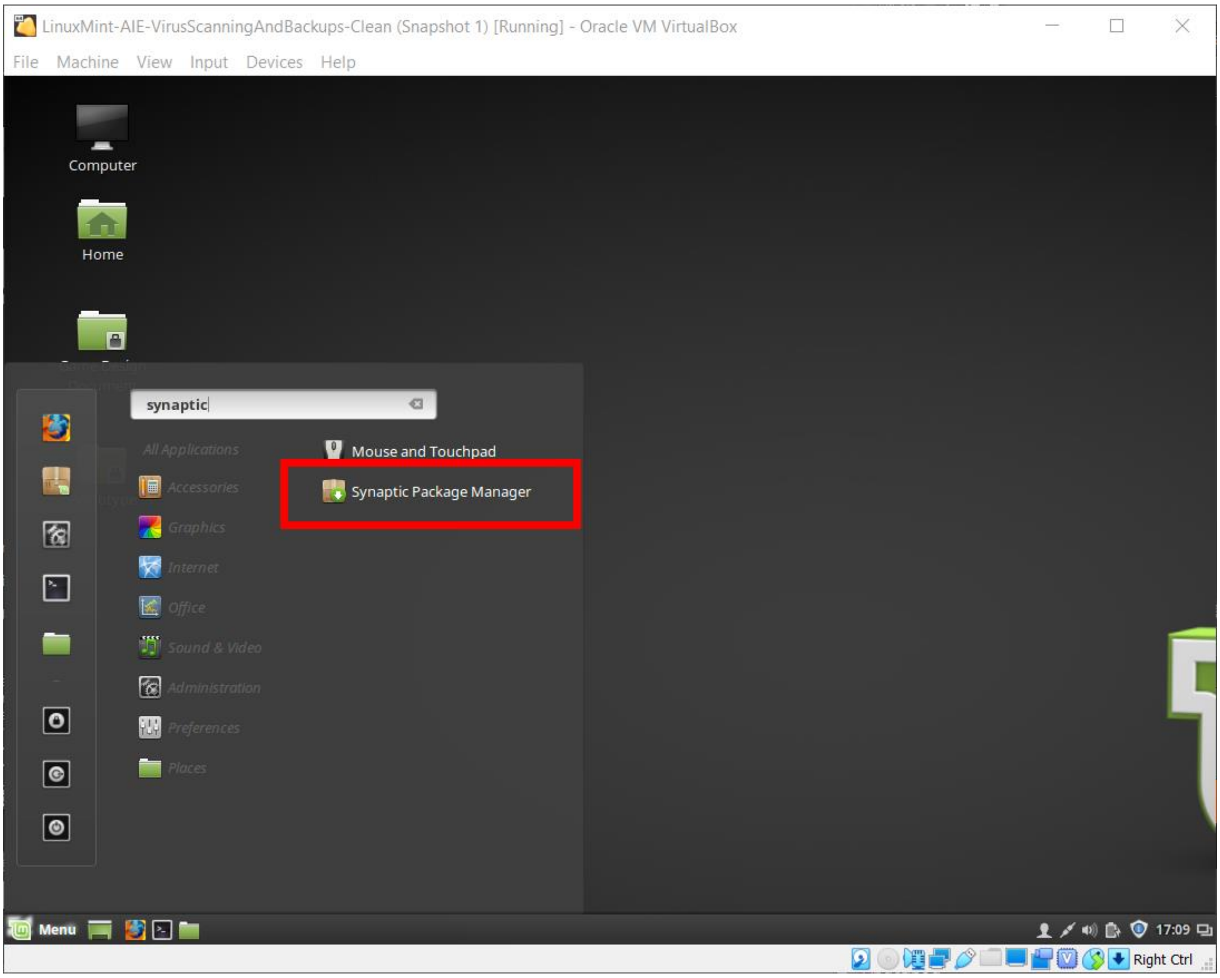
      - Note that the "`sudo`" prefix requests admin privileges and will usually require a password.

### Step 5: Installing software packages

Linux Mint includes many User-Interface tools to make administration and maintenance of the system simpler for users. One such example is the *Synaptic Package Manager*. We'll now use the tool to install a software package into the OS from a remote server.

Follow these steps to update the Linux OS packages from the CLI:

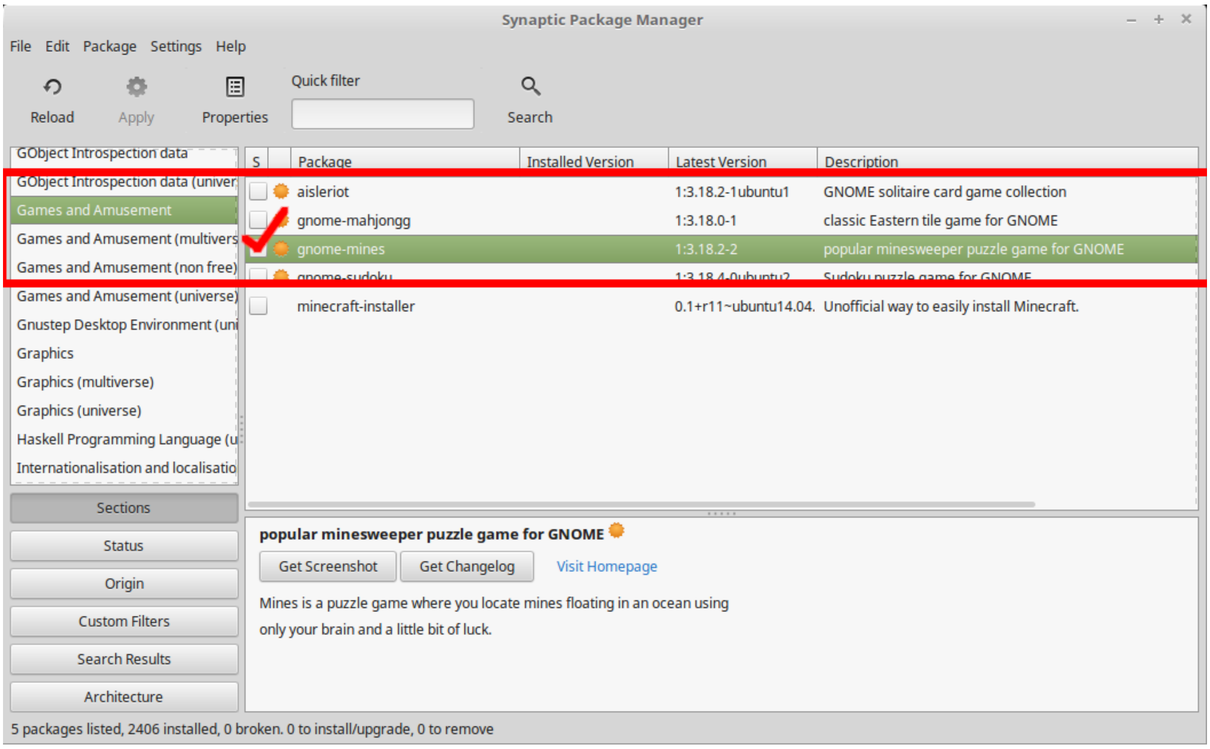
1. Open the **Synaptic Package Manager** from the **Administration** menu



2. Wait for the package manager to update its database…

3. Navigate to the **Games & Amusement** section

4. Check the **gnome-mines** option to install the Minesweeper game


Synaptic Package Manager
File Edit Package Settings Help
Reload
Apply
Properties
Quick filter
Search
GObject Introspection data
GObject Introspection data (univer
Games and Amusement
Games and Amusement (multivers
Games and Amusement (non free)
Games and Amusement (universe)
Gnustep Desktop Environment (uni
Graphics
Graphics (multiverse)
Graphics (universe)
Haskell Programming Language (u
Internationalisation and localisatio
Sections
Status
Origin
Custom Filters
Search Results
Architecture
Get Screenshot
Get Changelog
Visit Homepage

