OpenNA Linux 1.0

The Official OpenNA Linux Installation Guide

OpenNA Linux 1.0
OpenNA Linux Installation Guide

ISBN: N/A

Introduction

Welcome to the OpenNA Linux Installation Guide. This guide contains useful information to assist you during the installation of OpenNA Linux 1.0. From fundamental concepts such as installation preparation to the step-by-step installation procedure, this guide will be a valuable resource as you install OpenNA Linux.

This manual will walk you through a typical installation using the OpenNA Linux CD-ROMs. Once you have completed the installation as outlined in this manual, you will have a fully functioning system.

Is Your Hardware Compatible?

Hardware compatibility is particularly important if you have an older system or a system that you built yourself. OpenNA Linux 1.0 should be compatible with most hardware in systems that were factory built within the last four years. However, hardware specifications change almost daily, so it is hard to guarantee that your hardware will be 100% compatible.

Installation Disk Space Requirements

OpenNA Linux provides different type of complete secure server installation for your network and depending on the type of server that you would like to install, the disk space requirements will change accordingly. A basic installation will easily fit on **135** MB where a full Desktop installation will required **517** MB of disk size.

Server

A server installation requires **135** MB through **156** MB depending of the type of server you want to install.

Desktop

A Desktop installation requires at least **517** MB of free space.

Workstation

A Workstation installation, including a Graphical Desktop Environment (GUI) and most software development tools, requires at least **845** MB of free space.

Install Using the CD-ROM

The recommended method to install OpenNA Linux is with CD-ROM. Installing from a CD-ROM requires that you have purchased a OpenNA Linux 1.0 boxed set or have a OpenNA Linux CD-ROM and a CD-ROM drive. All modern computers will allow booting from the CD-ROM without any problem. Just check your BIOS setting if you have problem to boot from your CD-ROM drive.

Installation types

The following installation types are available with OpenNA Linux.

1. Server
A server installation is most appropriate if you intend your system to function as a secure and fast Linux-based server. Different predefined and secure type of server installation will be provided to you by the installer. Here is a list of available secure and fast server type.

- Web Server
- FTP Server
- DNS Server
- E-Mail Server
- Database Server
- Gateway Server
- Virtual Server

2. Desktop
A desktop installation is most appropriate if you would like your system to function as a secure and fast Linux-based desktop. This installation will create a system for your home, laptop or desktop use. A graphical environment (GUI) will be installed.

3. Workstation
A workstation installation is most appropriate if you would like a graphical environment, as well as software development tools to compile, develop and create source codes, program, etc with your secure & fast Linux system.

Installing OpenNA Linux

The following explain how to install OpenNA Linux from a CD-ROM.

- Starting the Installation Program
- The Text Mode Installation Program User Interface (TUI)
- Optional Configuration
- Manual Installation

Starting the Installation Program

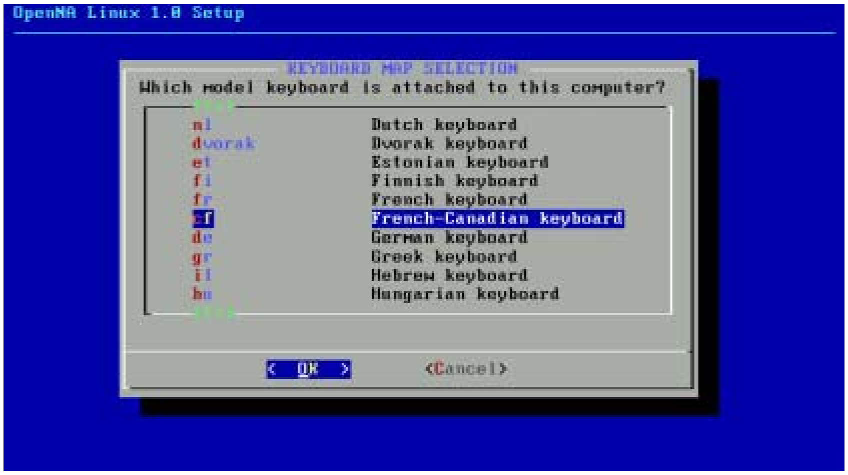
To start the installation, you must first boot the installation program. You can boot the OpenNA Linux installation program using a bootable CD-ROM drive to perform a local CD-ROM installation (this is the recommended method). The installation program will then probe your system and attempt to identify your CD-ROM drive. If found, you will continue to the next stage of the installation process by pressing **[ENTER]**.

The Text Mode Installation Program User Interface (TUI)

The OpenNA Linux text mode installation program uses a screen-based interface that includes most of the on-screen "widgets" commonly found on graphical user interfaces. The text mode installation program will let you install OpenNA Linux from scratch on your computer.

Step1. Keyboard Map Selection

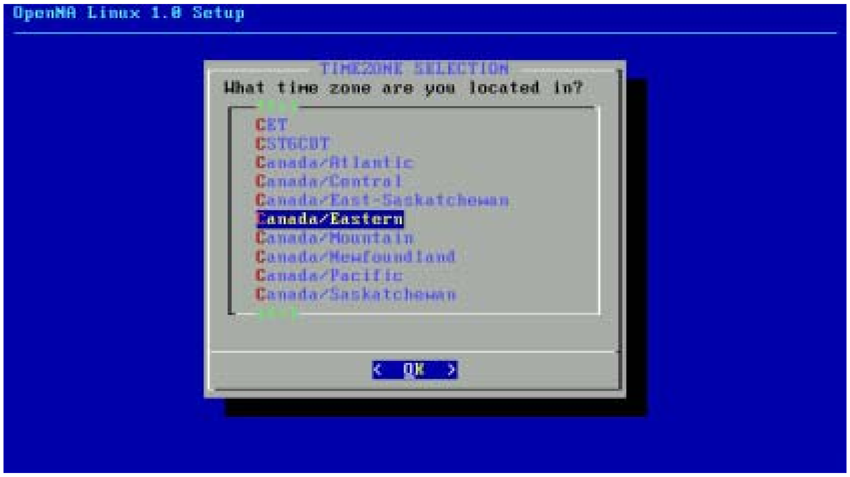
The first section of the installer is the Keyboard Map Selection screen which let you select the correct layout type for the keyboard you would prefer to use as the system default.



```
OpenNA Linux 1.0 Setup

              KEYBOARD MAP SELECTION
 Which model keyboard is attached to this computer?
        nl                 Dutch keyboard
        dvorak             Dvorak keyboard
        et                 Estonian keyboard
        fi                 Finnish keyboard
        fr                 French keyboard
        cf                 French-Canadian keyboard
        de                 German keyboard
        gr                 Greek keyboard
        il                 Hebrew keyboard
        hu                 Hungarian keyboard
```

Step2. Time Zone Selection

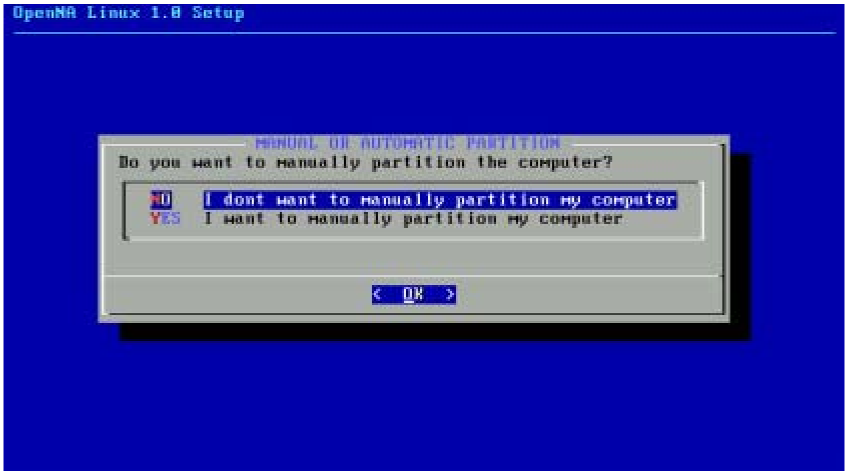
The second section of the installer will let you choose the time zone from where you are living.



```
OpenNA Linux 1.0 Setup

          TIMEZONE SELECTION
 What time zone are you located in?
     CET
     CST6CDT
     Canada/Atlantic
     Canada/Central
     Canada/East-Saskatchewan
     Canada/Eastern
     Canada/Mountain
     Canada/Newfoundland
     Canada/Pacific
     Canada/Saskatchewan
```

Step3. Manual or Automatic Partition

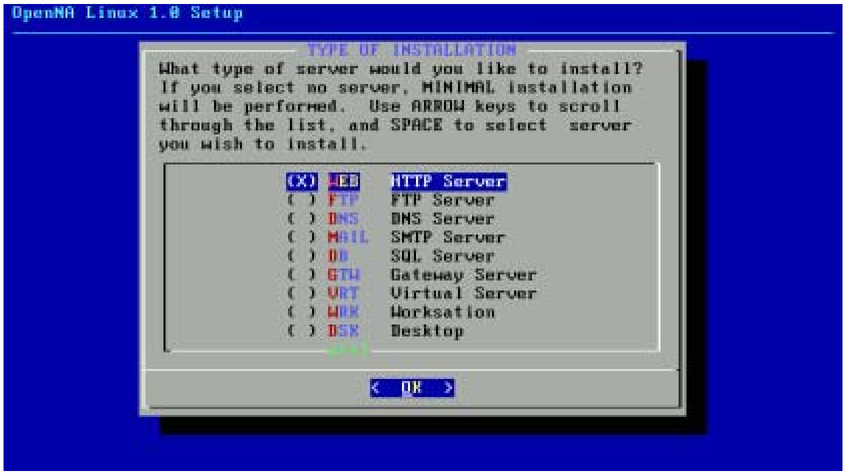
The third section of the installer will ask you whatever you would like to manually partition the computer or let the installer automatically do it for you. The most important difference between both choices are that automatic partition will automatically partition, format, install and configure the type of server you have decided to setup on your system. This is the fast and recommended method of installation for most users. Manual installation gives you more flexibility by letting you manually partition, format and install the type of server that you want. Both options provide the same feature and server type.



```
OpenNA Linux 1.0 Setup

MANUAL OR AUTOMATIC PARTITION
Do you want to manually partition the computer?

   NO    I dont want to manually partition my computer
   YES   I want to manually partition my computer

                    < OK >
```

Step4. Type of Installation

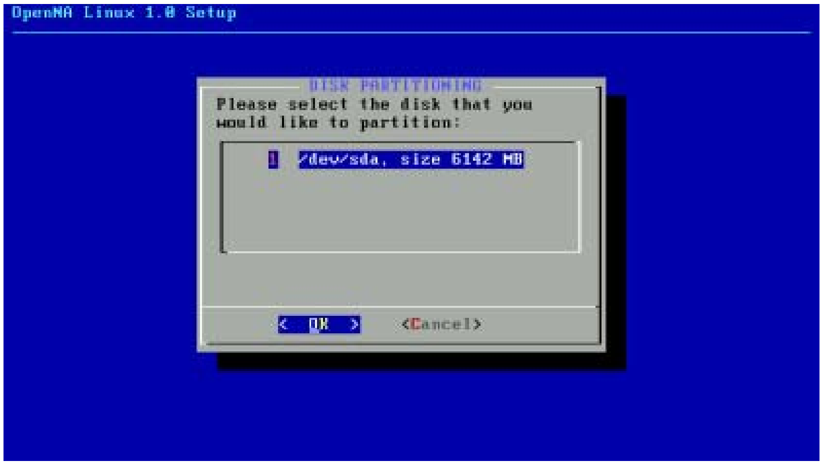
Once you have chosen whatever to let the installer automatically or to manually partition your computer, the next step will be to choose the type of server that you would like to install. Please note that depending on your choice, additional questions will be asked during the installation procedure. This will let you better control what need to be installed, configured and automated on your system.



```
OpenNA Linux 1.0 Setup

TYPE OF INSTALLATION
What type of server would you like to install?
If you select no server, MINIMAL installation
will be performed.  Use ARROW keys to scroll
through the list, and SPACE to select  server
you wish to install.

          (X) WEB    HTTP Server
          ( ) FTP    FTP Server
          ( ) DNS    DNS Server
          ( ) MAIL   SMTP Server
          ( ) DB     SQL Server
          ( ) GTW    Gateway Server
          ( ) VRT    Virtual Server
          ( ) WRK    Worksation
          ( ) DSK    Desktop

                 < OK >
```

Step5. Disk Partitioning

Once your choice has been made, the installer will ask you to select the disk that you would like it to automatically partition and format for you. If you have more than one disk drive on your computer, the installer will give you the possibility to choose from the list of available disks the one you would like to use to install and run OpenNA Linux.



