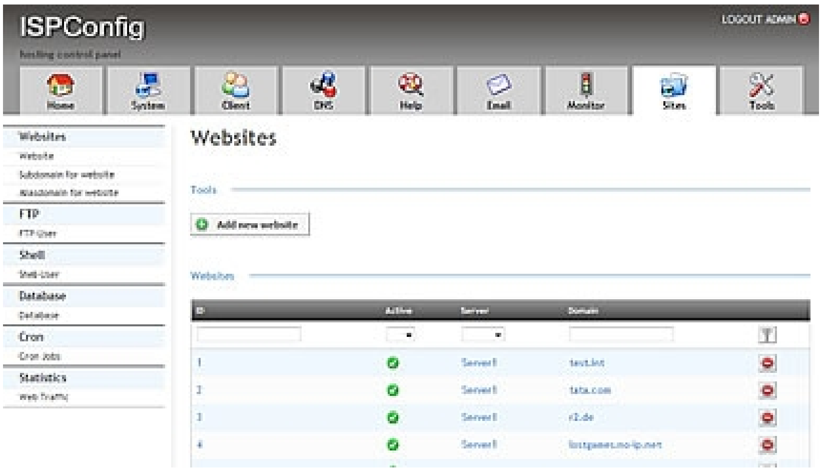
ISPConfig 3 Manual

Version 1.4 for ISPConfig 3.0.5
Author: Falko Timme <ft@falkotimme.com>
Last edited 2013-02-22

ISPConfig 3 is an open source hosting control panel for Linux and is capable of managing multiple servers from one control panel. ISPConfig 3 is licensed under BSD license.



ID	Active	Server	Domain
1	✓	Server1	test.int
2	✓	Server1	tata.com
3	✓	Server1	r2.de
4	✓	Server1	lostgames.no-ip.net

Managed Services and Features

- Manage one or more servers from one control panel (multiserver management)
- Different permission levels (administrators, resellers and clients) + email user level provided by a roundcube plugin for ISPConfig
- Apache2 and nginx (virtual hosts, domain- and IP-based)
- FTP, SFTP, SCP
- WebDAV (only with Apache2)
- DNS (A, AAAA, ALIAS, CNAME, HINFO, MX, NS, PTR, RP, SRV, TXT records)
- POP3, IMAP (Courier, Dovecot)
- Email autoresponder
- Server-based mail filtering
- Advanced email spamfilter and antivirus filter
- MySQL client-databases
- Webalizer and/or AWStats statistics
- Harddisk quota
- Mail quota
- Traffic limits and statistics
- IP addresses
- SSL
- SSI
- PHP (available PHP modes: **Apache2:** mod_php, FCGI, PHP-FPM, CGI and suPHP; **nginx:** PHP-FPM)

- Shell access
- Jailed shell access
- Firewall
- Server monitoring module
- MySQL client-database access trough phpMyAdmin
- Cron jobs (full cron jobs, jailed cron jobs, web cron jobs)
- Virtualization (OpenVZ)

If you have comments or annotations or would like to contribute to this manual, please contact the author:

Falko Timme <ft@falkotimme.com>

Table Of Contents

1 Conventions Used In This Manual

1.1 Commands

Commands to be executed on the command line are formatted as follows in this document:

```
php -q install.php
```

1.2 Contents Of Files

Contents of files are displayed as follows in this document:

```
127.0.0.1 localhost.localdomain localhost
# Auto-generated hostname. Please do not remove this comment.
78.46.230.214 server1.example.com server1
```

1.3 File Names, Protocol Names, System Specifications, Technical Specifications, User Names, Etc.

File names, protocol names, system specifications, technical specifications, user names, names of form fields, etc. are displayed as follows:

http://<hostname>:8080/
/var/vmail
/etc/fstab
admin
Email > Spamfilter > Blacklist

1.4 Highlighting

Very important details are highlighted as follows:

Please note that this automatic network configuration works only on Debian/Ubuntu and only if you have one network card which must be eth0.

2 ISPConfig Users - Admin, Resellers, And Clients

ISPConfig offers three levels of users which are all fully customizable - admin, resellers, and clients. The default user and at the same time the user with the highest permissions is *admin*. The *admin* account is created automatically when you install ISPConfig; all other users have to be created within ISPConfig (see chapters **4.5.1.1** for clients, **4.5.2.1** for resellers, and **4.9.1.1** for further admin users). *admin* has full control over the ISPConfig control panel and all its functions.

Please don't mix up *admin* with the *root* account - *root* is a system user whereas *admin* is an ISPConfig user; ISPConfig users can just log into the ISPConfig control panel, nothing more, i.e., they don't have shell access, for example.

admin can create further administrators that have the same or similar rights (see chapter **4.9.1.1**), for example you could create an administrator account with the rights to create web sites for clients, and you could create another administrator account that has full access to the DNS module only (for example if you have one web site specialist and another DNS specialist in your company).

admin can also create clients and resellers (resellers can then create clients themselves, but clients cannot create other clients - clients are the ISPConfig users with the lowest permissions). Resellers are companies or individuals that sell services (web hosting, email hosting, DNS hosting, etc.) to their clients without having to worry about the infrastructure behind it - this is all managed by *admin*. *admin* can impose limits on resellers so that they don't use up all of the server's resources. Reseller limits probably depend on what resellers are willing to pay for the service , but that is totally up to *admin* what limits he chooses.

Clients can be created by *admin* or resellers. They can have multiple web sites, email accounts, etc., but this depends on the client limits that *admin* and the reseller can set. You can have a client with 5GB of web space, 5 web sites and 10 email accounts, and you can have a client with 100GB of web space, 20 web sites, 100 email accounts and access to the DNS module.

All ISPConfig users (regardless of their role) can access ISPConfig 3 under *http(s)://<hostname>:8080/* or *http(s)://<ip_address>:8080/*.

2.1 Summary

2.1.1 admin

- *admin* manages and has full control over the system.
- *admin* can add other control panel users (users with administrator functions, resellers and clients).
- *admin* can have his own clients independent of resellers.

2.1.2 Resellers

- Resellers can have access to almost all modules (except the system configuration) or only to a limited set of modules, depending on the permissions given by *admin*.
- Resellers can create clients.
- Depending on the limits set by *admin*, resellers can see a limited set of resources to their clients (web space, email accounts, etc.).

2.1.3 Clients

- Clients can create web sites, email accounts, etc., but that depends on the resources given to them by their reseller or *admin*.

3 Installation & Updating

In this chapter I will explain how you can install ISPConfig 3 on your server(s). As ISPConfig 3 is multiserver-capable, we have to differentiate between three scenarios:

- The most common setup is to have one web, email, DNS, MySQL database server, i.e. a single server that hosts all services, and install ISPConfig 3 on it (**single server setup**).
- The second scenario is to control multiple servers from just one ISPConfig 3 installation, where each server can host all services (web, email, DNS, MySQL), but it is also possible to split up services (e.g. dedicated web servers, dedicated email servers, dedicated DNS servers, dedicated MySQL database servers) (**multiserver setup**).
- The third scenario is to have slave servers or mirrors of the ISPConfig 3 server. In this case you cannot create any items on the mirror (this server cannot be selected when you create a new item in ISPConfig 3), but instead the configuration (web site configuration, email configuration, etc.) will be copied from the master to the mirror (just the configuration, not any web site contents, etc. - if you want this, you can achieve this by using **rsync** or using a cluster filesystem like **GlusterFS** or some kind of network-attached storage, and you'd have to use one of these techniques on the directories */var/www* for the web sites' contents and */var/vmail* for the emails - for MySQL databases, you'd have to use **MySQL master-master replication**). If you select a master server in the *Is mirror of Server* field (see chapter **4.9.2.1**), the server for which you select the master will act as a mirror, not as a full-fledged server. If you have a failover-IP address that you can switch between the master and the mirror (e.g. automatically with **heartbeat**/**keepalived**/etc. or manually, e.g. from your hoster's control panel), you can achieve high-availability because if the master fails, the mirror can take over (**mirror setup**). Of course, this can be mixed with a multiserver setup (i.e., you can have a cluster with full-fledged servers like in the second scenario and with mirrors).

ISPConfig 3 has two installation modes called *standard* and *expert*. *expert* is needed only for multiserver and mirror setups (see chapters **3.2** and **3.3**) - in most cases you should use *standard* mode. In *expert* mode the installer asks if the server should join an existing ISPConfig multiserver setup, and if you answer with yes (*y*), the installer asks further questions about the master server (like database details).

3.1 Single Server Setup

You can find setup instructions for various versions of Debian, Ubuntu, CentOS, Fedora, and OpenSUSE on **http://www.ispconfig.org/ispconfig-3/documentation/**. It is strongly recommended to follow these to set up your Linux server before you install ISPConfig 3.

As ISPConfig 3.0.4 contains some new features like support for nginx and Mailman mailing lists, and these features are not covered by the older setup intructions on **http://www.ispconfig.org/ispconfig-3/documentation/**, please check out chapters **3.1.1** (Nginx/PHP-FPM) and **3.1.2** (Mailman) if you want to use Nginx (instead of Apache) and Mailman with ISPConfig 3 (otherwise proceed to chapter **3.1.3**). These must be set up before installing ISPConfig 3.

3.1.1 Nginx/PHP-FPM/CGI

If you want to use nginx instead of Apache with ISPConfig, please note that your nginx version must be at least 0.8.21, and you must install PHP-FPM as well. Because Debian Squeeze comes with an older nginx version and does not have a PHP-FPM package, this chapter covers nginx and PHP-FPM installation on Ubuntu 11.04 and newer.

Nginx is available as a package for Ubuntu which we can install as follows:

```
apt-get install nginx
```

If Apache2 is already installed on the system, stop it now...

```
/etc/init.d/apache2 stop
```

... and remove Apache's system startup links:

```
insserv -r apache2
```

Start nginx afterwards:

```
/etc/init.d/nginx start
```

(If both Apache2 and nginx are installed, the ISPConfig 3 installer will ask you which one you want to use - answer `nginx` in this case. If only one of these both is installed, ISPConfig will do the necessary configuration automatically.)

We can make PHP5 work in nginx through **PHP-FPM** (PHP-FPM (FastCGI Process Manager) is an alternative PHP FastCGI implementation with some additional features useful for sites of any size, especially busier sites) which we install as follows:

```
apt-get install php5-fpm
```

PHP-FPM is a daemon process (with the init script `/etc/init.d/php5-fpm`) that runs a FastCGI server on port `9000`.

To get MySQL support in PHP, we can install the `php5-mysql` package. It's a good idea to install some other PHP5 modules as well as you might need them for your applications. You can search for available PHP5 modules like this:

```
apt-cache search php5
```

Pick the ones you need and install them like this:

```
apt-get install php5-mysql php5-curl php5-gd php5-idn php-pear php5-imagick php5-imap
php5-mcrypt php5-memcache php5-ming php5-ps php5-pspell php5-recode php5-snmp php5-sqlite
php5-tidy php5-xmlrpc php5-xsl
```

Now restart PHP-FPM:

```
/etc/init.d/php5-fpm restart
```

To get CGI support in nginx, we install Fcgiwrap.

Fcgiwrap is a CGI wrapper that should work also for complex CGI scripts and can be used for shared hosting environments because it allows each vhost to use its own *cgi-bin* directory.

Install the *fcgiwrap* package:

```
apt-get install fcgiwrap
```

After the installation, the *fcgiwrap* daemon should already be started; its socket is */var/run/fcgiwrap.socket*. If it is not running, you can use the */etc/init.d/fcgiwrap* script to start it.

That's it! Now when you create an nginx vhost, ISPConfig will take care of the correct vhost configuration.

3.1.2 Mailman

This chapter covers the Mailman installation on Debian/Ubuntu. It is probably similar on other distributions (please note that ISPConfig expects the Mailman commands (like *newlist*, *change_pw*, *rmlist*) in the directory */usr/lib/mailman/bin/*, so you might have to create symlinks).

Install Mailman as follows:

```
apt-get install mailman
```

Before we can start Mailman, a first mailing list called *mailman* must be created:

```
newlist mailman
```

```
root@server1:~# newlist mailman
Enter the email of the person running the list: <-- admin email address, e.g. info@example.com
Initial mailman password: <-- admin password for the mailman list
To finish creating your mailing list, you must edit your /etc/aliases (or
equivalent) file by adding the following lines, and possibly running the
`newaliases' program:

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

```
mailman-request:      "|/var/lib/mailman/mail/mailman request mailman"
mailman-subscribe:    "|/var/lib/mailman/mail/mailman subscribe mailman"
mailman-unsubscribe:  "|/var/lib/mailman/mail/mailman unsubscribe mailman"

Hit enter to notify mailman owner... <-- ENTER

root@server1:~#
```

Open `/etc/aliases` afterwards...

```
vi /etc/aliases
```

... and add the following lines:

```
[...]
mailman:             "|/var/lib/mailman/mail/mailman post mailman"
mailman-admin:       "|/var/lib/mailman/mail/mailman admin mailman"
mailman-bounces:     "|/var/lib/mailman/mail/mailman bounces mailman"
mailman-confirm:     "|/var/lib/mailman/mail/mailman confirm mailman"
mailman-join:        "|/var/lib/mailman/mail/mailman join mailman"
mailman-leave:       "|/var/lib/mailman/mail/mailman leave mailman"
mailman-owner:       "|/var/lib/mailman/mail/mailman owner mailman"
mailman-request:     "|/var/lib/mailman/mail/mailman request mailman"
mailman-subscribe:   "|/var/lib/mailman/mail/mailman subscribe mailman"
mailman-unsubscribe: "|/var/lib/mailman/mail/mailman unsubscribe mailman"
```

Run

```
newaliases
```

afterwards and restart Postfix:

```
/etc/init.d/postfix restart
```

Finally we must enable the Mailman Apache configuration:

```
ln -s /etc/mailman/apache.conf /etc/apache2/conf.d/mailman.conf
```

This defines the alias `/cgi-bin/mailman/` for all Apache vhosts, which means you can access the Mailman admin interface for a list at `http://<vhost>/cgi-bin/mailman/admin/<listname>`, and the web page for users of a mailing list can be found at `http://<vhost>/cgi-bin/mailman/listinfo/<listname>`.

Under `http://<vhost>/pipermail` you can find the mailing list archives.

Restart Apache afterwards:

```
/etc/init.d/apache2 restart
```

If you use nginx instead of Apache, take a look at chapter 5.27.2 to find out how to configure Mailman for nginx.

Then start the Mailman daemon:

```
/etc/init.d/mailman start
```

3.1.3 ISPConfig 3 Installation

After you've set up the base system, you can install ISPConfig 3 as follows:

```
cd /tmp

wget http://www.ispconfig.org/downloads/ISPConfig-3-stable.tar.gz

tar xfz ISPConfig-3-stable.tar.gz

cd ispconfig3_install/install/
```

The next step is to run

```
php -q install.php
```

This will start the ISPConfig 3 installer. The installer will configure all services like postfix, sasl, courier, etc. for you.

```
root@server1:/tmp/ispconfig3_install/install# php -q install.php


--------------------------------------------------------------------------------
 _____ ___________   _____              __ _         ____
|_   _/  ___| ___  /  __             / _(_)       /__
  | |  `--.| |_/ / | /  / ___  _ __ | |_ _  __ _    _/ /
  | |  `--.   __/  | |    / _ | '_ |  _| |/ _` |  |_ |
 _| |_/__/ / |     | __/ (_) | | | | | | | (_| | ___
 ___/____/_|      ____/___/|_| |_|_| |_|__, | ____/
                                              __/ |
                                             |___/
--------------------------------------------------------------------------------


>> Initial configuration

Operating System: Debian 6.0 (Squeeze/Sid) or compatible

    Following will be a few questions for primary configuration so be careful.
    Default values are in [brackets] and can be accepted with <ENTER>.
    Tap in "quit" (without the quotes) to stop the installer.


Select language (en,de) [en]: <-- ENTER

Installation mode (standard,expert) [standard]: <-- ENTER

Full qualified hostname (FQDN) of the server, eg server1.domain.tld  [server1.exampl
e.com]: <-- ENTER

MySQL server hostname [localhost]: <-- ENTER

MySQL root username [root]: <-- ENTER

MySQL root password []: <-- yourrootsqlpassword

MySQL database to create [dbispconfig]: <-- ENTER

MySQL charset [utf8]: <-- ENTER

Apache and nginx detected. Select server to use for ISPConfig: (apache,nginx) [apach
e]: <-- ENTER

Generating a 2048 bit RSA private key
..+++
.......+++
writing new private key to 'smtpd.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]: <-- ENTER
State or Province Name (full name) [Some-State]: <-- ENTER
Locality Name (eg, city) []: <-- ENTER
Organization Name (eg, company) [Internet Widgits Pty Ltd]: <-- ENTER
Organizational Unit Name (eg, section) []: <-- ENTER
Common Name (eg, YOUR name) []: <-- ENTER
Email Address []: <-- ENTER
Configuring Jailkit
Configuring SASL
Configuring PAM
Configuring Courier
Configuring Spamassassin
Configuring Amavisd
Configuring Getmail
Configuring Pureftpd
Configuring BIND
Configuring Apache
Configuring Vlogger
Configuring Apps vhost
Configuring Bastille Firewall
Configuring Fail2ban
Installing ISPConfig
ISPConfig Port [8080]: <-- ENTER

Do you want a secure (SSL) connection to the ISPConfig web interface (y,n) [y]:
<-- ENTER

Generating RSA private key, 4096 bit long modulus
....................................................................................
.................................++
........................................................++
e is 65537 (0x10001)
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]: <-- ENTER
State or Province Name (full name) [Some-State]: <-- ENTER
Locality Name (eg, city) []: <-- ENTER
Organization Name (eg, company) [Internet Widgits Pty Ltd]: <-- ENTER
Organizational Unit Name (eg, section) []: <-- ENTER
Common Name (eg, YOUR name) []: <-- ENTER
Email Address []: <-- ENTER

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []: <-- ENTER
An optional company name []: <-- ENTER
writing RSA key
Configuring DBServer
Installing ISPConfig crontab
no crontab for root
no crontab for getmail
Restarting services ...
Rather than invoking init scripts through /etc/init.d, use the service(8)
utility, e.g. service mysql restart
```

```
Since the script you are attempting to invoke has been converted to an
Upstart job, you may also use the stop(8) and then start(8) utilities,
e.g. stop mysql ; start mysql. The restart(8) utility is also available.
mysql stop/waiting
mysql start/running, process 5440
 * Stopping Postfix Mail Transport Agent postfix
   ...done.
 * Starting Postfix Mail Transport Agent postfix
   ...done.
 * Stopping SASL Authentication Daemon saslauthd
   ...done.
 * Starting SASL Authentication Daemon saslauthd
   ...done.
Stopping amavisd: amavisd-new.
Starting amavisd: amavisd-new.
 * Stopping ClamAV daemon clamd
   ...done.
 * Starting ClamAV daemon clamd
Bytecode: Security mode set to "TrustSigned".
   ...done.
 * Stopping Courier authentication services authdaemond
   ...done.
 * Starting Courier authentication services authdaemond
   ...done.
 * Stopping Courier IMAP server imapd
   ...done.
 * Starting Courier IMAP server imapd
   ...done.
 * Stopping Courier IMAP-SSL server imapd-ssl
   ...done.
 * Starting Courier IMAP-SSL server imapd-ssl
   ...done.
 * Stopping Courier POP3 server...
   ...done.
 * Starting Courier POP3 server...
   ...done.
 * Stopping Courier POP3-SSL server...
   ...done.
 * Starting Courier POP3-SSL server...
   ...done.
 * Restarting Mailman master qrunner mailmanctl
 * Waiting...
   ...done.
   ...done.
 * Restarting web server apache2
 ... waiting    ...done.
Restarting ftp server: Running: /usr/sbin/pure-ftpd-mysql-virtualchroot -l mysql:/et
```

```
c/pure-ftpd/db/mysql.conf -l pam -E -Y 1 -8 UTF-8 -H -A -O clf:/var/log/pure-ftpd/tr
ansfer.log -D -b -u 1000 -B
Installation completed.
root@server1:/tmp/ispconfig3_install/install#
```

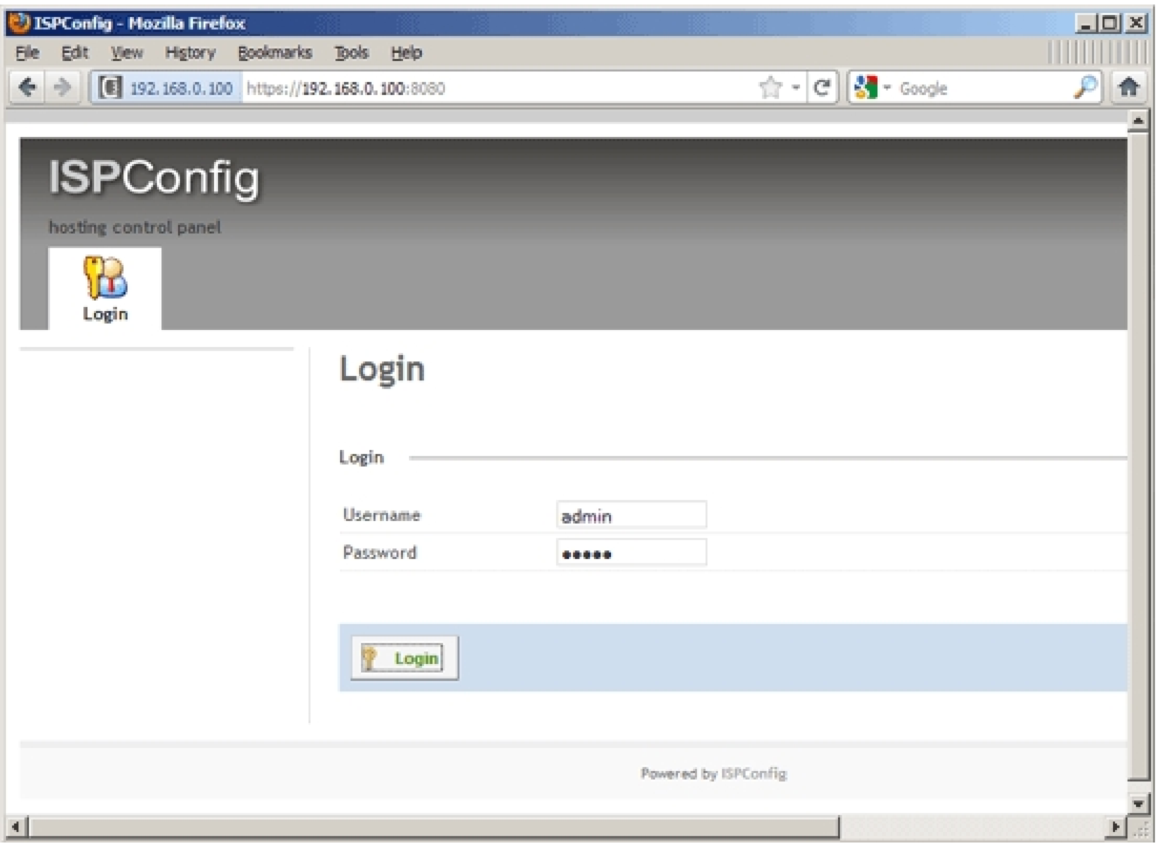
The installer automatically configures all underlying services, so no manual configuration is needed.

If you have both Apache and nginx installed, the installer asks you which one you want to use: `Apache and nginx detected. Select server to use for ISPConfig: (apache,nginx) [apache]:`

Type `nginx` if you want to use nginx, otherwise just press `ENTER` to accept Apache. If only Apache **or** nginx are installed, this is automatically detected by the installer, and no question is asked.

You now also have the possibility to let the installer create an SSL vhost for the ISPConfig control panel, so that ISPConfig can be accessed using `https://` instead of `http://`. To achieve this, just press `ENTER` when you see this question: `Do you want a secure (SSL) connection to the ISPConfig web interface (y,n) [y]:`. Of course, this works for both Apache and nginx.

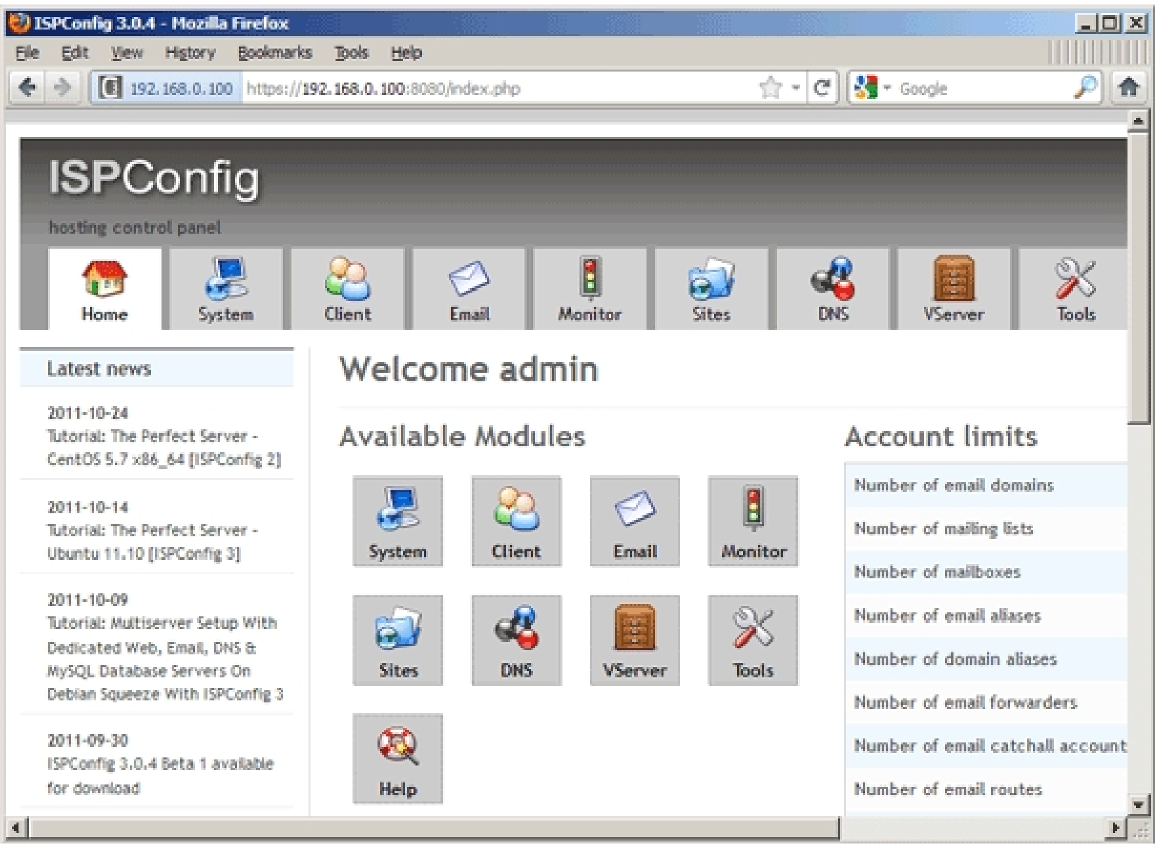
Afterwards you can access ISPConfig 3 under `http(s)://<hostname>:8080/` or `http(s)://<ip_address>:8080/` (`http` or `https` depends on what you chose during installation). Log in with the username `admin` and the password `admin` (you should change the default password after your first login):


ISPConfig - Mozilla Firefox
File Edit View History Bookmarks Tools Help
192.168.0.100 https://192.168.0.100:8080
Google
ISPConfig
hosting control panel
Login
Login
Login

Username admin

Password •••••

Login
Powered by ISPConfig


ISPConfig 3.0.4 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
192.168.0.100 https://192.168.0.100:8080/index.php
Google
ISPConfig
hosting control panel
Home
System
Client
Email
Monitor
Sites
DNS
VServer
Tools

Latest news

2011-10-24
Tutorial: The Perfect Server - CentOS 5.7 x86_64 [ISPConfig 2]

2011-10-14
Tutorial: The Perfect Server - Ubuntu 11.10 [ISPConfig 3]

2011-10-09
Tutorial: Multiserver Setup With Dedicated Web, Email, DNS & MySQL Database Servers On Debian Squeeze With ISPConfig 3

2011-09-30
ISPConfig 3.0.4 Beta 1 available for download

Welcome admin

Available Modules

Account limits

Number of email domains
Number of mailing lists
Number of mailboxes
Number of email aliases
Number of domain aliases
Number of email forwarders
Number of email catchall account
Number of email routes

The system is now ready to be used.

3.2 Multiserver Setup

The best way to describe a multiserver setup is to do this through an example. Here is a tutorial about a Debian Squeeze multiserver setup with dedicated web, email, DNS and MySQL database servers with ISPConfig 3 (i.e., the services are split up between the servers - of course, it is also possible to let all servers host all services instead of just one service).

3.2.1 Installing A Multiserver Setup With Dedicated Web, Email, DNS And MySQL Database Servers On Debian 6.0 With ISPConfig 3

This tutorial describes the installation of an ISPConfig 3 multiserver setup with dedicated web, email, database and two DNS servers all managed trough a single ISPConfig 3 control panel. The setup described below uses five servers and can be extended easily to a higher number of servers by just adding more servers. E.g. if you want to have two mailservers, do the setup steps from chapter 3.2.1.3 Installing The Mail Server on both of these servers. If you want to set up more web servers, then install ISPConfig on all other web servers in expert mode except of the first one.

3.2.1.1 Installing The Five Debian Base Systems

In this setup there will be one master server (which runs the web server and ISPConfig control panel interface) and four slave servers for database, email and DNS.

To install the clustered setup, we need five servers (or virtual servers) with a Debian 6.0 minimal install. The base setup is described in the following tutorial in the steps 1 - 6:

http://www.howtoforge.com/perfect-server-debian-squeeze-with-bind-and-dovecot-ispconfig-3

Install only steps 1 - 6 of the perfect server tutorial and not the other steps as they differ for a clustered setup!

In my example I use the following hostnames and IP addresses for the five servers:

Web Server

Hostname: *`web.example.tld`*
IP address: *`192.168.0.105`*

Mail Server

Hostname: *`mail.example.tld`*
IP address: *`192.168.0.106`*

DB Server

Hostname: *`db.example.tld`*
IP address: *`192.168.0.107`*

DNS Server (primary)

Hostname: *`ns1.example.tld`*
IP address: *`192.168.0.108`*

DNS Server (secondary)

Hostname: *`ns2.example.tld`*
IP address: *`192.168.0.109`*

Whereever these hostnames or IP addresses occur in the next installation steps you will have to change them to match the IP's and hostnames of your servers.

3.2.1.2 Installing The Web Server

Edit the hosts file and add the IP addresses and hostnames for all servers. The hostnames and IP addresses have to be adjusted to match your setup.

```
vi /etc/hosts
```

```
127.0.0.1       localhost
192.168.0.105   web.example.tld
192.168.0.106   mail.example.tld
```

```
192.168.0.107   db.example.tld
192.168.0.108   ns1.example.tld
192.168.0.109   ns2.example.tld

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

Set the hostname of the server:

```
echo web.example.tld > /etc/hostname

/etc/init.d/hostname.sh start
```

Run...

```
apt-get update
```

... to update the apt package database; then run...

```
apt-get upgrade
```

... to install the latest updates (if there are any).

It is a good idea to synchronize the system clock with an NTP (**n**etwork **t**ime **p**rotocol) server over the Internet. Simply run...

```
apt-get -y install ntp ntpdate
```

... and your system time will always be in sync.

Install the MySQL server. A MySQL server instance is necessary on every server as ISPConfig uses it to sync the configuration between the servers.

```
apt-get -y install mysql-client mysql-server
```

Enter the new password for MySQL when requested by the installer.

We want MySQL to listen on all interfaces on the master server, not just localhost, therefore we edit *`/etc/mysql/my.cnf`* and comment out the line *`bind-address = 127.0.0.1`*:

```
vi /etc/mysql/my.cnf
```

```
[...]

# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
#bind-address           = 127.0.0.1

[...]
```

Then restart MySQL:

```
/etc/init.d/mysql restart
```

Now install Apache2, PHP5, phpMyAdmin, FCGI, suExec, Pear, and mcrypt as follows:

```
apt-get -y install apache2 apache2.2-common
apache2-doc apache2-mpm-prefork apache2-utils libexpat1 ssl-cert
libapache2-mod-php5 php5 php5-common
php5-curl
```

php5-gd php5-mysql php5-imapphpmyadmin php5-cli php5-cgi libapache2-mod-fcgid apache2-suexec php-pear php-auth php5-mcrypt mcrypt php5-imagick imagemagicklibapache2-mod-suphp libruby libapache2-mod-ruby libapache2-mod-perl2 sudo zip wget

You will see the following question:

`Web server to reconfigure automatically:` <-- apache2

Then run the following command to enable the Apache modules `suexec`, `rewrite`, `ssl`, `actions`, `include`, `ruby`, `dav_fs`, `dav`, and `auth_digest`:

```
a2enmod suexec rewrite ssl actions include ruby dav_fs dav auth_digest
```

PureFTPd and quota can be installed with the following command:

```
apt-get -y install pure-ftpd-common pure-ftpd-mysql quota quotatool
```

Edit the file `/etc/default/pure-ftpd-common`...

```
vi /etc/default/pure-ftpd-common
```

... and make sure virtualchroot is set `VIRTUALCHROOT=true`:

```
[...]
VIRTUALCHROOT=true
[...]
```

Now we configure PureFTPd to allow FTP and TLS sessions. FTP is a very insecure protocol because all passwords and all data are transferred in clear text. By using TLS, the whole communication can be encrypted, thus making FTP much more secure.

