IT Loaner Laptop Install

Overview

\*While building the system we will set ALL users to administrator, the reason for this is because we need to log in to each individual account and modify things that only an account with ‘administrator’ rights can. The steps below show how to make the other user’s accounts, for now set the ‘account type’ to ‘administrator’. Before we actually give the pcs to the users we will go back in and change the ‘citylocaluser’ account to ‘standard’.

To succeed at linux you must learn how the system grants permissions and how the file system works/the nature of the file system etc. Basic knowledge on that will help dramatically with these install documents.

**The administrator account should be the first account you make, all of the instructions below however should be executed while logged into the ‘citylocaluser’ account. The reason for this is because that is the account the users will be using, the administrator does not need what the citylocaluser will need ie. Shrew icon on desktop.**

**COMPUTER INFO**

*USERS*

(Sudo User/Administrator)

*Username:* administrator

*Password:* m@r!e

(Regular User)

*Username:* citylocaluser

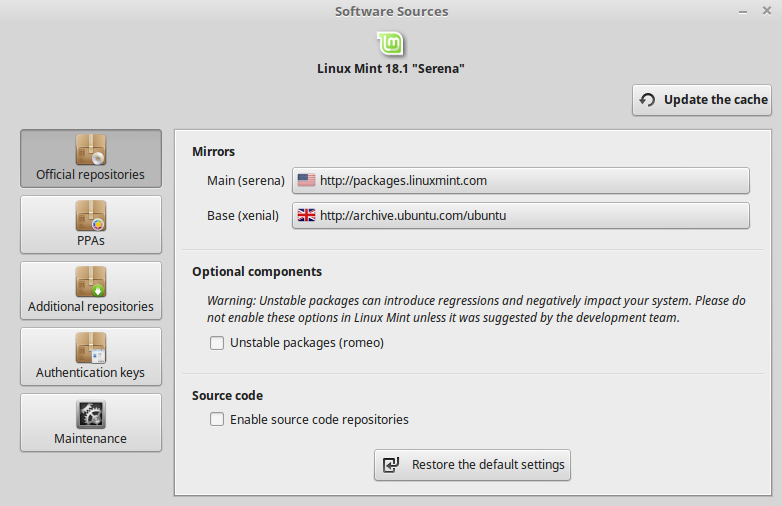
*Password:* cityuser

**NETWORK INFO**

*Connection Type:* Wireless

*Network Mirror Used:* [http://packages.linuxmint.com](http://packages.linuxmint.com/)

Click Menu>Administration>Software Sources



You can change the network mirror to any one, packages.linuxmint.com is worldwide and commonly used but you can change the mirror to a Canadian repo for faster speeds. I recommend the one set here however.

**Remove Unrequired Applications**

-sudo –s

-apt-get remove vino

-apt-get purge tomboy

-apt-get purge gimp

-apt-get purge pix

-apt-get purge hexchat

-apt-get purge pidgin

-apt-get purge thunderbird

-apt-get purge transmission-gtk

-apt-get purge brasero

-apt-get purge rhythmbox

-apt-get autoremove

**Upgrade OS**

-sudo apt-get update && apt-get upgrde && apt-get dist-upgrade

-reboot

**Set Up VNC**

Log into ‘administrator’ account, open a terminal and type:

-sudo apt-get install x11vnc

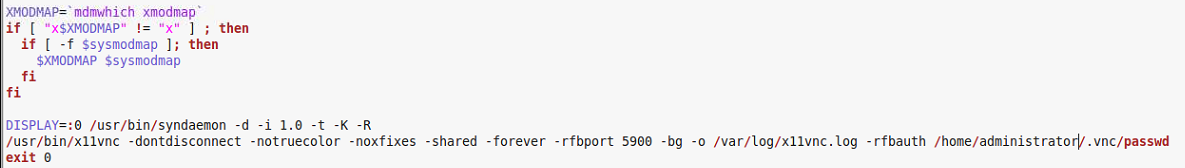
-open a new terminal and type:

i. x11vnc –storepasswd

ii. #set password to default VNC password ‘d@!!@s’