OpenNA Linux 1.0 Setup

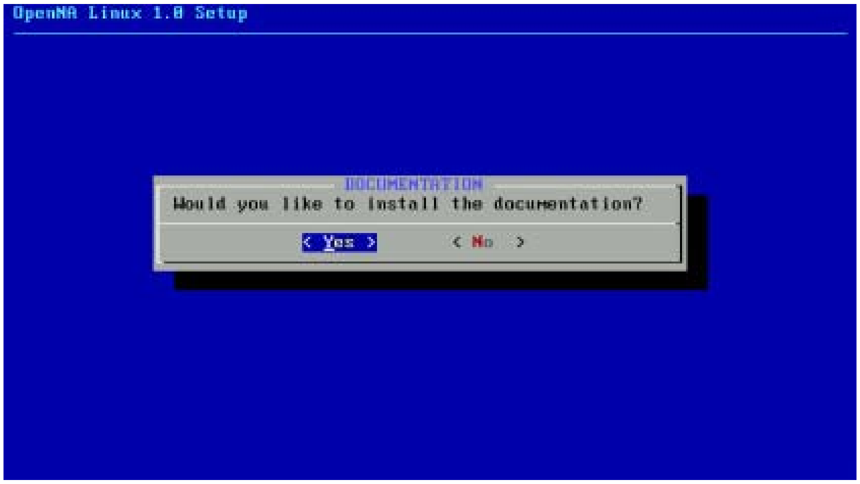
DISK PARTITIONING

Please select the disk that you would like to partition:

1 /dev/sda, size 6142 MB

Step6. Documentation

This option allows you to choose whatever you would like to install manual pages (man page) and related software onto the server. Manual pages (man page) handle very useful technical information when you need some quick help on-line on the operating system but are not always needed to be installed on all servers when we already have it available on other systems. Saying <No> to the documentation option will simply remove all manual pages (man page) and related program (groff, man) from your newly installed OpenNA Linux server.



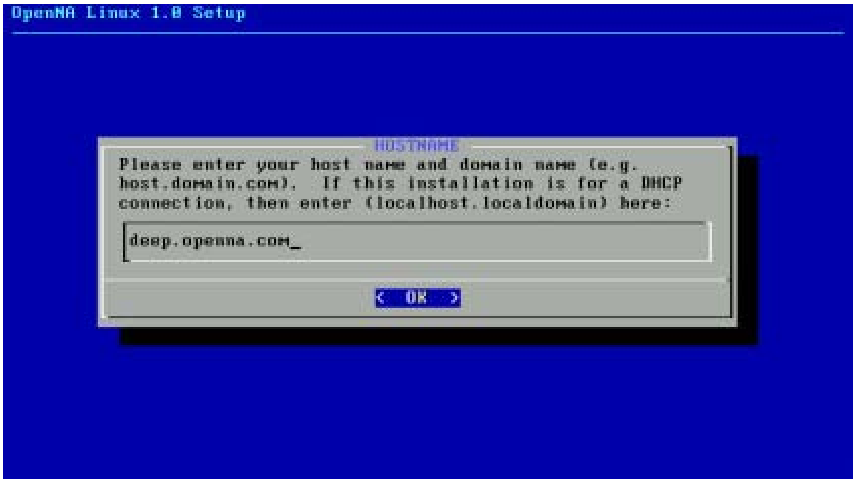
OpenNA Linux 1.0 Setup

DOCUMENTATION

Would you like to install the documentation?

Step7. Network Configuration

The Network Configuration part of the installation will ask you to enter some very important information about your networking environment. It is important to know what you do here or some programs will certainly fail to work as expected. The first information is related to your Host Name and Domain Name. You should enter the **F**ully **Q**ualified **D**omain **N**ame (FQDN) that you would like to assign to the system and not only the Host Name or Domain Name. For example, if you want to give to your newly installed OpenNA Linux server the following FQDN (www.server1.com), then you have to enter "www.server1.com" and not only "www" or just "server1.com".

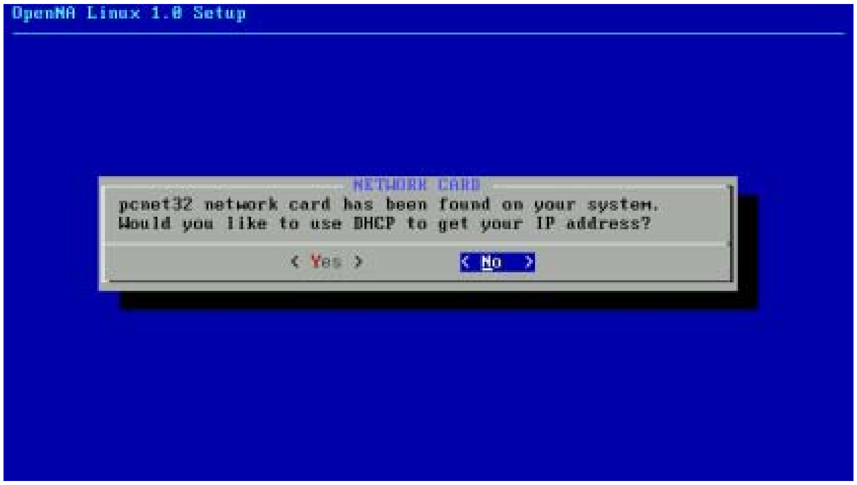


```
Please enter your host name and domain name (e.g.
host.domain.com).  If this installation is for a DHCP
connection, then enter (localhost.localdomain) here:

deep.openna.com_
```

If this installation of OpenNA Linux is for a desktop or for a computer that should get all of its IP information from a remote DHCP server, then you should enter as the FQDN (**localhost.localdomain**) here.

Step8. Network Card

Once your **F**ully **Q**ualified **D**omain **N**ame (FQDN) has been entered, the installer will automatically detect your network card model and will ask you if you would like to use DHCP protocol and service to get the IP address of the system (Dynamic Assigned IP Address) or if you prefer to configure a Static IP address for the server.



```
pcnet32 network card has been found on your system.
Would you like to use DHCP to get your IP address?
```

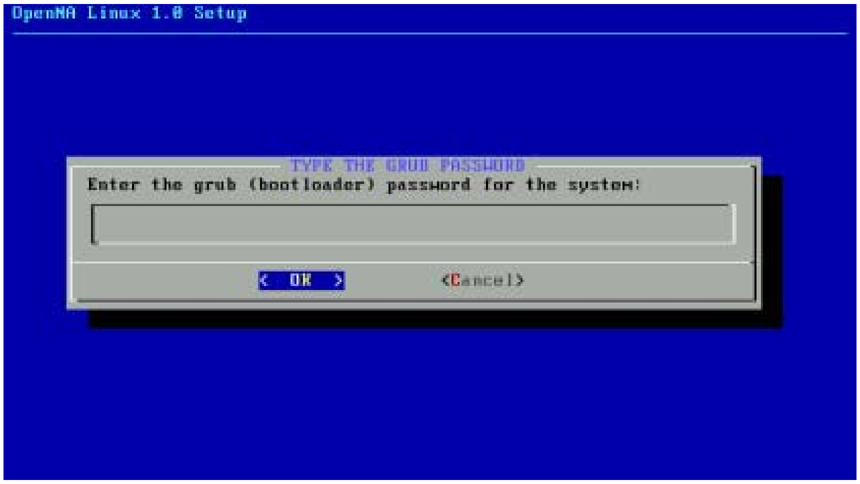
Step9. IP Configuration

If you say <No> to the previous step because you want to assign a Static IP address (mostly used on server), then you will be asked to enter the IP Address, Netmask, Broadcast, Network, Default Gateway, Primary Name Server, Secondary Name Server as well as the Search Option parameters for this server installation. Below is an example about how to enter all of the required network information.

- IP Address = 206.35.76.8
- Netmask = 225.255.255.214
- Broadcast = 206.35.76.31
- Network = 206.35.76.0
- Default Gateway = 206.35.76.1
- Primary Nameserver = 206.35.76.4
- Secondary Nameserver = 206.35.76.5
- Search Option = ns1.domain.com ns2.domain.com domain.com

Step10. Boot Loader Configuration

In order to boot the system without a boot diskette, you usually need to install a boot loader. A boot loader is the first software program that runs when a computer starts. It is responsible for loading and transferring control to the operating system kernel software. The kernel, in turn, initializes the rest of the operating system. GRUB (**GR**and **U**nified **B**ootloader), is the default boot loader used and installed with OpenNA Linux and you have to type a password to protect your system to be booted by unwanted people locally or remotely.



