PRIMA

Panel for Remote Infrastructure Management Applications

TECHNICAL CASE

VAssure | Virtualization Labs | trRIMS | Offshore-QA | BI | Portals

http://www.vassure.com

PROLOGUE

The Clientele is spread out globally, remote infrastructure management is done through an offshore development center, connectivity is through VPN/VoIP, SSL (Secure Sockets Layer) protocol is to be used for encrypted transmission, LDAP (Lightweight Directory Access Protocol) will be authorization point; Linux OS, Apache, MySQL and Perl-CGI are technologies running on Server. Setup can be customized for Infrastructure management services like web hosting solutions, mail server, access to data in the distributed/remote storage areas and multiple other service areas

PRIMA is a web-based interface for system administration for UNIX. Using any browser that supports tables and forms, you can setup user accounts, Apache, DNS, file sharing and so on.

PRIMA also happens to be a fast and easy to use tool for general UNIX system administration. This document attempts to introduce you to many of the concepts you will need to maintain a UNIX system using PRIMA. While no single volume can address every aspect of UNIX system administration, a real effort has been made to provide both a solid introduction to many important tasks, and a nearly comprehensive reference to a typical UNIX server and its parts.

It is VAssure Team hope that with nothing more than this document, a copy of PRIMA, and the documentation that accompanies your server, you will be able to configure the system to provide the most popular services, create a reasonable security policy, and manage your users and normal system maintenance tasks. Advanced topics are often covered, but we hope that it will not be at the expense of preventing you from seeing the forest for the trees.

PRIMA is developed using open source technologies and is inspired by other web administration tools of open source available in market.

Applicability

PRIMA is one of the unique in the UNIX world, in that it provides a one-to-one graphical interface to nearly every service and action needed to maintain a UNIX system. It is universally accessible, because it only requires a web browser. It can potentially be accessed from anywhere in the world via a network connection. It is simple, concise, and consistent in its presentation across a wide array of differing services, functions, and operating systems. It is predictable, in that it does not modify files unnecessarily or in incompatible ways. Configuration with PRIMA does not preclude configuration via other tools, or via the command line. Equally importantly, PRIMA will not damage files if it doesn't understand a particular option or directive in your existing configuration. If PRIMA does not understand a portion of your configuration, it will simply ignore it, and leave it untouched in the configuration file. PRIMA is also accessible, in the sense that it can be used successfully from nearly any browser. Text mode browsers, small screen displays, and nearly anything else can be accommodated through the appropriate use of themes and numerous configurable display parameters.

PRIMA is an excellent tool for both novice and experienced system administrators. As a tool for novices, it can provide a means of getting involved in system administration in a very visual way. All of the options available are presented in a clear and complete fashion. For new users, seeing the possibilities laid out so plainly can be a very effective teaching tool, as well as a helpful safety net to avoid many common pitfalls. It is possible to explore the possibilities of a system, without wading through obscure man pages (you only need wade through the pages in this book, which are perhaps less obscure).

For experienced admins, the advantages are less obvious but no less real. An administrator cannot possibly remember every option to every system function that he or she must configure and maintain. With PRIMA, an administrator no longer needs to remember complex syntax, or the exact directive needed to accomplish some task. Using PRIMA may not be as quick or flexible for some tasks and some users as the command line, and it should not be viewed as a complete replacement for study of traditional system administration tools and techniques. But it is an excellent helper for getting your job done without having to experiment with weird configuration file syntax.

I often tell people that PRIMA doesn't make being a good system administrator easy, it just makes the problems more visible and the solutions more consistent..

Introduction to PRIMA:

PRIMA is designed to allow the easy addition of new modules without changing any of the existing code. A module can be thought of as something like a Netscape or Photoshop plug-in - it can be written by someone other than the developers of PRIMA,

A module should be written to administer one service or server, such as the Unix password file or the Apache web server. Some complex system functions may even be split over several modules - for example, disk partitioning, mounting disks and disk quota management are 3 separate modules in the standard PRIMA distribution.

Modules can theoretically be written in any language. However, to make use of the PRIMA API Perl version 5.002 or above should be used. A module should be written entirely in Perl, with no C functions or external binary programs. The aim is for modules to be as portable as possible across different Unix systems and CPU types.

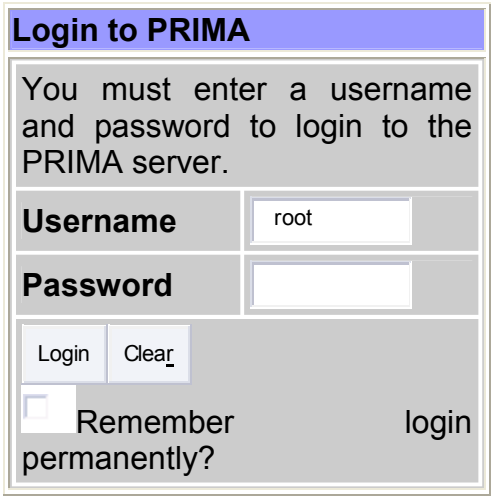
At their simplest, modules are really just directories of CGI programs that PRIMA's web server runs. However, there are certain rules that should be followed to make sure that they work with the PRIMA API, main menu and access control system. Even though you can just stick any existing CGI script into a module directory, this is not a good idea.

Login to PRIMA

Logging into PRIMA is easy. Open a web browser, such as Netscape or Internet Explorer, on any machine that has network access to the server on which you wish to login. Browse to port 10000 on the IP or host name of the server.

PRIMA will then respond with either an authentication window, or an authentication web form, in which you can enter the administrator user name (usually “root” or “admin”) and password. After successful authentication,