-sudo apt-get install gedit

-sudo gedit /etc/mdm/Init/Default

f) #Add the following:

|  |
| --- |
| /usr/bin/x11vnc -dontdisconnect -notruecolor -noxfixes -shared -forever -rfbport 5900 -bg -o /var/log/x11vnc.log -rfbauth /home/citylocaluser/.vnc/passwd |

**\*stores password to /home/administrator/.vnc/passwd\***

Log into ‘citylocaluser’ account:

\*right click on the desktop and make a file on the desktop called ‘VNC’\*

In terminal type:

-cd /home/citylocaluser/Desktop/VNC

-gedit Run-VNC.service

(Copy and paste the following yellow text into the new document)

/usr/bin/x11vnc –auth guess –forever –loop –noxdamage –repeat –rfbauth /home/administrator/.vnc/passwd –rfbport 5901 -shared

Save the file and close it and then in terminal type:

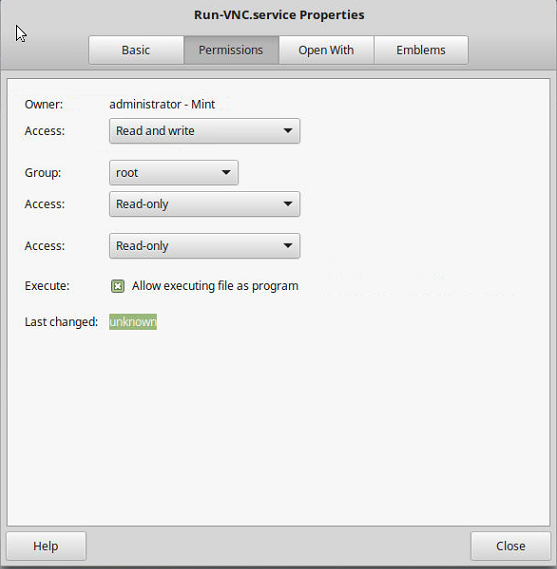
-systemctl daemon-reload

-sudo chown –R administrator /home/citylocaluser/Desktop/VNC/

-sudo chmod 755 /home/citylocaluser/Desktop/VNC/Run-VNC.service

-sudo chmod 777 /home/administrator/.vnc/passwd

\*right click on the file and click properties and change to allow executing as program\*



**Now we must enable the service to boot upon startup time**

Open terminal and type:

-sudo –s

-gedit /lib/systemd/system/x11vnc.service

\*Write the following in the new file\*

[Unit]

Description=Start x11vnc at startup.

After=multi-user.target

[Service]

Type=simple

ExecStart=/usr/bin/x11vnc -auth guess -forever -loop -noxdamage -repeat -rfbauth /home/administrator/.vnc/passwd -rfbport 5900 -shared

[Install]

WantedBy=multi-user.target

\*Save the file and in terminal type:\*

-systemctl daemon-reload

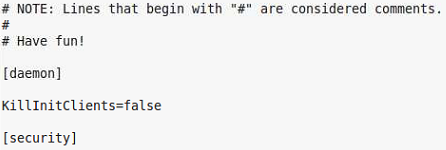
-systemctl enable x11vnc.service

This will start VNC in the background upon startup. **The only step you will need to repeat is where you actually create the ‘VNC’ folder on the desktop. Everything after the step ‘Enable service upon boot’ only needs to be done once. Just be sure to create the ‘VNC’ file on every user’s desktop and give it the proper permissions. Just repeat the steps from ‘\*stores password to\*’. We are making this folder so that if for some reason our automatic vnc launch fails, the user can launch vnc by running our script. This prevents confusion as running vnc any other way will need specifications and can confuse the user. The goal is to make the user do as little as possible.**

**Stop VNC from closing when logging into an account:**

-sudo gedit /etc/mdm/mdm.conf

-under [Daemon] section add: KillInitClients=false



-Restart Machine

**-Install VPN Shrew**

-download <https://www.shrew.net/download/ike>

-Extract Contents of tar file, right click and click ‘open with archive manager’

Log into ‘citylocaluser’ and open a terminal and type:

-sudo apt-get upgrade libqtgui4 && libqt4-dev && libedit-dev && libssl-dev

-sudo apt-get install cmake libqtcore4 libqt4-dev libqtgui4 libedit-dev libssl-dev checkinstall