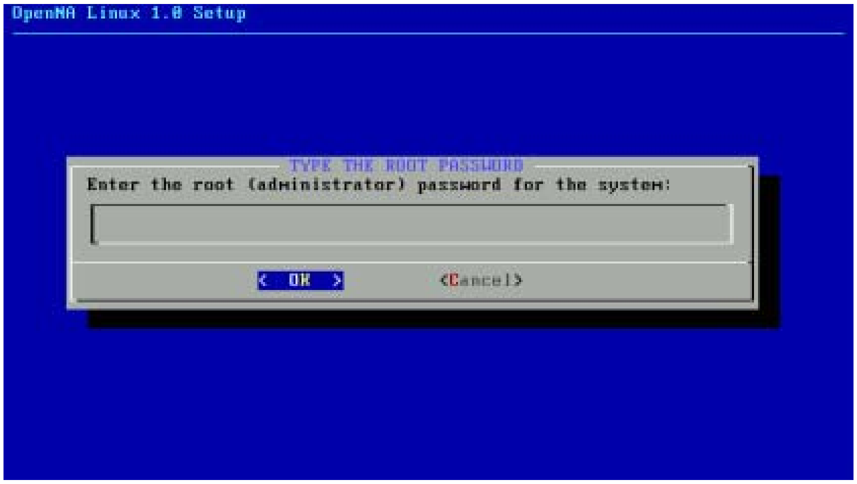
OpenNA Linux 1.0 Setup

TYPE THE GRUB PASSWORD

Enter the grub (bootloader) password for the system:

Step11. The root Password

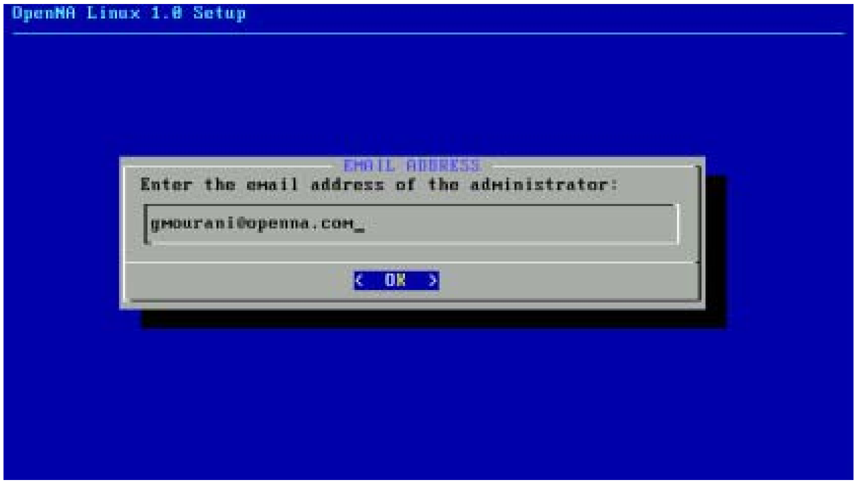
After, you will be asked to enter the "root" password. The "root" password is the password of the administrator of the system which is the most privileged account on a UNIX system. The "root" account has no security restrictions imposed upon it.



Enter the root (administrator) password for the system:

Step12. E-Mail Address Configuration

Different software installed on your server need to know the email address of the administrator of the system to automatically send auto-generated system messages and other important information about the state of your server. Here is where we provide this information by entering the email address of the administrator of this system.



Enter the email address of the administrator:

gmourani@openna.com_

The E-mail address is often the remote email address of the network administrator responsible to manage and maintain all servers into the network.

Optional Configurations

Depending on the type of server you have selected during the install time, different additional configuration parameters should be presented to you during the install stage. These additional configuration steps will help you to better customize and automate installed software. Here we show you all available additional configuration steps. Please note that some may be available for the type of server you have chosen where other may not be available depending again on the type of server you have decided to install.

A. Optional Configuration (Minimal Compiler Packages Install)

The "Minimal Compiler Packages Install" configuration step will ask you whether you would like to install minimal compiler packages on your system. This configuration screen appears only on mandatory installation of the operating system. Compiler packages contain programs and languages used to build software. If you want to be able to compile programs on your mandatory installation of OpenNA Linux, say <Yes>, otherwise say <No>.

B. Optional Configuration (SQL Server Install)

The "SQL Server Install" configuration step will ask you if you want to install an SQL server on your system. This configuration screen appears only on server type where an SQL server could be required by some installed software to properly work. If you manage all your databases via a remote server, you can safety say <No>, otherwise say <Yes> and the MySQL server package will be installed on your system.

C. Optional Configuration (Laptop Install)

The "Laptop Install" configuration step will ask you if you want to install additional packages which are required only on Laptop/Notebook computers. This configuration screen appears for all type of installation because we don't know if you are installing on a Laptop or other computer. Remember that Laptop are often used these days for server use and especially by army. If this installation is made on a Laptop computer, then say <Yes>, otherwise say <No> and the additional packages will not be installed.

D. Optional Configuration (Firewall Install)

The "Firewall Install" configuration step will ask you if you want to install and activate a Firewall on your system. This configuration screen appears for all type of installation and allows you to better customize your installation of OpenNA Linux. Depending of your network design, you may need to run, a firewall on your system. if you want a firewall to be installed, say <Yes>, otherwise say <No>.

E. Optional Configuration (FTP Install)

The "FTP Install" configuration step will ask you whether you would like to install an FTP server on your system. This configuration screen appears only on the Web server installation of the operating system. If you want to provide upload files capability into your Web server installation, say <Yes>, otherwise say <No>.

F. Optional Configuration (RSYNC Install)

The "RSYNC Install" configuration step will ask you if you want to install an RSYNC server on your system. This configuration screen appears only on the FTP server installation of the operating system. Rsync is a program for synchronizing files over network. It is used to provide mirroring capability for FTP servers that need it. If you want to provide mirroring capability into your FTP server installation, say <Yes>, otherwise say <No>.

G. Optional Configuration (Thttpd Install)

The "Thttpd Install" configuration step will ask you if you want to install a Thttpd server on your system. This configuration screen appears only on the FTP server installation of the operating system. Thttpd is a very compact no-frills HTTPD serving daemon that can handle very high loads. While lacking many of the advanced features of Apache, Thttpd operates without forking and is extremely efficient in memory use. Basic support for CGI scripts, authentication, and SSI is provided for. Advanced feature include the ability to throttle traffic. If you want to provide HTTP access with your installed FTP server, say <Yes>, otherwise say <No>.

H. Optional Configuration (WebMail Services Install)

The "WebMail Services Install" configuration step will ask you whether you would like to install a WebMail on your system. This configuration screen appears only for a Mail Server and Virtual Server installation. If you want to provide WebMail services with your Mail server or Virtual server installation, then say <Yes>, otherwise say <No>.

I. Optional Configuration (ODBC Drivers Install)

The "ODBC Drivers Install" configuration step will ask you if you want to install the ODBC drivers on your system. This configuration screen appears only for a Database server installation. If you want to access databases through ODBC or want to develop programs that will access data with ODBC, then say <Yes>, otherwise say <No>.

J. Optional Configuration (DHCPD Install)

The "DHCPD Install" configuration step will ask you if you want to install a DHCPD server on your system. This configuration screen appears only on a Gateway server installation. DHCP allows individual devices on an IP network to get their own network configuration information from a DHCP server. If you want to install a DHCPD server on your Gateway server, say <Yes>, otherwise say <No>.

K. Optional Configuration (Squid Install)

The "Squid Install" configuration step will ask you whether you would like to install a Proxy server on your system. This configuration screen appears only on a Gateway server installation. A proxy server installed into your Gateway server, will let you have more control about security and access right for any internal computers that want to access the Internet. If you want to install a Proxy server, say <Yes>, otherwise say <No>.

L. Optional Configuration (IDS Install)

The "IDS Install" configuration step will ask you if you want to install an **I**ntrusion **D**etection **S**ystem on your server. This configuration screen appears only on a Gateway server installation. Intrusion Detection System (IDS) is a packet sniffer/logger program which can be used as a lightweight Network Intrusion Detection. A web-based client server management system that comes with the installation of the IDS will help you to configure different configuration & signature files. If you want to install an IDS, say <Yes>, otherwise say <No>.

Manual Installation

This part of the installation guide is for those who prefer to use the Manual Installation option of the installer because they want to manually partition and set their disk drives. If you have already decided to install OpenNA Linux through the Automatic Installation method as discussed previously, then you don't need to read this section and can directly go to the next part where we will talk about how to finalize the configuration of your newly installed OpenNA Linux server. A Manual Installation is suitable for those who want to have complete control on the way to install OpenNA Linux on their system. Manual Installation allows you to manually partition and formats your system and it is intended for expert users only.

Partition the disk drives(s)

Partitioning the disk drive(s) is maybe the most complicated part of the manual installation and it's where new users have some problems. If you have more than one disk drive on your system, a list of available disk drive(s) will be listed. Just choose the one you want to run OpenNA Linux on it and press **[OK]**.

cfdisk is the partition software we use to partition the system for OpenNA Linux. All available commands with ***cfdisk*** to create and configure the partitions are:

Bootable
This command allows you to toggle bootable flag of the current partition. At least, one of your created partition should be set as bootable for Linux to boot (this is where you will have to use this command).

Delete
As its name imply, ***delete*** let you delete the current partition. You could have to use it often when you will create, change, and configure partitions for Linux.

Help
The ***help*** command will show you the help screen for ***cfdisk***. You can use this command when you need more information about how to use ***cfdisk*** commands.

Maximize
The ***maximize*** command is used to maximize disk usage of the current partition.

If you decide to use it, then this could make the partition incompatible with DOS, OS/2, ... file system. In most cases, we don't need to use it.

Print
The ***print*** command is used to print partition table to the screen or to a file. We can use it to check if all of our new created partitions are as we want them to be.

Quit

The ***quit*** command is used to exit program without writing any partition tables into the system. You will use this command if you have changed your mind and don't want to manually create partitions on your computer.

Type

The ***type*** command is used to change the file system type. In most cases, you will use it to set the Linux swap partition on the system.

Units

The ***units*** command is used to change units of the partition size display and rotates through MB, sectors and cylinders. This command is useful when you prefer to see human readable information about disk size.

Write

The ***write*** command is what you will use to write partition table to disk when you have finished creating all needed partitions on the system.

A working cfdisk example

Here is an example about how to use the ***cfdisk*** commands to create workable partitions on your system. In our example, the disk drive is an IDE with 20.0 GB in size and we will create four different partitions as follow:

/boot	our **boot** partition which is always needed
/	our **root** partition which is always needed
/var	our **var** partition which is *not* always needed to be defined separately
swap	our **swap** partition which is often needed

Before starting to create the partitions, we should be sure that nothing is already defined on the system. If some partitions are already available, then we should remove them with the **[Delete]** command before starting to create the new partitions.

Now, we start to create our partition. To begin, we use the **[New]** command to define our first partition. Once the **[New]** option is selected, a new sub-menu appears from where you will have the choice between **[Primary]** for a primary partition, or **[Logical]** for a logical partition, and **[Cancel]** to return to the previous menu. Here we select **[Primary]** to create a primary partition and enter for the size of the primary partition 60 for 60 MB. Once the size has been entered, ***cfdisk*** will ask you if you want to add this partition at the beginning of free space or at the end of free space, enter **[Beginning]**. Now you should see you newly created partition on the screen. This will be later labeled as our "*/boot*" partition.

Now we have to create our second partition. To do so, we have to move on the ***<Pri/Log Free Space>*** menu as displayed on the screen with the arrow key of your keyboard and select **[New]** to create and define the second partition. Choose **[Logical]** and press enter. Now, define for example 17000 for 17 GB and select **[Beginning]**. This will create our second partition as logical. We will label it as our "*/*" partition later.

Once this is done, we have to create our third partition. The steps are the same as previously. Move the arrow key to the ***<Pri/Log Free Space>*** menu and select **[New]**, then **[Logical]**, and enter 1900 (1.9 GB) for the disk size, then **[Beginning]**. This will be later labeled as our "*/var*" partition.

Finally, we have to create our last partition to use for the swap. Move the arrow key to the ***<Pri/Log Free Space>*** menu, select **[New]**, **[Logical]**, 1044.61 for the remaining size available, and then select **[Type]** to change the file system to Linux Swap. Choose **82** for the file system type to set the partition as a Linux swap partition (the number 82 is the one associated with Linux Swap).

At this stage, all the example partitions are created and defined but before writing the changes to the disk, we have to set one of the four partitions as *bootable*. This is a requirement, and Linux will not properly install if you forget to do it. So here we choose and set the first partition (*/boot*) as bootable by moving the arrow key to ***<hda1 Primary Linux 57.50>*** and selecting **[Bootable]**. Once this is done, we can write the partitions tables' information to the disk by selecting **[Write]** and typing **yes** to confirm.