If you want to allow FTP and TLS sessions, run

```
echo 1 > /etc/pure-ftpd/conf/TLS
```

In order to use TLS, we must create an SSL certificate. I create it in `/etc/ssl/private/`, therefore I create that directory first:

```
mkdir -p /etc/ssl/private/
```

Afterwards, we can generate the SSL certificate as follows:

```
openssl req -x509 -nodes -days 7300 -newkey rsa:2048 -keyout /etc/ssl/private/pure-ftpd.pem
-out /etc/ssl/private/pure-ftpd.pem
```

```
Country Name (2 letter code) [AU]: <-- Enter your Country Name (e.g., "DE").
  State or Province Name (full name) [Some-State]: <-- Enter your State or Province Name.
  Locality Name (eg, city) []: <-- Enter your City.
  Organization Name (eg, company) [Internet Widgits Pty Ltd]: <-- Enter your Organization Name (e.g., the name of your company).
  Organizational Unit Name (eg, section) []: <-- Enter your Organizational Unit Name (e.g. "IT Department").
  Common Name (eg, YOUR name) []: <-- Enter the Fully Qualified Domain Name of the system (e.g. "server1.example.com").
  Email Address []: <-- Enter your Email Address.
```

Change the permissions of the SSL certificate:

```
chmod 600 /etc/ssl/private/pure-ftpd.pem
```

Then restart PureFTPd:

```
/etc/init.d/pure-ftpd-mysql restart
```

Edit `/etc/fstab`. Mine looks like this (I added `,usrjquota=aquota.user,grpjquota=aquota.group,jqfmt=vfsv0` to the partition with the mount point `/`):

```
vi /etc/fstab
```

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
```

```
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
proc            /proc           proc    defaults        0       0
# / was on /dev/sda1 during installation
UUID=92bceda2-5ae4-4e3a-8748-b14da48fb297       /                                                       ext3
errors=remount-ro,usrjquota=aquota.user,grpjquota=aquota.group,jqfmt=vfsv0 0       1
# swap was on /dev/sda5 during installation
UUID=e24b3e9e-095c-4b49-af27-6363a4b7d094 none            swap    sw              0       0
/dev/scd0       /media/cdrom0   udf,iso9660 user,noauto     0       0
/dev/fd0        /media/floppy0  auto    rw,user,noauto  0       0
```

To enable quota, run these commands:

```
mount -o remount /
```

```
quotacheck -avugm

quotaon -avug
```

Install vlogger, webalizer, and awstats:

```
apt-get -y install vlogger webalizer awstats
```

Open */etc/cron.d/awstats* afterwards...

```
vi /etc/cron.d/awstats
```

... and comment out both cron jobs in that file:

```
#*/10 * * * * www-data [ -x /usr/share/awstats/tools/update.sh ] && /usr/share/awstats/tools/update.sh

# Generate static reports:
#10 03 * * * www-data [ -x /usr/share/awstats/tools/buildstatic.sh ] && /usr/share/awstats/tools/buildstatic.sh
```

Install Jailkit: Jailkit is needed only if you want to chroot SSH users. It can be installed as follows (important: Jailkit must be installed before ISPConfig - it cannot be installed afterwards!):

```
apt-get -y install build-essential autoconf automake1.9 libtool flex bison debhelper
```

```
cd /tmp

wget http://olivier.sessink.nl/jailkit/jailkit-2.14.tar.gz

tar xvfz jailkit-2.14.tar.gz
```

```
cd jailkit-2.14

./debian/rules binary

cd ..

dpkg -i jailkit_2.14-1_*.deb

rm -rf jailkit-2.14*
```

Install fail2ban: This is optional but recommended, because the ISPConfig monitor tries to show the log:

```
apt-get install fail2ban
```

To make fail2ban monitor PureFTPd, create the file `/etc/fail2ban/jail.local`:

```
vi /etc/fail2ban/jail.local
```

```
[pureftpd]

enabled  = true
port     = ftp
filter   = pureftpd
logpath  = /var/log/syslog
maxretry = 3
```

Then create the following filter file:

```
vi /etc/fail2ban/filter.d/pureftpd.conf
```

```
[Definition]
failregex = .*pure-ftpd: \(.*@<HOST>\) \[WARNING\] Authentication failed for user.*
ignoreregex =
```

Restart fail2ban afterwards:

```
/etc/init.d/fail2ban restart
```

Next we will install ISPConfig 3. To get the download URL of the latest ISPConfig 3 stable release, please visit the ISPConfig website: **http://www.ispconfig.org/ispconfig-3/download/**

This server is the master server in our setup which runs the ISPConfig control panel interface. To allow the other MySQL instances to connect to the MySQL database on this node during installation, we have to add MySQL root user records in the master database for every slave server hostname and IP address. The easiest way to do this is to use the web based phpmyadmin administration tool that we installed already. Open the URL *http://192.168.0.105/phpmyadmin* in a web browser, log in as MySQL root user and execute these MySQL queries:

```
CREATE USER 'root'@'192.168.0.106' IDENTIFIED BY 'myrootpassword';

GRANT ALL PRIVILEGES ON * . * TO 'root'@'192.168.0.106' IDENTIFIED BY 'myrootpassword' WITH
GRANT  OPTION  MAX_QUERIES_PER_HOUR  0  MAX_CONNECTIONS_PER_HOUR  0  MAX_UPDATES_PER_HOUR  0
MAX_USER_CONNECTIONS 0 ;
```

```
CREATE USER 'root'@'192.168.0.107' IDENTIFIED BY 'myrootpassword';

GRANT ALL PRIVILEGES ON * . * TO 'root'@'192.168.0.107' IDENTIFIED BY 'myrootpassword' WITH
GRANT  OPTION  MAX_QUERIES_PER_HOUR  0  MAX_CONNECTIONS_PER_HOUR  0  MAX_UPDATES_PER_HOUR  0
MAX_USER_CONNECTIONS 0 ;
```

```
CREATE USER 'root'@'192.168.0.108' IDENTIFIED BY 'myrootpassword';

GRANT ALL PRIVILEGES ON * . * TO 'root'@'192.168.0.108' IDENTIFIED BY 'myrootpassword' WITH
GRANT  OPTION  MAX_QUERIES_PER_HOUR  0  MAX_CONNECTIONS_PER_HOUR  0  MAX_UPDATES_PER_HOUR  0
MAX_USER_CONNECTIONS 0 ;
```

```
CREATE USER 'root'@'192.168.0.109' IDENTIFIED BY 'myrootpassword';

GRANT ALL PRIVILEGES ON * . * TO 'root'@'192.168.0.109' IDENTIFIED BY 'myrootpassword' WITH
GRANT  OPTION  MAX_QUERIES_PER_HOUR  0  MAX_CONNECTIONS_PER_HOUR  0  MAX_UPDATES_PER_HOUR  0
MAX_USER_CONNECTIONS 0 ;
```

```
CREATE USER 'root'@'mail.example.tld' IDENTIFIED BY 'myrootpassword';

GRANT  ALL  PRIVILEGES  ON  *  .  *  TO  'root'@'mail.example.tld'  IDENTIFIED  BY  'myrootpassword'
WITH  GRANT  OPTION  MAX_QUERIES_PER_HOUR  0  MAX_CONNECTIONS_PER_HOUR  0  MAX_UPDATES_PER_HOUR  0
MAX_USER_CONNECTIONS 0 ;
```

```
CREATE USER 'root'@'db.example.tld' IDENTIFIED BY 'myrootpassword';

GRANT ALL PRIVILEGES ON * . * TO 'root'@'db.example.tld' IDENTIFIED BY 'myrootpassword' WITH
GRANT  OPTION  MAX_QUERIES_PER_HOUR  0  MAX_CONNECTIONS_PER_HOUR  0  MAX_UPDATES_PER_HOUR  0
MAX_USER_CONNECTIONS 0 ;
```

```
CREATE USER 'root'@'ns1.example.tld' IDENTIFIED BY 'myrootpassword';

GRANT ALL PRIVILEGES ON * . * TO 'root'@'ns1.example.tld' IDENTIFIED BY 'myrootpassword' WITH
GRANT  OPTION  MAX_QUERIES_PER_HOUR  0  MAX_CONNECTIONS_PER_HOUR  0  MAX_UPDATES_PER_HOUR  0
```

```
MAX_USER_CONNECTIONS 0 ;
```

```
CREATE USER 'root'@'ns2.example.tld' IDENTIFIED BY 'myrootpassword';

GRANT ALL PRIVILEGES ON * . * TO 'root'@'ns2.example.tld' IDENTIFIED BY 'myrootpassword' WITH
GRANT  OPTION  MAX_QUERIES_PER_HOUR  0  MAX_CONNECTIONS_PER_HOUR  0  MAX_UPDATES_PER_HOUR  0
MAX_USER_CONNECTIONS 0 ;
```

In the above sql commands, replace the IP adresses (`192.168.0.106` - `192.168.0.109`) with the IP addresses of your servers and replace `mail.example.tld`, `db.example.tld`, `ns1.example.tld` and `ns2.example.tld` with the hostnames of your servers and `myrootpassword` with the desired root password.

Click on the reload permissions button or restart MySQL. Then close phpmyadmin.

Go back to the shell of `server1.example.tld` and download the latest ISPConfig 3 stable release:

```
cd /tmp

wget
http://www.ispconfig.org/downloads/ISPConfig-3-stable.tar.gz

tar xfz ISPConfig-3-stable.tar.gz

cd ispconfig3_install/install/
```

Then start the install script:

```
php -q install.php
```

```
Select language (en,de) [en]: <-- en
 Installation mode (standard,expert) [standard]: <-- expert
 Full qualified hostname (FQDN) of the server, eg server2.domain.tld
[web.example.tld]: <-- web.example.tld
 MySQL server hostname [localhost]: <-- localhost
 MySQL root username [root]: <-- root
 MySQL root password []: <-- Enter your MySQL root password here
 MySQL database to create [dbispconfig]: <-- dbispconfig
 MySQL charset [utf8]: <-- utf8
 Shall this server join an existing ISPConfig multiserver setup (y,n) [n]: <-- n
 Configure Mail (y,n) [y]: <-- n
 Configure Jailkit (y,n) [y]: <-- y
 Configure FTP Server (y,n) [y]: <-- y
 Configure DNS Server (y,n) [y]: <-- n
 Configure Apache Server (y,n) [y]: <-- y
 Configure Firewall Server (y,n) [y]: <--y
 Install ISPConfig Web-Interface (y,n) [y]: <--y
 ISPConfig Port [8080]: <-- 8080
  Enable SSL for the ISPConfig web interface (y,n) [y]: <-- y
_ Country Name (2 letter code) [AU]: <-- ENTER
 State or Province Name (full name) [Some-State]: <-- ENTER
 Locality Name (eg, city) []: <-- ENTER
 Organization Name (eg, company) [Internet Widgits Pty Ltd]: <-- ENTER
 Organizational Unit Name (eg, section) []: <-- ENTER
 Common Name (eg, YOUR name) []: <-- ENTER
 Email Address []: <-- ENTER
 A challenge password []: <-- ENTER
An optional company name []: <-- ENTER
```

Clean up the install directories:

```
cd /tmp

rm -rf /tmp/ispconfig3_install/install

rm -f /tmp/ISPConfig-3-stable.tar.gz
```

3.2.1.3 Installing The Mail Server

Edit the hosts file and add the IP addresses and hostnames for all servers. The hostnames and IP addresses have to be adjusted to match your setup.

```
vi /etc/hosts
```

```
127.0.0.1       localhost
192.168.0.105   web.example.tld
192.168.0.106   mail.example.tld
192.168.0.107   db.example.tld
192.168.0.108   ns1.example.tld
192.168.0.109   ns2.example.tld

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

Set the hostname of the server:

```
echo mail.example.tld > /etc/hostname
echo mail.example.tld > /etc/mailname

/etc/init.d/hostname.sh start
```

Run...

```
apt-get update
```

... to update the apt package database; then run...

```
apt-get upgrade
```

... to install the latest updates (if there are any).

It is a good idea to synchronize the system clock with an NTP (**n**etwork **t**ime **p**rotocol) server over the Internet. Simply run...

```
apt-get -y install ntp ntpdate
```

... and your system time will always be in sync.

Install postfix, dovecot and MySQL with one single command:

```
apt-get  -y  install  postfix  postfix-mysql  postfix-doc  mysql-client  mysql-server  openssl
getmail4 rkhunter binutils dovecot-imapd dovecot-pop3d
```

Enter the new password for mysql when requested by the installer and answer the next questions as decsribed below:

`General type of configuration?` <-- Internet site
`Mail name?` <-- mail.mydomain.tld

To install amavisd-new, SpamAssassin, and ClamAV, we run:

```
apt-get  -y  install  amavisd-new  spamassassin  clamav     clamav-daemon  zoo  unzip  bzip2  arj
nomarch  lzop  cabextract     apt-listchanges  libnet-ldap-perl  libauthen-sasl-perl  clamav-docs
daemon   libio-string-perl  libio-socket-ssl-perl  libnet-ident-perl  zip    libnet-dns-perl
```

If you want to use mailinglists on your server, then install mailman. This step is optional. mailman requires a apache webserver, so if you dont want to run a apache instance on your mailserver, then dont install mailman.

```
apt-get -y install mailman
```

The apt installer for mailman will ask you then to select the languages for the mailing list. Enable all languages that you want to use for mailman. Next create the "mailman" mailinglist.

```
newlist mailman
```

and enter the email address and new password for the mailinglist administrator. Thats the last step of the mailman install. The next command to install php has to be executed on every server, independently if you installed mailman or not.

Then install install the commandline version of PHP to be able to run PHP-based shell scripts for ISPConfig:

```
apt-get -y install php5-cli php5-mysql    php5-mcrypt mcrypt
```

Install fail2ban: This is optional but recommended, because the ISPConfig monitor tries to show the log:

```
apt-get install fail2ban
```

To make fail2ban monitor PureFTPd and Dovecot, create the file *`/etc/fail2ban/jail.local`*:

```
vi /etc/fail2ban/jail.local
```

```
[dovecot-pop3imap]

enabled = true
filter = dovecot-pop3imap
action = iptables-multiport[name=dovecot-pop3imap, port="pop3,pop3s,imap,imaps", protocol=tcp]
logpath = /var/log/mail.log
maxretry = 5
```

Then create the following filter file:

```
vi /etc/fail2ban/filter.d/dovecot-pop3imap.conf
```

```
[Definition]
failregex = (?: pop3-login|imap-login): .*(?:Authentication failure|Aborted login \(auth failed|Aborted login \(tried to use disabled|Disconnected \(auth failed|Aborted login \(\d+ authentication attempts).*rip=(?P<host>\S*),.*
ignoreregex =
```

Restart fail2ban afterwards:

```
/etc/init.d/fail2ban restart
```

Now I will install ISPConfig 3 on this server. To get the download URL of the latest ISPConfig 3 stable release, please visit the ISPConfig website: **http://www.ispconfig.org/ispconfig-3/download/**

Download the latest ISPConfig 3 stable release:

```
cd /tmp
```

```
wget
http://www.ispconfig.org/downloads/ISPConfig-3-stable.tar.gz

tar xfz ISPConfig-3-stable.tar.gz

cd ispconfig3_install/install/
```

Then start the install script:

```
php -q install.php
```

```
Select language (en,de) [en]: <-- en
 Installation mode (standard,expert) [standard]: <-- expert
 Full qualified hostname (FQDN) of the server, eg server1.domain.tld [mail.example.tld]: <-- mail.example.tld
 MySQL server hostname [localhost]: <-- localhost
 MySQL root username [root]: <-- root
 MySQL root password []: <-- Enter your MySQL root password here
 MySQL database to create [dbispconfig]: <-- dbispconfig
 MySQL charset [utf8]: <-- utf8
 Shall this server join an existing ISPConfig multiserver setup (y,n) [n]: <-- y
 MySQL master server hostname []: <-- web.example.tld
 MySQL master server root username [root]: <-- root
 MySQL master server root password []: <-- Enter the root password of the master server here
 MySQL master server database name [dbispconfig]: <-- dbispconfig
 Configure Mail (y,n) [y]: <-- y

Country Name (2 letter code) [AU]: <-- DE (Enter the ISO country code where you live here)
 State or Province Name (full name) [Some-State]: <-- Niedersachsen (Enter the state where you live here)
 Locality Name (eg, city) []: <-- Lueneburg (Enter the city here)
 Organization Name (eg, company) [Internet Widgits Pty Ltd]: <-- ENTER
 Organizational Unit Name (eg, section) []: <-- ENTER
 Common Name (eg, YOUR name) []: <-- ENTER
 Email Address []: <-- ENTER

Configure Jailkit (y,n) [y]: <-- n
 Configure FTP Server (y,n) [y]: <-- n
 Configure DNS Server (y,n) [y]: <-- n
 Configure Apache Server (y,n) [y]: <-- n
 Configure Firewall Server (y,n) [y]: <--y
 Install ISPConfig Web-Interface (y,n) [y]: <--n
```

Run...

```
rm -f /var/www/ispconfig
```

... to remove the ISPConfig interface link in the `/var/www` directory.

Clean up the install directories:

```
rm -rf /tmp/ispconfig3_install/install

rm -f /tmp/ISPConfig-3-stable.tar.gz
```

3.2.1.4 Installing The MySQL Database Server

Edit the hosts file and add the IP addresses and hostnames for all servers. The hostnames and IP addresses have to be adjusted to match your setup.

```
vi /etc/hosts
```

```
127.0.0.1       localhost
192.168.0.105   web.example.tld
192.168.0.106   mail.example.tld
192.168.0.107   db.example.tld
192.168.0.108   ns1.example.tld
192.168.0.109   ns2.example.tld

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

Set the hostname of the server:

```
echo db.example.tld > /etc/hostname

/etc/init.d/hostname.sh start
```

Run...

```
apt-get update
```

... to update the apt package database; then run...

```
apt-get upgrade
```

... to install the latest updates (if there are any).

It is a good idea to synchronize the system clock with an NTP (**n**etwork **t**ime **p**rotocol) server over the Internet. Simply run...

```
apt-get -y install ntp ntpdate
```

... and your system time will always be in sync.

Install MySQL client and server:

```
apt-get -y install mysql-client mysql-server
```

Enter the new password for MySQL when requested by the installer.

We want MySQL to listen on all interfaces, not just localhost, therefore we edit *`/etc/mysql/my.cnf`* and comment out the line *`bind-address = 127.0.0.1`*:

```
vi /etc/mysql/my.cnf
```

```
[...]

# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
#bind-address           = 127.0.0.1

[...]
```

Then restart MySQL:

```
/etc/init.d/mysql restart
```

Then install install the commandline version of PHP to be able to run PHP-based shell scripts for ISPConfig:

```
apt-get -y install php5-cli php5-mysql    php5-mcrypt mcrypt
```

Install fail2ban: This is optional but recommended, because the ISPConfig monitor tries to show the log:

```
apt-get install fail2ban
```

Next install ISPConfig 3 on this server. To get the download URL of the latest ISPConfig 3 stable release, please visit the ISPConfig website: **http://www.ispconfig.org/ispconfig-3/download/**

Download the latest ISPConfig 3 stable release:

```
cd /tmp
```

```
wget
http://www.ispconfig.org/downloads/ISPConfig-3-stable.tar.gz

tar xfz ISPConfig-3-stable.tar.gz

cd ispconfig3_install/install/
```

Then start the install script:

```
php -q install.php
```

```
Select language (en,de) [en]: <-- en
 Installation mode (standard,expert) [standard]: <-- expert
 Full qualified hostname (FQDN) of the server, eg server1.domain.tld
[db.example.tld]: <-- db.example.tld
 MySQL server hostname [localhost]: <-- localhost
 MySQL root username [root]: <-- root
 MySQL root password []: <-- Enter your MySQL root password here
 MySQL database to create [dbispconfig]: <-- dbispconfig
 MySQL charset [utf8]: <-- utf8
 Shall this server join an existing ISPConfig multiserver setup (y,n) [n]: <-- y
 MySQL master server hostname []: <-- web.example.tld
 MySQL master server root username [root]: <-- root
 MySQL master server root password []: <-- Enter the root password of the master server here
 MySQL master server database name [dbispconfig]: <-- dbispconfig
 Configure Mail (y,n) [y]: <-- n
 Configure Jailkit (y,n) [y]: <-- n
 Configure FTP Server (y,n) [y]: <-- n
 Configure DNS Server (y,n) [y]: <-- n
 Configure Apache Server (y,n) [y]: <-- n
 Configure Firewall Server (y,n) [y]: <--y
 Install ISPConfig Web-Interface (y,n) [y]: <--n
```

Run...

```
rm -f /var/www/ispconfig
```

... to remove the ISPConfig interface link in the `/var/www` directory.

Clean up the install directories:

```
rm -rf /tmp/ispconfig3_install/install

rm -f /tmp/ISPConfig-3-stable.tar.gz
```

3.2.1.5 Installing The Primary DNS Server

Edit the hosts file and add the IP addresses and hostnames for all servers. The hostnames and IP addresses have to be adjusted to match your setup.

```
vi /etc/hosts
```

```
127.0.0.1       localhost
192.168.0.105   web.example.tld
192.168.0.106   mail.example.tld
192.168.0.107   db.example.tld
192.168.0.108   ns1.example.tld
192.168.0.109   ns2.example.tld

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

Set the hostname of the server:

```
echo ns1.example.tld > /etc/hostname

/etc/init.d/hostname.sh start
```

Run...

```
apt-get update
```

... to update the apt package database; then run...

```
apt-get upgrade
```

... to install the latest updates (if there are any).

It is a good idea to synchronize the system clock with an NTP (**n**etwork **t**ime **p**rotocol) server over the Internet. Simply run...

```
apt-get -y install ntp ntpdate
```

... and your system time will always be in sync.

Install MySQL client and server:

```
apt-get -y install mysql-client mysql-server
```

Enter the new password for MySQL when requested by the installer.

Then install install the commandline version of PHP to be able to run PHP-based shell scripts for ISPConfig:

```
apt-get -y install php5-cli php5-mysql    php5-mcrypt mcrypt
```

Install fail2ban: This is optional but recommended, because the ISPConfig monitor tries to show the log:

```
apt-get install fail2ban
```

Install BIND DNS Server:

```
apt-get -y install bind9 dnsutils
```

Next install ISPConfig 3 on the dns server. To get the download URL of the latest ISPConfig 3 stable release, please visit the ISPConfig website: **http://www.ispconfig.org/ispconfig-3/download/**

Download the latest ISPConfig 3 stable release:

```
cd /tmp

wget
http://www.ispconfig.org/downloads/ISPConfig-3-stable.tar.gz

tar xfz ISPConfig-3-stable.tar.gz

cd ispconfig3_install/install/
```

Then start the install script:

```
php -q install.php
```

```
Select language (en,de) [en]: <-- en
 Installation mode (standard,expert) [standard]: <-- expert
 Full qualified hostname (FQDN) of the server, eg server2.domain.tld
[ns1.example.tld]: <-- ns1.example.tld
 MySQL server hostname [localhost]: <-- localhost
 MySQL root username [root]: <-- root
 MySQL root password []: <-- Enter your MySQL root password here
 MySQL database to create [dbispconfig]: <-- dbispconfig
 MySQL charset [utf8]: <-- utf8
 Shall this server join an existing ISPConfig multiserver setup (y,n) [n]: <-- y
 MySQL master server hostname []: <-- web.example.tld
 MySQL master server root username [root]: <-- root
 MySQL master server root password []: <-- Enter the root password of the master server here
 MySQL master server database name [dbispconfig]: <-- dbispconfig
 Configure Mail (y,n) [y]: <-- n
 Configure Jailkit (y,n) [y]: <-- n
 Configure FTP Server (y,n) [y]: <-- n
 Configure DNS Server (y,n) [y]: <-- y
 Configure Apache Server (y,n) [y]: <-- n
 Configure Firewall Server (y,n) [y]: <--y
 Install ISPConfig Web-Interface (y,n) [y]: <--n
```

Run...

```
rm -f /var/www/ispconfig
```

... to remove the ISPConfig interface link in the `/var/www` directory.

Clean up the install directories:

```
rm -rf /tmp/ispconfig3_install/install

rm -f /tmp/ISPConfig-3-stable.tar.gz
```

3.2.1.6 Installing The Secondary DNS Server

Edit the hosts file and add the IP addresses and hostnames for all servers. The hostnames and IP addresses have to be adjusted to match your setup.

```
vi /etc/hosts
```

```
127.0.0.1       localhost
192.168.0.105   web.example.tld
192.168.0.106   mail.example.tld
192.168.0.107   db.example.tld
192.168.0.108   ns1.example.tld
192.168.0.109   ns2.example.tld

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

Set the hostname of the server:

```
echo ns2.example.tld > /etc/hostname

/etc/init.d/hostname.sh start
```

Run...

```
apt-get update
```

... to update the apt package database; then run...

```
apt-get upgrade
```

... to install the latest updates (if there are any).

It is a good idea to synchronize the system clock with an NTP (**n**etwork **t**ime **p**rotocol) server over the Internet. Simply run...

```
apt-get -y install ntp ntpdate
```

... and your system time will always be in sync.

Install MySQL client and server:

```
apt-get -y install mysql-client mysql-server
```

Enter the new password for MySQL when requested by the installer.

Then install install the commandline version of PHP to be able to run PHP-based shell scripts for ISPConfig:

```
apt-get -y install php5-cli php5-mysql    php5-mcrypt mcrypt
```

Install fail2ban: This is optional but recommended, because the ISPConfig monitor tries to show the log:

```
apt-get install fail2ban
```

Install BIND DNS Server:

```
apt-get -y install bind9 dnsutils
```

Next install ISPConfig 3 on the dns server. To get the download URL of the latest ISPConfig 3 stable release, please visit the ISPConfig website: http://www.ispconfig.org/ispconfig-3/download/

Download the latest ISPConfig 3 stable release:

```
cd /tmp

wget
```

```
http://www.ispconfig.org/downloads/ISPConfig-3-stable.tar.gz

tar xfz ISPConfig-3-stable.tar.gz

cd ispconfig3_install/install/
```

Then start the install script:

```
php -q install.php
```

```
Select language (en,de) [en]: <-- en
 Installation mode (standard,expert) [standard]: <-- expert
 Full qualified hostname (FQDN) of the server, eg server2.domain.tld
[ns2.example.tld]: <-- ns2.example.tld
 MySQL server hostname [localhost]: <-- localhost
 MySQL root username [root]: <-- root
 MySQL root password []: <-- Enter your MySQL root password here
 MySQL database to create [dbispconfig]: <-- dbispconfig
 MySQL charset [utf8]: <-- utf8
 Shall this server join an existing ISPConfig multiserver setup (y,n) [n]: <-- y
 MySQL master server hostname []: <-- web.example.tld
 MySQL master server root username [root]: <-- root
 MySQL master server root password []: <-- Enter the root password of the master server here
 MySQL master server database name [dbispconfig]: <-- dbispconfig
 Configure Mail (y,n) [y]: <-- n
 Configure Jailkit (y,n) [y]: <-- n
 Configure FTP Server (y,n) [y]: <-- n
 Configure DNS Server (y,n) [y]: <-- y
 Configure Apache Server (y,n) [y]: <-- n
 Configure Firewall Server (y,n) [y]: <--y
 Install ISPConfig Web-Interface (y,n) [y]: <--n
```

Run...

```
rm -f /var/www/ispconfig
```

... to remove the ISPConfig interface link in the `/var/www` directory.

Clean up the install directories:

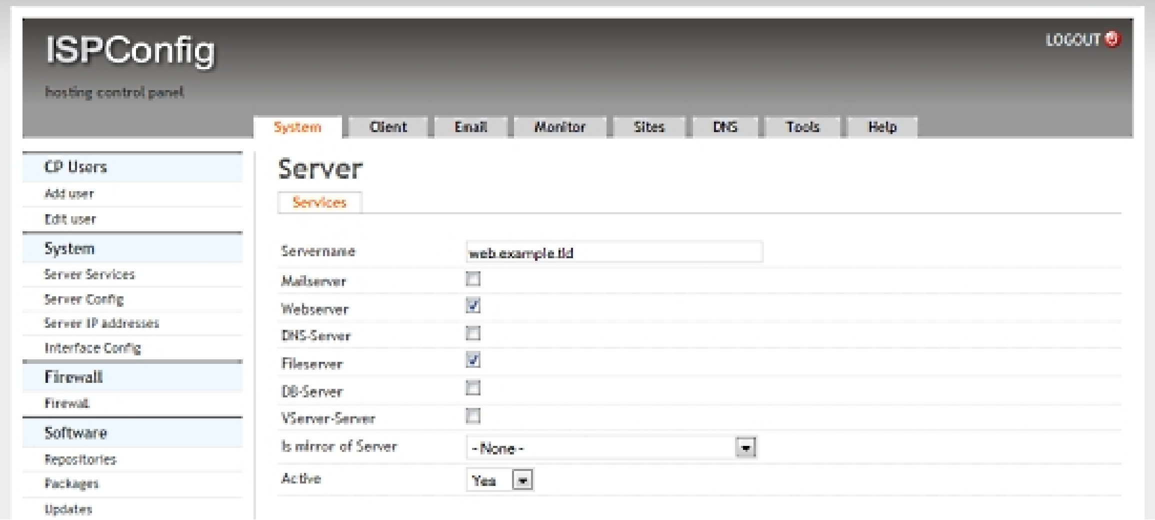
```
rm -rf /tmp/ispconfig3_install/install

