Implementing a Secure Home Intranet and VPN Solution Using Linux

Michael Todd Muskovin

The ability to connect to multiple computers from both a user's home and a remote location is very valuable. By utilizing Linux and open technologies, creating a home intranet and VPN can allow a user to access his or her own home network from a remote location and also provides the needed layer of security when traversing an unknown network such as a hotspot or other public Internet connection. This instructional presents clear and concise directions for creating a secure home intranet and VPN solution using Linux.

ISBN 978-1-4116-9716-4

Implementing a Secure Home Intranet and VPN Solution Using Linux

Michael Todd Muskovin

Copyright

Cover Art by Paper and Ink Designs
http://www.paperandinkdesigns.com

Dedication

I would like to dedicate this work to my wife and children. Thank you all for allowing me the needed time away for studies, and for the constant love and support especially in times of frustration. And to my parents for instilling in me the drive needed to further my education.

Acknowledgement

I would like to acknowledge Dr. Doug Blakemore for his forward thinking and ability to captivate students with his charismatic classroom instruction. I would also like to acknowledge Mrs. Diane Zoellmer, my fourth grade teacher, who first introduced me to the world of computing. I don't know what I'd be doing today without her guidance and special interest in my early development.

Table of Contents

INTENDED AUDIENCE 5

REQUIREMENTS 6

STANDARDS 7

GLOSSARY 8

CHAPTER 1 INTRODUCTION 10

CHAPTER 2 SECURITY PRACTICES 11

CHAPTER 3 CREATING THE LINUX SERVER 12

Samba Setup 12

DynDNS Setup 15

OpenVPN Setup 17

CHAPTER 4 MAC CLIENT SETUP 21

Connecting to the VPN 21

Connecting to the Samba Shares 22

Using VNC over VPN 23

CHAPTER 5 WINDOWS CLIENT SETUP 25

Connecting to the VPN 25

Connecting to the Samba Shares 26

Using VNC over VPN 27

RESOURCES 29

Intended Audience

This instructional is intended for an audience of technology professionals, hobbyists, and anyone interested in increasing accessibility to and reliability of data and services from remote locations. A general understanding of the Linux operating system, networking, and Internet technology is required. This instructional has been written specifically for Fedora Core 4. While other Linux distributions may be used in conjunction with the instructions provided, many commands and file locations may vary across each distribution. To learn more about Fedora Core or to download the latest release, visit http://www.redhat.com/fedora/. The Linux and Fedora communities are very large and more than willing to assist with questions. Some great forums include The Fedora Community Portal (http://fcp.homelinux.org/), Fedora Forum (http://www.fedoraforum.org/), and Linux Forums (http://www.linuxforums.org).

Requirements

The equipment required for implementing a home intranet and VPN solution using Linux includes a dedicated Linux computer for use as a server, a Mac or Windows computer for use as a client, and an Ethernet router. While a broadband Internet connection is not required for all aspects of implementation, it is highly recommended for improved performance and necessary for remote connections utilizing VNC.

Standards

Throughout this instructional, Linux commands will be referenced with a prompt of # (pound) or $ (dollar sign). Commands beginning with the # prompt require execution by a user with root authority. Those commands beginning with the $ prompt may be executed by any user. In order to execute a command as root, login with a standard user and issue the shell user command and the password for root.

```
$ su
Password:
```

When finished with the use of root authority exit the shell user.

```
# exit
```

As a rule, do not log in as root and do not use root authority for anything that does not require it. This prevents any unintended changes that may make your system vulnerable to attack.

Glossary

The following list defines technical terms as used in this document.

*nix	An abbreviation for Unix-based operating systems (e.g. Unix, Linux)
DNS	Domain Name System – used to match easily recognizable domain names (e.g. google.com) to IP addresses (e.g. 64.233.187.99)
Firewall	A piece of hardware or software that filters network traffic to prevent unauthorized access to a computer system or network
GUI	Graphical User Interface – A method of navigating a computer system using visual components such as windows and pointing devices
IP	Internet Protocol – A numeric value used to identify a computer system on a network (e.g. 192.168.0.1)
Intranet	A collection of computers connected for the purpose of sharing data and services
ISP	Internet Service Provider – A company who provides Internet connectivity
Port	A path by which specific services communicate between server and client
Root	The administrative account on a *nix system with complete system access
Service	An application that performs a specific function
Tunnel	The practice of wrapping data of one protocol in another encrypted protocol to enable secure communications