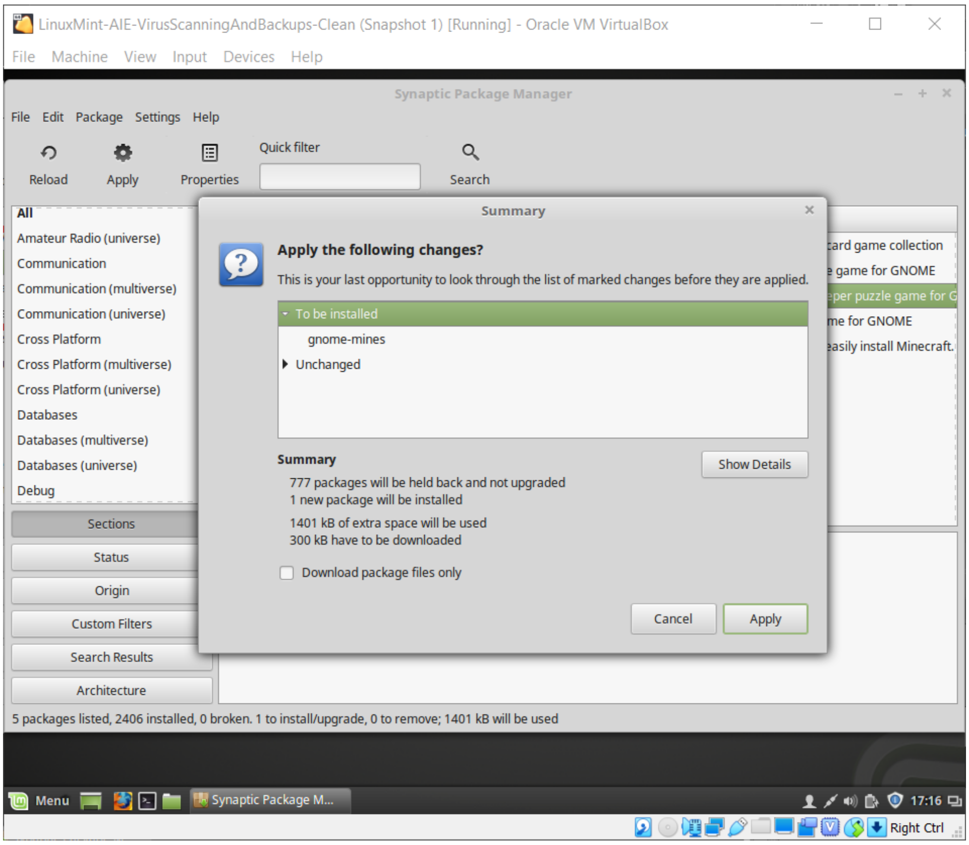
| S | Package | Installed Version | Latest Version | Description |
|---|---|---|---|---|
| □ | aisleriot | | 1:3.18.2-1ubuntu1 | GNOME solitaire card game collection |
| □ | gnome-mahjongg | | 1:3.18.0-1 | classic Eastern tile game for GNOME |
| ✓ | gnome-mines | | 1:3.18.2-2 | popular minesweeper puzzle game for GNOME |
| □ | gnome-sudoku | | 1:3.18.4-0ubuntu2 | Sudoku puzzle game for GNOME |
| □ | minecraft-installer | | 0.1+r11~ubuntu14.04. | Unofficial way to easily install Minecraft. |

**popular minesweeper puzzle game for GNOME**

Mines is a puzzle game where you locate mines floating in an ocean using only your brain and a little bit of luck.

5 packages listed, 2406 installed, 0 broken. 0 to install/upgrade, 0 to remove

5. Press the **Apply** button. A confirmation window will be displayed. Check the settings and then commence the installation by pressing **Apply** in the confirmation window.



**Apply the following changes?**

This is your last opportunity to look through the list of marked changes before they are applied.

- To be installed
  - gnome-mines
- Unchanged

**Summary**

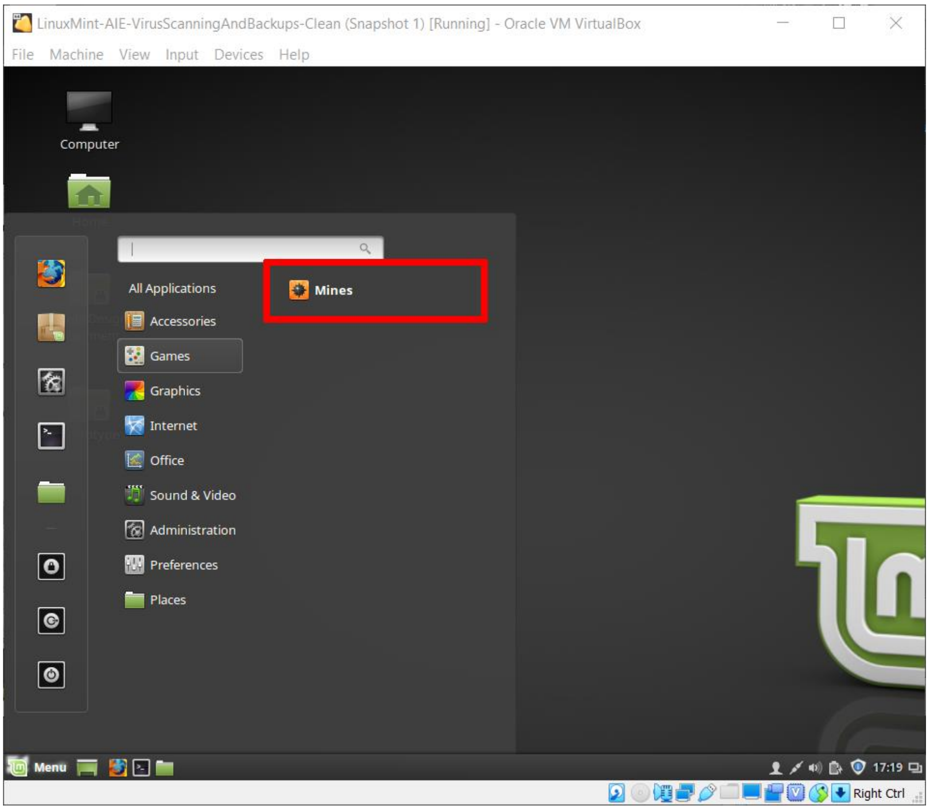
777 packages will be held back and not upgraded
1 new package will be installed

1401 kB of extra space will be used
300 kB have to be downloaded

□ Download package files only

5 packages listed, 2406 installed, 0 broken. 1 to install/upgrade, 0 to remove; 1401 kB will be used

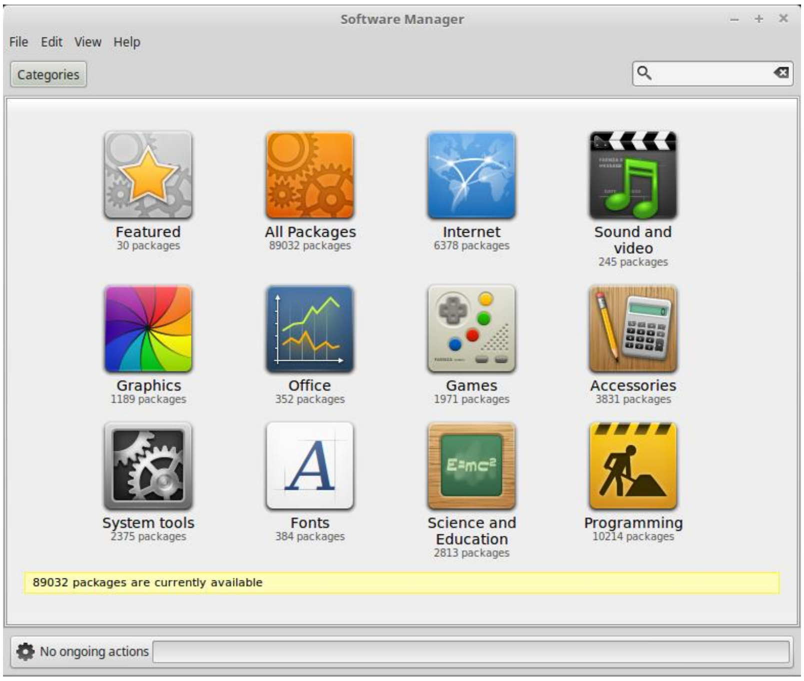
6. After the installation is complete, open the **Mines** program via the menu.



7. This process can be repeated for other applications via the Package Manager.

Next we’ll use a similar tool called the **Software Manager** from the **Administration** menu:

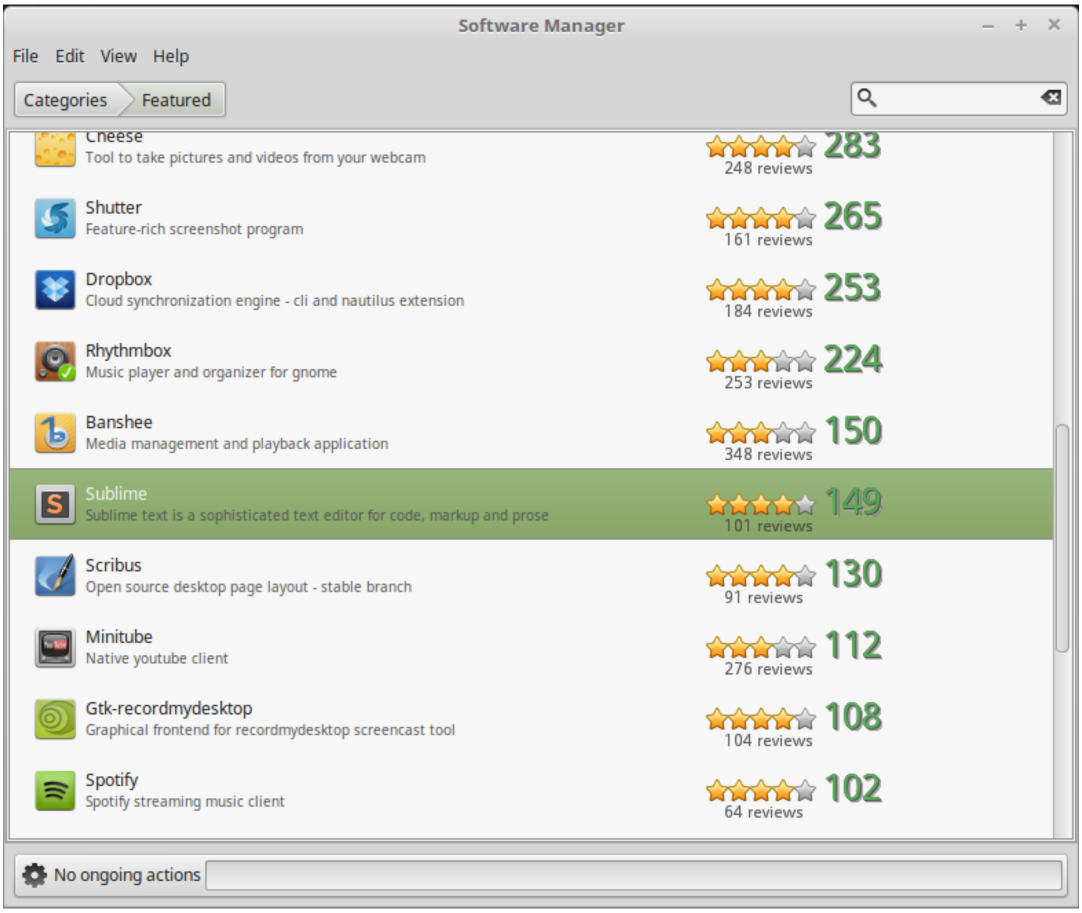
1. Open the **Software Manager** application from the **Administration** menu



89032 packages are currently available

No ongoing actions

2. Linux does have a few text-editors installed by default, but we'll install another called Sublime, which is a user-friendly Graphical Text Editor. **Double-click** the **Sublime** option from the *Featured* Category. Then press the **Install** button to install the software.



Cheese
Tool to take pictures and videos from your webcam
283
248 reviews

Shutter
Feature-rich screenshot program
265
161 reviews

Dropbox
Cloud synchronization engine - cli and nautilus extension
253
184 reviews

Rhythmbox
Music player and organizer for gnome
224
253 reviews

Banshee
Media management and playback application
150
348 reviews

Sublime
Sublime text is a sophisticated text editor for code, markup and prose
149
101 reviews

Scribus
Open source desktop page layout - stable branch
130
91 reviews

Minitube
Native youtube client
112
276 reviews

Gtk-recordmydesktop
Graphical frontend for recordmydesktop screencast tool
108
104 reviews

Spotify
Spotify streaming music client
102
64 reviews

No ongoing actions

3. Wait for the installation to complete…

4. You can now run the *Sublime text editor* from the *Main Menu*.

## Step 6: Scheduling regular backups

Most operating systems have mechanisms to schedule commands. This is especially useful when handling routine maintenance. It involves automatically running applications and scripts at pre-determined times (e.g. daily, weekly).

The Linux OS includes the "`tar`" command-line-tool to create archives (very similar to zip-files) without a graphical user-interface.

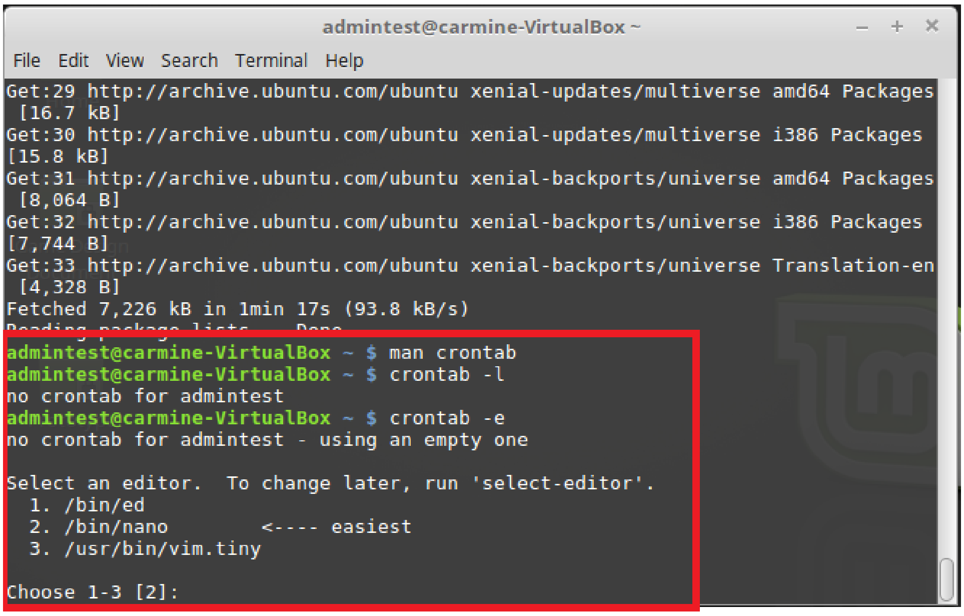
> We've seen in another tutorial how to backup our data using the **Backup Tool** that's part of the **Administration** menu in Linux Mint. This tool uses a graphical-user-interface which is difficult to automatically schedule because it requires manual user intervention. The tar command has no user-interface and can be scheduled/automated.

We'll use the **crontab** utility to automatically run the **tar** archiving tool. Follow this process to schedule automatic daily or weekly backups without any user interaction.

1. Open a Terminal. Type "`man crontab`" to learn more about the crontab utility.

2. Type "`crontab -l`" to list your current scheduled commands. Note that nothing is currently scheduled.

3. Type "`crontab -e`" to edit a new schedule. You'll be asked which editor to use – select the default (Nano)


admintest@carmine-VirtualBox ~
File Edit View Search Terminal Help

```
Get:29 http://archive.ubuntu.com/ubuntu xenial-updates/multiverse amd64 Packages
 [16.7 kB]
Get:30 http://archive.ubuntu.com/ubuntu xenial-updates/multiverse i386 Packages
[15.8 kB]
Get:31 http://archive.ubuntu.com/ubuntu xenial-backports/universe amd64 Packages
 [8,064 B]
Get:32 http://archive.ubuntu.com/ubuntu xenial-backports/universe i386 Packages
[7,744 B]
Get:33 http://archive.ubuntu.com/ubuntu xenial-backports/universe Translation-en
 [4,328 B]
Fetched 7,226 kB in 1min 17s (93.8 kB/s)
Reading package lists... Done
admintest@carmine-VirtualBox ~ $ man crontab
admintest@carmine-VirtualBox ~ $ crontab -l
no crontab for admintest
admintest@carmine-VirtualBox ~ $ crontab -e
no crontab for admintest - using an empty one

Select an editor.  To change later, run 'select-editor'.
  1. /bin/ed
  2. /bin/nano        <---- easiest
  3. /usr/bin/vim.tiny

Choose 1-3 [2]:
```

4. Each line in the crontab database is used to schedule a process. It has a special format which is documented in the crontab template file.

5. Scroll to the bottom of the file.

6. Add this line at the end-of-file to schedule a backup of your *Home* folder each night at 10:00pm:

   a. "`00 10 * * * tar -zcf ~/Desktop/backup.tgz ~/Home/`"

   b. Feel free to change the first two numbers to schedule a different time each day