rm -f /tmp/ISPConfig-3-stable.tar.gz
```

3.2.1.7 Adjust The Server Settings In ISPConfig

Log into ISPConfig on the master server with a web browser:

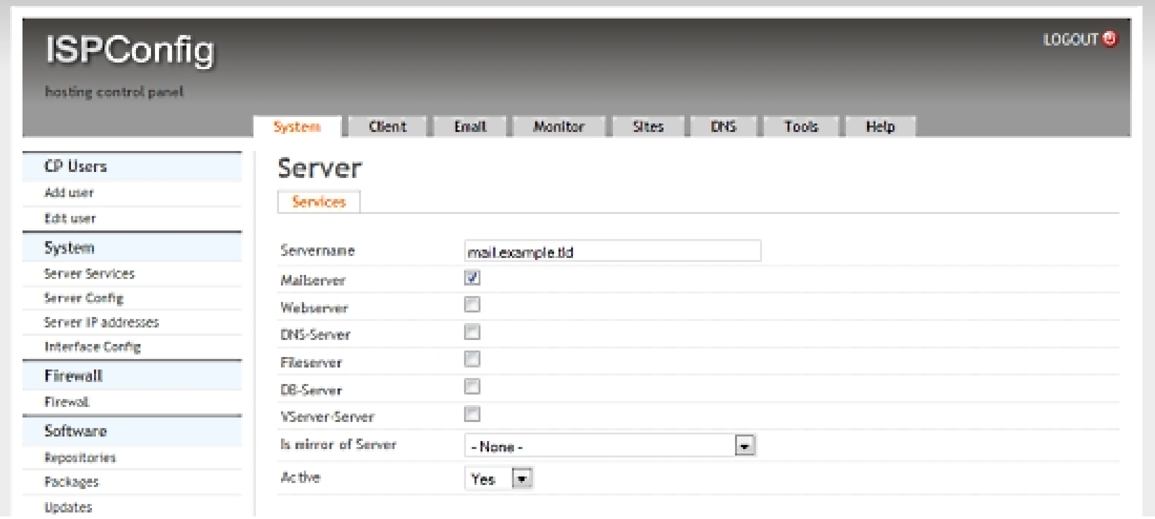
`https://192.168.0.105:8080`

Click on *`System > Server services > web.example.tld`* and disable all checkboxes except of the *`Webserver`* and *`Fileserver`* checkbox and click on *`Save`*.



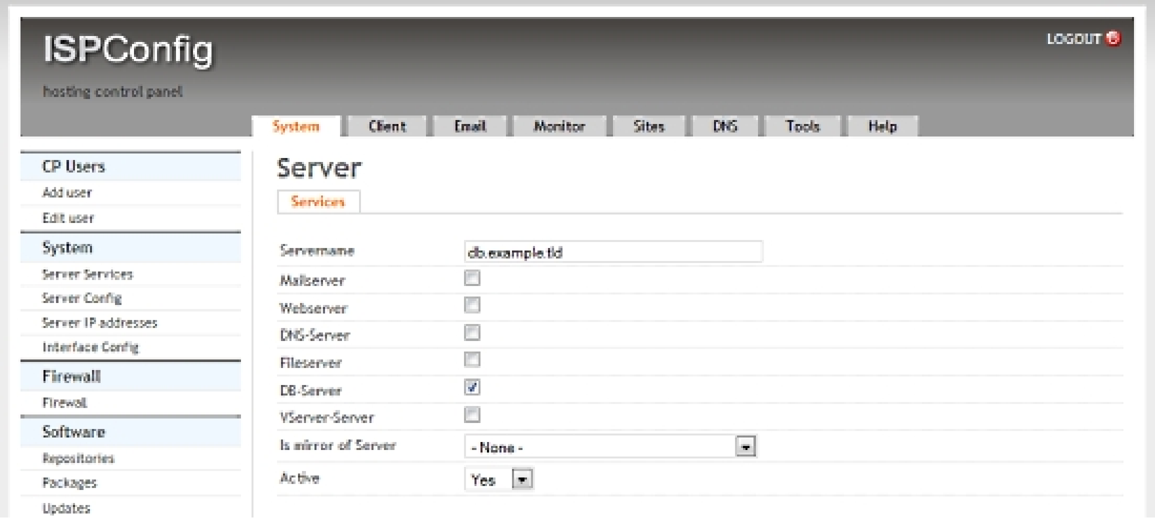
Servername	web.example.tld
Mailserver	☐
Webserver	☑
DNS-Server	☐
Fileserver	☑
DB-Server	☐
VServer-Server	☐
Is mirror of Server	- None -
Active	Yes

Click on *`System > Server services > mail.example.tld`* and disable all checkboxes except of the *`Mailserver`* checkbox and click on *`Save`*.



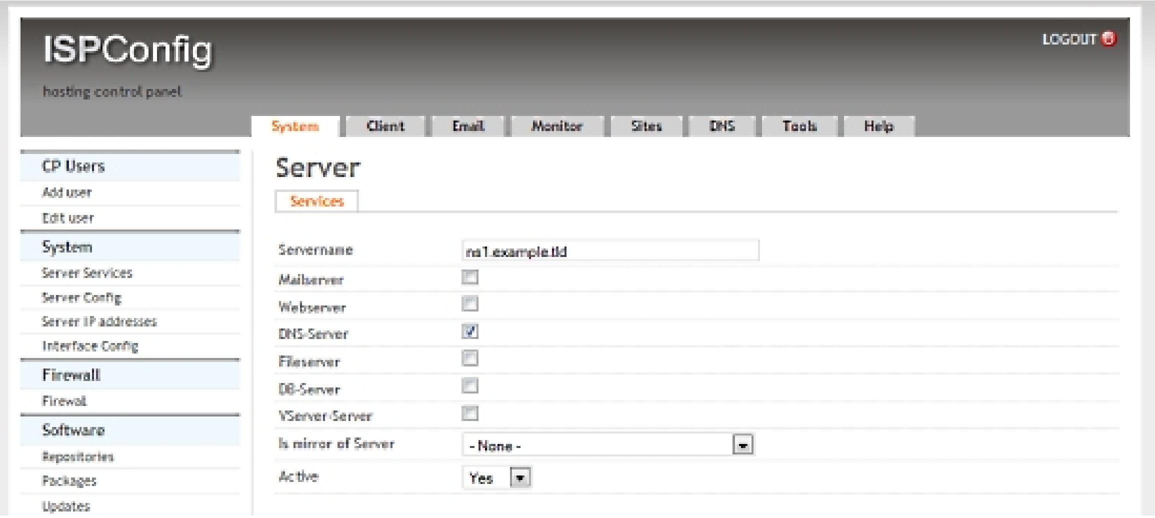
Servername	mail.example.tld
Mailserver	☑
Webserver	☐
DNS-Server	☐
Fileserver	☐
DB-Server	☐
VServer-Server	☐
Is mirror of Server	- None -
Active	Yes

Click on *`System > Server services > db.example.tld`* and disable all checkboxes except of the *`DB-Server`* checkbox and click on *`Save`*.



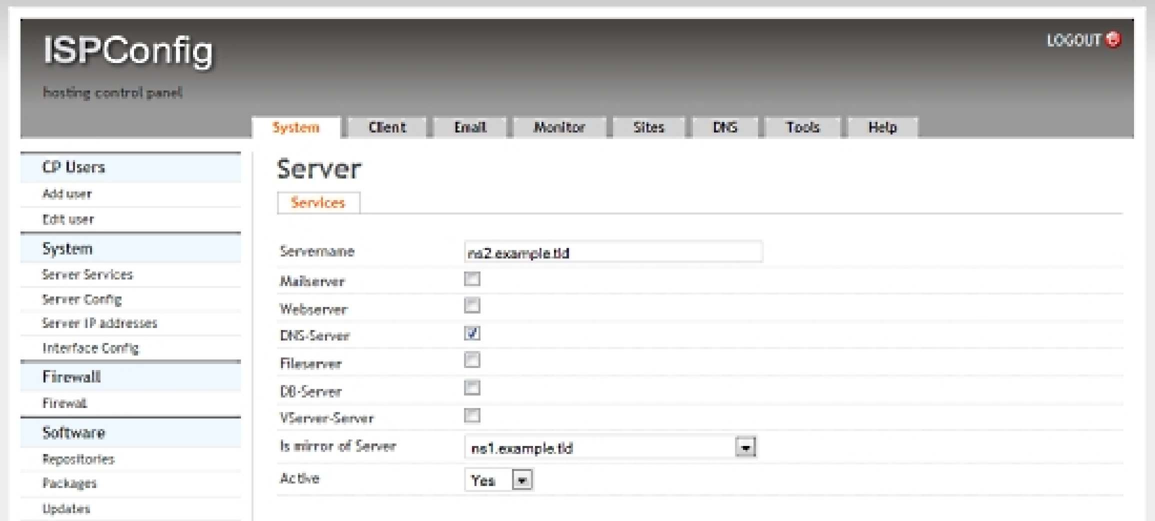
Servername: db.example.tld
Mailserver ☐
Webserver ☐
DNS-Server ☐
Fileserver ☐
DB-Server ☑
VServer-Server ☐
Is mirror of Server: - None -
Active: Yes

Click on *System > Server services > ns1.example.tld* and disable all checkboxes except of the *DNS-Server* checkbox and click on *Save*.



Servername: ns1.example.tld
Mailserver ☐
Webserver ☐
DNS-Server ☑
Fileserver ☐
DB-Server ☐
VServer-Server ☐
Is mirror of Server: - None -
Active: Yes

Click on *System > Server services > ns2.example.tld* and disable all checkboxes except of the *DNS-Server* checkbox and select *ns1.example.com* in the *Is mirror of Server* selectbox and click on *Save*.



Server

Services

Servername: ns2.example.tld
Mailserver ☐
Webserver ☐
DNS-Server ☑
Fileserver ☐
DB-Server ☐
VServer-Server ☐
Is mirror of Server: ns1.example.tld
Active: Yes

3.3 Mirror Setup

In a mirror setup, ISPConfig will copy just the configuration (web site configuration, email configuration, etc.) from the master to the mirror (i.e., not any web site contents, etc.). If you want to copy contents from the master to the mirror as well, there are several techniques that you can use, and you are free to set this up the way you like and that suits your needs best. For example, you can achieve this by using **rsync** or using a cluster filesystem like **GlusterFS** or some kind of network-attached storage, and you'd have to use one of these techniques on the directories *`/var/www`* for the web sites' contents and *`/var/vmail`* for the emails - for MySQL databases, you'd have to use **MySQL master-master replication**. If you have a failover-IP address that you can switch between the master and the mirror (e.g. automatically with **heartbeat**/**keepalived**/etc. or manually, e.g. from your hoster's control panel), you can achieve high-availability because if the master fails, the mirror can take over.

Again, it is best to demonstrate such a setup through an example. In the following tutorial, Unison is used to share contents between the master and the slave server.

3.3.1 Installing A Web, Email And MySQL Database Cluster On Debian 6.0 With ISPConfig 3

This tutorial describes the installation of a clustered web, email, database and DNS server to be used for redundancy, high availability and load balancing on Debian 6 with the ISPConfig 3 control panel. MySQL Master/Master replication will be used to replicate the MySQL client databases between the servers and Unison will be used to sync the *`/var/www`* (websites) and *`/var/vmail`* (email account data) folders.

3.3.1.1 Setting Up The Two Base Systems

In this setup there will be one master server (which runs the ISPConfig control panel interface) and one slave server which mirrors the web (apache), email (postfix and dovecot) and database (MySQL) services of the master server.

To install the clustered setup, we need two servers with a Debian 6.0 minimal install. The base setup is described in the following tutorial in the steps 1 - 8:

http://www.howtoforge.com/perfect-server-debian-squeeze-with-bind-and-dovecot-ispconfig-3

Install only steps 1 - 8 of the perfect server tutorial and not the other steps as they differ for a clustered setup!

In my example I use the following hostnames and IP addresses for the two servers:

Master Server

Hostname: *server1.example.tld*
IP-address: *192.168.0.105*

Slave server

Hostname: *server2.example.tld*
IP-address: *192.168.0.106*

Whereever these hostnames or IP addresses occur in the next installation steps you will have to change them to match the IP's and hostnames of your servers.

3.3.1.2 Installing The Two Servers

The following steps have to be executed on the master and on the slave server. If a specific step is only for the master or slave, then I've added a note in the description in red.

```
vi /etc/hosts
```

```
127.0.0.1       localhost
192.168.0.105   server1.example.tld
192.168.0.106   server2.example.tld

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

Set the hostname of the server:

```
echo server1.example.tld > /etc/hostname

/etc/init.d/hostname.sh start
```

Use server1.example.tld on the first server and server2.example.tld on the second server.

Edit the *sources.list* file...

```
vi /etc/apt/sources.list
```

... and ensure that your */etc/apt/sources.list* contains the squeeze-updates repository (this makes sure you always get the newest updates for the ClamAV virus scanner - this project publishes releases very often, and sometimes old versions stop working).

```
[...]
deb http://ftp.de.debian.org/debian/ squeeze-updates main
[...]
```

Run

```
apt-get update

apt-get upgrade
```

to install the latest updates (if there are any).

It is a good idea to synchronize the system clock with an NTP (**n**etwork **t**ime **p**rotocol) server over the Internet. Simply run

```
apt-get -y install ntp ntpdate
```

and your system time will always be in sync.

On server 1:

Now we create a private/public key pair on *server1.example.tld*:

```
ssh-keygen -t dsa
```

```
root@server1:~# ssh-keygen -t dsa
  Generating public/private dsa key pair.
  Enter file in which to save the key (/root/.ssh/id_dsa): <-- ENTER
  Created directory '/root/.ssh'.
  Enter passphrase (empty for no passphrase): <-- ENTER
  Enter same passphrase again: <-- ENTER
  Your identification has been saved in /root/.ssh/id_dsa.
  Your public key has been saved in /root/.ssh/id_dsa.pub.
  The key fingerprint is:
  1b:95:bc:4a:f4:9f:d8:ea:24:31:0f:c9:72:d5:a7:80 root@server1.example.com
  The key's randomart image is:
  +--[ DSA 1024]----+
  |                 |
  |         o o     |
  |         E * . .  |
  |        o = o o   |
  |      . S o .     |
  |        + O + .   |
  |         + + +    |
  |          o .     |
  |          .o      |
  +-----------------+
root@server1:~#
```

It is important that you do not enter a passphrase otherwise the mirroring will not work without human interaction so simply hit ENTER!

Next, we copy our public key to `server2.example.tld`:

```
ssh-copy-id -i $HOME/.ssh/id_dsa.pub root@192.168.0.106
```

```
root@server1:~# ssh-copy-id -i $HOME/.ssh/id_dsa.pub root@192.168.0.101
  The authenticity of host '192.168.0.101 (192.168.0.101)' can't be established.
  RSA key fingerprint is 25:d8:7a:ee:c2:4b:1d:92:a7:3d:16:26:95:56:62:4e.
  Are you sure you want to continue connecting (yes/no)? <-- yes (you will see this
only if this is the first time you connect to server2)
  Warning: Permanently added '192.168.0.101' (RSA) to the list of known hosts.
  root@192.168.0.101's password: <-- server2 root password
Now try logging into the machine, with "ssh 'root@192.168.0.101'", and check in:

 .ssh/authorized_keys

to make sure we haven't added extra keys that you weren't expecting.
```

Now check on `server2` if `server1`'s public key has correctly been transferred:

server2:

```
cat $HOME/.ssh/authorized_keys
```

```
ssh-dss
AAAAB3NzaC1kc3MAAACBAPhiAexgEBexnw0rFG8lXwAuIsca/V+lhmv5lhF3BqUfAbL7e2sWlQlGhxZ8I2UnzZK8Ypffq6Ks+lp46y
Os7MMXLqb7JBP9gkgqxyEWqOoUSt5hTE9ghupcCvE7rRMhefY5shLUnRkVH6hnCWe6yXSnH+Z8lHbcfp864GHkLDK1AAAAFQD
ddQckbfRG4C6LOQXTzRBpIiXzoQAAAIEAleevPHwi+a3fTDM2+Vm6EVqR5DkSLwDM7KVVNtFSkAY4GVCfhLFREsfuMkcBD9
Bv2DrKF2Ay3OOh39269Z1rgYVk+/MFC6sYgB6apirMlHj3l4RR1g09LaM1OpRz7pc/GqIGsDt74D1ES2j0zrq5kslnX8wEWSHapPR0tzi
in6UAAACBAJHxgr+GKxAdWpxV5MkF+FTaKcxA2tWHJegjGFrYGU8BpzZ4VDFMiObuzBjZ+LrUs57BiwTGB/MQl9FKQEyEV4J+
AgZCBxvg6n57YlVn6OEA0ukeJa29aFOcc0inEFfNhw2jAXt5LRyvuHD/C2gG78lwb6CxV02Z3sbTBdc43J6y root@server1.example.tld
```

Install postfix, dovecot and mysql with one single command:

```
apt-get  -y  install  postfix  postfix-mysql  postfix-doc  mysql-client  mysql-server  openssl
getmail4 rkhunter binutils dovecot-imapd dovecot-pop3d sudo
```

Enter the new password for the MySQL root user when requested by the installer. You should choose the same password for both servers. Then answer the next questions as decsribed below:

General type of configuration? <-- Internet site
Mail name? <-- server1.mydomain.tld
SSL certificate required <-- Ok

Use server1.example.tld on the first server and server2.example.tld on the second server.

We want MySQL to listen on all interfaces, not just localhost, therefore we edit */etc/mysql/my.cnf* and comment out the line *bind-address = 127.0.0.1*:

```
vi /etc/mysql/my.cnf
```

```
[...]

# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
#bind-address           = 127.0.0.1

[...]
```

Then restart MySQL:

```
/etc/init.d/mysql restart
```

Now we prepare the MySQL servers for mysql master/master replication.

On server 1:

Log into MySQL on the shell with...

```
mysql -u root -p
```

... and enter the MySQL root passord that you had choosen during mysql install. Then execute this commnd on the MySQL shell:

```
GRANT REPLICATION SLAVE ON *.* TO 'slaveuser'@'%' IDENTIFIED BY 'slave_user_password';

FLUSH PRIVILEGES;

quit;
```

Replace ***'slave_user_password'*** with a secure password that you want to use for the slave to connect to the master server. Replace this placeholder in the next steps with the password that you had choosen wherever the placeholder occurs.

Now let's configure our 2 MySQL nodes:

On server 1:

```
vi /etc/mysql/my.cnf
```

Search for the section that starts with `[mysqld]`, and put the following options into it (commenting out all existing ***conflicting*** options):

```
[...]
[mysqld]
server-id = 1
replicate-same-server-id = 0
auto-increment-increment = 2
auto-increment-offset = 1

master-host = 192.168.0.106
master-user = slaveuser
master-password = slave_user_password
master-connect-retry = 60

expire_logs_days        = 10
max_binlog_size         = 500M
log_bin                 = /var/log/mysql/mysql-bin.log
[...]
```

Then stop MySQL:

```
/etc/init.d/mysql stop
```

Now do nearly the same on `server2`...

On server 2:

```
vi /etc/mysql/my.cnf
```

Search for the section that starts with `[mysqld]`, and put the following options into it (commenting out all existing ***conflicting*** options):

```
[...]
[mysqld]
server-id = 2
replicate-same-server-id = 0
auto-increment-increment = 2
auto-increment-offset = 2

master-host = 192.168.0.105
master-user = slaveuser
```

```
master-password = slave_user_password
master-connect-retry = 60

expire_logs_days        = 10
max_binlog_size         = 500M
log_bin                 = /var/log/mysql/mysql-bin.log
[...]
```

Then stop MySQL:

```
/etc/init.d/mysql stop
```

Now we have to sync the two mysql servers. We do this by copying over the mysql data directory from the master to the slave and also the debian configuration file that contains the debian-sys-maint user. This can be done as we stopped mysql before on both servers.

On server 1:

```
scp -pr /var/lib/mysql/* root@server2.example.tld:/var/lib/mysql/

scp -pr /etc/mysql/debian.cnf root@server2.example.tld:/etc/mysql/debian.cnf
```

Now we start MySQL on the master server again:

```
/etc/init.d/mysql start
```

Log into the MySQL shell as root user...

```
mysql -u root -p
```

... and execute this command in the MySQL shell...

```
SHOW MASTER STATUS;
```

... to get the MySQL master status:

```
mysql> SHOW MASTER STATUS;
  +------------------+----------+--------------+------------------+
  | File             | Position | Binlog_Do_DB | Binlog_Ignore_DB |
  +------------------+----------+--------------+------------------+
  | mysql-bin.000002 |      106 |              |                  |
  +------------------+----------+--------------+------------------+
1 row in set (0.00 sec)
```

The information that we need for the next step is the binlog file `mysql-bin.000002` and the binlog position `106`. We need the same details for `server2` later below.

Now execute this command in the MySQL shell on the master to connect it to the slave:

```
STOP SLAVE;

CHANGE MASTER TO MASTER_HOST='192.168.0.106', MASTER_USER='slaveuser',
MASTER_PASSWORD='slave_user_password', MASTER_LOG_FILE='mysql-bin.000002',
MASTER_LOG_POS=106;

START SLAVE;
```

Then check the slave status:

```
SHOW SLAVE STATUS \G
```

It is important that both *Slave_IO_Running* and *Slave_SQL_Running* have the value *Yes* in the output.

On server 2:

Log into the MySQL shell as root user...

```
mysql -u root -p
```

... and execute this command in the MySQL shell:

```
STOP SLAVE;

CHANGE MASTER TO MASTER_HOST='192.168.0.105', MASTER_USER='slaveuser',
MASTER_PASSWORD='slave_user_password', MASTER_LOG_FILE='mysql-bin.000002',
MASTER_LOG_POS=106;

START SLAVE;
```

Then check the slave status:

```
SHOW SLAVE STATUS \G
```

It is important that both *Slave_IO_Running* and *Slave_SQL_Running* have the value *Yes* in the output

The configuration of the mysql master/master replication is finished now and we proceed to install the other software packages.

The next steps have to be executed on server 1 and server 2.

To install amavisd-new, SpamAssassin, and ClamAV, we run:

```
apt-get install amavisd-new spamassassin clamav clamav-daemon zoo unzip bzip2 arj nomarch
lzop cabextract apt-listchanges libnet-ldap-perl libauthen-sasl-perl clamav-docs daemon
libio-string-perl libio-socket-ssl-perl libnet-ident-perl zip libnet-dns-perl
```

The ISPConfig 3 setup uses amavisd which loads the SpamAssassin filter library internally, so we can stop SpamAssassin to free up some RAM:

```
/etc/init.d/spamassassin stop

update-rc.d -f spamassassin remove
```

Then install Apache2, PHP5, phpMyAdmin, FCGI, suExec, Pear, and mcrypt can be installed as follows:

```
apt-get -y install apache2 apache2.2-common apache2-doc apache2-mpm-prefork apache2-utils libexpat1 ssl-cert libapache2-mod-php5 php5 php5-common php5-curl php5-gd php5-mysql php5-imap phpmyadmin php5-cli php5-cgi libapache2-mod-fcgid apache2-suexec php-pear php-auth php5-mcrypt mcrypt php5-imagick imagemagick libapache2-mod-suphp libruby libapache2-mod-ruby php5-xcache libapache2-mod-perl2 sudo zip wget
```

You will see the following question:

`Web server to reconfigure automatically:` <-- apache2
`Configure database for phpmyadmin with dbconfig-common?` <-- No

Then run the following command to enable the Apache modules suexec, rewrite, ssl, actions, and include:

```
a2enmod suexec rewrite ssl actions include ruby dav_fs dav auth_digest
```

PureFTPd and quota can be installed with the following command:

```
apt-get -y install pure-ftpd-common pure-ftpd-mysql quota quotatool
```

Edit the file *`/etc/default/pure-ftpd-common`*...

```
vi /etc/default/pure-ftpd-common
```

... and make sure the start mode is set to standalone and set *`VIRTUALCHROOT=true`*:

```
[...]
STANDALONE_OR_INETD=standalone
[...]
VIRTUALCHROOT=true
[...]
```

Edit the file *`/etc/inetd.conf`* to prevent inetd from trying to start ftp:

```
vi /etc/inetd.conf
```

If there is a line beginning withftp stream tcp, comment it out (if there's no such file, then that is fine, and you don't have to modify /etc/inetd.conf):

```
[...]
#:STANDARD: These are standard services.
#ftp    stream  tcp     nowait  root    /usr/sbin/tcpd /usr/sbin/pure-ftpd-wrapper
[...]
```

If you had to modify */etc/inetd.conf*, restart inetd now:

```
/etc/init.d/openbsd-inetd restart
```

Now we configure PureFTPd to allow FTP and TLS sessions. FTP is a very insecure protocol because all passwords and all data are transferred in clear text. By using TLS, the whole communication can be encrypted, thus making FTP much more secure.

If you want to allow FTP and TLS sessions, run:

```
echo 1 > /etc/pure-ftpd/conf/TLS
```

In order to use TLS, we must create an SSL certificate. I create it in */etc/ssl/private/*, therefore I create that directory first:

```
mkdir -p /etc/ssl/private/
```

Afterwards, we can generate the SSL certificate as follows:

```
openssl req -x509 -nodes -days 7300 -newkey rsa:2048 -keyout /etc/ssl/private/pure-ftpd.pem
-out /etc/ssl/private/pure-ftpd.pem
```

```
Country Name (2 letter code) [AU]: <-- Enter your Country Name (e.g., "DE").
  State or Province Name (full name) [Some-State]: <-- Enter your State or Province Name.
  Locality Name (eg, city) []: <-- Enter your City.
  Organization Name (eg, company) [Internet Widgits Pty Ltd]: <-- Enter your Organization Name (e.g., the name of your company).
  Organizational Unit Name (eg, section) []: <-- Enter your Organizational Unit Name (e.g. "IT Department").
  Common Name (eg, YOUR name) []: <-- Enter the Fully Qualified Domain Name of the system (e.g. "server1.example.com").
  Email Address []: <-- Enter your Email Address.
```

Change the permissions of the SSL certificate:

```
chmod 600 /etc/ssl/private/pure-ftpd.pem
```

Then restart PureFTPd:

```
/etc/init.d/pure-ftpd-mysql restart
```

Edit */etc/fstab*. Mine looks like this (I added *,usrjquota=aquota.user,grpjquota=aquota.group,jqfmt=vfsv0* to the partition with the mount point */*):

```
vi /etc/fstab
```

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
proc            /proc           proc    defaults        0       0
# / was on /dev/sda1 during installation
UUID=92bceda2-5ae4-4e3a-8748-b14da48fb297       /                                                         ext3
errors=remount-ro,usrjquota=aquota.user,grpjquota=aquota.group,jqfmt=vfsv0 0       1
# swap was on /dev/sda5 during installation
UUID=e24b3e9e-095c-4b49-af27-6363a4b7d094 none            swap    sw              0       0
/dev/scd0       /media/cdrom0   udf,iso9660 user,noauto     0       0
/dev/fd0        /media/floppy0  auto    rw,user,noauto  0       0
```

To enable quota, run these commands:

```
mount -o remount /
```

```
quotacheck -avugm

quotaon -avug
```

Install BIND DNS Server:

```
apt-get -y install bind9 dnsutils
```

Install vlogger, webalizer, and awstats:

```
apt-get -y install vlogger webalizer awstats geoip-database
```

Open */etc/cron.d/awstats* afterwards...

```
vi /etc/cron.d/awstats
```

... and comment out both cron jobs in that file:

```
#*/10 * * * * www-data [ -x /usr/share/awstats/tools/update.sh ] && /usr/share/awstats/tools/update.sh
```

```
# Generate static reports:
 #10 03 * * * www-data [ -x /usr/share/awstats/tools/buildstatic.sh
```

Install Jailkit: Jailkit is needed only if you want to chroot SSH users. It can be installed as follows (important: Jailkit must be installed before ISPConfig - it cannot be installed afterwards!):

```
apt-get -y install build-essential autoconf automake1.9 libtool flex bison
```

```
cd /tmp

wget http://olivier.sessink.nl/jailkit/jailkit-2.14.tar.gz

tar xvfz jailkit-2.14.tar.gz

cd jailkit-2.14

./configure

make

make install

cd ..

rm -rf jailkit-2.14*
```

Install fail2ban: This is optional but recommended, because the ISPConfig monitor tries to show the log:

```
apt-get install fail2ban
```

To make fail2ban monitor PureFTPd and Dovecot, create the file *`/etc/fail2ban/jail.local`*:

```
vi /etc/fail2ban/jail.local
```

```
[pureftpd]
enabled  = true
port     = ftp
filter   = pureftpd
logpath  = /var/log/syslog
maxretry = 3

[dovecot-pop3imap]
enabled = true
filter = dovecot-pop3imap
```

```
action = iptables-multiport[name=dovecot-pop3imap, port="pop3,pop3s,imap,imaps", protocol=tcp]
logpath = /var/log/mail.log
maxretry = 5
```

Then create the following two filter files:

```
vi /etc/fail2ban/filter.d/pureftpd.conf
```

```
[Definition]
failregex = .*pure-ftpd: \(.*@<HOST>\) \[WARNING\] Authentication failed for user.*
ignoreregex =
```

```
vi /etc/fail2ban/filter.d/dovecot-pop3imap.conf
```

```
[Definition]
failregex = (?: pop3-login|imap-login): .*(?:Authentication failure|Aborted login \(auth failed|Aborted login \(tried to use disabled|Disconnected \(auth failed|Aborted login \(\d+ authentication attempts).*rip=(?P<host>\S*),.*
ignoreregex =
```

Restart fail2ban afterwards:

```
/etc/init.d/fail2ban restart
```

To install the SquirrelMail webmail client, run:

```
apt-get install squirrelmail
```

Then create the following symlink...

```
ln -s /usr/share/squirrelmail/ /var/www/webmail
```

... and configure SquirrelMail:

```
squirrelmail-configure
```

We must tell SquirrelMail that we are using Dovecot-IMAP/-POP3:

```
SquirrelMail Configuration : Read: config.php (1.4.0)
  ---------------------------------------------------------
  Main Menu --
  1.  Organization Preferences
  2.  Server Settings
  3.  Folder Defaults
  4.  General Options
  5.  Themes
  6.  Address Books
  7.  Message of the Day (MOTD)
  8.  Plugins
  9.  Database
  10. Languages

  D.  Set pre-defined settings for specific IMAP servers

  C   Turn color on
  S   Save data
  Q   Quit

  Command >> <-- D


  SquirrelMail Configuration : Read: config.php
  ---------------------------------------------------------
  While we have been building SquirrelMail, we have discovered some
  preferences that work better with some servers that don't work so
  well with others.  If you select your IMAP server, this option will
  set some pre-defined settings for that server.

  Please note that you will still need to go through and make sure
  everything is correct.  This does not change everything.  There are
  only a few settings that this will change.

  Please select your IMAP server:
    bincimap    = Binc IMAP server
    courier     = Courier IMAP server
    cyrus       = Cyrus IMAP server
    dovecot     = Dovecot Secure IMAP server
    exchange    = Microsoft Exchange IMAP server
    hmailserver = hMailServer
    macosx      = Mac OS X Mailserver
    mercury32   = Mercury/32
    uw          = University of Washington's IMAP server
    gmail       = IMAP access to Google mail (Gmail) accounts

    quit        = Do not change anything
Command >> <-- dovecot


  SquirrelMail Configuration : Read: config.php
  ---------------------------------------------------------
  While we have been building SquirrelMail, we have discovered some
  preferences that work better with some servers that don't work so
  well with others.  If you select your IMAP server, this option will
  set some pre-defined settings for that server.

  Please note that you will still need to go through and make sure
  everything is correct.  This does not change everything.  There are
  only a few settings that this will change.

  Please select your IMAP server:
    bincimap    = Binc IMAP server
    courier     = Courier IMAP server
    cyrus       = Cyrus IMAP server
    dovecot     = Dovecot Secure IMAP server
    exchange    = Microsoft Exchange IMAP server
    hmailserver = hMailServer
    macosx      = Mac OS X Mailserver
    mercury32   = Mercury/32
    uw          = University of Washington's IMAP server
    gmail       = IMAP access to Google mail (Gmail) accounts

    quit        = Do not change anything
Command >> dovecot

              imap_server_type = dovecot
         default_folder_prefix = <none>
                  trash_folder = Trash
                   sent_folder = Sent
                  draft_folder = Drafts
            show_prefix_option = false
          default_sub_of_inbox = false
show_contain_subfolders_option = false
            optional_delimiter = detect
                 delete_folder = false

  Press enter to continue... <-- press ENTER


  SquirrelMail Configuration : Read: config.php (1.4.0)
  ---------------------------------------------------------
  Main Menu --
  1.  Organization Preferences
  2.  Server Settings
  3.  Folder Defaults
  4.  General Options
  5.  Themes
  6.  Address Books
```

```
7.  Message of the Day (MOTD)
8.  Plugins
9.  Database
10. Languages

D.  Set pre-defined settings for specific IMAP servers

C   Turn color on
S   Save data
Q   Quit

Command >> <-- S


SquirrelMail Configuration : Read: config.php (1.4.0)
---------------------------------------------------------
Main Menu --
1.  Organization Preferences
2.  Server Settings
3.  Folder Defaults
4.  General Options
5.  Themes
6.  Address Books
7.  Message of the Day (MOTD)
8.  Plugins
9.  Database
10. Languages

D.  Set pre-defined settings for specific IMAP servers

C   Turn color on
S   Save data
Q   Quit

Command >> <-- Q
```

Next we enable a global Alias `/webmail` for Squirrelmail:

```
cd /etc/apache2/conf.d/

ln -s ../../squirrelmail/apache.conf squirrelmail.conf

/etc/init.d/apache2 reload
```

Now open `/etc/apache2/conf.d/squirrelmail.conf`...

```
vi /etc/apache2/conf.d/squirrelmail.conf
```

... and add the following lines to the `<Directory /usr/share/squirrelmail></Directory>` container that makes sure that mod_php is used for accessing SquirrelMail, regardless of what PHP mode you select for your website in ISPConfig:

```
[...]
Alias /webmail /usr/share/squirrelmail
<Directory /usr/share/squirrelmail>
  Options FollowSymLinks
<IfModule mod_php5.c>
  AddType application/x-httpd-php .php
  php_flag magic_quotes_gpc Off
  php_flag track_vars On
  php_admin_flag allow_url_fopen Off
  php_value include_path .
  php_admin_value upload_tmp_dir /var/lib/squirrelmail/tmp
  php_admin_value open_basedir /usr/share/squirrelmail:/etc/squirrelmail:/var/lib/squirrelmail:/etc/hostname:/etc/mailname
  php_flag register_globals off
</IfModule>
<IfModule mod_dir.c>
  DirectoryIndex index.php
</IfModule>

# access to configtest is limited by default to prevent information leak
<Files configtest.php>
  order deny,allow
  deny from all
  allow from 127.0.0.1
</Files>
</Directory>
[...]
```

Create the directory `/var/lib/squirrelmail/tmp`...

```
mkdir /var/lib/squirrelmail/tmp
```

... and make it owned by the user `www-data`:

```
chown www-data /var/lib/squirrelmail/tmp
```

Reload Apache again:

```
/etc/init.d/apache2 reload
```

That's it already - `/etc/apache2/conf.d/squirrelmail.conf` defines an alias called `/squirrelmail` that points to SquirrelMail's installation directory `/usr/share/squirrelmail`.

You can now access SquirrelMail from your web site as follows:

http://www.example.com/squirrelmail

You can also access it from the ISPConfig control panel vhost as follows (this doesn't need any configuration in ISPConfig):

http://server1.example.com:8080/squirrelmail

Next we install Unison. Unison is used to sync the */var/www* and */var/vmail* directories between master and slave

```
apt-get install unison
```

Now we install a unison configuration file on the first server.

On server 1:

Create a new file */root/.unison/default.prf* on *server1*...

```
mkdir /root/.unison

vi /root/.unison/default.prf
```

... and add the following content:

```
# Roots of the synchronization
root = /var
root = ssh://192.168.0.106//var/

# Paths to synchronize
path = www
path = vmail

# Some regexps specifying names and paths to ignore
#ignore = Path stats    ## ignores /var/www/stats
#ignore = Path stats/*  ## ignores /var/www/stats/*
#ignore = Path */stats  ## ignores /var/www/somedir/stats, but not /var/www/a/b/c/stats
#ignore = Name *stats   ## ignores all files/directories that end with "stats"
#ignore = Name stats*   ## ignores all files/directories that begin with "stats"
#ignore = Name *.tmp    ## ignores all files with the extension .tmp

#       When set to true, this flag causes the user interface to skip
#       asking for confirmations on non-conflicting changes. (More
#       precisely, when the user interface is done setting the
#       propagation direction for one entry and is about to move to the
#       next, it will skip over all non-conflicting entries and go
#       directly to the next conflict.)
auto=true
```

```
#	When this is set to true, the user interface will ask no
#	questions at all. Non-conflicting changes will be propagated;
#	conflicts will be skipped.
batch=true

#	!When this is set to true, Unison will request an extra
#	confirmation if it appears that the entire replica has been
#	deleted, before propagating the change. If the batch flag is
#	also set, synchronization will be aborted. When the path
#	preference is used, the same confirmation will be requested for
#	top-level paths. (At the moment, this flag only affects the
#	text user interface.) See also the mountpoint preference.
confirmbigdel=true

#	When this preference is set to true, Unison will use the
#	modification time and length of a file as a `pseudo inode
#	number' when scanning replicas for updates, instead of reading
#	the full contents of every file. Under Windows, this may cause
#	Unison to miss propagating an update if the modification time
#	and length of the file are both unchanged by the update.
#	However, Unison will never overwrite such an update with a
#	change from the other replica, since it always does a safe
#	check for updates just before propagating a change. Thus, it is
#	reasonable to use this switch under Windows most of the time
#	and occasionally run Unison once with fastcheck set to false,
#	if you are worried that Unison may have overlooked an update.
#	The default value of the preference is auto, which causes
#	Unison to use fast checking on Unix replicas (where it is safe)
#	and slow checking on Windows replicas. For backward
#	compatibility, yes, no, and default can be used in place of
#	true, false, and auto. See the section "Fast Checking" for more
#	information.
fastcheck=true

#	When this flag is set to true, the group attributes of the
#	files are synchronized. Whether the group names or the group
#	identifiers are synchronizeddepends on the preference numerids.
group=true

#	When this flag is set to true, the owner attributes of the
#	files are synchronized. Whether the owner names or the owner
#	identifiers are synchronizeddepends on the preference
#	extttnumerids.
owner=true

#	Including the preference -prefer root causes Unison always to
```

```
#	resolve conflicts in favor of root, rather than asking for
#	guidance from the user. (The syntax of root is the same as for
#	the root preference, plus the special values newer and older.)
#	This preference is overridden by the preferpartial preference.
#	This preference should be used only if you are sure you know
#	what you are doing!
prefer=newer

#	When this preference is set to true, the textual user interface
#	will print nothing at all, except in the case of errors.
#	Setting silent to true automatically sets the batch preference
#	to true.
silent=true

#	When this flag is set to true, file modification times (but not
#	directory modtimes) are propagated.
times=true
```

We want to automate synchronization, that is why we create a cron job for it on *server1.example.tld*:

```
crontab -e
```

```
*/5 * * * * /usr/bin/unison &> /dev/null
```

3.3.1.3 Installing ISPConfig On The First (Master) Server

In this step we will install ISPConfig on the master server. To get the download URL of the latest ISPConfig 3 stable release, please visit the ISPConfig website: ***http://www.ispconfig.org/ispconfig-3/download/***

Now we have to add two new mysql root user records in the master database to allow root access from the slave server hostname and IP address.

On server1:

Log into the MySQL database as root user...