Congratulation! Your partition tables have been successfully created; now choose **[Quit]** to exist ***cfdisk***.

Set up your swap partition(s)

Once we have successfully partitioned the disk(s), we have to continue on with the installation, by configuring our swap space. The installer will ask you to do so.

Set up your target partition(s)

In order for Linux to be successfully installed on your system, the program needs to know which file system to use, and how to label the partitions you have created previously. This is done by indicating your target partition to the system. A list of available partitions will be listed on the screen and the installer will prompt you to select a partition from the list to use for your root (*/*) Linux partition. The (*/*) Linux partition is mandatory and you must absolutely indicate which partition should be used by Linux for this purpose. In general (*hda5*, *sda5*) partition is a good choice and this is what we have made during disk partition time.

Once you have selected the partition to use for your root (*/*) Linux partition, the installer will ask you about how you would like to format it. You can choose from the list for a quick format with no bad block checking or for a slow format that checks for bad blocks.

Three different kinds of file systems are available with OpenNA Linux and you have to choose which one you prefer and want to use into your Linux system. You can even use different file system type for each partition but this is not recommended. OpenNA Linux use **ReiserFS** as its default file system because we consider it to be the fastest and most powerful **F**ile **S**ystem (FS) but you are free to choose what you like.

The above step should be made for each additional partition that you have on your system. At least, you should have (*/*), and (*/boot*) for Linux to properly run on your computer.

Configuring OpenNA Linux

This part of the manual explains how to finalize and fine-tune the configuration of your OpenNA Linux system once your installation of the operating system is completed. The following topics are discussed:

- Configuring your OpenNA Linux Web Server
- Configuring your OpenNA Linux FTP Server
- Configuring your OpenNA Linux DNS Server
- Configuring your OpenNA Linux E-Mail Server
- Configuring your OpenNA Linux Database Server
- Configuring your OpenNA Linux Gateway Server
- Configuring your OpenNA Linux Virtual Server
- Configuring your OpenNA Linux Desktop & Workstation

Each server type has its own section on this manual and you have to refer to the section that interest you for more information about the remaining steps to accomplish for your system to be completely functional and operational.

Configuring your OpenNA Linux Web Server

This part of the manual explains how to configure OpenNA Linux for a Web Server environment once your installation of the operating system is completed. A Web Server is a Linux server dedicated to provide Web (HTTPD) services to the Internet or to your network.

The following topics are covered:

- Core Components
- Technical Server Specifications
- Configuring Apache

Core Components

The following major components are included with the OpenNA Linux Web Server installation.

1. Kernel 2.4.22 with Grsecurity patch 1.9.12
2. GLIBC 2.3.2
3. OpenSSL 0.9.7c
4. Perl 5.8.1
5. GIPTables 1.1a
6. Postfix 2.0.16
7. Apache 2.0.47
8. Mod_Security 1.7
9. Mod_DoSevasive 1.8
10. PHP 4.3.3

Technical Server Specifications

The following major technical specifications apply to the OpenNA Linux Web Server installation.

- Used Hard Disk Space: 156 MB
- File System Type: ReiserFS
- Partitions: /, /boot, /usr, /chroot, /var, /var/lib, /tmp, /home

Configuring Apache

Apache is the software used to provide Web service on OpenNA Linux. It has been compiled and configured to give the best security and optimization and to support most of the features available with Apache.

Some of the security and optimization features provided into this version of the web server are:

- A module to provide an intrusion detection and prevention engine for web applications.
- A module to provide evasive action in the event of an HTTP DoS or DDoS attack.
- A module to provide content compression before it is delivered to the client.
- Much more...

Any specific information related to your IP address, domain name, e-mail address of the administrator and the like have been already configured into Apache by the installer at setup-time. The Apache configuration file is ready to be used but you can review it to make changes, adjust setting, add more parameters, etc. Just edit its **httpd.conf** file and check if everything is correctly set for your needs. You can start to transfer your web contents into your new OpenNA Linux Web server.

Configuring your OpenNA Linux FTP Server

This part of the manual explains how to configure OpenNA Linux for a FTP Server environment once your installation of the operating system is completed. A FTP Server is a Linux server dedicated to provide File Transfer Protocol (FTP) Services to the Internet or to your network.

The following topics are covered:

- Core Components
- Technical Server Specifications
- Configuring vsFTPd

Core Components

The following major components are included with the OpenNA Linux FTP Server installation.

1. Kernel 2.4.22 with Grsecurity patch 1.9.12
2. GLIBC 2.3.2
3. OpenSSL 0.9.7c
4. Perl 5.8.1
5. GIPTables 1.1a
6. Postfix 2.0.16
7. vsFTPd 1.2.0

Technical Server Specifications

The following major technical specifications apply to the OpenNA Linux FTP Server installation.

- Used Hard Disk Space: 141 MB
- File System Type: ReiserFS
- Partitions: /, /boot, /usr, /chroot, /var, /tmp, /home

Configuring vsFTPd

vsFTPd is the software used to provide FTP service on OpenNA Linux. The vsFTPd FTP Server configuration as explained in this tutorial is just the starting point to make it quickly work on your server (since every one have different requirement for FTP Server). Here are the minimal steps to make vsFTPd work on your OpenNA Linux.

Step 1

The default setting of vsFTPd with OpenNA Linux is to allow only local users FTP access on the server. Anonymous user is disable and should be enable inside the vsFTPd configuration file (**/etc/vsftpd.conf**). All you need to provide local users FTP access, is to create the local user into your system and provide a password for this user. Here are the steps to accomplish that.

- To create the local FTP user, use the following commands:

```
[root@something /]# useradd -m -s /sbin/nologin john
[root@something /]# passwd john
```

The "**useradd**" command is what we use to create a new user on the system. The "**-m**" option means to create the user home directory (**/home/john**), and the "**-s /sbin/nologin**" option specify that we don't want to allow the user to have a shell access (bash) on the server for security reason. The "**passwd**" command is used to set the password for the user (**john**).

If you prefer for example to have the user home directory created under the **/home/httpd/john** location, then you have to use the "**-m**" option as follow: **useradd -m /home/httpd/john -s /sbin/nologin john**

Configuring your OpenNA Linux DNS Server

This part of the manual explains how to configure OpenNA Linux for a DNS Server environment once your installation of the operating system is completed. A DNS Server is a Linux server dedicated to provide **D**omain **N**ame **S**erver (DNS) Services to the Internet or to your network.

The following topics are covered:

- Core Components
- Technical Server Specifications
- Configuring Bind

Core Components

The following major components are included with the OpenNA Linux DNS Server installation.

1. Kernel 2.4.22 with Grsecurity patch 1.9.12
2. GLIBC 2.3.2
3. OpenSSL 0.9.7c
4. Perl 5.8.1
5. GIPTables 1.1a
6. Postfix 2.0.16
7. BIND 9.2.2

Technical Server Specifications

The following major technical specifications apply to the OpenNA Linux DNS Server installation.

- Used Hard Disk Space: 140 MB
- File System Type: ReiserFS
- Partitions: /, /boot, /usr, /chroot, /var, /tmp, /home

Configuring BIND

BIND is the software used to provide DNS service on OpenNA Linux. With a DNS Server installation, BIND could be configured to run as a Primary DNS Server or as a Secondary DNS Server in a secure chroot jail environment. Configuring your DNS Server as a Primary or Secondary DNS Server requires many steps that we can't list into this manual and for this reason, we recommend you to read tutorial as listed further down into the Additional Documentation section.

Configuring your OpenNA Linux Mail Server

This part of the manual explains how to configure OpenNA Linux for a Mail Server environment once your installation of the operating system is completed. A Mail Server is a Linux server dedicated to provide Mail (SMTP) Services to the Internet or to your network.

The following topics are covered:

- Core Components
- Technical Server Specifications
- Configuring Exim
- Adding a new mail user account into your Mail server
- Configuring the WebMail

Core Components

The following major components are included in the OpenNA Linux Mail Server installation.

1. Kernel 2.4.22 with Grsecurity patch 1.9.12
2. GLIBC 2.3.2
3. OpenSSL 0.9.7c
4. Perl 5.8.1
5. GIPTables 1.1a
6. Exim 4.24
7. SpamAssassin 2.60
8. ClamAV 0.60

Technical Server Specifications

The following major technical specifications apply to the OpenNA Linux Mail Server installation.

- Used Hard Disk Space: 143 MB
- File System Type: ReiserFS
- Partitions: /, /boot, /usr, /chroot, /var, /var/lib, /tmp, /home

Configuring Exim

Exim is the software used to provide Mail service on OpenNA Linux. It has been compiled and configured to give the best security and optimization and to support most of all features available with Exim.

Some security and optimization features provided in this version of the Mail server with OpenNA Linux are:

- Complete Anti-Spam integration and protection into the mailer engine.
- Complete Anti-Virus integration and protection into the mailer engine.
- Complete Anti-Relay integration and protection into the mailer engine.
- Complete mailing list integration into the mailer engine.

Any specific information related to your IP address, domain name, e-mail address of the administrator and the like have been already configured into Exim by the installer at setup-time. The Exim configuration file is ready to be used and you can review it to make changes, adjust setting, add more parameters, etc. Just edit its **exim.conf** file and check if everything is correctly set for your needs. You can start to add new mail user accounts into your new OpenNA Linux Mail server.

With a Mail Server installation, Exim is configured to run as a Central Mail Hub Server and will act as the default Mail Server into your network from where all mail messages are received, forwarded, and sent. It is into this OpenNA Linux Mail Sever that all other type of OpenNA Linux systems will send all of their local system messages and where all of your users will be able to connect to send and receive mails.

Adding a new mail user account into your Mail server
This section of the manual applies to everyone who has installed the OpenNA Linux Mail Server into their system. In this section we deal with how to create and setup a new mail user account with Exim. Here are the minimal customizations needed for mail users to be able to connect to the server to send and receive mails.

Step 1
OpenNA Linux Mail Server is configured by default with Anti-Relay support. Every mail user account should be authenticated before being able to relay any mail messages on the server. For Exim to successfully authenticate the user, it has to verify the user password into its authentication file called **exim.auth**. This mean that every time we create a new mail user account, we have to copy his/her password into the **exim.auth** file for Exim to be able to successfully authenticate the user. Bellow we show you the required steps starting from the creation of a new mail user account to the end where we add his/her password into the Exim authentication file (**exim.auth**).

- To create a new mail user account, use the following commands:

```
[root@something /]# useradd -m -s /sbin/nologin john
[root@something /]# passwd john
```

The "**useradd**" command is what we use to create a new mail user account on the server. The "**-m**" option means to create the mail user account home directory (**/home/john**), and the "**-s /sbin/nologin**" option specify that we don't want to allow this new mail user account to have a shell access (bash) on the Mail Server for security reason. The "**passwd**" command is used to set the password for the mail user account (**john**).

- To add the user password into the Exim authentication file, use the following commands:

```
[root@something /]# cp /etc/shadow /etc/exim/exim.auth
```

The **/etc/shadow** file handles all user accounts on the Linux Mail Server and it is very dangerous to compromise in any way, crackers will have access to all user accounts and will be able to use some password cracking software to get users passwords. Therefore, we have to edit it and remove any lines referring to system accounts like "root", "bin" and users from which a mail account is not provided or required.

