ENSIM PRO - LINUX

Ensim Pro 10.1.0 for Linux
Technical Reference Guide

Published: 25 September 2006

CORPORATE HEADQUARTERS

ENSIM CORPORATION
1366 Borregas Avenue
Sunnyvale, California 94089
(408) 745-3300

www.ensim.com

Contents

About this guide

Introduction

This guide provides information and instructions on enhancing or customizing Ensim® Pro 10.1.0 for Linux® using Command Line Interface (CLI) scripts. The scripts enable you to perform basic administrative tasks. For simplicity and convenience, we refer to Ensim Pro 10.1.0 for Linux as Ensim Pro throughout this document.

Intended audience and required skills

This guide is intended for users of Ensim Pro and includes system administrators and technical support staff of Internet Service Providers (ISPs) who use Ensim Pro. To use this guide, you need to be familiar with the Linux command line interface and working of Ensim Pro.

Related documentation

The following documents provide additional information about Ensim Pro for Linux:

- *Ensim Pro and Ensim Basic for Linux Release Notes*
- *Ensim Pro for Linux Customization Guide*
- *Server Administrator's Help*
- *Reseller Administrator's Help*
- *Site Administrator's Help*
- *User Administrator's Help*
- *Ensim knowledge base articles*

Documentation is also available in the Ensim Pro for Linux section of the Ensim Support site, http://support.ensim.com. All customers receive passwords that allow access to this site. If you do not have a password, contact your organization's liaison to Ensim.

Document conventions

Throughout this guide, specific fonts are used to identify user input, computer code, and user interface elements. The following table lists conventions used in this guide.

Table 1. Document conventions

Convention	How it is used
Cross-references with page number links	Used to provide quick access to information in the document. The page number is a hyperlink. *Example:* See Legal and copyright notice on page 2.
Bold text, serif font	Used for information you type as well as for file names, path names, and CD names. *Example:* On the command line, type **continue.**
Italic text	Used for column names, field names, window names, and book titles. *Example:* The *Select Installation Folder* window is displayed.
Bold, italic text enclosed in angle brackets	Used for variables you replace with the appropriate information. *Example:* Type ***<server_name>*** where ***<server_name>*** is the IP address or host name of your server.
Bold text, sans serif font	Used for items you click or press, such as buttons, menus, and command keys. *Example:* Click Start > Settings > Control Panel.
`Non-proportional font`	Used for system messages, screen text, and code examples. *Example:* The following message is displayed: `The server has been added.`

Note: In addition, key information is sometimes displayed using special headings and formats, such as this one, to make it stand out from regular text.

Support and feedback

For Ensim online support or feedback, use the following links:

- http://support.ensim.com to create an Ensim Support account and access other documentation.
- https://onlinesupport.ensim.com to log in to Ensim Support using an existing account
- http://onlinesupport.ensim.com/kb_search.asp?product=lwp to search the knowledge base
- http://www.ensim.com/about/feedback.asp to provide feedback to Ensim

Note: All customers receive passwords that allow access to the Ensim Support site. If you do not have a password, contact your organization's liaison to Ensim.

CHAPTER 1

An introduction to Ensim Pro

Introduction

This section provides an overview of Ensim Pro.

In this section:

Overview of Ensim Pro for Linux

Ensim Pro is a robust control panel solution for small and large professional hosting providers. Ensim Pro includes all the tools and features that you, as a hosting provider, need to build compelling, commercial-quality plans for shared and reseller hosting.

With Ensim Pro, you can:

- Define targeted, compelling Service Plans for shared site and reseller hosting. By defining a Service Plan once, you can establish new customers quickly, while controlling application settings and resource quotas.
- Customize the look and feel of the control panel to suit your branding preferences.
- Review detailed reports of bandwidth and disk usage per site and allow accurate and complete billing of all traffic used by sites.
- Protect shared sites with a virtual private file system and provide additional site security.
- Delegate administration to your customers, including your resellers, site owners, and site users, while simultaneously reducing your support calls and improving customer satisfaction.

Server Administrators

The **Server Administrator** account is set up the first time the server owner accesses Ensim Pro. This account holder is different from the UNIX admin user on the server and therefore has no direct access to the server.

Server Administrator tasks

Server Administrators typically perform the following tasks.

- **System management tasks**

- Reviewing network settings
- Reviewing disk usage
- Changing the user name and password
- Changing the system time
- Restarting Ensim Pro
- Restarting the host server
- Exporting and importing data

- **Application management tasks**
  - Configuring the Web server
  - Configuring the DNS server
  - Configuring the FTP server
  - Configuring the Mail server
  - Configuring other applications
- **Reseller management tasks**
  - Creating Reseller Plans
  - Creating resellers
  - Suspending resellers
  - Managing reseller sites
- **Site management tasks**
  - Creating Service Plans
  - Offering IP-based and name-based sites
  - Creating sites
  - Managing sites
  - Suspending sites
  - Removing sites
  - Securing sites with SSL certification

Reseller Administrators

The **Reseller Administrator** is an individual who wants to resell or manage sites without actually owning or maintaining hardware. The reseller buys sites from an ISP or hosting provider and resells them to individual customers. The Reseller Administrator is strictly limited to the Reseller Administrator control panel and has no access to the server.

Reseller Administrator tasks

Reseller Administrators typically perform the following tasks.

- **System management tasks**
  - Reviewing site reports
  - Changing the user name and password

  - Exporting and importing data
- **Site management tasks**
  - Creating site plan
  - Offering IP-based and name-based sites
  - Creating sites
  - Managing sites
  - Suspending sites
  - Deleting sites
  - Securing sites using SSL

Site Administrators

The **Site Administrator** account is created when the site is created. It is both a UNIX account and an Ensim Pro control panel account. Each site has an associated site number. The Site Administrator account maps to an "admin" user and group created in **/etc/passwd** and **/etc/group** that is associated with the site number (for example, admin1).

The Site Administrator's access depends on the security level configured for the site. For more information on the different security levels, refer to the **Server Administrator Help**.

Site Administrator tasks

Site Administrators typically perform the following tasks.

- **Site management tasks**
  - Manage administrator information
  - Exporting and importing data
  - Managing files
  - Reviewing site reports
- **User management tasks**
  - Adding and managing users

User Administrators

The **User Administrator** is a user on an IP-based or a name-based site. The User Administrator has an entry in the virtual site file **/home/virtual/site#/fst/etc/passwd** and is restricted to **/home/virtual/site#/fst**.

User Administrator tasks

User Administrators typically perform the following tasks.

- Updating user information
- Exporting and importing data
- Reviewing usage data

- Managing email features

Working with Ensim Pro

As the Server Administrator, you can perform common administrative tasks in one of two ways. You can log on to Ensim Pro and use the Server Administrator control panel **or** you can use any Telnet or SSH client to access the server running Ensim Pro as a root user and execute a set of scripts from the Command Line Interface.

For information on using the Server Administrator control panel, access the integrated online Help. For information on using the Command Line Interface, see the Command Line Interface section on page 12.

CHAPTER 2

Using Command Line Interface scripts

Introduction

This section provides information on using Command Line Interface scripts.

> **Note:** Command Line Interface scripts are advanced features for which limited Customer Support is available. The syntax of these commands may change in future releases of Ensim Pro.
>
> Some of the commands in this chapter include the shell primitive \ which is used to enter a new line, without the shell executing the command. This is included for reasons of clarity and can be omitted if your command fits in one line.

In this section:

About Command Line Interface scripts

Ensim offers Command Line Interface scripts for automating some of the tasks you perform as an ISP or Server Administrator. Most of these scripts reside in the directory **/usr/local/bin** and some in **/usr/sbin**. Use any Telnet or SSH client to access the server running Ensim Pro and execute these scripts.

The command line is composed of PERL programs and modules. The arguments for a script are passed in an encoded format by the interface to avoid security problems with special shell escapes. As a rule the interface fills in defaults and the command line tool checks all the arguments it receives.

The error messages go to STDERR and the output goes to STDOUT. If the command fails, it exits with code 1. If there are error messages on STDERR and the exit code is 0, it means that the operation succeeded, but some problems were encountered.

Functions that add information try to delete all the changes they made if an error occurs (so they either totally succeed with no warning messages, or fail). Functions that delete information try to push through even if some errors (other than invalid parameters or domain not found in the configuration file or some other major problem in the very beginning) occur.

These operations can succeed and still give warnings. The operations that change information behave somewhat similarly to the ones that delete them.

To see the syntax for any command, type the following at the command line.

<script name> **--help**

Using Command Line Interface scripts

This section explains the usage and syntax of scripts.

You can use scripts for:

Looking up site information

The Ensim Pro file system maintains site information by assigning each domain a number and a corresponding top-level UNIX user through which it identifies the site's following basic information.

- The domain's root directory
- The name of the domain on which the site resides
- The user name of the Site Administrator

Each site on your server is known to the file system as site*<n>* (called the site handle). The top-level UNIX user is known as admin*<n>*, where ***n*** is a number matching the site number.

The top-level UNIX user is the user handle to the site. Process lists will not show Site Administrator names, but instead show top-level UNIX user names.

> **Note:** Using the **ps** command will not show which process belongs to which domain. Use the **sitelookup** command after **ps**, to map a UNIX user to a site and view information about the sites on your server.

Syntax

```
/usr/local/bin/sitelookup [-a] [-w <wp_user>] [-s <site_handle>] \
[-d <domain>] [-u <site_admin>] \
[domain, wp_user, site_handle, site_root, site_admin]
```

where:

- `-a` returns information for all domains.
- `-w` returns site information for the site identified by the top level UNIX user ***<wp_user>*** you specify.
- `-s` returns site information for the site identified by the site handle ***<site_handle>*** you specify.
- `-d` returns site information for the site identified by the domain name ***<domain>*** you specify.
- `-u` returns site information for the site identified by the user name of the domain's Site Administrator ***<site_admin>*** you specify.

The command returns the following information:

- `site_root` - the domain's root directory.
- `domain` - the name of the domain on which the site resides.
- `wp user` - the top level UNIX user.
- `site admin` - the user name of the Site Administrator.
- `site handle` - the file system's name for the site.

The following section lists some examples of this syntax.

Example 1

The command:

```
sitelookup -w admin1 domain,site_handle
```

returns the following information associated with the top level UNIX user **admin1**:

- domain name
- file system's name of the site

For example:

```
example.com,site1
```

Example 2

The command:

```
sitelookup -s site25 site_root
```

returns the name of the root directory of the site with the site handle **site25**.

For example:

```
/home/virtual/example1.example.com
```

Example 3

The command:

```
sitelookup -a
```

returns the following for all the sites you manage.

- domain name
- top-level UNIX user
- site handle
- site root directory
- name of the Site Administrator

For example:

```
example1.example.com,admin1,site1,
 /home/virtual/example1.example.com,Pawan
example2.example.com,admin2,site2,
 /home/virtual/example2.example.com,Dave
```

Changing the Server Administrator password

To change the password of the Server Administrator (server owner), use the **passwd_appl_admin** script. The new password must be entered on standard input to complete the script. The change is effected immediately without the need to restart Ensim Pro.

Syntax

```
/usr/sbin/passwd_appl_admin <appl_admin_name>
```

where:

<appl_admin_name> is the user name of the Server Administrator.

Changing the Site Administrator password

To change the password of the Site Administrator (domain owner), use the **ChangeDomainPasswd** script. This script will not exit until you complete the operation by entering the new password for the domain on standard input.

Syntax

```
/usr/local/bin/ChangeDomainPasswd <domain name> <domain password on stdin>
```

where:

- ***<domain name>*** is the name of the domain for which you want to change the password.
- ***<domain password on stdin>*** is the new password for the Site Administrator that you must enter to complete the script.

Changing the Site Administrator email address

To change the Site Administrator's email address, use the **ChangeEmail** script.

Syntax

```
/usr/local/bin/ChangeEmail <domain name> <email address>
```

where:

- ***<domain name>*** is the name of the domain for which you want to change the password.
- ***<email address>*** is the Site Administrator's new email address.

Changing the User Administrator password

To change the password of the User Administrator (domain user), use the **ChangePasswdVirtUser** script. This script will not exit until you complete the operation by entering the user's new password on standard input.

Syntax

```
/usr/local/bin/ChangePasswdVirtUser <domain name> <username> <user's password on stdin>
```

where:

- ***<domain name>*** is the name of the domain.
- ***<username>*** is the user's login name.
- ***<user's password on stdin>*** is the user password you must enter on the standard input to complete the script.

```
ChangePasswdVirtUser example.com myname newpass
```

In this example, the script changes a user's password with the following specifications.

- The domain is called example.com.
- The username is Myname.
- The user's new password is newpass (entered by you on stdin).

Adding a Service Plan

Ensim Pro is shipped with a single default Service Plan that contains all the services and options you need to create and make Ensim Pro domains usable by your customers. However, you can add any number of additional Service Plans to suit the needs of your customers and their businesses.

Many Internet service providers find it most efficient to use a template-type Service Plan that provides the basic services and options that all their customers will use; then create additional Service Plans that add other optional services to meet specific customer needs. This way, they do not have to specify all services and options whenever they create a new Service Plan. By using a template Service Plan, they can automatically assign most services and options; then add specific additional services and options for new Service Plans.

To add a new Service Plan, use the **AddPlan** script.

Syntax

```
/usr/local/bin/AddPlan
    [-p <source plan> | -s <source site> | -d <source domain> |
   -i <source IP> | -t <source path>]
   [-c <service>,<option>=<value>,[on|off],... ...]
   [-f | --force] <target plan>
```

where:

- ***<source plan>*** is the name of the Service Plan you want to use as the foundation or template for the new Service Plan. This option can be omitted, in which case the default Service Plan is used.
- ***<source site>***, ***<source domain>***, and ***<source IP>*** are three ways to specify a site whose configuration you want to use as the foundation or template for the new Service Plan.

  Note: If the -t option is used, the path will be **/home/virtual/*<sourcepath>*/info/current**.
- ***<service>*** is the name of the service for which you want to specify options in this Service Plan.
- ***<option>*** is optional features, if any, that you want to specify for the service.
- ***<value>*** is what the option specifies, such as a name, password, or measurement. ***<value>*** may be a single string (for example, **1** or **My User**) or a list (for example, **\[1, 2, 3\]**)
- Each service has an **enabled** option, which may be used to enable or disable the service by setting this option to **1** or **0**, respectively. Alternatively, the strings **on** and **off** may be used as abbreviations for the strings: enabled=**1** and disabled=**0**.
- ***<target plan>*** is the name of the new Service Plan.

The following table lists the services you can specify when you create a new Service Plan, as well as the services' options and their values.

Table 2. Service Plan services and options I

Service	Option	Value	Description
siteinfo	enabled	1 (enabled) **or** 0 (disabled)	Whether site information is enabled or disabled for the site.
spam_filter	None	on (enabled) **or** off (disabled)	Whether spam filter is enabled or disabled for the site.
mailscanner		on (enabled) **or** off (disabled)	Whether virus scanning is enabled or disabled for the site.

Table 2. Service Plan services and options I

Service	Option	Value	Description
	scan_incoming	1 (enabled) **or** 0(disabled)	Whether virus scanning is enabled or disabled for incoming email messages.
	scan_outgoing	1 (enabled) **or** 0(disabled)	Whether virus scanning is enabled or disabled for outgoing email messages.
	domain	plain text (for example: myco.com)	The default domain name for a site created with this Service Plan.
	admin_user	plain text	The user name for the Site Administrator.
	password	plain text	The Site Administrator's password. The administrator will be prompted for the plain text password.
	tpasswd	plain text	The Site Administrator's password. Specify the password by typing **tpasswd=*<plain text password>*** at the command line.
	cpasswd	encrypted text	The Site Administrator's password. Specify the password by typing **cpasswd=*<password in encrypted text>*** at the command line.
	email	plain text (for example: admin@myco.com)	The Site Administrator's email address.
aliases	enabled	1 (enabled) **or** 0 (disabled)	Whether the aliases option is enabled or disabled for the site.
	aliases	list of text items	The domain's aliases. Each alias can be used as the domain portion of an email address, a URL, or the target host of an FTP, Telnet, SSH, IMAP, or POP connection. This must be specified as a list, as described for **ipaddrs** (see page 2-9).
analog	enabled	1 (enabled) **or** 0 (disabled)	Whether Analog log analyzer is enabled or disabled for the site.