```
mysql -u root -p
```

... and execute these mysql queries:

```
CREATE USER 'root'@'192.168.0.106' IDENTIFIED BY 'myrootpassword';

GRANT ALL PRIVILEGES ON * . * TO 'root'@'192.168.0.106' IDENTIFIED BY 'myrootpassword' WITH
GRANT OPTION MAX_QUERIES_PER_HOUR 0 MAX_CONNECTIONS_PER_HOUR 0 MAX_UPDATES_PER_HOUR 0
MAX_USER_CONNECTIONS 0 ;
```

```
CREATE USER 'root'@'server2.example.tld' IDENTIFIED BY 'myrootpassword';

GRANT ALL PRIVILEGES ON * . * TO 'root'@'server2.example.tld' IDENTIFIED BY 'myrootpassword'
WITH GRANT OPTION MAX_QUERIES_PER_HOUR 0 MAX_CONNECTIONS_PER_HOUR 0 MAX_UPDATES_PER_HOUR 0
MAX_USER_CONNECTIONS 0 ;

FLUSH PRIVILEGES;

QUIT;
```

In the above SQL commands, replace `192.168.0.106` with the IP address of the second server, replace `server2.example.tld` with the hostname of the second server and 'myrootpassword' with the desired root password.

Now you shpuld be back on the shell on `server1.example.tld` and download the latest ISPConfig 3 stable release:

```
cd /tmp

wget
http://www.ispconfig.org/downloads/ISPConfig-3-stable.tar.gz

tar xfz ISPConfig-3-stable.tar.gz

cd ispconfig3_install/install/
```

Start the install script:

```
php -q install.php
```

```
Select language (en,de) [en]: <-- en
  Installation mode (standard,expert) [standard]: <-- standard
  Full qualified hostname (FQDN) of the server, eg server1.domain.tld
[server1.example.tld]: <-- server1.example.tld
  MySQL server hostname [localhost]: <-- localhost
  MySQL root username [root]: <-- root
  MySQL root password []: <-- Enter your mysql root password here
  MySQL database to create [dbispconfig]: <-- dbispconfig1 (the local ispconfig database name of
the master and slave must be different, as both servers share the same data directory)
  MySQL charset [utf8]: <-- utf8

Country Name (2 letter code) [AU]: <-- DE (Enter the ISO country code where you live here)
  State or Province Name (full name) [Some-State]: <-- Niedersachsen (Enter the state where
you live here)
  Locality Name (eg, city) []: <-- Lueneburg (Enter the city here)
  Organization Name (eg, company) [Internet Widgits Pty Ltd]:
```

```
  Organizational Unit Name (eg, section) []:
  Common Name (eg, YOUR name) []:
  Email Address []:

Installing ISPConfig
ISPConfig Port [8080]:

Do you want a secure (SSL) connection to the ISPConfig web interface (y,n) [y]:

Generating RSA private key, 4096 bit long modulus
  ........................++
  ................++
  e is 65537 (0x10001)
  You are about to be asked to enter information that will be incorporated
  into your certificate request.
  What you are about to enter is what is called a Distinguished Name or a DN.
  There are quite a few fields but you can leave some blank
  For some fields there will be a default value,
  If you enter '.', the field will be left blank.
  -----
  Country Name (2 letter code) [AU]:
  State or Province Name (full name) [Some-State]:
  Locality Name (eg, city) []:
  Organization Name (eg, company) [Internet Widgits Pty Ltd]:
  Organizational Unit Name (eg, section) []:
  Common Name (eg, YOUR name) []:
  Email Address []:

Please enter the following 'extra' attributes
  to be sent with your certificate request
  A challenge password []:
An optional company name []:
```

Clean up the install directories:

```
rm -rf /tmp/ispconfig3_install

rm -f /tmp/ISPConfig-3-stable.tar.gz
```

3.3.1.4 Installing ISPConfig 3 On The Second Server

In this step we will install ISPConfig on the slave server (*server2.example.tld*). This time we use the expert mode of the ISPConfig installer to add this node to the master ispconfig server and database. To get the download URL of the latest ISPConfig 3 stable release, please visit the ISPConfig website: ***http://www.ispconfig.org/ispconfig-3/download/***

On server 2:

Download the latest ISPConfig 3 stable release...

```
cd /tmp

wget
http://www.ispconfig.org/downloads/ISPConfig-3-stable.tar.gz

tar xfz ISPConfig-3-stable.tar.gz

cd ispconfig3_install/install/
```

... and start the install script:

```
php -q install.php
```

```
Select language (en,de) [en]: <-- en
  Installation mode (standard,expert) [standard]: <-- expert
  Full qualified hostname (FQDN) of the server, eg server2.domain.tld
[server2.example.tld]: <-- server2.example.tld
  MySQL server hostname [localhost]: <-- localhost
  MySQL root username [root]: <-- root
  MySQL root password []: <-- Enter your mysql root password here
  MySQL database to create [dbispconfig]: <-- dbispconfig2 (the local ispconfig database name of
the master and slave must be different, as both servers share the same data directory)
  MySQL charset [utf8]: <-- utf8

  The next two questions are about the internal ISPConfig database user and
password.
  It is recommended to accept the defaults which are 'ispconfig' as username and a
random password.
  If you use a different password, use only numbers and chars for the password.

ISPConfig mysql database username [ispconfig]: <-- ispconfig2
  ISPConfig mysql database password [54c243fd3f9ca68de7b08527c81dd5ef]: <-- (press return
to accept the default)

  Shall this server join an existing ISPConfig multiserver setup (y,n) [n]: <-- y
  MySQL master server hostname []: <-- server1.example.tld
  MySQL master server root username [root]: <-- root
  MySQL master server root password []: <-- Enter the root password of the master server here
  MySQL master server database name [dbispconfig]: <-- dbispconfig1
  Configure Mail (y,n) [y]: <-- y

Country Name (2 letter code) [AU]: <-- DE (Enter the ISO country code where you live here)
  State or Province Name (full name) [Some-State]: <-- Niedersachsen (Enter the state where
you live here)
  Locality Name (eg, city) []: <-- Lueneburg (Enter the city here)
  Organization Name (eg, company) [Internet Widgits Pty Ltd]:
```

```
  Organizational Unit Name (eg, section) []:
  Common Name (eg, YOUR name) []:
  Email Address []:

Configure Jailkit (y,n) [y]: <-- y
  Configure FTP Server (y,n) [y]: <-- y
  Configure DNS Server (y,n) [y]: <-- y
  Configure Apache Server (y,n) [y]: <-- y
  Configure Firewall Server (y,n) [y]: <--y
  Install ISPConfig Web-Interface (y,n) [y]: <--y
  Installing ISPConfig
  ISPConfig Port [8080]:

Enable SSL for the ISPConfig web interface (y,n) [y]: <-- y

Generating RSA private key, 4096 bit long modulus
  .................++

....................................................................................
.........................++
  e is 65537 (0x10001)
  You are about to be asked to enter information that will be incorporated
  into your certificate request.
  What you are about to enter is what is called a Distinguished Name or a DN.
  There are quite a few fields but you can leave some blank
  For some fields there will be a default value,
  If you enter '.', the field will be left blank.
  -----
  Country Name (2 letter code) [AU]:
  State or Province Name (full name) [Some-State]:
  Locality Name (eg, city) []:
  Organization Name (eg, company) [Internet Widgits Pty Ltd]:
  Organizational Unit Name (eg, section) []:
  Common Name (eg, YOUR name) []:
  Email Address []:

Please enter the following 'extra' attributes
  to be sent with your certificate request
  A challenge password []:
  An optional company name []:
```

Clean up the install directories:

```
rm -rf /tmp/ispconfig3_install/install

rm -f /tmp/ISPConfig-3-stable.tar.gz
```

In a last configuration step, we want to connect the ISPConfig interface of the slave directly to the master database. This step is only required if you want to access ISPConfig on port 8080 on the master and on the slave server. Log into the master server as root user on the shell...

On server 1:

... and execute this command:

```
scp                    -p                    /usr/local/ispconfig/interface/lib/config.inc.php
root@192.168.0.106:/usr/local/ispconfig/interface/lib/config.inc.php
```

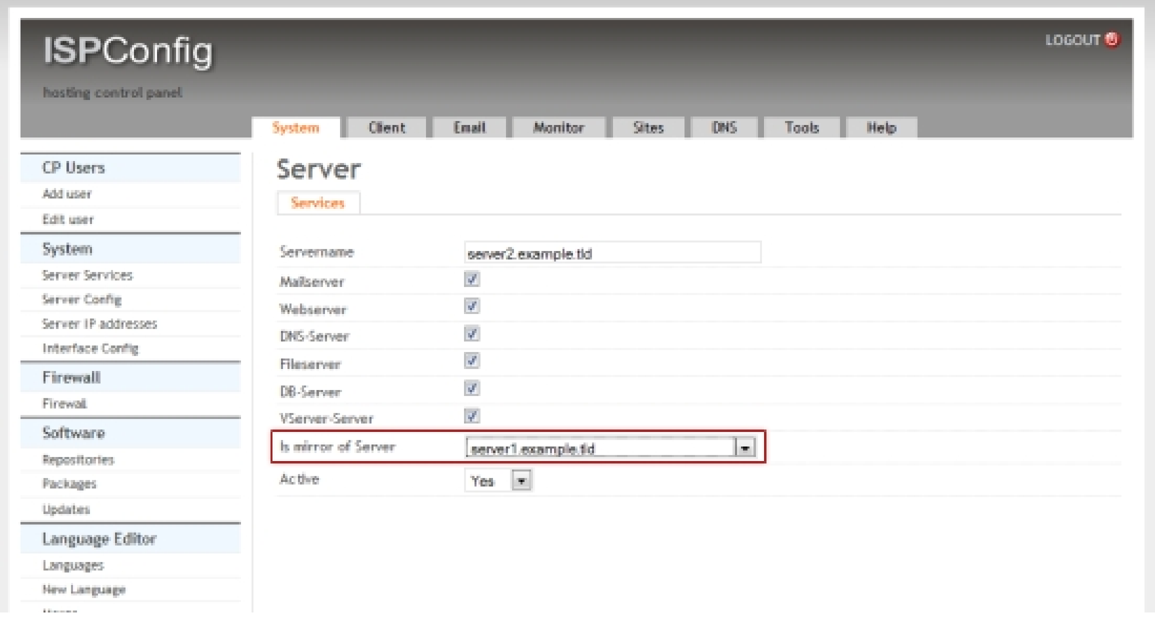
This command has to be excuted after each ISPConfig update again after you updated ISPConfig on the master and on the slave with the normal ISPConfig update command (*`ispconfig_update.sh`*).

3.3.1.5 Configure Replication In ISPConfig

Log in to ISPConfig on the master server with a web browser:

`https://192.168.0.105:8080`

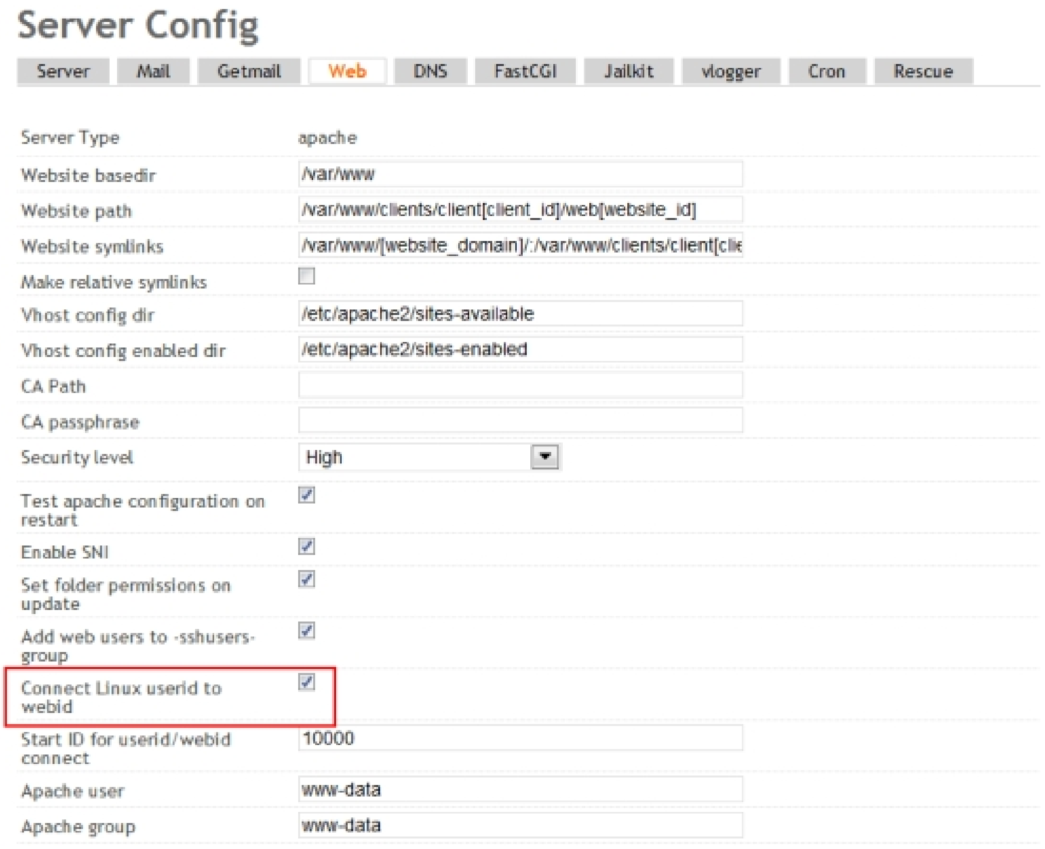
Click on *`System > Server services > server2.example.tld`*:



Servername	server2.example.tld
Mailserver	☑
Webserver	☑
DNS-Server	☑
Fileserver	☑
DB-Server	☑
VServer-Server	☑
Is mirror of Server	server1.example.tld
Active	Yes

Selecte *`server1.example.tld`* in the *`Is mirror of Server`* field and click on *`Save`*.

Then open *`System > Server Config`* and enable the checkbox "Connect Linux userid to webid" on the "Web" tab:



Server Type	apache
Website basedir	/var/www
Website path	/var/www/clients/client[client_id]/web[website_id]
Website symlinks	/var/www/[website_domain]/:/var/www/clients/client[clie
Make relative symlinks	□
Vhost config dir	/etc/apache2/sites-available
Vhost config enabled dir	/etc/apache2/sites-enabled
CA Path	
CA passphrase	
Security level	High
Test apache configuration on restart	☑
Enable SNI	☑
Set folder permissions on update	☑
Add web users to -sshusers- group	☑
Connect Linux userid to webid	☑
Start ID for userid/webid connect	10000
Apache user	www-data
Apache group	www-data

3.3.1.6 Additional Notes

If you want to activate a firewall on the master or slave server, ensure that you open port `3306` for MySQL on both servers.

3.4 Updating

Whenever there is a new ISPConfig 3 release, you can either update ISPConfig from within ISPConfig itself (see chapter 4.9.6.2) or from the command line which is stronlgy recommended right now. The procedure described in chapter 4.9.6.2 is considered experimental and should not be used on production systems.

Please note that with the command line update, you can update only the server on which you run the update, not the whole cluster (in case you run a multiserver/mirror setup). This is different from the procedure described in chapter 4.9.6.2 where you can update the whole cluster at once.

If you use the command line update to update multiple servers, it is strongly recommended to run the update on the master first and afterwards on the slave(s)!

3.4.1 Creating A Backup

Also, as a measure of precaution, you should make a backup of your ISPConfig installation before you do the update. The following items should be backed up:

- `/usr/local/ispconfig` directory
- `/etc` directory (contains configuration files of all services managed through ISPConfig)
- the ISPConfig MySQL database

You can back up these items as follows:

```
cd /usr/local

tar -pczf ispconfig.tar.gz ispconfig/
```

This creates the backup `ispconfig.tar.gz` in the `/usr/local` directory. In case you need to restore the backup, do the following:

```
cd /usr/local

rm -fr ispconfig/

tar xvfz ispconfig.tar.gz
```

To create a backup of the `/etc` directory, do the following:

```
cd /

tar -pczf etc.tar.gz etc/
```

This creates the backup `etc.tar.gz` in the `/` directory. In case you need to restore the backup, do the following:

```
cd /

rm -fr etc/

tar xvfz etc.tar.gz
```

To create a backup of your ISPConfig database in the `/usr/local` directory, do the following (assuming that your ISPConfig database is called `dbispconfig`):

```
cd /usr/local

mysqldump -h localhost -u root -p[database password] -c
--add-drop-table --add-locks --all --quick --lock-tables dbispconfig
> dbispconfig.sql
```

Please note: there's no space between *-p* and the password!

To restore the database from the SQL dump, run:

```
cd /usr/local

mysql -h localhost -u root -p[database password] dbispconfig < dbispconfig.sql
```

Please note: there's no space between *-p* and the password!

3.4.2 Command Line Update

To update ISPConfig from the command line, just run the command

```
ispconfig_update.sh
```

as root.

You can update to the last stable version or to the last version from svn. For production systems select *stable*. The update from svn is only for development systems and may break your current setup (if you want to use the svn update, please make sure that Subversion is installed on the system - on Debian/Ubuntu, you can install it as follows:

```
aptitude install subversion
```

).

It is also strongly recommended to let the update script reconfigure all services controlled by ISPConfig and also the crontab to make sure your system can make use of new ISPConfig features that come with the update.

Here is a sample output from the *ispconfig_update.sh* script (by pressing *ENTER* you accept the default value which is displayed in square brackets *[]*):

```
server1:~# ispconfig_update.sh


--------------------------------------------------------------------------------
 _____ ___________   _____              __ _
|_   _/  ___| ___   /  __              / _(_)
  | |   `--.| |_/ / | /  / ___  _ __ | |_ _  __ _
  | |  `--.    __/  | |    / _  | '_  |  _| |/ _` |
 _| |_/ __/ / |     |  __/  (_) | | | | | | | (_| |
  ___/ ____/ _|       ____/ ___/|_| |_|_| |_| __, |
                                              __/ |
                                             |___/
--------------------------------------------------------------------------------
```

```
>> Update

Please choose the update method. For production systems select 'stable'.
The update from svn is only for development systems and may break your current setup
.

Select update method (stable,svn) [stable]: <-- ENTER

[...]
# The update script downloads the new ISPConfig release here.
[...]

--------------------------------------------------------------------------------
 _____ ___________   _____              __ _
|_   _/  ___| ___   /  __              / _(_)
  | |   `--.| |_/ / | /   / ___  _ __ | |_ _  __ _
  | |  `--.    __/  | |    / _  | '_  |  _| |/ _` |
 _| |_/ __/ / |     |  __/  (_) | | | | | | | (_| |
  ___/ ____/ _|       ____/ ___/|_| |_|_| |_| __, |
                                              __/ |
                                             |___/
--------------------------------------------------------------------------------


>> Update

Operating System: Debian Lenny or compatible

This application will update ISPConfig 3 on your server.
MySQL root password []: <-- yourrootsqlpassword

Reconfigure Services? (yes,no) [yes]: <-- ENTER

Configuring Postfix
Configuring Jailkit
Configuring SASL
Configuring PAM
Configuring Courier
Configuring Spamassassin
Configuring Amavisd
Configuring Getmail
Configuring Pureftpd
Configuring BIND
Configuring Apache
Configuring vlogger
Configuring Apps vhost
Configuring Database
Configuring Firewall
Updating ISPConfig
ISPConfig Port [8080]: <-- ENTER

