

Parallely Sharing a Linux Computer

Rohit Rawat
Linux User Group
UT Arlington

Need for remote access

- Accessing your own machine remotely
- Providing technical support
- Sharing premium computing resources for running computations or builds
- Saving costs in setting up teaching labs

Multiuser aspects of Linux

- The latest kernel can support over 9000 users at a time
- Libraries share memory footprints
- Can be used with
 - Shell – install ssh server
 - GUI desktop interface – explained in this guide

Security Warning

- Enabling remote access to a machine connected to the internet is a security risk
- Most of the methods described here have no encryption and weak passwords
- Either use them through SSH tunnelling or use TeamViewer
- Make sure your router is blocking external access to the ports used here

Part 1 - VNC

VNC

- Virtual Network Computing
- Client-server based
 - VNC Server – runs on the machine being controlled
 - VNC Client – controls the remote
- VNC uses Remote Frame Buffer (RFB) protocol
 - Transmits graphics drawing commands from server to client
 - Can be slow as images are transmitted
 - Speed is improved by mirror drivers which detect and transmit only the changes

VNC Desktop Sharing

- When the server has its own “real” display
- Server's display and local display will be identical, user must already be logged in
- Server side:
 - Ubuntu comes installed with “Desktop Sharing” (vino) application – not turned on by default
 - 'x11vnc' – is a universal alternative
- Client side:
 - Install 'tightvncviewer' (available for Linux/Win/Mac)
 - Multiple clients can view/control the same screen

Demo

- Server:
 - Ubuntu – run 'Desktop Sharing', enable Sharing/Control options, set a password, enable/disable confirmation
 - Others – install 'x11vncviewer' package
 - Online guides available
 - Can be set up to autostart
- Client:
 - Install xtightvncviewer:
 - `sudo apt-get install xtightvncviewer`
 - Run: `vncviewer server-address`

Headless VNC/Multiple sessions

- Remote server does not need a physical display connected
- Launching 'vncserver' creates a new virtual desktop/display in the background
- Multiple desktops can be launched in parallel
 - Keep in mind the server speed and memory
- Sessions continue to run even after logout
 - Each user can log in, start their VNC session, leave
 - Can be started automatically by following these instructions:
 - <http://superuser.com/questions/147109/automatically-start-vnc-server-on-startup/421039#421039>

Launching VNC sessions

- Server:
 - Install tightvncserver:
 - `sudo apt-get install tightvncserver`
 - Launch 'vncserver':
 - `vncserver -geometry 800x600`
 - Specifying geometry is optional, but keep it less than the size of the client's display so it fits
 - On its first run it will ask you to set your password
 - Make a note of the display number:
 - New 'X' desktop is server-name:**1**
 - To shut down the server:
 - `vncserver -kill :1`

VNC Display Numbers

- VNC server listens on TCP port numbers 5900+
- The first session launched on the server is given display number :0, and the port is 5900
- The second session is at display :1, port 5901 ..
- Each user gets a unique display number
- You can only connect to a session/port if you know its password
- You can only kill a VNC server if you launched it

On the client

- Make sure xtightvncviewer is installed
- Run: vncviewer server-address:port
 - vncviewer 192.160.10.3:1
 - Enter the password
- Windows users can use TightVNC or UltraVNC
 - Only the 'viewer' executables are needed, so admin privileges not necessary

VNC Security Concerns

- Passwords are only 8 characters long
 - Not difficult to crack
- Someone may listen in on your connection and intercept keystrokes etc
- VNC spinoffs exist that support encryption
- VNC server can be restricted to localhost and ports 5900+ can be tunnelled through SSH
- Still okay to use in a private environment

Alternatives for single user remote access

- TeamViewer
 - Free and available for Linux
 - Fully functional and fast
- Other services like LogMeIn, GoToMeeting etc. do not support Linux at this time.