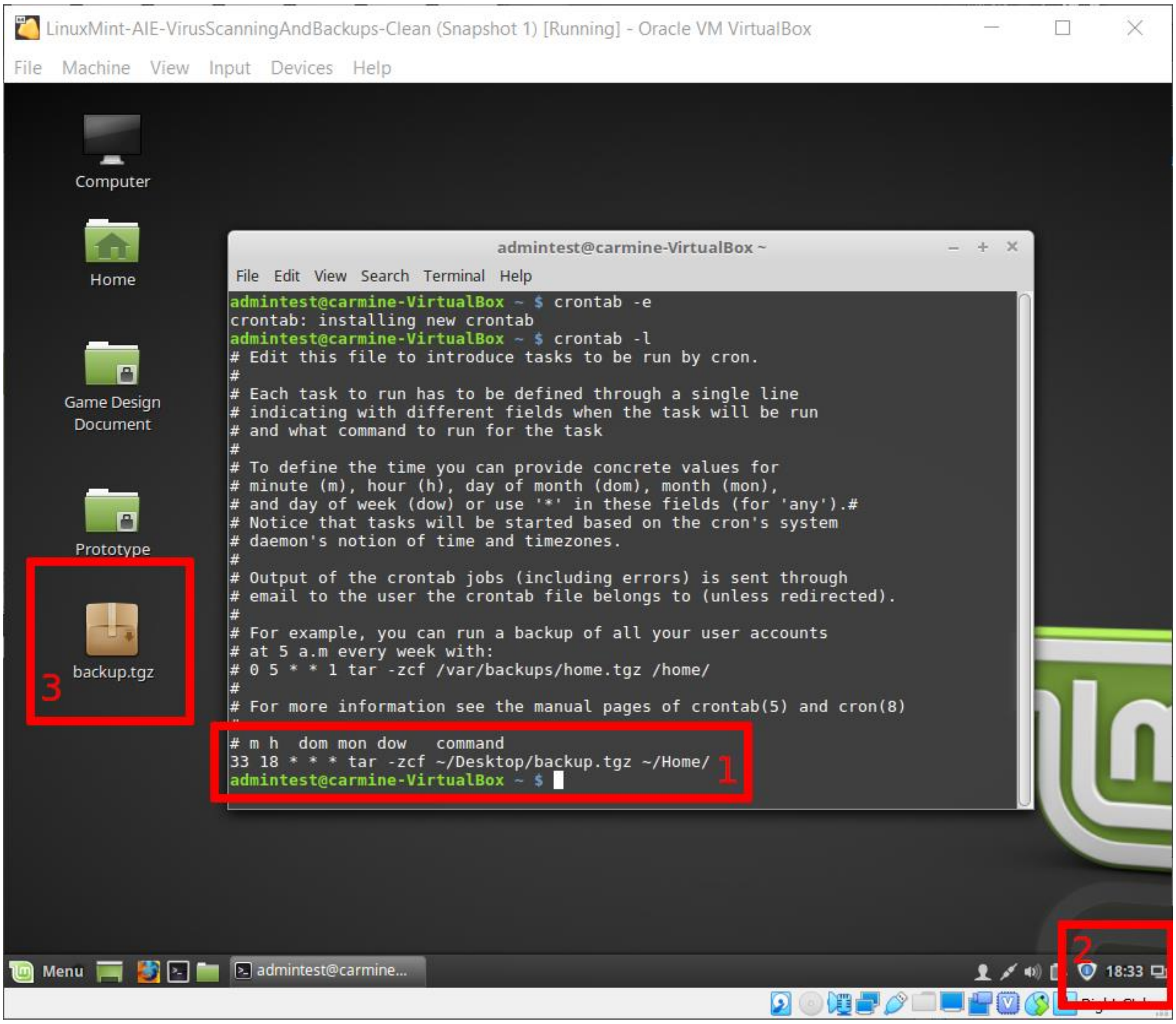
7. Press **Ctrl-O** keys to save file, then press **Enter** to select the default filename

8. Press **Ctrl-X** keys to close the editor.

9. After successfully creating your scheduled backup file, open the resulting .tgz file using the ***Archive Utility*** (a tool similar to 7-Zip or WinZip) by double-clicking the file.

   a. The following screenshot is an example **crontab** file with a scheduled command set for 18:33 (6:33pm). Notice that the **tar** command was automatically executed (at 18:33 in the VM) and created a **backup.tgz** file on the Desktop.

   b. *Note*: More information about the crontab utility can be found in the appendix.



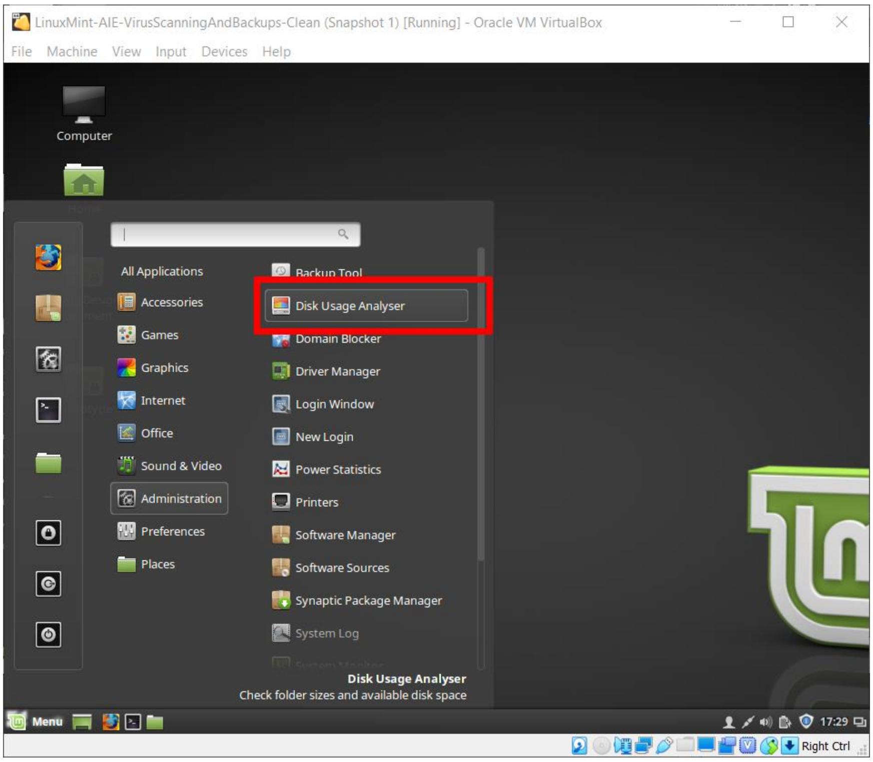
```
admintest@carmine-VirtualBox ~ $ crontab -e
crontab: installing new crontab
admintest@carmine-VirtualBox ~ $ crontab -l
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
33 18 * * * tar -zcf ~/Desktop/backup.tgz ~/Home/
admintest@carmine-VirtualBox ~ $
```

## Step 7: Maintaining the file system

We'll now use tools to examine the file system on the Virtual Machine. We'll then run applications to clean the drive.

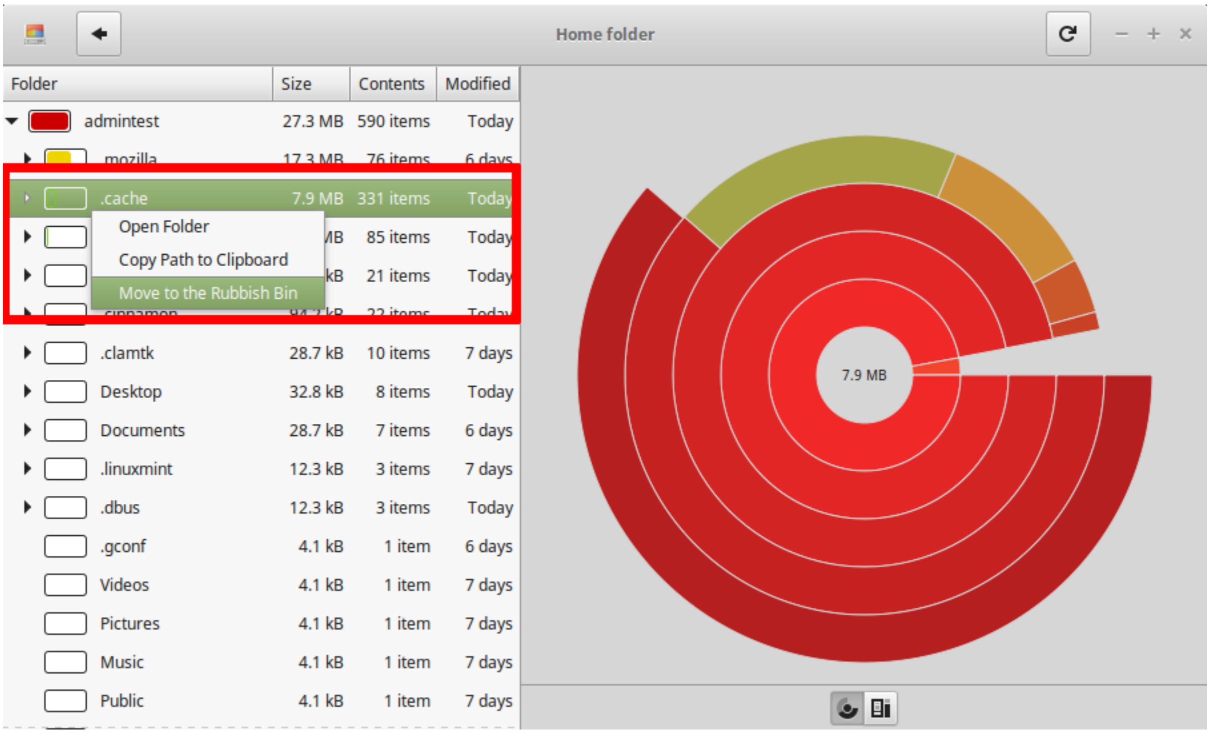
First up, let's examine our own home folder and clear the cache:

1. Open the **Disk Usage Analyser** tool



2. Select the /Home folder to examine its contents

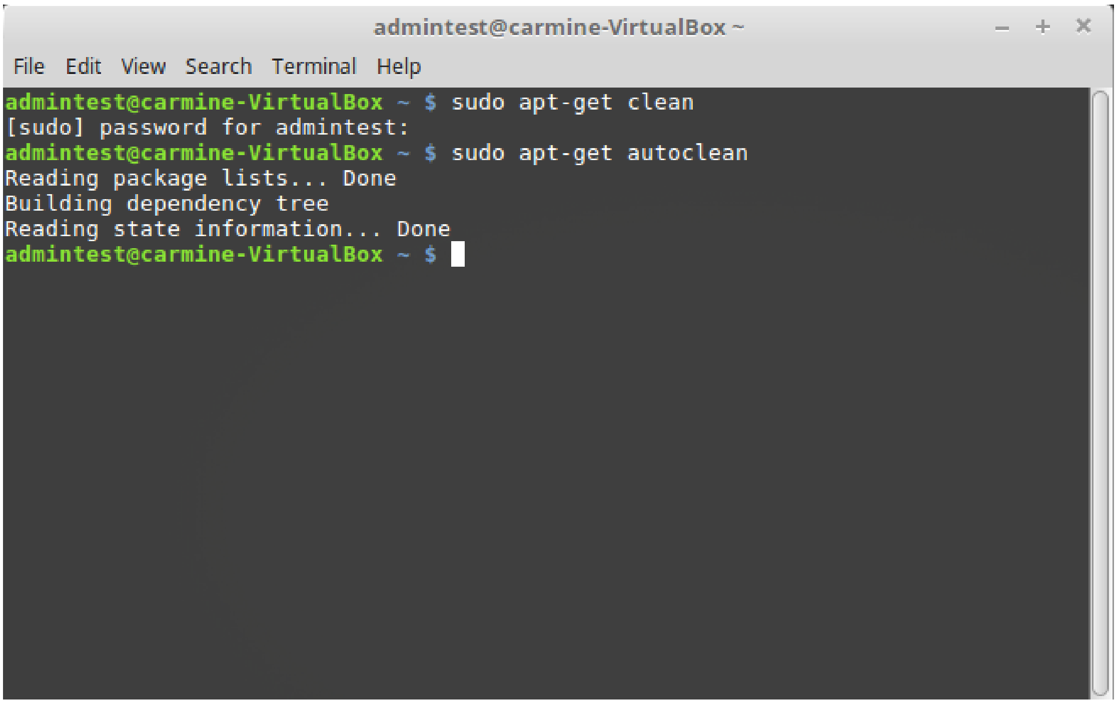
3. Right-click on the **.cache** folder and select "Move to Rubbish Bin" to clear the temporary cache folder



| Folder | Size | Contents | Modified |
|---|---|---|---|
| admintest | 27.3 MB | 590 items | Today |
| mozilla | 17.3 MB | 76 items | 6 days |
| .cache | 7.9 MB | 331 items | Today |
|  | IB | 85 items | Today |
|  | kB | 21 items | Today |
| cinnamon | 94.7 kB | 22 items | Today |
| .clamtk | 28.7 kB | 10 items | 7 days |
| Desktop | 32.8 kB | 8 items | Today |
| Documents | 28.7 kB | 7 items | 6 days |
| .linuxmint | 12.3 kB | 3 items | 7 days |
| .dbus | 12.3 kB | 3 items | Today |
| .gconf | 4.1 kB | 1 item | 6 days |
| Videos | 4.1 kB | 1 item | 7 days |
| Pictures | 4.1 kB | 1 item | 7 days |
| Music | 4.1 kB | 1 item | 7 days |
| Public | 4.1 kB | 1 item | 7 days |

Next up, we’ll use the apt-get to clear unused software packages via the CLI:

1. Open a Linux Terminal
2. Type the following commands to clean any unused packages:
   a. “`sudo apt-get clean`”
   b. “`sudo apt-get autoclean`”


admintest@carmine-VirtualBox ~
File Edit View Search Terminal Help

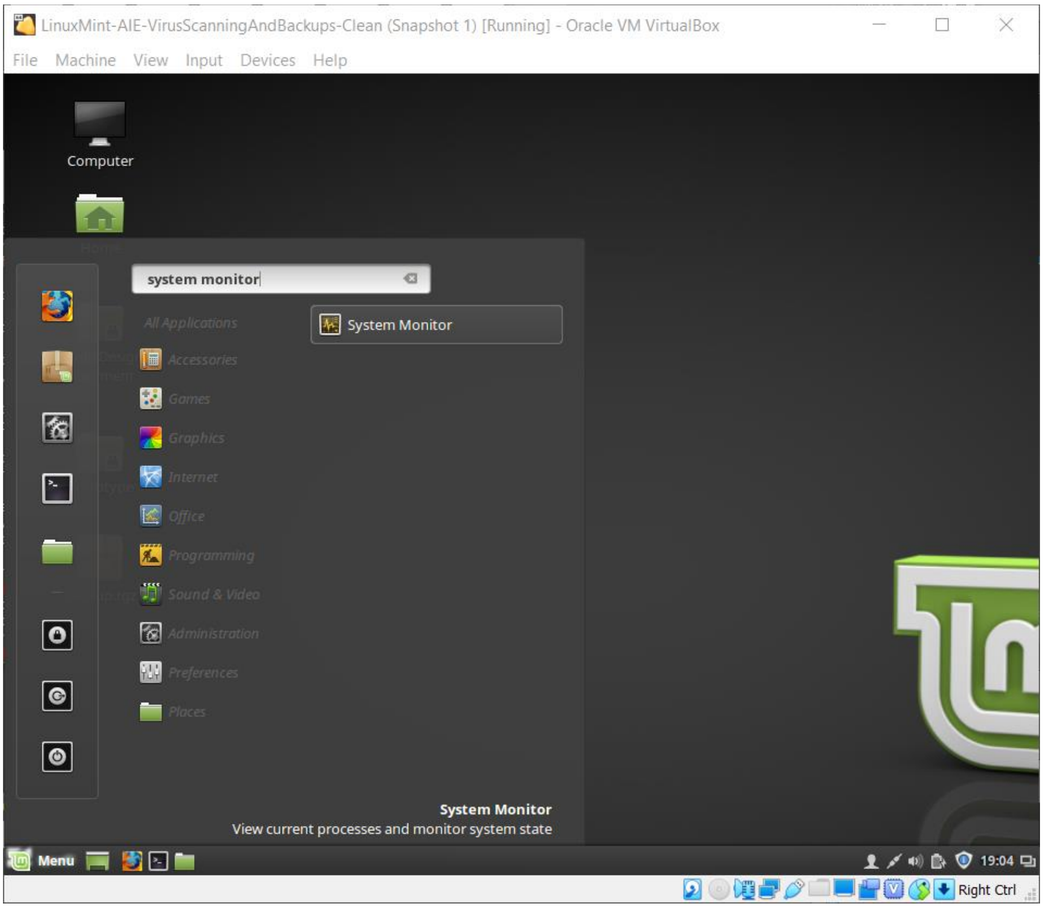
```
admintest@carmine-VirtualBox ~ $ sudo apt-get clean
[sudo] password for admintest:
admintest@carmine-VirtualBox ~ $ sudo apt-get autoclean
Reading package lists... Done
Building dependency tree
Reading state information... Done
admintest@carmine-VirtualBox ~ $
```