VNC	Virtual Network Client – A GUI connection to a remote computer
VPN	Virtual Private Network – A secure network connection between multiple computers across the Internet

Chapter 1 Introduction

Today's market has made it possible for home users to connect multiple computers with a router or switch. However, the overriding motivation for such network connectivity has been to share a single broadband Internet connection. Many individuals are not even aware of the great power they have in their homes already. By harnessing the power of existing consumer level equipment and utilizing open source and free technologies, it is possible to do much more with today's home network. Adding a dedicated Linux server, which is really nothing more than a desktop computer reserved for serving data and services, provides home users the ability to store data in a centralized location, access data on multiple intranet computers from a remote location, access the Internet from a remote location securely through the home VPN, and much more.

Chapter 2 Security Practices

Security has become an essential part of today's computing environment. There exist standard security practices that should always be followed no matter what environment you are working with. For instance, when creating a password, make sure that it is long, not in the dictionary, and includes letters, numbers, and special characters. By following these guidelines, you are making it more difficult for crackers to reveal your password using brute force dictionary attacks. And resist the temptation to use a password more than once. In this way, you are segmenting one point of authentication from another. Another security risk is access to your network from external attackers. By making sure that only services in use are started and unused ports are blocked at the firewall/router level, you are making it extremely difficult to access the internal network. (Laporte & Gibson, 2005)

Chapter 3 Creating the Linux Server

As stated earlier, the Linux distro used for this instruction will be Fedora Core 4. Most aspects of this instructional can be completed using the GUI interface such as Gnome or KDE. However, there are additional options when configuring services using the command prompt than configuring them with the GUI. After your installation is complete, make sure to update all packages on your system. A great package management application called yum (yellow-dog update manager) works well for making sure that you have the latest versions of all packages. Be aware that this may take some time depending on the number of packages you chose in the initial installation. You will need to perform yum as root.

```
$ su
# yum upgrade
```

Samba Setup

Samba (or SMB as it is formally known) is a protocol for sharing file and printing services across networked computers. Because it is supported in Linux, Mac, and Windows operating systems, it is ideal for use in our intranet implementation. (Ts, Eckstein, and Collier-Brown, 2004)

Before we create a share, let's create a directory named share on the root of the filesystem and give permissions to all user.

```
# mkdir /share
# chmod 777 /share
```

Now that we have created a directory for sharing data throughout the network, it is time to configure and start Samba. All configuration files and daemons necessary for running Samba are already installed on your server if you included the Samba package in the initial install. First, make a backup of your Samba configuration file and then take a look at it.

```
# cp /etc/samba/smb.conf /etc/samba/smb.conf.orig
# vi /etc/samba/smb.conf
```

The comments in this file (designated with a pound sign or semi-colon) are very descriptive and should allow you to configure the Samba shares as you'd like. Some important lines to note are listed below.

Designate your workgroup name here (for Windows file sharing)

```
# workgroup = NT-Domain-Name or Workgroup-Name
  workgroup = MYGROUP
```

Make sure to uncomment this line and add the appropriate subnets including 10.8.0. for use across the VPN (to be explained in later in the VPN section).

```
# This option is important for security. It allows you to restrict
# connections to machines which are on your local network. The
# following example restricts access to two C class networks and
# the "loopback" interface. For more examples of the syntax see
# the smb.conf man page
  hosts allow = 192.168.1. 10.8.0. 127.
```

The following section allows you to define your server as the computer on the network that will survey and provide network share information to the rest of the computers on the network. This can be very useful if you are operating in an environment with Windows. All Windows computers are, by default, assigned as the Browse Master. This means that they will attempt to gather information about the network's workgroup and share information and make that information available to the rest of the network's computers. This becomes a problem when multiple Windows systems exist in the same network. Because all of the Windows systems attempt to become the browse master, they each negotiate to become the master. This most often occurs quickly and has no effect on the network. However, it may occasionally take a long time to resolve the browse master negotiation. Until a browse master has been decided, no workgroup or share information is available to the Windows computers making networked computing difficult if not impossible. Knowing that this was a problem with Windows, the developers of Samba implemented browse master options in the configuration file. Using the Samba server

as a browse master and setting its OS level above the default level of Windows systems makes it possible to prevent any delayed negotiation.

```
# Browser Control Options:
# set local master to no if you don't want Samba to become a master
# browser on your network. Otherwise the normal election rules apply
    local master = yes

# OS Level determines the precedence of this server in master browser
# elections. The default value should be reasonable
    os level = 33

# Preferred Master causes Samba to force a local browser election on
startup
# and gives it a slightly higher chance of winning the election
    preferred master = yes
```