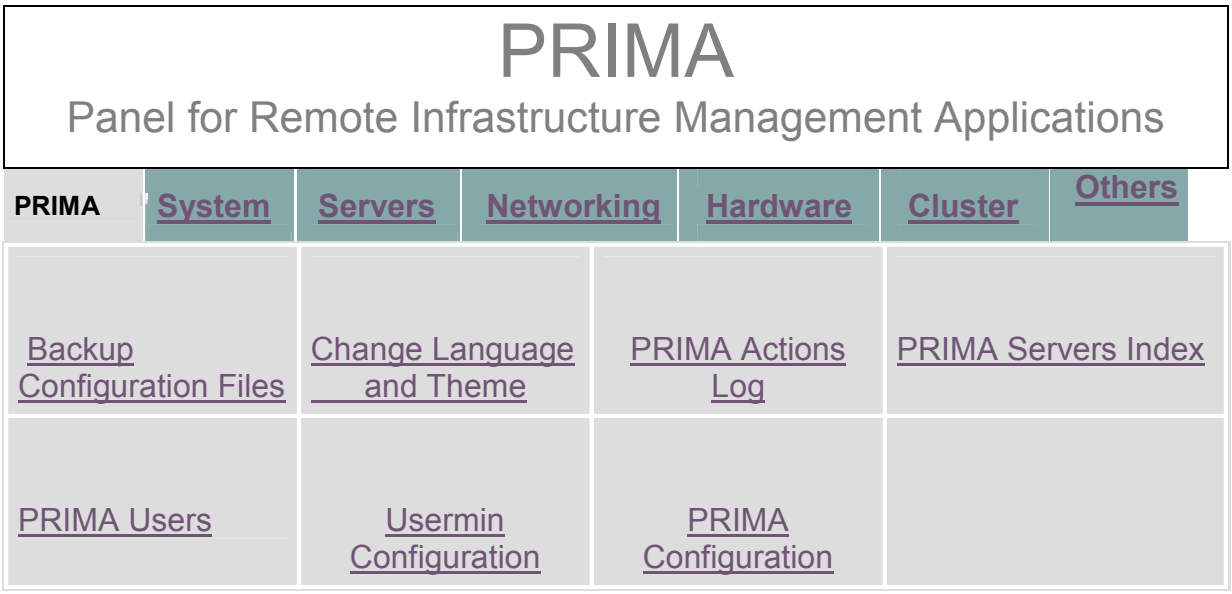
Figure: Session Authentication



A First Look

PRIMA is divided into a number of modules that each allows you to administer a single aspect of your system. Modules exist for most common, and many uncommon, system administration tasks. The standard modules provide a graphical interface for: *Apache, Squid, Bind, NFS, man pages, Sendmail, Postfix, Samba,* and much more. There also exist wide arrays of third party modules that provide even more extensive functionality. This book focuses on the standard modules, but may expand to encompass other modules in time.

Upon first logging in, you'll see a row of tabs and a number of icons as shown in figure. The tabs are labeled **PRIMA**, **System**, **Servers**, **Hardware**, **Cluster**, and **Others**. You may also have, depending on your OS and version, one or two additional tabs. The selected tab when first logging in is always PRIMA. This category is where all of the PRIMA-related configuration details are located.



PRIMA Category

PRIMA provides a number of configurable options, access control features, and flexible action logging that provides you with maximum flexibility and security of the PRIMA server and the various PRIMA system administration modules. These features are accessed through the PRIMA tab on the index page of PRIMA. When you display the **PRIMA** tab, you see icons for **Backup Configuration Files,Change Language and Themes, PRIMA Action Log,PRIMA Servers Index,PRIMA Users**,**Usermin Configuration** and **PRIMA Configuration**. Keep in mind that the modules located under the **PRIMA** tab are for configuring PRIMA *itself*, not the underlying system. So, for example, creating a user in the **PRIMA Users** module will not create a system user, only a PRIMA user. Likewise, the **PRIMA Actions Log** module allows you to search and view the **PRIMA log**, not any system or service log that might exist. We'll get to those kinds of options later. For the moment, we're going to skip over **Usermin Configuration** because **Usermin** receives full coverage in the next chapter.

PRIMA Actions Log

The PRIMA Actions Log page provides access to the PRIMA log. You can configure this log for each module and individual users. This module does not configure the logs, but provides you with a means to search the logs for actions performed by particular logged users, or actions performed in given logged modules. Configuration of PRIMA logging capabilities is covered in the **PRIMA Configuration** section.

With this module it is possible to search for actions by specific users, within specific modules, for a given range of dates, or any combination of those qualifications. For example, if you manage a number of junior system administrators and you'd like to find out if one of them has edited an Apache virtual server configuration in the past week, this module makes those kinds of questions easy to answer (assuming logging to that degree is enabled, of course).

PRIMA Configuration

The **PRIMA Configuration** module allows you to configure most of the important aspects of PRIMA itself, as well as install new modules, upgrade existing modules, and upgrade PRIMA itself. It also provides a means to change the port and address where the PRIMA miniserv.pl web server listens for connections, select different languages, enable or disable SSL encryption, and configure the PRIMA built-in logging features.

IP Access Control

PRIMA has its own web server, called miniserv.pl, which provides a simple IP access control feature. This page allows you to configure this option. You may enter IP networks (such as 192.168.1.0), IP host addresses (such as 192.168.1.79), and host names (such as vassure.). It is wise to limit access to the PRIMA server to just those addresses that are trusted. While PRIMA has no known exploits in versions greater than 0.970, if someone were to obtain your password, this would provide an additional level of protection from unauthorized access. This option configures the accept and deny directives in the miniserv.conf file. The default is to allow any address to access PRIMA.

Port and Address

The PRIMA server will, by default, listen on every active IP address on the system. But if you have multiple addresses and would prefer PRIMA to only listen on one of them, you may use this option. So, for example, if you have one network interface connected directly to your local network and a second network interface connected to the Internet, you could improve security by causing PRIMA to only listen on the local network. In this case, any requests from the Internet at large would be ignored, but it would still be possible to connect from local computers. This can be a very effective first line of defense. After all, if the bad guys can't even talk to the PRIMA server, they certainly can't try anything funny to break into it.

The **Listen on Port** option specifies the network port on which PRIMA will listen. In a standard PRIMA install this will be port 10000, although Caldera installs it on port 1000. Some firewalls may restrict access to ports below 1024, and some may restrict even ports above 1024. If your network has strict proxy restrictions that prevent connecting on port 10000, you may wish to try port 553 or 443 (assuming these ports are not already in use on your PRIMA server for normal SSL service). These ports will nearly always be usable through a proxy, even when using an SSL enabled PRIMA.

As mentioned briefly in the installation chapter, it is possible to alter these configuration settings in the miniserv.conf configuration file in addition to graphical configuration with the PRIMA Configuration module. This may be necessary if a firewall prevents you from accessing port 10000, and you only have console or SSH access to the machine. In this case, editing the port option will alter the port, and the bind directive configures the address on which PRIMA listens. Whenever editing the miniserv.conf file, PRIMA must be restarted for changes to take effect.

Logging

As mentioned earlier, PRIMA provides very flexible logging features. With these features, you can very easily monitor what actions those users with administrator privileges are performing on the server. It is also possible to log actions based on the module where the actions are performed. The option **Log resolved host names** will cause PRIMA to provide a host name rather than just an IP address for the client computer that performed an action. And **Clear logfiles every...hours** causes PRIMA to rotate its own logs and keep them from overfilling the disk with old logs. If long-term logs are needed for security auditing purposes, it may be wise to include the PRIMA log in your normal system backup rotation.