j

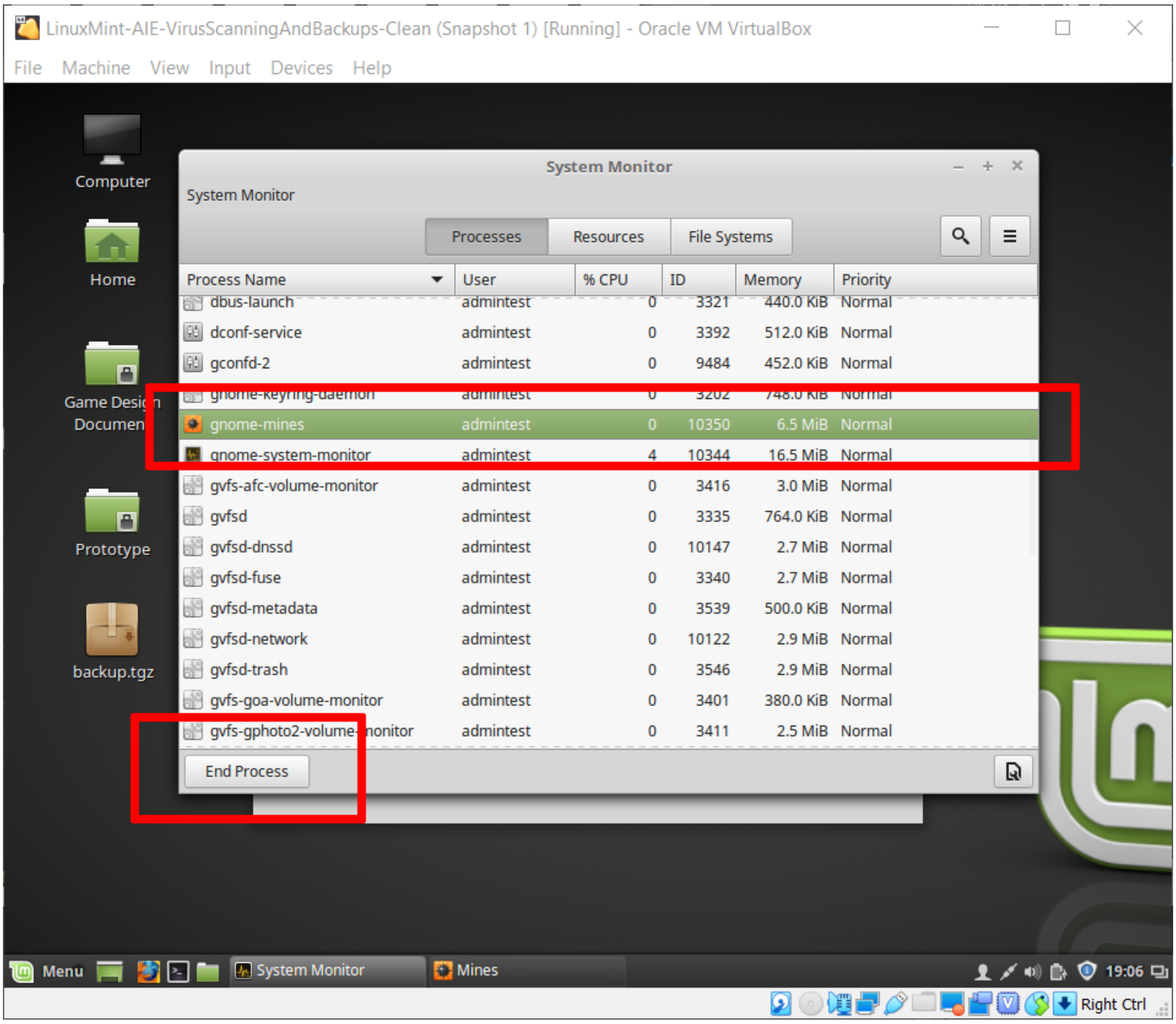
### Step 8: Killing rouge processes

It sounds pretty grim, but in essence we'll simply use the **System Monitor** tool to view currently running applications/services and stop one directly via the monitor interface. But we don't want to 'kill' an important OS service, so instead we'll start the Minesweeper application, and then kill it via the System Monitor. Follow these steps:

1. Open the **Mines** game via the **Main Menu**
2. Then open the **System Monitor** tool via the **Administration** menu



3. Under the **Processes** tab, you'll see all the applications & services currently running on the OS. While you're at it, explore the **Resources** and **File System** tabs too.
4. Select on the **Mines** application from the **Processes** tab
5. Then press the **End Process** button to kill the process

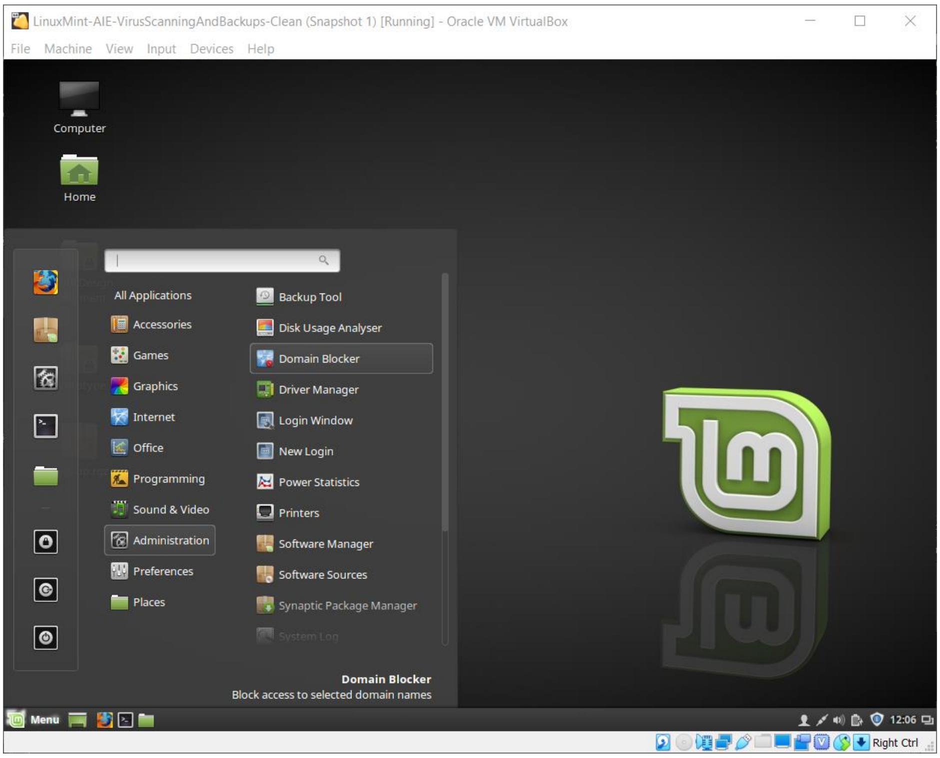


| Process Name | User | % CPU | ID | Memory | Priority |
|---|---|---|---|---|---|
| dbus-launch | admintest | 0 | 3321 | 440.0 KiB | Normal |
| dconf-service | admintest | 0 | 3392 | 512.0 KiB | Normal |
| gconfd-2 | admintest | 0 | 9484 | 452.0 KiB | Normal |
| gnome-keyring-daemon | admintest | 0 | 3202 | 748.0 KiB | Normal |
| gnome-mines | admintest | 0 | 10350 | 6.5 MiB | Normal |
| gnome-system-monitor | admintest | 4 | 10344 | 16.5 MiB | Normal |
| gvfs-afc-volume-monitor | admintest | 0 | 3416 | 3.0 MiB | Normal |
| gvfsd | admintest | 0 | 3335 | 764.0 KiB | Normal |
| gvfsd-dnssd | admintest | 0 | 10147 | 2.7 MiB | Normal |
| gvfsd-fuse | admintest | 0 | 3340 | 2.7 MiB | Normal |
| gvfsd-metadata | admintest | 0 | 3539 | 500.0 KiB | Normal |
| gvfsd-network | admintest | 0 | 10122 | 2.9 MiB | Normal |
| gvfsd-trash | admintest | 0 | 3546 | 2.9 MiB | Normal |
| gvfs-goa-volume-monitor | admintest | 0 | 3401 | 380.0 KiB | Normal |
| gvfs-gphoto2-volume-monitor | admintest | 0 | 3411 | 2.5 MiB | Normal |

## Step 9: Blocking Domains

In this section we'll use the *Domain blocker* to manage our network connection. It's used to restrict access to certain websites.

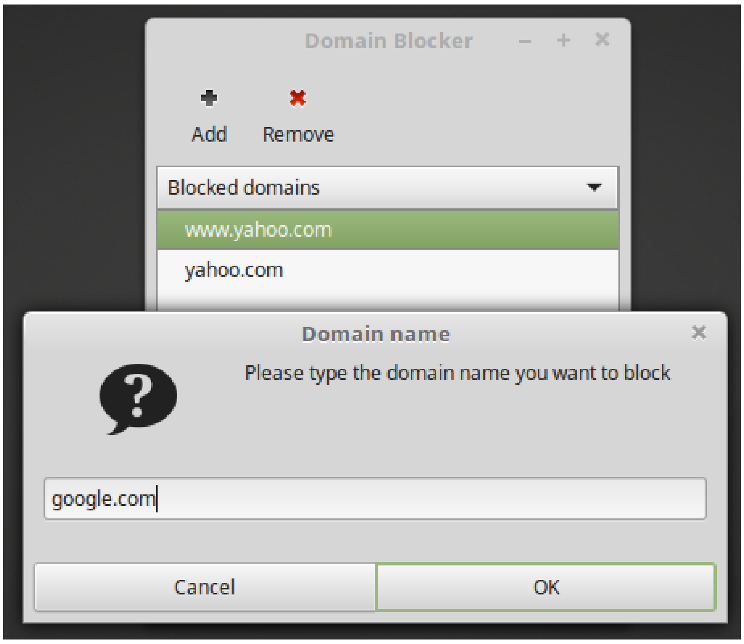
1. Open the **Domain Blocker** tool from the Administration menu