Part 2 – X Server

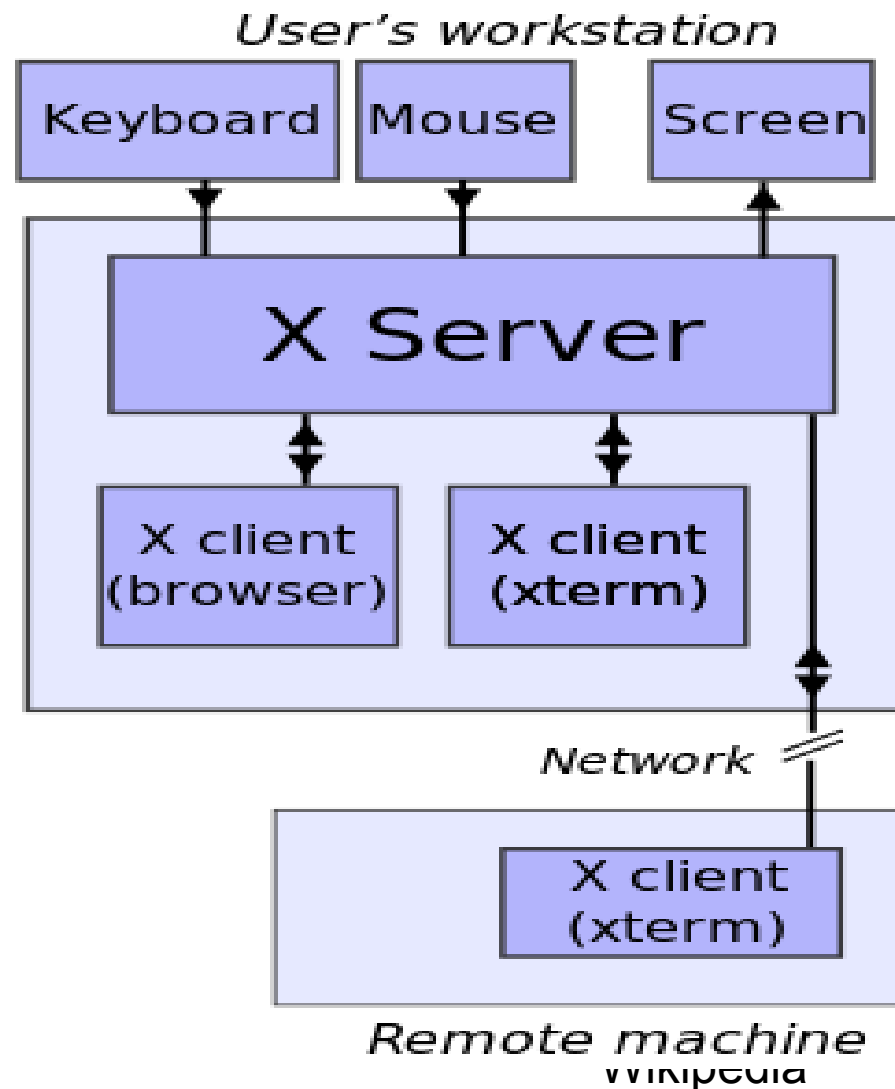
X Windows Protocol

- Current Linux/Unix display are based on X
 - Developed by X Consortium/MIT in 1984
- X-server manages the windows on the machine
- X-clients can connect to server and perform drawing operations
- Connects over TCP ports 6000+
- First display is :0, port 6000, second is :1, port 6001 etc.

X Protocol basics

- Client send “Requests” - draw a line, fill a window etc.
- Server sends back “Replies” or “Errors” to these requests
- Client can send keystroke or mouse “Events” to the server
- Server can send back “Events” like window resized, window exposed etc.

Typical usage scenario



Enabling X forwarding in SSH

- In the ssh client, add the -X switch
 - `ssh -X user@servername`
- Windows: In puTTY configuration, Connection->SSH->X11, enable X11 forwarding
- The ssh server will change the \$DISPLAY variable for all programs launched through it and forward them to the ssh client
- The program still gets executed on the server, just the display commands get forwarded

Interpreting forwarded X11 commands

- The client needs to be able to interpret the X11 commands
- In Linux, your existing X-server will do this
- In Windows, you can install an X-server implementation like X-Ming

Comparing to VNC

- With X11 forwarding, you can launch one application at a time, like gedit or xterm, or you can launch the full desktop manager. With VNC you always get the full desktop.
- X11 transmits high level window elements/widgets, which VNC operates on a lower level. So X11 is more bandwidth efficient compared to VNC.
- X11 forwarded applications terminate when the SSH connection drops. You must use a 'screen' type program to keep your session.

Part 3 - XDMCP

- Display manager
 - Provides login screen
 - Upon successful login, it launches the desktop environment
 - Examples are GDM, KDM, LXDM, LightDM (Ubuntu)
- The protocol allows for it to be run with a remote X-server

Warnings

- No encryption and NO SECURITY
- No compression, needs very high bandwidth
- Thus the server function is disabled by default
 - Can be enabled by editing the desktop manager config:
<https://wiki.ubuntu.com/xdmcp>
 - Not recommended, use X11 forwarding over ssh instead