



PLESK 7

ADVANCED FEATURES MANUAL

Distribution of this work or derivative of this work in any form is prohibited unless prior written permission is obtained from the copyright holder.

All trademarks and copyrights are the property of their respective owners.

13800 Coppermine Road, Suite 112, Herndon, VA, 20171 USA, Ph.: 703 815-5670, Fax.: 703 815-5675

Table of Contents

1. Introduction	1
2. Event manager	3
3. Domain skeleton manager	13
Creating a skeleton	13
Activating a custom skeleton in Plesk control panel	14
4. Creation Utilities	17
5. Chili!Soft ASP support	29
6. Customizable httpd.include per domain	33
7. Global access control list in named.conf	35
8. Manageable Tomcat connectors ports	37
9. Restoring mail configuration	39

Chapter 1. Introduction

In addition to operations available via the Plesk control panel, the Plesk software provides several advanced management capabilities, available to the administrator from the command line. The administrator can:

- Set up an event manager that will help establish Plesk system connection to external systems, such as various billing modules,
- Create and manage the domain skeletons,
- Install the Sun Chili!Soft ASP support and configure it for working with Plesk,
- Use the Creation Utilities for creating and managing client accounts, domains, hosting accounts, various settings and services,
- Include domain-specific Apache configuration directives into `httpd.include` for domains,
- Use the global access control list in `named.conf` for allowing dns transfers,
- Configure the ports to be used by Tomcat connectors,
- Restore the mail configuration by Plesk database.

Chapter 2. Event manager

The event manager feature is intended to provide data interchange between Plesk and external systems. It allows the Plesk administrator to assign a specific command (event handler) to be executed on a certain system event. The following events can be processed:

Table 2.1.

Event (action) name	Description
client_create	Client account created
client_update	Client account updated
client_delete	Client account deleted
domain_create	Domain created
domain_update	Domain properties updated
domain_delete	Domain deleted
subdomain_create	Subdomain created
subdomain_update	Subdomain properties updated
subdomain_delete	Subdomain deleted
phys_hosting_create	Physical hosting created
phys_hosting_update	Physical hosting account updated
phys_hosting_delete	Physical hosting account deleted
mailname_create	Mail name created
mailname_update	Mail name updated
mailname_delete	Mail name deleted
webuser_create	Web user created
webuser_update	Web user properties updated
webuser_delete	Web user account deleted
client_limits_update	Client limits updated

Event (action) name	Description
domain_limits_update	Domain limits updated
maillist_create	Mailing list created
maillist_update	Mailing list updated
maillist_delete	Mailing list deleted
cp_user_login	User logged in to control panel
cp_user_logout	User logged out of control panel
dl_user_update	Domain user properties updated
siteapp_install	Site application installed
siteapp_reconfigure	Site application reconfigured
siteapp_unchain	Information about the site application removed from the database, however the application still functions on the domain
siteapp_uninstall	Site application deinstalled
siteapppkg_install	Site application package added to the application vault
siteapppkg_uninstall	Site application package removed from the application vault

You can assign an event handler by defining a command line and parameters of the invoked utility and specifying a user, on behalf of which the event handler is executed. Instead of event handler options, the <parameter> templates that can be used are presented in the table below:

Table 2.2.

Component name/description	Command line parameter		Notes
	Old component value	New component value	
For the events 'client_create', 'client_update', 'client_delete'			
Login Name	old_login_name	new_login_name	required
Contact Name	old_contact_name	new_contact_name	required
Company Name	old_company_name	new_company_name	

Component name/description	Command line parameter		Notes
	Old component value	New component value	
Phone	old_phone	new_phone	
Fax	old_fax	new_fax	
E-mail	old_email	new_email	
Address	old_address	new_address	
City	old_city	new_city	
State/Province	old_state_province	new_state_province	
Postal/ZIP Code	old_postal_zip_code	new_postal_zip_code	
Country	old_country	new_country	
For the events 'domain_create', 'domain_update', 'domain_delete'			
Domain Name	old_domain_name	new_domain_name	required
For the events 'subdomain_create', 'subdomain_update', 'subdomain_delete'			
Subdomain Name	old_subdomain_name	new_subdomain_name	required
Parent Domain Name	old_domain_name	new_domain_name	required
FTP account	old_system_user_type	new_system_user_type	
Subdomain owner's login name	old_system_user	new_system_user	
Hard disk quota	old_hard_disk_quota	new_hard_disk_quota	
SSI support	old_ssi_support	new_ssi_support	
PHP support	old_php_support	new_php_support	
CGI support	old_cgi_support	new_cgi_support	
Perl support	old_mod_perl_support	new_mod_perl_support	
Python support	old_mod_python_support	new_mod_python_support	
ColdFusion support	old_coldfusion_support	new_coldfusion_support	
Apache::ASP support	old_apache_asp_support	new_apache_asp_support	
SSL support	old_ssl_support	new_ssl_support	