The Share Definitions section provides many different examples of a share using various parameters such as groups and authorities. For our project, we'll use the following share definition to share the directory that we created earlier.

```
# A shared directory accessible and writable by username
[share]
   comment = Shared Storage
   path = /share
   writable = yes
   public = no
   valid user = username
```

After you have edited the smb.conf file to your satisfaction and created the directory shares you desire, assign a Samba password for the username user.

```
$ smbpasswd username
New SMB password:
Retype new SMB password:
```

Now start the Samba service.

```
# /etc/rc.d/init.d/smb start
```

The following are the commands for obtaining status, restarting and stopping a service. The status command can be very helpful because it returns whether the service is currently running along with the services process id (pid).

```
# /etc/rc.d/init.d/service status
# /etc/rc.d/init.d/service restart
# /etc/rc.d/init.d/service stop
```

Once Samba has been successfully started, enable it to start on boot.

```
# /sbin/chkconfig --add smb
```

Samba is amazingly powerful and we have just barely scratched the surface of its capabilities. For additional configurations read the freely available Using Samba by Jay Ts, Robert Eckstein, and David Collier-Brown at http://us2.samba.org/samba/docs/using_samba/toc.html

DynDNS Setup

If you have a broadband connection with a static IP address you may skip this section. For the majority of broadband subscribers, however, a dynamic IP address is more likely. A dynamic IP address means that the IP address assigned to you by your ISP may change. In order to access the intranet from a remote location, you must be able to identify the IP address at your home. By utilizing the Custom DNS service by DynDNS, it is possible to assign a unique domain name to your changing IP address. (Custom)

In order to keep our dynamic IP address up-to-date with your unique domain, we'll utilize software for automatically updating the Custom DNS service every time your IP changes.
Visit http://www.dyndns.com/support/clients/ and download an available update client for Linux. At the time of this writing, ddclient v3.6.6 by

Paul Burry and Others was available so the following instructions will be specific to that client. After downloading the client to your Linux server, unpack it to your home directory. A README file exists inside the ddclient-3.6.6 directory. README offers complete installation instructions but I will copy them here for convenience. In order to complete the following steps, you must have root authority. You can do this by issuing the su command.

INSTALLATION:

```
cp ddclient /usr/sbin/
mkdir /etc/ddclient
cp sample-etc_ddclient.conf /etc/ddclient/ddclient.conf
vi /etc/ddclient/ddclient.conf
-- and change hostnames, logins, and passwords appropriately
```

UNCOMMENT THE LINE OFFERING IP ADDRESS UPDATE FROM WEB STATUS PAGE. UNCOMMENT THE LOGIN AND PASSWORD LINE. IF YOU ARE USING DYNDNS CUSTOM DNS SERVICES AS INSTRUCTED IN THE PREVIOUS SECTION, UNCOMMENT THE dyndns.org custom addresses SECTION.

BECAUSE WE ARE USING FEDORA CORE WE WILL FOLLOW THIS SECTION ALSO.

```
## For those using Redhat style rc files and using daemon-mode:
cp sample-etc_rc.d_init.d_ddclient /etc/rc.d/init.d/ddclient
## enable automatic startup when booting
/sbin/chkconfig --add ddclient
## start the first time by hand
/etc/rc.d/init.d/ddclient start
```

YOU WILL GET AN ERROR STATING THAT THE FILES MUST ONLY BE ACCESSIBLE BY THEIR OWNER. WHEN THE SERVICE STARTS THIS PROBLEM IS AUTOMATICALLY RESOLVED BY CHANGING THE AUTHORITY TO –rw------

---

After you have completed this installation, check to make sure the post is running correctly by logging in to your dyndns.com account and checking the last update time. If everything is functioning properly you may delete the ddclient-3.6.6 directory in your user home directory. (Ddclient)

OpenVPN Setup

OpenVPN is free virtual private network software (both client and server) that allows multiple computers to connect, from anywhere, through a secure tunnel. Let's say that you want to be able to access your shared directories at home when you're at a hotel. With OpenVPN, you can connect to your home computer as if it were in the next room. Or better yet, say you are connected to a hotspot at a cafe and want to access the Internet. All of your Internet traffic is available to network users with packet sniffing tools such as Ethereal. By utilizing OpenVPN, you can connect to your home server and use remote desktop software to access the Internet using encrypted packets that cannot be cracked using Ethereal.