To recap, you have to edit the **exim.auth** file and ONLY keep inside this file the lines related to users who have mail account access on the mail server. Any other lines referring, for example, to "root", "nobody", etc should absolutely be removed from the file (**exim.auth**).

- Edit the **exim.auth** file (**vi /etc/mail/exim.auth**) and remove any users from which you don’t want to provide mail on the server.

```
root:$1$jqtyOw5r$Z9fjd7BYqW9ZdYMdA5OPQ1:12062:0:99999:5:::     <-- Remove
bin:x:12062:0:99999:5:::                                       <-- Remove
daemon:x:12062:0:99999:5:::                                    <-- Remove
lp:x:12062:0:99999:5:::                                        <-- Remove
sync:x:12062:0:99999:5:::                                      <-- Remove
shutdown:x:12062:0:99999:5:::                                  <-- Remove
halt:x:12062:0:99999:5:::                                      <-- Remove
mail:x:12062:0:99999:5:::                                      <-- Remove
uucp:x:12062:0:99999:5:::                                      <-- Remove
nobody:x:12062:0:99999:5:::                                    <-- Remove
```

```
vcsa:x:12062:0:99999:5:::                              <-- Remove
rpm:x:12062:0:99999:5:::                               <-- Remove
sshd:!:12062:0:99999:5:::                              <-- Remove
john:$1$99D6.K61$p/j3DljATBEan/ZiQJMzW1:11821::::::    <-- Keep
```

In the above example, we only keep the user “**john**” inside the **exim.auth** file because "**john**” is the only user allowed to have a mail account on the mail server.

Step 2
Finally, restart your Mail Server software with the following command for the changes to take effect.

```
[root@something /]# /etc/init.d/exim restart
```

Configuring the WebMail
This section of the manual applies only if you have chosen to install a WebMail with your OpenNA Linux Mail Server during setup-time. In this section we discuss how to configure the WebMail to run on your server. Here are the minimal customizations needed for the WebMail to work on your system.

Step 1
The main purpose of WebMail software is to provide a Web interface for your mail users to access, read and send their mails. This mean that any mail users who want to get access onto the WebMail interface should already exist on your system. If this is not the case, then please refer to the previous part where we discuss how to create a new mail user account with your OpenNA Linux Mail Server.

- To customize WebMail's configuration for your site, run:

```
[root@something /]# perl /home/httpd/squirrelmail/config/conf.pl
```

Step 2
Once your WebMail software has been configured, you have to inform all your mail users to point their browser at the following URL: http://your.domain.com/members/webmail/ to access the WebMail interface.

Configuring your OpenNA Linux Database Server

This part of the manual explains how to configure OpenNA Linux for a Database Server environment once your installation of the operating system is completed. A Database Server is a Linux server dedicated to provide Database (SQL) Services to the Internet or to your network.

The following topics are covered:

- Core Components
- Technical Server Specifications
- Configuring MySQL
- Additional Documentation

Core Components
The following major components are included in the OpenNA Linux Database Server installation.

1. Kernel 2.4.22with Grsecurity patch 1.9.12
2. GLIBC 2.3.2
3. OpenSSL 0.9.7c
4. Perl 5.8.1
5. GIPTables 1.1a
6. Postfix 2.0.16

7. MySQL 4.0.16
8. unixODBC 2.2.6
9. MyODBC 3.51.06

Technical Server Specifications
The following major technical specifications apply to the OpenNA Linux Database Server installation.

- Used Hard Disk Space: 166 MB
- File System Type: ReiserFS
- Partitions: /, /boot, /usr, /chroot, /var, /var/lib, /tmp, /home

Configuring MySQL
MySQL is the software used to provide Database service on OpenNA Linux. Other types of Database software are available with OpenNA Linux, you can use PostGreSQL or OpenLDAP if you want, just check the OpenNA Linux CD-ROM for the RPMS packages.

The MySQL Database Server configuration as explained in this tutorial is just a starting point to make it quickly work on your server. Every one has different requirements for Database Server and we highly recommend reading other article about MySQL. Here are the minimal steps to make MySQL work on your OpenNA Linux server.

Step 1
For security reasons, it’s important to assign a password to the MySQL root user, since by default after the installation of the SQL server, the initial root password is empty and allows anyone to connect with this name and therefore do anything to the database.

- To specify a password for the MySQL root user, perform the following actions:

```
[root@something /]# mysql -u root mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1 to server version: 4.0.14

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> SET PASSWORD FOR root=PASSWORD('mypasswd');
Query OK, 0 rows affected (0.00 sec)
mysql> \q
Bye
```

The value '**mypasswd**' as shown above is where you put the password you want to assign to the MySQL root user (this is the only value you must change in the above command). Once the root password has been set you must, in the future, supply this password to be able to connect as root to the SQL database.

Configuring your OpenNA Linux Gateway Server

This part of the manual explains how to configure OpenNA Linux for a Gateway Server environment again once your installation of the operating system is completed. A Gateway Server is a Linux server dedicated to provide Gateway (DHCP/PROXY) Services to the Internet or to your network.

The following topics are covered:

- Core Components
- Technical Server Specifications
- Configuring DHCP
- Configuring Squid
- Configuring Apache
- Configuring IDS

Core Components

The following major components are included in the OpenNA Linux Gateway Server installation.

1. Kernel 2.4.22 with Grsecurity patch 1.9.12
2. GLIBC 2.3.2
3. OpenSSL 0.9.7c
4. Perl 5.8.1
5. GIPTables 1.1a
6. Postfix 2.0.16
7. DHCP 3.0.pl2
8. Squid 2.5.STABLE4
9. Apache 2.0.47

Technical Server Specifications

The following major technical specifications apply to the OpenNA Linux Gateway Server installation.

- Used Hard Disk Space: 141 MB
- File System Type: ReiserFS
- Partitions: /, /boot, /usr, /chroot, /var, /var/spool, /tmp, /home

Configuring DHCP

DHCP is one of the software used to provide Gateway service on OpenNA Linux. It has been compiled and configured to give the best security and optimization.

Some security and optimization features provided into this version of the DHCP server are:

- Run in a chrooted environment for optimum security.
- Capable to handle up to 4096 client connections.
- Much more...

Every specific information related to your IP address, domain name, netmask, broadcast, e-mail address of the administrator and the like, have been already configured into DHCP by the installer at setup-time. The DHCP configuration file is ready to be used but you can review it to make changes, adjust setting, add more parameters, etc. Just edit its **dhcpd.conf** file and check if everything is correctly set for your needs.

Configuring Squid

Squid is another software used to provide Gateway service on OpenNA Linux, it is the proxy server used on the Gateway server and has been compiled and configured to give the best security and optimization.

All specific information related to your IP address, domain name, e-mail address of the administrator and the like have been already configured into Squid by the installer at setup-time. The Squid configuration file is ready to be used but you can review it to make changes, adjust setting, add more parameters, etc. Just edit its **squid.conf** file and check if everything is correctly set for your needs.

Configuring Apache

Apache is the software used to provide Web service on OpenNA Linux. It has been compiled and configured to give the best security and optimization and to support most of the features available with Apache.

Some security and optimization features provided into this version of the web server with OpenNA Linux are:

- A module to provide an intrusion detection and prevention engine for web applications.
- A module to provide evasive action in the event of an HTTP DoS or DDoS attack or brute force attack.
- A module to provide content compression before it is delivered to the client.
- Much more...

Any specific information related to your IP address, domain name, e-mail address of the administrator and the like has been already configured into Apache by the installer at setup-time. The Apache configuration file is ready to be used and you can review it for changes, adjust setting, add more parameters, etc. Just edit its **httpd.conf** file and check if everything is correctly set for your needs.

Configuring IDS

This section of the manual applies only if you have chosen to install an Intrusion Detection System with your OpenNA Linux Gateway server during setup-time. In this section we discuss how to configure the Intrusion Detection System to run on your server as well as how to create the required databases and tables needed by the software to properly run onto your server. **I**ntrusion **D**etection **S**ystem (IDS) is a packet sniffer/logger program which can be used as a Lightweight Network Intrusion Detection. A web-based client-server management system will help you configure different configuration & signature files.

Step 1

If this is a fresh installation of OpenNA Linux, then your MySQL server doesn't have the MySQL 'root' password defined, and this will be the first step that we need to do for security reason or anybody will be able to access your databases.

- To create a password for the MySQL 'root' user, use the following commands:

```
[root@something /]# mysql -u root mysql
mysql> SET PASSWORD FOR root=PASSWORD('myrootpasswd');
mysql> \q
```

In the above example, the 'root' user password has been set to "**myrootpasswd**".

Step 2
Now that the MySQL 'root' password has been created, we can start to create the Snort database, username and password. To do so, we connect again to the MySQL database with the MySQL 'root' password. Snort is one of the required components installed into your system if you have chosen to run an Intrusion Detection System.

- To create the Snort user and its database, use the following commands:
  ```
  [root@something /]# mysql -u root mysql -p
  mysql> create database snort;
  mysql> grant all on snort.* to stuser@localhost identified by 'stpasswd';
  mysql> \q
  ```

The above commands will create the Snort database called "**snort**", with Snort username set to "**stuser**" and Snort password set to "**stpasswd**".

Step 3
Once the Snort database has been created into the MySQL server, we have to populate it with the required Snort tables. The script file containing all Snort tables that should be created inside the "**snort**" database is located into the **/root** directory of your server and it's called "**create_mysql**". Move into this directory before using the commands bellow.

- To create the snort tables, use the following commands:
  ```
  [root@something /]# cd /root
  [root@something root]# mysql -u stuser -p snort < create_mysql
  ```

The "**-u**" option is used to define the Snort username which is in our example "**stuser**", and the "**-p**" option will ask you to enter the Snort password (**stpasswd**). The "**snort**" name that comes after the "**-p**" option informs MySQL command that we want to access the "**snort**" database.

Step 4
Now, we have to edit the **/chroot/snort/etc/snort/snort.conf** file and change the lines related to MySQL database for the username and password defined into the MySQL server.

- Edit the **snort.conf** file (**vi /chroot/snort/etc/snort/snort.conf**) and change the following lines:
  ```
  output database: log, mysql, user=myusername dbname=snort host=localhost
  password=mypassword
  ```
  To read:
  ```
  output database: log, mysql, user=stuser dbname=snort host=localhost password=stpasswd
  ```

  ```
  output database: alert, mysql, user=myusername dbname=snort host=localhost
  password=mypassword
  ```
  To read:
  ```
  output database: alert, mysql, user=stuser dbname=snort host=localhost password=stpasswd
  ```

In the above example, we set the username and password of Snort to the one defined previously into the MySQL database server. That's all you need to do for Snort.

Step 5
Acid which is one of the component of the Intrusion Detection System software completely depend on Snort database to property run. Here we have to edit the Acid configuration file and define some parameters to inform Acid where it should find and connect to the Snort database. This mean that Snort database is already created with proper username and password as explained further up.

- Edit the **acid_conf.php** file (**vi /home/httpd/htdocs/admin/acid/acid_conf.php**) and change:

  ```
  $alert_dbname   = "snort_log";
  $alert_host     = "localhost";
  $alert_port     = "";
  $alert_user     = "root";
  $alert_password = "mypassword";
  ```

  To read:

  ```
  $alert_dbname   = "snort";
  $alert_host     = "localhost";
  $alert_port     = "";
  $alert_user     = "stuser";
  $alert_password = "stpasswd";
  ```

Here, "**snort**" is the Snort database name we have created further up in this part of the manual, "**stuser**" is the Snort username and "**stpasswd**" is the Snort password.

