

Setting up VNC, SAMBA and SSH on Ubuntu Linux PCs **Getting More Benefit out of Your Local Area Network**

What Are These Programs?

VNC (Virtual Network Computing) is a networking application that allows one computer's screen to be viewed by, and optionally controlled by one or more other computers through a network connection. A user who sets up and activates the VNC server software on his machine allows other remote users to view his machine's screen or even use their keyboard and mouse to control his computer just as though they were sitting at his screen and keyboard. The remote user needs to have installed VNC client software to view or control the PC that has the VNC server software. The two computers don't have to have the same operating system, so you can use VNC to view a Linux machine from your Windows PC or vice versa.

SSH (Secure SHell) is a networking application that allows you to establish and use a secure connection between two PCs on the same LAN or across the Internet. The secure connection can then be used to securely communicate, pass information, and transfer files between network-connected computers, including Windows or Linux PCs. For example, as user can use SSH to log into a server and all their interactions will be "tunneled" through an encrypted session so that if someone intercepts their data they cannot read it. SSH can also be used to establish a secure "tunnel" for VNC. (For those who want to learn how to do tunneling, it is not included in this instruction set, but you can find abundant information by "Googling" on these words: openssh tunnel .)

SAMBA (not an acronym; and in this case not a Brazilian dance style) is a networking application that allows Linux PCs to share files (and printers) with Windows PCs. Samba is a free software implementation of Microsoft's networking system.

You may find you don't need all three of these applications to meet your needs, but they are worth installing and learning so you really know which one(s) suit(s) your needs. We're betting that you want to keep and use all three.

Note: These instructions assume you are using DHCP in your LAN so that assignment of static IP addresses is unnecessary. Usually your router can provide the necessary DHCP and DNS server functions for your LAN. These instructions will tell you how to set up a Linksys WRT54GL router to serve these functions. If you have a router that cannot provide DHCP and DNS service, one of your Linux PCs can do it, but that configuration is beyond the scope of this howto.

Setting up VNC, SAMBA and SSH on Ubuntu Linux PCs

Getting More Benefit out of Your Local Area Network

Pre-Configuration Before Installing the Applications

Before you can successfully run VNC, or SSH, there are some networking configurations that must be done on all your computers. We need to make sure that your network is properly working and that all your PCs can "see" and "recognize" each other on the network. The following steps will, hopefully, accomplish the necessary network configurations.

Do this on your router -- Configure your router to be the DHCP and DNS server for your LAN. If you are using a Linksys WRT54GL router, hacked to use the HyperWRT software -- as are most of the Sunland Village East Linux Users' Group -- follow these directions:

- Open your web browser and enter your IP address in the address window. The usual IP address for home network routers is 192.168.1.1, but your router's IP may vary. Shortly, a graphical interface for configuring your router will open.
- Log into the router.
- On the setup menu select the "Setup" tab. Scan about half way down the page -- to the part labeled (in the left pane) "Network Address Server Settings (DHCP)." Check the button to enable "DHCP Server."
- On the bottom line of the same pane you will see a line called "DNS." Check the box "Use Dnsmasq for DNS."
- Be sure to click "Save Settings" at the bottom of the page, or your changes will not be saved.
- Close the web browser.

Do this on each of your Windows computers --

1. Use Notepad (on Windows machines) to open:
C:\WINDOWS\system32\drivers\etc\hosts Make sure this file exists, and that it has a line like this:

```
127.0.0.1 localhost
```

edit the line by adding your computer's hostname. If you don't know the hostname, you can find it in step 2 below. It is shown as "Computer Name." Your new line should look something like this:

```
127.0.0.1 localhost roamer # roamer is the hostname of this machine
```

2. Make sure your Windows machines are all on the same workgroup. To do this, open Start > Control Panel > System > Computer Name (tab) > Change (button). Select "Member of" workgroup, and type in the name "WORKGROUP". If you're already a member of some differently named workgroup, you can use that name. (You'll need to remember to substitute that name later in these instructions for Linux boxes.)

Setting up VNC, SAMBA and SSH on Ubuntu Linux PCs

Getting More Benefit out of Your Local Area Network

3. Open "Network Connections" and right click on "Local Area Connections." Select "Properties." Select "Internet Protocol (TCP/IP)" and right click on it. Select "Properties." Make sure that "Obtain DNS server address automatically" is checked.