-sudo apt-get install build-essential

-cd Downloads/ike

-Run:

i. cmake -DCMAKE\_INSTALL\_PREFIX=/usr -DQTGUI=YES -DETCDIR=/etc -DNATT=YES

ii. make

iii. sudo make install

iv. cd /etc/

v. sudo mv iked.conf.sample iked.conf

vi. cd

vii. sudo apt-get install ike-qtgui

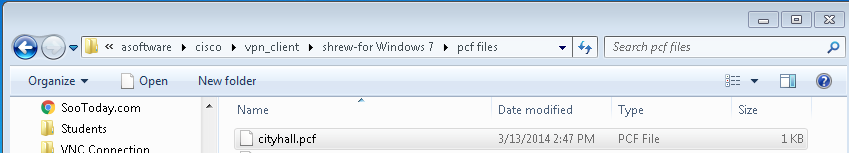
To ensure that Shrew was installed correctly enter:

i. sudo iked

ii. qikea &

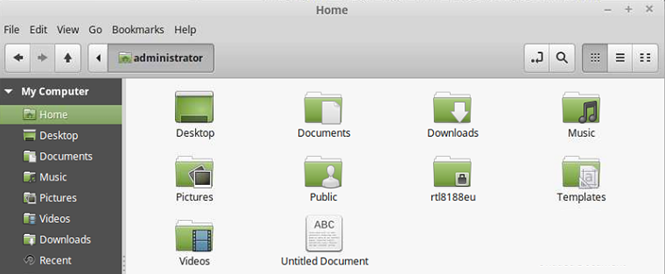
Copy cityhall.pcf file and import into shrew as user

[\\cityit\software$\asoftware\cisco\vpn\_client\shrew-for Windows 7\pcf files\](file:///\\cityit\software$\asoftware\cisco\vpn_client\shrew-for%20Windows%207\pcf%20files\)



**Setup Users Shrew Desktop Icon**

Create an empty document in home folder.



Enter the following:

|  |
| --- |
| #!/bin/bash  # Run Shrew  sudo iked  qikea & |

Save the file as START-SHREW



Allow users to execute the file:

i. sudo chmod 755 START-SHREW

Create a new empty document on the desktop name it Shrew.desktop



Edit the file using gedit

Enter the following:

|  |
| --- |
| [Desktop Entry]  Name=Start Shrew  Comment=Start iked and qikea  Exec=/home/\*YOUR\_USER’S\_USERNAME\*/START-SHREW  Icon=applications-internet  Terminal=false  Type=Application  For example, for the administrator it would look like this:  [Desktop Entry]  Name=Start Shrew  Comment=Start iked and qikea  Exec=/home/administrator/START-SHREW  Icon=applications-internet  Terminal=false  Type=Application  For the citylocaluser it would look like this:  [Desktop Entry]  Name=Start Shrew  Comment=Start iked and qikea  Exec=/home/citylocaluser/START-SHREW  Icon=applications-internet  Terminal=false  Type=Application |

i. Looks like:



**Give permission to user so that they can run iked and qikea**

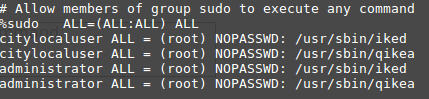
-sudo visudo

-Find the lines matching:

|  |
| --- |
| # Allow members of group sudo to execute any command  %sudo ALL = (ALL:ALL) ALL |

-Underneath Enter the following:

|  |
| --- |
| # this is the format to use, enter username you are using <username> ALL = (root) NOPASSWD: /usr/sbin/qikea  # assuming you are setting up an administrator account along with a local account make sure  # to enter the lines with the respective account names. In this case we did for the #administrator and the citylocaluser  citylocaluser ALL = (root) NOPASSWD: /usr/sbin/iked  citylocaluser ALL = (root) NOPASSWD: /usr/sbin/qikea  administrator ALL = (root) NOPASSWD: /usr/sbin/iked  administrator ALL = (root) NOPASSWD: /usr/sbin/qikea |