OpenVPN functions by creating a virtual network adapter that is actually a tunnel identifying itself in the 10.8. range (or whatever non-routable range you prefer). When a client connects to an OpenVPN server, the ensuing activity is secure by accessing the 10.8. IP identified as the server (usually 10.8.0.1). The server and clients are secured with a 1024-bit RSA key. (OpenVPN)

The most current stable version of OpenVPN available at the time of this writing was version 2.05 and may be downloaded at http://www.openvpn.net/. The instructions included here are a clear and simple version of the complete instruction found at http://www.openvpn.net/howto.html.

Fedora Core utilizes an easy package manager called RPM (RedHat Package Manager) for application installation. The easiest way to download and install OpenVPN is by using this automated process. FedoraProject.org has a RPM available for the latest release of OpenVPN at http://fedoraproject.org/extras/4/i386/repodata/repoview/openvpn-0-2.0.5-3.fc4.html

After downloading the RPM file, install it.

```
# rpm -ivh openvpn-2.0.5-3.fc4.i386.rpm
```

After installation is complete, we'll need to move the directories to the most appropriate location (within /etc).

```
# cp -R /usr/share/doc/openvpn-2.0.5 /etc/openvpn
# cp -R /usr/share/openvpn/easy-rsa /etc/openvpn/easy-rsa
```

The first step in OpenVPN setup is creating a master certificate authority certificate and key.

```
# cd /etc/openvpn/easy-rsa
# . ./vars
# ./clean-all
# ./build-ca
```

At this point you'll be asked to identify yourself for the Certificate Authority. Enter all information accordingly and when prompted for common name enter 'OpenVPN-CA'
Next, let's create the server key (using common name "server")

```
# ./build-key-server server
```

And now, create the client keys (you can create as many as you'd like just make sure to give them appropriate common names client2, client3, etc.).

```
# ./build-key client1
```

The next step will create the necessary 1024-bit Diffie Hellman parameters (for generating a secure cipher).

```
# ./build-dh
```

At this point, you may navigate to the keys subdirectory (/etc/openvpn/easy-rsa/keys) and copy the client files (client1.crt, client1.key, etc.) to the appropriate client making sure that the communication is secure because the *.key file is the secure key.

Now we need to edit the OpenVPN configuration file.
Copy the sample configuration file to the OpenVPN directory.

```
# cp /etc/openvpn/sample-config-files/server.conf /etc/openvpn
```

Make sure to change the following lines as shown:

```
# SSL/TLS root certificate (ca), certificate
# (cert), and private key (key).  Each client
# and the server must have their own cert and
# key file.  The server and all clients will
# use the same ca file.
#
# See the "easy-rsa" directory for a series
# of scripts for generating RSA certificates
# and private keys.  Remember to use
# a unique Common Name for the server
# and each of the client certificates.
#
# Any X509 key management system can be used.
# OpenVPN can also use a PKCS #12 formatted key file
# (see "pkcs12" directive in man page).
ca easy-rsa/keys/ca.crt
cert easy-rsa/keys/server.crt
key easy-rsa/keys/server.key  # This file should be kept secret

# Diffie hellman parameters.
# Generate your own with:
#   openssl dhparam -out dh1024.pem 1024
# Substitute 2048 for 1024 if you are using
# 2048 bit keys.
dh easy-rsa/keys/dh1024.pem
```

In order for OpenVPN to function outside the intranet, it is necessary to forward port 1194 to the server's IP at the firewall.

Now that all else is set, let’s start OpenVPN.

```
# /etc/rc.d/init.d/openvpn start
```

Once OpenVPN has been successfully started, enable it to start on boot.