Do this on each of your Linux computers --

1. System > Preferences > Remote Desktop: Check "Allow other users to view your desktop," and "Allow other users to control your desktop." Then check "Require the user to enter this password:" You need to supply a password you won't forget. (Note: You are opening up your computer for remote control. Don't even think of doing this unless you are behind a strong, active firewall, and you have set your router to block outside (Internet) access to port 22.)
2. System > Administration > Services: Make sure that "Folder Sharing Server" is turned on. If you plan on using your router as the DHCP server, make sure "DHCP Server" here is unchecked. If no such service exists, then open System > Administration > Shared Folders, and specify that you want the Samba and NFS services to be started.
3. System > Administration > Shared Folders: Add "/" and "Workgroup" (or othername if you gave the workgroup a name other than workgroup).
4. `sudo gedit /etc/hosts` : Make sure this file exists, and that it has a line something like this:

```
127.0.0.1 localhost
```

Edit the line by adding your computer's hostname. If you don't know the hostname, you can open a terminal and type `hostname` at the command line. The machine will return its name. Your new line should look something like this:

```
127.0.0.1 localhost supernova # supernova is the name of this machine
```

5. `sudo gedit /etc/dhcp3/dhclient.conf` : Open this file and scroll down about 14 lines to the line that reads `#send host-name "andre.fugue.com";` . Make a new line beneath this line, and add:

Setting up VNC, SAMBA and SSH on Ubuntu Linux PCs Getting More Benefit out of Your Local Area Network

6. send host-name "your-machines-name"; (don't forget the ; at the end of this line)

(Substitute the real name of your computer for your-machines-name. If you can't remember your machine's name, open another terminal and type `hostname`, and your machine will tell you its name. Note: Be sure you do not have a # at the start of this line, or the command will not work.)

6. Either reboot, or restart your network interface card by issuing the following commands (assumes you're using eth0 as your network interface card):

```
sudo ifdown eth0
```

```
sudo ifup eth0
```

Do this on all your computers --

Make sure all your machines are connected to your router, and are able to connect to the Internet. If not, try these troubleshooting ideas:

- **ping 127.0.0.1** on each machine. If you get good echoes, then your TCP/IP stack is okay. If not, you have basic TCP/IP configuration issues that must be resolved.
- ping the IP address of each machine. If you get good echoes, then both your TCP/IP stack and your NIC (network interface card) are working.
- ping the hostname of each machine. If you get good echoes, then you that hostname resolution is occurring.
- **ping google.com** . If you get a confirming return, your DNS server is working properly. If you don't, **ping 64.233.167.99** . If you get a good return from the IP address of Google, but not from its web address (google.com), then your DNS server is not functioning properly.
- on your Windows machine right click on "My Network Places" and select "Properties." Then right click on "Local Area Connection" and again select "Properties". Find "Internet Protocol (TCP/IP)" and make sure it is checked. Then click "Properties" and make sure you "Obtain an IP Address automatically." Also check "Obtain DNS server address automatically." Then click "Advanced" and make sure "DHCP Enabled" is checked.
- Reboot all your machines. Hopefully everything is set up okay now. If not, individualized help will be required.

Once your machines can all ping each other successfully (both by IP address and hostname) and can all reach the Internet, you are ready to download and install the three applications this how-to is all about.

Setting up VNC, SAMBA and SSH on Ubuntu Linux PCs

Getting More Benefit out of Your Local Area Network

Download, Install, and Test VNC

1. On all your Windows machines download and install VNC free edition. Download from <http://www.realvnc.com/cgi-bin/download.cgi> . Installation is self guided. This package includes both server and viewer. Size is about 720 K. (filename is vnc-4_1_2-x86_win32.exe)
2. On all your Ubuntu Linux machines open System > Administration > Synaptic Package Manager. Search for VNC. Make sure that “vino,” “vnc-common,” and “xvncviewer” are already installed. If not, install the missing ones.
3. Test VNC from Windows: Run VNC viewer from your Windows machine. You should be able to select any Linux or Windows box by hostname and connect to it; that is, if these boxes are running VNC server. If not, probably you have restricted remote desktop connections. To unrestrict, right click on "My Computer" and select "Properties." Select the remote tab. Enable remote desktop connections.
4. Test VNC from Linux: Run VNC from your Linux machine by selecting Applications > Internet > Terminal Server Client. Under the “general” tab, type in the hostname of the other machine you want to connect to, and select the VNC protocol. Then type in your name (the name you sign in with on each of your machines). Press connect. You will be asked for a password -- in a tiny box in the upper left corner of your screen. You need not move your cursor to this password box. Just type your password, and the password box somehow collects it. You should connect. If not, try removing password protection – at least during troubleshooting -- by going to System > Preferences > Remote Desktop and un-checking password protection.
5. Firewalls frequently are the cause of failed connections. If all the above methods fail, it is a good idea to look at your firewall settings. Assuming you are using Windows built-in firewall, go to "Control Panel" and select "Windows Firewall." Click on the "Exceptions" tab. Click the "Add Port" button. Add "VNC-1" and use port 5900. Then, click "Add Port" again and add "VNC-2" at port 5800.