Reconfigure Crontab? (yes,no) [yes]: <-- ENTER

Updating Crontab
Restarting services ...
Stopping MySQL database server: mysqld.
Starting MySQL database server: mysqld.
Checking for corrupt, not cleanly closed and upgrade needing tables..
Stopping Postfix Mail Transport Agent: postfix.
Starting Postfix Mail Transport Agent: postfix.
Stopping SASL Authentication Daemon: saslauthd.
Starting SASL Authentication Daemon: saslauthd.
Stopping amavisd: (not running).
Starting amavisd: amavisd-new.
Stopping ClamAV daemon: clamd.
Starting ClamAV daemon: clamd .
Stopping Courier authentication services: authdaemond.
Starting Courier authentication services: authdaemond.
Stopping Courier IMAP server: imapd.
Starting Courier IMAP server: imapd.
Stopping Courier IMAP-SSL server: imapd-ssl.
Starting Courier IMAP-SSL server: imapd-ssl.
Stopping Courier POP3 server: pop3d.
Starting Courier POP3 server: pop3d.
Stopping Courier POP3-SSL server: pop3d-ssl.
Starting Courier POP3-SSL server: pop3d-ssl.
Restarting web server: apache2 ... waiting .
Restarting ftp server: Running: /usr/sbin/pure-ftpd-mysql-virtualchroot -l mysql:/et
c/pure-ftpd/db/mysql.conf -l pam -O clf:/var/log/pure-ftpd/transfer.log -E -H -b -A
-u 1000 -B
Update finished.
server1:~#
```

4 Reference

In the reference I explain all modules, functions, and forms in the ISPConfig control panel, i.e., I describe all input fields and give examples of what to fill in.

4.1 Tabs

ISPConfig 3 has the following tabs, depending on the modules that are enabled for the account that you used to log in:

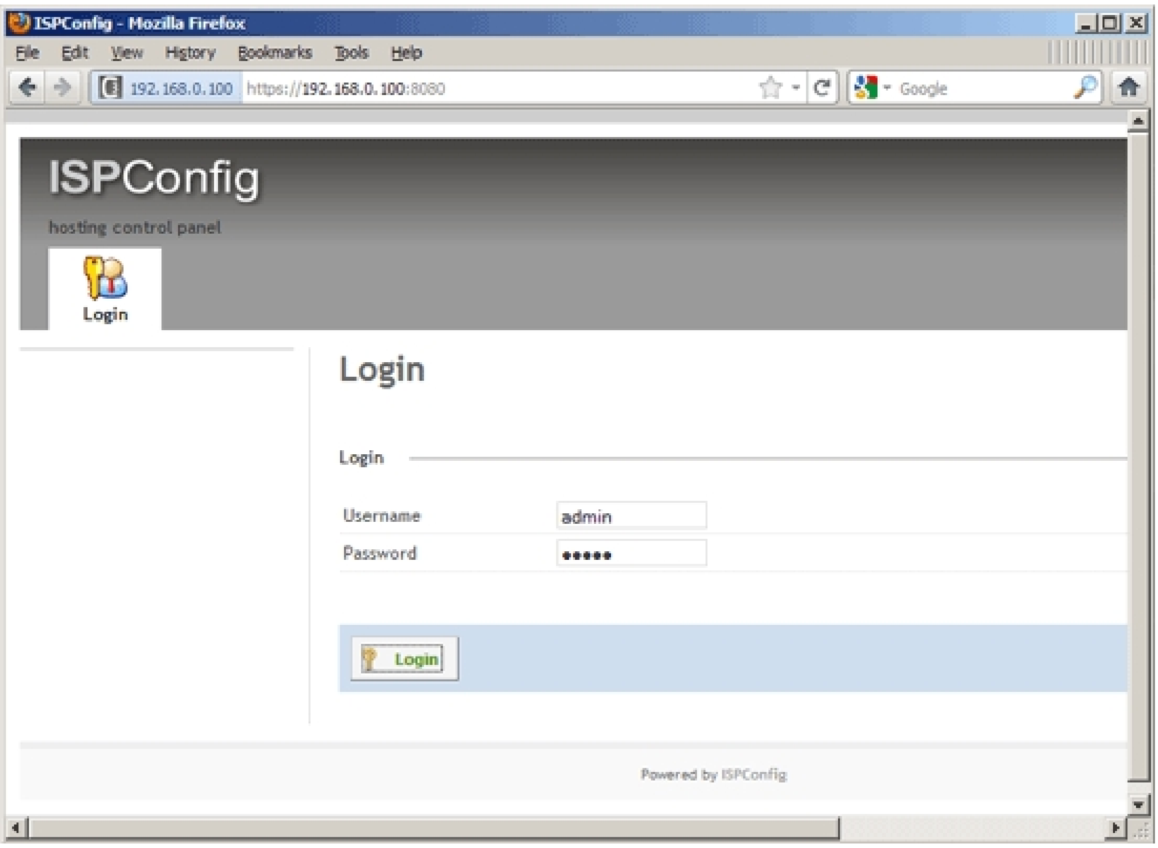
- `Login` (only visible before login)
- Home
- Sites
- Email
- Monitor
- System
- DNS
- Help
- `Domains` (usually not enabled by default)
- Client
- Tools

The order might differ for you. In the following the tabs and their submenus will be described in functional order, i.e., in the order that allows you to create client accounts, email accounts, web sites, etc.

4.2 Login

The ISPConfig 3 web interface can be accessed on port `8080`. Go to `http(s)://server1.example.com:8080` and log in with the default username and password:

- Username: `admin`
- Password: `admin`


ISPConfig - Mozilla Firefox
File Edit View History Bookmarks Tools Help
192.168.0.100 https://192.168.0.100:8080
Google
ISPConfig
hosting control panel
Login

Login

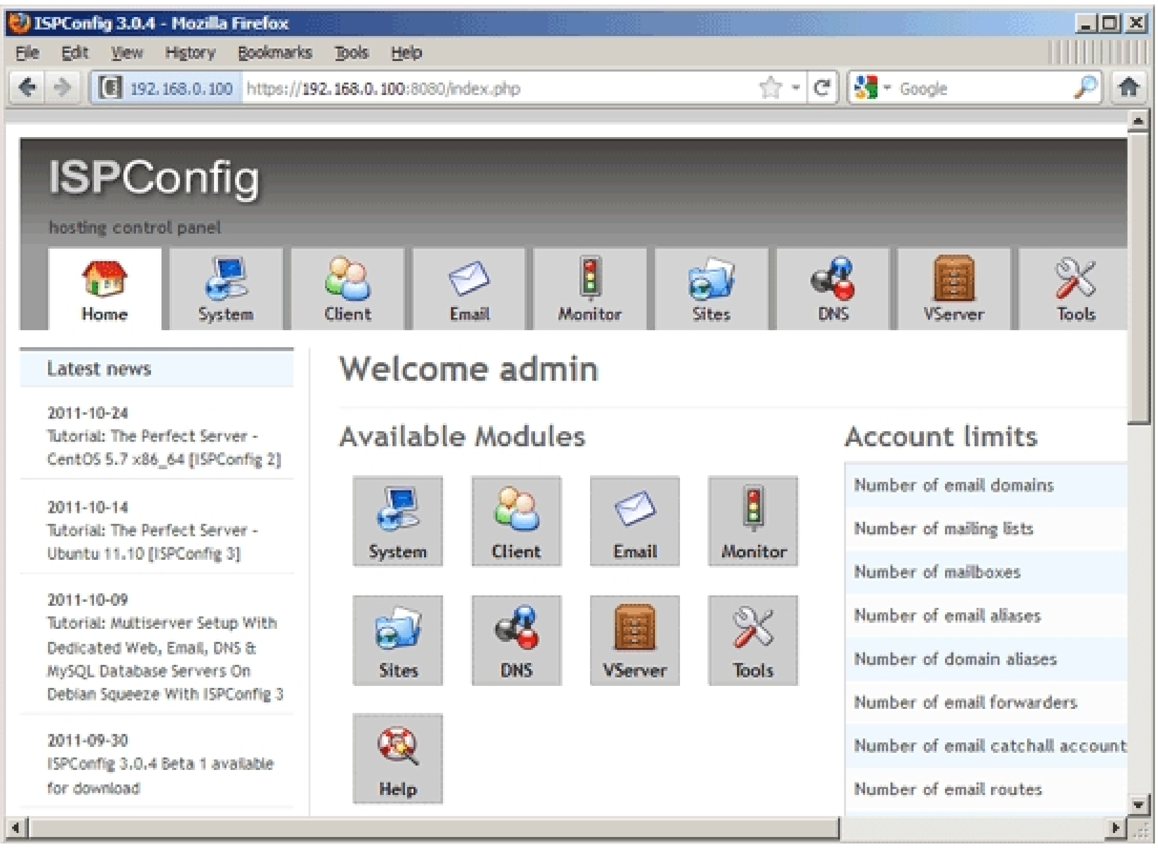
Login

Username admin

Password •••••

Login
Powered by ISPConfig

This is how the ISPConfig 3 control panel looks after your first login:



2011-10-24
Tutorial: The Perfect Server - CentOS 5.7 x86_64 [ISPConfig 2]

2011-10-14
Tutorial: The Perfect Server - Ubuntu 11.10 [ISPConfig 3]

2011-10-09
Tutorial: Multiserver Setup With Dedicated Web, Email, DNS & MySQL Database Servers On Debian Squeeze With ISPConfig 3

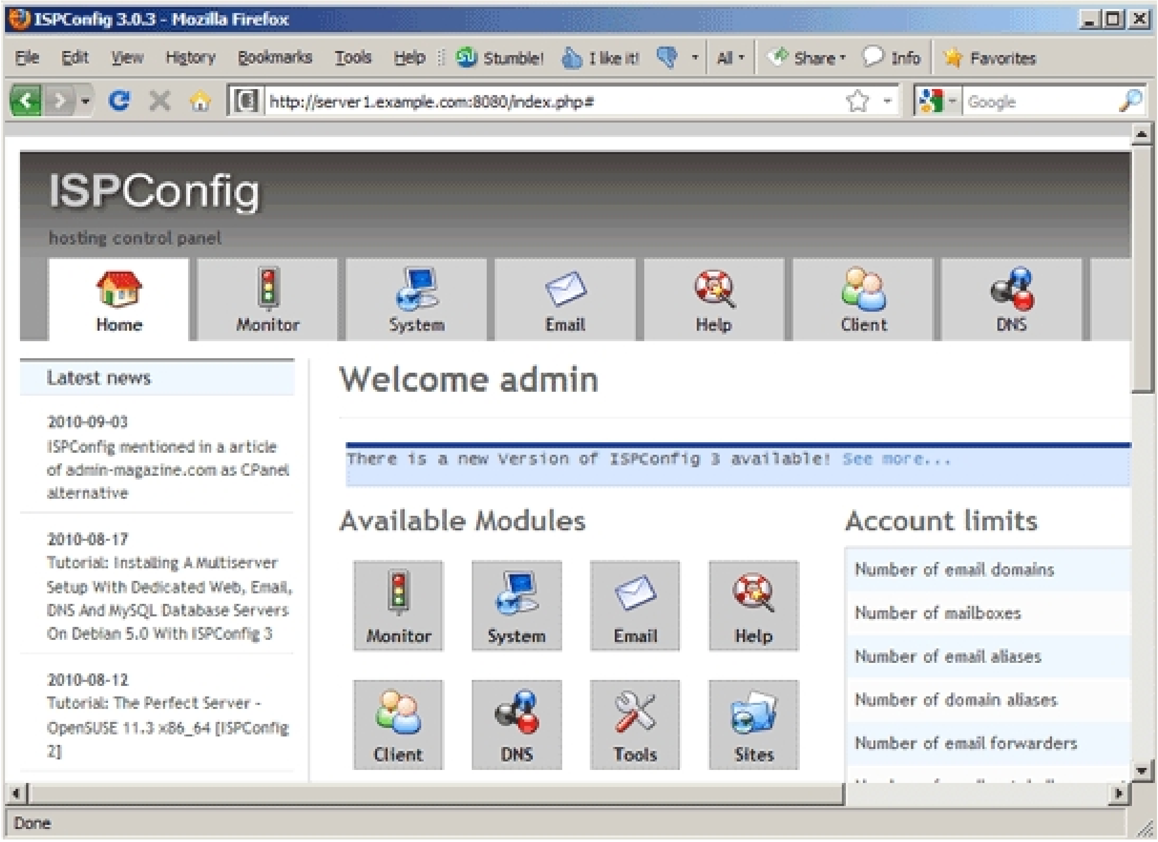
2011-09-30
ISPConfig 3.0.4 Beta 1 available for download

Number of email domains
Number of mailing lists
Number of mailboxes
Number of email aliases
Number of domain aliases
Number of email forwarders
Number of email catchall account
Number of email routes

After your first login, you should immediately change the password - to do this, go to `Tools > User Settings > Password and Language`.

4.3 Home

Under the *Home* tab, you can find the ISPConfig 3 dashboard with links to all available modules, an overview of your account limits, and the latest news about ISPConfig (new ISPConfig releases, new tutorials, etc.). If a new ISPConfig 3 version is available, this will also be shown on the dashboard so that you can upgrade your ISPConfig installation if you like.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
ISPConfig
hosting control panel
Home Monitor System Email Help Client DNS
Latest news
Available Modules
Monitor System Email Help
Client DNS Tools Sites
Account limits
Done

2010-09-03
ISPConfig mentioned in a article of admin-magazine.com as CPanel alternative

2010-08-17
Tutorial: Installing A Multiserver Setup With Dedicated Web, Email, DNS And MySQL Database Servers On Debian 5.0 With ISPConfig 3

2010-08-12
Tutorial: The Perfect Server - OpenSUSE 11.3 x86_64 [ISPConfig 2]

Welcome admin

There is a new Version of ISPConfig 3 available! See more...

Number of email domains
Number of mailboxes
Number of email aliases
Number of domain aliases
Number of email forwarders

4.4 Tools

4.4.1 User Settings

4.4.1.1 Password and Language

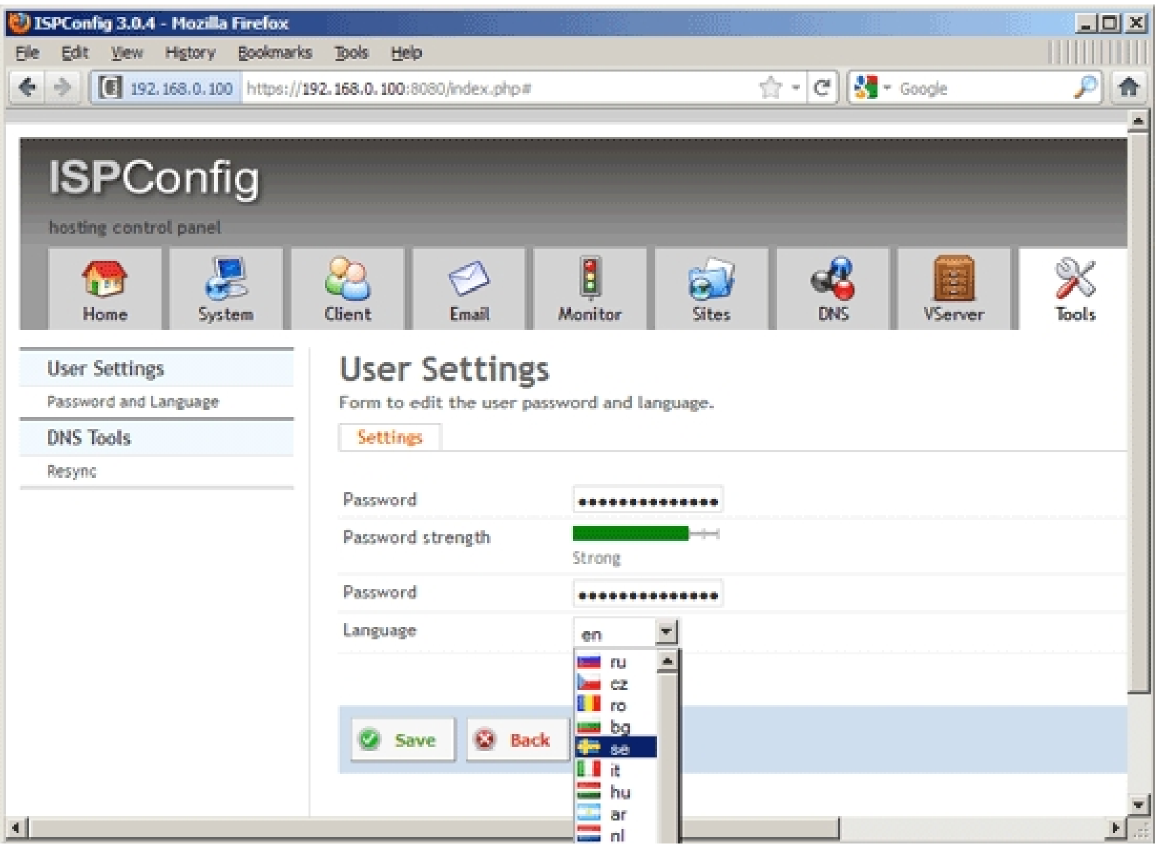
Here you can change the password and the language of the currently logged in ISPConfig user. If you log in for the first time, it is strongly recommended to immediately change the default password.

- *Password*: Type in the new password.

- *Password strength*: This field shows how strong the new password is (a strong password should include numbers, symbols, upper and lowercase letters; password length should be 8 characters or more; avoid any password based on repetition, dictionary words, letter or number sequences, usernames, relative or pet names, or biographical information).

- *Repeat Password*: Type in the new password again to make sure you made no typo.

- *Language*: Select the desired interface language of the ISPConfig control panel. If you change the language, you must log out and log back in for the changes to take effect.



User Settings

Form to edit the user password and language.

Settings

Password: ••••••••••••••

Password strength: Strong

Password: ••••••••••••••

Language: en

4.4.2 Interface

4.4.2.1 Interface

Here you can change ISPConfig's theme and the start module:

- `Design`: you can select from a list of themes here, if more than one theme is available.
- `Startmodule`: this defines the module that you will be directed to after you log into ISPConfig. By default this is ISPConfig's dashboard.

4.4.3 Sync Tools

4.4.3.1 Resync

You can use the Resync tool to make ISPConfig rewrite all configuration files (web server configuration, email configuration, etc.). This is useful, for example, if you somehow lost your configuration (becaue someone deleted it manually) or move to a new server and only have the MySQL dump of the ISPConfig database.

To do a resync, just select which configuration files you want to have rewritten (web site configuration, FTP users, shell users, cron jobs, MySQL databases, mail boxes, DNS configuration), and click on the `Start` button.

Please note that if you have a multiserver setup, this will resync configurations on all servers.

4.4.4 Import

This section contains tools to import settings from remote servers.

4.4.4.1 ISPConfig 3 mail

You can use this form to import email configuration settings from a remote ISPConfig 3 server to the local one. Please note that this imports just the settings (email username, password, etc., not the contents of the mail box)! To use this feature, you must set up a remote user on the remote ISPConfig 3 server (see ***4.9.1.2 Remote Users***).

- *`Remote API URL`*: Fill in the URL of the remote ISPConfig server. The URL has the following form: https://www.example.com:8080/remote/
- *`Remote User`*: Specify the username of the remote user you've created on the remote ISPConfig 3 server.
- *`Remote password`*: Specify the password of the remote user.

4.4.4.2 PowerDNS Tupa

This form can be used to import DNS zones and records from a PowerDNS server with the ***TUPA*** control panel into ISPConfig 3. This tool connects directly to the remote PowerDNS MySQL database, so make sure that remote connections to the PowerDNS database are allowed.

- *`Tupa database hostname`*: Fill in the hostname of the PowerDNS database server.
- *`Tupa database name`*: Specify the PowerDNS MySQL database name.
- *`Tupa database user`*: Specify the MySQL username for the PowerDNS database.
- *`Tupa database password`*: Fill in the MySQL password.

4.5 Client

4.5.1 Clients

A client is a company or individual that buys web hosting services from either you (i.e., the company or individual that runs the ISPConfig server) or from a reseller (see chapter ***4.5.2***). You should create at least one client before you go on and create web sites, email accounts, etc. because all these hosting services must have a client that they can be assigned to.

4.5.1.1 Add Client

You can create clients using this form. Clients can log into ISPConfig and manage their own web hosting services, like web sites, email accounts, etc. A client can belong either to a reseller or directly to the company/individual that runs the ISPConfig server.

The *Add Client* form is split up into two tabs, *Address* and *Limits*:

Address

This is where you type in the name, address, and login details of the client. The form has the following fields:

- *Company name* (optional): Fill in the name of the company.
- *Contact name*: Fill in the name of the person that is responsible for this ISPConfig account.
- *Customer No.* (optional): If the client has a customer number, you can specify it here.
- *Username*: Fill in the desired ISPConfig username for the client. This is the username that is used to log into ISPConfig.
- *Password*: Type in a password for the user (or use the *Generate Password* link to have ISPConfig generate one for you).
- *Password strength*: This field shows how strong the new password is (a strong password should include numbers, symbols, upper and lowercase letters; password length should be 8 characters or more; avoid any password based on repetition, dictionary words, letter or number sequences, usernames, relative or pet names, or biographical information).
- *Repeat Password*: Confirm your password.
- *Language*: Select the desired interface language of the ISPConfig control panel.
- *Theme*: Here you can select the theme of the ISPConfig control panel.
- *Street* (optional): Specify the street of the client.
- *ZIP* (optional): Fill in the client's postcode.
- *City* (optional): Fill in the client's city.

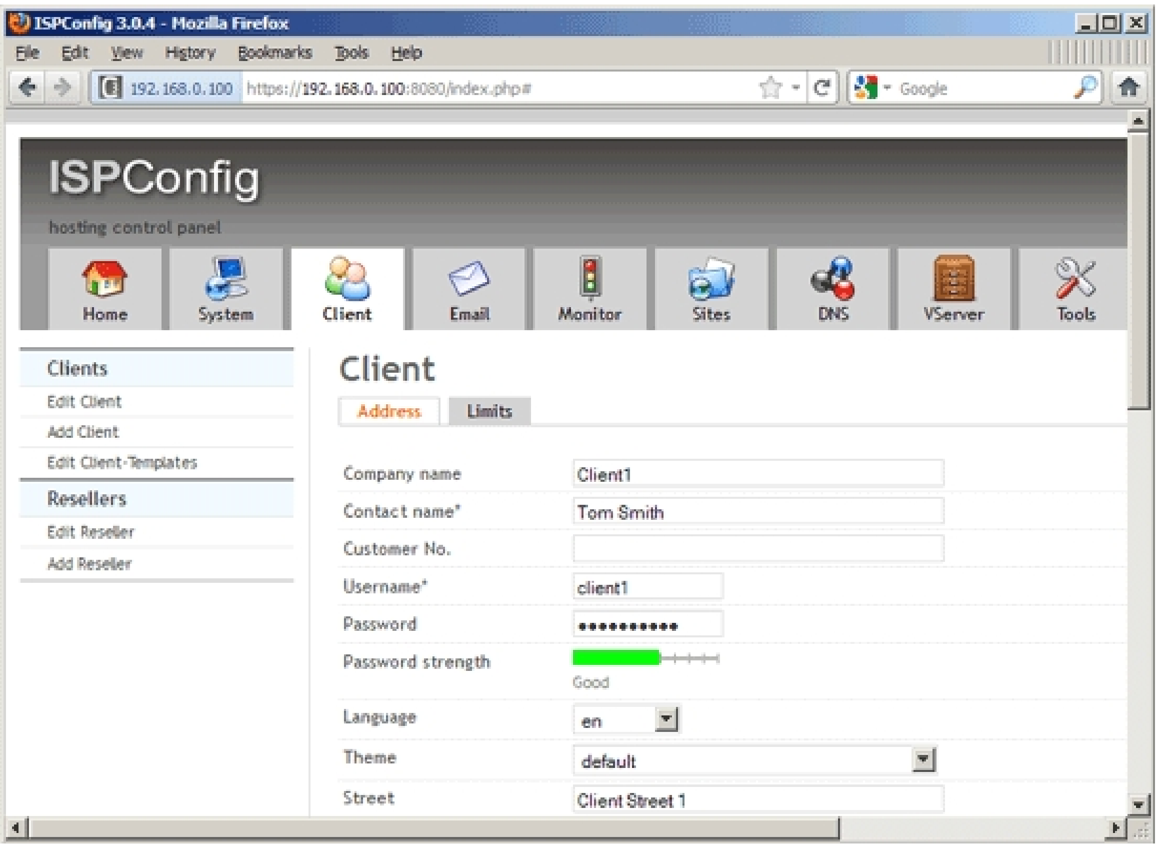
- *`State`* (optional): Specify the client's state, e.g. California, Bavaria, etc.
- *`Country`*: Select the client's country from the drop-down menu.
- *`Telephone`* (optional): Specify the client's landline number.
- *`Mobile`* (optional): Specify the client's mobile number.
- *`Fax`* (optional): Specify the client's fax number.
- *`Email`* (optional): Fill in the client's email address.
- *`Internet`* (optional): Fill in the URL of the client's web site (beginning with *`http://`* or *`https://`*).
- *`ICQ`* (optional): Specify the client's ICQ number.
- *`VAT ID`* (optional): Specify the client's VAT ID number.
- *`Company/Entrepreneur ID`* (optional): Specify the client's Company/Entrepreneur ID.

The following optional fields can be used to store payment details (bank details and PayPal email address) for this customer. These are for your internal use, they have no further use in ISPConfig (yet).

- *`Bank account owner`* (optional): Fill in the name of the owner of the bank account that is associated with this customer.
- *`Bank account no.`* (optional): Fill in the bank account number.
- *`Bank code`* (optional): Fill in the bank code.
- *`Bank name`* (optional): Specify the name of the bank.
- *`IBAN`* (optional): Fill in the International Bank Account Number (IBAN) (used for bank transfers across national borders).
- *`BIC / Swift`* (optional): Fill in the international Bank Identifier Code (BIC or Swift) (used for bank transfers across national borders).
- *`PayPal Email`* (optional): Fill in the customer's PayPal email address.

And finally one more field:

- *`Notes`* (optional): Here you can add notes and comments.



Client

Company name	Client1
Contact name*	Tom Smith
Customer No.	
Username*	client1
Password	●●●●●●●●●●●
Password strength	Good
Language	en
Theme	default
Street	Client Street 1

Limits

This is where the resources are defined that the client can use. If you select a master or addon template, click on *Save*, and the values in the rest of the form will be adjusted according to the templates. To select or de-select an addon template, it is not enough to click on Save - you must click on the Add additional template or Delete additional template button before. If you select the *Custom* template in the *Master template* field, you have to enter your limits manually.

There are two kinds of templates, main templates and additional templates. In a main template you can define a basic set of limits. An additional template differs from a main template in that the values of the addtitional template are ***added*** to the value of the main template. For example, if you define in a main template with a max. number of two web domains and an additional template with a max. number of five web domains, and you select that main template and additional template for the client/reseller, the client/reseller can have the ***sum*** of both, i.e., seven web domains.

- *Master template*: If you have defined a template for client limits that you want to apply to this client (so that you don't have to define all the client limits manually in the following fields), you can select that template here. Select *Custom* if you want to define the client limits manually.
- *Addon template*: If you have defined an additional template that you want to add to the main template, select that template here. To select or de-select an addon template, it is not enough to click on Save - you must click on the Add additional template or Delete additional template button before.

- *`Active Addons`*: Addon templates that are currently in use are listed here.
- *`Default Webserver`*: Select the default webserver for the client. The default webserver will be pre-selected for this client when web items (web sites, etc.) are created for the client, but this selection can be changed in the appropriate form.
- *`Max. number of web domains`*: Specify the max. amount of web domains that this client can create. *`-1`* means unlimited.
- *`Web Quota`*: Specify the max. hard drive space (in MB) that this client's web sites can use. *`-1`* means unlimited.
- Traffic Quota: Specify the max. monthly traffic (in MB) that this client can use. -1 means unlimited.
- *`PHP Options`*: Specify which PHP modes should be available for the client when he creates/modifies a web site. The following four modes are available: Fast-CGI, CGI, Mod-PHP, SuPHP.

    - Fast-CGI:

      Advantages:

        - Scripts will be executed with user privileges of the web site;
        - More than one PHP version can be run as FastCGI;
        - Might be better in speed compared to CGI and suPHP.

      Disadvantages:

        - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the *`Custom php.ini settings`* field on the *`Options`* tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).

    - CGI:

      Advantages:

        - Scripts will be executed with user privileges of the web site;
        - More than one PHP version can be run as CGI.

      Disadvantages:

        - CGI might use a little more memory (RAM) - therefore, it's not recommended to run PHP as CGI on slow virtual servers;
        - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the *`Custom php.ini settings`* field on the *`Options`* tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).

    - Mod-PHP:

Advantages:

- Speed;
- Needs less memory (RAM) than CGI;
- php.ini values can be changed via PHP scripts, vhost files, .htaccess files.

Disadvantages:

- Scripts are being executed with Apache privileges, which might lead to some security related problems;
- Only one version of PHP can be installed as Apache module;
- You ***cannot*** use the *`Custom php.ini settings`* field on the *`Options`* tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).

- SuPHP:

  Advantages:

  - Scripts will be executed with user privileges of the web site;
  - Each vhost can have its own php.ini file;
  - Needs less memory (RAM) than CGI;
  - More than one PHP version can be run as suPHP.

  Disadvantages:

  - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the *`Custom php.ini settings`* field on the *`Options`* tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***);
  - SuPHP might be a little slower than mod_php.

- PHP-FPM:

  Advantages:

  - Scripts will be executed with user privileges of the web site;
  - More than one PHP version can be run as PHP-FPM;
  - Adaptive process spawning;
  - Advanced process management with graceful stop/start;
  - Emergency restart in case of accidental opcode cache destruction;
  - Might be better in speed compared to CGI and suPHP.

Disadvantages:

- php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the `Custom php.ini settings` field on the `Options` tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).

- ***Recommendations:***

  Apache:

  - High-Traffic Web Sites: Fast-CGI + suExec or PHP-FPM + suExec
  - Low-Traffic Web Sites: CGI + suExec or SuPHP

  nginx:

  - PHP-FPM

- *`CGI available`*: This enables the CGI checkbox in the web form so that the client can select this feature for his web sites.
- *`SSI available`*: This enables the SSI checkbox in the web form so that the client can select this feature for his web sites.
- *`Perl available`*: This enables the Perl checkbox in the web form so that the client can select this feature for his web sites. This requires Apache's mod-perl and is therefore ***available only on Apache servers***.
- *`Ruby available`*: This enables the Ruby checkbox in the web form so that the client can select this feature for his web sites. This requires Apache's mod-ruby and is therefore ***available only on Apache servers***.
- *`Python available`*: This enables the Python checkbox in the web form so that the client can select this feature for his web sites. This requires Apache's mod-python and is therefore ***available only on Apache servers***.
- *`SuEXEC forced`*: If you select this feature, the client's PHP processes will be forced to use SuExec. This is useful especially for FastCGI, CGI, and PHP-FPM. SuExec is ***available on Apache servers only***.
- *`Custom error docs available`*: This enables the *`Custom error docs`* checkbox in the web form so that the client can select this feature for his web sites.
- *`Wildcard subdomain available`*: If you select this feature, the client will be able to select * (in addition to "none" and "www") as the auto-subdomain for his web sites.
- *`SSL available`*: This enables the SSL checkbox in the web form so that the client can select this feature for his web sites and create SSL certificates.
- *`Max. number of web aliasdomains`*: Specify the max. amount of web aliasdomains that this client can create. *`-1`* means unlimited.

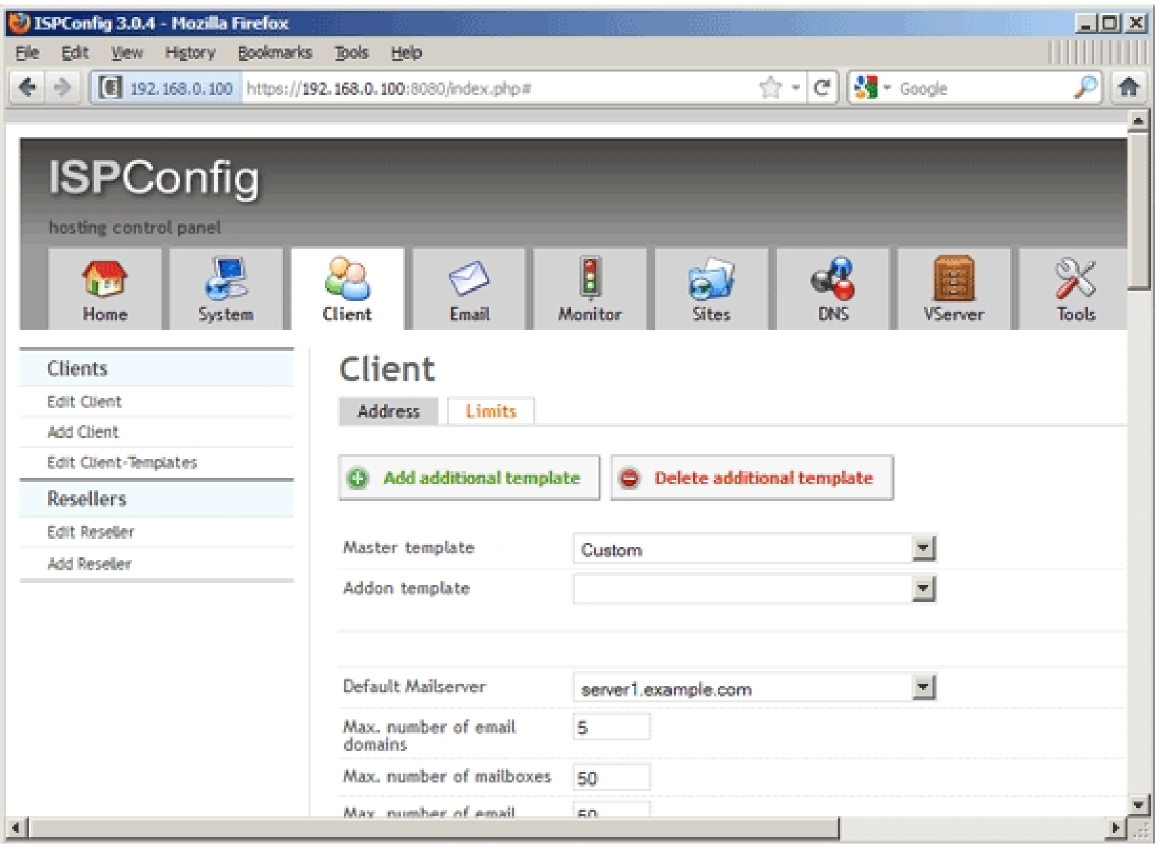
- *Max. number of web subdomains*: Specify the max. amount of web subdomains that this client can create. *-1* means unlimited.
- *Max. number of FTP users*: Specify the max. amount of FTP users that this client can create. *-1* means unlimited.
- *Max. number of Shell users*: Specify the max. amount of shell users that this client can create. *-1* means unlimited.
- *SSH-Chroot Options*: Specify which SSH modes should be available for the client when he creates/modifies a shell account. The *None* mode means that the shell user can browse the whole file system and is limited only by file/directory permissions - this can be a security risk. The *Jailkit* mode means that the shell user will be limited to his home directory (chrooted) and can only browse directories inside his home directory.
- *Max. number of Webdav users*: Specify the max. amount of WebDAV users that this client can create. *-1* means unlimited.
- *Default Mailserver*: Select the default mailserver for the client. The default mailserver will be pre-selected for this client when email items (email accounts, etc.) are created for the client, but this selection can be changed in the appropriate form.
- *Max. number of email domains*: Specify the max. amount of email domains that this client can create. *-1* means unlimited.
- *Max. number of mailboxes*: Specify the max. amount of mailboxes that this client can create. *-1* means unlimited.
- *Max. number of email aliases*: Specify the max. amount of email aliases that this client can create. *-1* means unlimited.
- *Max. number of domain aliases*: Specify the max. amount of domain aliases that this client can create. *-1* means unlimited.
- *Max. number of mailing lists*: Specify the max. amount of mailing lists that this client can create. *-1* means unlimited.
- *Max. number of email forwarders*: Specify the max. amount of email forwarders that this client can create. *-1* means unlimited.
- *Max. number of email catchall accounts*: Specify the max. amount of email catchall accounts that this client can create. *-1* means unlimited.
- *Max. number of email routes*: Specify the max. amount of email routes that this client can create. *-1* means unlimited.
- *Max. number of email filters*: Specify the max. amount of email filters that this client can create. *-1* means unlimited.
- *Max. number of fetchmail accounts*: Specify the max. amount of fetchmail accounts that this client can create. *-1* means unlimited.
- *Mailbox quota*: Specify the max. hard drive space (in MB) that this client's email accounts can use. *-1* means unlimited.

- *Max. number of spamfilter white / blacklist filters*: Specify the max. amount of whitelist and blacklist filters for the spamfilter that this client can create. *-1* means unlimited.
- *Max. number of spamfilter users*: Specify the max. amount of spamfilter users that this client can create. *-1* means unlimited.
- *Max. number of spamfilter policies*: Specify the max. amount of spamfilter policies that this client can create. *-1* means unlimited.
- *Default Database Server*: Select the default database server for the client. The default database server will be pre-selected for this client when a database is created for the client, but this selection can be changed in the appropriate form (if you are logged in as admin).
- *Max. number of Databases*: Specify the max. amount of databases that this client can create. *-1* means unlimited.
- *Max. number of cron jobs*: Specify the max. amount of cron jobs that this client can create. *-1* means unlimited.
- *Max. Allowed Cronjob types (chrooted and full implies url)*: Specify which kind of cron jobs should be available for the client when he creates/modifies a cron job.
    - *Full Cron*: *Full Cron* means that you can use any command for the cron job, and it will ***not*** run in a chroot environment.
    - *Chrooted Cron*: If *Chrooted Cron* is selected in the limits of the client that owns the cron job, the cron jobs are chrooted (using Jailkit).
    - *URL Cron*: This means that the client can only create wget cron jobs, i.e., he specifies a URL in the cron job command line, and that URL will be accessed via wget.

- *Min. delay between executions*: This specifies the minimal delay (in minutes) how often a cron job can be executed. If you specify *5* here, for example, a cron job cannot be run every minute, but only every five minutes.
- *Default DNS Server*: Select the default DNS server for the client. The default DNS server will be pre-selected for this client when DNS items (zones, etc.) are created for the client, but this selection can be changed in the appropriate form (if you are logged in as admin).
- *Max. number of DNS zones*: Specify the max. amount of DNS zones that this client can create. *-1* means unlimited.
- *Max. number of secondary DNS zones*: Specify the max. amount of secondary DNS zones that this client can create. *-1* means unlimited.
- *Max. number DNS records*: Specify the max. amount of DNS records that this client can create. *-1* means unlimited.
- *Max. number of virtual servers*: Specify the max. amount of virtual servers that this client can create. *-1* means unlimited.