Component name/description	Command line parameter		Notes
	Old component value	New component value	
For the events 'phys_hosting_create', 'phys_hosting_update'			
Domain Name	old_domain_name	new_domain_name	required
IP Address	old_ip_address	new_ip_address	
IP Type	old_ip_type	new_ip_type	
System User	old_system_user	new_system_user	
System User Password	old_system_user_password	new_system_user_password	
Shell Access	old_system_shell	new_system_shell	
FP Support	old_fp_support	new_fp_support	
FP-SSL Support	old_fpssl_support	new_fpssl_support	
FP Authoring	old_fp_authoring	new_fp_authoring	
FP Admin Login	old_fp_admin_login	new_fp_admin_login	
FP Admin Password	old_fp_admin_password	new_fp_admin_password	
SSI Support	old_ssi_support	new_ssi_support	
PHP Support	old_php_support	new_php_support	
CGI Support	old_cgi_support	new_cgi_support	
Mod Perl Support	old_mod_perl_support	new_mod_perl_support	
Apache ASP Support	old_apache_asp_support	new_apache_asp_support	
SSL Support	old_ssl_support	new_ssl_support	
Web Statistics	old_web_statistics	new_web_statistics	
Custom Error Documents	old_apache_error_documents	new_apache_error_documents	
Hard Disk Quota	old_hard_disk_quota	new_hard_disk_quota	
For the event 'phys_hosting_delete'			
Domain Name	old_domain_name	new_domain_name	required

Component name/description	Command line parameter		Notes
	Old component value	New component value	
For the events 'mailname_create', 'mailname_delete'			
Mail name	old_mailname	new_mailname	required (in the format mailname@domain)
For the event 'mailname_update'			
Mail name	old_mailname	new_mailname	required (in the format mailname@domain)
Mailbox	old_mailbox	new_mailbox	
Password	old_password	new_password	
Mailbox Quota	old_mailbox_quota	new_mailbox_quota	
Redirect	old_redirect	new_redirect	
Redirect Address	old_redirect_address	new_redirect_address	
Mail Group	old_mail_group	new_mail_group	
Autoresponders	old_autoresponders	new_autoresponders	
Mail User Control Panel Access	old_mail_controlpanel_access	new_mail_controlpanel_access	
For the event 'webuser_delete'			
Domain Name	old_domain_name	new_domain_name	required
Webuser Name	old_webuser_name	new_webuser_name	required
For the events 'webuser_create', 'webuser_update'			
Domain Name	old_domain_name	new_domain_name	required
Web User Name	old_webuser_name	new_webuser_name	required
Web User Password	old_webuser_password	new_webuser_password	
SSI Support	old_ssi_support	new_ssi_support	
PHP Support	old_php_support	new_php_support	
CGI Support	old_cgi_support	new_cgi_support	

Component name/description	Command line parameter		Notes
	Old component value	New component value	
Mod Perl Support	old_mod_perl_support	new_mod_perl_support	
Mod Python Support	old_mod_python_support	new_mod_python_support	
Apache ASP Support	old_apache_asp_support	new_apache_asp_support	
Hard Disk Quota	old_hard_disk_quota	new_hard_disk_quota	
For the event 'client_limits_update'			
Contact Name	old_contact_name	new_contact_name	required
Maximum Number of Domains	old_maximum_domains	new_maximum_domains	
Maximum Amount of Disk Space	old_maximum_disk_space	new_maximum_disk_space	
Maximum Amount of Traffic	old_maximum_traffic	new_maximum_traffic	
Maximum Number of Web Users	old_maximum_webusers	new_maximum_webusers	
Maximum Number of Databases	old_maximum_databases	new_maximum_databases	
Maximum Number of Mailboxes	old_maximum_mailboxes	new_maximum_mailboxes	
Mailbox Quota	old_maximum_mailbox_quota	new_maximum_mailbox_quota	
Maximum Number of Mail Redirects	old_maximum_mail_redirects	new_maximum_mail_redirects	
Maximum Number of Mail Groups	old_maximum_mail_groups	new_maximum_mail_groups	
Maximum Number of Mail Autoresponders	old_maximum_mail_autoresponders	new_maximum_mail_autoresponders	
Maximum Number of Mailing Lists	old_maximum_mail_lists	new_maximum_mail_lists	

