7.3 Install on Linux and Initial Configurations

This section describes how to install SoftEther VPN Server to a Linux operating system. This assumes that in the Linux operating system, no extra application software is installed after performing a clean install of the system. This also assumes that, as a basic rule, the firewall and similar functions included in the Linux distribution are not being used, and that the function for blocking communication to TCP/IP ports from the outside (firewall function) is disabled.

7.3.1 Recommended System

Recommended Operating System Configuration

The Linux version of SoftEther VPN Server can operate in most cases on platforms with Linux kernel 2.4 or later; however, SoftEther VPN Project recommends only those environments using the following Linux distributions. (As of the time of writing this manual, this is the recommended environment; however, this may change to higher specifications in the future.)

- **Red Hat Enterprise Linux**
- **Fedora**
- **CentOS**

For more information about the system requirements, please refer to Specifications.

The descriptions for installing SoftEther VPN Server in this chapter are based on the use of one of the above operating systems and the fact that VPN Server will be installed to the newly created directory **/usr/local/vpnserver/**.

Installing Linux

For Linux distribution, support is only provided for environments where a clean installation of the system was performed with one of the following methods.

1. Perform a clean installation of Linux. Avoid cases where inconsistencies may occur, such as in the libraries after upgrading from an earlier version of Linux.
2. When creating a partition on the hard disk, be sure to allocate sufficient disk space to the partition with the **/usr/** directory. The examples in the descriptions below are based on VPN Server being installed to **/usr/local/vpnserver/**. In

addition, we recommend allocating sufficient disk space to the partition to allow VPN Server to write log files to the same directory.

3. At the stage of selecting components to be installed, at the minimum, the **development tools (compiler, etc.) and development libraries** are installed at the same time. When installing VPN Server, the **make** and **gccbinutils** utilities and the **libc (glibc), zlib, openssl, readline,** and **ncurses** development library versions (also called devel) are required.
4. After installing the operating system, update to the latest Linux kernel **(2.6.9-22 or later)**. Because there are problems in the parallel and synchronous processing of the kernel included in the initial install of Linux, the operations of VPN Server may become unstable. Be sure to update the kernel.
5. Complete the installation of the program with the firewall and SELinux functions disabled. After confirming that VPN Server is properly installed, you can enable these functions only if necessary.

7.3.2 Selecting the Installation Mode

As described in 3.1 Operating System Requirements and 3.2 Operating Modes, SoftEther VPN Server can be operated in either service mode or user mode. When configuring VPN Server for use as part of an everyday operation system in a company, we recommend installing SoftEther VPN Server in service mode. To install the VPN Server program to the system in service mode on a Linux operating system, you must register the vpnserver program as a daemon program in the Linux startup script.

7.3.3 Checking the Required Software and Libraries

The following software and libraries are required to install VPN Server to a Linux operating system. Check that the following software and libraries are installed to the system and are enabled. (If the recommended environment distribution is installed using the method specified in 7.3.1, these libraries are also installed.)

- **gcc software**
- **binutils software**
- **tar, gzip or other software for extracting package files**
- **chkconfig system utility**
- **cat, cp or other basic file operation utility**
- **EUC-JP, UTF-8 or other code page table for use in a Japanese language environment**
- **libc (glibc) library**
- **zlib library**
- **openssl library**
- **readline library**
- **ncurses library**
- **pthread library**

7.3.4 Extracting the Package

Preparing the Installer File

To install VPN Server, you need to prepare the file containing the VPN Server program (package file compressed with tar.gz format). You can download the latest VPN Server installer file from the SoftEther VPN Project website (**http://www.softether.org/**).

Extracting the Package File for Installation

Extract the package file for installation using the tar command. Copy the tar.gz file to a directory and extract the file as follows.

```
[root@machine root]# tar xzvf vpnserver-5070-rtm-linux-x86.tar.gz
vpnserver/
vpnserver/vpnserver.a
vpnserver/vpncmd.a
vpnserver/hamcore.se2
vpnserver/libcrypto.a
vpnserver/Makefile
vpnserver/libssl.a
vpnserver/License_ReadMeFirst.txt
vpnserver/License_ReadMeFirstUtf.txt
vpnserver/License_ReadMeFirstSjis.txt
vpnserver/.install.sh
```

When the package is extracted, the directory "vpnserver" is created in the working folder, and the required installation files are extracted.