Step 6
Now, we have to create the required Acid tables into the Snort database. The script file containing all Acid tables definition that should be created inside the "**snort**" database is located in the **/root** directory on your server and it's called "**create_acid_tbls_mysql.sql**". Move into this directory before using the commands bellow.

- To create the Acid tables, use the following commands:

  ```
  [root@something /]# cd /root
  [root@something root]# mysql -u stuser -p snort < create_acid_tbls_mysql.sql
  ```

The "**-u**" option is used to define the Snort username which is in our example "**stuser**", and the "**-p**" option will ask you to enter the Snort password (**stpasswd**). The "**snort**" name that comes after the "**-p**" option informs the MySQL command that we want to access the "**snort**" database. If you don't know what is the Snort username, password, database, etc, then read the part related to Snort further up for more information.

Step 7
The part related to Acid is now completed and we have to go to the SnortCenter part where we have to create a database for SnortCenter which is another component required by the Intrusion Detection System software to work.

- To create the database for SnortCenter, use the following commands:

```
[root@something /]# cd /root
[root@something root]# mysql -u root mysql -p
mysql> create database snortcenter;
mysql> grant all on snortcenter.* to stcenter@localhost identified by 'stcpasswd';
mysql> \q
```

From the above command, you can see that we have set the SnortCenter username to "**stcenter**" and its password to "**stcpasswd**". The database is "**snortcenter**". Please, note that the username and password for SnortCenter database should be different from the one used for the Snort database.

Step 8
Now, edit the **/home/httpd/htdocs/admin/snortcenter/config.php** configuration file of SnortCenter and change:

```
$DB_user     = "root";
$DB_password = "";
```

To read:

```
$DB_user     = "stcenter";
DB_password = "stcpasswd";
```

```
$send_mail = 0
$mail['host'] = '';
$mail['port'] = 25;
$webmaster_email = "";
```

To read:

```
$send_mail = 1
$mail['host'] = 'smtp.domain.com';
$mail['port'] = 25;
$webmaster_email = "gmourani@domain.com";
```

Where "**stcenter**" and "**stcpasswd**" is the SnortCenter username and password created in Step 7. The "**smtp.domain.com**" value corresponds to your mail server domain name and "**gmourani@domain.com**" to the email address of the person who should receive SnortCenter information.

Step 9
Finally, we have to configure the Snort Sensor Agent. To do so, just run the following commands and answer the questions, then use the username and password that you have entered during configuration of the Snort Sensor Agent to access the SnortCenter web interface.

- To run the Snort Sensor Agent, use the following commands:

  ```
  [root@something root]# cd /home/httpd/htdocs/admin/snortcenter/sensor
  [root@something sensor]# ./setup.sh
  ```

That's all you need to do. Now you can reboot your server or manually restart **Snort** and start the newly installed **sensor** initialization script file before connecting to the Acid and SnortCenter web interface.

- To manually restart the Snort daemon, use the following command:

  ```
  [root@something /]# /etc/init.d/snort restart
  ```

- To manually start the sensor daemon, use the following command:

  ```
  [root@something /]# /etc/init.d/sensor restart
  ```

Step 10
Connect to the Acid and SnortCenter web interface and see if everything is OK. Just point your browser to the following URL: http://your.domain.com/admin/acid/

⚠The first time that you Login to the Management Console of SnortCenter, you have to use login name: **admin** & password: **change**. Once logged, you can change it through the web interface of the software.

Configuring your OpenNA Linux Virtual Server

This part of the manual explains how to configure OpenNA Linux for a Virtual Server environment once your installation of the operating system is completed. A Virtual Server is a Linux server dedicated to provide Virtual Services including HTTPD, SMTP, SQL, and FTP to the Internet or to your network.

The following topics are covered:

- Core Components
- Technical Server Specifications
- Configuring Apache
- Configuring vsFTPd
- Configuring Exim
- Adding a new mail user account into your Virtual Mail Server
- Configuring the WebMail
- Configuring MySQL
- Additional Documentation

Core Components
The following major components are included in the OpenNA Linux Virtual Server installation.

1. Kernel 2.4.22 with Grsecurity patch 1.9.12
2. GLIBC 2.3.2
3. OpenSSL 0.9.7c
4. Perl 5.8.1
5. GIPTables 1.1a

6. Apache 2.0.47
7. Mod_Security 1.7
8. Mod_DoSevasive 1.8
9. vsFTPd 1.2.0
10. Exim 4.24
11. MySQL 4.0.16

Technical Server Specifications
The following major technical specifications apply to the OpenNA Linux Virtual Server installation.

- Used Hard Disk Space: 156 MB
- File System Type: ReiserFS
- Partitions: /, /boot, /usr, /chroot, /var, /var/lib, /tmp, /home

Configuring Apache
Apache is the software used to provide Virtual Web service on OpenNA Linux. It has been compiled and configured to give the best security and optimization and to support most of the features available with Apache.

Some security and optimization features provided in this version of the web server are:

- A module to provide an intrusion detection and prevention engine for web applications.
- A module to provide evasive action in the event of an HTTP DoS or DDoS attack or brute force attack.
- A module to provide content compression before it is delivered to the client.
- Much more...

Any specific information related to your IP address, domain name, e-mail address of the administrator and the like has been already configured into Apache by the installer at setup-time. The Apache configuration file is ready to be used to provide default web service for one domain but for virtual web services use, you will have to edit its **httpd.conf** file and add any virtual web hosting. Here we show you an example on how to add a new virtual web hosting into your OpenNA Linux Virtual server with Apache.

Step 1
In this example, we will suppose that you have a new customer with domain name set to "www.site1.com" and virtual IP address set to "208.34.67.89" to add into your Virtual Web Server.

- Edit the **httpd.conf** file (**vi /etc/httpd/conf/httpd.conf**) and add under "Section 3: Virtual Hosts":

```
NameVirtualHost 208.34.67.89:80

<VirtualHost 208.34.67.89:80>
ServerAdmin customer1@site1.com
ServerName site1.com
ServerAlias www.site1.com
DocumentRoot "/home/httpd/site1"

<Directory "/home/httpd/site1">
 <IfModule mod_deflate.c>
   SetOutputFilter DEFLATE
   BrowserMatch ^Mozilla/4 gzip-only-text/html
   BrowserMatch ^Mozilla/4\.0[678] no-gzip
   BrowserMatch \bMSIE !no-gzip !gzip-only-text/html
```

```
  SetEnvIfNoCase Request_URI \
    \.(?:gif|jpe?g|png)$ no-gzip dont-vary
 </IfModule>
  Options Indexes MultiViews
  AllowOverride None
  Order allow,deny
  Allow from all
</Directory>

ErrorLog /var/log/httpd/error_site1_log
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-agent}i\""
TransferLog /var/log/httpd/access_site1_log
</VirtualHost>
```

Where "**208.34.67.89**" is the IP address of the virtual web site, "**customer1@site1.com**" the e-mail address of the webmaster for this virtual web site, "**site1.com**" the domain name of this virtual site, "**www.site1.com**" the fully qualified domain name of this virtual site, "**/home/httpd/site1**" the documentroot directory of the virtual site where customer1 will add all of his/her web content, "**/var/log/httpd/error_site1_log**" the name and location of the log file for this virtual site that will handle all error logs, and "**/var/log/httpd/access_site1_log**" the name and location of the log file for this virtual site that will handle all access logs.

Step 2
Now, you have to restart your Web Server with the following command for the changes to take effect:

```
[root@something /]# /etc/init.d/httpd restart
```

Configuring vsFTPd
vsFTPd is the software used to provide Virtual FTP service on OpenNA Linux. On a virtual server, it is vital to provide FTP access for your customers to let them upload their web contents into the system. Different approaches exist and we show you here an example that will let you do it with grace.

Step 1
The first step is to create the virtual FTP user in your system and provide a password for this virtual FTP user. Here are the steps to do according to the previous virtual web site example that we have configured into Apache.

- To create the virtual FTP user, use the following commands:

```
[root@something /]# useradd -m /home/httpd/site1 -s /sbin/nologin customer1
[root@something /]# passwd customer1
```

The "**useradd**" command is what we use to create a new FTP user on the system. The "**-m /home/httpd/site1**" option means to create the user home directory under (**/home/httpd/site1**) since this is where we have located the documentroot directory for this virtual web site previously, and the "**-s /sbin/nologin**" option specifies that we don't want to allow the virtual FTP user to have a shell access (bash) on the server for security reason. The "**passwd**" command is used to set the password for the virtual FTP user (**customer1**).

Configuring Exim
We use Exim software to provide Virtual Mail service on OpenNA Linux. It has been compiled and configured to give the best security and optimization.

Some security and optimization features provided in this version of the Mail server are:

- Complete Anti-Spam integration and protection into the mailer engine.
- Complete Anti-Virus integration and protection into the mailer engine.
- Complete Anti-Relay integration and protection into the mailer engine.
- Complete mailing list integration into the mailer engine.
- Complete Virtual Mail integration into the mailer engine.
- Much more...

Any specific information related to your IP address, domain name, e-mail address of the administrator and the like has been already configured into Exim by the installer at setup-time. The Exim configuration file is ready to be used for one domain name but for a virtual mail server, you will have to edit its **virtualdomains**, **localdomains**, and **relaydomains** files to make the appropriate configuration as explained bellow.

Step 1
The **/etc/exim/virtualdomains** file is used to define virtual aliasing mail accounts to virtual domains hosted on your server. You should use it every time you need to define aliases for virtual mail accounts on your system.

- Edit the **virtualdomains** file (**vi /etc/exim/virtualdomains**) and add:
  webmaster@site1.com: mark

In the above example, we permit any email addressed to "**webmaster@site1.com**" to be redirected to user "**mark**" on this virtual domain name.

Step 2
For every incoming or outgoing virtual connection, Exim looks up the sender's email address in the "**virtualdomains**" file. Because Exim may have to search through thousands of virtual email addresses in the "**virtualdomains**" file, it is a good idea to create a copy of the file in a separate db database file to significantly improve lookup time. A small script file called "**newvirtualdomains**" comes with Exim to achieve this. We have to use it directly from the console each time we make modification to the "**virtualdomains**" file.

- To update modifications made inside the "**virtualdomains**" files, run:
  [root@something /]# **/usr/sbin/newvirtualdomains**

Step 3
Remember that Exim by default does not allow you to relay without proper authentication, this is also true for virtual domains too. You have to be sure that the virtual domain in question is added into the "**relaydomains**" and "**localdomains**" files to be allowed to relay. This is very important or relaying will be denied.

- Edit your **relaydomains** file (**vi /etc/exim/relaydomains**) and add:
  site1.com

- Edit your **localdomains** file (**vi /etc/exim/localdomains**) and add:
  site1.com

Step 4
Now we have to restart the Exim daemon for the changes to take effect.

- To restart Exim, use the following command:
  ```
  [root@something /]# /etc/init.d/exim restart
  ```

Adding a new mail user account into your Virtual Mail Server
In this section we cover how to create and setup a new mail user account with Exim. Here are the minimal customizations needed for the virtual mail user account to be able to connect, send and receive mails.