The decisions regarding what to log, whose actions to log, and how long to store those logs, should be carefully considered for your situation. In some cases, a log is unnecessary, while in others it may be required by company policy or useful in addressing the security needs of your environment. If logging is enabled, care should be taken to insure PRIMA will have plenty of disk space in the PRIMA log directory, as some options can lead to quite verbose logging (**Log changes made to files by each action**, for example). Remember that PRIMA action logging has nothing to do with the logging features of other parts of the system. Syslog is configured separately in the **System: System Logs** module, while application specific logging is usually configured within the application module.

Proxy Servers

PRIMA provides several tools that must connect to the Internet to operate correctly. These include the **PRIMA Update** feature, the **Software Packages** module and others. If your local network uses a proxy to access Web or FTP sites on the

Internet, you may configure those settings here. If your proxy requires authentication, the username PRIMA will use to login can also be configured on this page in the **Username for proxy** and **Password for proxy** fields.

User Interface

The PRIMA user interface is configurable in a number of ways. In this module you may configure the colors of your PRIMA pages. The colors are expected to be in standard hex triplets, as used in HTML markup on the Internet. You may also choose to use the standard fonts of your browser to display page titles, rather than the font provided by the theme you are using. Finally, you may configure where on the page PRIMA will display the login name and host name of the server. This page does not configure PRIMA themes, which are configured on their own page, and the changes that can be made here are mild by comparison to the possibilities when using themes. Be aware also that these changes may not take effect when using a theme other than the old standard PRIMA theme. For example, the new MSC.Linux theme overrides all of these options with its own standard values.

PRIMA Modules

As previously mentioned, one of the best things about PRIMA is that it is completely modular. Every server daemon, every system feature, every PRIMA feature, has its own module that connects to the core PRIMA libraries and answers to the PRIMA miniserv.pl web server. Because of the elaborate, but still easily comprehensible, modular framework that PRIMA provides, it is very easy to write full featured modules that integrate seemlessly into PRIMA and your operating system.

Install Module

From this page, you can install new modules, either from a local file, an uploaded file, or a file downloaded from an FTP site or website. PRIMA module packages are simply tar archive files, that contain the complete directory structure of the module.

Clone Module

The **Clone Module** feature provides an impressive amount of flexibility for administrators who must provide limited administration access for several instances of the same software on the same machine. If, for example, you have two different Apache configurations running on your system, you could clone the Apache module to allow different users to access the different Apache configurations.

To clone a module, select the module to clone from the drop-down menu, then enter a new name for the module. To avoid the problem of the new module interfering with the original module, you will want to carefully consider the service being administered by the cloned module. Usually, you will need to set up the new clone

with a wholly separate installation of the service being configured. So, for example, if you have cloned Squid so that you may run two different Squid processes you *must* configure them to use separate configuration files, cache directories, log files, and process IDs. If this precaution is not taken, one or both of the processes will behave erratically or fail to work at all.

Delete Modules

In this section, you may select any modules that you'd like to delete from your PRIMA installation. Beware that using this form will delete the selected modules entirely from the system. If you decide later to use a deleted module, you will have to download the module again and reinstall it. It is usually a better idea to simply remove the module from each users access list (possibly even including root), rather than deleting the module here. However, if disk space is a concern, you can use this to delete all unneeded modules from your system.

Operating System

PRIMA knows how to interact with your system based on configuration files for each module, that are selected based on the operating system configured here. If your system has PRIMA pre-installed, you usually will not need to concern yourself with this. But if you upgrade your system, and the new version moves some configuration files to new locations, updating this may be necessary. On this page you may also set the search path for both programs (like system commands), and for libraries (such as for the password encryption library). Again, these options rarely need to be changed unless you have installed system tools and configuration files in odd locations on your system.

Language

PRIMA supports a large number of languages for titles and module text. This page allows you to choose the language of your PRIMA. New languages are being added regularly. Users of languages that are not supported are encouraged to write a translation and send it to the author of PRIMA. He's always happy to receive new translations, and users are always happy to find that their native language is one that is provided with the distribution.

Index Page Options

This page allows you to configure the layout of the PRIMA index pages. You may choose the number of icons to display per row using the **Number of Columns** field. The **Categorize modules?** Selects whether modules will be grouped under category tabs based on the type of function they perform. The **Default category** is the category that will be displayed when first logging into PRIMA. An alternative header can be used by selecting the **Use alternative header** option, which provides

a somewhat different appearance by placing the host information on the upper right side of the display rather than below the PRIMA title. Finally, selecting **Go direct to module if user only has one?** will cause a user to see *only* the module they have access to, rather than the PRIMA index page when logging in.

Upgrade PRIMA

Using this page, you may upgrade your PRIMA to the latest version automatically from the PRIMA home page, or from a local or uploaded file. This module will use a package management system to perform the update if one is available on your system. If, for example, you have an RPM based system like Caldera, Red Hat, or Mandrake, this feature will upgrade from an RPM package (it even knows how to find the correct package type for your system on the PRIMA homepage!).

Authentication

PRIMA provides some nice features for preventing brute force password cracking attacks on your server, as well as protection against "forgetful users." If your PRIMA server is widely accessible, and provides service to many users, it is probably wise to make use of these features to maximize the security of your system. Security policy in your company may even require usage of some or all of these features.

Password timeouts provide a means to prevent brute force password attacks by limiting the frequency of login attempts by a given user. If enabled, PRIMA will block hosts that have a given number of failed login attempts. The time to block the host is configurable in seconds. PRIMA will expand the delay on continuing failed login attempts from the same host. Logging of blocked logins can also be enabled.

The next option, **Log blocked hosts, logins and authentication failures to syslog** configures PRIMA to log authentication failures and blocked addresses attempting to login to syslog. These logs will usually appear in the secure or auth file in your system log directory.

Session authentication provides a means of logging users out after a specified time of inactivity. This can help prevent unauthorized users from accessing the server by simply using the computer of someone who does have access. This isn't fool-proof, as many browsers now have password management features and authorized users may store their passwords on the local computer, making them accessible to anyone with access to the computer. If security is a concern, you should strongly discourage users from saving login information for the server on their local machine, as well as discouraging leaving open browser sessions when away from their desk or office.

Finally, you may choose to allow logins from users on the same machine where PRIMA is running based on the user name. This feature should only be used for

single user machines, where security is not a major concern. If enabled, anyone with access to the local machine will easily be able to gain root access to your system.