Table 2. Service Plan services and options I

Service	Option	Value	Description
logrotate	enabled	1 (enabled) **or** 0 (disabled)	Whether logrorate is enabled or disabled for the site.
ipinfo	namebased	1 (name-based). **or** 0 (IP-based)	Whether the site is name-based or IP-based.
	ipaddrs	One or more IP addresses, separated by commas.	The list of IP addresses that will be configured if this site is IP-based. The format requires brackets: [*<address1>*, *<address2>*] **Note:** Some shells treat brackets as special characters. You may need to include escape characters for the brackets: \[*<address1>*, *<address2>*\]]
diskquota	enabled	1 (enabled) **or** 0 (disabled)	Whether disk quota is enabled or disabled for the site.
	units	• B or b for bytes • KB or kb for kilobytes • MB or mb for megabytes • GB or gb for gigabytes	The unit of measurement for disk quota.
	quota	***<quota> <unit>*** (for example: 500 MB)	The number specifying the size of the quota, in the units specified.
telnet	enabled	1 (enabled) **or** 0 (disabled)	Whether Telnet is enabled or disabled for the site.
	jail	1 **or** 0	If set to 1, the Site Administrator's shell access to the site through Telnet will be jailed within the site's file system. If set to 0, the Site Administrator may browse the entire server's file system, except where restricted by directory ownership or permissions.

Table 2. Service Plan services and options I

Service	Option	Value	Description
bandwidth	enabled	1 (enabled) **or** 0 (disabled)	Whether bandwidth montoring is enabled or disabled for the site. **Important:** Disabling bandwidth while adding a domain using the **AddVirtDomain** script produces erratic mail behavior. Do not disable bandwidth when you add a domain using the **AddVirtDomain** script.
	threshold	number	The number of bytes, after which the Site Administrator should be notified that the site has exceeded its bandwidth allocation.
	rollover	number (any number between 1 and 31)	The date that will be used to calculate monthly totals. **Note:** If set to 0, the last date of each month is used.
	units	B or b (Bytes) KB or kb (Kilobytes) MB or mb (Megabytes) GB or gb (Gigabytes)	The unit of measurement used to indicate bandwidth threshold.
ssh	enabled	1 (enabled) **or** 0 (disabled)	Whether SSH access is enabled or disabled for the site.
	jail	1 **or** 0	If set to 1, the Site Administrator's shell access to the site through SSH will be jailed within the site's file system. If set to 0, the Site Administrator can browse the entire server's file system, except where restricted by directory ownership or permissions.
imap	enabled	1 (enabled) **or** 0 (disabled)	Whether the IMAP mail protocol is enabled or disabled for the site.
bind	enabled	1 (enabled) **or** 0 (disabled)	Whether the BIND name server protocol is enabled or disabled for the site.

Table 2. Service Plan services and options I

Service	Option	Value	Description
users	maxusers	number	The maximum number of users allowed on this site.
proftpd	ftpserver	plain text	The FTP server's domain name. **Note:** If you are using domain aliasing, and the name of this server contains a prefix (such as FTP), followed by the domain name as specified in **siteinfo** (see page 2-7), the aliasing function will attach this prefix to all other aliases to generate FTP server names.
apache	enabled	1 (enabled) **or** 0 (disabled)	Whether the Apache Web server protocol is enabled or disabled for the site.
	jail	1 (enabled) **or** 0 (disabled)	If set to 1, a high security site is created. If set to 0, a 3.1 compatible site or low security site is created.
	webserver	plain text	The Apache Web server's domain name. **Note:** If you are using domain aliasing, and the name of this server contains a prefix (such as apache), followed by the domain name as specified in **siteinfo** (see page 2-7), the aliasing function will attach this prefix to all other aliases to generate Web server names.
	jail	1 **or** 0	If set to 1, certain Apache features and Apache-related services will be restricted access to other sites' data. In particular, mod_perl and mod_php will be disabled for the domain (interpretation and execution of PHP and Perl scripts will be re-routed through jailed CGI versions of PHP and Perl), and the use of "Options FollowSymlinks" will be denied. If set to 0, these restrictions will be removed.

Table 2. Service Plan services and options I

Service	Option	Value	Description
	mailserver	plain text	The mail server's domain name. **Note:** If you are using domain aliasing, and the name of this server contains a prefix (such as mail), followed by the domain name as specified in **siteinfo** (see page 2-7), the aliasing function will attach this prefix to all other aliases to generate mail server names.
	preference	number	The preference to give to MX records for the mail server names.
anonftp	enabled	1 (enabled) **or** 0 (disabled)	Whether Anonymous FTP access is enabled or disabled for the site.
openssl	enabled	1 (enabled) **or** 0 (disabled)	Whether OpenSSL access is enabled or disabled for the site.
cgi	enabled	1 (enabled) **or** 0 (disabled)	Whether CGI scripting is enabled or disabled for the site.
	scriptalias	plain text	The leading component (after the host name) of URLs referencing CGI scripts for this site.
mod_perl	enabled	1 (enabled) **or** 0 (disabled)	Whether mod_perl for Apache is enabled or disabled for the site.
	alias	plain text	The leading component (after the host name) of URLs referencing Perl scripts for this site.
reseller	enabled	1 (enabled) **or** 0 (disabled)	Whether reseller access is enabled or disabled for the site.
	reseller_id	plain text	The name of the reseller.
tomcat4	enabled	1 (enabled) **or** 0 (disabled)	Whether Tomcat is enabled or disabled for the site.

Table 2. Service Plan services and options I

Service	Option	Value	Description
develenv	enabled	1 (enabled) **or** 0 (disabled)	Whether GNU development tools is enabled or disabled for the site.
ssi	enabled	1 (enabled) **or** 0 (disabled)	Whether Server Side Includes (SSI) is enabled or disabled for the site.
sendmail	enabled	1 (enabled) **or** 0 (disabled)	Whether the Sendmail mail server protocol is enabled or disabled for the site.
subdomain	enabled	1 (enabled) **or** 0 (disabled)	Whether subdomains are enabled or disabled for the site.
	max	number	The maximum number of subdomains that can be created for a site. **Note: -1** indicates unlimited subdomains for a site.
	wildcards	1 (enabled) **or** 0 (disabled)	Whether subdomain wildcards are enabled or disabled for the site. **Note:** Enabling subdomain wildcards for a site (for example, example1.com), will cause the range of (sub)domain names, "*.example1.com" to be reserved for the site. No other site on the Ensim Pro server will be allowed to have the same site name.
	base	plain text	This indicates the base directory, relative to the site's file system, under which all subdomains for the site will be located. **Note:** This restriction only applies to regular subdomains. User subdomains will have their directory in **/home/*<owner>*/public_html/**
weblogs	enabled	1 (enabled) **or** 0 (disabled)	Whether Web logs is enabled or disabled for the site.

Table 2. Service Plan services and options I

Service	Option	Value	Description
vacation	enabled	1 (enabled) **or** 0 (disabled)	Whether vacation message is enabled or disabled for the site.
majordomo	enabled	1 (enabled) **or** 0 (disabled)	Whether Majordomo mailing list is enabled or disabled for the site.
sqmail	enabled	1 (enabled) **or** 0 (disabled)	Whether SquirrelMail Web-based email is enabled or disabled for the site.
frontpage	enabled	1 (enabled) **or** 0 (disabled)	Whether Microsoft FrontPage is enabled or disabled for the site.
mivamerchant	enabled	1 (enabled) **or** 0 (disabled)	Whether Miva Merchant is enabled or disabled for the site.
webalizer	enabled	1 (enabled) **or** 0 (disabled)	Whether Webalizer log analyzer is enabled or disabled for the site.
mysql	enabled	1 (enabled) **or** 0 (disabled)	Whether the MySQL database server protocol is enabled or disabled for the site.
	dbaseadmin	plain text	The user name of the site's database administrator.
	dbaseprefix	plain text	The specified string that is prefixed to the name of any database that the Site Administrator creates. **Note:** Since this prefix defaults to the first 30 characters of the domain name, we recommend that you not change this unless it conflicts with an already existing prefix.
	dbasenum	number	The number of databases that can be created by the Site Administrator.
vhbackup	enabled	1 (enabled) **or** 0 (disabled)	Whether site and user level backup is available for a site.

Table 2. Service Plan services and options I

Service	Option	Value	Description
files	enabled	1 (enabled) **or** 0 (disabled)	Whether File Manager access is enabled or disabled for the site.
Power Tools			
scriptsmgr	enabled	on (enabled) **or** off (disabled)	Whether Power Tools is enabled or disabled for the site.
<tool_name> such as formmail-4.2b	enabled	1 (enabled) **or** 0 (disabled)	Whether the tools are enabled or disabled for the site. You must specify the appropriate value for each tool.

The following section lists some examples of the service plan syntax.

Example 1

This example adds a new Service Plan called Gold. Gold uses the default Service Plan as its template, but excludes the CGI service.

```
AddPlan -p default -c cgi,off Gold
```

Example 2

This example adds a new Service Plan called Gold. Gold uses the default Service Plan as its template, but includes the aliases service.

```
AddPlan -p default -c aliases,on,aliases=\[.org,.net\] Gold
```

Modifying a Service Plan

To modify a Service Plan, use the **EditPlan** script.

Syntax

```
/usr/local/bin/EditPlan
    [-p <source plan> | -s <source site> | -d <source domain> |
   -i <source IP> | -t <source path>]
   [-c <service>,<option>=<value>,[on|off],... ...]
   [-f | --force] <target plan>
```

where:

- ***<source plan>*** is the name of an existing Service Plan you want to use as the foundation or template when editing the Service Plan. This option can be omitted, in which case the target Service Plan is used as the basis of the edit operation.
- ***<source site>***, ***<source domain>***, and ***<source IP>*** are three ways to specify a site whose configuration you want to use as the foundation or template to edit when editing the Service Plan.

  Note: If the -t option is used, the path will be **/home/virtual/*<sourcepath>*/info/current**.
- ***<service>*** is the name of the service for which you want to specify options in this Service Plan.
- ***<option>*** is optional features, if any, that you want to specify for the service.
- ***<value>*** is what the option specifies, such as a name, password, or measurement. ***<value>*** may be a single string (for example, **1** or **My User**) or a list (for example, **\[1, 2, 3\]**)
- Each service has an **enabled** option, which may be used to enable or disable the service by setting this option to **1** or **0**, respectively. Alternatively, the strings **on** and **off** may be used as abbreviations for the strings: enabled=**1** and disabled=**0**.
- ***<target plan>*** is the name of the Service Plan created as a result of the edit operation.

See *Table 2-1* for a list of the services you can change or add to a Service Plan, as well as the services' options and their values.

Removing a Service Plan

To remove a Service Plan, use the **DeletePlan** script.

Syntax

```
/usr/local/bin/DeletePlan <plan name>
```

where ***<plan name>*** is the name of the Service Plan you want to remove.

Adding a domain

To add a domain, use the **AddVirtDomain** script. This script adds a domain with the specified domain information and services. If you want to create a domain using pre-configured settings, specify the name of the corresponding Service Plan.

Important: Disabling bandwidth while adding a domain using the **AddVirtDomain** script produces erratic mail behavior. Do not disable bandwidth when you add a domain using the **AddVirtDomain** script.

Syntax

```
AddVirtDomain -p default \
-c siteinfo,domain=example.com,admin_user=myname,tpasswd=go12 \
-c ipinfo,namebased=0,ipaddrs=\[10.8.3.65\] \
-c telnet,off \
-c analog,on \
-c scriptsmgr,on,formmail-4.2b=1,gallery-1.3.4=1 \
-c users,maxusers=75 \
-c spam_filter,on \
-c mailscanner,on,scan_incoming=1,scan_outgoing=1 \
-c aspmgr,on,siteopti_add=1,siteopti_manage=1,emailmark_manage=1
```

In this example, the script adds a domain example.com with the following specifications.

- The domain uses the default Service Plan.
- The domain is called example.com.
- The Site Administrator's user name is myname, whose password is go12.
- The domain's IP address is 10.8.3.65.
- The Telnet service, originally a part of the default Service Plan is disabled.
- The Analog service, which was not originally in the default Service Plan, has been added to the domain.
- The tools, formmail-4.2b and gallery-1.3.4, which was not originally in the default Service Plan, have been enabled for the domain.
- The maximum number of users allowed on the domain is 75.
- The Mail Scanning service and the Spam Filtering service is enabled for the domain.
- The Site Optimization and Email Marketing services are enabled for the domain. Site Administrators are enabled to:
  - Subscribe to and manage the Site Optimization service
  - Manage the Email Marketing service

Enabling Power Tools for a domain

You can enable Power Tools for a site using the **scriptsmgr** option. Once you enable Power Tools, Site Administrators can install and manage these tools using the Site Administrator control panel. Each tool follows the naming convention ***<tool_name>-<tool_version>***, where ***<tool_name>*** is the name of the tool and ***<tool_version>*** is the version number of the tool. **Example.** gallery-1.3.4.

To enable Power Tools for a domain, enable the **scriptsmgr** option and the individual tools you want to enable. Individual tools that are not explicitly enabled will not be available to the site even if you enable the **scriptsmgr** option. Refer to the example for help on enabling Power Tools for a domain.

Changing the domain quota

To change the disk space allocated to a domain, use the **ChangeQuota** script.

Syntax

```
/usr/local/bin/ChangeQuota <domain name> <quota>
```

where:

- ***<domain name>*** is the name of the domain for which you want to change the disk quota.
- ***<quota>*** is the amount of disk space you want to allocate to the domain.

Modifying a domain

To modify a domain's information or Service Plan, use the **EditVirtDomain** script. The script allows you to update domain information and service settings, and enable or disable Power Tools and other services for a domain. Refer to the examples for help on usage of the script.

Syntax

```
/usr/local/bin/EditVirtDomain
    [-p <source plan> | -s <source site> | -d <source domain> |
    -i <source IP> |-t <source path>]
          [-c <service>,<option>=<value>,[on|off],... ...]
    [-f | --force]
    [<target site> | -I <target IP> | -D <target domain>]
```

where:

- ***<source plan>*** is the name of an existing Service Plan you want to use as the foundation or template to edit when editing the site. This option can be omitted, in which case the target site's current service options are used as the basis of the edit operation.
- ***<source site>***, ***<source domain>***, and ***<source IP>*** are three ways to specify a site whose configuration you want to use as the foundation or template to edit when editing the site.

  Note: If the -t option is used, the path will be **/home/virtual/*<sourcepath>*/info/current**.
- ***<service>*** is the name of the service for which you want to specify options in this Service Plan.
- ***<option>*** is optional features, if any, that you want to specify for the service.
- ***<value>*** is what the option specifies, such as a name, password, or measurement. ***<value>*** may be a single string (for example, 1 or **My User**) or a list (for example, **\[1, 2, 3\]**).
- Each service has an **enabled** option, which may be used to enable or disable the service by setting this option to **1** or **0**, respectively. Alternatively, the strings **on** and **off** may be used as abbreviations for the strings: enabled=**1** and disabled=**0**.
- ***<target site>***, ***<target IP>*** and ***<target domain>*** are three ways in which to specify the site you want to edit.

For a list of service options you can change or add to a domain, as well as the options' values, see the Service Plan services and options table on page 17.

The following section lists some examples of this syntax.

Example 1

In this example, the script changes the Service Plan of the domain example.com to Gold, disables the formmail-4.2b tool, and enables the phpnuke-6.9 tool and CGI service for the domain, not originally a part of the Gold Service Plan.

```
EditVirtDomain -p Gold -c cgi,on, -c scriptsmgr,on,formmail-4.2b=0,phpnuke-6.9=1
example.com
```

Example 2

In this example, the script changes the Service Plan of the domain example.com to Basic, disables the Spam Filtering service, disables virus scanning for outgoing emails, and enables the aliases service for the domain.

```
EditVirtDomain -p Basic -c spam_filter,off \
-c mailscanner,on,scan_outgoing=0 \
-c aliases,on,aliases=\[domainname.org,domainname.net\] \
-c aspmgr,siteopti_add=1,siteopti_manage=1,emailmark_manage=0 \ example.com
```

Disabling a domain

Disabling a domain can be useful in managing customers whose accounts are overdue or in question. To disable a domain, use the **DisableVirtDomain** script.

Syntax

```
/usr/local/bin/DisableVirtDomain <domain name>
```

where ***<domain name>*** is the name of the domain you want to disable.

Enabling a domain

To enable a domain, use the **EnableVirtDomain** script.

Syntax

```
/usr/local/bin/EnableVirtDomain <domain name>
```

where ***<domain name>*** is the name of the domain you want to enable.

Viewing a list of all the domains

To view a list of all the domains on a specific Ensim Pro server, use the **ListAllVirtDomains** script.