Component name/description	Command line parameter		Notes
	Old component value	New component value	
Maximum Number of Web Applications	old_maximum_tomcat_web_applications	new_maximum_tomcat_web_applications	
Expiration Date	old_expiration_date	new_expiration_date	
For the event 'domain_limits_update'			
Domain Name	old_domain_name	new_domain_name	required
Maximum Amount of Disk Space	old_maximum_disk_space	new_maximum_disk_space	
Maximum Amount of Traffic	old_maximum_traffic	new_maximum_traffic	
Maximum Number of Web Users	old_maximum_webusers	new_maximum_webusers	
Maximum Number of Databases	old_maximum_databases	new_maximum_databases	
Maximum Number of Mailboxes	old_maximum_mailboxes	new_maximum_mailboxes	
Mailbox Quota	old_maximum_mailbox_quota	new_maximum_mailbox_quota	
Maximum Number of Mail Redirects	old_maximum_mail_redirects	new_maximum_mail_redirects	
Maximum Number of Mail Groups	old_maximum_mail_groups	new_maximum_mail_groups	
Maximum Number of Mail Autoresponders	old_maximum_mail_autoresponders	new_maximum_mail_autoresponders	
Maximum Number of Mailing Lists	old_maximum_mail_lists	new_maximum_mail_lists	
Maximum Number of Web Applications	old_maximum_tomcat_web_applications	new_maximum_tomcat_web_applications	
Expiration Date	old_expiration_date	new_expiration_date	
For the events 'maillist_create', 'maillist_update', 'maillist_delete'			
Domain Name	old_domain_name	new_domain_name	required

Component name/description	Command line parameter		Notes
	Old component value	New component value	
Mailing list name	old_mail_list_name	new_mail_list_name	required
Mailing list enabled	old_mail_list_enabled	new_mail_list_enabled	
For the events 'cp_user_login', 'cp_user_logout'			
Contact Name	old_contact_name	new_contact_name	
For the event 'dl_user_update'			
Allow domain user access	old_allow_domain_user_access	new_allow_domain_user_access	
Login Name	old_login_name	new_login_name	required
Domain Name	old_domain_name	new_domain_name	required
Contact Name	old_contact_name	new_contact_name	
Company Name	old_company_name	new_company_name	
Phone	old_phone	new_phone	
Fax	old_fax	new_fax	
E-mail	old_email	new_email	
Address	old_address	new_address	
City	old_city	new_city	
State/Province	old_state_province	new_state_province	
Postal/ZIP Code	old_postal_zip_code	new_postal_zip_code	
Country	old_country	new_country	
For the events 'siteapp_install', 'siteapp_reconfigure', 'siteapp_unchain', siteapp_uninstall'			
Site application package name	old_site_application_package_name	new_site_application_package_name	required
Domain type (domain or subdomain)	old_site_application_domain_type	new_site_application_domain_type	required
Installation path (httpdocs or httpsdocs)	old_site_application_directory	new_site_application_directory	required

Component name/description	Command line parameter		Notes
	Old component value	New component value	
Installation path within the destination directory	old_site_application_installation_prefix	new_site_application_installation_prefix	required
For the events 'siteapppkg_install', 'siteapppkg_uninstall'			
Site application package name	old_site_application_package_name	new_site_application_package_name	required

Example of adding a custom event handler

Let's set up an event handler for the 'client creation', which accepts a client name as the first parameter, and the client's login as the second. For simplicity we will use a shell-script called test-handler.sh that looks as follows:

```
#!/bin/bash
echo "-----" >> /tmp/event_handler.log
/bin/date >> /tmp/event_handler.log #
information on the event date and time
/usr/bin/id >> /tmp/event_handler.log #
information on the user, on behalf of which the script was
executed (to ensure control)
echo "client created" >> /tmp/event_handler.log #
information on the created client
echo "name: $1" >> /tmp/event_handler.log # client's
name
echo "login: $2" >> /tmp/event_handler.log # client's
login
echo "-----" >> /tmp/event_handler.log
```

This script prints some information to a file so that we could control its execution (we cannot output information to stdout/stderr, as the script is executed in the background mode).

Suppose, that our handler is located in the directory /usr/local/psa/bin (for instance). Let's register it using several SQL queries. You can do that by opening a shell at the machine with running Plesk and starting a MySQL client.

To see the list of available actions:

```
SELECT * FROM actions;
```

To find out the ID for a required action ('client_create'):

```
SELECT id FROM actions WHERE name='client_create';
```

ID will be equal to 1.