****

**Allow user to install updates:**

-sudo chattr -i /usr/share/polkit-1/actions/com.ubuntu.pkexec.synaptic.policy

-sudo gedit

-hit file and click open

-click other locations > computer > usr > share > polkit-1 > actions > com.ubuntu.pkexec.synaptic.policy

i. change: <allow\_active>auth\_admin</allow\_active>

ii. to: <allow\_active>yes</allow\_active>

-sudo chattr +i /usr/share/polkit-1/actions/com.ubuntu.pkexec.synaptic.policy

**7) Remote Desktop**

-sudo apt-get install remmina

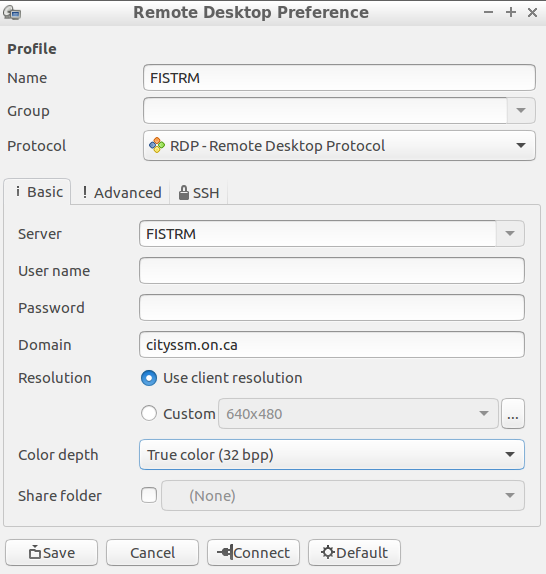
-sudo apt-get install remmina-plugin-rdp

-Click menu and type in remmina, right click on remmina and click ‘add to desktop’

-Right click on remmina and and click ‘open with gedit’

-Find ‘Icon=remmina’ and change it to ‘Icon=mdmflexiserver’

-Change the name of the application from ‘Remmina’ to ‘Remote Desktop’ by right clicking and opening the properties



-Check to make sure you can connect

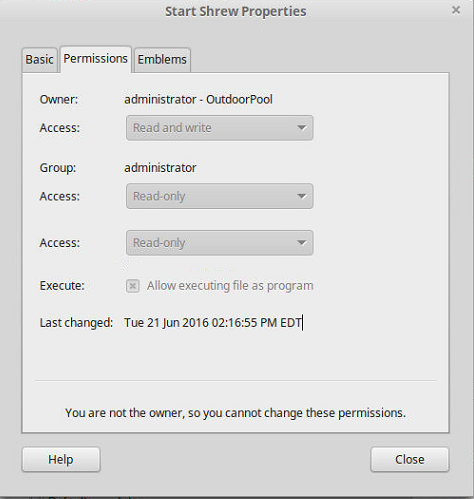
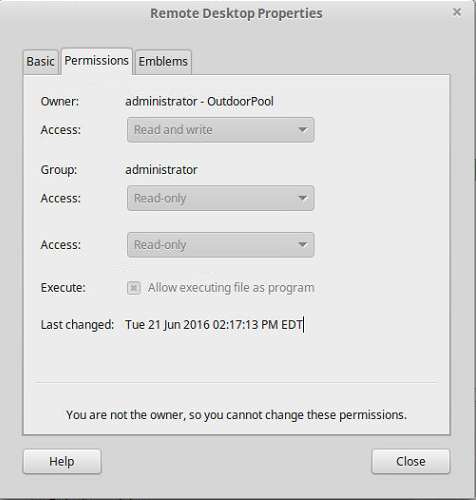
**8) Copy the Remote Desktop Icon and Start Shrew Icon into the user’s desktop folder as root.**

-chmod 751 /home/administrator

-Ensures that user cannot read or write files to administrator folder but can still execute START-SHREW

-Remove terminal and account details from /usr/share/applications make a shortcut on the administrator desktop so its easy to access.

-Remove terminal and account details from the taskbar.