This script returns the following:

- The name of the domain
- Whether the domain is enabled or disabled (1/0)
- Whether the domain is name-based or IP-based
- The Site Administrator's user name and email address
- All the services and options of the Service Plan that were provided to the domain

Syntax

```
/usr/local/bin/ListAllVirtDomains
```

Removing a domain

To remove a domain from a server, use the **DeleteVirtDomain** script.

Syntax

```
/usr/local/bin/DeleteVirtDomain <domain name>
```

where ***<domain name>*** is the name of the domain you want to remove.

Disabling a service

To disable a service on a specific domain, use the **DisableVirtOption** script. This script disables a specific service, but otherwise does not change the domain's Service Plan.

Syntax

```
/usr/local/bin/DisableVirtOption <domain name> <service>
```

where:

- ***<domain name>*** is the name of the domain for which you want to disable the service
- ***<service>*** is the name of the service you want to disable as described in the Service Plan and service options table on page 17.

Enabling a service

To enable a service on a specific domain, use the **EnableVirtOption** script.

Syntax

```
/usr/local/bin/EnableVirtOption <domain name> <service>
```

where:

- ***<domain name>*** is the name of the domain for which the service is enabled.
- ***<service>*** is the name of the service you want to enable as described in the Service Plan and service options table on page 17.

Hiding a service

To hide a service, use the **hide_service** script. This script removes the service from the Ensim Pro control panel so that the Server Administrator cannot create domains with this service enabled.

Note: The services that you can hide or reveal are:

- Bind
- Miva Merchant
- MySQL
- Telnet
- Tomcat

Syntax

```
/usr/local/bin/hide_service <service>
```

where ***<service>*** is the name of the service you want to hide from the Ensim Pro control panel.

Example:

```
/usr/local/bin/hide_service bind
```

or

```
/usr/local/bin/hide_service telnet
```

Configuring service restart queues

Ensim Pro restarts services on a domain after each of the following site operations: add, update, suspend, and resume. These site operations modify the configuration file of the services. The restart operation ensures that the services operate with new or updated settings. However, restarting services after every operation result in disconnected sessions, inaccessibility to sites, and inordinate resource consumption.

Ensim Pro now provides a configuration tool **QConfigurator** that allows you to create service queues for the services that need to be restarted and configure a time interval at which these services can be restarted. It provides a command line utility to configure the **qsvcd** daemon that creates and manages the service queues. The settings are stored in the database (**qsvc_conf** table) and are retained even after migrating or upgrading to later versions.

The current release supports queued restarts for the following services:

- Apache
- MailScanner

Important: The **qsvcd** daemon must be running at all times. By default, it is disabled for each service.

The tool is located at **/usr/sbin/QConfigurator** and must be run as a **root** user.

How QConfigurator works

Each service has its own queue. You must configure the queue for each service using the **QConfigurator** tool. When you perform any operation that requires a service to be restarted, the tool adds the request to the service queue. As you perform more operations that require services to be restarted, the queue is updated. When the queue interval reaches the set threshold, the service is restarted for all the queued requests. The QConfigurator tool cuts back on frequent expensive restart operations enhancing service accessibility and ensuring better utilization of system resources. If you disable a service before a queued restart operation, the current queue is flushed and further restart operations for the service are not attempted.

You can force the **qsvcd** daemon to review the current service configuration and reset the queues. The restart requests that are queued prior to resetting are retained to be addressed in the next cycle.

Syntax

To view the list of services on the Ensim Pro server, type the following command:

```
QConfigurator -l
```

To view help on the various command options you can use, type the following command:

```
QConfigurator -h
```

To configure the service queues, type the following command:

```
QConfigurator -s <service> <command_options>
```

where:

`-s` ***<service>*** is the name of the service for which you want to configure the restart interval. The service names are case-sensitive. To configure queued restart for the apache service, type **apache**; to configure queued restart for the mailscanner service, type **MailScanner**.

<command_options> is the option you want to use for configuring the service queues. The following table lists the various options you can use with the command.

Table 3. QConfigurator command options

Option	Value	Description
-e	1 (enabled) or 0 (disabled)	If set to 1, the service is enabled. If set to 0, the service is disabled.
-q	Number	The restart interval in seconds. The restart interval should range between **300** to **86400** seconds. Values outside this range are invalid. **Note:** Type **0** to disable queued restart for a service. Once you disable, you must manually restart the service. It may be useful to set the restart interval to 0 when you perform script operations.
-r	Text	The type of restart you want to configure for the service. Each service has its own restart options. Refer to the service help for the restart options you can use.
-c	1 (enabled) or 0 (disabled)	If set to 1, the service is enabled to restart one more time. If set to 0, the restart operation is not repeated.
-x	Text	Restart the operating environment with the specified environment.
-v	None	Shows the current service configuration. It also shows the queue status for the specified service.
-a	1 (enabled) or 0 (disabled)	Autostart starts a service each time the service queue is refreshed and finds that the service is not running. Service restart requests will not be queued if Autostart is enabled. If set to 1, autostart is enabled for the services enabled. If set to 0, the autostart is disabled.

The following example sets the queue interval for the Apache service to 1000 seconds.

```
QConfigurator -s apache -q1000
```

Revealing (unhiding) a service

To reveal a hidden service, use the **unhide_service** script. This script reinstates the service in the Ensim Pro control panel.

Note: The services that you can hide or reveal are:

- Bind
- Miva Merchant
- MySQL
- Telnet
- Tomcat

Syntax

```
/usr/local/bin/unhide_service <service>
```

where ***<service>*** is the name of the service you want to reinstate in the Ensim Pro control panel.

Example:

```
/usr/local/bin/unhide_service bind
```

or

```
/usr/local/bin/unhide_service telnet
```

Adding a user to a domain

To add a user to a domain, use the **AddVirtUser** script. This script adds a user to a domain, specifying the information Ensim Pro needs to add the user, as well as the server applications to which the user should be granted access.

You can specify the user's password in either of two ways.

- By including the user's password as part of the command syntax.
- By entering the new user's password on standard input.

Syntax

```
/usr/local/bin/AddVirtUser [tpasswd=<ctxtpwd> | cpasswd=<crpwd> | passwd] \ <domain name> <username> <user's full name> <user's disk quota> \ <service>=<value>
```

where:

- ***<ctxtpwd>*** is the user's password in plain text. Use the `tpasswd` option to include the user's password in the command in plain text.
- ***<crpwd>*** is the user's password in encrypted text. Use the `cpasswd` option to include the user's password in the command in encrypted text.

  Note: To use this option, you first need to employ encryption software to encrypt the password. Enter the encrypted text in this variable.

- `passwd` Use this option to enter the user's password on the standard input. The command will not exit until you enter and confirm the password.

  Note: Use any one of the above three options to enter the user's password depending on the security level desired while entering the password.

- ***<domain name>*** is the name of the domain to which you are adding the user.
- ***<username>*** is the user's login name.

- ***<user's full name>*** is the user's first and last name. To include spaces between the first and last name, enclose the full name in "".
- ***<user's disk quota>*** is the amount of disk space you are allocating to this user.
- ***<service>*** is the server application to which you are granting the user access.

  Note: The services currently available are Telnet, ProFTPd, and SSH.

- ***<value>*** is a number, either 1 to enable or 0 to disable, the service.

```
AddVirtUser tpasswd=MTVrules example.com uname "John Doe" 20 \
telnet=1 ssh=1 proftpd=1
```

In this example, the script adds a user with the following specifications.

- The user's password entered in plain text is MTVrules.
- The domain is called example.com.
- The user's user name is uname and his full name is John Doe.
- The user was given 20 MB quota for disk space.
- The user was granted access to Telnet, ProFTPD and SSH connections.

Changing the number of users allowed on a domain

To change the maximum number of users allowed on a domain, use the **ChangeMaxUsers** script.

Syntax

```
/usr/local/bin/ChangeMaxUsers <domain name> <number of users>
```

where:

- ***<domain name>*** is the name of the domain for which you want to change the number of users.
- ***<number of users>*** is the new maximum number of users you want to specify.

Changing a user's full name

To change a user's full name, use the **ChangeFullNameVirtUser** script.

Syntax

```
/usr/local/bin/ChangeFullNameVirtUser <domain name> <username> <user's full name>
```

where:

- ***<domain name>*** is the name of the domain.
- ***<username>*** is the user's login name.
- ***<user's full name>*** is the user's first and last name. To include spaces between the first and last name enclose the full name in "".

```
ChangeFullNameVirtUser example.com Myname "Myname New"
```

Here, the script changes the user's full name with the following specifications.

- The domain is called example.com.
- The username is Myname; her new full name is Myname New.

Changing a user's information

To change a user's information, use the **ChangeInfoVirtUser** script. This script changes a user's full name and the amount of disk space allocated to the user.

Syntax

```
/usr/local/bin/ChangeInfoVirtUser <domain name> <username> <user's full name> <quota>
```

where:

- ***<domain name>*** is the name of the domain.
- ***<username>*** is the user's login name.
- ***<user's full name>*** is the user's first and last name. To include spaces between the first and last name enclose the full name in "".
- ***<quota>*** is the new amount of disk space you are allocating to this user.

```
ChangeInfoVirtUser example.com Myname "Myname New" 30
```

Here, the script changes the user's information with the following specifications.

- The domain is called example.com.
- The user's user name is Myname and her new full name is Myname New.
- The user was given 30 MB of disk space.

Removing a user from a domain

To remove a user from a domain, use the **DeleteVirtUser** script.

Syntax

```
/usr/local/bin/DeleteVirtUser <domain name> <username>
```

where:

- ***<domain name>*** is the name of the domain from which you are deleting this user.
- ***<username>*** is the user's login name.

Configuring logrotate

To configure logrotate settings, use the **logrotate_be** script

Syntax

```
logrotate_be [options]
```

Options

- `a,` --action specify the action to perform (0)

- c, --compress whether compression is enabled (1)
- d, --domain name of the domain of interest (2)
- e, --email address for sending out-of-existence logs (!)
- f, --frequency of log rotation (3)
- h, --help print this help message and exit
- m, --missingok whether missing log files is OK (4)
- n, --narchives number of archives to keep (5)
- s, --size maximum log file size for rotation (6)
- v, --version print version information and exit

Notes

- (!) Disabled in this version
- (0) One of the following strings: disable, enable, read, rotate, or write.
- (1) Either "y" or "n", indicating whether compression is off or on.
- (2) Mandatory argument.
- (3) One of the following strings: "daily4", "daily", or "weekly"—representing log rotation frequency (4 times a day, daily, or weekly).
- (4) Either "y" or "n", indicating whether missing log files is okay.
- (5) A non-zero integer with a maximum value of 5.
- (6) A non-zero integer representing maximum log size, optionally followed by "k" (for kilobytes) or "M" (for megabytes). For example, 100, 100k and 100M are valid specifications.

Getting quota reports

You can use the **quota_report** script to obtain a statistical report on the storage limits and usage of disk space allocated to "users" or "groups". When you run the script, it invokes a system call **quotactl** that returns the following quota storage and usage information, one line for each unit of information, for the specified user ID and group ID.

Each line in the output translates to the following information:

Output Line 1: User quota enabled/disabled : Group quota enabled/disabled.
Uses "1" or "0" to indicate whether the user or group quotas are enabled or disabled respectively.

Output Line 2: Current disk space occupied by quotas (in 1024-byte blocks)
Displays the current disk space occupied by quotas in blocks of 1024-bytes.

Output Line 3: Absolute limit on disk space (in Kilobytes)
The absolute or "hard" limit indicates the maximum permissible limit for disk space usage within the set time limit. On exceeding the limit or the time period, the user is prevented from using additional disk space until disk space consumption is moderated below the preferred limit.

Output Line 4: Time limit for excessive disk usage
The time, in number of minutes, available for using or allocating disk space in excess of the limit. On expiry of the time limit you will be unable to allocate or use additional disk space. The default time limit is set to one week or seven days.

Output Line 5: Number of inodes used currently
The number of inodes that are used currently.

Output Line 6: Preferred limit for inodes
The preferred or "soft" limit indicates the maximum number of inodes that can be stored on a user's allocated disk space. If you set the absolute limit and the time limit for additional inodes then the user can exceed the preferred limit for inodes until:

- The number of inodes exceeds the absolute limit set
- The time limit assigned for excessive inodes expires

Once the absolute limit or the time limit expires, additional inodes cannot be created until the user reduces disk space usage below the preferred limit. This resets the time limit assigned for additional inodes.

Important: If the absolute limit or the time limit for additional inodes is not set, then inodes in excess of the preferred limit cannot be supported.

Output Line 7: Absolute limit on inodes
The absolute or "hard" limit indicates the maximum permissible limit for the number of inodes that can be stored on the disk space within the set time limit. If the absolute limit or the time period is exceeded, no additional inodes can be created until disk space consumption is moderated below the preferred limit.

Output Line 8: Time limit for excessive inode usage
The time, in number of minutes, available for supporting additional inodes in excess of the "preferred" limit. On expiry of the time limit you will be unable to support additional inodes. The default time limit is set to one week or seven days.

Note: Ensim Pro sets and manages quotas with the following default parameters.

- The "preferred" and "absolute" quota limits are equal.
- inode limits are not set.

Syntax

```
/usr/bin/quota_report -d <directory> [-q|-u <uid>|-g <gid>]
```

Sample script and output

```
quota_report -d / -g 504
1:1
36376
512000
0
4445
0
0
0
```

Exporting complete, reseller, site, or user's data

To export complete, reseller, site, or user's data, use the **vhexport** script.

Syntax

```
/usr/local/bin/vhexport
  -h|--help
  -a|--appliance-info
  -r|--reseller reseller1[,info][,site1,site2,...,site< n>]
  -s|--sites site1,site2,...,site< n>
  -u|--users site1,user1,user2,...,user< n> ...
  -u|--users user1@domain.com,user2@domain.net,... ...
  [-U|--URL <destination URL>] [-f|--url-info-file=<file>]
  [-t|--split-threshold <size>]
  [-z|--compressed]
  [-c|--crypt] [-A|--algo=<algorithm>] [-k|--key-from-fd=<fd>]
```

Note: Individual exports will be made for each reseller, site, and user specified. Note that site exports include the necessary information to recreate users when imported. When specifying a ***<destination URL>***, the proper '.tar.gz' or '.tar' ending will automatically be added to the end of the URL, based on whether compression is set or not, respectively.

The ***<destination URL>*** may also contain format specifiers for the purpose of automatically generating URLs when one or more exports are being made with a single invocation.

The following time-related format specifiers are allowed (these are the format specifiers as understood by the server's C library **strftime()** function; see the **strftime()** manual reference for more information):

- %a Locale's abbreviated weekday name.
- %A Locale's full weekday name.
- %b Locale's abbreviated month name.
- %B Locale's full month name.
- %c Locale's appropriate date and time representation.
- %C The century number (year/100) as a 2-digit integer.
- %d Day of the month as a decimal number [01,31].
- %D Equivalent to %m/%d/%y.
- %e Like %d, the day of the month as a decimal number, but a leading zero is replaced by a space.
- %F Equivalent to %Y-%m-%d.
- %h Equivalent to %b.
- %H Hour (24-hour clock) as a decimal number [00,23].
- %I Hour (12-hour clock) as a decimal number [01,12].
- %j Day of the year as a decimal number [001,366].
- %m Month as a decimal number [01,12].
- %M Minute as a decimal number [00,59].
- %p Locale's equivalent of either AM or PM.
- %s The number of seconds since the Epoch; that is, since January 1, 1970
- %S Second as a decimal number [00,61].
- %U Week number of the year (Sunday as the first day of the week) as a decimal number[00,53]. All days in a new year preceding the first Sunday are considered to be in week 0.

- %w Weekday as a decimal number [0(Sunday),6].
- %W Week number of the year (Monday as the first day of the week) as a decimal number [00,53]. All days in a new year preceding the first Sunday are considered to be in week 0.
- %x Locale's appropriate date representation.
- %X Locale's appropriate time representation.
- %y Year without century as a decimal number [00,99].
- %Y Year with century as a decimal number.
- %Z Time zone name (or by no characters if no time zone exists).
- %% A literal "%" character.

In addition, the following format specifiers allow you to include information about the type of export being made:

- %(type) The type of export [complete, reseller, site, or user]
- %(name) The name of the export [complete, reseller username, domain name, or user@domain]
- %(host) The exporting host server.

