Setting up a VPN-Concentrator based upon OpenVPN 2.1 rc19

Author: Pieter Hotting
Date: July-2009
1 About this document

This document describes how to setup an OpenVPN-server. It will take you by the hand to get a fully working VPN-concentrator.

2 What component do we need

I used the following components:
- A virtual machine on VMWare server
  - Virtual harddisk: 3GB
  - 2 ethernet cards
  - Memory 512 Mb
  - 1 Processor
- Operating system: OpenSuse 11.1 – 32 bit
- VPN-software: OpenVPN 2.1 rc15
- Smarty 2.6.20
- Modified version of OpenVPN-Web-GUI

Ofcause you can change the settings of the virtual machine to your liking. For operating system I used the 32 bit version, because I like to use this virtual machine on VMWare server running on windows 32-bit (like XP) as well. If you make a dedicated server the 64 bit version is prefered.

Optional you could create a second small harddisk for swap. A virtual machine doesn't care about another harddisk and it will give you the flexibility to enlarge or reduce the size of swap-space by just replacing the virtual harddisk.
3 Different types of VPN-concentrators

3.1 NAT or no NAT

It is possible to build VPN-concentrators that NAT or do not NAT. In the next chapters I will highlight the difference.

3.1.1 No NAT

The source address of an incoming VPN-connection is the IP-address of the VPN-tunnel, not an address on the local network. The local network has to be aware how to reach that network. This means that the routing tables must be adjusted, so the machines on the local network know where to reach the remote machine. The advantage however is that machines on the local network can reach the remote PC and the remote PC can reach the machines on the local network.

3.1.2 NAT

With NAT enabled, the source address of an incoming VPN-connection will be translated to a IP-address on the local network. In that case no routing adjustment are needed on the local network. The source address of the incoming VPN-connection will be hidden with NAT behind the local interface of the VPN-concentrator. One thing to remember with hiding NAT is that incoming connections are possible, but connections from the inside to the remote machine is not possible. This could be considered a handicap or a security advantage. In this document we have choosen for the setup with NAT.
3.2 One interface or two interfaces

You can setup a VPN-concentrator with 2 interfaces or just with one interface. In the next chapters you'll find a description of the difference between the two.

3.2.1 Two interfaces

In this case the external interface is used for incoming VPN-connections. The internal interface connects to the internal network. Below you will see a sample network layout:

![Diagram showing two interfaces](image1)

3.2.2 One interface

In an one interface configuration the incoming VPN-connections and the local network connect both to the same interface. Below you will see a sample network layout:

![Diagram showing one interface](image2)

3.2.3 Conclusion

The 2 interfaces setup is considered more secure, because it separates the (mean and ugly) outside world from the (need and clean) inside world. The 1 interface setup is easier and more suitable for small networks (with only one internal network). With careful configuration it is possible to make a safe environment with the 1 interface setup as well. In this document we will build a VPN-concentrator that can handle both.
4 The Installation

4.1 OpenSuse version 11.1

4.1.1 Requirements

The only requirements are a server (on VMWare Server) and a OpenSuse 11.1 DVD. I will assume that the virtual machine which we will install OpenSuse on is already setup according to the specifications you can find in Chapter 2. Please note that almost everything (if not everything) in this document will also work if it is installed on a normal PC or server.

4.1.2 Steps

1. Boot the machine from the OpenSuse DVD
2. From the menu choose: Install
3. Press F5 and select safe setting (needed for some versions of VMWare Server)
4. Press the Enter key to start the installation.

5. Next the following screen will appear

6. Leave the Language en Keyboard settings and accept the license by clicking on Next.
7. Select **New Installation** and click on **Next**

8. Select your country and region and click **Next**

9. In the screen for Desktop Selection:
   - Choose **Other**
   - Choose **Minimal Server Selection (Text Mode)**
10. At Suggested Partitions:
   - Choose **Partition Base** and accept defaults.

11. Now you must create an user. Any username and password will do. Please note by default it is selected to use the password also as root password. We will need this later. Uncheck the box for Autologon. Click on **Next**
12. On the next screen you will see an overview of all things that are selected for installation. You can still change them, but leave them as they are and click on **Install**.

13. A confirmation window will popup. Just click on **Install** once more.

14. After a while the installation process will stop and the system is rebooted for the first time.

15. At the prompt, login as user **root** with the password you selected earlier.

16. With the command “passwd” you can change the root password to whatever you like.

17. Remove the smartd service, because it doesn't work in VMWare.
   
   **type:** `inserv -r /etc/init.d/smartd`

18. Remove the acpid service, because it doesn't work in VMWare.
   
   **type:** `inserv -r /etc/init.d/acpid`

19. Remove the nfs service, because it doesn't work in VMWare.
   
   **type:** `inserv -r /etc/init.d/nfs`

20. Start yast:
   
   **type:** `yast`

21. The next descriptions could be a bit short if you are not familiar with yast. Here are some basics to try:
   
   – Use the up and down arrow to go through the menus
   – Use the tab-key to go to the next menu/box
   – Use the enter-key to select something
   – The yellow letters point to a quick link. You can choose that option by holding the alt-key while pressing the letter.