There are at least two alternative ways to run VNC from your Linux boxes: (a) install Gnome-RDP and use it; or (b) open a terminal window and type: xvncviewer hostname . All methods will give you the same GUI window. For ease of use, we prefer the terminal window approach, but use whichever method you like. Do try them all.

Note: Networking with VNC is insecure, so VNC should be used "stand alone" only on your LAN and only behind a strong firewall. If you want to use VNC over the Internet, it needs to be paired with SSH --described below -- so the VNC connection is "tunneled" in an encrypted, secure connection. Furthermore, VNC servers should not be left running and open unless there you know there are no users on your LAN that you cannot trust -- as in your own home -- and there is a strong firewall preventing access from the Internet.

Setting up VNC, SAMBA and SSH on Ubuntu Linux PCs

Getting More Benefit out of Your Local Area Network

Download, Install, and Test SSH

1. On all your Windows machines download and install OpenSSH client. Download from <http://www.wm.edu/it/index.php?id=2928> The download site provides installation instructions. This package is the *client side only*, which allows you to initiate a SSH session from the machine you install it on. Size is about 5.7 MB. (filename is sshsecureshellclient-3.2.9.exe)
2. On all your Windows machines download and install the COPSSH server for Windows. To do this, open a browser and search Google for COPSSH. Look for the Sourceforge download site, and select it. Your download will be from this URL: http://sourceforge.net/project/showfiles.php?group_id=69227&package_id=127780 . Note that SSH Servers for Windows are mostly very expensive. To get around this expense someone developed Cygwin, a tiny Linux system that embeds in Windows and serves COPSSH from within a Linux environment on your Windows machine. This free COPSSH- server software allows your Windows PC to "serve" up its information to a remote PC (Linux or Windows). The access of your Windows-machine's drives is through the Linux folder called cygdrive.
3. On all your Ubuntu Linux machines open System > Administration > Synaptic Package Manager. Search for ssh. Make sure openssh-client and openssh-server are both installed. If they are not, install them.
4. Test SSH from Windows: From within your Windows machine's Start menu, select "SSH Secure Shell Client." Then select Secure File Transfer Client. Select Quick Connect, and type in the name of the machine you want to connect to. Add your username, and click Connect.
5. Test SSH from Linux: From within your Linux machine, select Places > Connect to Server. Under Service Type, select SSH. Add the name of the Server you want to connect to, and press Connect. This should put an icon on your desktop and in your Nautilus file browser.

Setting up VNC, SAMBA and SSH on Ubuntu Linux PCs

Getting More Benefit out of Your Local Area Network

Download, Install, and Test SAMBA

1. You need do nothing on your Windows machines. Microsoft Windows Networking is the Windows version of Samba, and it is preinstalled.
2. Assuming that you are also the user of the other machine that would request access to this one, on each Linux box open a terminal and type:

```
sudo smbpasswd -a "username"
```

where "username" is the name you use to sign onto each machine. You must do this because Samba does not readily use Linux passwords, so it tries to apply its own password system. We supply Samba with the same password we used for Linux so it will still authenticate you. (Note: You will do this differently if you are trying to allow others to share your machine's information through Samba.)

3. Test SAMBA on Windows: Open Windows Explorer (right click Start and select Explore). Select My Network Places > Entire Network > Microsoft Windows Network > Workgroup. You should see all your machines listed here if we successfully configured SAMBA.
4. Test SAMBA on Linux: Select Places > Connect to Server. Select "Windows share" as the service type. Type the hostname of your Windows machine into the Server box. Add your user name. Hit connect. You should get a desktop icon for the Windows file system on your Linux desktop if everything is configured properly.

version 4.2

mof: 3-17-2007