Let's register our handler for the event 'client_create' (id=1), the handler will be executed, for instance, on behalf of the user "nobody" with priority 10:

```
insert into event_handlers (action_id, priority, user,
command) values (1, 10, 'nobody',
'/usr/local/psa/bin/test-handler.sh <new_contact_name>
<new_login_name>;');
```

In the command prompt you may see the parameters in the angle brackets <new_contact_name> and <new_login_name>. Before handler execution, they will be replaced with name and login of created client respectively. The entire list of available parameters is listed in the Table 2.2. You should keep in mind that with the removal operations, the parameters of type new_xxx, contain an empty string. And with creation operations the parameters of type old_xxx contain an empty string.

NOTE

You cannot assign handlers like '/bin/echo <new_contact_name> >> /tmp/event_handler.log'

Now if you login to your Plesk control panel and create a new client, specifying the value 'Some Client' in the 'Contact name' field, and 'some_client' in the field 'Login', the handler will be invoked, and the following records will be added to the /tmp/event_handler.log:

```
...
-----
Sat Dec 30 18:31:34 NOVT 2003
uid=99(nobody) gid=99(nobody) groups=99(nobody)
client created
name: Some client
login: some_client
-----
```

If you want to specify one or few handlers more, repeat the actions above for another handler. Note, you can set the handlers execution sequence using the field 'priority' of the table event_handlers, the handlers are executed in accordance with the priority (lesser value corresponds to a higher priority).

Chapter 3. Domain skeleton manager

Skeletons are file structure templates, which are used for fast automatic creation of predefined virtual host content when creating physical hosting.

Skeleton file may contain the following top-level directories only:

- httpdocs
- httpsdocs
- cgi-bin
- conf
- anon_ftp
- error_docs

Any other directories will be ignored during skeleton application.

Allowed skeleton file types are *.tgz and *.zip archives. Zip-archives can be used only if you have the 'unzip'-software installed on the server.

Creating a skeleton

Follow these steps to create a skeleton:

1. Create the required directories (available directories are: httpdocs, httpsdocs, cgi-bin, conf, anon_ftp, error_docs)
2. Place the necessary files into the directories. These files will appear in the corresponding directories of each domain that will be created and set-up on physical hosting.
3. Pack those directories and files into an archive file. Make sure that the skeleton directories are located in the root of the archive file and not in a subdirectory.

Now the skeleton file is ready to be uploaded.

! IMPORTANT

If you upgraded from Plesk 6.x.x to Plesk 7.0.0-7.0.1, then you are likely to encounter certain compatibility problems with virtual host skeletons, as upload of skeletons containing conf directory and vhost.conf file parsing were disabled in Plesk 7.0.0-7.0.1. Thus, if you used a skeleton containing conf directory and vhost.conf file with your Plesk 6.x.x, after upgrading to Plesk 7.0.0-7.0.1 and setting up hosting for a domain you would encounter problems, including syntax errors in Apache configuration file (since entries like @domain@ will not be replaced with actual domain names). In this regard, we highly recommend upgrading to Plesk version 7.0.2 or later, as it will resolve the issue.

In respect to conf directory, the skeleton mechanism in Plesk 7.0.2 works as follows:

1. Entries like @domain@ are always resolved;
2. Upload of skeletons containing the conf directory is allowed for administrator at the Server administration page only;
3. When clients (customers) are uploading a skeleton containing the conf directory, Plesk removes the conf directory from the archive for security reasons.

Activating a custom skeleton in Plesk control panel

To activate a new custom skeleton in Plesk, follow these steps:

i NOTE

Note that each new skeleton replaces the previously used one.

1. Login to Plesk as administrator.
2. Select the Server shortcut in the navigation pane.
3. Click Skeleton.
4. Select the archive file that contains the skeleton. Use the Browse button to locate the desired file.
5. Click Send File. The new skeleton will be uploaded and activated.

You can always revert to the default skeleton. To do so, just click the Default button on the Skeleton management page. The default skeleton will replace the

one currently used and will be activated.

Chapter 4. Creation Utilities

The creation utilities allow to create and manage client accounts, domains, hosting accounts, and manage domain preferences and services from the command line.

To manage client accounts, use the `client.sh` utility located in `/usr/local/psa/bin/`. The following commands and options can be used:

Table 4.1.