**Remove computer and home icons from desktop**

gsettings set org.nemo.desktop computer-icon-visible false

gsettings set org.nemo.desktop home-icon-visible false

**If TightVNC cannot connect these settings may be present and conflicting**

-gsettings set org.gnome.desktop.remote-access notify-on-connect false

-gsettings set org.gnome.desktop.remote-access prompt-enabled false

-gsettings set org.gnome.desktop.remote-access require-encryption false

**To allow users to be able to open network settings without being sudo**

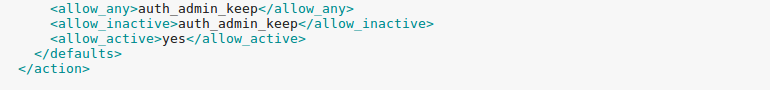
Open terminal and type

-sudo gedit /usr/share/polkit-1/actions/org.freedesktop.NetworkManager.policy

-navigate to the line in the file that says:

F:\Screenshot from 2017-07-12 08-39-15.png-Change <allow\_active>auth\_admin\_keep</allow\_active> to the following: <allow\_active>yes </allow\_active>

Should look like below screenshot



**Create a script that will fix the network issue if it ever persists.**

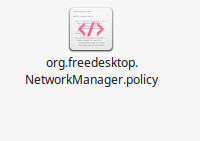
-click menu > file system > usr > share > polkit-1 > actions > org.freedesktop.NetworkManager.policy

-double click to open and highlight everything in the file and copy it

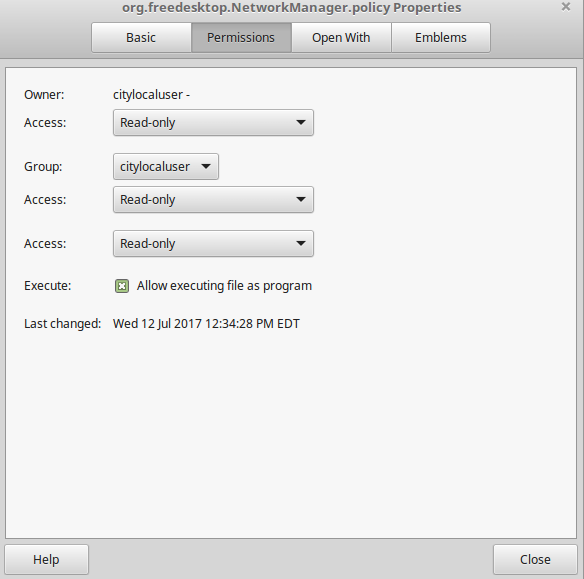
-open up gedit and paste all of the code in the file

-navigate to line 888, like in step ‘11’ we are going to make sure it says ‘<allow\_active>yes</allow\_active>’ as opposed to ‘<allow\_active>auth\_admin\_keep</allow\_inactive>’. Since you copied it right after you already fixed the file in step 11 it should already have the correction, refer to above screenshot again.

-make a folder on the desktop called ‘Solution’ and save this new document in it as ‘org.freedesktop.NetworkManager.policy’. Should look like the screenshot below:



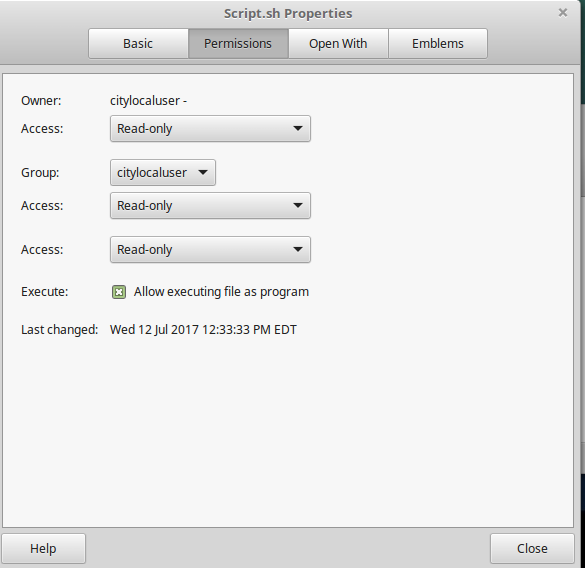
-Right click on the file and click ‘properties’. Then click ‘permissions’ and check off ‘Allow executing file as program’.