Reassign Modules

As mentioned earlier, PRIMA categorizes modules based on the function they perform, by default. This page provides a simple means for moving modules to new categories if you find the default categorization is confusing to you. Some third party modules, written before the categorization features were added to PRIMA, are mis-categorized into the **Others** category by default, so you may wish to manually move them to their more sensible locations using this module.

Edit Categories

Instead of moving modules within existing categories it may be most sensible to create a *new* category for your favorite modules, or for custom modules written just for your organization. This page allows you to create new module categories, as well as rename or relabel old ones.

Trusted Referrers

Because PRIMA is web-based, it is accessed from your browser. Browsers often store authentication information and will automatically resend it on demand from the PRIMA server. Because of this, it *could* be possible for remote web sites to trigger dangerous actions on your PRIMA server (assuming the web site owner has malicious intentions--it would not happen accidentally!). This page allows you to configure which hosts may refer to actions on your PRIMA server.

Anonymous Module Access

In some circumstances it may be useful to have one or more PRIMA modules accessible to any user, without requiring authentication. For example, it may be useful to allow users to view some read-only statistics about the server, or allow a user to mount or unmount a device using a custom command, or similar. Extreme caution should be taken when using this feature of PRIMA, as giving users access to the wrong module can easily lead to an exploitable condition. To be more explicit, very few standard modules are harmless enough to be safely usable with this feature.

SSL Encryption

If your system has the OpenSSL libraries installed, as well as the Net::SSLeay Perl module, you will be able to use SSL encrypted connections to your PRIMA server. This increases the security of your server by allowing password and user information to be sent in an encrypted form. If you will be accessing your PRIMA server from

across the Internet, it is strongly suggested that you use SSL encrypted sessions. Now that both the export restrictions on encryption have been relaxed and the RSA patent has expired, it is becoming more common for Linux and UNIX versions to always ship with the necessary libraries and Perl module for this to be enabled out of the box. But if you do need some help setting this option up, there is a nice tutorial on the Using SSL with PRIMA page.

Certificate Authority

This page allows you to configure the SSL certificate for this server. Using this, you may configure your system to allow logins without a user name and password. If configured, clients may request a personal certificate in the **PRIMA Users** module, and from then on the browser will authenticate itself via the certificate provided. If your users are located in controlled and secured environments, this feature can make using PRIMA simpler.

To create a certificate, simply fill in the authority information (this can be any information you'd like to include, such as the name of the administrator of the PRIMA server), and click **Setup certificate authority**.

PRIMA Servers

This page provides access to every PRIMA server on your local network. Clicking any icon will direct your browser to the login page of the server clicked. Clicking **Broadcast for servers** will cause PRIMA to send out a broadcast request to port 10000 over your local network. Every PRIMA server on your network will reply and identify itself. PRIMA will then add those servers to the list of servers. You may also scan specific networks for servers, if you manage PRIMA servers remotely. Simply enter the subnet to search and click **Scan for servers**.

Clicking on a server icon will simply direct your browser to the PRIMA port on the selected server, allowing you to log in. You may also configure PRIMA to connect you to the server through a proxy connection, if you provide a user name and password for the other server. This can be useful when connecting remotely to a front-end PRIMA server on the routable Internet that also connects to a non-routable private network, allowing an administrator outside of the private network to tunnel through to administer systems inside the private network.

PRIMA Users

This page allows you to configure any number of users and give each some specified subset of the system to maintain. It would allow you, for example, to create a mail administrator who only had access to the Sendmail module, a DNS administrator who could only modify the DNS records, and a Squid administrator who only had permission to edit the Squid configuration. In this way, delegation of authority is very simply and securely handled.

Editing a PRIMA User

To edit a PRIMA user and the modules they have access to, click on the name of the user on the **PRIMA Users** index page. Each user has a list of accessible modules, and a number of additional options that can be configured. From the **Edit PRIMA User** page, it is possible to change the user's password, select an SSL certificate for them to use for authentication, alter the language and theme from the default, and specify the IP addresses from which this user can login. Also on this page is a list of all modules that are installed on the machine, each with a check box beside it. If checked, the user will have access to the module.

PRIMA also allows finer grained control over many modules, and this functionality is becoming more flexible with every release. For example, a user with permission to use the **Apache Module** can be denied the ability to edit some specific aspects of the configuration. In the example below you can see that the user is being granted permission to edit only one of the many available virtual servers. To edit fine grained access controls, browse to the **PRIMA Users** index page, and click on the module name link beside the user whose access controls you wish to edit.

Creating PRIMA Users

Creating new users is also easy. Click the **Create a new PRIMA User** link and choose a user name or use one of an existing user on the system. Choose between using the users UNIX password, or choose a new one. Select which modules the user will have access to, and click **Save**. Now you can edit the fine grained access controls for the user, or accept the defaults. Adding or deleting modules from the user's access list can be performed by clicking on the user name from the **PRIMA Users** page, and then editing the user in whatever manner is required.