2. Click the **Add** button to block a new domain

3. Enter any URL you'd like to test (In the example we use *google.com* or *yahoo.com*)

    a. Notice that the tool will also add www domains automatically



Blocked domains

www.yahoo.com

yahoo.com

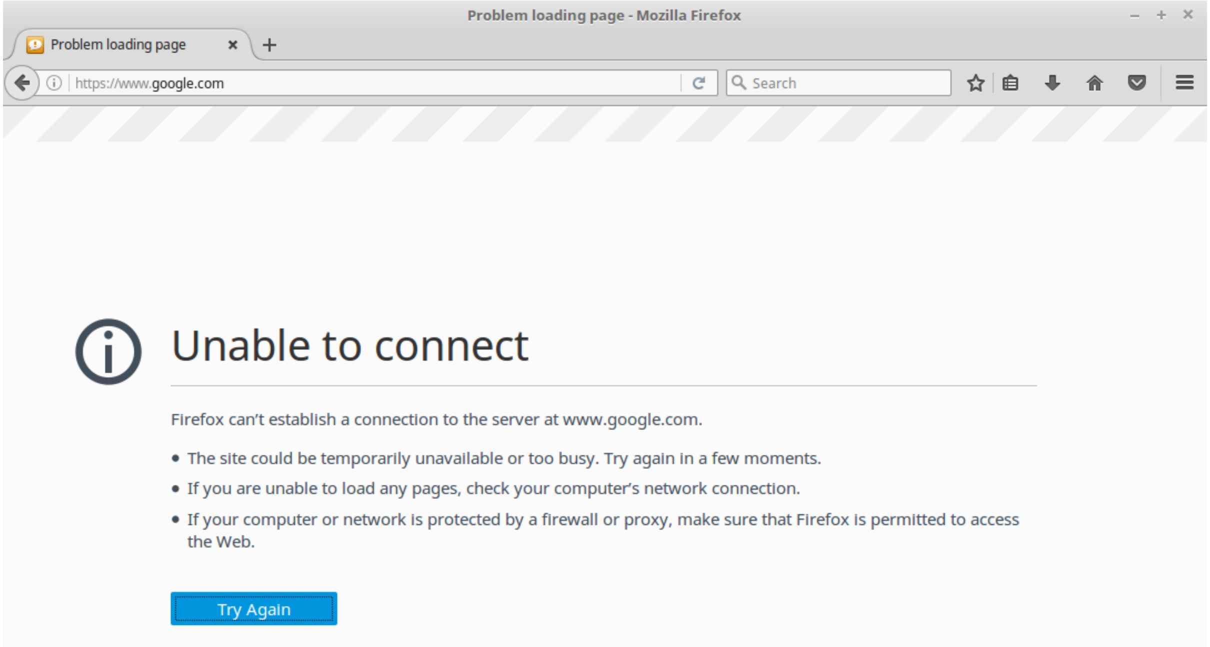
Domain name

Please type the domain name you want to block

google.com

4. Test the domain in the *admintest* account by opening a browser in the VM (e.g. Firefox) and navigating to the blocked domain. You'll receive an error like the one below:

    a. Note you may have to logout/login to activate the domain blocks



Unable to connect

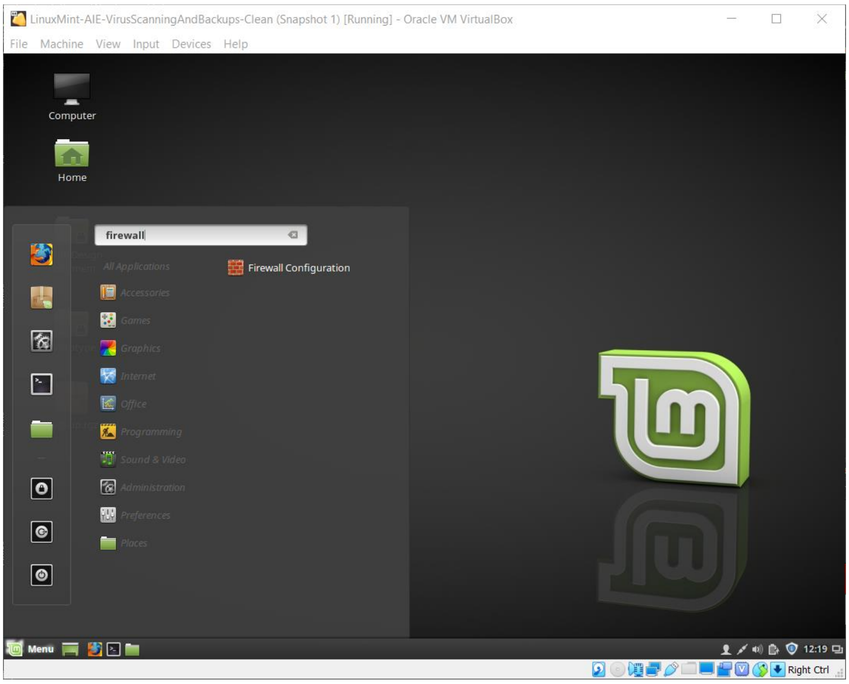
Firefox can't establish a connection to the server at www.google.com.

- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the Web.

## Step 10: Managing the Firewall

In this section we'll use the ***Firewall tool*** to help manage our network connection. The Firewall tool is used to block incoming connections at specific network port numbers. We're now going to shut off all incoming and outgoing network access via the Firewall.

1. Open the **Firewall** tool from the Administration menu



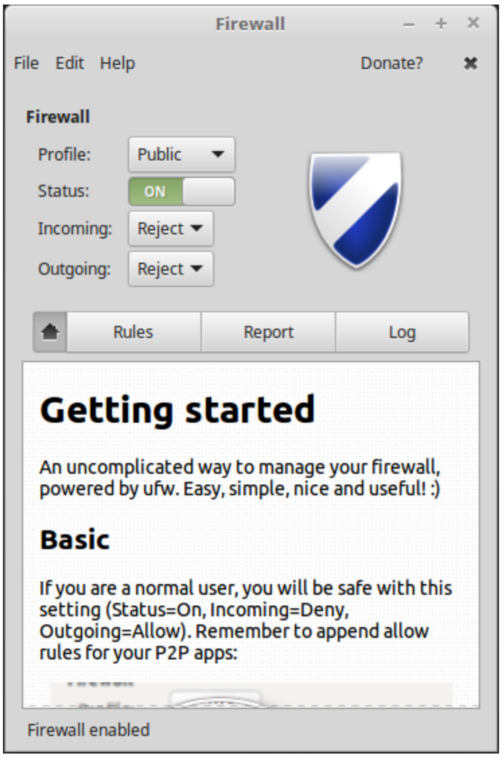
2. The Firewall tool uses profiles to store settings for different scenarios. Change the Firewall profile from **Home** to **Public** (by default it's set to *Home* allowing all connections).

3. **Reject** all connections. Your settings should look like this:

4. Now test the internet connection by browsing to any website in the VM.

5. Use the **Log** and **Report** menus to examine details about the recent connection attempts.

6. Feel free to explore the **Rules** menu to create your own specific Firewall connection rules. This is an advanced section, but recommended for anyone interested in how Firewalls control network access.



Firewall

Profile: Public

Status: ON

Incoming: Reject

Outgoing: Reject

**Getting started**

An uncomplicated way to manage your firewall, powered by ufw. Easy, simple, nice and useful! :)

**Basic**

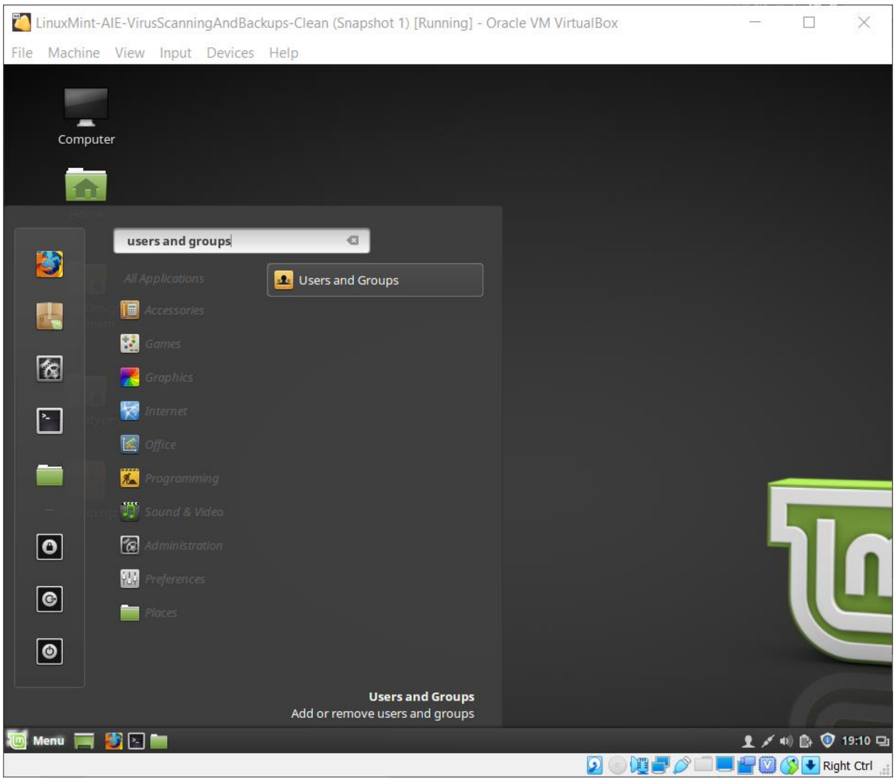
If you are a normal user, you will be safe with this setting (Status=On, Incoming=Deny, Outgoing=Allow). Remember to append allow rules for your P2P apps:

Firewall enabled

## Step 11: Creating User Accounts

The *admintest* account is configured as an administrator account, giving full access to all the system diagnostic and administration tools. This includes the ability for us to manage users. In this next step we'll create a user account, and then log back in as the new user.