22. Now make a choice about your configuration:
   
   - If you have no internet connection skip to step 33.
   - If you have a DHCP-server in your network, you can skip to step 26.
   - If you have no DHCP-server in your network, continue at step 23.
23. Select **Network Devices** → **Network Settings** and hit the **Enter** key.

![Network Settings](image1)

24. Select eth0 and select **Edit**:

![Edit eth0](image2)

25. Select **Statically assigned IP Address** and enter the required **IP-address** and **Subnet Mask**. Then select **Next** and hit the **Enter** key. Hit the **Enter** key once more to get back into the main menu.
26. Perform an online update (if you have a direct Internet connection). This will take care of any known problems and security issues. 
   Select **Online Update** and press the **Enter** key.

27. A list of updates will be shown:

28. Select **Accept** and press the **Enter** key. The download of the updates is started.
29. Press Enter.

30. Select Online Update again and press Enter. A list with downloaded, but not yet installed updates is shown.

31. Select Accept and press Enter. And the updates will be installed.
32. After this leave the yast-tool. Select **Quit** and press **Enter**.
33. Reboot the system, because some updates might need this:
   type: `shutdown -r now`
34. When the system is rebooted login again as root.
35. Start yast at the prompt type:
   `yast`

36. Go to **Network Devices → Network Card** and press the **Enter** key.

   ![Network Devices](image)

   **Note:** By selecting one of the ethernet cards as soon above, you can see the device name on the bottom in the grey box!

37. Choose **eth0** and select: **Edit**
38. Check **No IP Address (for Bonding Devices)** than select **Next** and hit the Enter key.

39. Now repeat step 31 and 32 for the eth1 interface.

40. Go back to the main menu of yast (select **OK** and hit the Enter key)

41. Go to **Security and Users → Firewall** and press the **Enter** key.

42. Check **Disable Firewall Automatic Starting**, select **Next** and hit the Enter key.
43. Select **Yes** and hit the Enter key.

44. Select **Finish** and hit the Enter key.

45. To manage the OpenVPN-server from remote, in ssh should be used. You can use a program like putty to do this. Standard OpenSuse uses reverse DNS to identify who you are. It could be that your PC from which you connect to the server is not known in the DNS. In that case the DNS query would fail, resulting in a delay before the logging prompt appears. I find this very irritating so it is best to disable this DNS query. This is done as following:
   - Edit: `/etc/ssh/sshd_config`
   - Add the following line: `UseDNS no`. Exit the editor.
   - Restart ssh-service: `/etc/init.d/sshd restart`.

46. Install ntp to keep the correct time. This only works when you have a permanent internet connection. If you don't have it skip to step XXX.
   Correct time settings is important for certificates. E.g. If you create a new certificate and distribute it to a client, than you want that certificate to become active directly. If there is a time difference you could see the message: The certificate is not active yet and depending on the time difference you might wait a few minutes up to several hours. The only way to prevent this is to keep the correct time on both sides. Please note this is a default installation of NTP. For tuning it for VMWare see appendix B:
   - Start yast: `yast`
   - Goto: **Network Services → NTP Configuration** and then press the **Enter** key.

47. Set: **Now and On Boot**. A popup will appear to tell you need a permant internet connection. Just close it.
48. Select **OK** and hit the **Enter** key. Please note, the configuration of NTP is done later by replacing the ntp.conf script during the installation of the openvpn-web-gui software.

49. As a last step you need to add the required packages from yast:
   - **Select: Software → Software Management** and hit the **Enter** key.

50. Install the packages listed below. I find it the easiest to use search to locate and select them.
    - gcc
    - lzo-devel
    - libopenssl-devel
    - pkcs11-helper
    - pkcs11-helper-devel
    - make
    - apache2
    - apache2-mod_php5
    - gd
    - php5-gd
    - php5-openssl
    - php5-zlib

51. When you got them all, select **Accept** and hit Enter.
52. A message will appear to show you which automatic changes will be applied. Just select **OK** and hit **Enter**.

A full list of all installed packages can be found in Appendix A.