Command	Parameter	Action
<code>--create</code> or <code>-c</code>	<login_name>	creates a new client account
<code>--update</code> or <code>-u</code>	<login_name>	updates client account
<code>--remove</code> or <code>-r</code>	<login_name>	removes client account
<code>--info</code> or <code>-i</code>	<login_name>	retrieves client information
<code>--on</code>	<login_name>	enables client account
<code>--off</code>	<login_name>	disables client account
<code>--help</code> or <code>-h</code>		displays the help on utility usage
Option	Parameter	Note
<code>-status</code>	<true false>	enables/disables client account (default: true)
<code>-company</code>	<string>	company name
<code>-name</code>	<string>	contact name (required for creation)
<code>-login</code>	<login_name>	control panel login name (may be used only with update command)
<code>-passwd</code>	<passwd>	control panel password (required for creation)
<code>-phone</code>	<number>	phone number
<code>-fax</code>	<number>	fax
<code>-email</code>	<string>	e-mail address

Command	Parameter	Action
-addr	<string>	street
-city	<string>	city
-state	<string>	state/province
-pcode	<string>	postal/zip code
-country	<string>	country
-notify	<true false>	enables disables notifying of client account creation

To edit client accounts, use the `client_pref.sh` utility located in `/usr/local/psa/bin/`. The following commands and options can be used:

Table 4.2.

Command	Parameter	Action
--update or -u	login_name	updates an existing client account
--info or -i	login_name	retrieves client information
--help or -h		displays the help on utility usage
Option	Parameter	Note
-create_domains	<true false>	allows client to create domains
-manage_phosting	<true false>	allows managing physical hosting
-change_limits	<true false>	allows changing domain limits
-manage_dns	<true false>	allows managing DNS
-manage_log	<true false>	allows managing log rotation
-manage_crontab	<true false>	allows managing Crontab (Scheduler)
-manage_anonftp	<true false>	allows managing Anonymous FTP
-manage_webapps	<true false>	allows managing Tomcat web applications

Command	Parameter	Action
-manage_maillists	<true false>	allows managing mailing lists
-manage_sh_access	<true false>	allows managing system access
-manage_subdomains	<true false>	allows managing subdomains
-manage_quota	<true false>	allows changing hard disk quota
-make_dumps	<true false>	allows the use of backup/restore functions
-max_dom	<number>	limits number of available domains (-1 is unlimited)
-disk_space	<number>	limits amount of available disk space to the specified value (in Megabytes)
-max_traffic	<number>	limits the amount of available traffic to the specified value (in Megabytes)
-max_box	<number>	limits the allowed number of mailboxes to the specified value
-mbox_quota	<number>	limits the mailbox quota to the specified size (in Kilobytes)
-max_redir	<number>	limits the number of mail redirects
-max_mg	<number>	limits the number of mail groups
-max_resp	<number>	limits the number of mail autoresponders
-max_wu	<number>	limits the number of web users
-max_db	<number>	limits the number of databases
-max_maillists	<number>	limits the number of mailing lists
-max_webapps	<number>	limits the number of Tomcat web applications allowed
-max_subdom	<number>	limits the number of subdomains
-expiration	<date>	limits the validity period of client account

Command	Parameter	Action
-ip_pool	<add del>:<ip1>,<ip2>,...,<ipN>	adds/deletes ip addresses to/from client's IP pool.

i NOTE

The <number> tag means any positive (or zero) integer value. The <date> tag implies date in the following format: 'YYYY-MM-DD'.

You may use the '-1' value for "unlimited" when defining any limit applicable to a client account.

You may specify any number of -ip_pool options to achieve the desired results, for example: `client_pref.sh -u 'c1' -ip_pool add:127.0.0.1,127.0.0.2 -ip_pool del:127.0.0.3`

To manage domains, use the domain.sh utility located in `/usr/local/psa/bin/`. The following commands and options can be used:

Table 4.3.

Command	Parameter	Action
--create or -c	<domain_name>	creates a new domain
--update or -u	<domain_name>	updates an existing domain
--remove or -r	<domain_name>	removes an existing domain
--info or -i	<domain_name>	retrieves domain information
--on	<domain_name>	enables domain
--off	<domain_name>	disables domain
--shells or -s		lists available shells
--help or -h		displays help on utility usage
Option	Parameter	Note
-status	<true false>	turns domain on/off (default: true)
-dom_user	<true false>	turns the domain level user account on/off (default: false)

Command	Parameter	Action
-du_passwd	<passwd>	sets up password for domain level user
-dns	<true false>	turns DNS zone for the domain on/off (by default DNS template status)
-www	<true false>	adds www prefix (default: true)
-hosting	<true false>	enables/disables hosting for the domain (default: false)
-hst_type	<phys std frm>	sets up the specified hosting type (default: physical)
-target_url	<URL>	sets up target URL (required if hst_type is "std" or "frm")
-ip	<ip_address>	ip address
-login	<login>	FTP user login name (required if hosting is physical)
-passwd_type	<plain crypt>	FTP user password type (default: plain)
-passwd	<password>	FTP password (no password is set by default)
-shell	<shell_name help>	system shell (you can use the --shells command to list available shells)
-hard_quota	<MB>	hard disk quota (0 is unlimited)
-fp	<true false>	turns on/off FrontPage support on domain (default: false)
-fp_ssl	<true false>	turns on/off frontpage over ssl support on domain (default: false)
-fpauth	<true false>	FrontPage authoring (default: false)
-fplogin	<login>	FrontPage login name
-fppasswd	<password>	FrontPage password