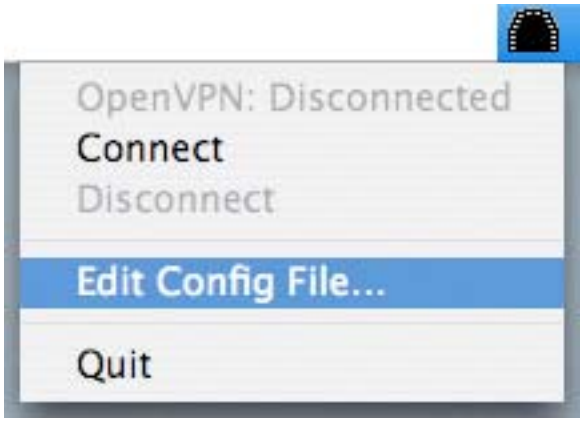
```
# /sbin/chkconfig --add openvpn
```

Chapter 4 Mac Client Setup

Connecting to the VPN

Before continuing to any additional procedures, copy the ca.crt, client1.crt, and client1.key files from the OpenVPN folder on your Linux server to a location on your client system (make sure not to use the same client key on computers that will be connecting at the same time, if multiple computers will be connecting, create additional client keys (e.g. client2, client3, etc.)). Because the *.key file contain secure connection information, make sure to transfer it using secure means (in other words, don't use email). (OpenVPN howto)

To connect to the VPN as a client using Mac OS X, download Tunnelblick (http://tunnelblick.net/) and install it using the standard installation image. After installation is complete, you will see a new icon in the menu bar that looks like a tunnel.
Click on the tunnel icon and choose "Edit Config File…"



This opens the OpenVPN config file for this client machine. Edit and save the config file with the following changes.

Change the remote hostname to your DynDNS address.

```
# The hostname/IP and port of the server.
# You can have multiple remote entries
# to load balance between the servers.
remote your.dyndnsaddress.com. 1194
;remote my-server-2 1194
```

Change the paths to the location where you saved the ca.crt, client1.crt, and client1.key files.

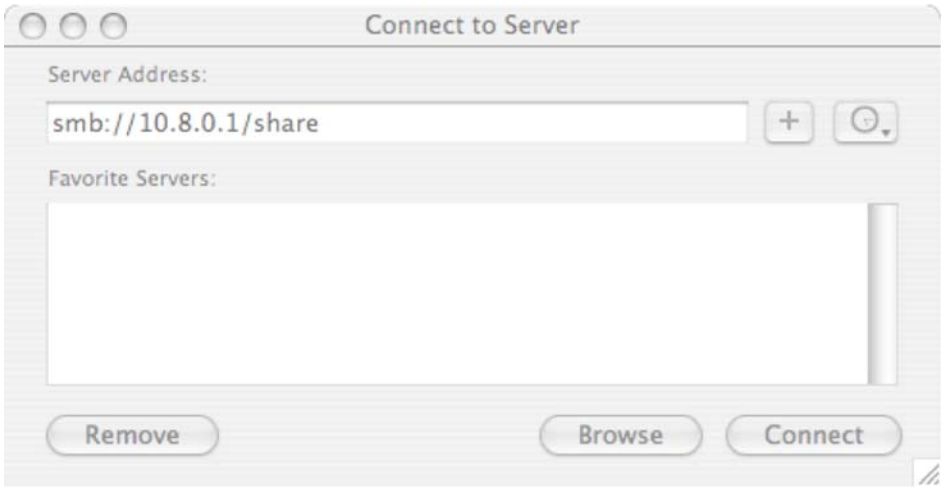
```
# SSL/TLS parms.
# See the server config file for more
# description.  It's best to use
# a separate .crt/.key file pair
# for each client.  A single ca
# file can be used for all clients.
ca /users/username/documents/ca.crt
cert /users/username/documents client1.crt
key /users/username/documents client1.key
```

Now you are ready to connect the your VPN server. Simply click the tunnel icon and choose connect.

Connecting to the Samba Shares

Now that we have created a secure connection over VPN, we can connect to our Samba share.
From the Finder, click “Go” and “Connect to Server” (shortcut command-K).

In the Connect to Server window, enter the samba address using the smb protocol, the VPN server address (always 10.8.0.1) and the share name.