- To create a new virtual mail user account, use the following commands:
  ```
  [root@something /]# useradd -m -s /sbin/nologin mark
  [root@something /]# passwd mark
  ```

The "**useradd**" command is what we use to create a new virtual mail user account on the system. The "**-m**" option means to create the virtual mail user account home directory under (**/home/mark**), and the "**-s /sbin/nologin**" option specifies that we don't want to allow this new virtual mail user account to have a shell access (bash) on the Virtual Mail Server for security reason. The "**passwd**" command is used to set the password for the virtual mail user account (**mark**).

- To add the user password into the Exim authentication file, use the following commands:
  ```
  [root@something /]# cp /etc/shadow /etc/exim/exim.auth
  ```

The **/etc/shadow** file handles all user accounts on the Linux Virtual Mail Server and it is very dangerous to compromise in any way, crackers will have access to all user accounts and will be able to use some password cracking software to get users passwords. Therefore, we have to edit it and remove any lines referring to the system accounts like "root", "bin" and users to whom a virtual mail account is not provided or required.

To recap, you have to edit the **exim.auth** file and ONLY keep inside this file the lines related to users who have mail account access on the server. Any other lines referring, for example, to "root", "nobody", etc should absolutely be removed.

- Edit **exim.auth** file (**vi /etc/mail/exim.auth**) and remove any users from which you don't want to provide virtual mail access on the server:

```
bin:x:12062:0:99999:5:::                                <-- Remove
daemon:x:12062:0:99999:5:::                             <-- Remove
lp:x:12062:0:99999:5:::                                 <-- Remove
sync:x:12062:0:99999:5:::                                       <-- Remove
shutdown:x:12062:0:99999:5:::                           <-- Remove
halt:x:12062:0:99999:5:::                                       <-- Remove
mail:x:12062:0:99999:5:::                                       <-- Remove
uucp:x:12062:0:99999:5:::                                       <-- Remove
nobody:x:12062:0:99999:5:::                             <-- Remove
vcsa:x:12062:0:99999:5:::                                       <-- Remove
rpm:x:12062:0:99999:5:::                                        <-- Remove
sshd:!:12062:0:99999:5:::                                       <-- Remove
john:$1$99D6.K61$p/j3DljATBEan/ZiQJMzW1:11821::::::             <-- Keep
```

In the above example, we *only* keep the virtual user account "**mark**" inside the **exim.auth** file because "**mark**" is the only virtual user allowed to have a virtual mail account on the virtual server.

Step 2
Finally, restart your Mail Server software with the following command for the changes to take effect:

```
[root@something /]# /etc/init.d/exim restart
```

Configuring the WebMail
This section of the manual applies only if you have chosen to install a WebMail with your OpenNA Linux Virtual Mail Server during setup-time. In this section we cover how to configure the WebMail to run on your server. Here are the minimal customizations needed for the WebMail to work properly on your system.

Step 1
The main purpose of WebMail software is to provide a Web interface for your virtual mail users to access, read and send their mails. This mean that all virtual mail users who want to have access to the WebMail interface of the server should already exist on your system. If this is not the case, then please refer to the previous part where we discuss how to create a new virtual mail user account with your OpenNA Linux Virtual Mail Server.

- To customize WebMail's configuration for your site, run:

```
[root@something /]# perl /home/httpd/squirrelmail/config/conf.pl
```

Step 2
Once your WebMail software has been configured, you have to inform all your virtual mail users to point their browser at the following URL: http://your.domain.com/members/webmail/ to access the WebMail interface.

Configuring MySQL
MySQL is the software used to provide Database service on OpenNA Linux. Other types of Database software are available with OpenNA Linux, you can use PostGreSQL or OpenLDAP if you want, just check the OpenNA Linux CD-ROM for the RPMS packages.

The MySQL Database Server configuration as explained in this tutorial is just a starting point to make it quickly work on your server. Every one has different requirement for Database Server and we highly recommend you to read other article about MySQL. Here are the minimal steps to make MySQL work on your OpenNA Linux server.

Step 1
For security reasons, it’s important to assign a password to the MySQL root user, since by default after the installation of the SQL server, the initial root password is empty and allows anyone to connect with this name and therefore do anything to the database.

- To specify a password for the MySQL root user, perform the following actions:

```
[root@something /]# mysql -u root mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1 to server version: 4.0.16

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> SET PASSWORD FOR root=PASSWORD('mypasswd');
Query OK, 0 rows affected (0.00 sec)
mysql> \q
Bye
```

The value '**mypasswd**' as shown above is where you put the password you want to assign to the MySQL root user (this is the only value you must change in the above command). Once the root password has been set you must, in the future, supply this password to be able to connect as root to the SQL database.

Configuring your OpenNA Linux Desktop

This part of the manual explains how to configure OpenNA Linux for a Desktop environment once your installation of the operating system is completed.

The following topics are covered:

- Core Components
- Technical Server Specifications
- Configuring XFree86

Core Components

The following major components are included in the OpenNA Linux Workstation installation.

1. Kernel 2.4.22
2. GLIBC 2.3.2
3. OpenSSL 0.9.7c
4. Perl 5.8.1
5. GIPTables 1.1a
6. Postfix 2.0.16
7. XFree86 4.3.0
8. Gnome 2.4.1

Technical Server Specifications

The following major technical specifications apply to the OpenNA Linux Workstation installation.

- Used Hard Disk Space: 517 MB
- File System Type: ReiserFS
- Partitions: /, /boot, /home

Configuring XFree86

XFree86 is the X server software used to provide **G**raphical **U**ser **I**nterface (GUI) on OpenNA Linux. The XFree86 configuration as explained in this tutorial is just a starting point to make it quickly work on your system because everyone has different monitor, graphical card, mouse, and sound card. We highly recommend you to read other article about XFree86. Here are the minimal steps to make XFree86 work properly on your OpenNA Linux system.

A default XFree86 configuration file suitable for your system and hardware has been automatically generated during setup-time. This configuration file should work since in most case, X successfully detect and setup all peripheries but this is not 100% perfect and sometime you have to manually edit the file and adjust it (especially on Laptop computers).

Step 1

Our first step will be to test and see if everything is running and looks as you want it. Most of the time, you will need/want to adjust the video card resolution and frequency. You have to run the XFree86 server with Gnome (as super-user root) and see if the default resolution is as you want it.

- To start X with Gnome as super-user "root", use the following command:

```
[root@something root]# startx
```

Don't use the "**startx**" command as regular user to start the GUI. The "**startx**" command works only with the super-user "**root**" and will fail to load for regular users. This is a security measure and for regular users, you have to use the **/usr/bin/gdm** command to start the GUI.

Step 2
If the default mode resolution doesn't satisfy your need, then edit the **XF86Config** file and made the appropriated changes into the "**Monitor**" section of the file.

- To edit the **XF86Config** file, use the following command:

```
[root@something /]# vi /etc/X11/XF86Config

Section "Monitor"
      Identifier   "Monitor0"
      VendorName   "Monitor Vendor"
      ModelName    "Monitor Model"
      HorizSync    31.5 - 90.0             <- To be added
      VertRefresh  59.0 - 75.0             <- To be added
      Option       "DPMS"
EndSection
```

In the above example, we set our horizontal sync line to "**31.5 - 90.0**" and our vertical refresh line to "**59.0 - 75.0**". This is just an example, and your setting will certainly be different from the one show here. Please try to check inside the manual that comes with your monitor if the vertical and horizontal refresh parameters are available, and use them for your setting. If you cannot find any information related to these values with your monitor, then see the Linux Laptop web site (http://www.linux-laptop.net/).

Step 3
It is very dangerous to start and use the **G**raphical **U**ser **I**nterface (GUI) as super-user "**root**" and you have to create a regular user that you will use to start and use the GUI. Here we show you how to create this regular user account.

- To create a regular user account to start and use the GUI, use the following commands:

```
[root@something /]# useradd -m john
[root@something /]# passwd john
```

The "**useradd**" command is what we use to create the regular user account on the system. The "**-m**" option means to create the regular user account home directory under (**/home/john**). The "**passwd**" command is used to set the password for the regular user account (**john**).

Step 4
Once this regular user account is created, and that you know that you GUI is working fine, you have to edit the "**inittab**" file and change the default run level to make you system start in Graphical User Interface at start-up.

- Edit the **inittab** file (**vi /etc/inittab**) and change:
  id:3:initdefault:
  To read:
  id:**5**:initdefault:

- Restart your system with the following command:

```
[root@something /]# reboot
```

Configuring your OpenNA Linux Workstation

This part of the manual explains how to configure OpenNA Linux for a Workstation environment once your installation of the operating system is completed.

The following topics are covered:

- Core Components
- Technical Server Specifications
- Configuration

Core Components

The following major components are included in the OpenNA Linux Development Server installation.

1. Kernel 2.4.22
2. GLIBC 2.3.2
3. GCC 3.3.2
4. OpenSSL 0.9.7c
5. Perl 5.8.1
6. Python 2.3
7. XFree86 4.3.0
8. Gnome 2.4.1

Technical Server Specifications

The following major technical specifications apply to the OpenNA Linux Development Server installation.

- Used Hard Disk Space: 845 MB
- File System Type: ReiserFS
- Partitions: /, /boot, /var

Configuration

With the OpenNA Linux Workstation, we install most of all components required for complete development of Linux software. All RPM packages related to compilation tools, headers, libraries etc, are installed to provide a complete environment to develop software from source code. Please read the part related to "Configuring your OpenNA Linux Desktop" for more information about how to make the **G**raphical **U**ser **I**nterface (GUI) installed with your Workstation to work.

OpenNA How-to

There are a lot of other software packages provided by OpenNA Linux and available into the OpenNA Linux CD-ROM that we don't install by default. You may have the need to install and use them on your system and this is why we provide some additional information about the way to configure them here.

How-to phpMyAdmin

phpMyAdmin is intended to handle the administration of MySQL over the web. Currently it can : create and drop databases, create, copy, drop and alter tables, delete, edit and add fields, execute any SQL-statement, even batch-queries, manage keys on fields, load text files into tables, create and read dumps of tables, export data to CSV value, administer multiple servers and single databases.

This mini How-to about **phpMyAdmin** will help you create the required databases and users who should be able to access the phpMyAdmin web interface installed in your server. In the following example, we will suppose that you have one customer site called "www.site1.com" and that you want to create a database called "site1" for this site with username set to "customer1" and password set to "password1".

Step 1
If this is a fresh installation of OpenNA Linux, then your MySQL server doesn't have the MySQL 'root' password defined, and this is the first step that we need to do for security reason or everybody will be able to access your database.

- To create a password for the MySQL 'root' user, use the following commands:

```
[root@something root]# mysql -u root mysql
mysql> SET PASSWORD FOR root=PASSWORD('myrootpasswd');
mysql> \q
```

In the above example, the 'root' user password has been set to "**myrootpasswd**".

Step 2
Now that the MySQL 'root' password has been created, we can start to create the user database, username and password as discussed previously. To do it, we will connect again to the MySQL database with the MySQL 'root' password.

- To create the new user database, with the required information, use the following commands:

```
[root@something root]# mysql -u root mysql -p
mysql> create database site1;
mysql> grant all on site1.* to customer1@localhost identified by 'password1';
mysql> \q
```

The above commands will create the database called "**site1**", with username set to "**customer1**" and password set to "**password1**".