Command	Parameter	Action
-ssi	<true false>	SSI support on domain (default: false)
-php	<true false>	PHP support on domain (default: false)
-cgi	<true false>	CGI support on domain (default: false)
-perl	<true false>	Perl support on domain (default: false)
-asp	<true false>	ASP support on domain (default: false)
-python	<true false>	Python support on domain (default: false)
-coldfusion	<true false>	ColdFusion support on domain (default: false)
-ssl	<true false>	SSL support on domain (default: false)
-webstat	<true false>	Webalizer support on domain (default: false)
-err_docs	<true false>	custom error documents support on domain (default: false)
-log_rotate	<true false>	log rotation status (default: true)
-log_bysize	<KB>	enables log rotation by size
-log_bytime	<daily weekly monthly>	enables log rotation by time (default: daily)
-log_max_num	<number>	defines maximum number of log file instances (default: 3)
-log_compress	<true false>	enables log files compression (default: true)
-log_email	<email>	enables sending log files to e-mail
-clogin	<login>	login of existing client the domain will belong to (required for domain creation)

Command	Parameter	Action
-mail_service	<true false>	turns on/off the mail service for the domain (default: true)
-notify	<true false>	enables disables notifying of domain creation

To manage domain level preferences, use the `domain_pref.sh` utility located in `/usr/local/psa/bin/`. The following commands and options can be used:

Table 4.4.

Command	Parameter	Action
--update or -u	domain_name	sets domain preferences
--info or -i	domain_name	retrieves domain preferences information
--help or -h		displays the help on utility usage
Option	Parameter	Note
-disk_space	<MB>	limits amount of available disk space
-max_traffic	<MB/Month>	limits the amount of traffic for a domain
-max_box	<number>	limits number of mailboxes
-mbox_quota	<KB>	limits mailbox quota
-max_redir	<number>	limits the number of mail redirects
-max_mg	<number>	limits number of mail groups
-max_resp	<number>	limits the number of mail autoresponders
-max_wu	<number>	limits the number of web users
-max_db	<number>	limits the number of databases
-max_maillists	<number>	limits the number of mailing lists
-max_webapps	<number>	limits the number of allowed web applications

Command	Parameter	Action
-max_subdom	<number>	limits the number of subdomains
-expiration	<YYYY-MM-DD>	limits a validity period for the domain
-www	<true false>	www prefix for the domain
-wuscripts	<true false>	enables web users' scripting support
-webmail	<true false>	allows use of webmail
-no_usr	<bounce:txt email>	sets up bounce or catch-all for nonexisting user mail (obsolete and will be removed from future releases)
-keep_traf_stat	<number>	sets the system to retain traffic statistics for N months (specify 0 if you wish not to delete statistics)

To manage mail accounts, use the mail.sh utility located in `/usr/local/psa/bin`. The following commands and options can be used:

Table 4.5.

Command	Parameter	Action
--create or -c	<mail_name>@<domain>	creates mail account
--update or -u	<mail_name>@<domain>	updates mail account parameters
--remove or -r	<mail_name>@<domain>	removes mail account
--on	<domain>	enables mail service for domain
--off	<domain>	disables mail service for domain
--info or -i	<mail_name>@<domain>	retrieves mail account information
--help or -h		displays help on utility usage
Option	Parameter	Note
-cp_access	<true false>	enables control panel access

Command	Parameter	Action
-mailbox	<true false>	creates/removes mailbox
-passwd	<passwd>	sets mailbox password
-boxpass	<passwd>	obsolete alias for option "passwd" (this option may be removed from future releases)
-passwd_type	<plain crypt>	specifies the type of mailbox password, ignored if no password specified
-mbox_quota	<KB>	limits the mailbox quota to the desired amount
-aliases	<add del>:<name1[,name2]>	Adds or deletes mail alias(es) to/from mail name
-mgroups	<add del>:<list1[,list2]>	adds or removes mail name to/from mail group
-redirect	<true false>	switches mail redirect on/off
-rediraddr	<addr>	sets redirect to address (required if redirect is enabled)
-group	<true false>	switches mail group on/off
-groupmem	<add del>:<addr1[,addr2]>	adds/removes address(es) to/from mail group
-repo	<add del>:<file1[,file2]>	adds/removes file to/from attachments repository
-autorsp	<true false>	switches all autoresponders on/off
-autoname	<name>	autoresponder name (required for all autoresponder options)
-autostatus	<true false>	switches on/off autoresponder with specified name
-autoreq	<subj body>:<string> or <always>	defines the condition for the autoresponder to be activated whether the specified pattern is encountered in the subject or body, or to respond always.