7.3.5 Creating an Executable File

Executing a make

To install VPN Server, you must execute a make and create a vpnserver executable file.

First, go to the vpnserver directory extracted in the previous subsection and type [make].

Next, the message "Do you want to read the License Agreement for this software?" is displayed. Select [1] to continue.

```
[root@machine vpnserver]# make
./.install.sh

Do you want to read the License Agreement for this software ?
1. Yes
2. No
Please choose one of above number:
1
```

Next, the end-user license agreement is displayed. Please read and understand the license agreement. The license agreement is displayed over several pages, so use a terminal emulator or SSH client software with a scroll function to view the entire license agreement. If you are unable to read the entire license agreement, press Ctrl + C to cancel the make, and then use a text editor to directly open and view the contents of the text file with the license agreement located in the vpnserver directory.

At the end of the license agreement, the message "Did you read and understand the License Agreement?" is displayed. If you read and understood the license agreement, select [1].

```
EULA



Did you read and understand the License Agreement ?
(If you couldn't read above text, Please read License_ReadMe.txt
file with any text editor.)
1. Yes
2. No
Please choose one of above number:
1
```

Next, the message "Do you agree to the License Agreement?" is displayed. If you agree to the license agreement, select [1].

```
Did you agree the License Agreement ?
1. Agree
2. Do Not Agree
Please choose one of above number:
1
```

Once you agree to the license agreement, the vpnserver program is automatically created.

```
make[1]: Entering directory `/root/vpnserver'
ranlib libssl.a
ranlib libcrypto.a
ranlib vpnserver.a
gcc vpnserver.a -pthread -lrt -lm -lz libssl.a libcrypto.a -lpthread -ld
-lreadline -lcurses -o vpnserver
strip vpnserver
ranlib vpncmd.a
gcc vpncmd.a -pthread -lrt -lm -lz libssl.a libcrypto.a -lpthread
-ldl -lreadline -lcurses -o vpncmd
strip vpncmd
make[1]: Leaving directory `/root/vpnserver'

[root@machine vpnserver]#
```

If an error occurs during this process, creation of the vpnserver program fails. In this case, see 7.3.1 and 7.3.3 again and check whether any required libraries are missing.

7.3.6 VPN Server Location

After the vpnserver program is created, we recommend moving the vpnserver directory, which is created when the package is extracted, to the /usr/local/ directory. Use the following method to move the vpnserver directory to /usr/local/. The operations hereafter must be performed as a root user.

```
[root@machine vpnserver]# cd ..
[root@machine root]# mv vpnserver /usr/local
[root@machine root]# ls -l /usr/local/vpnserver/
Total 13000
-rwxrwxrwx 1 root root 20245 12Œ8 16:14 License_ReadMeFirst.txt*
-rwxrwxrwx 1 root root 20317 12Œ8 16:14 License_ReadMeFirstSjis.txt*
-rwxrwxrwx 1 root root 30210 12Œ8 16:14 License_ReadMeFirstUtf.txt*
-rwxrwxrwx 1 root root 609 12Œ8 16:14 Makefile*
-rwxrwxrwx 1 root root 4018399 12Œ8 16:14 hamcore.se2*
-rwxrwxrwx 1 root root 1942994 12Œ9 02:23 libcrypto.a*
-rwxrwxrwx 1 root root 336070 12Œ9 02:23 libssl.a*
-rwxr-xr-x 1 root root 1814216 12Œ9 02:23 vpncmd*
-rwxrwxrwx 1 root root 1630858 12Œ9 02:23 vpncmd.a*
-rwxr-xr-x 1 root root 1814120 12Œ9 02:23 vpnserver*
-rwxrwxrwx 1 root root 1630304 12Œ9 02:23 vpnserver.a*
[root@machine root]#
```

Confirm that all of the files are moved to the /usr/local/vpnserver/ directory, as shown above.