This concludes the installation of the base system. In the next chapters the applications will be installed.
4.2 OpenVPN 2.1 rc19

Getting software on a server that doesn't have a network connection is slightly difficult. Therefore I made a iso-file of
the openvpn-software, which can be mounted as a virtual cd. You can find this iso-file on the website. I assume that you
have “inserted” this virtual cd in the cdrom-drive. Than follow the next steps:

1. Mount the Openvpn CD:
   mount /dev/cdrom /mnt
2. Create the required directories:
   mkdir /opt/openvpn
   mkdir /opt/openvpn/source
   mkdir /opt/openvpn/bin
   mkdir /var/log/openvpn
3. copy the source-tarball “openvpn-2.1_rc19.tar.gz” to the /opt/openvpn/source directory
   cp /mnt/openvpn-2.1_rc19.tar.gz /opt/openvpn/source
4. At this point we don't need the CD anymore, so unmount it:
   umount /mnt
5. cd /opt/openvpn/source
6. tar -xzf openvpn-2.1_rc19.tar.gz
7. cd openvpn-2.1_rc19

   The next 3 steps will build the openvpn software:
   8. ./configure
   9. make
   10. make install
   11. Copy the easy-rsa scripts to an usefull spot:
       cp -r /opt/openvpn/source/openvpn-2.1_rc19/easy-rsa/2.0/* /opt/openvpn/bin
4.3 Apache2

Although Apache2 is already installed with yast, we need to make some adjustments. We will run in plan (none secure!) http and not in https mode. I assume that the server will be placed in a DMZ or a separate server-segment in the network. In that case “listening on the line” and capturing sensitive information is secured in another way.

Here are the steps how to do this.

1. In the file /etc/apache2/default-server.conf, in the section `<Directory “/srv/www/htdocs”>` change the following line:
   
   Options None
   
   into
   
   Options Indexes +FollowSymlinks +Includes Multiviews
   AuthType Basic
   AuthName "Unauthorized access is not allowed!"
   AuthUserFile /etc/apache2/passwords/openvpn
   Require valid-user

2. Insert Apache as startup service:
   
   `insserv /etc/init.d/apache2`

3. Now we will create the directory to hold the passwords:
   
   `mkdir /etc/apache2/passwords`

4. Create an username and password that is allowed to access the website. Below you can see how you can create a new user called admin.
   
   `htpasswd2 -c /etc/apache2/passwords/openvpn admin`

   You will be asked for a password twice. Enter the password you like and remember it!
4.4 The OpenVPN-Web-GUI

The GUI is based strongly on the OpenVPN-Web-GUI. However I modified it to the latest version of OpenVPN and also to my liking. What I like about the GUI is:

- It gives you an overview of the connected clients
- It makes it very easy to create certificates
- It makes it very easy to create a zip-file that can easily be installed on a client
- Logging can be viewed
- Revoking certificates works!

To make the installation easier, I also create a CD with all the stuff needed to make the Web-GUI work. This CD can be found on the website. I assume that you have “inserted” this virtual cd in the cdrom-drive. Than follow the next steps:

1. Mount the Openvpn CD:
   ```bash
   mount /dev/cdrom /mnt
   ```

2. Install the gui:
   ```bash
   cd /
   tar -xvzf /mnt/OpenVPNGUI-2009-07-27.tgz
   chown root.www /opt/openvpn/bin
   chmod 664 /opt/openvpn/bin
   ```

3. At this point we don't need the CD any more:
   ```bash
   umount /mnt
   ```

4. In the `/opt/openvpn/bin-vars` file you'll find a section where the defaults are stored for generating certificates. Setting the default right values will make it easier to quickly add users. Edit vars and set the red values to your preference:
   ```bash
   export KEY_COUNTRY="NL"
   export KEY_PROVINCE="Utrecht"
   export KEY_CITY="Nieuwegein"
   export KEY_ORG="OpenVPN-GUI"
   export KEY_OU="Support"
   ```
   If required you can also look at the other values and adjust them to your needs. Please be careful. The defaults should be ok.

5. Edit the OpenVPN-configuration file `/opt/openvpn/bin/openvpn-config` and modify the ETH0 section to match your local network. Please only modify the thing marked in red below:
   ```bash
   ETH0_IP=192.168.3.51
   ETH0_MASK=255.255.255.0
   ```

6. At this point the server will not yet understand routing. After rebooting you can connect to the OpenVPN-Web-GUI from the local network. If this is not enough, please also edit `/etc/routes` and add the required routing. However this is easier done from the GUI.