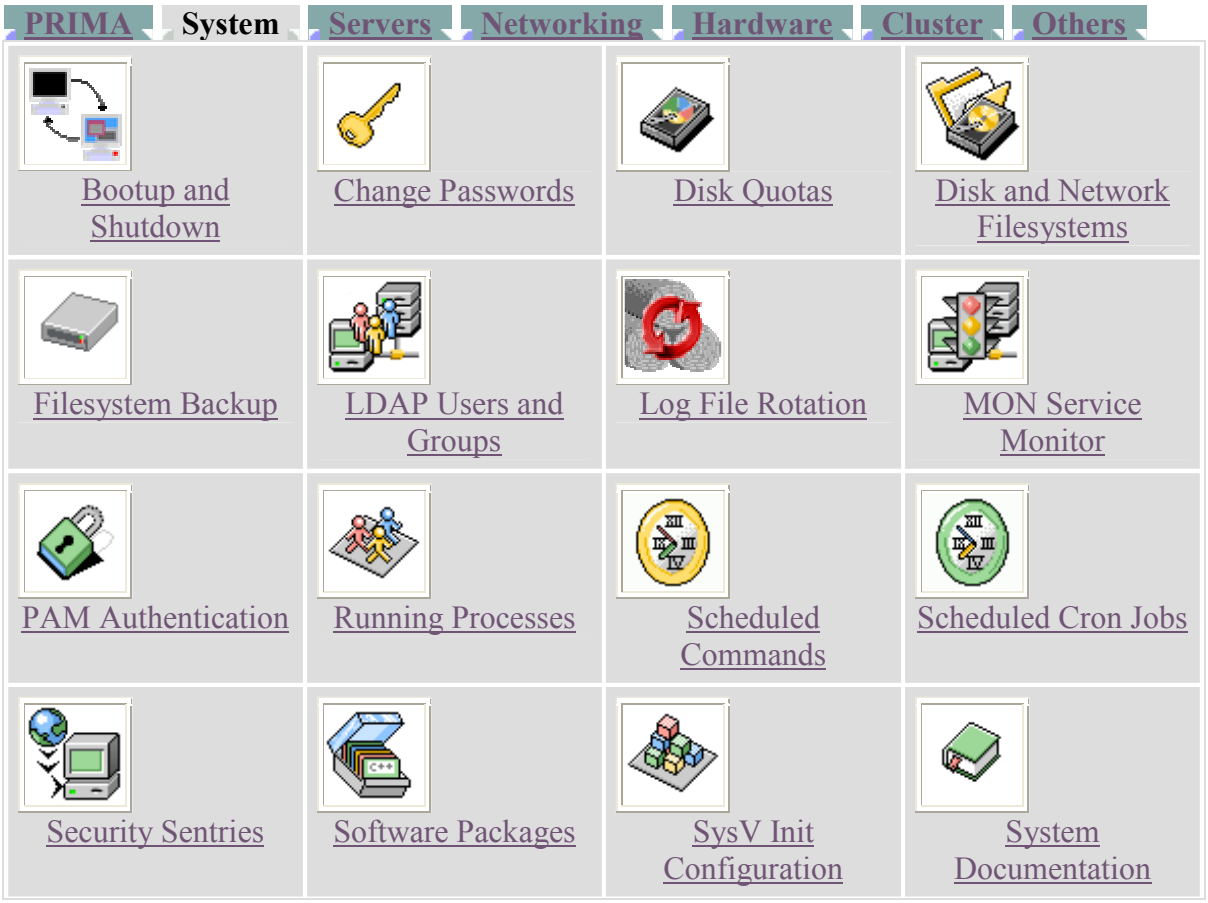
PRIMA Groups

PRIMA, like UNIX, understands the concept of groups. Groups in PRIMA are similar to UNIX groups in that they ease administration of heavily populated servers by allowing easy creation of any number of users with the same set of permissions and access controls. To create a group click on the **Create a new PRIMA group** link, give it a name, and select the modules that members of the new group should have

access to. After saving the new group, any user can be assigned to the group and automatically receive the module access of the group, plus whatever modules are specified for the user. Currently users can only be a member of one group, so the PRIMA groups feature is somewhat less flexible than that of most modern UNIX variants and Linux where users can be members of a primary group in addition to a number of supplemental groups.

System Category

The second category on the PRIMA tab bar is System. Clicking it will allow you to edit such system features as: Bootup and Shutdown behavior, Disk Quotas, Filesystems, Manual Pages, NFS Exports, Processes, Cron Jobs, System Logs, and more. “System Category” shows the options available on a Linux system (specifically a server that's running Red Hat Linux).



Bootup and Shutdown

Clicking bootup and shutdown brings you to a page of bootup options. In the case of a Red Hat system, it provides access to all of the init scripts found in /etc/rc.d/init.d. Clicking on any of the script names will provide the ability to edit, start, stop, and delete the init script. Usually, each init script provides functions to start, stop, and restart system services such as Sendmail, named, and Apache, as well as perform basic system initializations such as setting up network devices and routing tables. An easy way to add a new service or command to the system startup routine, if it does not have an init script, is to add it in /etc/rc.d/rc.local or /etc/rc.local.

Also on this page you'll see the **Reboot** and **Shutdown** buttons. They do just what you would assume, after a confirmation screen.

Change Password

The **Change Password** module allows administrators to change the passwords of some or all users on the system, depending on access control configuration. Generally, the root user can change passwords of all users on the system. This module provides the same functions as the passwd command, but offers a little more flexibility with regard to dictating which passwords can be modified by whom.

Use of the module is mostly self-explanatory. If logged in as a user that has control over more than your own password, you will first see a list of usernames. Click on the one you would like to change the password for, and enter the new password twice (the second field is a confirmation field, to insure the first instance was entered correctly). Depending on the module configuration for the user, it may be necessary to enter the old password before changing the password will be permitted.

Configuring Access Control for Change Password

This module has quite flexible options for dictating what passwords can be changed by a given user. To edit the ACL for this module, browse to the **PRIMA:PRIMA Users** module, and locate the user you wish to edit. Then click on the **Change Password** link for that user. This will open the **Module Access Control** page for the selected user.

Like all modules, the first option is **Can edit module configuration?**, which simply specifies whether the module configuration can be altered by the user. More

interesting options are those that select which users whose passwords can be changed. The **Users whose passwords can be changed** option allows you to choose from a number of options, including All users, Only this user, Only users which allows selection of any number of users from a list, and All except which allows specification of one or more users which cannot be edited while all others can be. The next choice, Users with UID in range is particularly useful, as most UNIX and Linux systems segregate system and non-system users into ranges of UIDs. Specifically, on most Linux systems, system users fall into the 0-499 range, while normal users begin on UID 500. The Users with primary group option allows you to select a primary group to permit the user to change passwords for. This can be useful for segregating users into workgroups with group administrators who can reset passwords (and perform other actions based on the same criteria in other modules). The final option Users matching performs a simple text match on the username.XYZ?

Other options include whether the password must be entered a second time for confirmation (recommended), whether the old password must be entered before a new one can be set, and whether other password restricted services will also be changed. This final option selects whether the new password will apply to databases, Samba, Squid authentication, etc.

Disk Quotas

Disk quotas allow an administrator to specify the amount of space that users are allowed to use before they are no longer allowed to write to the filesystem. PRIMA supports the quota systems on most of its supported operating systems and versions, though the capabilities and specific details are slightly different across the various systems. In the tutorial at the end of the quotas section, there is a walk-through of initializing quota support under Linux and configuring a set of user and group quotas on the /home filesystem.

Disk quotas can be applied to either users or groups or both. They are applied to all files and directories within a given filesystem that belong to a given user or group. It is possible to apply quotas to some users while allowing others to have unrestricted use of the filesystem. Most operating systems support soft limits, which allow the user to surpass their limit for some period of time, and hard limits which immediately stop writing data upon reaching the limit. Limits can be placed on the amount of space (usually in disk block increments) and on the number of files.

Disk and Network File system:

The **Disk and Network Filesystem** page provides a detailed view of the filesystems listed in /etc/fstab. From this page you can edit mount points, create new mount points, umount and mount partitions, and add execute and setuid restrictions to specific mount points for security. This module configures the /etc/fstab file

System Documentation

This page provides access to the extensive help that is available on most UNIX systems through man pages, in addition to the PRIMA help files, installed package documentation files, Perl module documentation, as well as results from the Google search engine.