- `Force virtual server template`: If an OpenVZ template is selected here, the client can use only this template to create virtual machines. If no template is selected, the client can choose from all available OpenVZ templates.

- `Max. number of APS instances`: Specify the max. amount of APS packages that this client can install with the APS Installer. `-1` means unlimited.



Master template: Custom

Addon template:

Default Mailserver: server1.example.com

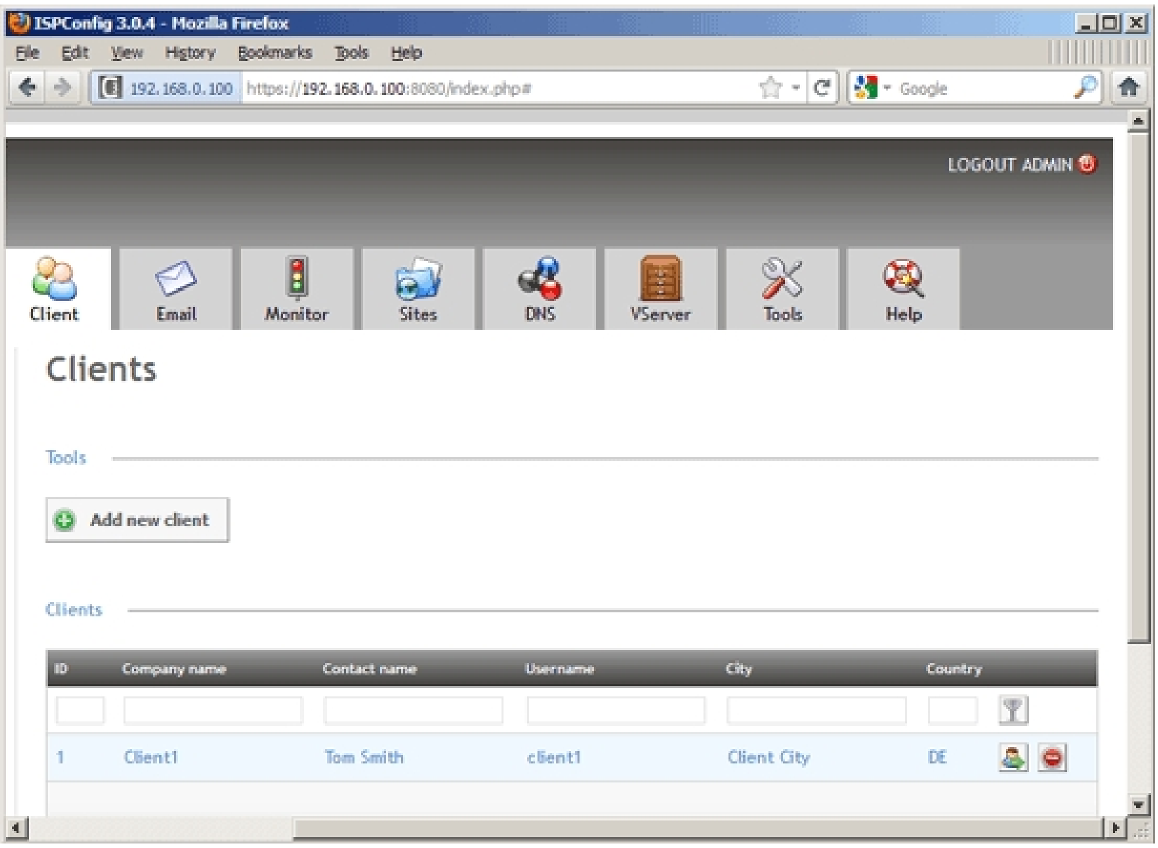
Max. number of email domains: 5

Max. number of mailboxes: 50

Max. number of email: 50

4.5.1.2 Edit Client

Under `Edit Client` you can find a list of existing clients:


ISPConfig 3.0.4 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
192.168.0.100 https://192.168.0.100:8080/index.php#
Google
LOGOUT ADMIN
Client
Email
Monitor
Sites
DNS
VServer
Tools
Help
Tools
Add new client
Clients

Clients

ID	Company name	Contact name	Username	City	Country
1	Client1	Tom Smith	client1	Client City	DE

By clicking any of them, you will get to the `Address` and `Limits` tabs of that client (that you already know from chapter 4.5.1.1) where you can modify the settings of that client.

Above the list you can find filters that allow you to search for specific parameters in all clients. The following filters are available:

- ID
- Company name
- Contact name
- Username
- City
- Country

Click the

button to start a search.

From the client list, it is also possible to directly log in as a client - just click the

button next to the client.

To delete a client, click the ⛔ button. A confirmation message will pop up, asking you if you really want to delete the record.

4.5.1.3 Edit Client-Templates

You can edit and create client templates here. A template is a pre-defined set of limits that can be assigned to a client. Let's assume you sell five different hosting plans to your clients - instead of defining limits manually whenever you create a new client, you could create five templates (one for each hosting plan) and use such a template when you create a new client. That way, creating clients is less error-prone and time-consuming.

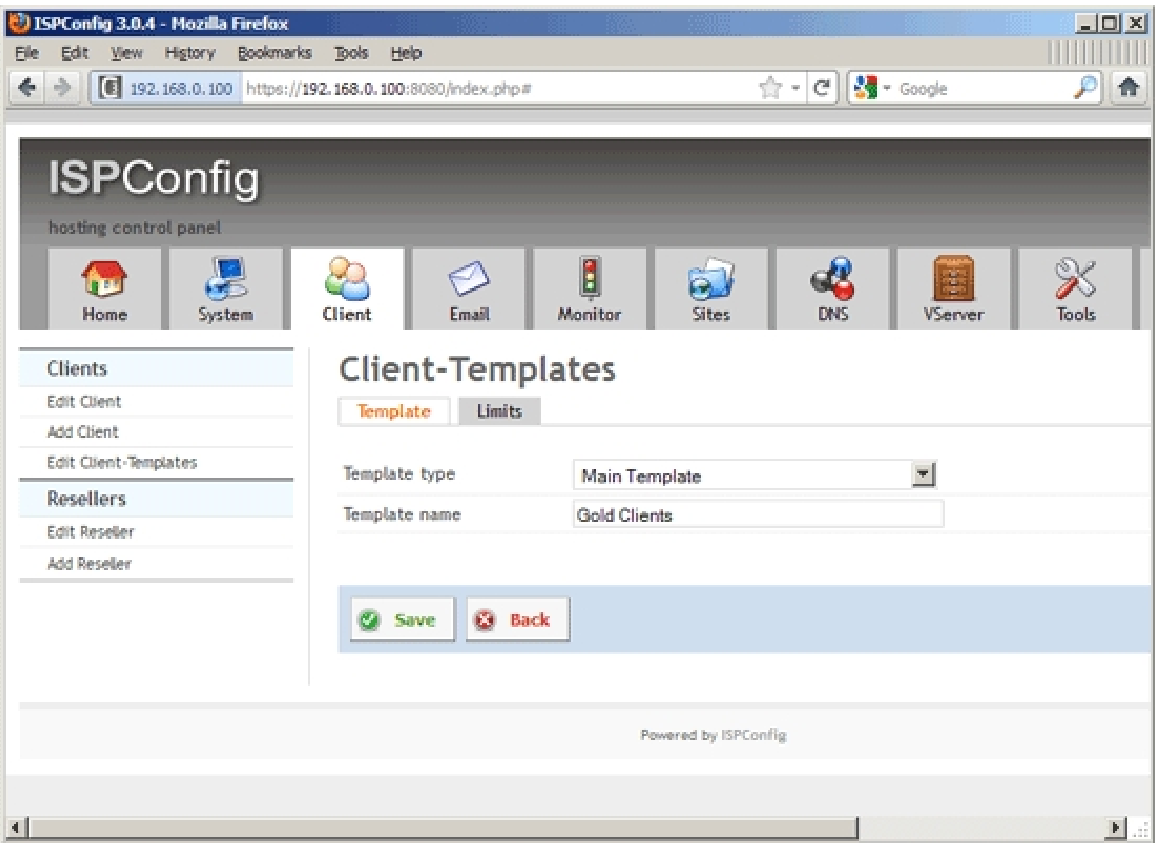
There are two kinds of templates, main templates and additional templates. In a main template you can define a basic set of limits. An additional template differs from a main template in that the values of the addtitional template are ***added*** to the value of the main template. For example, if you define in a main template with a max. number of two web domains and an additional template with a max. number of five web domains, and you select that main template and additional template for the client/reseller, the client/reseller can have the ***sum*** of both, i.e., seven web domains.

Creating A Template

Click the *Add new record* button in the *Tools* section. You will get to the *Client-Templates* form that consists out of two tabs, *Template* and *Limits*.

Template

Here you can enter a name for the template and select if it's a *Main Template* or an *Additional Template*.



Template type: Main Template

Template name: Gold Clients

Limits

You can define the following limits for your template:

- *`Max. number of web domains`*: Specify the max. amount of web domains that this client can create. *`-1`* means unlimited.
- *`Web Quota`*: Specify the max. hard drive space (in MB). *`-1`* means unlimited.
- *`Traffic Quota`*: Specify the max. monthly traffic (in MB). *`-1`* means unlimited.
- *`PHP Options`*: Specify which PHP modes should be available for the client when he creates/modifies a web site. The following four modes are available: Fast-CGI, CGI, Mod-PHP, SuPHP.
  - Fast-CGI:

    Advantages:

    - Scripts will be executed with user privileges of the web site;
    - More than one PHP version can be run as FastCGI;
    - Might be better in speed compared to CGI and suPHP.

    Disadvantages:

    - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the `Custom php.ini settings` field on the `Options` tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).

- CGI:

  Advantages:

    - Scripts will be executed with user privileges of the web site;
    - More than one PHP version can be run as CGI.

  Disadvantages:

    - CGI might use a little more memory (RAM) - therefore, it's not recommended to run PHP as CGI on slow virtual servers;
    - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the `Custom php.ini settings` field on the `Options` tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).

- Mod-PHP:

  Advantages:

    - Speed;
    - Needs less memory (RAM) than CGI;
    - php.ini values can be changed via PHP scripts, vhost files, .htaccess files.

  Disadvantages:

    - Scripts are being executed with Apache privileges, which might lead to some security related problems;
    - Only one version of PHP can be installed as Apache module;
    - You ***cannot*** use the `Custom php.ini settings` field on the `Options` tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).

- SuPHP:

  Advantages:

    - Scripts will be executed with user privileges of the web site;
    - Each vhost can have its own php.ini file;
    - Needs less memory (RAM) than CGI;

        - More than one PHP version can be run as suPHP.

        Disadvantages:

        - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the *`Custom php.ini settings`* field on the *`Options`* tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***);
        - SuPHP might be a little slower than mod_php.

    - PHP-FPM:
      Advantages:

        - Scripts will be executed with user privileges of the web site;
        - More than one PHP version can be run as PHP-FPM;
        - Adaptive process spawning;
        - Advanced process management with graceful stop/start;
        - Emergency restart in case of accidental opcode cache destruction;
        - Might be better in speed compared to CGI and suPHP.

        Disadvantages:

        - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the *`Custom php.ini settings`* field on the *`Options`* tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).

    - ***Recommendations:***
      Apache:

        - High-Traffic Web Sites: Fast-CGI + suExec or PHP-FPM + suExec
        - Low-Traffic Web Sites: CGI + suExec or SuPHP

        nginx:

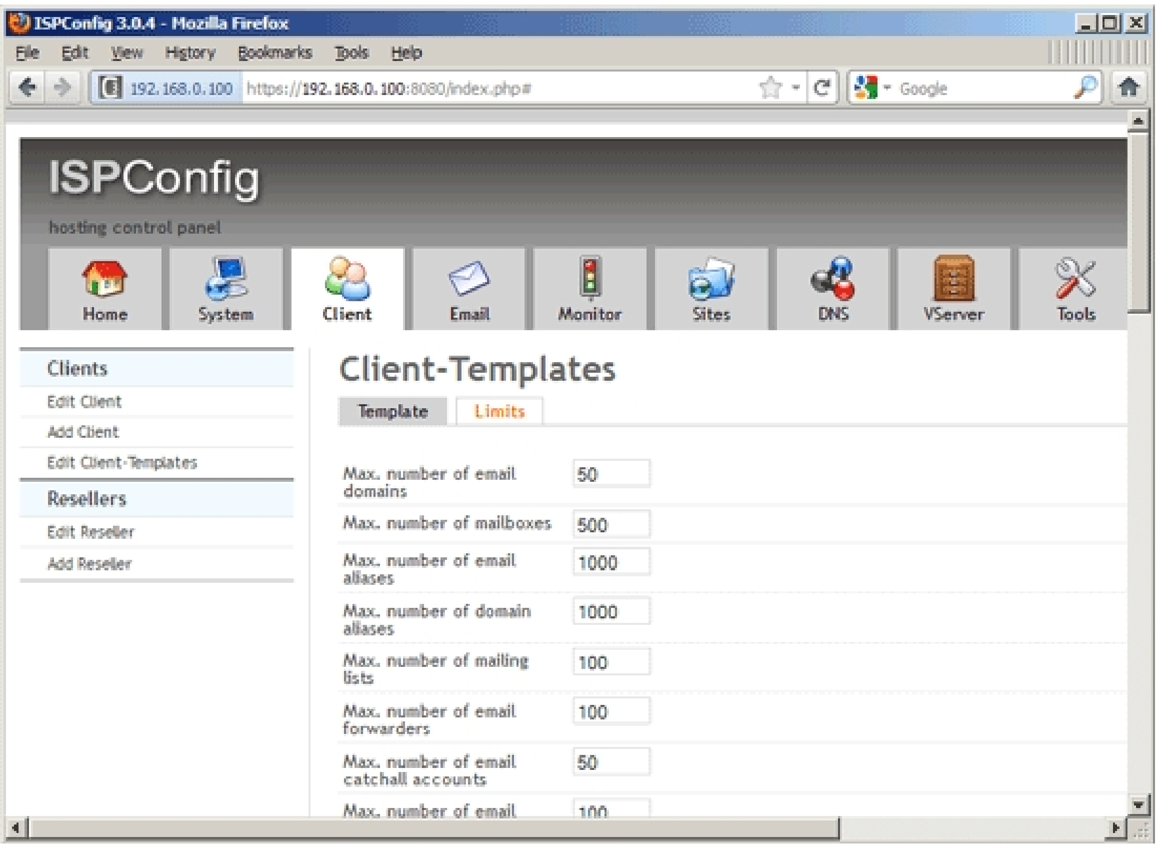
        - PHP-FPM

- *`CGI available`*: This enables the CGI checkbox in the web form so that the client can select this feature for his web sites.

- `SSI available`: This enables the SSI checkbox in the web form so that the client can select this feature for his web sites.
- `Perl available`: This enables the Perl checkbox in the web form so that the client can select this feature for his web sites. This requires Apache's mod-perl and is therefore ***available only on Apache servers***.
- `Ruby available`: This enables the Ruby checkbox in the web form so that the client can select this feature for his web sites. This requires Apache's mod-ruby and is therefore ***available only on Apache servers***.
- `Python available`: This enables the Python checkbox in the web form so that the client can select this feature for his web sites. This requires Apache's mod-python and is therefore ***available only on Apache servers***.
- `SuEXEC forced`: If you select this feature, the client's PHP processes will be forced to use SuExec. This is useful especially for FastCGI, CGI, and PHP-FPM. SuExec is ***available on Apache servers only***.
- `Custom error docs available`: This enables the `Custom error docs` checkbox in the web form so that the client can select this feature for his web sites.
- `Wildcard subdomain available`: If you select this feature, the client will be able to select * (in addition to "none" and "www") as the auto-subdomain for his web sites.
- `SSL available`: This enables the SSL checkbox in the web form so that the client can select this feature for his web sites and create SSL certificates.
- `Max. number of web aliasdomains`: Specify the max. amount of web aliasdomains. `-1` means unlimited.
- `Max. number of web subdomains`: Specify the max. amount of web subdomains. `-1` means unlimited.
- `Max. number of FTP users`: Specify the max. amount of FTP users. `-1` means unlimited.
- `Max. number of Shell users`: Specify the max. amount of shell users. `-1` means unlimited.
- `SSH-Chroot Options`: Specify which SSH modes should be available for the client when he creates/modifies a shell account. The `None` mode means that the shell user can browse the whole file system and is limited only by file/directory permissions - this can be a security risk. The `Jailkit` mode means that the shell user will be limited to his home directory (chrooted) and can only browse directories inside his home directory.
- `Max. number of Webdav users`: Specify the max. amount of WebDAV users. `-1` means unlimited.
- `Max. number of email domains`: Specify the max. amount of email domains. `-1` means unlimited.
- `Max. number of mailboxes`: Specify the max. amount of mailboxes. `-1` means unlimited.
- `Max. number of email aliases`: Specify the max. amount of email aliases. `-1` means unlimited.
- `Max. number of domain aliases`: Specify the max. amount of domain aliases. `-1` means unlimited.
- `Max. number of mailing lists` Specify the max. amount of mailing lists. `-1` means unlimited.
- `Max. number of email forwarders`: Specify the max. amount of email forwarders. `-1` means unlimited.
- `Max. number of email catchall accounts`: Specify the max. amount of email catchall accounts. `-1` means unlimited.
- `Max. number of email routes`: Specify the max. amount of email routes. `-1` means unlimited.

- `Max. number of email filters`: Specify the max. amount of email filters. `-1` means unlimited.
- `Max. number of fetchmail accounts`: Specify the max. amount of fetchmail accounts. `-1` means unlimited.
- `Mailbox quota`: Specify the max. hard drive space (in MB). `-1` means unlimited.
- `Max. number of spamfilter white / blacklist filters`: Specify the max. amount of whitelist and blacklist filters for the spamfilter. `-1` means unlimited.
- `Max. number of spamfilter users`: Specify the max. amount of spamfilter users. `-1` means unlimited.
- `Max. number of spamfilter policies`: Specify the max. amount of spamfilter policies. `-1` means unlimited.
- `Max. number of Databases`: Specify the max. amount of databases. `-1` means unlimited.
- `Max. number of cron jobs`: Specify the max. amount of cron jobs. `-1` means unlimited.
- `Max. type of cron jobs (chrooted and full implies url)`: Specify which kind of cron jobs should be available for the client when he creates/modifies a cron job.
  - `Full Cron`: `Full Cron` means that you can use any command for the cron job, and it will ***not*** run in a chroot environment.
  - `Chrooted Cron`: If `Chrooted Cron` is selected in the limits of the client that owns the cron job, the cron jobs are chrooted (using Jailkit).
  - `URL Cron`: This means that the client can only create wget cron jobs, i.e., he specifies a URL in the cron job command line, and that URL will be accessed via wget.

- `Min. delay between executions`: This specifies the minimal delay (in minutes) how often a cron job can be executed. If you specify `5` here, for example, a cron job cannot be run every minute, but only every five minutes.
- `Max. number of DNS zones`: Specify the max. amount of DNS zones. `-1` means unlimited.
- `Max. number of secondary DNS zones`: Specify the max. amount of secondary DNS zones. `-1` means unlimited.
- `Max. number DNS records`: Specify the max. amount of DNS records. `-1` means unlimited.
- `Max. number of virtual servers`: Specify the max. amount of virtual servers. `-1` means unlimited.
- `Force virtual server template`: If an OpenVZ template is selected here, the client can use only this template to create virtual machines. If no template is selected, the client can choose from all available OpenVZ templates.
- `Max. number of APS instances`: Specify the max. amount of APS packages that can be installed with the APS Installer. `-1` means unlimited.



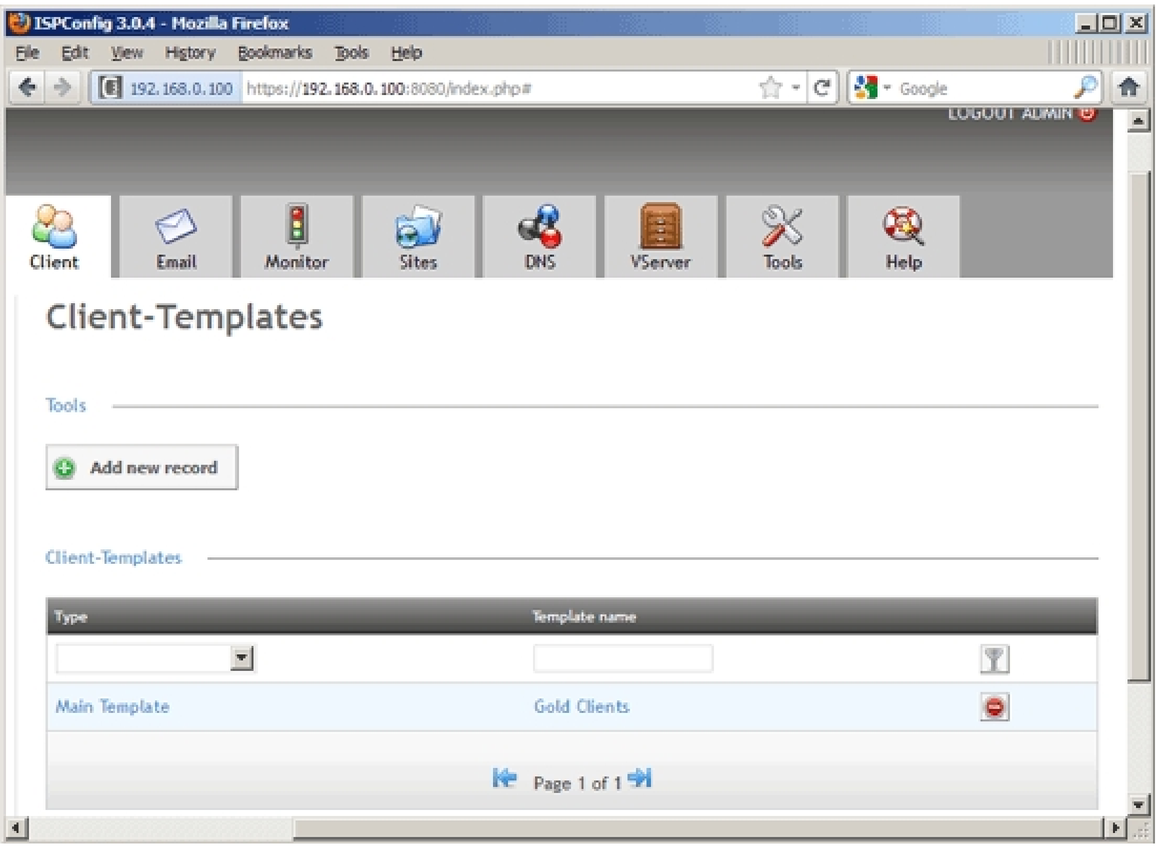
Client-Templates

Template | Limits

Max. number of email domains	50
Max. number of mailboxes	500
Max. number of email aliases	1000
Max. number of domain aliases	1000
Max. number of mailing lists	100
Max. number of email forwarders	100
Max. number of email catchall accounts	50
Max. number of email	100

Editing A Template

In the `Client Templates` section you can find a list of existing templates:


ISPConfig 3.0.4 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
192.168.0.100 https://192.168.0.100:8080/index.php#
Google
LOGOUT ADMIN
Client
Email
Monitor
Sites
DNS
VServer
Tools
Help

Client-Templates

Tools

Add new record

Client-Templates

Type	Template name
Main Template	Gold Clients

Page 1 of 1

By clicking any of them, you will get to the `Template` and `Limits` tabs of that template (that you already know from the "Creating A Template" chapter) where you can modify the settings of that template.

Above the list you can find filters that allow you to search for specific parameters in all templates. The following filters are available:

- Type
- Template name

Click the ⛛ button to start a search.

To delete a template, click the ⊖ button. A confirmation message will pop up, asking you if you really want to delete the record.

4.5.2 Resellers

ISPConfig allows you to create resellers. A reseller is a company or individual that purchases bulk hosting from a supplier (i.e., from the company or the individual that runs the ISPConfig server) with the intention of reselling it to a number of consumers (clients) at a profit.

4.5.2.1 Add Reseller

Here you can add resellers (e.g. hosting companies) that can have clients and sell hosting services to these clients. These resellers can log into ISPConfig 3 and manage clients, clients' web sites, email accounts etc.

The *Add Reseller* form is split up into two tabs, *Address* and *Limits*:

Address

This is where you type in the name, address, and login details of the reseller:

- *Company name* (optional): Fill in the name of the company.
- *Contact name*: Fill in the name of the person that is responsible for this ISPConfig account.
- *Customer No.* (optional): If the reseller has a customer number, you can specify it here.
- *Username*: Fill in the desired ISPConfig username for the reseller. This is the username that is used to log into ISPConfig.
- *Password*: Type in a password for the user (or use the *Generate Password* link to have ISPConfig generate one for you).
- *Password strength*: This field shows how strong the new password is (a strong password should include numbers, symbols, upper and lowercase letters; password length should be 8 characters or more; avoid any password based on repetition, dictionary words, letter or number sequences, usernames, relative or pet names, or biographical information).
- *Repeat Password*: Confirm your password.
- *Language*: Select the desired interface language of the ISPConfig control panel.
- *Theme*: Here you can select the theme of the ISPConfig control panel.
- *Street* (optional): Specify the street of the reseller.
- *ZIP* (optional): Fill in the reseller's postcode.
- *City* (optional): Fill in the reseller's city.
- *State* (optional): Specify the reseller's state, e.g. California, Bavaria, etc.
- *Country*: Select the reseller's country from the drop-down menu.
- *Telephone* (optional): Specify the reseller's landline number.
- *Mobile* (optional): Specify the reseller's mobile number.
- *Fax* (optional): Specify the reseller's fax number.

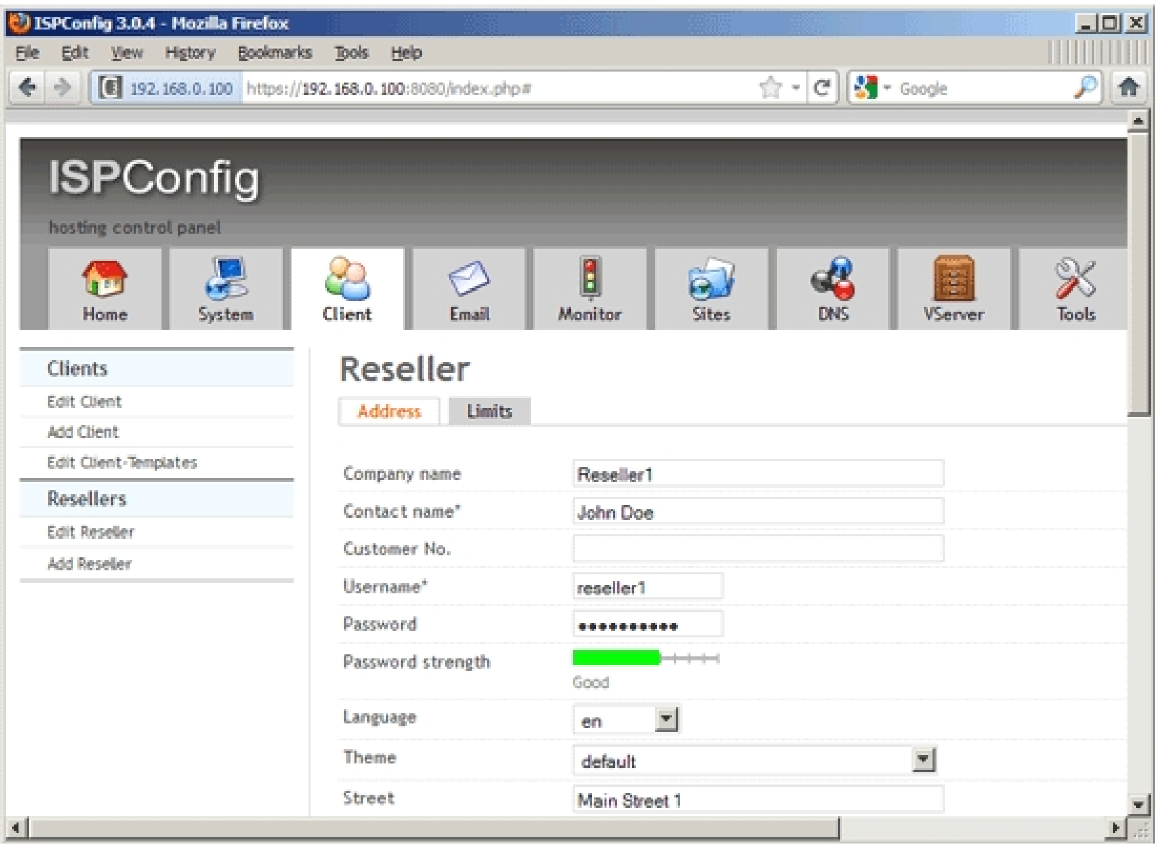
- *`Email`* (optional): Fill in the reseller's email address.
- *`Internet`* (optional): Fill in the URL of the reseller's web site (beginning with *`http://`* or *`https://`*).
- *`ICQ`* (optional): Specify the reseller's ICQ number.
- *`VAT ID`* (optional): Specify the reseller's VAT ID number.
- *`Company/Entrepreneur ID`* (optional): Specify the client's Company/Entrepreneur ID.

The following optional fields can be used to store payment details (bank details and PayPal email address) for this reseller. These are for your internal use, they have no further use in ISPConfig (yet).

- *`Bank account owner`* (optional): Fill in the name of the owner of the bank account that is associated with this reseller.
- *`Bank account no.`* (optional): Fill in the bank account number.
- *`Bank code`* (optional): Fill in the bank code.
- *`Bank name`* (optional): Specify the name of the bank.
- *`IBAN`* (optional): Fill in the International Bank Account Number (IBAN) (used for bank transfers across national borders).
- *`BIC / Swift`* (optional): Fill in the international Bank Identifier Code (BIC or Swift) (used for bank transfers across national borders).
- *`PayPal Email`* (optional): Fill in the reseller's PayPal email address.

And finally one more field:

- *`Notes`* (optional): Here you can add notes and comments.



Reseller

Company name	Reseller1
Contact name*	John Doe
Customer No.	
Username*	reseller1
Password	••••••••••
Password strength	Good
Language	en
Theme	default
Street	Main Street 1

Limits

This is where the resources are defined that the reseller can pass on to his clients. These limits define the ***total*** amount of resources available to the reseller - the reseller must split these resources up between his clients. If you select a master or addon template, click on *Save*, and the values in the rest of the form will be adjusted according to the templates. To select or de-select an addon template, it is not enough to click on Save - you must click on the Add additional template or Delete additional template button before. If you select the *Custom* template in the *Master template* field, you have to enter your limits manually.

There are two kinds of templates, main templates and additional templates. In a main template you can define a basic set of limits. An additional template differs from a main template in that the values of the addtitional template are ***added*** to the value of the main template. For example, if you define in a main template with a max. number of two web domains and an additional template with a max. number of five web domains, and you select that main template and additional template for the client/reseller, the client/reseller can have the ***sum*** of both, i.e., seven web domains.

- *Max. number of Clients*: Specify the max. amount of clients that this reseller can create. *-1* means unlimited.
- *Default Webserver*: Select the default webserver for the reseller. The default webserver will be pre-selected for this reseller when web items (web sites, etc.) are created for the reseller, but this selection can be changed in the appropriate form.

- `Max. number of web domains`: Specify the max. amount of web domains that this reseller can create. `-1` means unlimited.
- `Web Quota`: Specify the max. hard drive space (in MB) that this reseller's web sites can use. `-1` means unlimited.
- `Traffic Quota`: Specify the max. monthly traffic (in MB) that this reseller can use. `-1` means unlimited.
- `PHP Options`: Specify which PHP modes should be available for the reseller when he creates/modifies a web site. The following four modes are available: Fast-CGI, CGI, Mod-PHP, SuPHP.
  - Fast-CGI:

    Advantages:

    - Scripts will be executed with user privileges of the web site;
    - More than one PHP version can be run as FastCGI;
    - Might be better in speed compared to CGI and suPHP.

    Disadvantages:

    - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the `Custom php.ini settings` field on the `Options` tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).
  - CGI:

    Advantages:

    - Scripts will be executed with user privileges of the web site;
    - More than one PHP version can be run as CGI.

    Disadvantages:

    - CGI might use a little more memory (RAM) - therefore, it's not recommended to run PHP as CGI on slow virtual servers;
    - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the `Custom php.ini settings` field on the `Options` tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).
  - Mod-PHP:

    Advantages:

    - Speed;
    - Needs less memory (RAM) than CGI;

- php.ini values can be changed via PHP scripts, vhost files, .htaccess files.

Disadvantages:

- Scripts are being executed with Apache privileges, which might lead to some security related problems;
- Only one version of PHP can be installed as Apache module;
- You ***cannot*** use the *`Custom php.ini settings`* field on the *`Options`* tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).

- SuPHP:

  Advantages:

  - Scripts will be executed with user privileges of the web site;
  - Each vhost can have its own php.ini file;
  - Needs less memory (RAM) than CGI;
  - More than one PHP version can be run as suPHP.

  Disadvantages:

  - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the *`Custom php.ini settings`* field on the *`Options`* tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***);
  - SuPHP might be a little slower than mod_php.

- PHP-FPM:

  Advantages:

  - Scripts will be executed with user privileges of the web site;
  - More than one PHP version can be run as PHP-FPM;
  - Adaptive process spawning;
  - Advanced process management with graceful stop/start;
  - Emergency restart in case of accidental opcode cache destruction;
  - Might be better in speed compared to CGI and suPHP.

  Disadvantages:

  - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the *`Custom php.ini settings`* field on the *`Options`* tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).

- ***Recommendations:***

  Apache:

  - High-Traffic Web Sites: Fast-CGI + suExec or PHP-FPM + suExec
  - Low-Traffic Web Sites: CGI + suExec or SuPHP

  nginx:

  - PHP-FPM

- *`CGI available`*: This enables the CGI checkbox in the web form so that the reseller can select this feature for his web sites.
- *`SSI available`*: This enables the SSI checkbox in the web form so that the reseller can select this feature for his web sites.
- *`Perl available`*: This enables the Perl checkbox in the web form so that the reseller can select this feature for his web sites. This requires Apache's mod-perl and is therefore ***available only on Apache servers***.
- *`Ruby available`*: This enables the Ruby checkbox in the web form so that the reseller can select this feature for his web sites. This requires Apache's mod-ruby and is therefore ***available only on Apache servers***.
- *`Python available`*: This enables the Python checkbox in the web form so that the reseller can select this feature for his web sites. This requires Apache's mod-python and is therefore ***available only on Apache servers***.
- *`SuEXEC forced`*: If you select this feature, the reseller's PHP processes will be forced to use SuExec. This is useful especially for FastCGI, CGI, and PHP-FPM. SuExec is ***available on Apache servers only***.
- *`Custom error docs available`*: This enables the *`Custom error docs`* checkbox in the web form so that the reseller can select this feature for his web sites.
- *`Wildcard subdomain available`*: If you select this feature, the reseller will be able to select * (in addition to "none" and "www") as the auto-subdomain for his web sites.
- *`SSL available`*: This enables the SSL checkbox in the web form so that the reseller can select this feature for his web sites and create SSL certificates.
- *`Max. number of web aliasdomains`*: Specify the max. amount of web aliasdomains that this reseller can create. *`-1`* means unlimited.
- *`Max. number of web subdomains`*: Specify the max. amount of web subdomains that this reseller can create. *`-1`* means unlimited.