That's all you need to do. Now you can access, this new user database on-line through phpMyAdmin web interface. Point your browser at http://your.domain.com/admin/phpMyAdmin/ to access the SQL web interface.

How-to AWStats

Advanced Web Statistics (AWStats) is a powerful web server log-file analyzer that shows you all your Web statistics including visitors, pages, hits, hours, and search engines, keywords used to find your site, broken links, robots and more... This mini How-to will help you create the required statistic pages for your customers and sites.

Step 1
The first step will be to customize Awstats's configuration for your site(s). If you have a single site from which you want to provide statistics, then you only need to edit the **awstats.conf** file and customize it for your web site. If you want to provide virtual site for multiple users, then you will need to copy the original **awstats.conf** file and **awstats.cron** file for each additional virtual site from which you want to provide statistics.

- Edit the **awstats.conf** file (**vi /etc/awstats/awstats.conf**), and change:
  SiteDomain   = "domain.com"
  To read:
  SiteDomain   = "**www.site1.com**"

  HostAliases   = "domain.com 1.2.3.4 localhost 127.0.0.1"
  To read:
  HostAliases   = "**www.site1.com 208.37.76.13 localhost 127.0.0.1**"

  SkipHosts="domain.com www.domain.com 127.0.0.1 localhost"
  To read:
  SkipHosts="**site1.com www.site1.com 127.0.0.1 localhost**"

Step 2
Now, point your browser at http://your.domain.com/cgi-bin/awstats/awstats.pl to access the statistics interface.

How-to Bugzilla

Bugzilla is the bug tracking system developed by mozilla.org. Mozilla.org is a group within Netscape that acts as a clearinghouse for Netscape source code.

This mini Howto will help you create the required database, tables and user for **Bugzilla** to run on your server.

Step 1
If this is a fresh installation of OpenNA Linux, then your MySQL server doesn't have the MySQL 'root' password defined, and this is the first step that we need to do for security reason or everybody will be able to access your database.

- To create a password for the MySQL 'root' user, use the following commands:

```
[root@something root]# mysql -u root mysql
mysql> SET PASSWORD FOR root=PASSWORD('myrootpasswd');
mysql> \q
```

In the above example, the 'root' user password has been set to "**myrootpasswd**".

Step 2
Now that the MySQL 'root' password has been created, we can start to create the Bugzilla database, username and password as discussed previously. To do so, we will connect again to the MySQL database with the MySQL 'root' password.

- To create the bugzilla user and its database, use the following commands:

  ```
  [root@something root]# mysql -u root mysql -p
  mysql> create database bugzilla;
  mysql> grant all on bugzilla.* to bguser@localhost identified by 'bgpasswd';
  mysql> \q
  ```

The above commands will create the database called "**bugzilla**", with username set to "**bguser**" and password set to "**bgpasswd**".

Step 3
Next, run the "**checksetup.pl**" script. This script is designed to make sure your MySQL database and other configurations options are consistents with the Bugzilla CGI files. It will make sure Bugzilla files and directories have reasonable permissions, sets up the data directory, and create all the MySQL tables.

- To run the "**checksetup.pl**" script, use the following commands:

  ```
  [root@something root]# cd /home/httpd/bugzilla/
  [root@something bugzilla]# ./checksetup.pl
  ```

Step 4
The first time you run the "**checksetup.pl**" script, it will create a file called **localconfig**. This file contains a variety of settings you may need to adjust including how Bugzilla should connect to the MySQL database. Here are the options that we need to change.

- Edit the **localconfig** file (**vi /home/httpd/bugzilla/localconfig**) and change the following options.

  ```
  $webservergroup = "nobody"
  ```
  To read
  ```
  $webservergroup = "www";
  ```

  ```
  $db_host = "localhost";
  $db_port = 3306;
  $db_name = "bugs";
  $db_user = "bugs"
  ```
  To read
  ```
  $db_host = "localhost";
  $db_port = 3306;
  $db_name = "bugzilla";
  $db_user = "bguser";
  ```

  ```
  $db_pass = ''
  ```
  To read
  ```
  $db_pass = 'bgpasswd';
  ```

In the above example, we set the username and password of Bugzilla to the one defined previously in the MySQL database server.

Step 5
Once you are happy with the settings, re-run **checksetup.pl**. On this second run, it will create the database and an administrator account for which you will be prompted to provide information.

- To re-run the "**checksetup.pl**" script, use the following commands:

```
[root@something root]# cd /home/httpd/bugzilla/
[root@something root]# ./checksetup.pl
```

Step 6
Finally, you have to configure Bugzilla and set all parameters to their appropriate values. The URL to access the Bugzilla page parameters is: **http://your.domain.com/bugzilla/editparams.cgi**

That's all you need to do. Now you can point your browser to the following URL to access your Bugzilla web interface -> **http://your.domain.com/bugzilla/**

How-to Mailman

Mailman is software to help manage email discussion lists, much like Majordomo and Smartmail. Unlike most similar products, Mailman gives each mailing list a web page, and allows users to subscribe, unsubscribe, etc. over the web. Even the list manager can administer his or her list entirely from the web. Mailman also integrates most things people want to do with mailing lists, including archiving, mail news gateways, and so on.

This mini How-to will help you to configure and create your mailing list(s) once the mailman package has been installed on your system. In our example, we will create a new mailing list called "**site1-support**".

Step 1
The first step will be to create the mailman administrator password for the list, this administrator password will be used to access the mailing list administration web interface from where you can make changes to different parameters related to your newly created mailing.

- To create the mailman administrator password, use the following command:

```
[root@something root]# /var/lib/mailman/bin/mmsitepass
New site password:
Again to confirm password:
Password changed.
```

Step2
Once the administrator password for the new mailing list has been created, you have to edit the "**mm_cfg.py**" file and customize mailman's configuration for you site. What you should add inside this file is your domain name.

- Edit the **mm_cfg.py** file (**vi /var/lib/mailman/Mailman/mm_cfg.py**) and change:

```
DEFAULT_URL_HOST   = 'mm_cfg_has_not_been_edited_to_set_host_domains'
To read:
DEFAULT_URL_HOST   = 'www.site1.com'

DEFAULT_EMAIL_HOST = 'mm_cfg_has_not_been_edited_to_set_host_domains'
To read:
DEFAULT_EMAIL_HOST = 'www.site1.com'
```

In the above example "**www.site1.com**" is the domain name of this new mailing list.

Step 3
Now, we have to create our mailing list. To do so, we have to create mailman list at first, then create any additional list that we need. In this example, our mailing list will be called "**site1-support**".

- To create the mailman list, use the following command:

```
[root@something root]# /var/lib/mailman/bin/newlist
Enter the name of the list: mailman
Enter the email of the person running the list: postmaster@your.domain.com
Initial mailman password:
To finish creating your mailing list, you must edit your /etc/aliases (or equivalent) file by adding
the following lines, and possibly running the `newaliases' program:

## mailman mailing list
mailman:             "|/var/lib/mailman/mail/mailman post mailman"
mailman-admin:       "|/var/lib/mailman/mail/mailman admin mailman"
mailman-bounces:     "|/var/lib/mailman/mail/mailman bounces mailman"
mailman-confirm:     "|/var/lib/mailman/mail/mailman confirm mailman"
mailman-join:        "|/var/lib/mailman/mail/mailman join mailman"
mailman-leave:       "|/var/lib/mailman/mail/mailman leave mailman"
mailman-owner:       "|/var/lib/mailman/mail/mailman owner mailman"
mailman-request:     "|/var/lib/mailman/mail/mailman request mailman"
mailman-subscribe:   "|/var/lib/mailman/mail/mailman subscribe mailman"
mailman-unsubscribe: "|/var/lib/mailman/mail/mailman unsubscribe mailman"

Hit enter to notify mailman owner...
```

- To create your mailing list, use the following command:

```
[root@something root]# /var/lib/mailman/bin/newlist
Enter the name of the list: site1-support
Enter the email of the person running the list: mark@www.site1.com
Initial site1-support password:
To finish creating your mailing list, you must edit your /etc/aliases (or equivalent) file by adding
the following lines, and possibly running the `newaliases' program:

## site1-support mailing list
site1-support:             "|/var/lib/mailman/mail/mailman post site1-support"
site1-support-admin:       "|/var/lib/mailman/mail/mailman admin site1-support"
site1-support-bounces:     "|/var/lib/mailman/mail/mailman bounces site1-support"
site1-support-confirm:     "|/var/lib/mailman/mail/mailman confirm site1-support"
site1-support-join:        "|/var/lib/mailman/mail/mailman join site1-support"
site1-support-leave:       "|/var/lib/mailman/mail/mailman leave site1-support"
site1-support-owner:       "|/var/lib/mailman/mail/mailman owner site1-support"
site1-support-request:     "|/var/lib/mailman/mail/mailman request site1-support"
site1-support-subscribe:   "|/var/lib/mailman/mail/mailman subscribe site1-support"
site1-support-unsubscribe: "|/var/lib/mailman/mail/mailman unsubscribe site1-support"

Hit enter to notify site1-support owner...
```

✋ The mailman list is required or you will receive an error message at startup about Site list missing. Don't forget to create the mailman list the first time you install the software. After, you can add any additional list that you want.

Step 4
To finish creating your mailing list, you must edit the **aliases** file of Exim by adding the above lines related to the list you have created and run the "**newaliases**" program to update the file.

- Edit the **aliases** file (**vi /etc/exim/aliases**) and add at the end of the file:

```
## site1-support mailing list
site1-support:              "|/var/lib/mailman/mail/mailman post site1-support"
site1-support-admin:        "|/var/lib/mailman/mail/mailman admin site1-support"
site1-support-bounces:      "|/var/lib/mailman/mail/mailman bounces site1-support"
site1-support-confirm:      "|/var/lib/mailman/mail/mailman confirm site1-support"
site1-support-join:         "|/var/lib/mailman/mail/mailman join site1-support"
site1-support-leave:        "|/var/lib/mailman/mail/mailman leave site1-support"
site1-support-owner:        "|/var/lib/mailman/mail/mailman owner site1-support"
site1-support-request:      "|/var/lib/mailman/mail/mailman request site1-support"
site1-support-subscribe:    "|/var/lib/mailman/mail/mailman subscribe site1-support"
site1-support-unsubscribe:  "|/var/lib/mailman/mail/mailman unsubscribe site1-support"
```

- To update your **aliases** file, use the following command:

```
[root@something root]# /usr/sbin/newaliases
```

Step 5
Now, we have to start the mailman daemon on the server.

- To start mailman daemon, use the following command:

```
[root@something root]# /etc/init.d/mailman start
```

Step 6
You will receive email instructions on how to visit the list you just created.

Tips, FAQs, HOWTOs, and Books

Linux FAQ: (http://www.tldp.org/FAQ/Linux-FAQ/index.html)
The best place to start if you are new to Linux or have general Linux questions.

Linux Installation & Getting Started: (http://www.tldp.org/LDP/gs/gs.html)
New user guide to Linux distributed freely under the GPL at the Linux Documentation Project.

HOWTOs: (http://www.tldp.org/HOWTO/HOWTO-INDEX/howtos.html)
An extensive index of HOWTOs and Mini HOWTOs written by a variety of experienced Linux users and developers. Here you will find reliable information on everything from installing Linux to configuring your machine.

Books: (http://www.openna.com/osCommerce/product_info.php?products_id=29)
The well know book about Linux security and optimization.