Command	Parameter	Action
-autosubj	<original string>	the subject line to be set up into autoresponder ("Re: <incoming subject>") or a custom string.
-auto_replyto	<string>	return address that will be set up into the autoresponder's messages
-autotext	<string>	autoresponder message text
-autoatch	<add del>:<file1[,file2]>	adds/removes autoresponder attachment files
-autofrq	<number>	defines the maximum number of responses to a unique e-mail address per day
-autostor	<number>	defines the number of unique addresses to be stored for autoresponder
-autored	<addr>	defines the e-mail address to forward all incoming mail to

For managing subdomains, use the `subdomain.sh` utility located in `/usr/local/psa/bin`. The following commands and options can be used:

Table 4.6.

Command	Parameter	Action
--create or -c	<subdomain>	creates a subdomain for domain specified with -domain option
--update or -u	<subdomain>	updates an existing subdomain for domain specified with -domain option
--remove or -r	<subdomain>	removes subdomains specified with -subdomains option from the domain, specified with -domain option.
--info or -i		retrieves information on subdomains
--help or -h		displays help on utility usage

Command	Parameter	Action
Option	Parameter	Note
-domain or -d	<domain_name>	use to specify the main (parent) domain for creating the subdomain for
-new_name	<new_subdomain>	change the subdomain's name to the specified one (the option cannot be used when creating new subdomains)
-ftp_user	<native main>	use a separate FTP user account for this subdomain (native) or the same FTP user as that of the main domain (main). By default the FTP account of the main (parent) domain is used for managing the subdomain
-login	<login>	FTP user login name (can be specified if separate FTP account is used)
-passwd	<password>	FTP password (no password is set by default, and it may be specified when separate FTP account is used)
-hard_quota	<MB>	hard disk quota (use 0 for "unlimited", which is set by default). You can specify this option when using a separate FTP account.
-ssi	<true false>	SSI support on subdomain (default: false)
-php	<true false>	PHP support on subdomain (default: false)
-cgi	<true false>	CGI support on subdomain (default: false)
-perl	<true false>	Perl support on subdomain (default: false)
-asp	<true false>	ASP support on subdomain (default: false)

Command	Parameter	Action
-python	<true false>	Python support on subdomain (default: false)
-coldfusion	<true false>	ColdFusion support on subdomain (default: false)
-subdomains or -s	<subdomain[,...]>	use for specifying the subdomains to be removed or to retrieve subdomain information (--remove and --info commands respectively). By default, the --info command outputs information on all existing subdomains.

Chapter 5. Chili!Soft ASP support

Chili!Soft ASP allows to run Microsoft ASP and VBScript/JScript applications on an Apache server.

As Chili!Soft ASP has its own web based control panel Plesk has no built-in support for it. However, Chili!Soft ASP can be easily used with Plesk. Follow the instructions below to install:

1. Download the Sun Chili!Soft ASP package from the web server to the temporary directory.
2. Extract the files of Sun Chili!Soft ASP using the command `# tar xvf chiliasp-3.6.2L.1047a.tar`
3. Run the install.sh script: `# ./install.sh`
4. Read the license agreement and confirm acceptance by entering "yes"
5. Press Enter to install Sun Chili!Soft ASP by default to the directory `/opt/casp`
6. Press "y" to confirm entering the serial number, then enter the serial number when prompted. If you do not have the serial number to supply, press "n".
7. When the wizard prompts to install the bundled Apache, press "n".
8. Select the language. Press Enter for default.
9. Select the first option (Use the bundled 1.3.1 JRE) to install the bundled JRE 1.3.1 and confirm installing, pressing the y button.
10. At this step you are prompted to define how to search for the web server the Chili!Soft ASP will work with. Select the option 4 (Don't search) for the installer to skip search and go to the next step.
11. Select option 3 (Do not configure a Web server)
12. At this step you are suggested to set the administrator's shell configuration. Select the option 1 (Default configuration)
13. The installation script work finishes at this step, however the installation is not yet finished as the web server needs to be properly configured, which will require several additional steps.
14. Determine the Apache version installed:

```
# rpm -q apache. Press Enter.
```

```
apache-1.3.23-14
#
```

So you can see that the Apache version is apache 1.3.23. After that you need to check whether there is an appropriate module in the Sun Chili!Soft ASP product installation.

```
# ls -l /opt/casp/module/linux2_optimized/ Press Enter.
```

```
apache_1.3.11
apache_1.3.12
apache_1.3.14
apache_1.3.19
apache_1.3.20
apache_1.3.22
apache_1.3.6
apache_1.3.9
netscape
netscape_4.x
netscape_6.x
zeus
#
```