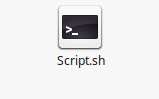
-open up gedit and type the following (without the single quotes):

‘cp –f /home/citylocaluser/Desktop/Solution/org.freedesktop.NetworkManager.policy /usr/share/polkit-1/actions/org.freedesktop.NetworkManager.policy’

-Save the file in the folder ‘Solution’ on the desktop as ‘Script.sh’. Now right click on the file and click ‘properties’. Then click the ‘permissions’ and click ‘Allow executing file as program’



Should look like the screenshot below:



Anytime the user runs the script you just made it will locate the original file in the file system that Linux is running and replace it with the new ‘org.freedesktop.NetworkManager.policy’ file that you created with the proper correction in it. Now we are going to grant read and execute permissions to the script and file to prevent the user from modifying it but allow for them to read and execute. Open terminal and type:

-sudo chown administrator /home/citylocaluser/Solution/org.freedesktop.NetworkManager.policy

-sudo chown administrator /home/citylocaluser/Solution/Script.sh

-changes the administrator as the owner of both files in the folder ‘Solution’

-sudo chown administrator /usr/share/polkit-1/actions/org.freedesktop.NetworkManager.policy

-changes the administrator as the owner of the file that Linux is reading

-sudo chmod 755 /home/citylocaluser/Solution/org.freedesktop.NetworkManager.policy

-sudo chmod 755 /home/citylocaluser/Solution/Script.sh

-these commands will allow the user to read and execute but not modify either of the scripts

-sudo chmod 765 /usr/share/polkit-1/actions/org.freedesktop.NetworkManager.policy

This command allows the user to be able to change the original file if need be, it allows the script you made to run and successfully write to the file, considering the administrator owns the file but you have read and write privileges

Now we want to prevent the user from making further changes to the desktop so execute these two commands in terminal:

-sudo chown –R administrator /home/citylocaluser/Desktop

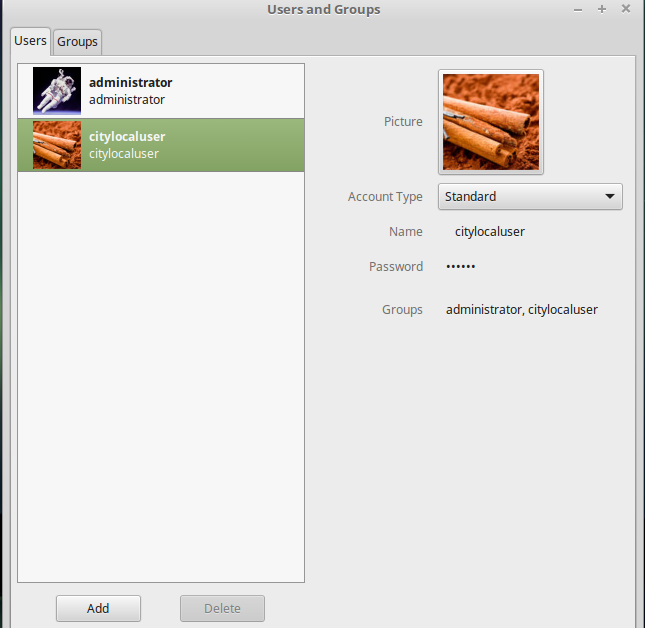
-sudo chmod 755 /home/citylocaluser/Desktop/

Both commands change the owner of the desktop to the administrator and only allows for the user to read and execute what resides on the desktop, they cannot create anything on the desktop or modify

**Change ‘citylocaluser’ to a standard account with no sudo rights, do when you are absolutely done making the PC and are ready to give to the user. Log in as the administrator and do the following:**

-navigate to menu > administration > users and groups

-click citylocaluser and change ‘account type’ to ‘standard’



-exit

The user now is considered a standard user, they cannot even execute the ‘sudo’ command, if you need to work on the pc and edit things within the citylocaluser account, log into the administrator account and navigate to this same location and change citylocaluser from ‘standard’ back to ‘administrator’. When you finish your work and are prepared to give the PC back to the user be sure to change the citylocaluser account to ‘standard’ and not leave it as administrator