Scheduled Commands

The `at` command provides a simple means to execute a specified command at a specified time. Its usage is simple, made even simpler by the PRIMA interface. It can be very useful for a number of tasks, such as running one-time CPU intensive tasks at off-hours, notifying you of appointments, etc.

To create a new `at` job, simply fill in the details. Specifically, the **Run as user** option dictates the user under which the command will be run. **Run on date** and **Run at time** specifies the date and time at which the command will run. The **Run in directory** option specifies where the `at` command will be run from, as a change directory command will be run before the command is executed. This directory must be accessible by the user under which the command is run. Finally, the **Commands to execute** is where you may enter the commands to be run by `at` at the specified time. Any number of linefeed separated commands may be entered and they will be executed in sequence

Scheduled Cron Jobs

The **Cron Jobs** module is used for editing the crontab on your system. Cron is a daemon that runs constantly on most UNIX systems, and allows users and the adminstrator to run specified tasks automatically at selected times. Ordinarily, crond is configured from the system-wide crontab as well as one or more configuration directories in /etc/cron.d, and on Red Hat Linux systems and some other Linux distributions crond draws its configuration from /etc/cron.hourly, /etc/cron.daily, /etc/cron.weekly, and /etc/cron.monthly. Note that even on Red Hat and similar systems, /etc/cron.d and /etc/crontab still exists and can be used just as on any other UNIX system.

Configuration of crond is much simplified by use of the PRIMA module. To create a new cron job, click **Create a new cron job**. The **Create Cron Job** page allows you to select the user that the cron job will run as, thereby limiting its permissions to those of the selected user. As in all permissions situations, it is best to choose a user with the least permissions required to actually accomplish the task needed. There are fields for entering the **Command** you want to be executed, as well as for any **Input to command** you might have. The **Active** option dictates whether the command is enabled or disabled by commenting it out with a hash mark at the beginning of the line.

Software Packages

The **Software Packages** module allows an administrator to perform software upgrades and package maintenance via a quite friendly interface. Alhough the actual implementation can vary quite a lot depending on which software packaging system your operating environment uses, PRIMA masks most differences and the overall usage of each is very similar.

System Logs

System Logs provides a method for controlling the **syslogd** daemon used on most UNIX system to provide standard logging functions. The module opens with a list of all currently existing logs. By clicking on the **Log destination** of a log file, you can edit the logging properties.On the editing page there is also a **View log** button, that allows you to view a configurable number of lines from the end of the log file. It also allows a constantly refreshing log view if selected

Users and Groups

UNIX is, at its very core, a multi-user operating system. It was built from the ground up to provide services to a number of simultaneous users. Because of these features, UNIX provides a simple, but effective, method for restricting users to only the parts of the system they should have access to. The **Users and Groups** module in PRIMA attempts to provide a nice front-end for those features of the system. Each version of UNIX has differences in how users and groups are implemented. However, PRIMA hides those details quite effectively so that you never have to think about it. The PRIMA **Users and Groups** module edits several system files, depending on your OS. Usually, the files are `/etc/passwd` and `/etc/shadow`, for user names and passwords, and `/etc/group` and `/etc/gshadow`, for groups and group passwords. Note that the *shadow* versions of the preceding files are far more secure than standard `passwd` files because they are only accessible

by the root user. Shadow passwords are standard on most Linux distributions today, and many other systems as well.

Clicking on a user name or group name will take you to an **Edit User** page, allowing you to edit all facets of the account. Note that changing the user or group ID at some point in time after the account is created is risky, as permissions are set by ID, not user/group name. While the module will change these for you on the home directory, there may be user programs or even system programs that rely on the UID to remain the same. Also note that on some systems (Red Hat and probably other Linux distributions) the user and the users primary group are always the same name by default. Red Hat Linux includes the `adduser` command, which will create a group of the same name and ID as the user, and therefore PRIMA can do the same. You should not change this behavior, unless you really know what you're doing, as the system relies on this for much of its access control flexibility. Unlike some traditional UNIX variants, Linux users can have many secondary groups active at all times, which can be set to any group(s) you need.

Server

PRIMA | System | **Servers** | Networking | Hardware | Cluster | Others

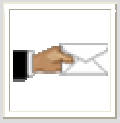

Apache Webserver


BIND DNS Server

CVS Server

DHCP Server

Dovecot IMAP/POP3 Server


Fetchmail Mail Retrieval


Frox FTP Proxy


Jabber IM Server


Majordomo List Manager

MySQL Database Server

OpenSLP Server

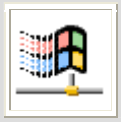

Postfix Configuration


PostgreSQL Database Server


ProFTPD Server


Procmail Mail Filter

QMail Configuration

 Read User Mail	 SSH Server	 Samba Windows File Sharing	 Sendmail Configuration
 SpamAssassin Mail Filter	 Squid Analysis Report Generator	 Squid Proxy Server	 WU-FTP Server
 Webalizer Logfile Analysis			

Clicking the **Servers** tab on the PRIMA category bar brings you to what is probably the most interesting of the PRIMA pages. It is here that all of the various complex servers and daemons can be configured. PRIMA provides standard modules for a large number of the most popular servers and daemons in use on network systems in the world today, and more are being written all the time

Introduction to Servers

The **Servers** PRIMA category allows for administration of the server applications that run on a system, which provide some service to clients on the network. One example is the Apache web server daemon. Clicking on the Apache icon in this category allows you to edit the Apache configuration files, which are usually located in `/etc/httpd/conf`. Most modules located in **Servers** will enable you to edit some configuration file found in `/etc` or some subdirectory therein. One of the most impressive features of PRIMA is the ability to allow you to edit files without damaging existing hand-edited configuration details.

The root of much of PRIMA's popularity is the ability for an administrator to perform some tasks through the PRIMA interface without being forced to do *all* tasks with PRIMA. Unlike some graphical front-ends for UNIX systems, PRIMA leaves an edited file intact as much as is possible. Comments are untouched and the ordering of directives is not changed. This results in a system that can be configured through PRIMA, and through hand-editing of configuration files, but with no conflict between the two methods.

A side effect of this feature is that PRIMA generally must provide a one-to-one interface to configuration files in order to insure that configuration options are not confused and to insure that a savvy administrator gets what is expected from the PRIMA output. This means that PRIMA is not an "easier" way to administer a UNIX system. The administrator still must understand the tools he or she is administering with PRIMA. For example, configuring DNS from PRIMA requires an understanding of **named** and its required configuration files. The same applies to **squid**, **httpd**, **sendmail**, etc. PRIMA *can* make the learning process somewhat quicker, however, as all of the options are present on the display, which may or may not be the case with configuration files.