1. Open the **Users and Groups** tool from the Administration menu

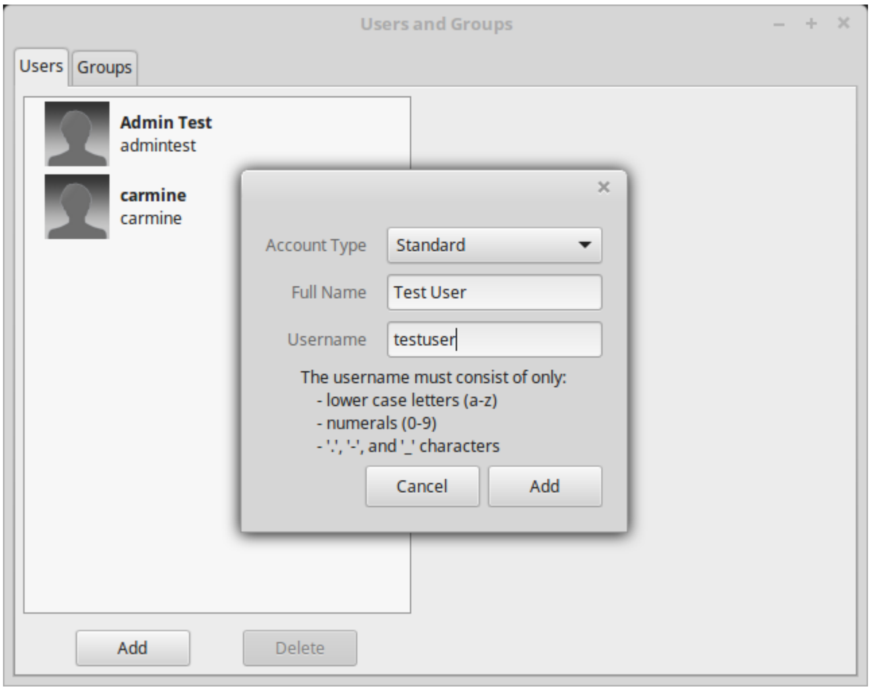


LinuxMint-AIE-VirusScanningAndBackups-Clean (Snapshot 1) [Running] - Oracle VM VirtualBox

users and groups

Users and Groups
Add or remove users and groups

2. Press the **Add** button, then enter details for your new user. The new account type is set to "Standard", thus the new user ***doesn't have*** *administrator privileges*.



Users and Groups

Admin Test
admintest

carmine
carmine

Account Type: Standard
Full Name: Test User
Username: testuser

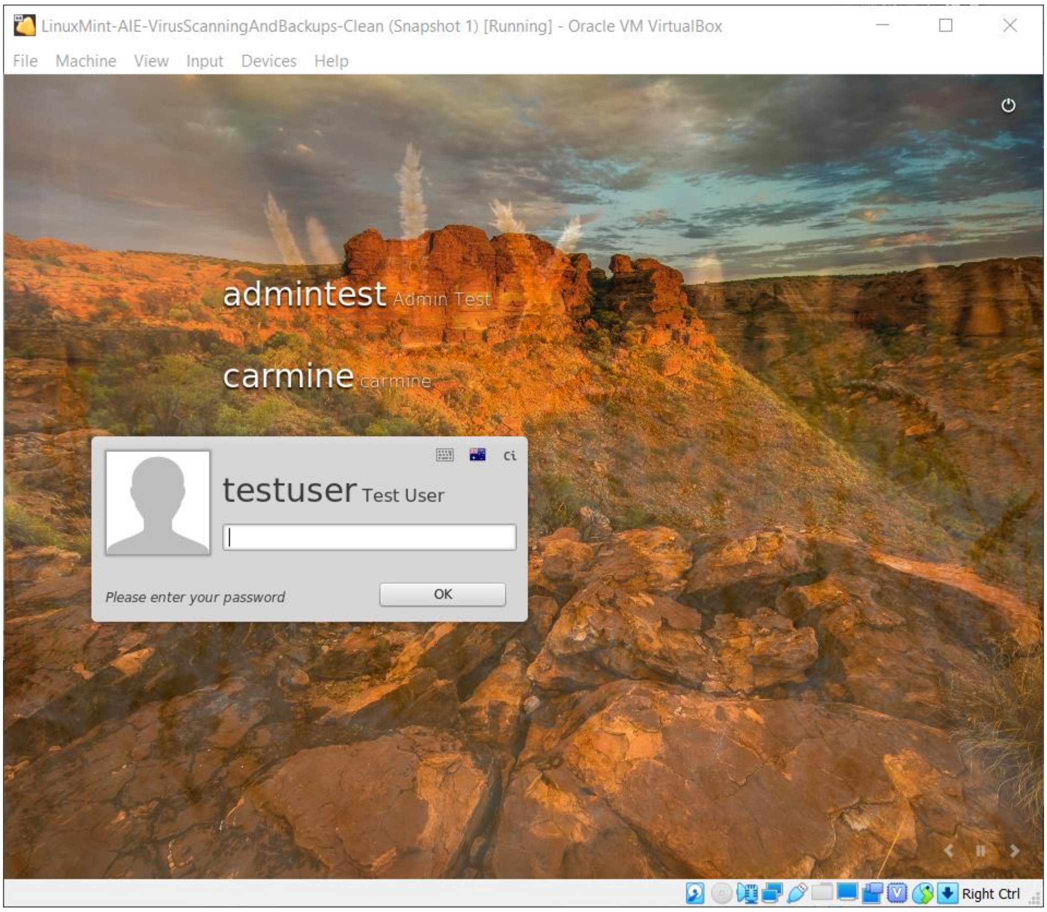
The username must consist of only:
- lower case letters (a-z)
- numerals (0-9)
- '.', '-', and '_' characters

3. Select the new user, click the "**No password set**" button and change their default password.

4. Logout the system

5. The login using the new user's details…

6. Notice this new user has their own *Home* folder and their *Documents* folder is currently empty (unlike the admintest folder)


LinuxMint-AIE-VirusScanningAndBackups-Clean (Snapshot 1) [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
admintest Admin Test
carmine carmine
testuser Test User
Please enter your password
OK
Right Ctrl

7. While logged in as *testuser* try using browser inside the VM (e.g. Firefox) and browsing to the website that you previously blocked via the admintest account (using the Domain Blocker).

   a. You'll notice that the *domain* is now been *blocked* for all users

Let's now simulate a user account that requires immediate termination…

1. Log back in to the OS as *admintest*

2. Open the Users and Groups tool

3. Remove the ***testuser*** account from the system…

## Appendices

### Appendix 1: Crontab examples

The **crontab** command opens the *cron table* for editing. The *cron table* is the list of tasks scheduled to run at regular time intervals on the system.

More information about the crontab and cron commands are available in the Linux terminal using the “`man crontab`” and “`man cron`” commands, and from webpage like the following:

https://www.computerhope.com/unix/ucrontab.htm

https://www.howtogeek.com/101288/how-to-schedule-tasks-on-linux-an-introduction-to-crontab-files/

## Running crontab

```
crontab -e
```

Edit your crontab.

```
crontab -l
```

Display ("list") the contents of your crontab.

```
crontab -r
```

Remove your crontab, effectively un-scheduling all crontab jobs.

```
sudo crontab -u charles -e
```

Edit the crontab of the user named **charles**. The **-u** option requires administrator privileges, so the command is executed using **sudo**.

```
sudo crontab -l -u jeff
```

View the crontab of user **jeff**.

```
sudo crontab -r -u sandy
```

Remove the crontab of user **sandy**.

## Crontab entries

The following are examples of entries which could be included in a crontab.

Run the shell script **/home/melissa/backup.sh** on January 2 at 6:15 A.M:

```
15 6 2 1 * /home/melissa/backup.sh
```

Days and months can be listed by name (**Monday**) or abbreviation (**Jan**). Zeroes at the beginning of a number are valid, which can help you make multiple entries line up visually.

For instance, the next example runs the same script as above, at 12:01 AM, every Monday in January:

```
01 00 * Jan Monday /home/melissa/backup.sh
```

Run **/home/carl/hourly-archive.sh** every hour, on the hour, from 9 A.M. (**09**:**00**) through 6 P.M. (**18**:**00**), every day:

```
00 09-18 * * * /home/carl/hourly-archive.sh
```

Run **/home/wendy/script.sh** every Monday, at 9 A.M. and 6 P.M:

```
0 9,18 * * Mon /home/wendy/script.sh
```

Run **/usr/local/bin/backup** at 10:30 P.M., every weekday:

```
30 22 * * Mon,Tue,Wed,Thu,Fri /usr/local/bin/backup
```