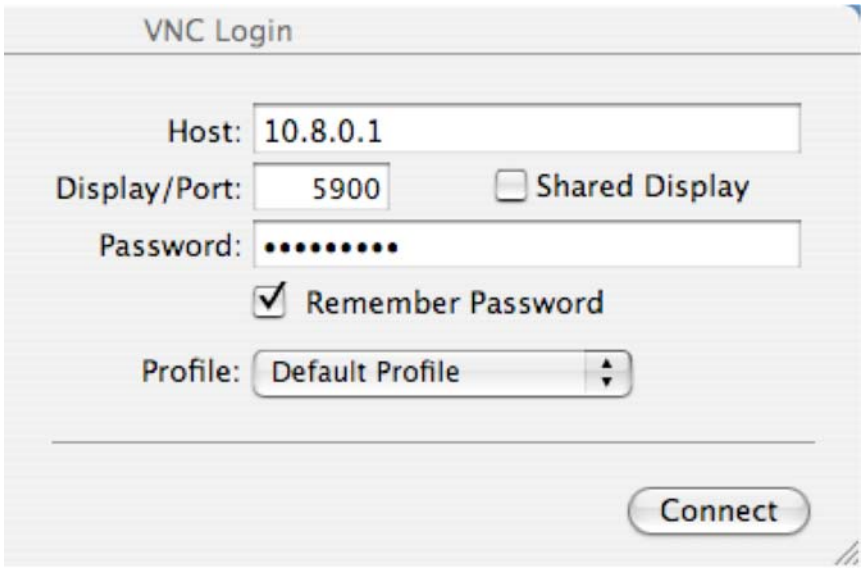
When prompted for a username and password, enter the username you assigned in the Samba section earlier.

Using VNC over VPN

VNC (virtual network computing) makes it possible to remotely connect the desktop of your VPN server from a remote computer. The greatest use for VNC over VPN is connecting to the Internet from an insecure network connection. An insecure connection, such as a hotel or café, can be eavesdropped with free packet sniffing software. Therefore, connecting through your secure VPN tunnel and surfing the Internet from your home server, protects you from prying eyes.

An excellent VNC client for OS X is Chicken of the VNC. It can be downloaded at http://sourceforge.net/projects/cotvnc/. Installation of COTVNC is fairly simple and use is even simpler.

After installing COTVNC, open the application and enter the VPN server address, port number, and password.



Click Connect and you’re off to surfing over an encrypted connection.

Chapter 5 Windows Client Setup

Connecting to the VPN

Before continuing to any additional procedures, copy the ca.crt, client1.crt, and client1.key files from the OpenVPN folder on your Linux server to a location on your client system (make sure not to use the same client key on computers that will be connecting at the same time, if multiple computers will be connecting, create additional client keys (e.g. client2, client3, etc.)). Because the *.key file contain secure connection information, make sure to transfer it using secure means (in other words, don't use email). (OpenVPN howto)

To connect to the VPN as a client using Windows XP, download OpenVPN GUI (http://openvpn.se/download.html) and install it accepting all of the standard options. After installation is complete, you will see a new icon in the system tray that looks like a network connection with a globe.

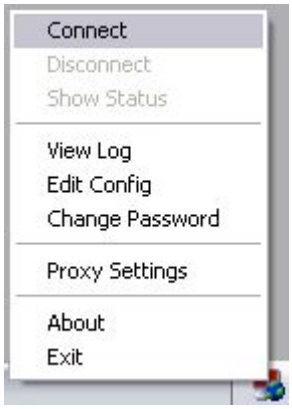
Copy C:\Program Files\OpenVPN\sample-config\client.ovpn to C:\Program Files\OpenVPN\config\client.ovpn and edit the file making appropriate changes below.
This opens the OpenVPN config file for this client machine. Edit and save the config file with the following changes.

Change the remote hostname to your DynDNS address.

```
# The hostname/IP and port of the server.
# You can have multiple remote entries
# to load balance between the servers.
remote your.dyndnsaddress.com. 1194
;remote my-server-2 1194
```

Change the paths to the location where you saved the ca.crt, client1.crt, and client1.key files.

```
# SSL/TLS parms.
# See the server config file for more
# description.  It's best to use
# a separate .crt/.key file pair
# for each client.  A single ca
# file can be used for all clients.
ca c:\\ca.crt
cert c:\\client2.crt
key c:\\client2.key
```