Apache Webserver

The **Apache** module is broken up into several sections to address different aspects of an Apache configuration. On the main page, these sections are grouped into **Global Configuration** and **Virtual Servers** groups. Using virtual servers it is possible to locate several web sites with unique domain names on a single IP address, in order to conserve the rapidly diminishing IPv4 address space. In the context of the **Apache** module, **Global Configuration** refers to configuration information that will apply to *all* virtual hosts that are run from the same **httpd**

daemon. It is usually unnecessary to run more than one **httpd** daemon on a single machine but it is possible. It is also possible to manage more than one such daemon with PRIMA via the module cloning feature.

BIND

DNS, or the Domain Name System, is absolutely vital to the functioning of the Internet. In fact, though you rarely interact directly with the DNS the Internet as we know it could not exist without its constant presence. DNS associates, or *binds*, host names and domain names to IP addresses and thus allows you to type instead of the much less memorable IP 216.40.244.74. Further, it makes it possible for mail servers to easily locate the correct host to send mail to for a given domain, the correct administrative contact when strange things are originating from the domain, and more. But for our purposes, as ordinary system administrators, all we need to really keep in mind is that BIND is our method of providing DNS information for our network. It will provide information to our local users, when their client applications need to access various sites by name. And it will provide information to clients (primarily other DNS servers acting as DNS clients in order to fetch the correct information for their clients) on the Internet at large in order to advertise to the world how host names on our network can be reached. Think of it as a fancy telephone book, or even better a telephone operator, for networked computers. The client computer has a name, but needs the number in order to reach it across the vast Internet. So it contacts the DNS server and asks for the number, and BIND is happy to do its best to return the correct number.

Every host on a TCP/IP network has an *IP address*. This address must be unique for the network on which the address is routable. So, every host that is accessible via the Internet has a unique IP address that may, theoretically, be reached from anywhere else on the Internet. Because these addresses are doled out, roughly, according to physical location on the network, and because routers keep up with which other routers have access to which subnets, this simple number is all that is needed for your computer to establish a connection with any other computer on the Internet in seconds. Unfortunately the topic of routing on the Internet falls quite outside of the scope of this document, as PRIMA is not designed to manage routers or the more complex routing features of Linux, FreeBSD, and the other operating systems that are supported.

Sendmail

Sendmail is the de facto standard mail transfer agent, or *MTA*, in use on the Internet today. While there are now several worthy contenders for the title of best or most popular MTA, including Postfix and QMail (both of which have very good PRIMA modules, and Postfix is documented in the preceding chapter), more mail probably passes through Sendmail than any other single MTA.

An MTA is the software that provides mail services for a network. A client mail user agent, or *MUA* sends email, usually via the Simple Mail Transport Protocol, or *SMTP*, to the MTA. The MTA uses one of several transport protocols, most often via SMTP, to deliver it either directly to the recipient (if the address is served by the same server) or to the mail server for the user. Clients then access the mail on the server using either POP3 or IMAP. So, Sendmail will operate on your server and provide those intermediary services, both sending and receiving mail, for clients and other MTAs on the Internet

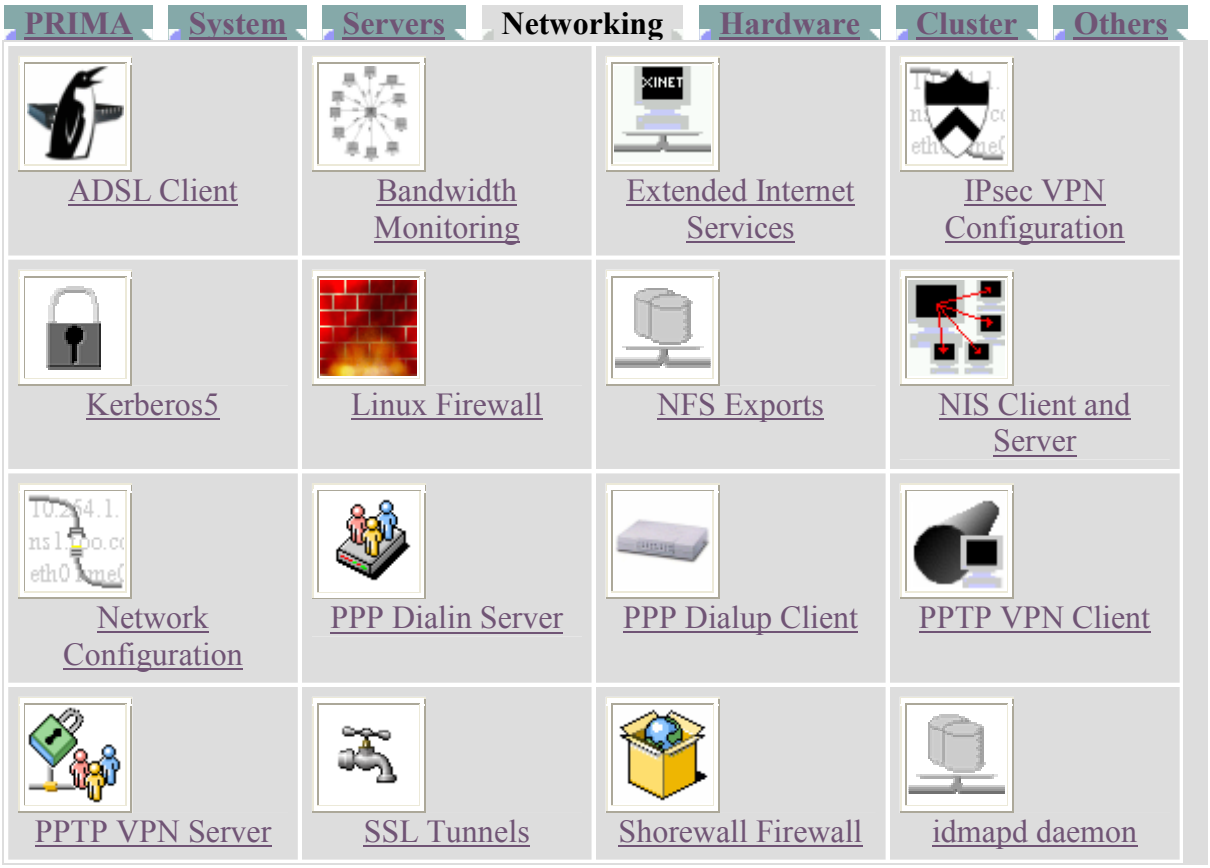
Squid

Squid is a feature-rich and extremely flexible Web-caching proxy daemon. Most configuration is performed by editing a simple configuration file called squid.conf which is usually located in /usr/local/squid/etc/squid.conf or, on systems derived from Red Hat Linux, /etc/squid/squid.conf. Each behavior is set by a directive followed by one or more options. The PRIMA interface provides access to most of

the directives available for configuring Squid. Because Squid is a quite complex package, the PRIMA interface opens with a series of icons to represent the different types of configuration options.