As you can see there is no appropriate Apache version and the latest apache version that can be used is apache_1.3.22. That is why you can simply copy this directory recursively to the directory with number of your Apache:

```
# cp -pR
/opt/casp/module/linux2_optimized/apache_1.3.22
/opt/casp/module/linux2_optimized/apache_1.3.23 Press
Enter.
```

15. Now you need to configure the ASP server to work with the client Apache:

15.1. Run the configuration script: `# /opt/casp/configure-server`

15.2. Select the option 1 (Configure the ASP Server)

15.3. Select the option 1 (Specify the Web server)

15.4. Select the option 1 (Apache)

15.5. Specify the path to the client Apache configuration file
`/etc/httpd/conf/httpd.conf`

15.6. Specify the path to the apache binary `/usr/sbin/httpd`

15.7. Select the option 1 (Apache Web server) to configure the web

server that you specified at the previous steps.

15.8. Enter "y" to confirm that all the entered information is correct.

15.9. Select the option 1 (Default configuration)

That's it. The Chili!Soft ASP is completely installed.

Now you can manage the ASP server via the control panel, which is accessible at URL <http://yourservername.com:5100>

Chapter 6. Customizable `httpd.include` per domain

In Plesk each domain has virtual hosts configuration stored in a separate file:

```
/home/httpd/vhosts/<domain-name>/conf/httpd.include
```

This file is overwritten each time the virtual host configuration is changed, thus any manual alterations made to the file are discarded. To use custom directives or redefine those inserted by Plesk, you need to create the files `vhost.conf` and/or `vhost_ssl.conf` with necessary directives in the directory `/home/httpd/vhosts/<domain-name>/conf/`

If any (or both) of these files exist by the time the main configuration file is generated, Plesk inserts the appropriate directive `Include` `/home/httpd/vhosts/<domain-name>/conf/vhost.conf` or `Include` `/home/httpd/vhosts/<domain-name>/conf/vhost_ssl.conf` into the HTTP and/or HTTPS virtual host context respectively.

For security reasons, only root can create the `vhost.conf` and `vhost_ssl.conf` files.

For the changes to take effect, you need to run the following:

```
/usr/local/psa/admin/sbin/websrvnmng --reconfigure-vhost  
--vhost-name=<domain_name>
```

Chapter 7. Global access control list in named.conf

To allow DNS transfers server-wide on Plesk, the administrator can use global access control list in named.conf.

To set up an acl, the administrator should insert into the Plesk database the values describing servers to which DNS transfers are allowed. It can be done with mysql query:

```
insert into misc (param,val) values
('DNS_Allow_Transfer1', '1.1.1.1/24');
```

To specify more hosts, use the parameters like "DNS_Allow_Transfer2" and so on:

```
insert into misc (param,val) values
('DNS_Allow_Transfer2', '2.2.2.2');
```

Hosts should be specified by IP address and optional mask.

Once you added all the required IPs, run the following command to update /etc/named.conf:

```
admin/sbin/dnsmng update any.of.your.domains
```

After that, the named.conf file will be updated with the following entries:

```
acl common-allow-transfer {
    1.1.1.1/24;
    2.2.2.2;
};
```

Name of this acl will be added to the allow-transfer section of each DNS zone.

```
zone "zone1.com" {
    type master;
    file "zone1.com";
    allow-transfer {
        common-allow-transfer;
    };
};
```

Chapter 8. Manageable Tomcat connectors ports

The default port numbers for Coyote and Warp connectors in Plesk are 9080 and 9008.

If you want Tomcat to work on other ports (e.g. 8090 and 8009), you should add two parameters to the database as in the following example:

```
insert into misc (param,val) values  
( 'coyote_connector_port', '8090');
```

```
insert into misc (param,val) values  
( 'warp_connector_port', '8009');
```

NOTE

It is recommended that you change the Tomcat ports right after Plesk is installed on server, or prior to enabling the Tomcat service for your domains.

Chapter 9. Restoring mail configuration

Mchk is an internal utility intended for use by Plesk Control Panel. However, as administrator, you can use it for restoring the Qmail and Courier-imap configuration by internal Plesk database when needed. By default mchk is running in the background mode, to execute it in the foreground, use the `-v` option.

Example: `/usr/local/psa/admin/sbin/mchk -v`

NOTE

You may not wish to restore SpamAssassin settings for mail accounts, as it requires running Perl interpreter. To speed up restoring use the `--without-spam` option.