7. At OpenVPN to your startup scripts:
   ```bash
   inserv /etc/init.d/custom
   inserv /etc/init.d/openvpn-server
   ```

8. At this point you have installed all software. You are ready to start the OpenVPN-Web-GUI for the first time, but before you do it is a good idea to reboot, so you are sure you are starting with a clean machine:
   ```bash
   shutdown -r now
   ```
Appendix A – Full packages list

This is a full list of all installed packages on the base system. Most of them are installed by default with the installation of OpenSuse. The command `rpm -qa | sort` was used to retrieve this list.

```plaintext
ConsoleKit-0.2.10-14.2
DirectFB-1.1.1-48.1
Mesa-7.0.3-35.1
PolicyKit-0.8-14.1
PolicyKit-doc-0.8-17.1
SuSEfirewall2-3.6_SVNr1959-9.2
zp2ps-4.13-1275.1
aaa_base-11.0-79.2
acl-2.2.4-6.1
acpid-1.0-6.63.2
alsa-plugins-1.0.16-57.1
apache2-2.2.8-28.1
apache2-mod_php-5-5.2.6-0.1
apache2-prefork-2.2.8-28.1
apache2-utils-2.2.8-28.1
apparmor-docs-2.3-24.1
apparmor-parser-2.3-23.1
apparmor-profiles-2.3-11.1
apparmor-utils-2.3-14.1
at-3.1.8-1040.1
atk-1.2.2-0.23.1
attr-2.3-49.40.1
audit-1.6-8.18.1
audit-libs-1.6.8-18.1
autos-5.0.3-40.1
autofs-2-installation-2.16-19.0.1
avalon-0.6.22-68.1
bash-3.2-112.1
bash-doc-3.2-71.1
bc-1.06-896.1
bind-libs-9.4.2-39.2
bind-utils-9.4.2-29.2
binutils-2.18.50.20080409-11.1
bltkrace-0.99.3-99.1
bluez-libs-3.3-2.1
bootcycle-0.3-208.1
bootsplash-3.3-126.1
bootsplash-branding-openSUSE-3.1-16.1
branding-openSUSE-11.0-12.1
bundle-lang-common-en-11.0-118.1
bzzip2-1.0.5-13.1
cairo-1.4-14.32.1
cdktkit-ctools-compat-1.1.7-16.1
checkmedia-2.1-113.1
cifs-mount-3.2-20.22.1
coreutils-6.11-9.1
cpio-2.9-60.1
cpp-4.3-39.1
cpp43-4.3.1-20080507-6.1
cracklib-2.8.12-24.2
cracklib-dist-full-2.8.12-20.1
cron-4.1-172.1
cryptsetup-1.0.5_SVNr46-21.1
cups-1.3-27.1

dhcpcl-3.2-3.6.1
diffutils-2.8.7-118.1
dirmngr-1.0-21.1
dmraid-1.0.14c-63.1
dos2unix-3.1-413.1
dosfs Toolkit-2.11-119.1
d2fsprogs-1.40-8.20.1
de-0.2-975.1
detect-2.1.0-116.1
dhool-6-52.1
expat-2.0.1-62.1
fbset-2.1-892.1
file-4.24-13.1
filesystem-11.0-42.1
filoops-1.42-216.1
filters-2007.8.9-124.1
findutils-4.4.0-13.1
finger-1.3-118.1
fontconfig-2.4-2.9-90.1
foomatic-filters-3.2-221.1
freetype-2.3.3-5.62.2
fuse-2.7-32.2
gawk-3.1.5h-37.1
gcc-4.3.39.1
gcc43-4.3.1-20080507-6.1
gconf2-2.22-0.28.1
gconf2-branding-openSUSE-2.20-53.2
gd-2.0-36.RC1-19.1
gdmn-1.83-344.1
genisoimage-1.1.7-16.1
ggettext-runtime-0.17-40.1
gx/font-0.4.14-4.1
gx/font-branding-openSUSE-4.0.14-4.1
ghostscript-fonts-other-8.62-17.1
ghostscript-fonts-std-8.62-17.1
ghostscript-library-8.62-17.1
ghostscript-omni-8.62-17.1
glib2-2.16.3-20.2
glib2-branding-openSUSE-2.16-15.1
glibc-2.8-14.1
glibc-devel-2.8-14.1
glibc-locale-2.8-14.1
glitz-0.5.6-144.1
gmp-4.2.2-30.1
gnome-keyring-2.22.1-22.1
gnutls-2.2.2-17.1
gpart-d1-1h-595.1

gpg-pubkey-0dfb318841-ed929b

gpg-pubkey-307e3d54-8df130a

gpg-pubkey-3d255f3d-9-36e1204

gpg-pubkey-7e2e3605-4816488f

gpg-pubkey-9e800ca-491543a

gpg-pubkey-a1912206-46fa0899

```
hal-0.5-18.1
hal-0.5-18.1
hal-0.5-18.1
hal-0.5-18.1
hal-0.5-18.1
hal-0.5-18.1
hal-0.5-18.1
hal-0.5-18.1
hal-0.5-18.1
hal-0.5-18.1
hal-0.5-18.1
hal-0.5-18.1
hal-0.5-18.1
hal-0.5-18.1
hal-0.5-18.1
hal-0.5-18.1
```