Networking Configuration

The fourth category tab in PRIMA is for **Networking** related configuration. Specifically, this is where you'll find the modules to configure inetd or xinetd, iptables, NFS exports, NIS client and server, and network interfaces. The **Networking** category was introduced in PRIMA version 1.0, so if for some reason you are using an earlier PRIMA revision (upgrade, already, it's free!), these modules will be found under other category tabs.



Hardware Configuration

The **Hardware** configuration tab, presents several options for hardware level settings This includes disk partitions, boot loader, printer administration, system time and date, and more. The exact options available vary greatly depending on platform.

PRIMA	System	Servers	Networking	**Hardware**	Cluster	Others


CD Burner


GRUB Boot Loader


Linux RAID


Logical Volume Management


Partitions on Local Disks


Printer Administration


SMART Drive Status


System Time


Voicemail Server

Linux Boot Configuration

If you are running Linux, PRIMA provides module for editing the /etc/lilo.conf file that the *LInux LOader*, or LILO for short, uses. LILO is the boot manager most commonly used with Linux. It works by writing a small boot sector to the MBR on your boot disk. This boot sector contains the code needed to first present a prompt:

LILO:

This allows you to select the kernel or operating system to boot. LILO can boot multiple operating systems and multiple versions of the Linux kernel. When you open the **Linux Bootup Configuration** page, you should see at least one *boot kernel* or *boot partition*. Boot kernels are for Linux kernels, while boot partitions are for other operating systems.

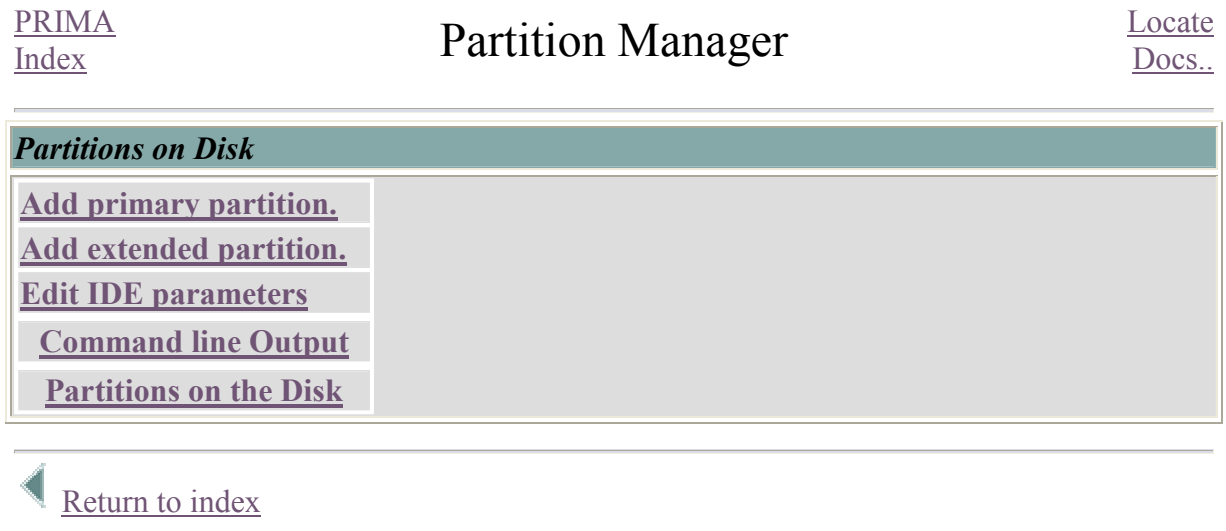
GRUB Boot Loader

Much like the LILO bootloader discussed in the previous section, GRUB, or the GRand Unified Bootloader, is a bootloader that provides boot selection for any

number of Linux kernel revisions and other operating system variants. It is more powerful and flexible than LILO, but also somewhat more complex in operation than LILO. Most modern Linux distributions provide both GRUB and LILO boot loaders, with GRUB being the default. GRUB operates slightly differently than LILO, as it does not have to be rewritten to the MBR anytime the kernel or boot parameters are changed. GRUB knows how to read many filesystem types, and therefore can mount the system boot disk and read in its configuration files and new kernels during the boot process.

Partition Manager

The partition manager provides a graphical interface to both the **fdisk** partition editing command and the /etc/fstab file on your system **fdisk** is commonly used to create and remove partitions on a disk, while the /etc/fstab file is where the mounting information for the system is configured.



The partition manager page provides a list of links that is related to the partitions on disk.Links are Add primary partitions, Add extended partitions,Edit IDE partitions, Command line output, Partitions on the disk To create a new partition on a disk with space available, click either **Add primary partition** or **Add logical partition**. Choose the type of partition to create (usually Linux or Linux swap) and then the size of the partition in the **Extent** field. This is the range of cylinders to use for this partition, so take care to correctly identify the cylinders, and to not overwrite any existing partitions. Finally, click **Create**.

Click on the IDE parameters it will show the IDE parameter.How much the space available then go for the Partitions on the disk. Whenever giving the command line on text field it will show the information regarding to your local disk

Add Primary Partitions:

PRIMA
Index
Module
Index

Create Partition

Partition Details			
Location	IDE device A partition 4	**Device file**	/dev/hda4
Type	Linux	**Extent**	0 - 0 of 1044
Status	Not created yet	**Size**	Not created yet

Create

Partition on the disk

PRIMA
Index

Partition Manager

Locate
Docs..

Edit Partitions

No.	Type	Extent	Start	End	Use	Free
1	Linux		1	319	/	1 %
2	Linux swap		320	580	swap	
3	Extended		581	1044		

Printer Administration

The **Printer Administration** module may behave somewhat differently under different operating systems, as printer driver details and configuration often varies between systems, and even between Linux distributions. Nonetheless, PRIMA minimizes the differences between systems, and these directions work more or less unchanged

This paper is not intended to be a definitive implementation guide. Many factors are not addressed in this document. Expertise may be required to solve logistical problems when the system is designed and built. VAssure team has not tested this procedure with all the combinations of hardware and software options available on all OS variants. There may be significant differences in your configuration that will alter the procedures necessary to accomplish the objectives outlined in this paper.