For example, the command:

```
vhexport -u joe@mydomain.com,bob@yourdomain.net -U file:///tmp/%H_%t_%N_%Y_%B_%
```

on the server examples1.ensim.com will result in the following exports being created (assuming the date is December 23, 2002):

```
/tmp/examples1.ensim.com_user_joe@mydomain.com_2002_December_23.tar

/tmp/examples1.ensim.com_user_bob@yourdomain.net_2002_December_23.tar
```

If the -z option was specified, the resulting exports would be as follows:

```
/tmp/examples1.ensim.com_user_joe@mydomain.com_2002_December_23.tar.gz

/tmp/examples1.ensim.com_user_bob@yourdomain.net_2002_December_23.tar.gz
```

`-f|--url-info-file` specifies a file that contains additional information to be used with any URLs specified. It is generally not safe to include a password on the command line; you can use this option to point to a file that contains the password information. Note that if you choose to do this, you should make sure that the file is readable only by you. The file may contain two or three bits of information: host and user are mandatory; pass is optional.

An example file might contain:

- host ftp.ensim.com
- user ensimuser
- pass ensimpasswd

`-t|--split-threshold-size` determines how large each output file is allowed to grow before another output file is started, and may only be specified if the -U|--URL argument is specified too. The argument must be a number optionally followed by a unit; the unit may be any one of b (1), kd (1000), k (1024), md (1,000,000), m (1,024,768), gd (1,000,000,000) or g (1,073,741,824), and defaults to "m". This option is useful when exporting a large site or user, and the target server cannot handle files that are larger than a certain size.

For example, assume the site example.com would consume 3 GB of space, but the destination server cannot handle files larger than 1 GB. The command:

```
vhexport --sites example.com -U ftp://mybackupserver.com/%t_%N -t 1g
```

will result in the following files being created:

- **ftp://mybackupserver.com/site_example.com.tar**
- **ftp://mybackupserver.com/site_example.com.1.tar**
- **ftp://mybackupserver.com/site_example.com.2.tar**

When importing, only the first URL in the above list should be specified. The import process will automatically attempt to retrieve the other files.

`-c|--crypt` specifies that the resulting export data should be encrypted before being written to the destination. A symmetric cipher is used. If the -k|--key-from-fd option is not specified, you will be prompted to enter a passphrase. -k|--key-from-fd may be used to specify a file descriptor from which the passphrase may be read. If set to 0, this will be equivalent to reading from standard in. -a|--algo allows you to change the cipher algorithm used (this option is not currently supported; that is, there is only a single type of cipher). Note that you must use the -U|--URL argument to specify the export destination if you don't use the -k|--key-from-fd option.

Importing complete, reseller, site, or user's data from a backup

To import complete, reseller, site, or user's data from a backup, use the **vhimport** script.

Syntax

```
/usr/local/bin/vhimport
  -r|--role=appliance
  reseller:<reseller name or id>
  siteadmin:<domain name or site id>
  siteuser:<user@domain or user@site id>
  [-t|--target=<reseller name or id>]
   [-t|--target=<domain name or site id>]
   [-t|--target=<user@domain or user@site id>]
   [--restore-user-configs] [--restore-site-configs]
   [--restore-reseller-configs] [--overwrite-data]
   [-U|--URL=<source URL>] [-R|--recurse]
   [-f|--url-info-file=<file>]
   [-S|--services service1,...]
   [-d|--decrypt] [-A|--algo=<algorithm>] [-k|--key-from-
   fd=<fd>]
```

If the role argument is not **appliance**, you must specify additional information to identify the reseller, site administrator, or site user on whose behalf the import is being performed.

If only importing a single file, you may specify the intended target of the import. For example, if the role was set to **appliance** or **reseller**, the user would be able to import a backup made from site A into an existing site B, by specifying site B as the target. Targets may only be specified for imports where the imported entity has fewer privileges than the importing role, and the importing role must control the intended target. That is, an appliance may specify any target for reseller, site, or user imports, whereas a reseller may only specify a target for site or user imports, and the target must be a site or user under his or her control.

If a target is not specified, the target will either be assumed from the role argument or from the import data. For example, if the role was set to **reseller** and the import file was a reseller backup, then the target would be the reseller given with the role identifier. If the import file had been a site backup, then the target would be the domain described by the import file, and will be allowed as long as the specified reseller owns that site, or the site does not exist.

The **--overwrite-data** option signifies that the import should overwrite existing data. If not specified, only data present in the import but not in the target will be restored. Note that this flag does not affect reseller, site, or user configurations.

The **--restore-user-configs** option signifies that if any user imports are encountered, and the user currently exists, then the user's configuration will be set to the configuration saved in the archive. Without this option, an existing user's configuration will not be changed.

The **--restore-site-configs** option signifies that if any site imports are encountered, and the site currently exists, then the site's configuration will be set to the configuration saved in the archive. Without this option, an existing site's configuration will not be changed.

The **--restore-reseller-configs** option signifies that if any reseller imports are encountered, and the reseller currently exists, then the reseller's configuration will be set to the configuration saved in the archive. Without this option, an existing reseller's configuration will not be changed.

The **--restore-all-configs** option is equivalent to specifying --restore-user-configs --restore-site-configs --restore-reseller-configs together.

The **--recurse** option will cause any imports made at the same time as one of the imports specified on the command line to be imported as well. For example, if a user exported everything on the server (complete, resellers, sites) in a single invocation of vhexport, then all the user would have to do to completely recover the system is import the complete backup by specifying the --recurse option.

-f|--url-info-file specifies a file that contains additional information to be used with any URLs specified. It is generally not safe to include a password on the command line; you can use this option to point to a file that contains the password information. Note that if you choose to do this, you should make sure that the file is readable only by you. The file may contain two or three bits of information; host and user are mandatory, and pass is optional. An example file might contain:

- host ftp.ensim.com
- user ensimuser
- pass ensimpasswd

-c|--crypt specifies that the import data is encrypted and must be decrypted before being restored. If the -k|--key-from-fd option is not specified, you will be prompted to enter a passphrase. -k|--key-from-fd may be used to specify a file descriptor from which the passphrase may be read. If set to 0, this will be equivalent to reading from standard in. -a|--algo allows you to change the cipher algorithm used (this option is not currently supported; that is, there is only a single type of cipher). Note that you must use the -U|--URL argument to specify the import source if you don't use the -k|--key-from-fd option, or if you set the -k|--key-from-fd option to 0.

Backing up the postgresql database

To back up the postgresql database, you need to run the script **SetPgCron**. The SetPgCron script enables or disables the cron job **backupdb**. The cron **backupdb** backs up the postgresql database located at **/var/lib/pgsql/data** as a postgresql user and runs every day at 5 A.M. When you run the script, you can set the number of archives you want to retain and the path where you want to back up the database. No default path is assumed by the cron.

> **Important:** The specified archive path must be a path on which the postgresql user has write permissions, otherwise the script fails with an error.

The files are backed up in the format **db_dump_*<archiveno>*.gz**, where ***<archiveno>*** is the nth archive in the list of archives created by the cron. On reaching the specified archive limit, the next backup overwrites the first archive.

The format of the script is as follows:

```
SetPgCron --enable -r<no. of archive> -p<backup path>
```

The following table explains the options.

Table 4. Script options

Options	Description
--enable	Indicates that the cron **backupdb** is enabled to run as scheduled
--disable	Indicates that the cron **backupdb** is disabled
--rotation | -r	Indicates the number of archives that must be retained before the cron replaces the oldest archive with a new backup
--path | -p	Indicates the complete path where you want to back up the database. No default path is assumed; if not specified, the script fails with an error.

In the following example, the script enables the cron job to back up the postgresql database at **/tmp** and retains 3 archives.

```
SetPgCron --enable -r3 -p/tmp/
```

Configuring recursive DNS settings

Recursion is a process by which a DNS server contacts the network of DNS servers to resolve a request. DNS servers enabled to perform recursion first contact the root servers, which respond with the names of top-level DNS servers, which may in turn provide a referral to the next level of DNS servers. This recursive search continues till the authoritative name server for the request is located. DNS servers that are not enabled to perform recursion respond with whatever data is locally available.

Ensim Pro is bundled with the BIND protocol, which enables it to act as a name server. You can modify the recursion settings of your local DNS server by running the script **recursivedns**.

Using the script, you can:

- View the current recursion settings for the server

- Enable or disable recursion on the server
- Configure the server to accept or deny recursion requests

The script ends with one of the following exit codes:

- **Code 1.** Indicates that DNS recursion is enabled for the name server.
- **Code 0.** Indicates that DNS recursion is disabled for the name server.

Syntax

```
recursivedns [on/off] [+all/-all] [+<ip_address_server1>] [+<ip_address_server2>] [-<ip_address_server3>] [-<ip_address_server4>] [--help]
```

Viewing the current recursion settings

You can view the current recursion settings for your local DNS server by typing the following command:

```
recursivedns
```

Enabling recursion on the local DNS server

A DNS server enabled to perform recursion first contacts the root servers, which respond with the names of top-level DNS servers, which may in turn provide a referral to the next level of DNS servers. This recursive search continues till the authoritative name server for the request is located.

You can enable the local DNS server to perform recursive lookups by typing the following command:

```
recursivedns on
```

Disabling recursion on the local DNS server

A DNS server that is not enabled to perform recursion responds with locally available data.

You can disable recursion on the local DNS server by typing the following command:

```
recursivedns off
```

Accepting recursion requests from clients

You can configure a server to perform recursion for a specific client or for all clients that contact the server for lookups. When a request is received, the server uses its own cache or queries other name servers to resolve the request.

To accept recursive requests from a specific client, type the following command:

```
recursivedns +<ip_address_server>
```

Note: You must specify the IP address of the server for which you want to enable recursion.

To accept recursive requests from all clients that contact the server, type the following command:

```
recursivedns +all
```

Denying recursion requests from clients

You can configure a server to refuse recursive requests from a specific client or from all clients that contact the server for lookups. When a request is received, the server responds with locally available data or forwards the request to another server.

To deny recursive requests from a specific client, type the following command:

```
recursivedns -<ip_address_server>
```

Note: You must specify the IP address of the server for which you want to refuse recursion.

To deny recursive requests from all clients that contact the server, type the following command:

```
recursivedns -all
```

Synchronizing the site file system with RPM updates

Ensim Pro creates a Virtual Private File System (VPFS) for each site created. When you create a site, the file system for that site is populated with a default set of service files from a template. When a service RPM is upgraded, the corresponding service files present in the sites' file systems become stale as they are not automatically updated. You must synchronize the files in the site file systems with the RPM updates either by scheduling a periodic update using the **scheduleMaintenance** script or by forcing an update using the **synchronizeFST** script as described in the following sections.

Scheduling file system updates

When the sites' file systems are updated with recent RPM updates, Ensim Pro runs in maintenance mode. During the maintenance mode, the sites hosted on the server are inaccessible. It is recommended that you schedule this maintenance during a period of low activity.

You can schedule the maintenance to occur periodically to ensure that the sites' file systems are updated with the latest files. You can schedule this update process by running the **scheduleMaintenance** script as the root user. The **scheduleMaintenance** script schedules the execution of the **synchronizeFST** script by updating the cron file with the schedule settings you specify for the **scheduleMaintenance** script. You can view the existing schedule information in the file **/etc/cron.d/scheduleMaintenance**.

The following table lists the options you can use with the script.

Table 6. Script options (scheduleMaintenance)

Options	Description / Usage
-h | --help	Displays this help message

Table 6. Script options (scheduleMaintenance)

Options	Description / Usage
-H *<hour>* --hou*r=<hour>*	Schedules the maintenance to run at the specified hour (0-23) of the day. The option can be used with any of the following frequency values: **daily**, **weekly**, and **monthly**. If not specified, the value defaults to **0** (12.00 A.M).
-M *<minute>* --minute=*<minute>*	Schedules the maintenance to run at the specified minute (0-59) past the specified hour. The option can be used with any of the following frequency values: **daily**, **weekly**, and **monthly**. If not specified, the value defaults to **0** (minutes past the hour).
-W *<day>* --day-of-week=*<day>*	Schedules the maintenance to run on the specified day of the week. The option can be used only with the following frequency value: **weekly**. *<day>* can be the full name of the week day, such as Sunday or a value from 0 to 6, each value indicating a particular day of the week as specified below: Sunday = 0, Monday = 1, Tuesday = 2, Wednesday = 3, Thursday = 4, Friday = 5, Saturday = 6
-D *<day>* --day-of-month=*<day>*	Schedules the maintenance to run on the specified day (1-31) of the month. The option can be used only with the following frequency value: **monthly**.
-e *<email>* --email=*<email>*	Specifies the email address at which you want to receive the output of the cron job. By default, the root user receives the email at the specified email address.
<frequency>	Specifies the number of times the maintenance must run. The value can be one of the following: daily, weekly, or monthly. If the option **cancel** is used, any existing schedule is cancelled.
cancel	Indicates that any existing schedule will be cancelled.

Syntax

```
scheduleMaintenance <options> <frequency> | cancel
```

where:

<options> is any of the options listed in the above table

<frequency> is the number of times you want to schedule maintenance (daily, weekly, monthly)

The following examples illustrate the usage of the script.

To view the current schedule:

```
scheduleMaintenance
```

To schedule a weekly maintenance

```
scheduleMaintenance weekly
```

To schedule a maintenance on Monday every week:

```
scheduleMaintenance -W Monday weekly
```

Manually synchronizing the sites' file systems

You can force the synchronization process to occur after an RPM update by running the **synchronizeFST** script. The script causes Ensim Pro to go through the maintenance cycle for the synchronization process. When Ensim Pro runs in maintenance mode, the sites hosted on the server are inaccessible.

The following table lists the options you can use with the script.

Table 8. Script options (synchronizeFST)

Option	Description / Usage
-h --help	Displays this help message
-v --verbose	Prints verbose debugging information
-p | --previous	Displays the time when the script was last executed successfully. The time stamp of a failed or aborted run of the script is not recorded.
-l | --listrpms	Lists the RPMs that were updated since the script was last executed successfully.
-t ***<time_format>*** --touch=***<time_format>***	Indicates the time you want the script to assume for the last successful run. ***<time_format>*** is the time stamp you force the script to display as the last successful run. This option is useful for diagnostics, when you want the script to believe that it was last run at the time specified by ***<time_format>***. It can hold the special value **now**, or it can hold the time since epoch in seconds (ticks), or it can hold a reasonably formatted time stamp, such as **04 January 2006**.
-R ***<format>*** --rpmformat=***<format>***	Indicates the format in which you want RPM names to be displayed. Use the option when you want to change the default display format of an RPM name, which is **%(name)s %(version)s** (for example, spamassassin-3).
-T ***<format>*** --timeformat=***<format>***	Indicates the format in which you want the time to display for all time-related information. Use the option when you want to change the default format, which is **%a %b %d %Y, %I:%M:%S %p %Z**.(for example, Tue May 09 2006, 12:00:00 AM PST)

Table 8. Script options (synchronizeFST)

Option	Description / Usage
-S | --synchronize	Synchronizes the sites' file systems with the latest RPM updates. **Note:** This option causes Ensim Pro to enter the maintenance mode for synchronization if any RPM has been updated since the last run. The sites hosted on the server will be inaccessible until Ensim Pro exits the maintenance mode.

Syntax

```
synchronizeFST <options>
```

where ***<options>*** is any of the options listed in the script options table on page 46.

The following examples illustrate the usage of the script.

To view the time when the script was last run successfully:

```
synchronizeFST -p
```

Displays the output as: `Tue May 09 2006, 12:00:00 AM PST`

To change the default format of the time stamp and view the time when the script was last run successfully:

```
synchronizeFST -p -T '%A'
```

Displays the output as `Tuesday`

Using the Reseller Command Line Interface scripts

This section explains the usage and syntax of scripts you can use to manage reseller accounts.

You can use command line scripts for:

- *Viewing the list of reseller accounts*
- *Creating a reseller account*
- *Modifying a reseller account*
- *Removing a reseller account*

Refer to the following table for the complete list of services, options, and values that you can specify for a reseller account.

Note: You must specify values for services or options that you want to enable; no default values are set other than those explicitly specified in the table. Further, you cannot leave an option blank.

The following table lists the services, options, and values that you can specify for a reseller account.