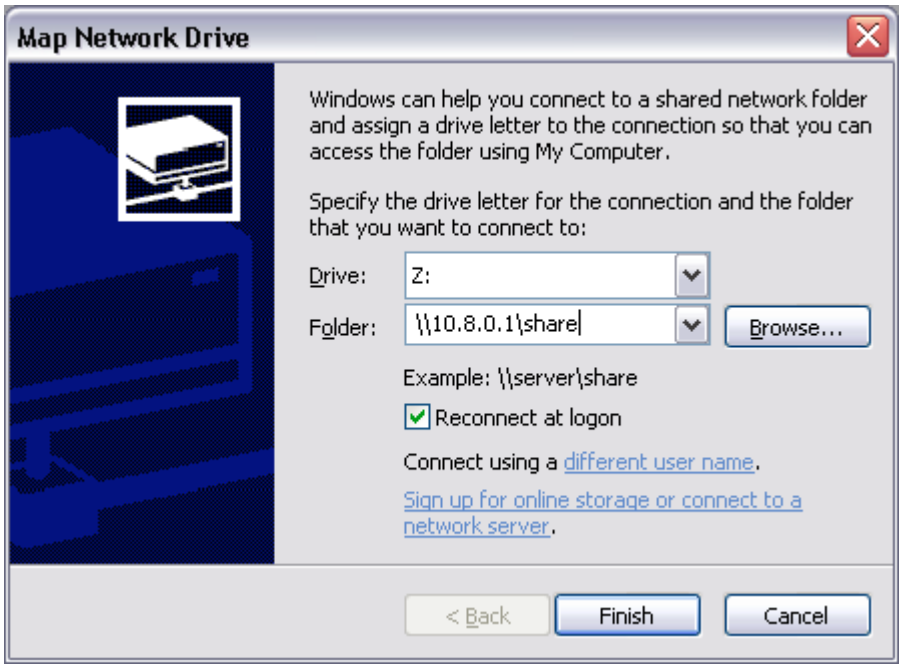


Now you are ready to connect the your VPN server. Simply right-click the OpenVPN icon and choose connect.

Connecting to the Samba Shares

Now that we have created a secure connection over VPN, we can connect to our Samba share. From within My Computer, click “Tools”, then “Map Network Drive …”

In the Map Network Drive window, choose a drive letter, type the address of the Samba share (\\VPNServerAddress\SambaShareName), and click Finish.



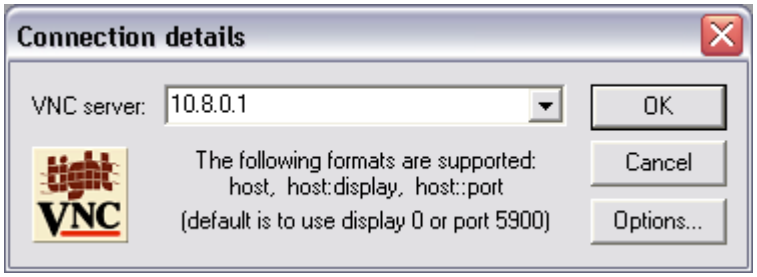
When prompted for a username and password, enter the username you assigned in the Samba section earlier.

Using VNC over VPN

VNC (virtual network computing) makes it possible to remotely connect the desktop of your VPN server from a remote computer. The greatest use for VNC over VPN is connecting to the Internet from an insecure network connection. An insecure connection, such as a hotel or café, can be eavesdropped with free packet sniffing software. Therefore, connecting through your secure VPN tunnel and surfing the Internet from your home server, protects you from prying eyes.

An excellent VNC client for Windows XP is TightVNC. It can be downloaded at http://www.tightvnc.com/download.html. All default options should be accepted when executing the installer.

After installing TightVNC, open the application and enter the VPN server address.



Click OK, enter the VNC password, and you're off to surfing over an encrypted connection.

Resources

Custom DNS. (n.d.) Retrieved March 14, 2006, from http://www.dyndns.com/services/dns/custom/

How to install. (n.d.) Retrieved March 14, 2006, from http://ddclient.sourceforge.net/

Laporte, L. & Gibson, S. (September 8, 2005). Personal Password Policy. *Security Now!, 4.* [Podcast]. Available from Security Now! Web site, http://grc.com/securitynow.htm

OpenVPN. (n.d.) Retrieved March 14, 2006, from http://openvpn.net/

OpenVPN 2.0 howto. (n.d.) Retrieved March 14, 2006, from http://openvpn.net/howto.html

Ts, J., Eckstein, R., & Collier-Brown, D. (2003). *Using Samba* (2nd ed.). California: O'Reilly. Available from Using Samba 2nd Edition Web site, http://us2.samba.org/samba/docs/using_samba/toc/html