If the user does not have root permissions, the files in the vpnserver directory cannot be read, so change and protect the permissions.

```
[root@machine root]# cd /usr/local/vpnserver/
[root@machine vpnserver]# chmod 600 *
[root@machine vpnserver]# chmod 700 vpncmd
[root@machine vpnserver]# chmod 700 vpnserver
[root@machine vpnserver]# ls -l
Total 13000
-rw------- 1 root root 20245 12ŒŽ 8 16:14 License_ReadMeFirst.txt
-rw------- 1 root root 20317 12ŒŽ 8 16:14 License_ReadMeFirstSjis.txt
-rw------- 1 root root 30210 12ŒŽ 8 16:14 License_ReadMeFirstUtf.txt
-rw------- 1 root root 609 12ŒŽ 8 16:14 Makefile
-rw------- 1 root root 4018399 12ŒŽ 8 16:14 hamcore.se2
-rw------- 1 root root 1942994 12ŒŽ 9 02:23 libcrypto.a
-rw------- 1 root root 336070 12ŒŽ 9 02:23 libssl.a
-rwx------ 1 root root 1814216 12ŒŽ 9 02:23 vpncmd*
-rw------- 1 root root 1630858 12ŒŽ 9 02:23 vpncmd.a
-rwx------ 1 root root 1814120 12ŒŽ 9 02:23 vpnserver*
-rw------- 1 root root 1630304 12ŒŽ 9 02:23 vpnserver.a
[root@machine vpnserver]#
```

This completes the changing of the location of the vpnserver program.

7.3.7 Using the vpncmd Check Command to Check Operations

We recommend performing a final check to see whether VPN Server can operate properly on your computer system before starting vpnserver.

You can use the **check** command on the vpncmd command line management utility to automatically check whether the system has sufficient functions to operate VPN Server. For details, please refer to 6.6 VPN Tools Command Reference.

First, start vpncmd by typing **[./vpncmd]**. Next, select [Use of VPN Tools (certificate creation or communication speed measurement)] and execute the check command.

```
[root@machine vpnserver]# ./vpncmd
vpncmd command - SoftEther VPN Command Line Management Utility
SoftEther VPN Command Line Management Utility (vpncmd command)
```

```
By using vpncmd program, the following can be achieved.

1. Management of VPN Server or VPN Bridge
2. Management of VPN Client
3. Use of VPN Tools (certificate creation and communication speed measur

Select 1, 2 or 3: 3

VPN Tools was launched. By inputting HELP, you can view a list of the co
hat can be used.

VPN Tools>check
Check command - Check if SoftEther VPN Operation is Possible
------------------------------------------------
SoftEther VPN Operation Environment Check Tool

If this operation environment check tool is run on a system and that sys
es, it is highly likely that SoftEther VPN software can operate on that
his check may take a while. Please wait...

Checking 'Kernel System'...
[Pass]
Checking 'Memory Operation System'...
[Pass]
Checking 'ANSI / Unicode string processing system'...
[Pass]
Checking 'File system'...
[Pass]
Checking 'Thread processing system'...
[Pass]
Checking 'Network system'...
[Pass]

All checks passed. It is highly likely that SoftEther VPN Server / Bridg
rate normally on this system.

The command terminated normally.
VPN Tools>exit
[root@machine vpnserver]#
```

If, after executing the check command, the message "Passed all checks. It is likely that VPN Server / Bridge will operate properly on this system." is displayed, as shown above,

it is likely that your system has satisfied the VPN Server operation requirements and VPN Server can safely be used.

If, however, the system fails at any of the above check items, we recommend checking 7.3.1 and 7.3.3 again.

7.3.8 Registering a Startup Script

After installing vpnserver to the /usr/local/vpnserver/ directory using the method described above, you can configure your system to operate the vpnserver program as a service mode program by registering the **/usr/local/vpnserver/vpnserver** program as a daemon process that continues to run in the background while Linux is starting.