Table 10. Services and options - II

Service	Option	Value	Description
apache	jail	1 (enabled) or 0 (disabled)	If set to 1, a reseller can only create “jailed” sites (sites that restrict the Apache daemon to the chrooted portion of the site’s file system). If set to 0, a reseller can create “jailed” or “non-jailed” sites.
ssh	jail	1 (enabled) or 0 (disabled)	If set to 1, a reseller can only create sites where the SSH remote access service is “jailed” (remote shell access to the site using SSH is restricted to the chrooted portion of the site’s file system). If set to 0, a reseller can create sites with “jailed” or “non-jailed” SSH access.
telnet	jail	1 (enabled) or 0 (disabled)	If set to 1, a reseller can only create sites where the Telnet remote access service is “jailed” (remote shell access to the site using Telnet is restricted to the chrooted portion of the site’s file system). If set to 0, a reseller can create sites with “jailed” or “non-jailed” Telnet access.
bandwidth	units	B or b (Bytes) KB or kb (Kilobytes) MB or mb (Megabytes) GB or gb (Gigabytes)	The unit for measuring the rate of data transmission. The default is B.
	threshold	number	The total bandwidth (in the selected units) that can be assigned to sites created by a reseller.
diskquota	units	B or b (Bytes) KB or kb (Kilobytes) MB or mb (Megabytes) GB or gb (Gigabytes)	The unit for measuring the disk space allocated to a reseller. The default is B.
	quota	number	The total disk space (in the selected units), that can be assigned to sites created by a reseller.
ipinfo	ipbased	number	The maximum number of IP-based sites that can be created by a reseller.
	namebased	number	The maximum number of name-based sites that can be created by a reseller.
reseller	username	plain text	The user name of the reseller

Table 10. Services and options - II

Service	Option	Value	Description
	fullname	plain text	The full name of the reseller
	email	plain text (for example, admin@example.com)	The email address of the reseller
	passwd	plain text	The reseller's password. The reseller will be prompted for the plain text password.
	tpasswd	plain text	The reseller's password. Specify the password by typing the following at the command line prompt: `tpasswd=`***< plain text password>***
	cpasswd	encrypted text	The reseller's password. Specify the password by typing the following at the command line prompt: `cpasswd=`***<encrypted text password>***
	enabled	1 (enabled) or 0 (disabled)	Indicates whether the reseller account is enabled or disabled.
users	maxusers	number	The total number of user accounts for sites created by a reseller.

Viewing the list of reseller accounts

To view the list of reseller accounts, use the **ListResellers** script. The script returns the list of resellers, one line of text for each reseller, with the specified account information, services, enabled options, and the corresponding values. In addition to the services and options enabled for the reseller, the list will also display the following information for the specified service.

Table 14. Additional information displayed in the script output

Service	Attribute	Value	Description
diskquota	alloc_quota	number	The actual amount of disk space allocated to the sites created by the reseller.
bandwidth	alloc_threshold	number	The actual amount of bandwidth allocated to the sites created by the reseller.

Table 14. Additional information displayed in the script output

Service	Attribute	Value	Description
ipinfo	alloc_ipbased	number	The actual number of IP-based sites created by the reseller.
	alloc_namebased	number	The actual number of name-based sites created by the reseller.
users	alloc_maxusers		The number of users that can be created for all the sites by the reseller.
reseller	reseller_id	number	The reseller ID of the reseller.

Note: The password of a reseller will not be listed in the script output. Some values may have backslash-quoted characters (for example, spaces in the reseller full name). Also, note that bandwidth and quota values are returned in bytes.

Syntax

```
/usr/local/bin/ListResellers
```

A sample output produced by the **ListResellers** script

```
apache,jail=0 users,maxusers=100,alloc_maxusers=0 ssh,jail=0
diskquota,alloc_quota=0,quota=524288000
reseller,username=Myname,fullname=Myname,enabled=1,reseller_id=4,
email=myname@example.com
ipinfo,alloc_ipbased=0,namebased=10,alloc_namebased=0,ipbased=10
```

Note: The sample output used in the example has been word-wrapped for readability. The actual output is displayed in a single line, with the various services separated by a single space.

In this example, the script lists a reseller with the following account information:

- User name: Myname
- Full Name: Myname
- Reseller ID:4
- Email: myname@example.com
- Apache, jail option: disabled
- Disk quota, allocated: 524288000 bytes, used: 0
- Users, allocated: 100, used:0
- IP-based sites, allocated: 10, used=0
- Name-based sites, allocated: 10, used=0

Creating a reseller account

To create a reseller account, use the **AddReseller** script. If you want to create reseller accounts that have a common set of services, you can use an existing reseller account as a template for the new reseller.

For the list of services, with corresponding options and values, that you can enable for the reseller account, see the Service Plans and service options table. On successful creation of the account, the message, `Reseller <reseller name> successfully created`, is displayed.

Syntax

```
/usr/local/bin/AddReseller [ -r <source reseller> ] [ -c <service>, <option>=<value>, ...
...]
```

where:

- ***<source reseller>*** is the name of an existing reseller account whose account information you want to use as a template for the new reseller account. If you do not specify the ***<source reseller>***, all the services and corresponding options of all services mentioned in the services and options table on page 47 are required.
- ***<service>*** is the name of the service for which you want to specify options for the reseller.
- ***<option>*** is the arguments or parameters that you want to specify for the reseller.
- ***<value>*** is what the option specifies, such as a name, password, or measurement.

Creating a reseller account without using a template

```
AddReseller -c apache,jail=0 -c bandwidth,threshold=10485760000 -c
users,maxusers=100 -c ssh,jail=0 -c diskquota,quota=524288000 -c
reseller,username=Myname,fullname=Myname,enabled=1,
passwd,email=myname@example.com -c telnet,jail=0 -c ipinfo,namebased=10,ipbased=10
```

In this example, the script adds a reseller account with the following account information:

- Apache; jail option disabled
- Bandwidth; allocated: 10485760000
- Users; allocated: 100
- SSH; jail option disabled
- Disk quota; disk space allocated: 524288000
- User name: Myname
- Full name: Myname
- Email address: myname@example.com
- Telnet; jail option disabled
- IP-based sites; allocated:10
- Name-based sites; allocated:10

Creating a reseller account using a template

```
AddReseller -r reseller1 -c reseller,username=reseller2
```

In this example, the script creates a new reseller account, with the user name **reseller2**, using another reseller account, **reseller1**, as the template on which to base the account information.

Modifying a reseller account

To modify a reseller account, use the **EditReseller** script. For the list of services, with corresponding options and values, see the services and options table on page 47. On successful modification, the message, `Reseller <reseller_name> successfully updated`, is displayed.

Syntax

```
/usr/local/bin/EditReseller [ -r <source reseller> ] [ -c <service>,<option>=<value>,...
...] <reseller username>...
```

where:

- ***<source reseller>*** is the name of an existing reseller, whose account information you want to use as a template for the reseller account. If you do not specify ***<source reseller>***, all the options default to the current values specified for the reseller.
- ***<service>*** is the name of the service for which you want to specify options for the reseller.
- ***<option>*** is arguments or parameters that you want to specify for the reseller.
- ***<value>*** is what the option specifies, such as a name, password, or measurement.
- ***<reseller username>*** is the existing user name of the reseller account you want to modify.

```
EditReseller -c reseller,enabled=1,username=new_reseller, -c
ipinfo,ipbased=15,namebased=15 old_reseller
```

In this example, the script modifies the user name and site information for an existing reseller account.

Removing a reseller account

To remove a reseller account, use the **DeleteReseller** script.

Syntax

```
/usr/local/bin/DeleteReseller <reseller username>
```

where

<reseller username> is the user name of the reseller you want to remove

```
/usr/local/bin/DeleteReseller reseller1
```

In this example, the script removes the reseller account with the user name, reseller1.

CHAPTER 3

Customizing Ensim Pro

Introduction

This chapter provides information and instructions on customizing Ensim Pro.

Important: Ensim Pro relies on certain RPMs for the operation of its control panel and for virtualization. We recommend that you consult Ensim Support before you upgrade these RPMs. Refer to the standard packages table on page 54 for the list of non-customizable RPMs.

Some of the examples given below include the shell primitive \ which is used to enter a new line without the shell executing the command. This is included for reasons of clarity and can be omitted if your command fits in one line.

In this section:

Customizing the Ensim Virtual Private File System

Ensim Pro creates a Virtual Private File System (VPFS) for each site created. When a site is created, the file system for that site is populated with a default set of services from a template. This is done using hard links.

Earlier versions of Ensim Pro contained a pre-built VPFS template that contained a copy of every service possible. This template could only be changed through an upgrade to the Ensim Pro version released by Ensim. In Ensim Pro, the template mechanism has been enhanced. It no longer contains a copy (that is, binaries) of the service. Instead the template contains a filter to select services from the root file system. This template is used during site creation and modification to dynamically generate the VPFS for any site. You can now modify the template without requiring a software upgrade from Ensim.

The following are the benefits of this change:

- The size of the template (**virtualhosting-fst-*<service>*.rpm**) is much smaller.
- Using this mechanism, service updates (such as security fixes) can be quickly installed at the root level and then applied to all existing sites.
- You can modify the template to deploy additional services to each site.

Procedure to update RPMs

In order to synchronize updated services (RPMs) into all site file systems, you need to take Ensim Pro through its "maintenance mode" state which causes it to update the sites. Run the following steps:

1. Upgrade, or reinstall, any RPM.
2. Run the following commands as root:
   - `set_pre_maintenance`
   - `set_maintenance`
   - `set_post_maintenance`
   - `service epld restart`

Advanced usage: If you want to add a new RPM to any service, edit the appropriate **.sh** file for that service in **/etc/virtualhosting/filelists/*<servicename>*.sh**. Follow the instructions included in that file. Ensure that you know exactly what you are doing when manipulating these files. After editing the file, make Ensim Pro go through maintenance mode (See Step 2 above).

Use of standard packages

Earlier versions of Ensim Pro customized a number of key services to resolve issues, enhance security, and improve the performance of hosting domains. While these customizations provided a superior hosting environment, they also impacted flexibility in managing these services and increased the latency period for releasing critical service security patches.

Ensim Pro removes customizations made by Ensim to critical services enabling self-managed deployment of services. However, Ensim Pro retains customizations to certain services. Since these are no longer available or supported, Ensim is providing these as a value added enhancement.

The services that are customized are those that are originally available in the tarball format. Ensim Pro uses the RPM format for managing services. However, certain services are released by vendors only in the tarball format requiring Ensim Pro to convert these services to the RPM format and thus customize them.

However, the source RPMs of these services (containing Ensim customizations) will be made available for download and use to customers whenever a security patch is released by the service vendor, prior to the official release from Ensim. This enables you to use the source RPMs and self-create the RPM packages for installation on to an Ensim Pro server as soon as a security patch for the service is released. Note that when you self-upgrade RPMs with a release from the service vendor, you must restart the service after the upgrade to ensure effective functioning of the service.

The following table provides the list of customized and non-customized RPMs.

Table 18. List of customized and non-customized RPMs

List of standard packages used (non-customized RPMs)	
analog	xinetd
apache 2.0	net-snmp
gettext	quota
imap	sendmail
mod_perl (Apache 2.0)	mysql
mod_ssl (Apache 2.0)	gcc
mx	make
openssh	perl
openssl	php
postgresql	proftpd
python2	telnet
webalizer	
Name of the customized RPM	**Why you should not customize the RPM**
cronolog	• Converted the tarball into an RPM • Resolved issues concerning the date and time format of log file names
frontpage	Converted the tarball into an RPM
majordomo	• Modified the configuration file • Included security fixes • Modified the build process so as to enable it to be built by non-root users
mod_jk	Modified for improved integration with Apache
perl-Quota	Changed to support quotas on multiple Linux kernel versions
apache-mod_fastcgi	Resolved session timeout issues

poprelay	Created an RPM for the script
phpMyAdmin	Converted the tarball into an RPM
squirrelmail	Converted the tarball into an RPM
vacation	Converted the tarball into an RPM
mivaempressa	Converted the tarball into an RPM
mivamerchant_unl	Converted the tarball into an RPM
tomcat	Converted the tarball into an RPM
halcyon	Converted the tarball into an RPM

Customizing the Ensim Pro configuration file

The Ensim Pro configuration file **epl.conf** contains a set of directives that defines the settings for Ensim Pro. For more information, see the list of directives table on page 56. To modify the settings of these directives, see Modifying the configuration file on page 60.

List of directives

The following table lists the various directives in the configuration file.

Table 20. List of directives

Name of the directive	Type / Default Value	Significance / Notes
virtual_root_path	Type: String Default Value: **/home/virtual**	The base directory for the file system of all the sites on the server. Do not modify this value.
logfile	Type: String Default Value: **/var/log/ensim/epl.log**	The file where the backend messages from the control panel are saved.
import_export_priority	Type: Integer Default Value: 10	The numeric value indicating the priority of the Export/Import process. This is set to 10, so that the Export/Import process has a lower priority while running to avoid overloading the system. Do not change this value unless you really want to assign a high priority to the process. A negative value causes the export/import process to run with elevated priority overloading your system.

Table 20. List of directives

Name of the directive	Type / Default Value	Significance / Notes
redirect_protocol	Type: String Default Value: https	The protocol used by the control panel while redirecting Web pages. The directive is used if you access the control panel using the HTTPS protocol, otherwise, it is ignored. Do not modify this value.
redirect_port	Type: Integer Default Value: 19638	The secure port on which Ensim Pro responds to HTTPS requests. If you change this value, make sure that the welcome page **index.shtml** located at **/var/www/html/** and **/var/www/html/admin/** are modified to redirect requests to the new port.
gui_logfile	Type: String Default Value: **/var/log/ensim/eplcp.log**	The file where messages from the control panel (frontend) are recorded.
gui_pidfile	Type: String Default Value: **/var/run/eplcp.pid**	The file containing the process ID (PID) of the running control panel process. Do not change this value.
server_admin	Type: String Default Value: root@localhost	The email address of the Server Administrator as indicated by the HTTP directive `ServerAdmin`, in the configuration file **eplhttpd.conf**. See http://httpd.apache.org/docs/2.2/mod/core.html#serveradmin for details.
server_phpmyadmin	Type: String Default Value: The host name of the Ensim Pro server	The server on which phpMyAdmin is installed. Typically, phpMyAdmin runs on the Ensim Pro server, so you don't need to modify this value. However, modify the directive if you access the control panel using a host name or IP Address that is different from the Ensim Pro server, for example, when you access the control panel from a remote location, and the host name / IP address undergoes network address translation. The URL eventually displayed by Ensim Pro is **http://*<server_phpmyadmin>*/MyAdmin/index.php**

Table 20. List of directives

Name of the directive	Type / Default Value	Significance / Notes
server_domainpreview	Type: String Default Value: The host name of the Ensim Pro server	The host name of the Ensim Pro server on which a domain can be previewed. Ensim Pro allows a domain to be accessed using **http://*<server_name>*/*<domain_name>*/**. Modify this value to a different host name or IP address if you expect to preview the domain using **http://*<new_host_name>*/*<domain_name>*/** or **http://*<new_ip_address>*/*<domain_name>*/**
maxRequestBodySize	Type: Integer Default Value: 100 * 1024 * 1024 (100 MB expressed as bytes)	The maximum size (in bytes) of a file that can be uploaded using the File Manager service in Ensim Pro.
cacheRefreshPolicy	Type: Boolean Default Value: True	**True:** The Ensim Pro server forces the client browsers to reload certain resources every time a page is accessed and to validate certain pages every time it is accessed. This is to ensure that the browser displays customized settings such as skin preferences without requiring the user to manually clear the browser's cache. This also means more HTTP traffic, since the browsers are not allowed to cache resources aggressively. Set this to False if you do not expect your users to change their preferences often. **False:** The Ensim Pro server allows client browsers to cache resources to reduce HTTP traffic. See also *cacheMustReload*, *cacheMustVerify*, and *cacheVerificationDate* directives.

Table 20. List of directives

Name of the directive	Type / Default Value	Significance / Notes
cacheMustReload	Type: String Default Value: .css$	If the *cacheRefreshPolicy* directive is set to True, this directive indicates that certain resources must be reloaded every time a page is accessed. Notes: • This string is expected to be a valid Python regular expression. See http://docs.python.org/lib/re-syntax.html for syntax details. • Use this variable with care. The browser is not allowed to cache any resource whose URL matches this expression; it is fetched by the browser every time the page loads. • As the default value indicates, all the css files (HTML stylesheets) are loaded each time to ensure a consistent experience while changing skin preferences.
cacheMustVerify	Type: String Default Value: (.gif|.jpg|.jpeg|.png|.bmp)$	If the *cacheRefreshPolicy* directive is set to True, this directive indicates that certain resources must be revalidated every time a page is accessed in the control panel. Notes: • This string is expected to be a valid Python regular expression. See http://docs.python.org/lib/re-syntax.html for syntax details. • Use this variable with care. The browser is forced to revalidate each time any resource whose URL matches this expression. The browser may choose to serve these pages from the cache, if the resource has not changed. • As the default value indicates, all the image files(GIF, JPEG, PNG, BMP) are re-validated every time to ensure that new images of a skin are loaded by the browser whenever the skin changes.