To register vpnserver to Linux as a daemon process, create a startup script, as shown below, with the name **/etc/init.d/vpnserver**. (The following startup script is a description example, and you may have to rewrite part of the script for it to work properly on your system.)

```
#!/bin/sh
# chkconfig: 2345 99 01
# description: SoftEther VPN Server
DAEMON=/usr/local/vpnserver/vpnserver
LOCK=/var/lock/subsys/vpnserver
test -x $DAEMON || exit 0
case "$1" in
start)
$DAEMON start
touch $LOCK
;;
stop)
$DAEMON stop
rm $LOCK
;;
restart)
$DAEMON stop
sleep 3
$DAEMON start
;;
*)
echo "Usage: $0 {start|stop|restart}"
exit 1
esac
exit 0
```

You can use a text editor or the cat command to write the above script to /etc/init.d/vpnserver as a text file. To use the cat command to create the script, press Ctrl + D after the line break in the final line, as shown below.

```
[root@machine vpnserver]# cat > /etc/init.d/vpnserver
#!/bin/sh
# chkconfig: 2345 99 01
# description: SoftEther VPN Server
DAEMON=/usr/local/vpnserver/vpnserver
LOCK=/var/lock/subsys/vpnserver
test -x $DAEMON || exit 0
case "$1" in
start)
$DAEMON start
touch $LOCK
;;
stop)
$DAEMON stop
rm $LOCK
;;
restart)
$DAEMON stop
sleep 3
$DAEMON start
;;
*)
echo "Usage: $0 {start|stop|restart}"
exit 1
esac
exit 0
```

After creating the /etc/init.d/vpnserver startup script, change the permissions for this script so that the script cannot be rewritten by a user without permissions.

```
[root@machine vpnserver]# chmod 755 /etc/init.d/vpnserver
```

Lastly, use the chkconfig command to allow the above startup script to start automatically in the background when the Linux kernel starts.

```
[root@machine vpnserver]# /sbin/chkconfig --add vpnserver
```

VPN Server is now prepared to run as a service mode program.

7.3.9 Starting and Stopping Service

VPN Server registered as a service mode program automatically starts when Linux starts and automatically stops when Linux shuts down. You can manually stop or restart the VPN Server service if you need to do so for management reasons.

To start or stop VPN Server registered as a service mode program, type the command below.

Starting the VPN Server Service

With the VPN Server service not running and with root permissions, type the following to start the VPN Server service.

```
[root@machine vpnserver]# /etc/init.d/vpnserver start
```

Stopping the VPN Server Service

With the VPN Server service running and with root permissions, type the following to stop the VPN Server service.

```
[root@machine vpnserver]# /etc/init.d/vpnserver stop
```

Cases in Which You Must Stop the VPN Server Service

The VPN Server service must be manually stopped in the following cases.

- When manually editing or replacing the configuration file
- When updating the vpnserver program and other files after the release of a new version of VPN Server (To replace the vpnserver, vpncmd and hamcore.se2 files, be sure to stop the service in advance.)
- When you want to restart the service due to erratic behavior of the operating VPN Server

Forcibly Terminating the vpnserver Process

It is unlikely that VPN Server would malfunction due to a problem with the physical memory of the computer or a software bug. If this should occur and the VPN Server service does not respond when you try to stop the service using the method above, you can stop the service by forcibly terminating the vpnserver process. For the detailed method for forcibly terminating the vpnserver process, please refer to the method of using the kill command described in 3.2 Operating Modes.

7.3.10 Limitations when Starting with General User Rights

The Linux version of VPN Server can also be started with general user rights. When starting VPN Server as a user mode program with general user rights, the program cannot be registered as a system service, but when a general user starts the VPN Server program in the background by typing **[./vpnserver start]**, unlike the Windows version, the Linux version of the vpnserver process can continue to run even after that user logs out. SoftEther VPN Project does not recommend actually operating VPN Server in user mode for the following reasons.

- The local bridge function cannot be used. (For details, please refer to 3.6 Local Bridges.)
- After starting the system, the user must log on and manually start the vpnserver process, decreasing operability.