Table 20. List of directives

Name of the directive	Type / Default Value	Significance / Notes
cacheVerificationDate	Type: String Default Value: Tue, Jan 04 1977 00:00:00 GMT	The date when the contents of the browser's cache are reviewed for validity. HTTP browser caching works on the principle of URL expiration dates wherein the browser reuses the cache contents till the specified expiration date without sending a request to the server. This date is sent by Ensim Pro for all the resources that match the pattern in the cacheMustVerify directive. To force a re-validation, an expiration date which has been exceeded, should be returned by the server.
honourLookAndFeel	Type: Boolean Default Value: False	**True:** When a higher level administrator auto-logs in to a lower level of the control panel, the look and feel of the lower level control panel is retained. **False:** When a higher level administrator auto-logs in to a lower level of the control panel, the look and feel of the higher level is retained at the lower level.

Modifying the configuration file

You can modify the configuration file **epl.conf** using the command line utility **eplconf** located at **/usr/local/sbin/**. To modify the configuration file, you must log into the Ensim Pro server as a root user.

The default top-level configuration file is located at **/etc/ensim/** and is overwritten during an upgrade. Do not modify this file. Your customizations are placed in the **epl.conf** file located at **/etc/appliance/customization/**. These changes are preserved during an upgrade. All the changes you make to **epl.conf** using the **set** and **unset** operations affect this file. This file is included in the top-level file **/etc/ensim/epl.conf**.

Important: Do not manually edit the configuration file. Use the command line utility **eplconf** to modify the file.

Viewing the list of directives

To view the list of directives, type one of the following commands.

Syntax

```
eplconf
```

or

```
eplconf get
```

Retrieving the settings of a directive

To view the current settings of a directive, type the following command.

Syntax

```
eplconf get <directive_name>
```

where ***<directive_name>*** is the name of the directive

Modifying a directive

Directive values or settings belong to a certain data type such as integer, string, or boolean. Make sure that the setting you configure for a directive holds a compatible value. The changes you make to a directive are placed in the file **/etc/appliance/customization/epl.conf** and are preserved during an upgrade.

To modify the current setting of a directive, type the following command.

Syntax

```
eplconf set <directive_name> <directive_value>
```

or

```
eplconf set <directive_name>=<directive_value>
```

where ***<directive_name>*** is the name of the directive and ***<directive_value>*** is the new value for the directive

Reverting a directive to its default value

You can revert a directive to its default settings by using the **unset** option of the command. The changes you make to a directive are placed in the file **/etc/appliance/customization/epl.conf** and are preserved during an upgrade.

To revert the settings, type the following command.

Syntax

```
eplconf unset <directive_name>
```

where ***<directive_name>*** is the name of the directive

Customizing Ensim Pro for NAT

NAT is short for network address translation. It is a technique in which the source and/or destination addresses of IP packets are rewritten as they pass through a router or firewall. It is most commonly used to enable hosts on a private network to access the Internet using a single public IP address. Note that Ensim Pro does not translate the IP address of the server. You must self-configure your hosting network to provide appropriate IP translation for your server.

By default, NAT is easily recognized by all relative links used in Ensim Pro. Once you log in to the control panel using the address **http://*<servername>*:8080/**, all the URLs (which are typically hyperlinks or form actions), such as **/isp/listsites** and **/isp/listplans** are relative URLs. In such situations, the browser automatically prefixes the protocol **http://**, the server name, and the port **8080**, to the URL before accessing the link. For example, if the browser sees a hyperlink called **/isp/listsites** on a page **http://< server>:8080/isp/main**, then the URL is automatically set to **http://server:8080/isp/listsites**.

However, Ensim Pro does not support automatic reconfiguration for the following scenarios:

- SSL tunnel redirection in Ensim Pro on page 62
- External links in Ensim Pro on page 62

SSL tunnel redirection in Ensim Pro

The principal need for an SSL tunnel is when a client wishes to securely communicate with a non-secure daemon. In this case, a middle layer is required, which will negotiate the encryption parameters (public key/certificate) with the client, and will communicate with the non-secure daemon in a non-secure way, after decrypting the data that was sent by the client. While earlier versions of Ensim Pro used **stunnel**, a universal SSL tunnel wrapper, it now uses a more powerful redirection using Apache and mod_rewrite.

The problem with the new approach is that the configuration file **eplhttpd.conf** actually contains the IP address of the server to which it should communicate when using the non-secure port. In a NAT environment, if you are accessing Ensim Pro from outside your network, the IP address of the server, for example, **https://1.2.3.4:19638/isp/**, may get translated to an internal IP address, **https://10.12.3.4:19638/isp/**. In this case, although the HTTPS URL contains the IP address 1.2.3.4, the Ensim Pro daemon **eplhttpd** should fetch the non-secure page from **10.12.3.4**. Similarly, any absolute links that refer to the same Ensim Pro server should refer to **1.2.3.4**, as that is the IP address from which the server will be accessed.

To ensure successful secure connections to the Ensim Pro interface, you must modify the **eplhttpd_ipaddress** directive in the configuration file **/usr/lib/ensim/frontend/httpd/conf/eplhttpd.conf** as required. The default vale of this directive is set to the server's IP address.

External links in Ensim Pro

In the NAT context, the term external links refers to any hyperlink or HTTP redirection URLs, that are absolute, as opposed to relative. To ensure successful resolution of these URLs, you must modify the appropriate directive in the Ensim Pro configuration file **epl.conf**. the following table lists all the links that use the name or IP address of the server.

Table 23. List of links that use the server name/IP address

Name of the link	File(s) affected	Description	Configuration item	Default value
Non-secure IP address of the Ensim Pro interface	**/usr/lib/ensim/frontend/httpd/conf/eplhttpd.conf**	The right-hand side of the Rewrite Rule indicates the server on which the non-secure UI (port 8080) is running	eplhttpd_ipaddress	IP address of the server
phpMyAdmin link in the Server Administrator control panel	None	This hyperlink is displayed in the Server Administrator control panel for the MySQL option as **http://*<server_phpmyadmin>*/MyAdmin/index.php**	server_phpmyadmin	Name of the server
/admin/ and /user/ redirects for a site	/etc/httpd/conf/virtual/< *siteN*>	When you type **http://*<sitename>*/admin/** or **http://*<sitename>*/user/**, you are redirected to the Site Administrator URL (**https://*<eplhttpd_ipaddress:19638>*/siteadmin/?ocw_login_domain=site**) or the User Administrator URL (**https://*<eplhttpd_ipaddress:19638>*/siteuser/?ocw_login_domain=site**) for that site.	eplhttpd_ipaddress	IP address of the server
Domain preview	None	Ensim Pro allows a domain to be accessed using **http://*<server_name>*/*<domain>*/**. The domain preview URL appears in the Site Administrator control panel, under the Configuration option.	server_domainpreview	Name of the server

Modifying the directives for NAT

To modify the directives, refer to Modifying the configuration file on page 60.

Customizing domains

Ensim Pro creates a chroot (change root) environment for each domain you create. Effectively, changing the root of a domain limits the part of a file system a process can access, since the directory you specify becomes the root directory for all subsequent file system references. The chroot file system contains the Web pages, CGI scripts, users' mailboxes and home directories, FTP, and other files.

There may be instances where you might want to add enhanced capabilities to this chrooted environment. For example, you might want to install additional HTML or SHTML (SSI) pages, and other executable programs on the domain for users with Telnet or SSH access to the site. You can make these functions available through shell scripts and custom .tar files.

The scale of customizations possible depend on the security level set for the domain. Please read the overview of domain security in the following section before customizing domains.

Overview of domain security

When multiple domains are hosted on a single server, sharing system resources, there is a high possibility of sabotage or inadvertent activity that may compromise the integrity of data. Setting appropriate security levels for a domain can check misuse or malevolent activity.

Depending on the security level chosen, certain services for the domain run in protected mode within the restricted environment of the domain's file system, technically referred to as a chrooted environment. This prohibits the resources of the secured domain from unauthorized access; also, the administrator and users of the secured domain cannot access data or resources pertaining to other domains on the Ensim Pro server.

Ensim Pro offers three security levels:

- High security
- Medium security
- Low security

For details on each security level, refer to the Server Administrator Help (accessible through the Help option on the *System Menu* of the control panel.)

Customizing domains with virtDomain.sh

Ensim Pro allows you to customize any new domain that you create by using the customization script, **virtDomain.sh**. Once the customization is complete, all new domains you create reflect the changes made by your script. However, existing domains remain unchanged unless you manually execute the **virtDomain.sh** script (passing the correct arguments and password) for those domains.

To use this option, follow these steps before you create a domain.

- If you have created custom HTML pages and executables, collect them into a tar file called **virtDomain.tar** *i*n the directory **/etc/appliance/customization**.
- If you created a script to execute additional customizations, name the script **virtDomain.sh**. The **virtDomain.sh** script should also reside in the directory **/etc/appliance/customization**.

Caution: The Apache Web server is compiled with suexec support, which is sensitive to security issues and does not run CGI scripts that do not conform to its standards. If your **virtDomain.tar** file installs scripts into the domain's cgi-bin directory, you must set the cgi-bin permissions to ownership by the Site Administrator and turn off group and other write privileges by using the **chmod 0755** command.

You must exercise caution while customizing domains using the **virtDomain.sh** script. Improper usage can adversely affect the functioning of all new domains that you create.

The customization process works as follows:

Once you have created a new domain and assigned services to it (using the Site Manager option on the Server Administrator page), Ensim Pro looks in the directory **/etc/appliance/customization** for the file **virtDomain.tar**. If it finds this file, it untars the file into the root directory of the chroot file system.

After the file is unpacked, Ensim Pro runs the executable script **virtDomain.sh** as root. On the command line, the script receives the following three arguments.

- The name of the domain.
- The name of the Site Administrator.
- The IP address of the domain. For name-based domains, the script accesses the IP address of the server.

The script receives the password of the newly created domain from the standard input. If it encounters any errors, it displays them in the Details window of the Server Administrator control panel.

Important: CLI scripts reside in the directory **/usr/local/bin**. You must include the full path name for the CLI script that you are calling - that is, **/usr/local/bin/*<CLI script>***. Using the script to automatically customize a domain will fail if the script calls the required CLI script without including the full path name.

Transferring file permissions to the new domain owner

```
#!/bin/sh
DOMAIN=$1
WP_USER=`/usr/local/bin/sitelookup -d $DOMAIN wp_user`
#Assumes script alias is set to cgi-bin
chown -R $WP_USER:$WP_USER \
/home/virtual/$WP_USER/var/www/cgi-bin/
chmod -R 0755 /home/virtual/$WP_USER/var/www/cgi-bin/
```

Using script hooks

In addition to **virtDomain.sh**, you can take advantage of script hooks to customize Ensim Pro domains. Script hooks allow you to **add** custom scripts to augment the functionality of a script as desired. For example, when you modify a site, you can add a custom script that enables you to notify the Site Administrator of the change. If you deploy a centralized DNS server for various servers, you can use script hooks to notify the central DNS server of any updates. The scripts must be located at **/etc/appliance/customization**.

On the command line the scripts receive **site*<n>*,** where ***<n>*** uniquely identifies the site, as the argument. For example, you can write scripts to send custom notifications or log details of an event.

editVirtDomain.sh

The **editVirtDomain.sh** script hook runs custom commands placed in the script file. The script will be run whenever the specified domain is modified.

```
Syntax: editVirtDomain.sh site<n>
```

where <n> is the site identifier of the site being modified. Information about the domain can be obtained by using the **sitelookup** API.

Notifying Site Administrators when sites are modified

```
#!/bin/sh
#the site id is the first argument
siteid="$1"
# look up the site administrator name from the site id
siteadmin_email=`cat /home/virtual/$siteid/info/current/siteinfo |grep ^email |cut
-d' ' -f3`
# send mail to the site admin
echo '< some email content>' |
mail -s 'your domain is being edited' $siteadmin_email
```

enableVirtDomain.sh

The **enableVirtDomain.sh** script hook runs custom commands placed in the script file. The script will be run whenever the specified domain is enabled.

```
Syntax: enableVirtDomain.sh site<n>
```

where ***<n>*** is the site identifier of the site being enabled. Information about the domain can be obtained by using the **sitelookup** API.

For example, you could use the script provided to notify Site Administrators whenever their sites are enabled. For more information, see Notifying Site Administrators when sites are modified on page 66.

disableVirtDomain.sh

The **disableVirtDomain.sh** script runs custom commands placed in the script file. The script will be run whenever the specified domain is suspended.

```
Syntax: disableVirtDomain.sh site<n>
```

where ***<n>*** is the site identifier of the site being disabled. Information about the domain can be obtained by using the **sitelookup** API.

For example, you could use the script provided to notify Site Administrators whenever their sites are disabled. For more information, see Notifying Site Administrators when sites are modified on page 66.

deleteVirtDomain.sh

The **deleteVirtDomain.sh** script hook runs custom commands placed in the script file. The script is run whenever the specified domain is deleted.

```
Syntax: deleteVirtDomain.sh site<n>
```

where **<n>** is the site identifier of the site being deleted. Information about the domain can be obtained by using the **sitelookup** API.

For example, you could use the script provided to notify Site Administrators whenever their sites are deleted. For more information, see Notifying Site Administrators when sites are modified on page 66.

Customizing subdomains

You can customize subdomains by using subdomain hooks. Subdomain hooks enable you to perform additional actions when a subdomain is added or deleted. For example, you can use subdomain hooks to notify the Site Administrator whenever a subdomain is added for any site. The subdomain scripts are not domain specific and apply to all subdomains of all corresponding root domains on Ensim Pro.

In order to do this, you must write the script hook and place it in the **/etc/appliance/customization/** directory, set executable permissions for the script, and place the commands to be executed in the script file.

createSubDomain.sh

The **createSubDomain.sh** script hook can be used to run custom scripts whenever a subdomain is created.

```
Syntax: createSubdomain.sh
```

See the following table for the list of parameters that can be used with the script. These parameters are not specified on the command line. The script can be called with these parameters on standard input.

Notifying the Site Administrator when a subdomain is added

```
#!/bin/sh
# get all the parameters passed on standard input
export `cat /dev/stdin`
# get the root domain name for which this subdomain is being added.
parent_domain=$psi_domain
subdomain=$lsd_domain
subdomain_name=$lsd_name
# now email the Server Administrator about this
appliance_admin_email=`cat /etc/appliance/appliance.ini |grep adminemail |cut -d'
' -f3`
echo 'A subdomain $subdomain is being created for $parent_domain' |mail -s
'subdomain creation'
$appliance_admin_email
```

The following parameters can be passed to the script on the standard input.

Table 28. List of parameters for subdomain hooks

Parameter	What it indicates
Parent Site Info	
psi_domain	The name of the root domain. For example, myco.com

Table 28. List of parameters for subdomain hooks

Parameter	What it indicates
psi_version	The Ensim Pro version on which the root domain is hosted For example, 3.7.0-13
psi_admin_user	The name of the Site Administrator For example, myadmin
psi_admin	The Unix account information of the Site Administrator For example, admin1
psi_passwd1	The encrypted password of the Site Administrator For example, ***
psi_email	The email address of the Site Administrator For example, example@example.com
psi_passwd2	The encrypted password of the Site Administrator For example, ***
Parent IP Address Info	
pip_ipaddrs	The IP address of the root domain For example, 1.2.3.4
pip_namebased	The type of domain; indicates whether the root domain is name based. **Note:** The value “1” indicates a name-based domain.
pip_version	The version of Ensim Pro on which the root domain is hosted. For example, 3.7.0-13
pip_nbaddrs	The IP address of the root domain For example, 1.2.3.4
Global Subdomain Info	
gsd_wildcards	A boolean value indicating whether wildcard subdomains are enabled for the root domain. **Note:** The default value is 0.
gsd_max	The maximum number of subdomains that can be created.
gsd_version	The Ensim Pro version on which the subdomain is hosted
gsd_enabled	A Boolean value indicating whether subdomains are enabled for the root domain. **Important:** The value for this argument must be 1.

Table 28. List of parameters for subdomain hooks

Parameter	What it indicates
gsd_base	The base directory for subdomains created on the root domain. For example, **/var/www**
Local Subdomain Info	
lsd_cgi	A Boolean value indicating whether CGI service is enabled for the subdomain.
lsd_cgi_root	The CGI directory where CGI scripts for the subdomain are placed. For example, **/var/www/test2/cgi-bin**
lsd_cgi_extensions	The script extensions permissible for CGI scripts For example, **cgi,pl**
lsd_user_subdomain	A Boolean value indicating whether the subdomain is a user subdomain
lsd_owner	The owner of the subdomain For example, **myadmin**
lsd_aliases	A Boolean value indicating whether aliases are enabled for the subdomain.
lsd_bind	A Boolean value indicating whether the subdomain has a DNS entry.
lsd_document_root	The document root of the subdomain. For example, **/var/test2**
lsd_domain	The host name of the subdomain. For example, **example.example1.com**
lsd_name	The lower-level subdomain name. For example, the subdomain name **example** in the host name **example.example1.com**

deleteSubDomain.sh

The **deleteSubDomain.sh** script can be used to run custom scripts whenever a subdomain is deleted.

```
Syntax: deleteSubdomain.sh
```

See the above table for the list of parameters that can be used with the script. These parameters are not specified on the command line. The script can be called with these parameters on standard input.

Additional executable files

When Ensim Pro creates the chroot environment, it makes the following executable Linux commands available to users who can then access the newly created domain using Telnet or SSH.

Table 30. List of parameters for subdomain hooks

/bin/awk	/bin/more	/usr/bin/find	/usr/bin/troff
/bin/bash	/bin/mv	/usr/bin/groff	/usr/bin/uncompress
/bin/cat	/bin/ping	/usr/bin/grotty	/usr/bin/whois
/bin/chmod	/bin/rm	/usr/bin/gtbl	/usr/bin/fwhois
/bin/chown	/bin/sed	/usr/bin/gzip	/usr/bin/makemap
/bin/cp	/bin/sh	/usr/bin/head	/usr/bin/newaliases
/bin/date	/bin/su	/usr/bin/id	/usr/bin/procmail
/bin/egrep	/bin/tar	/usr/bin/less	/usr/bin/vacation
/bin/false	/bin/touch	/usr/bin/man	/usr/bin/ftp
/bin/gawk	/bin/true	/usr/bin/passwd	/usr/bin/ncftp
/bin/grep	/bin/vi	/usr/bin/perl	/usr/bin/ncftpget
/bin/gzip	/bin/zcat	/usr/bin/python	/usr/bin/ncftpput
/bin/gunzip	/bin/dnsdomainname	/usr/bin/run-parts	/usr/bin/analog
/bin/ln	/bin/hostname	/usr/bin/tail	/usr/bin/nslookup
/bin/ls	/usr/bin	/usr/bin/tbl	
/bin/mkdir	/usr/bin/compress		

Customizing the Web site welcome pages

You can customize your Web site welcome pages in the following ways.

- Inserting custom welcome pages
- Assigning custom home pages to users
- Inserting links to login screens

Inserting custom welcome pages

When you create a new Web site, Ensim Pro assigns a default welcome page (**index.html**) to it. To replace the default page with your own custom welcome page, replace the file **index.html** in the **directory /etc/virtualhosting/templates/apache/var/www/html/** with your own **index.html** file.

Assigning custom home pages to users

When you create a new Web site, Ensim Pro assigns a default home page (index.html) for users on the site. This home page is located in the directory, /home/virtual/***<domain name>***/**etc/skel/public_html/**, where ***<domain name>*** is the domain name of the new Web site.

You can replace the default page with your own home page, replace the file **index.html** in this directory with your own **index.html** file.

> **Note:** The most efficient way to assign a custom home page to each user is to create a script that replaces the default file **index.html** in the directory **/home/virtual/<domain_name>/etc/skel/public_html** with your custom **index.html** file.

Inserting links to login screens

To make logging on quick and convenient for administrators, you can add a link from your custom welcome page to the login screens for both Site Administrators and User Administrators.

To insert a link to the Site Administrator login screen, add the following code to the custom welcome page in **/etc/virtualhosting/templates/apache/var/www/html/index.html**.

Access to your Site Administrator: **< a href="http://ENSIM_DOMAINNAME/admin/">http://ENSIM_DOMAINNAME/admin/**

When you create a domain and install the custom welcome page, Ensim Pro replaces **ENSIM_DOMAINNAME** with the name of the domain.

To insert a link to the User Administrator login screen, add the following code to the custom welcome page in **/etc/virtualhosting/templates/apache/var/www/html/index.html**.

Access to your User Administrator: **< a href="http://ENSIM_DOMAINNAME/user/">http://ENSIM_DOMAINNAME/user/**

When you create a domain and install the custom welcome page, Ensim Pro replaces **ENSIM_DOMAINNAME** with the name of the domain.

Changing the message of the day (motd)

When users use Telnet to access the server, you can display a message on the terminal screen above the prompt. This message allows you to inform users of temporary system outages, changes that might occur in their Service Plans, or any other information of your choice.

You can customize the message of the day in one of two ways. You can create a separate message for each domain, or you can create a common message across all domains.

Creating a separate message for each domain

▼ To create a separate message for each domain

1. Use any Telnet or SSH client to access your server and log on as any user.
2. Type `su -` to assume root user privileges.
3. Create the file **/home/virtual/*<domain_name>*/etc/motd** with the message of the day, where ***<domain_name>*** is the name of the domain for which you want the message of the day to be displayed.

4 Change the ownership of the file, motd, to the Site Administrator and group using **chown** and then set the permissions to **0644**. The Site Administrator cannot create or delete this file but can edit it. If you do not want the Site Administrator to change this file, set the ownership to **root.**

Creating a common message across all domains

▼ **To create a common message across all domains:**

1 Use any Telnet or SSH client to access your server and log on as any user.

2 Type `su -` to assume root user privileges.

3 Using an editor of choice, edit the message of the day in the file **/etc/motd** at the root level of the server on which Ensim Pro is installed.

4 Save the changes to the file.

5 Use the `ln` command to hard link the file **/etc/motd** individually to each domain.

For example, to create a hard link from the Ensim Pro server to the domain **mycompany.com** enter the command,

```
ln /etc/motd /home/virtual/mycompany.com/etc/motd
```

Customizing services

In order to customize services, you need to modify the **custom.py** file specific to each service, located under the folder, **/usr/share/doc/webppliance-SERVICENAME-SERVICEVERSION/**

Do not modify the original copy of the custom.py file.

The customization can either be global or site-specific. The list of the services that can be customized is as follows:

- anonftp
- openssl
- apache
- proftpd
- cgi
- ssi
- subdomain (global only)
- sqmail
- mod_perl
- tomcat4

Global customizations

Global customizations affect the way certain services (such as Apache) are configured for all sites on the Ensim Pro server. For example, in the case of Apache, you can customize the way the VirtualHost container is written.

Each service that can be globally customized has a file explaining its variables in the file, **/usr/share/doc/webppliance-*<servicename>*/custom.py**.

To globally customize any service, you must copy the file **/usr/share/doc/webppliance-*<servicename>*/custom.py** into **/usr/lib/ensim-python/site-packages/vh3/custom/*<servicename>*.py**

For example, to globally modify the behavior of the Apache service on Ensim Pro, you must run the following command:

```
cp /usr/share/doc/webppliance-apache-3.5.0/custom.py /usr/lib/ensim-python/site-packages/vh3/custom/apache.py
```

Then, modify the file **/usr/lib/ensim-python/site-packages/vh3/custom/apache.py** as required. The significance of each of the customizable variables is given in the **custom.py** file.

Global customizations are retained when you back up and subsequently restore the Ensim Pro server.

▼ To apply global customizations

1. Copy **/usr/share/doc/webppliance-*<servicename>*/custom.py** as **/usr/lib/ensim-python/site-packages/vh3/custom/*<servicename>*.py**.

   In the ***<servicename>*** field enter the name of the service you want to customize.

   For example, to customize the anonftp service, copy **/usr/share/doc/webppliance-anonftp/custom.py** as **/usr/lib/python2.1/site-packages/vh3/custom/anonftp.py**.

2. Modify the file ***<servicename>*.py** as per your requirements.

3. Save the file.

4. To restart Ensim Pro, type the command
   `/etc/rc.d/init.d/epld restart`

Site-specific customizations

Site-specific customizations affect the way services (such as Apache) are configured for a particular site on the Ensim Pro server.

Each service that can be customized for a site has a file explaining its variables in the file, **/usr/share/doc/webppliance-*<servicename>*/custom.py**.

To customize any service for a site, you must copy the file **/usr/share/doc/webppliance-*<servicename>*/custom.py** into **/home/vitual/site*<n>*/info/custom/*<servicename>*.py**, where ***<n>*** represents the unique site number.

For example, to modify the behavior of the Apache service for a site, you must run the following command:

```
cp /usr/share/doc/webppliance-apache-3.5.0/custom.py
/home/vitual/site12/info/custom/apache.py
```

Then, modify the file **/home/vitual/site12/info/custom/apache.py**, as required. The significance of each of the customizable variables is given in the custom.py file.

▼ To apply site-specific customizations

1 Copy **/usr/share/doc/webppliance-*<servicename>*/custom.py** *as* **/home/virtual/site*< n>*/info/custom/*<servicename>*.py**

where:

n in site***<n>*** stands for the site-specific number and ***<servicename>*** refers to the name of the service you want to customize.

For example, to customize the anonftp service, copy **/usr/share/doc/webppliance-anonftp/custom.py** as **/home/virtual/site1/info/custom/anonftp.py**.

2 Modify the file ***<servicename>*.py** as per your requirements.

3 Save the file.

4 To restart Ensim Pro, type the command
`/etc/rc.d/init.d/epld restart`

Changing the server host name and IP address

Use **netconf** to change the host name and IP address, (which is included in the **linuxconf** rpm in the Fedora 1 distribution). Note that another utility netconfig is usually present on Fedora 1 servers. The netconfig utility allows you to set the IP address, but not the hostname.

After setting the new hostname and IP address, verify that the **/etc/hosts** file contains the new entry. Restart your server to complete the change.

Adding and removing virtual DNS servers for Reseller Administrators

Ensim Pro supports the addition and removal of virtual DNS for Reseller Administrators. In order to allow Reseller Administrators to satisfy the dual DNS server requirements of their customers, you must add the virtual DNS server.

Adding a virtual DNS server

▼ To add a virtual DNS server

1 Note down the name and IP address of the new virtual DNS server. For example, ns1.reseller.com 1.2.3.4

2 Access the server and log on as root user.

3 To add an IP alias to the Ensim Pro server, enter the command,
`/sbin/applifconfig alias 1.2.3.4`

4 To get the IP address aliased each time you start Ensim Pro, add the appropriate lines to the end file **rc.local**, located at **/etc/rc.d/.**

`/sbin/applifconfig alias 1.2.3.4`

...

where `1.2.3.4` is your reseller's DNS IP address.

5 Modify the file **options.conf.wp**, located at **/etc/bind/**, and add the IP address to the listen-on option.

Example:

```
//This file stores the options statement maintained by Ensim
options {
directory “/var/named”;
listen-on {127.0.0.1; 1.2.3.4;};< --add the IP address here
};
```

6 Add the line, `ns1.reseller.com 1.2.3.4`, to the file **virtualDNS**, located at **/var/named/**.

7 To restart the name server, enter the command,

```
/etc/init.d/named restart
```

8 Using the Server Administrator control panel, create zone record for reseller.com.

Note: While creating reseller.com make sure that ns1.reseller.com points to 1.2.3.4 and new NS records are added.

9 To test the new virtual DNS server, run a query (**dig/nslookup**) on it.

Note: The newly added virtual DNS server will not be displayed in the virtual DNS page of the Server Administrator control panel.

Removing a virtual DNS server

▼ To remove a virtual DNS server

1 Note down the host name and IP address of the virtual DNS server to be removed. For example: ns1.reseller.com 1.2.3.4

2 Access the server and log on as root user.

3 To remove an IP address bound to the server or network card, enter the following command.

```
/sbin/applifconfig delete 1.2.3.4
```

4 Remove the following lines from the file **rc.local**, which is located at **/etc/rc.d/**.

```
/sbin/applifconfig alias 1.2.3.4
```

...

where `1.2.3.4` is your reseller’s DNS IP address.

5 Remove the IP addresses from the file **options.conf.wp**.

Example:

```
//This file stores the options statement maintained by Ensim
options {
directory “/var/named”;
listen-on {127.0.0.1; 1.2.3.4;}; //< --remove this IP address.
};
```

6 Remove the following line from the file **virtualDNS** located at **/var/named/**.

```
ns1.reseller.com 1.2.3.4
```

7 Through the Server Administrator control panel, modify the zone record for reseller.com.

Note: While modifying **reseller.com**, make sure that **ns1.reseller.com** does not point to 1.2.3.4 and new NS records are set.

8 To restart **named**, enter the following command.

```
/etc/rc.d/init.d/named restart
```

9 To restart virtual hosting, enter the following commands.

```
/etc/rc.d/init.d/virtualhosting stop
/etc/rc.d/init.d/virtualhosting start
```

Passing additional environment variables to CGI programs

Apache's suexec cgi-wrapper has been modified so that it can be configured to pass additional environment variables apart from those considered "safe" at the time it is compiled.

In order to add additional variables to this list of "safe" variables, place the names of the additional environment variables in the file, **/etc/suexec.env.ensim**, one variable per line.

> **Important:** To ensure security, this file must be owned by root, with only user read and write permissions enabled (indicated by 0600 UNIX permission).

If you want to allow Apache to pass an environment variable named **MY_TRUSTED_ENV** to a CGI program, place the following line in the file, **/etc/suexec.env.ensim**.

```
MY_TRUSTED_ENV
```

CHAPTER 4

Customizing Tomcat

Introduction

This section provides information and instructions on customizing domains using Tomcat.

In this section:

About Tomcat

Ensim Pro supports the latest version of Tomcat (v-4.0.6), developed by the Jakarta-Apache Project. Tomcat 4.0 implements the final released versions of the Java Servlet 2.3 and JaveServer Pages™ (JSP) 1.2 specifications. As required by the specifications, Tomcat 4.0 also supports Web applications built for Java Servlet 2.2 and JSP 1.1 specifications, with no changes. Additionally, the Tomcat 4.0 Servlet Container (Catalina) provides greater flexibility and performance.

The Tomcat 4 package includes a set of tools designed to host Java™ Web applications. The package is shipped with the following:

- Sun Java 2 Standard Edition (J2SE) development environment version 1.4.2, which includes J2SDK™ 1.4.2 and JRE (installed in **/usr/java/j2sdk1.4.2/** and **/usr/java/j2sdk1.4.2/jre/** respectively)
- mod_jk-2.0 (Apache 2.0)
- Tomcat 4.0.6, installed in **/var/tomcat4**
- Database driver for MySQL and postgres, installed in **/var/tomcat4/common/lib**:
  - mysql-connector-java-3.0.8-stable-bin.jar for MySQL
  - pg73jdbc3.jar for postgres
  - Tyrex, installed in **/var/tomcat4/common/lib/tyrex-0.9.7.0.jar**
  - JNDI™, installed in **/var/tomcat4/common/lib/jndi.jar**
  - Mail, installed in **/var/tomcat4/common/lib/mail.jar**

Important: Ensure that you meet the following minimum requirements for Tomcat:
* **RAM:** The server must have at least 512 MB of RAM.
*__Disk usage:__ The server must have at least 21 MB free disk space available.

About JSPs and servlets

Ensim Pro (with the mod_jk module) has JSP and servlet functionality enabled on the Apache Web server. The Tomcat engine executes JSP and servlets. Once you enable Tomcat for a site, Ensim Pro instructs the Web server to send certain requests to Tomcat for processing. Tomcat requires an execution environment known as context, to be set up before it can execute JSPs and servlets properly.

A context is basically a directory structure on the server from which Tomcat can execute servlets and JSPs. Ensim Pro automatically sets contexts for you. You can also add a new context by uploading Java Web archive files (with the extension .war) into Web content directories.

Note: Java Web archive files can be uploaded only after they are uncompressed. To uncompress the Java Web archive files, you must use the Site Administrator user account. Since the Tomcat daemon runs as a Tomcat user, it does not have the requisite privileges to uncompress files owned by the Site Administrator.

Deploying JSPs and servlets on Ensim Pro

Deploying JSPs and servlets on Ensim Pro is as easy as uploading a Web site using any FTP client. When you enable Tomcat for a site, Ensim Pro creates a default Tomcat context, for the virtual site in the Web directory. The context path of each virtual site is **/home/virtual/domainname/var/www/html/**.

A new directory called **WEB-INF** is created in this directory. The WEB-INF directory contains some configuration information for Tomcat and a directory in which servlet class files are placed for deployment.

As long as the file extension of the JSP file is .jsp, the Site Administrator can simply load a JSP page in the directory **/var/www/html/**, and the Web server automatically passes it on to Tomcat for processing.

For example, if you have a file named **test.jsp**, the Site Administrator can use an FTP client to upload the file in the directory **/var/www/html/**. You can then access the dynamic JSP page through **http://*<site name>*/test.jsp**. The first time you access this page, Tomcat takes a few seconds to compile the page but subsequent accesses are much quicker. Compiling a servlet source Java file creates a servlet class file. Deploying a servlet on Ensim Pro includes uploading the servlet class file in **/var/www/html/WEB-INF/classes/**.

For example, if you have a file named **test.class**, you would use an FTP client to upload the file in **/var/www/html/WEB-INF/classes/**. The servlet is executed through **http://*<site name>*/servlet/test**.

Compiling a Java servlet source file

You can develop servlets manually through the command line and a text editor. Ensim Pro provides a Java software development kit and a Java run time environment for the Site Administrator when you enable the service **Development Tools** for the site. The service installs a script **java.sh** in **/etc/profile.d**/. When you log onto the server, your shell runs this script and relevant Java-environment variables are set up for you. You can add additional **$CLASSPATH**, for any site, by editing the script **/etc/profile.d/java.sh**.

▼ To compile a Java servlet source file into a Java class file for Tomcat

1. Log on to Ensim Pro as the Site Administrator.
2. Change to the directory in which the source servlet code file is located.
3. Enter the following command.

   `javac` ***<myServlet.java>*** (substitute your file name for the file ***myServlet.java***).
4. If the compilation is successful and no errors are reported, the file **myServlet.class** can be located in the same directory.
5. To use this servlet, copy the file **myServlet.class** in the directory **WEB-INF/classes**.

Customizing Tomcat

The standard, out-of-the-box configuration of Apache Tomcat may not fit your development needs. This is particularly true if your application requires elevated permissions, a custom JDBC™ driver or access to other Java classes stored on the server.

Reviewing site permissions

The default site permissions are set by the file **/var/tomcat4/conf/sites.policies.d/site*<n>*.policy** and the site context is set by the file **/var/tomcat4/conf/sites.xml.d/site*<n>*.xml** where **site*<n>*** is the site index.

You can review the site permissions anytime.

▼ To review site permissions

1. Log on to the server as root user.
2. At the command prompt, type the following:

   `sitelookup -d < domain name>`

   The following output displays.

   `<domain name>, admin<n>,site<n>,/home/virtual/<domain name>,<admin name>.`

   The output indicates that the site index of `<domain name>` is `site<n>`. Use the site index to determine site permissions.

For example, if a site named **www.myco.com** has the site index site***<n>***, then the file **/var/tomcat4/conf/sites.policies.d/site*<n>*.policy** will consist of default site permissions for site**<>**. This file appends to the file **catalina.policy**.

Reviewing site context

The default site context is set by the file **/var/tomcat4/conf/sites.xml.d/site<*n*>.xml**.

The contents of the file **site<*n*>.xml** are:

```
< Host name="myco.com" appbase="/home/virtual/site< n>/fst/var/www/html">
< Alias>www.myco.com< /Alias>
< !-- Global logger unless overridden at lower levels -->
< Logger className="org.apache.catalina.logger.FileLogger"
directory="/home/virtual/site< n>/fst/var/log"
prefix="tomcat4_log."
timestamp="true"/>
< Realm className="org.apache.catalina.realm.MemoryRealm" />
            < Context path=""
docBase="/home/virtual/site< n>/fst/var/www/html"
crossContext="false"
reloadable="true" />
sinclude(`/var/tomcat4/conf/sites.xml.d/site< n>.xml.custom')
< /Host>
```

The default context for **myco.com** is **/var/www/html** in the domain file system of **myco.com**. This file appends to the file **server.xml**.

Adding security permissions

All Tomcat contexts, added by Ensim Pro, receive a default set of security permissions. The default security permissions are as follows:

```
grant codeBase "file:/home/virtual/site<n>/fst/var/www/html/

-" {permission java.net.SocketPermission "*", "connect";};
```

▼ To add security permissions for a site:

1. Log on to Ensim Pro as the root user.
2. Change directory to **/var/tomcat4/conf/ sites.policies.d/.**
3. In this directory, create the file **site<*n*>.policy.custom**.
4. Add the required security permissions to the file **site<*n*>.policy.custom**.
5. Regenerate the file **/var/tomcat4/conf/catalina.policy** using the following command.

   ```
   m4 /var/tomcat4/conf/catalina.policy.template >
   /var/tomcat4/conf/catalina.policy
   ```

6. To clear the cache and to enable new changes in **/var/tomcat4/work/domainname**, type the following commands.

   ```
   cd /var/tomcat4/work/

   rm -rf *
   ```

7. To restart Tomcat, type the following command.

   ```
   /etc/rc.d/init.d/tomcat4 restart
   ```

   When adding security permissions for a site:

   - Do **not** make any direct changes to the files **/var/tomcat4/conf/catalina.policy** or **site<*n*>.policy**, as these changes will be overwritten by the configuration scripts in Ensim Pro.

- Edit the file **site<*n*>.policy.custom**. This will be appended to the file **catalina.policy**.
- Be aware that allowing all security permissions may open up the site to hackers.
- Be aware that syntax errors may result in Tomcat failing to start.
- Edit **/var/tomcat4/conf/catalina.policy.template** to set global settings - applicable to all sites.
- Back up .conf files before editing them.

Adding an additional context path

All Tomcat enabled sites receive the default context path as **/var/www/html**/. As the Server Administrator, you can add an additional context path.

▼ To add an additional context path:

1. Log on to Ensim Pro as the root user.
2. Change directory to **/var/tomcat4/conf/sites.xml.d/**.
3. In this directory, create a file **site<*n*>.xml.custom**.
4. Add the additional context to this file.
5. Regenerate the file **/var/tomcat4/conf/server.xml** using the following command.

   ```
   m4 /var/tomcat4/conf/server.xml.template > /var/tomcat4/conf/server.xml
   ```

6. To clear the cache in order to enable new changes in **/var/tomcat4/work/domain name**, type the following commands.

   ```
   cd /var/tomcat4/work/
   rm -rf *
   ```

7. To restart Tomcat, type the following command.

   ```
   /etc/rc.d/init.d/tomcat4 restart
   ```

   When adding an additional context path:

   - Do **not** make any direct changes to the files **/var/tomcat4/conf/server.xml** or **site<*n*>.xml**, as these changes will be overwritten by the configuration scripts in Ensim Pro.
   - Edit the file **site<*n*>.xml.custom**. This will be appended to the file **server.xml**.
   - Be aware that syntax errors may result in Tomcat failing to start.
   - Edit **/var/tomcat4/conf/server.xml.template** to set global settings applicable to all sites.
   - Back up conf files before editing them.

Adding additional class paths

Ensim Pro provides database drivers for MySQL and postgres. You may come across a situation where the default class path for Tomcat does not suffice and you need to indicate the path for your own classes that exist elsewhere on the file system.

If you want to add additional classes (for example, a new JDBC driver) to Tomcat's default CLASSPATH, you must add **CLASSPATH=$CLASSPATH:/path/to/my/new/class** in the file **/etc/tomcat4/conf/tomcat4.conf**.

For the changes to take effect, restart the Tomcat engine. Only root users can set new CLASSPATH for any Web site. If you want to add a new database driver or component, which should be available to all sites, add it to:

- **/var/tomcat4/common/classes** – if it is a class file
- **/var/tomcat4/common/lib** – if it is a jar file

Enabling the Tomcat development environment

You can create a Web site with the domain name **myco.com** with Tomcat development environment enabled and set the Site Administrator name as **adminmyco**.

Assumptions:

- The Web site is developed on Sun J2SE 1.4
- The site number is 1

▼ To enable the Tomcat development environment:

1. Log on to **myco.com** as adminmyco, using any FTP client.
2. Upload all class files to **/var/www/html/WEB-INF/classes**.
3. Upload all lib files to **/var/www/html/WEB-INF/lib**.
4. Upload the file **web.xml** to **/var/www/html/WEB-INF/**.
5. Upload the remaining Web site to **/var/www/html/**.

Note: To compile .java files, use **javac *.java**.

Adding permissions

▼ To add permissions to connect to the SMTP server:

1. Log on to Ensim Pro as the root user.
2. Change directory to **/var/tomcat4/conf/sites.policy.d**.
3. Create a new file **site1.policy.custom**.
4. In the file **site1.policy.custom**, include the following lines to add permissions to connect to the SMTP server.

```
Permission java.net.SocketPermission "127.0.0.1:25", connect, resolve";
Permission java.net.SocketPermission "localhost:25", connect, resolve";
```

5. Save the file **site1.policy.custom**.
6. Regenerate the file **/var/tomcat4/conf/catalina.policy** using the following command.

```
m4 /var/tomcat4/conf/catalina.policy.template >
/var/tomcat4/conf/catalina.policy
```

7. To restart Tomcat, type the following command.

```
/etc/rc.d/init.d/tomcat4 restart
```

Adding additional context

▼ **To add additional context:**

1. Log on to Ensim Pro as the root user.
2. Change directory to **/var/tomcat4/conf/sites.xml.d**.
3. Create a new file **site1.xml.custom**.
4. Add new context to this file.
5. Save the file.
6. Regenerate the file **/var/tomcat4/conf/server.xml** using the following command.

   ```
   m4 /var/tomcat4/conf/server.xml.template > /var/tomcat4/conf/server.xml
   ```

7. To restart Tomcat, type the following command.

   ```
   /etc/rc.d/init.d/tomcat4 restart
   ```

Starting Tomcat

By default, Ensim Pro always displays the Tomcat status as **ON** (enabled). You can start Tomcat either manually or automatically.

Manually

▼ **To manually start Tomcat:**

1. Log on to Ensim Pro as the Server Administrator.
2. In the Server Administrator control panel, click Services in the left navigation bar.
3. In the Services window, locate the Tomcat service.
4. In the Actions column, click the Start icon.

Automatically

▼ **To automatically start Tomcat:**

1. Log on to Ensim Pro as the root user.
2. Type the following commands in the order specified.

   ```
   /etc/rc.d/init.d/tomcat4 stop
   /sbin/chkconfig --add tomcat4
   /etc/rc.d/init.d/tomcat4 start
   ```

Disabling Tomcat

You can disable Tomcat permanently if you no longer want to use Tomcat. You can turn off the Java virtual machines that handles the requests and prevent Tomcat from launching in the future.

▼ To disable Tomcat:

1. Log on to Ensim Pro as the Server Administrator.
2. Type `su -`. The system prompts you for a password.
3. Enter the root password.

   You now have root privileges, which means that you can change anything on the server.
4. Type the following commands in the order specified.

```
/sbin/chkconfig --del tomcat4
/etc/rc.d/init.d/tomcat4 stop
```

Additional resources

For more information on Tomcat, refer to the following links.

- http://jakarta.apache.org/tomcat/tomcat-4.0-doc/index.html
- http://www.jguru.com/faq/Tomcat
- http://java.sun.com/j2se/1.4/index.html
- http://jakarta.apache.org/site/mail.html
- http://mmmysql.sourceforge.net
- http://jdbc.postgresql.org
- http://jdbc.postgresql.org/doc.html
- http://www.exolab.org http://www.exolab.org
- http://jakarta.apache.org/ant/index.html

APPENDIX A

Customizing disk partitions

Introduction

This section explains how to customize disk partitions to manage the available disk resources on your target server.

Customizing disk partitions

If the target server where you install Ensim Pro has single and medium capacity hard disks (typically 40 GB and less) the default partitioning scheme of /boot = 50 MB, swap = twice the amount the RAM, and /= rest of the hard disk space will suffice.

However, for disks with a capacity of 40 GB and above or for systems with more than one hard disk, the above scheme is not feasible. In such cases, a custom partitioning scheme must be specified.

If the target server has high capacity hard disks or multiple hard disks, it is recommended that you customize disk partitions so as to have multiple partitions on your hard disks.

Disk partitioning schemes for Ensim Pro

If the capacity of the hard disk is greater than 40 GB or if the number of hard disks exceed one, the following partitions can be used:

- **/** = 5 GB
- **/boot** = 101 MB
- **swap** = 1 GB and above or multiple swap partitions
- **/var** = at least 5 GB
- **/home** = largest

If the target server has only one hard disk, you can opt for the above partitioning scheme on the same disk. However, if the target server has more than one hard disk (for example, two), the above scheme can be spread across the two disks as follows:

First disk (40 GB):

- **/** = ***<remaining disk space>***
- **/boot** = 101 MB
- **swap** = 1024 MB

- **/var** = 20 GB
- **/tmp** = 2 GB

Second disk (40 GB):

- **swap** = 1024 MB
- **/home** = ***<remaining disk space>***

Setting up Ensim Pro to use multiple partitions

To allow the server's root user to configure where Ensim Pro temporary files are created, change the path name to the temporary files by editing the file **/etc/virtualhosting/tmpdirs**. Ensim Pro essentially makes changes to files located under **/etc** and **/home/virtual**.

> **Important:** You must not edit the file **tmpdirs** until you finish installing Ensim Pro. If you attempt to create the directories as detailed below prior to installing Ensim Pro, the installation will fail. This is because the installation checks for the existence of **/etc/virtualhosting** and **/home/virtual** directories.

Additionally, any changes made in the file **tmpdirs** are overwritten during the installation process.

When you change the path name to the temporary files, each line in the **tmpdirs** file should use the following format.
<path-prefix>*:*<directory>
where ***<path-prefix>*** matches the beginning of file path names whose associated temporary file locations you want to modify, and ***<directory>*** is the path to an existing directory that lies on the same partition as ***<path-prefix>***.

For security purposes, ***<directory>*** should be root-owned, with no permissions for either group or other (for example, 0700), and all path components leading to this directory should not be editable by anyone other than root. If this file is empty, by default, all temporary files will be created in **/var/cache**.

> **Important:** You have to restart Ensim Pro to complete this configuration process. To restart Ensim Pro, type the following command at the command prompt:
> `/etc/rc.d/init.d/epld restart`

Assume the disk partitioning of the server is as follows:

- **/** is on **/dev/hda1**
- **/var** is on **/dev/hda2**
- **/home** is on **/dev/hda3**
- **/usr** is on **/dev/hda5**

Since **/etc, /var, /usr and /home** are on different partitions, a temporary directory must be specified for each partition. By default, Ensim Pro creates temporary directories on the **/etc** and **/home** partitions.

- **/etc:/etc/virtualhosting/tmp**
- **/home:/home/virtual/FILESYSTEMTEMPLATE/.tmp**

You can add an entry for the **/usr** partiton in **/etc/virtualhosting/tmpdirs** as follows:
`/usr:/usr/.tmp`

To create the directory **/usr/.tmp**, type the following command.

```
mkdir /usr/.tmp; chown 700 /usr/.tmp
```

You can add an entry for the **/var** partition in **/etc/virtualhosting/tmpdirs** as follows:
`/var:/var/tmp`

By default, Ensim Pro uses **/var/tmp** as the temporary directory. If **/var** is mounted on a separate partition, you must make an entry for the temporary directory in the **/etc/virtualhosting/tmpdirs** file for the **/** partition. This is because Ensim Pro uses hard links when files are transferred and hard links do not work across partitions. So, you must ensure that the corresponding temporary directory for a partition resides on the same partition.

For example, if the dir **/tmp** is on the **/** partition, you must edit the configuration file **/etc/virtualhosting/tmpdirs** to include the following entry: `/:/tmp`

Summating the above assumptions, the file **/etc/virtualhosting/tmpdirs** would include the following sequence of entries.

```
/etc:/etc/virtualhosting/tmp

/home:/home/virtual/FILESYSTEMTEMPLATE/.tmp

/usr:/usr/.tmp

/var:/var/tmp

/:/tmp
```

The sequence of entries in the file is important. During file operations, Ensim Pro uses the first partition that is reported by the system to store the requisite directory or file. For example, placing the **/:/tmp** entry before the entry **/var:/var/tmp**, will cause any search for files in the **/var** directory to default to the **/** directory.

Note: For more details on Linux partitions, contact Ensim Support at https://onlinesupport.ensim.com.

Index

U

ENSIM PRO - LINUX

ENSIM CORPORATION
1366 Borregas Avenue
Sunnyvale, California 94089
www.ensim.com