- *`Max. number of FTP users`*: Specify the max. amount of FTP users that this reseller can create. *`-1`* means

unlimited.

- *Max. number of Shell users*: Specify the max. amount of shell users that this reseller can create. *-1* means unlimited.
- *SSH-Chroot Options*: Specify which SSH modes should be available for the reseller when he creates/modifies a shell account. The *None* mode means that the shell user can browse the whole file system and is limited only by file/directory permissions - this can be a security risk. The *Jailkit* mode means that the shell user will be limited to his home directory (chrooted) and can only browse directories inside his home directory.
- *Max. number of Webdav users*: Specify the max. amount of WebDAV users that this reseller can create. *-1* means unlimited.
- *Default Mailserver*: Select the default mailserver for the reseller. The default mailserver will be pre-selected for this reseller when email items (email accounts, etc.) are created for the reseller, but this selection can be changed in the appropriate form.
- *Max. number of email domains*: Specify the max. amount of email domains that this reseller can create. *-1* means unlimited.
- *Max. number of mailing lists*: Specify the max. amount of mailing lists that this reseller can create. *-1* means unlimited.
- *Max. number of mailboxes*: Specify the max. amount of mailboxes that this reseller can create. *-1* means unlimited.
- *Max. number of email aliases*: Specify the max. amount of email aliases that this reseller can create. *-1* means unlimited.
- *Max. number of domain aliases*: Specify the max. amount of domain aliases that this reseller can create. *-1* means unlimited.
- *Max. number of email forwarders*: Specify the max. amount of email forwarders that this reseller can create. *-1* means unlimited.
- *Max. number of email catchall accounts*: Specify the max. amount of email catchall accounts that this reseller can create. *-1* means unlimited.
- *Max. number of email routes*: Specify the max. amount of email routes that this reseller can create. *-1* means unlimited.
- *Max. number of email filters*: Specify the max. amount of email filters that this reseller can create. *-1* means unlimited.
- *Max. number of fetchmail accounts*: Specify the max. amount of fetchmail accounts that this reseller can create. *-1* means unlimited.
- *Mailbox quota*: Specify the max. hard drive space (in MB) that this reseller's email accounts can use. *-1* means unlimited.
- *Max. number of spamfilter white / blacklist filters*: Specify the max. amount of whitelist and blacklist filters for the spamfilter that this reseller can create. *-1* means unlimited.

- *`Max. number of spamfilter users`*: Specify the max. amount of spamfilter users that this reseller can create. *`-1`* means unlimited.

- *`Max. number of spamfilter policies`*: Specify the max. amount of spamfilter policies that this reseller can create. *`-1`* means unlimited.

- *`Default Database Server`*: Select the default database server for the reseller. The default database server will be pre-selected for this reseller when a database is created for the reseller, but this selection can be changed in the appropriate form.

- *`Max. number of Databases`*: Specify the max. amount of databases that this reseller can create. *`-1`* means unlimited.

- *`Max. number of cron jobs`*: Specify the max. amount of cron jobs that this reseller can create. *`-1`* means unlimited.

- *`Max. type of cron jobs (chrooted and full implies url)`*: Specify which kind of cron jobs should be available for the reseller when he creates/modifies a cron job.

  - *`Full Cron`*: *`Full Cron`* means that you can use any command for the cron job, and it will ***not*** run in a chroot environment.

  - *`Chrooted Cron`*: If *`Chrooted Cron`* is selected in the limits of the reseller that owns the cron job, the cron jobs are chrooted (using Jailkit).

  - *`URL Cron`*: This means that the reseller can only create wget cron jobs, i.e., he specifies a URL in the cron job command line, and that URL will be accessed via wget.

- *`Min. delay between executions`*: This specifies the minimal delay (in minutes) how often a cron job can be executed. If you specify *`5`* here, for example, a cron job cannot be run every minute, but only every five minutes.

- *`Default DNS Server`*: Select the default DNS server for the reseller. The default DNS server will be pre-selected for this reseller when DNS items (zones, etc.) are created for the reseller, but this selection can be changed in the appropriate form.

- *`Max. number of DNS zones`*: Specify the max. amount of DNS zones that this reseller can create. *`-1`* means unlimited.

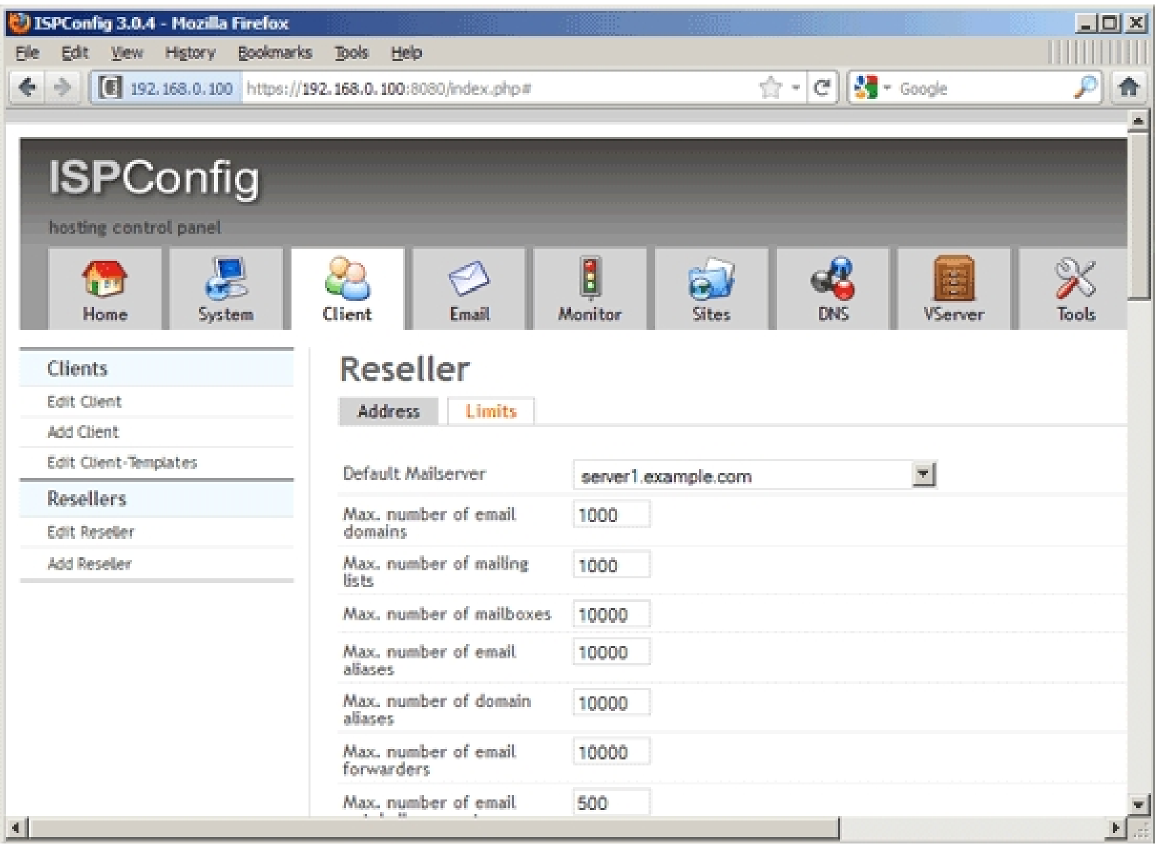
- *`Max. number of secondary DNS zones`*: Specify the max. amount of secondary DNS zones that this reseller can create. *`-1`* means unlimited.

- *`Max. number DNS records`*: Specify the max. amount of DNS records that this reseller can create. *`-1`* means unlimited.

- *`Max. number of virtual servers`*: Specify the max. amount of virtual servers that this reseller can create. *`-1`* means unlimited.

- *`Force virtual server template`*: If an OpenVZ template is selected here, the reseller can use only this template to create virtual machines. If no template is selected, the reseller can choose from all available OpenVZ templates.

- `Max. number of APS instances`: Specify the max. amount of APS packages that this reseller can install with the APS Installer. `-1` means unlimited.

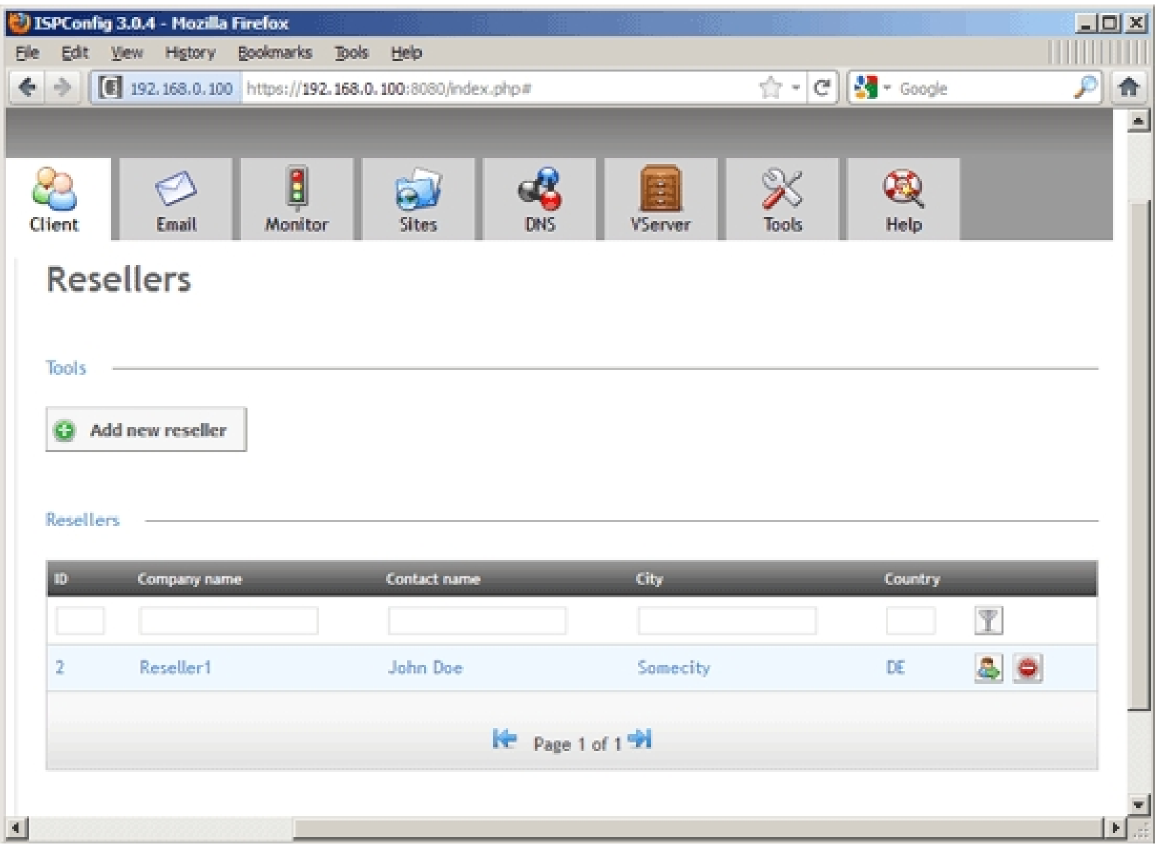


Reseller

Default Mailserver	server1.example.com
Max. number of email domains	1000
Max. number of mailing lists	1000
Max. number of mailboxes	10000
Max. number of email aliases	10000
Max. number of domain aliases	10000
Max. number of email forwarders	10000
Max. number of email	500

4.5.2.2 Edit Reseller

Under `Edit Reseller` you can find a list of existing resellers:


ISPConfig 3.0.4 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
192.168.0.100 https://192.168.0.100:8080/index.php#
Google
Client
Email
Monitor
Sites
DNS
VServer
Tools
Help
Tools
Add new reseller
Page 1 of 1

Resellers

Resellers

ID	Company name	Contact name	City	Country
2	Reseller1	John Doe	Somecity	DE

By clicking any of them, you will get to the *Address* and *Limits* tabs of that reseller (that you already know from chapter 4.5.2.1) where you can modify the settings of that reseller.

Above the list you can find filters that allow you to search for specific parameters in all resellers. The following filters are available:

- ID
- Company name
- Contact name
- City
- Country

Click the

button to start a search.

From the reseller list, it is also possible to directly log in as a reseller - just click the

button next to the reseller.

To delete a reseller, click the

⛔ button. A confirmation message will pop up, asking you if you really want to delete the record.

4.5.3 Messaging

You can use the messaging feature to send email messages (e.g. to announce downtime because of maintenance, etc.) to all customers and resellers and also to groups (called "circles") of customers and resellers, i.e., you can define circles of customers and resellers and then send a message to that circle only. We use the term "circle" here so that you don't mix this up with ISPConfig groups or web groups.

4.5.3.1 Edit Client Circle

You can use this link to create circles (= groups) of customers and/or resellers. Please note that if you don't create circles, you will send your messages always to ***all*** customers and resellers.

The form has the following fields:

- *Circle Name*: Specify a name for the circle, e.g. "Clients", "Resellers", "Clients in Data Center 10", "Clients with Hosting Package A", etc.
- *Clients/Resellers*: Select all members of this circle.
- *Description* (optional): Fill in a text that describes this circle.
- *Active*: Use this checkbox to activate/deactivate a circle. Messages can be sent only to active circles.

4.5.3.2 Send Email

This is where you compose your email message, select the recipient circle (or all customers and resellers if no circles are available) and send the message out to the recipients.

The form has the following fields:

- *Sender email address*: Type in the email address of the sender, e.g. *info@example.com*.
- *Recipient*: Select the recipent circle here.
- *Subject*: Type in the subject of the message.
- *Message*: Type in the message text. Available variables are listed to the right of the text area, you can insert them simply by clicking on the variable link (the variable will be inserted at the current cursor position).

4.6 Sites

On this tab we can create web sites, subdomains, FTP accounts, shell users, MySQL databases, and cron jobs, and take a look at traffic statistics.

4.6.1 Websites

4.6.1.1 Website

This is where we can create new and edit/delete existing web sites.

To create a new web site, click the *`Add new website`* button. This will lead you to the *`Web Domain`* form with the tabs *`Domain`*, *`Redirect`*, *`SSL`*, *`Statistics`*, and *`Options`*.

Some fields are relevant to Apache only, others to nginx and are only shown if the appropriate http server is installed.

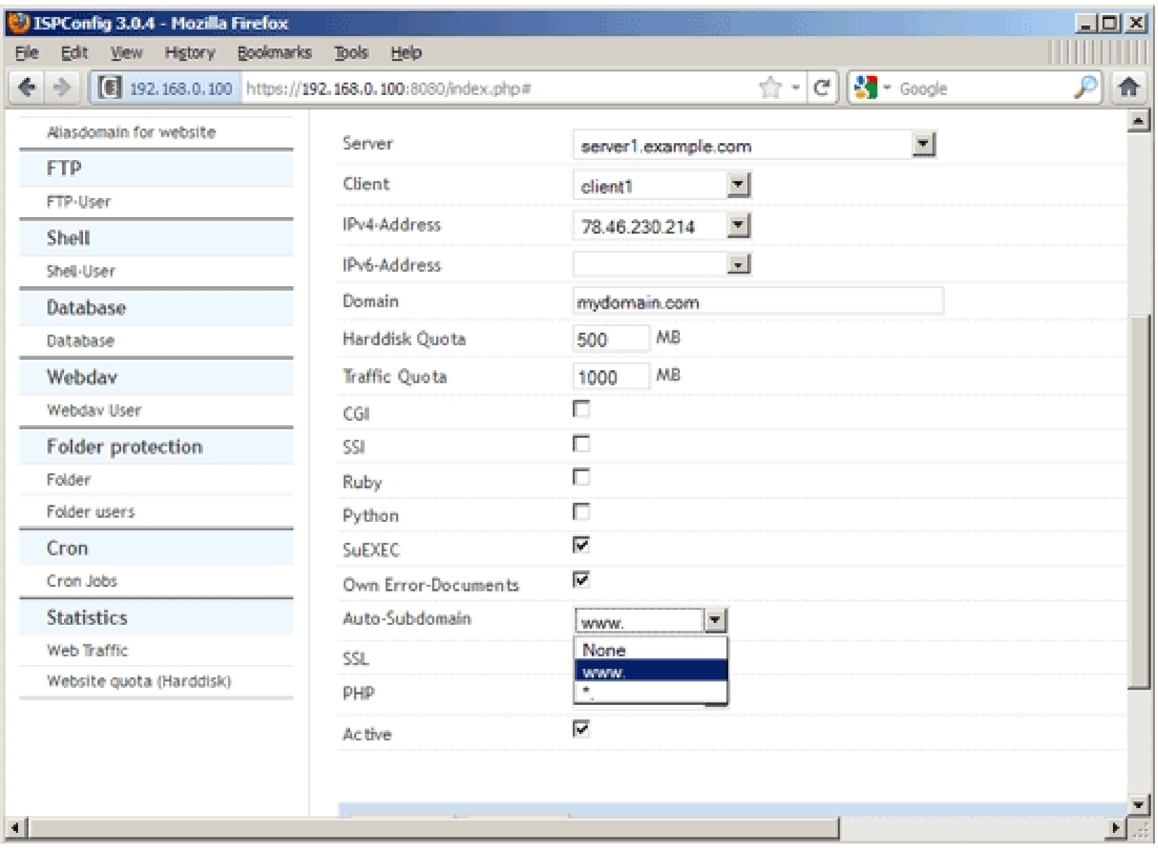
Web Domain

Domain

This is where the web site is actually created. Here you specify the web site domain, the client who owns the web site, the IP address, quota, the features (like PHP, CGI, SSL, etc.) that the web site will have, etc. The form has the following fields:

- *`Server`*: If more than one server is available, you can select the server on which the web site will be created.
- *`Client`*: Here you select the client that owns the new web site.
- *`IPv4-Address`*: Select the IPv4 address on which the web site will respond. *`*`* means all available IP addresses. Please note that you still might have to create the appropriate DNS records for your domains so that they point to the correct IP address.
- *`IPv6-Address`* (optional): Select the IPv6 address on which the web site will respond. If no IPv6 address is selected, no IPv6 vhost will be created. Please note that you still might have to create the appropriate DNS records for your domains so that they point to the correct IPv6 address.
- *`Domain`*: This is the main domain of your web site, e.g. *`example.com`* (without subdomain like *`www`*).
- *`Harddisk Quota`*: This is the max. amount of web space (in MB) that is available for the web site. *`-1`* means unlimited.
- *`Traffic Quota`*: This is the max. amount of traffic per month (in MB) that is available for the web site. *`-1`* means unlimited.
- *`CGI`*: Allows the web server to execute cgi scripts in a certain directory (*`cgi-bin`*).
- *`SSI`*: Activates Server Side Includes (SSI) (file extension .shtml).

- `Ruby` ***(Apache only)***: Allows the web server to execute Ruby scripts (file extensions `.rb` and `.rbx`).
- `Python` ***(Apache only)***: Allows the web server to execute Python scripts (file extension `.py`).
- `Perl` ***(Apache only)***: If you have mod_perl installed, this option allows you to run `.pl` scripts from within your document root and subdirectories instead of from `cgi-bin`.
- `SuEXEC` ***(Apache only)***: This makes that CGI scripts (including PHP scripts that are executed as Fast-CGI or CGI) are executed as the user and group of the current web site. You should check this checkbox for security reasons. This does not apply to PHP scripts that are executed under Mod-PHP and SuPHP.
- `Own Error-Documents`: Allows to define your own error pages instead of using the standard ones.
- `Auto-Subdomain`: Here you can define whether you want no automatic subdomain for the web site (in this case you can access the site only by using the domain, e.g. `http://example.com`), an automatic `www` subdomain (you can then access the site using `http://example.com` and `http://www.example.com`), or a wildcard subdomain (`*.`) which means you can access the site with any subdomain that does not point to another web site.



Server	server1.example.com
Client	client1
IPv4-Address	78.46.230.214
IPv6-Address	
Domain	mydomain.com
Harddisk Quota	500 MB
Traffic Quota	1000 MB
CGI	□
SSI	□
Ruby	□
Python	□
SuEXEC	☑
Own Error-Documents	☑
Auto-Subdomain	www. None www. *.
SSL	
PHP	
Active	☑

- *SSL*: With this checkbox you can enable SSL for this web site. Please note that you can have only one SSL web site per IP address, and it is not possible to use a wildcard (`*`) in the `IP-Address` field.
- *PHP*: You can disable/enable PHP for this web site here. If you want to enable PHP, the following five modes are available: Fast-CGI ***(Apache only)***, CGI ***(Apache only)***, Mod-PHP ***(Apache only)***, SuPHP ***(Apache only)***,

PHP-FPM (Apache and nginx).

- Fast-CGI:
  Advantages:
  - Scripts will be executed with user privileges of the web site;
  - More than one PHP version can be run as FastCGI;
  - Might be better in speed compared to CGI and suPHP.

  Disadvantages:
  - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the *`Custom php.ini settings`* field on the *`Options`* tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).

- CGI:
  Advantages:
  - Scripts will be executed with user privileges of the web site;
  - More than one PHP version can be run as CGI.

  Disadvantages:
  - CGI might use a little more memory (RAM) - therefore, it's not recommended to run PHP as CGI on slow virtual servers;
  - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the *`Custom php.ini settings`* field on the *`Options`* tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).

- Mod-PHP:
  Advantages:
  - Speed;
  - Needs less memory (RAM) than CGI;
  - php.ini values can be changed via PHP scripts, vhost files, .htaccess files.

  Disadvantages:
  - Scripts are being executed with Apache privileges, which might lead to some security related problems;
  - Only one version of PHP can be installed as Apache module;

- You ***cannot*** use the `Custom php.ini settings` field on the `Options` tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).

- SuPHP:
  Advantages:
  - Scripts will be executed with user privileges of the web site;
  - Each vhost can have its own php.ini file;
  - Needs less memory (RAM) than CGI;
  - More than one PHP version can be run as suPHP.

  Disadvantages:
  - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the `Custom php.ini settings` field on the `Options` tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***);
  - SuPHP might be a little slower than mod_php.

- PHP-FPM:
  Advantages:
  - Scripts will be executed with user privileges of the web site;
  - More than one PHP version can be run as PHP-FPM;
  - Adaptive process spawning;
  - Advanced process management with graceful stop/start;
  - Emergency restart in case of accidental opcode cache destruction;
  - Might be better in speed compared to CGI and suPHP.

  Disadvantages:
  - php.ini values cannot be changed via PHP scripts, vhost files, .htaccess files. But it is possible to use the `Custom php.ini settings` field on the `Options` tab of a web site in ISPConfig to specify custom php.ini settings (see chapter ***4.6.1.1 Website***).

- ***Recommendations:***
  Apache:
  - High-Traffic Web Sites: Fast-CGI + suExec or PHP-FPM + suExec

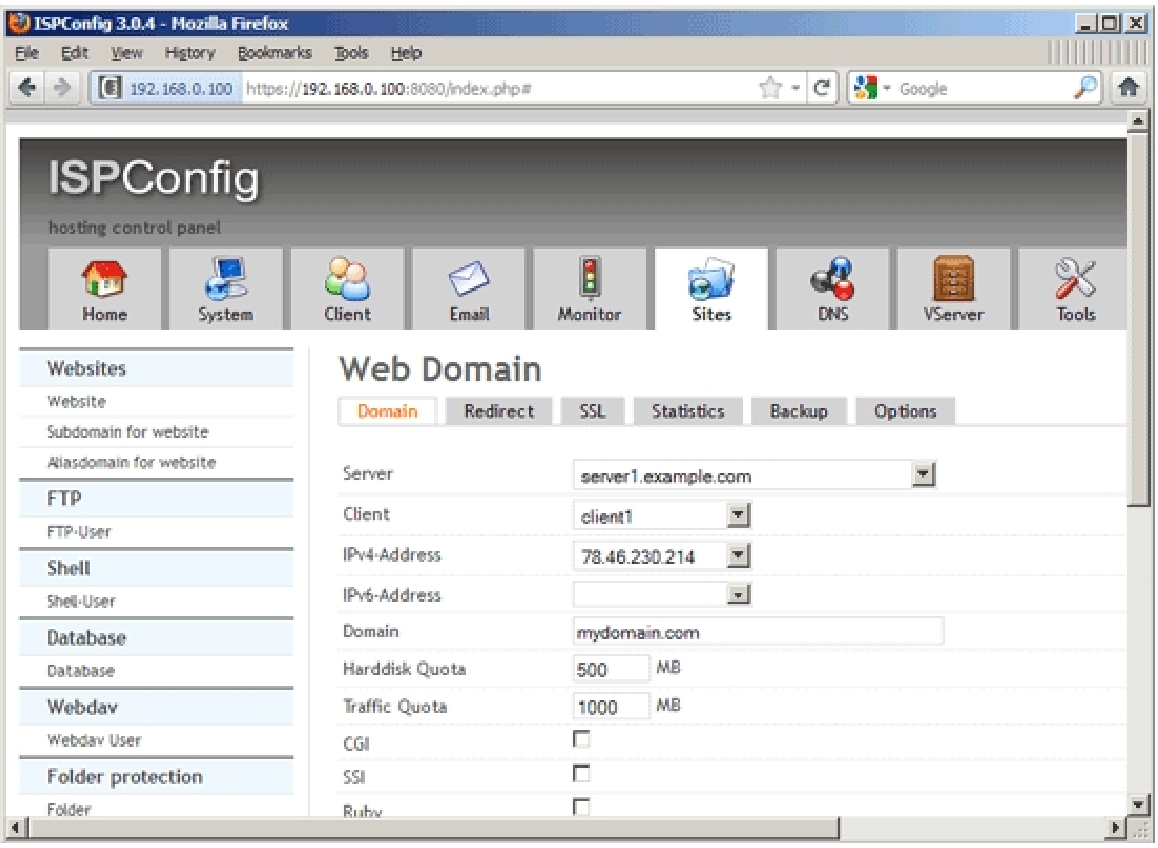
- Low-Traffic Web Sites: CGI + suExec or SuPHP

nginx:

- PHP-FPM

- `PHP Version` ***(FastCGI and PHP-FPM only)***: If more than one PHP version is available, you can select the desired PHP version for this web site here. Please note that this feature is available only for FastCGI and PHP-FPM. You can add PHP versions to ISPConfig under System > Additional PHP Versions (see chapter ***4.9.2.4 Additional PHP Versions***).

- `Active`: Defines whether this web site is active or not.



Server	server1.example.com
Client	client1
IPv4-Address	78.46.230.214
IPv6-Address	
Domain	mydomain.com
Harddisk Quota	500 MB
Traffic Quota	1000 MB
CGI	□
SSI	□
Ruby	□

Redirect

This form allows you to redirect the web site to another web site or to a specific directory on the server. This is done by using Apache/nginx rewrite rules.

- *`Redirect Type`*: Here you can specify if you want to disable/enable a redirect, and if you decide to use a redirect, which flag to use.

  Flags:

  - No flag: Don't use any flags.
  - R ***(Apache only)***: Use of the [R] flag causes a HTTP redirect to be issued to the browser. If a fully-qualified URL is specified (that is, including http://servername/) then a redirect will be issued to that location. Otherwise, the current servername will be used to generate the URL sent with the redirect.
  - L ***(Apache only)***: The [L] flag causes mod_rewrite to stop processing the rule set. In most contexts, this means that if the rule matches, no further rules will be processed.
  - R,L ***(Apache only)***: You will almost always want to use [R] in conjunction with [L] (that is, use [R,L]) because on its own, the [R] flag prepends http://thishost[:thisport] to the URI, but then passes this on to the next rule in the ruleset, which can often result in 'Invalid URI in request' warnings.
  - last ***(nginx only)***: Completes processing of rewrite directives, after which searches for corresponding URI and location.
  - break ***(nginx only)***: Completes processing of rewrite directives and breaks location lookup cycle by not doing any location lookup and internal jump at all.
  - redirect ***(nginx only)***: Returns temporary redirect with code 302; it is used if the substituting line begins with http://.
  - permanent ***(nginx only)***: Returns permanent redirect with code 301.
  - proxy ***(nginx only)***: This option (which does not refer to an official nginx rewrite flag) allows to proxy requests which allows to display contents from somewhere else without changing the web site URL. If *`Redirect Path`* is an external URL, the contents is fetched from that URL ($request_uri is appended to that URL); if *`Redirect Path`* is a directory or the URL is local (i.e., it points to a subdirectory of the current web site, like the web site address is *`http://www.example.com`*, and you add *`http://www.example.com/subdir`* to the *`Redirect Path`* field), the request isn't actually proxied, but the web site's document root is set to the subdirectory (so that the document root is something like *`/var/www/example.com/web/subdir`* instead of *`/var/www/example.com/web`*). If the proxy option is selected, a further field called *`Proxy Directives`* is added to the *`Options`* tab where you can add custom nginx proxy directives (directives like, for example, *`proxy_set_header`*, *`proxy_redirect`*, *`proxy_buffer_size`*, etc.) - these will be added to the vhost configuration if *`Redirect Path`* is an external URL.

More details about flags can be found here:

- Apache: ***http://httpd.apache.org/docs/2.2/rewrite/flags.html***
- nginx: ***http://wiki.nginx.org/NginxHttpRewriteModule#rewrite***

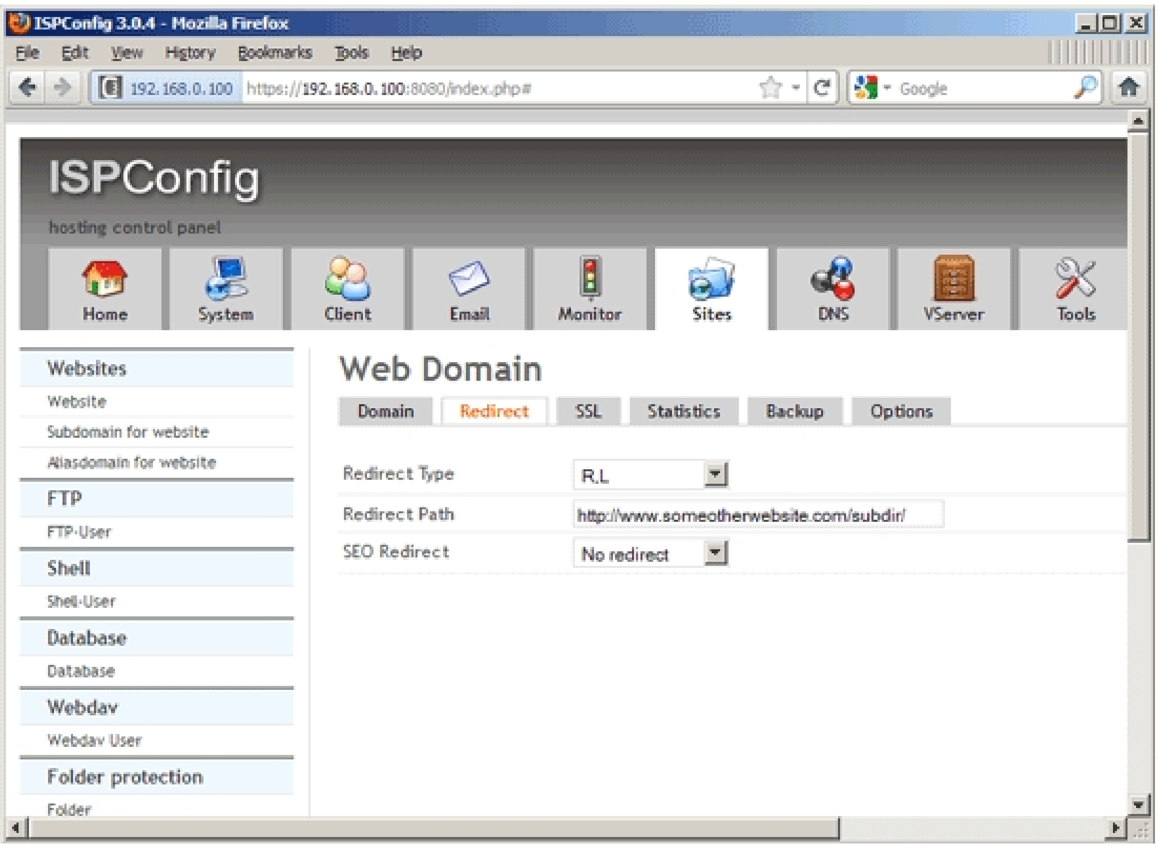
- *`Redirect Path`*: This is the target, i.e., the path (full path or path relative to the document root) or URL where the redirect should point to.

- *SEO Redirect*: Here you can do search-engine optimization for your website and configure a redirect to avoid duplicate content. You can redirect your non-www website to your www website (e.g. visitors to *example.com* will be redirected permanently to *www.example.com*) or vice versa (*www.example.com* to *example.com*). These are the options:

  - *No Redirect*: Don't use any SEO redirect.
  - *domain.tld => www.domain.tld*: Redirect requests for *example.com* to *www.example.com*.
  - *www.domain.tld => domain.tld*: Redirect requests for *www.example.com* to *example.com*.
  - **.domain.tld => domain.tld*: Redirect all subdomains of *example.com* (including *www.example.com*) to *example.com*.
  - **.domain.tld => www.domain.tld*: Redirect all subdomains (including *example.com* itself) to *www.example.com*.
  - ** => domain.tld*: Redirect everything that is not *example.com* to *example.com* (this includes subdomains and also alias domains).
  - ** => www.domain.tld*: Redirect everything that is not *www.example.com* to *www.example.com* (this includes *example.com*, subdomains and also alias domains).

If you want to do a URL redirect, you should use the R,L flags, while for a directory redirect it is recommended to justuse the L flag.

If you want to do a URL redirect, please specify the redirect target URL in the *Redirect Path* field (e.g. *http://www.someotherwebsite.com/subdir/* or *http://www.someotherwebsite.com/*). Please note that the URL should have a trailing slash:

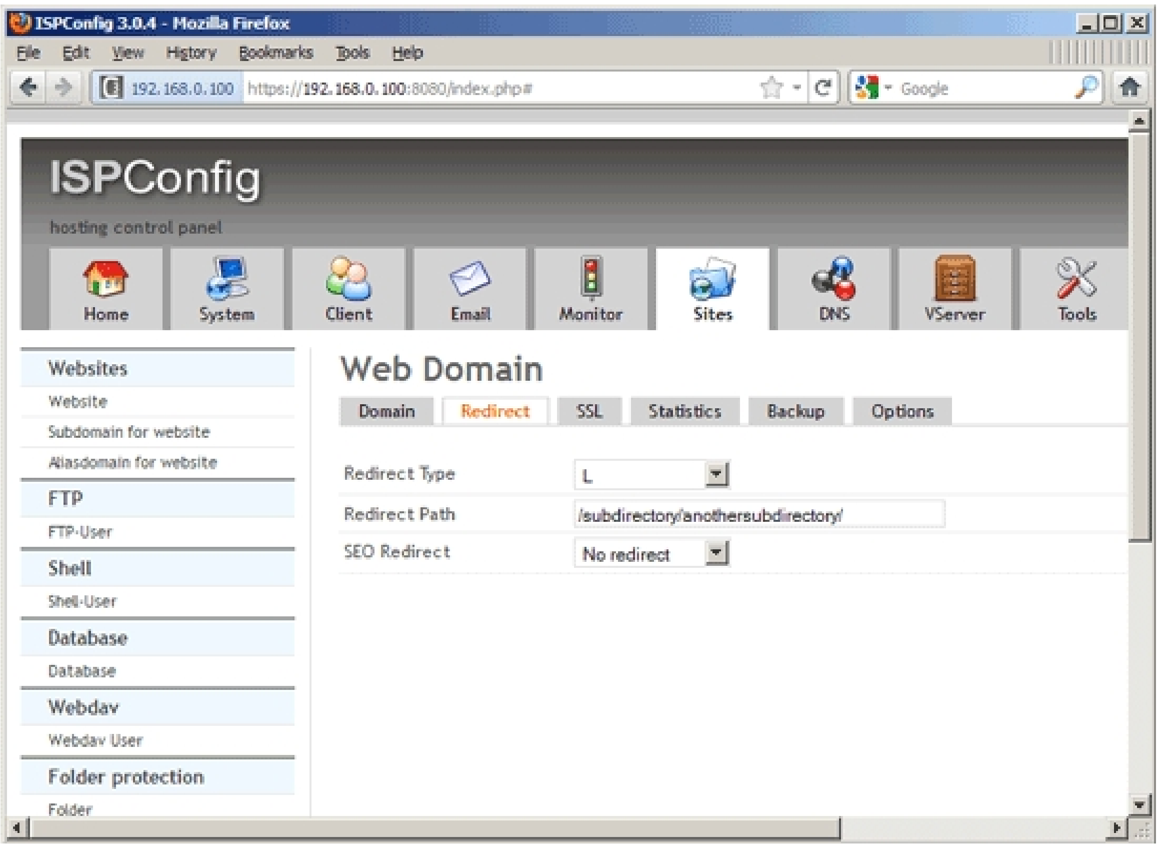


Redirect Type: R,L

Redirect Path: http://www.someotherwebsite.com/subdir/

SEO Redirect: No redirect

If you want to do a redirect to a subdirectory of your web site, please specify the subdirectory or the path to the subdirectory (relative to the document root of your web site) in the *Redirect Path* field. Please note that the path must begin and end with a slash (e.g. */subdirectory/anothersubdirectory/*):



Web Domain

Redirect Type: L

Redirect Path: /subdirectory/anothersubdirectory/

SEO Redirect: No redirect

SSL

On the *SSL* tab you can create a self-signed SSL certificate together with a certificate signing request (CSR) that you can use to apply for an SSL certificate that is signed by a trusted certificate authority (CA) such as Verisign, Comodo, Thawte, etc. It's not necessary to buy such a trusted SSL certificate, but you should note that if you use a self-signed SSL certificate, browsers will display a warning to your visitors.

Please note that you can have just one SSL web site per IP address, unless you use SNI (see chapter ***4.9.2.2 Server Config***). SNI is short for `Server Name Indication` and allows you to run multiple SSL vhosts on one IP address. Please note that currently SNI is not supported by all browsers/operating systems. Browsers/clients with support for TLS server name indication:

- Opera 8.0 and later (the TLS 1.1 protocol must be enabled)

- Internet Explorer 7 or later (under Windows Vista and later only, not under Windows XP)

- Firefox 2.0 or later

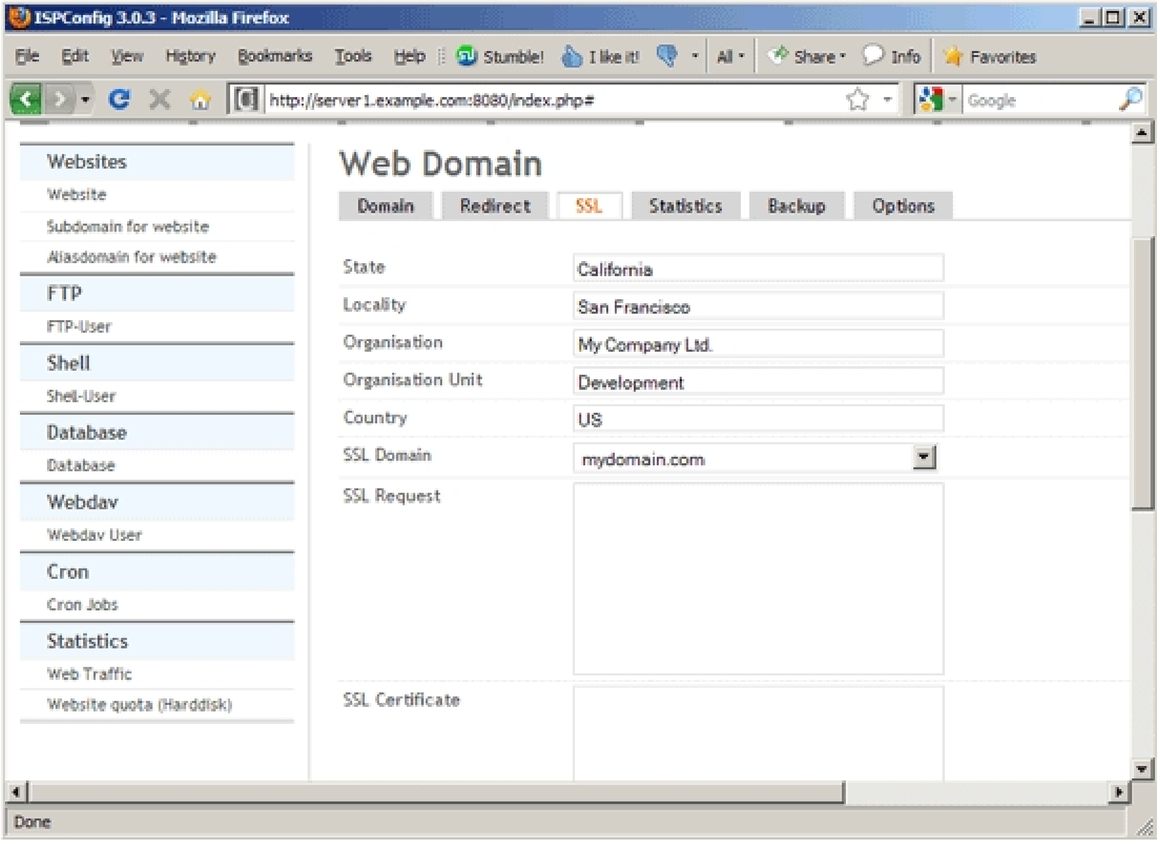
- Curl 7.18.1 or later (when compiled against an SSL/TLS toolkit with SNI support)

- Chrome 6.0 or later (on all platforms - releases up to 5.0 only on specific OS versions)

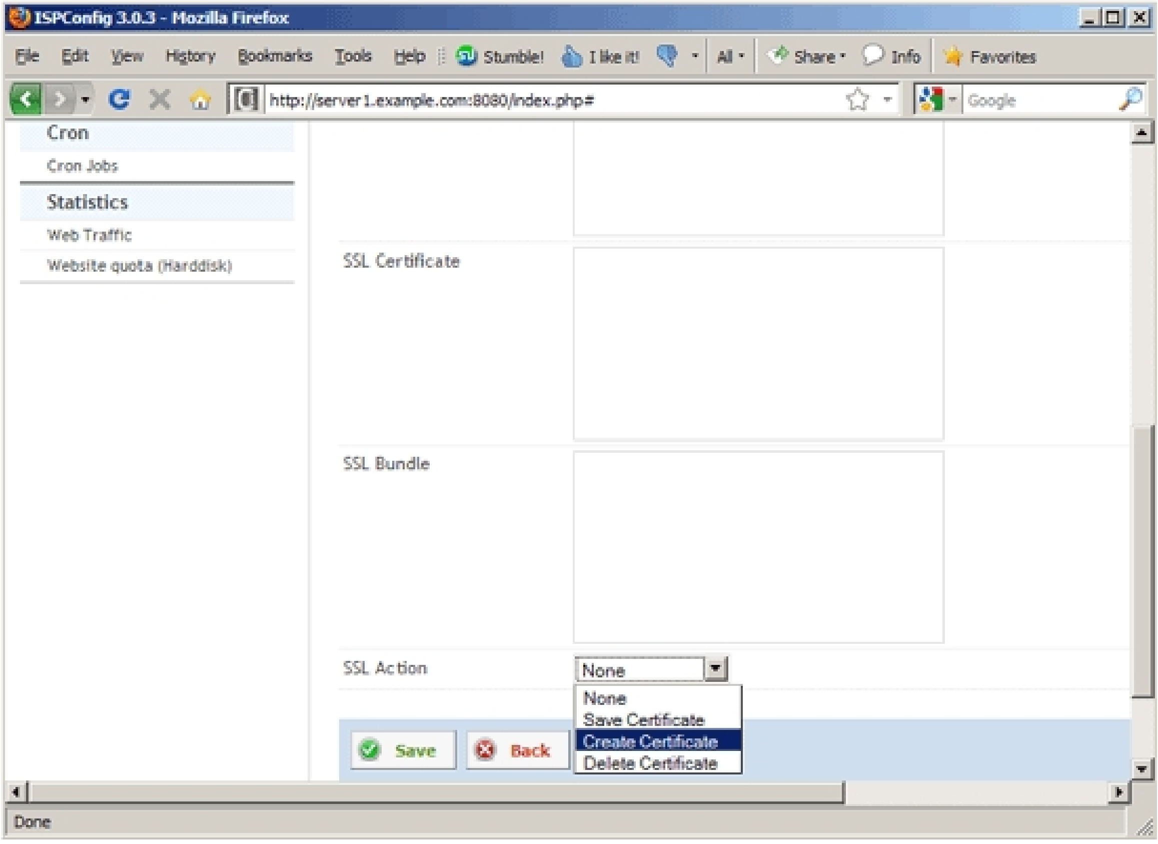
- Safari 3.0 or later (under OS X 10.5.6 or later and under Windows Vista and later)

To find out if your browser supports SNI, you can go to ***https://alice.sni.velox.ch/***.

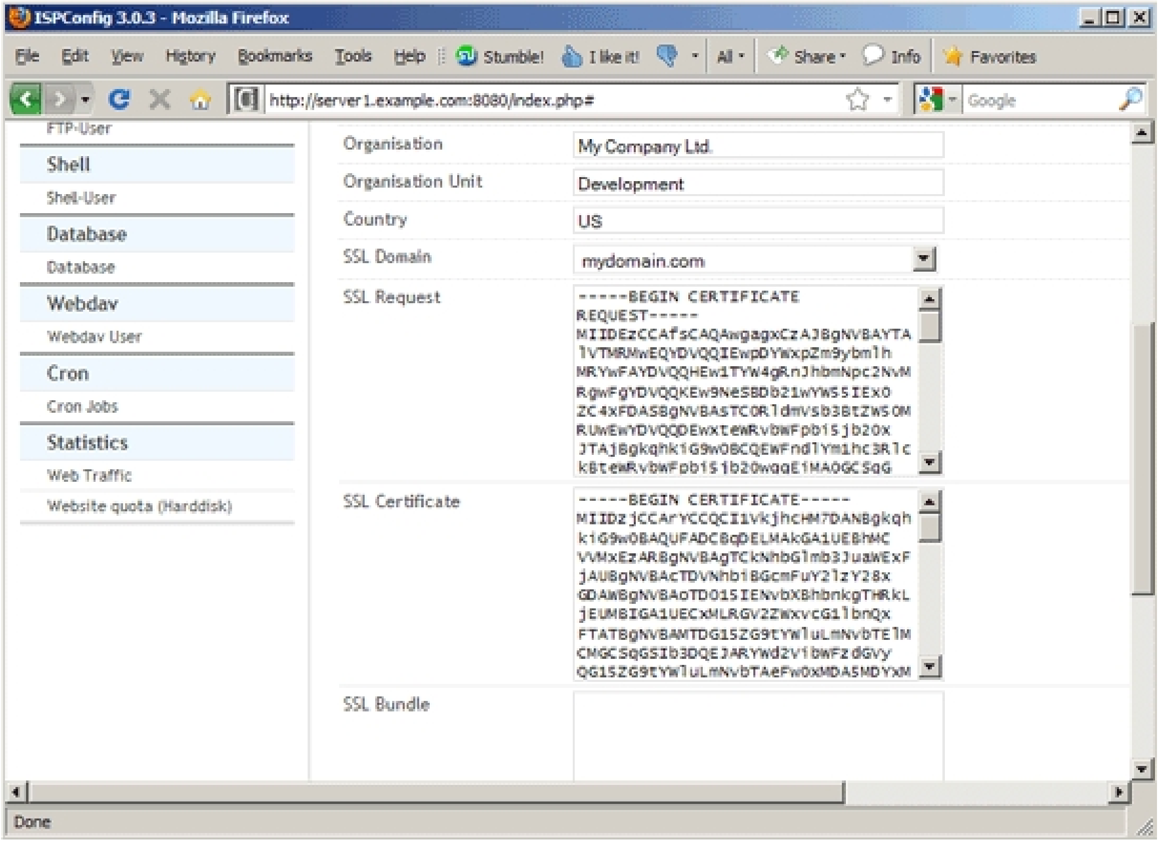
To create a self-signed certificate, please fill out the fields `State`, `Locality`, `Organisation`, `Organisation Unit`, `Country`, and `SSL Domain`, and then select `Create Certificate` from the `SSL Action` drop-down menu, and click on `Save`. Leave the fields SSL Key, `SSL Request`, `SSL Certificate`, and `SSL Bundle` empty - the fields SSL Key, `SSL Request` and `SSL Certificate` will be filled out by the system.



State	California
Locality	San Francisco
Organisation	My Company Ltd.
Organisation Unit	Development
Country	US
SSL Domain	mydomain.com
SSL Request	
SSL Certificate	



After the self- signed certificate was created, you will find data in the SSL Key, *SSL Request*, and *SSL Certificate* fields (it can take one or two minutes until the data appears in the fields):


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
FTP-User
Shell
Shell-User
Database
Database
Webdav
Webdav User
Cron
Cron Jobs
Statistics
Web Traffic
Website quota (Harddisk)
Done

Organisation: My Company Ltd.

Organisation Unit: Development

Country: US

SSL Domain: mydomain.com

SSL Request:

```
-----BEGIN CERTIFICATE
REQUEST-----
MIIDEzCCAfsCAQAwgagxCzAJBgNVBAYTA
lVTMRMwEQYDVQQIEwpDYWxpZm9ybmlh
MRYwFAYDVQQHEw1TYW4gRnJhbmNpc2NvM
RgwFgYDVQQKEw9NeSBDb21wYW55IExO
ZC4xFDASBgNVBASTC0RldmVsb3BtZW5OM
RUwEwYDVQQDEwxtewRvbWFpbi5jb20x
JTAjBgkqhkiG9w0BCQEWFndlYm1hc3Rlc
kBtewRvbWFpbi5ib20wggEiMA0GCSqG
```

SSL Certificate:

```
-----BEGIN CERTIFICATE-----
MIIDzjCCArYCCQCI1VkjhcHM7DANBgkqh
kiG9w0BAQUFADCBqDELMAkGA1UEBhMC
VVMxEzARBgNVBAgTCkNhbGlmb3JuaWExF
jAUBgNVBAcTDVNhbi BGcmFuY2lzY28x
GDAWBgNVBAoTD015IENvbXBhbnkgTHRkL
jEUMBIGA1UECxMLRGV2ZWxvcG1lbnQx
FTATBgNVBAMTDG15ZG9tYWluLmNvbTElM
CMGCSqGSIb3DQEJARYWd2VibWFzdGVy
QG15ZG9tYWluLmNvbTAeFw0xMDA5MDYxM
```

SSL Bundle:

If you want to buy an SSL certificate from a trusted CA, you have to copy the data from the SSL Request field - this is the certificate signing request (CSR). With this CSR, you can apply for a trusted SSL certificate at your CA - the CA will create an SSL certificate from this CSR, and you can paste the trusted SSL certificate into the *SSL Certificate* field. Sometimes your CA will also give you an SSL bundle - paste this into the *SSL Bundle* field. Select *Save Certificate* from the *SSL Action* drop-down menu and click on the *Save* button. You have just replaced your self-signed certificate with a trusted SSL certificate.

If you already have an SSL certificate that you would like to use with this web site, it's not necessary to create a self-signed certificate first. Just paste the key, the certificate, the bundle certificate (if needed) and the CSR (optional, but will be needed if you want to buy a new certificate for the same key, for example after the old certificate has expired) in the appropriate fields, select *Save Certificate* from the *SSL Action* drop-down menu and click on *Save* (the other fields such as *State*, *Organisation*, etc. can be left empty).

To delete a certificate, select *Delete Certificate* from the *SSL Action* drop-down menu and click on the *Save* button.

Here's the meaning of the other fields on the *SSL* tab:

- *State*: The state or province where your organization is located. Can not be abbreviated. Examples: Florida, Bavaria, Noord-Holland, etc.
- *Locality*: The city where your organization is located. Examples: London, Paris, Seattle, Hamburg, etc.
- *Organisation*: The exact legal name of your organization. Do not abbreviate your organization name. Examples: Internet Widgets Pty Ltd, My Company GmbH, etc.

- `Organisation Unit`: This entry is for the name of the unit in your organization. Examples: Marketing, Sales, Development, etc.
- `Country`: The two-letter ISO abbreviation for your country. Examples: AU for Australia, DE for Germany, US for the United States, NL for The Netherlands, etc.
- `SSL Domain`: A fully qualified domain name that resolves to the SSL web site. For example, if you intend to secure the URL `https://ssl.example.com`, then the `SSL Domain` must be `ssl.example.com`. This must be an exact match. Some CAs automatically include `www.example.com` in the certificate if you create a CSR for `example.com`, while for other CAs, you need to create a CSR for `www.example.com` if you want to have both `example.com` and `www.example.com` covered - it's therefore recommended to create CSRs for `www.example.com` instead of just `example.com`. If you want to create a wildcard certificate, please select `*.example.com` in this field.

Statistics

ISPConfig 3 can create web statistics for your web sites, but only if you specify a password for the webstatistics user - these will be generated once a day (at 0.30h) and are available in the `/stats` folder of your web site (e.g. `http://www.example.com/stats`). You can password-protect that directory by specifying a password in the `Webstatistics password` field (the Webstatistics username is defined by ISPConfig, it's `admin`).

In the `Webstatistics program` drop-down menu, you can select the software that will create the statistics for you - you have the choice between ***Webalizer*** and ***AWStats***.

- `Webstatistics username`: This is preset by ISPConfig, the username is `admin`.
- `Webstatistics password`: Type in a password for the user (or use the `Generate Password` link to have ISPConfig generate one for you).
- `Repeat Password`: Confirm your password.
- `Webstatistics program`: Select if you want to use Webalizer or AWStats for generating the web site statistics.

Backup

On the `Backup` tab you can specify whether you want to create backups of the current web site and all MySQL databases that belong to the web site (separate backups will be created for the web site and for the MySQL dumps so that it is possible to restore either one or the other). If the document root of the web site is `/var/clients/client1/web1/web`, the **contents** of the `/var/clients/client1/web1` directory (including the `web` folder, but excluding the `log` folder) will be compressed and stored in the backup directory that is specified under `System > Server Config > Backup directory` (the default directory is `/var/backup`). For `web1` ISPConfig would create the subdirectory `/var/backup/web1` and store the backups in that directory.

- *Backup interval*: Select whether you want ISPConfig to create backups for this web site, and if so, how often (daily/weekly/monthly).
- *Number of backup copies*: Specify how many backups should be kept on the system. For example, if you select to have a daily backup and pick *10* in the *Number of backup copies* field, the sytem will keep backups of the last ten days; backups that are older will automatically be deleted.

To restore a backup, click on the *Restore* button next to the backup you want to restore. The restore will be processed in the next few minutes. If you want to download a backup, click on the *Download* button; within the next few minutes, the backup will be placed in the *backup* directory from where you can download it via FTP.

Options

(This tab is visible only for the ISPConfing *admin* user.)

- *Linux User*: This shows the Linux user under which this web site is run. If you have chosen PHP Fast-CGI + SuEXEC, PHP CGI + SuEXEC or SuPHP, this is the user under which your PHP scripts will be executed. This setting cannot be changed.
- *Linux Group*: This shows the Linux group under which this web site is run. If you have chosen PHP Fast-CGI + SuEXEC, PHP CGI + SuEXEC or SuPHP, this is the group under which your PHP scripts will be executed. This setting cannot be changed.
- *Apache AllowOverride* ***(Apache only)***: Specifies what directives are allowed in *.htaccess* files. Possible values: *All|None|AuthConfig|FileInfo|Indexes|Limit|Options[= Option ,...]* See ***http://httpd.apache.org/docs/2.2/mod/core.html#allowoverride*** for more details.
- *Use Socket For PHP-FPM*: By default, ISPConfig configures TCP connections for PHP-FPM. If you check this box, a socket connection is configured instead which reduces networking overhead. In addition to that, no port is used.
- PHP-FPM Process Manager: Select the desired PHP-FPM process manager (*static*, *dynamic*, or *ondemand* - *ondemand* requires PHP Version 5.3.9 or later). For minimal resource usage, *ondemand* is the preferred value. Depending on what you select here, the following PHP-FPM settings fields will appear or disappear. Here is a brief description of the three process managers:

```
static  - a fixed number (pm.max_children) of child processes;
dynamic - the number of child processes are set dynamically based on the
          following directives. With this process management, there will be
          always at least 1 children.
          pm.max_children      - the maximum number of children that can
                                 be alive at the same time.
          pm.start_servers     - the number of children created on startup.
          pm.min_spare_servers - the minimum number of children in 'idle'
                                 state (waiting to process). If the number
                                 of 'idle' processes is less than this
```

```
                         number then some children will be created.
        pm.max_spare_servers - the maximum number of children in 'idle'
                         state (waiting to process). If the number
                         of 'idle' processes is greater than this
                         number then some children will be killed.
ondemand - no children are created at startup. Children will be forked when
        new requests will connect. The following parameter are used:
        pm.max_children       - the maximum number of children that
                           can be alive at the same time.
        pm.process_idle_timeout  - The number of seconds after which
                           an idle process will be killed.
```

- *PHP-FPM pm.max_children*: The maximum number of child processes to be created when pm is set to 'dynamic'. This value sets the limit on the number of simultaneous requests that will be served. Equivalent to the ApacheMaxClients directive with mpm_prefork. Equivalent to the PHP_FCGI_CHILDREN environment variable in the original PHP CGI.
- *PHP-FPM pm.start_servers* ***(dynamic only)***: The number of child processes created on startup. Default Value: min_spare_servers + (max_spare_servers - min_spare_servers) / 2
- *PHP-FPM pm.min_spare_servers* ***(dynamic only)***: The desired minimum number of idle server processes.
- *PHP-FPM pm.max_spare_servers* ***(dynamic only)***: The desired maximum number of idle server processes.

Values of PHP-FPM pm settings must be as follows: pm.max_children >= pm.max_spare_servers >= pm.start_servers >= pm.min_spare_servers > 0

- *PHP-FPM pm.max_requests*: The number of requests each child process should execute before respawning. This can be useful to work around memory leaks in 3rd party libraries. For endless request processing specify '0'. Equivalent to PHP_FCGI_MAX_REQUESTS.
- *PHP-FPM pm.process_idle_timeout* ***(ondemand only)***: The number of seconds after which an idle process will be killed. Default Value: *10* (seconds).
- *PHP open_basedir*: The *open_basedir* directive in php.ini limits PHP file accesses (such as file opening, writing and deleting) within a designated directory so that it doesn't endanger the rest of the system in any way. With proper Apache permissions and PHP installed as an Apache module, PHP inherits whatever privileges Apache has. You can specify multiple directories here, seperated by a colon (*:*). To disable open_basedir, please specify the string *none* (leaving the field empty will not work).
- *Custom php.ini settings*: If this web site needs special PHP settings that differ from what's in the system's global php.ini, you can override the global PHP settings here. You can use normal php.ini syntax here. Please specify one directive per line. Please note that you can use this field only with Fast-CGI, CGI, or SuPHP - you cannot use it if you have enabled Mod-PHP for this web site. Also note that if you use this field and change your global php.ini afterwards, the changes in the global php.ini will not be available to this web site immediately - only after you modify settings of this web site in ISPConfig so that this web site's configuration gets rewritten. If you have defined PHP Directive Snippets under *System > Directive Snipptes* (see chapter ***4.9.2.5 Directive Snippets***), you will find thise snippets listed by their name to the right of the textarea. If you click on the name of a snippet, the contents of the snippet will be inserted at the current cursor position.

Examples:

```
memory_limit = 32M
magic_quotes_gpc = Off
file_uploads = Off
```

- *`Apache Directives`* ***(Apache only)***: This field offers you the opportunity to write additional Apache directives into the site's virtual host container manually, one directive per line (***Directive Quick Reference***). If you have defined Apache Directive Snippets under `System > Directive Snipptes` (see chapter ***4.9.2.5 Directive Snippets***), you will find thise snippets listed by their name to the right of the textarea. If you click on the name of a snippet, the contents of the snippet will be inserted at the current cursor position.
  Examples:

  ```
  <Location '/wiki/images'>
   php_admin_flag engine off
   AddType text/plain .html .htm .shtml .php
  </Location>
  php_flag register_globals off
  Options -Indexes
  Options +FollowSymLinks
  ErrorDocument 404 /index.php
  ```

  (As you can see, you can change PHP settings here as well using *`php_admin_flag`* and *`php_flag`*, but this works only if you use Mod-PHP. You can find more details about this here: ***http://php.net/manual/en/configuration.changes.php***)

- *`nginx Directives`* ***(nginx only)***: This field offers you the opportunity to write additional nginx directives into the site's virtual host container manually, one directive per line (***Directive Quick Reference***). If you have defined nginx Directive Snippets under `System > Directive Snipptes` (see chapter ***4.9.2.5 Directive Snippets***), you will find thise snippets listed by their name to the right of the textarea. If you click on the name of a snippet, the contents of the snippet will be inserted at the current cursor position.
  Please note that if you use a location here that is already in use in the vhost (like `location ~ /. {}` or `location @php {}`, the default behaviour of ISPConfig is to replace the original location block with the one you specify (unless you add the string `##merge##` right of the location line which will make ISPConfig merge your location block into the original location block).
  Examples:

  ```
          location / {
                  if ($query_string ~ ".+") {
                          return 405;
                  }
                  # pass requests from logged-in users to Apache
                  if ($http_cookie ~ "DRUPAL_UID" ) {
                          return 405;
                  } # pass POST requests to Apache
                  if ($request_method !~ ^(GET|HEAD)$ ) {
                          return 405;
                  }
                  error_page 405 = @nocache;
                  # do not allow browsers to cache HTML
                  add_header Expires "Sun, 19 Nov 1978 05:00:00 GMT";
  ```

```
add_header Cache-Control "no-store, no-cache, must-revalidate, post-check=0
```

```
# serve requested content from the cache if available, otherwise pass the r
```

```
                try_files /cache/normal/$host/${uri}_.html /cache/perm/$host/${uri}_.css /c
        }

        location @nocache {
                try_files $uri $uri/ /index.php?$args;
        }

        location ~*  .(jpg|jpeg|png|gif|css|js|ico)$ {
                expires max;
                log_not_found off;
        }
```

This will replace the original location `location @php {}` block:

```
        location @php {
                try_files $uri =404;
                include /etc/nginx/fastcgi_params;
                fastcgi_pass unix:/var/lib/php5-fpm/web1.sock;
                fastcgi_index index.php;
                fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
                fastcgi_param PATH_INFO $fastcgi_script_name;
                fastcgi_intercept_errors on;
                fastcgi_buffer_size 128k;
                fastcgi_buffers 256 16k;
                fastcgi_busy_buffers_size 256k;
                fastcgi_temp_file_write_size 256k;
                fastcgi_max_temp_file_size 0;
                fastcgi_read_timeout 7200;
        }
```

And this will merge the directives inside the location block into the original location block:

```
        location @php { ##merge##
                fastcgi_buffer_size 128k;
                fastcgi_buffers 256 16k;
                fastcgi_busy_buffers_size 256k;
                fastcgi_temp_file_write_size 256k;
                fastcgi_max_temp_file_size 0;
                fastcgi_read_timeout 7200;
        }
```

- `Proxy Directives` **(nginx only)**: If you are using a redirect of the type proxy to an external URL on the Redirect tab, this field offers you the opportunity to write additional proxy directives into the site's virtual host container manually, one directive per line (***Directive Quick Reference***). If you have defined Proxy Directive Snippets under `System > Directive Snipptes` (see chapter ***4.9.2.5 Directive Snippets***), you will find thise snippets listed by their name to the right of the textarea. If you click on the name of a snippet, the contents of the snippet will be inserted at the current cursor position.

  Examples:

  ```
  proxy_set_header Host $host;
    proxy_redirect off;
  ```

```
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_pass_header Authorization;
    client_max_body_size 0;
    client_body_buffer_size 1m;
    proxy_intercept_errors on;
    proxy_buffering on;
    proxy_buffer_size 128k;
    proxy_buffers 256 16k;
    proxy_busy_buffers_size 256k;
    proxy_temp_file_write_size 256k;
    proxy_max_temp_file_size 0;
  proxy_read_timeout 300;
```

4.6.1.2 Subdomain for website

This is where we can create new and edit/delete existing subdomains. With this feature, you can add subdomains to an existing web site so that the subdomain shows the same content as the web site's main domain. It is also possible to point the subdomain to a subdirectory of the web site - this is done using Apache rewrite rules. Please note that you should not use such a rewrite rule if you plan to install a CMS such as Wordpress, Joomla, Drupal, etc. in that subdirectory because most modern CMS systems also use rewrite rules that will most likely collide with the rewrite rules that redirect the subdomain to the subdirectory. If you want to install a CMS in a directory of its own and use a subdomain for that directory, you should create a whole new web site for that subdomain and install the CMS in that web site. But if you plan to place static HTML files in the subdirectory or other stuff that doesn't come with any rewrite rules, you can create a subdomain and redirect it to that subdirectory without any problem.

The difference between a subdomain and an aliasdomain is that the subdomain uses the same domain name as the main domain of the web site, whereas an aliasdomain uses a different domain name. For example, if the web site's main domain is `example.com`, and you want to point the hostname `sub.example.com` to the same web site, you'd use a subdomain, whereas if you have a totally different domain such as `yourseconddomain.com` that you want to point to the `example.com` web site, you'd use an aliasdomain.

To create a new subdomain, click the `Add new subdomain` button. This will lead you to the `Subdomain for website` form with the tab `Domain`.

Subdomain for website

Domain

Here you can create/edit the subdomain. The form has the following fields:

- `Host`: This is where you enter the hostname, i.e., the subdomain without the main domain name. For example, if you want to create the subdomain `sub.example.com`, you enter `sub` in this field.

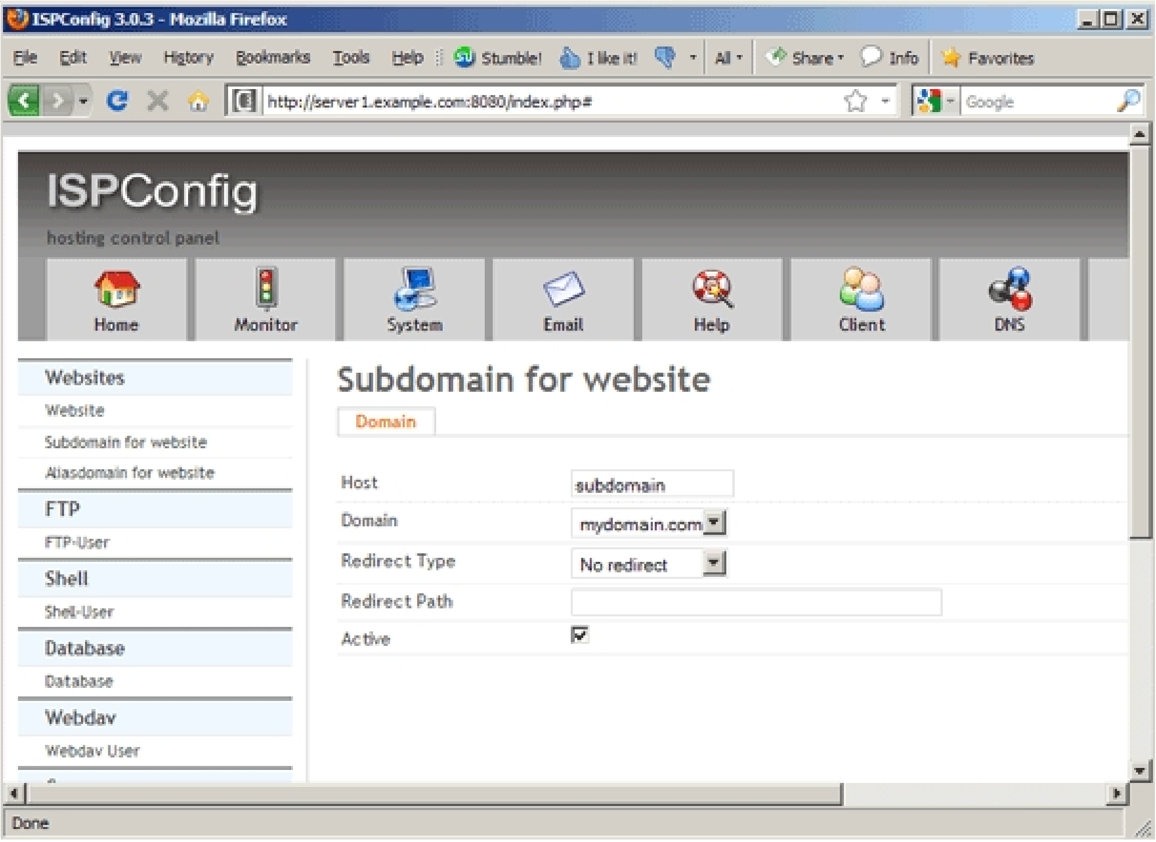
- *Domain*: Here you select the main domain. If you want to create the subdomain *sub.example.com*, this would be *example.com*.
- *Redirect Type*: Here you can specify if you want to disable/enable a redirect, and if decide to use a redirect, which flag to use. (Redirects work exactly as shown for web sites in chapter ***4.6.1.1 Website***.)
  Flags:

  - No flag: Don't use any flags.
  - *R* ***(Apache only)***: Use of the *[R]* flag causes a HTTP redirect to be issued to the browser. If a fully-qualified URL is specified (that is, including *http://servername/*) then a redirect will be issued to that location. Otherwise, the current servername will be used to generate the URL sent with the redirect.
  - *L* ***(Apache only)***: The *[L]* flag causes mod_rewrite to stop processing the rule set. In most contexts, this means that if the rule matches, no further rules will be processed.
  - *R,L* ***(Apache only)***: You will almost always want to use *[R]* in conjunction with *[L]* (that is, use *[R,L]*) because on its own, the *[R]* flag prepends *http://thishost[:thisport]* to the URI, but then passes this on to the next rule in the ruleset, which can often result in 'Invalid URI in request' warnings.
  - *last* ***(nginx only)***: Completes processing of rewrite directives, after which searches for corresponding URI and location.
  - *break* ***(nginx only)***: Completes processing of rewrite directives and breaks location lookup cycle by not doing any location lookup and internal jump at all.
  - *redirect* ***(nginx only)***: Returns temporary redirect with code 302; it is used if the substituting line begins with *http://*.
  - *permanent* ***(nginx only)***: Returns permanent redirect with code 301.
  - *proxy* ***(nginx only)***: This option (which does not refer to an official nginx rewrite flag) allows to proxy requests which allows to display contents from somewhere else without changing the web site URL. In this case only URLs are allowed in the *Redirect Path* field, not paths; the contents is fetched from that URL, and *$request_uri* is appended to that URL. If the proxy option is selected, a further tab called *Options* is added to the form, and it contains the field *Proxy Directives* (please refer to chapter ***4.6.1.1 Website*** to learn how to use this field) where you can add custom nginx proxy directives (directives like, for example, *proxy_set_header*, *proxy_redirect*, *proxy_buffer_size*, etc.) - these will be added to the vhost configuration.

More details about flags can be found here:

- Apache: ***http://httpd.apache.org/docs/2.2/rewrite/flags.html***
- nginx: ***http://wiki.nginx.org/NginxHttpRewriteModule#rewrite***

- *Redirect Path*: This is the target, i.e., the path (full path or path relative to the document root) or URL where the redirect should point to.
- *Active*: This defines if the subdomain is active or not.



Subdomain for website

Domain

Host	subdomain
Domain	mydomain.com
Redirect Type	No redirect
Redirect Path	
Active	☑

Options

This tab is displayed only if you select *proxy* in the *Redirect Type* field on the *Domain* tab.

- Proxy Directives: Please refer to chapter ***4.6.1.1 Website*** to learn how to use this field***.***

4.6.1.3 Subdomain (Vhost)

A Vhostsubdomain is a mixture of a normal web site (vhost) and a subdomain. Basically, it is a subdomain with its own vhost configuration and with a document root that is relative to the parent web site's directory path. Let's assume that the parent web site is called *example.com* and has the base directory */var/www/example.com* and the document root */var/www/example.com/web*. With a Vhostsubdomain, we can now create the subdomain *sub.example.com* that uses */var/www/example.com/somedir* or */var/www/example.com/web/someotherdir* or even the same directory as the parent web site (*/var/www/example.com/web*) as its document root, without any rewrite rules. You can install CMS software in the Vhostsubdomain's document root and don't have to adjust the CMS system's rewrite rules relative to the parent web site's document root, and you can even create an SSL certificate just for the Vhostsubdomain. Of course, processes like PHP run with the permissions of the parent web site to make sure no permission problems arise.

To use this featuere, it has to be activated under *System > Main Config* (see chapter ***4.9.3.1 Main Config***).

Because a Vhostsubdomain is in fact a real vhost, the form to create a Vhostsubdomain is very similar to the one for creating a web site (see chapter ***4.6.1.1 Website***). There are just a few differences:

The *Server*, *Client*, *IPv4-Address*, and *IPv6-Address* fields are missing because they are defined by the parent web site. Instead, we have a *Hostname* field and a *Domain* field that is a drop-down menu instead of a text field. We also have a *Web folder* field:

- *Hostname*: Fill in the subdomain relative to the parent web site. For example, if you want to create the subdomain *sub.example.com*, just fill in *sub* here.
- *Domain*: Select the parent web site here, e.g. *example.com*.
- *Web folder*: Fill in the Vhostsubdomain's document root relative to the parent web site's base path. Let's assume the parent web site's base path is */var/www/example.com*. If you fill in *test* or */test* or */test/* (all these are equivalent), the Vhostsubdomain's document root is */var/www/example.com/test*. To use the same document root as the parent web site, fill in *web* or */web* or */web/*. To use a subdirectory of the parent web site's document root, (e.g. *blog*), use *web/blog* (or */web/blog* or */web/blog/*). This field must not be empty, and the value */* is not allowed.

4.6.1.4 Aliasdomain for website

This is where we can create new and edit/delete existing aliasdomains. With this feature, you can add aliasdomains to an existing web site so that the aliasdomain shows the same content as the web site's main domain. It is also possible to point the aliasdomain to a subdirectory of the web site - this is done using Apache rewrite rules. Please note that you should not use such a rewrite rule if you plan to install a CMS such as Wordpress, Joomla, Drupal, etc. in that subdirectory because most modern CMS systems also use rewrite rules that will most likely collide with the rewrite rules that redirect the aliasdomain to the subdirectory. If you want to install a CMS in a directory of its own and use an aliasdomain for that directory, you should create a whole new web site for that aliasdomain and install the CMS in that web site. But if you plan to place static HTML files in the subdirectory or other stuff that doesn't come with any rewrite rules, you can create an aliasdomain and redirect it to that subdirectory without any problem.

The difference between a subdomain and an aliasdomain is that the subdomain uses the same domain name as the main domain of the web site, whereas an aliasdomain uses a different domain name. For example, if the web site's main domain is *example.com*, and you want to point the hostname *sub.example.com* to the same web site, you'd use a subdomain, whereas if you have a totally different domain such as *yourseconddomain.com* that you want to point to the *example.com* web site, you'd use an aliasdomain.

To create a new aliasdomain, click the *Add new aliasdomain* button. This will lead you to the *Web Aliasdomain* form with the tab *Domain*.

Web Aliasdomain

Domain

Here you can create/edit the aliasdomain. The form has the following fields:

- *`Domain`*: This is where you enter the aliasdomain, e.g. *`yourseconddomain.com`*. It is also possible to specify a subdomain, e.g. *`sub.yourseconddomain.com`*.
- *`Parent Website`*: Here you select the parent web site, i.e.. the web site that the aliasdomain should point to.
- *`Redirect Type`*: Here you can specify if you want to disable/enable a redirect, and if decide to use a redirect, which flag to use. (Redirects work exactly as shown for web sites in chapter ***4.6.1.1 Website***.)

  Flags:

  - *`R`* ***(Apache only)***: Use of the *`[R]`* flag causes a HTTP redirect to be issued to the browser. If a fully-qualified URL is specified (that is, including *`http://servername/`*) then a redirect will be issued to that location. Otherwise, the current servername will be used to generate the URL sent with the redirect.
  - *`L`* ***(Apache only)***: The *`[L]`* flag causes mod_rewrite to stop processing the rule set. In most contexts, this means that if the rule matches, no further rules will be processed.
  - *`R,L`* ***(Apache only)***: You will almost always want to use *`[R]`* in conjunction with *`[L]`* (that is, use *`[R,L]`*) because on its own, the *`[R]`* flag prepends *`http://thishost[:thisport]`* to the URI, but then passes this on to the next rule in the ruleset, which can often result in 'Invalid URI in request' warnings.
  - *`last`* ***(nginx only)***: Completes processing of rewrite directives, after which searches for corresponding URI and location.
  - *`break`* ***(nginx only)***: Completes processing of rewrite directives and breaks location lookup cycle by not doing any location lookup and internal jump at all.
  - *`redirect`* ***(nginx only)***: Returns temporary redirect with code 302; it is used if the substituting line begins with *`http://`*.
  - *`permanent`* ***(nginx only)***: Returns permanent redirect with code 301.
  - *`proxy`* ***(nginx only)***: This option (which does not refer to an official nginx rewrite flag) allows to proxy requests which allows to display contents from somewhere else without changing the web site URL. In this case only URLs are allowed in the *`Redirect Path`* field, not paths; the contents is fetched from that URL, and *`$request_uri`* is appended to that URL. If the proxy option is selected, a further tab called *`Options`* is added to the form, and it contains the field *`Proxy Directives`* (please refer to chapter ***4.6.1.1 Website*** to learn how to use this field) where you can add custom nginx proxy directives (directives like, for example, *`proxy_set_header`*, *`proxy_redirect`*, *`proxy_buffer_size`*, etc.) - these will be added to the vhost configuration.

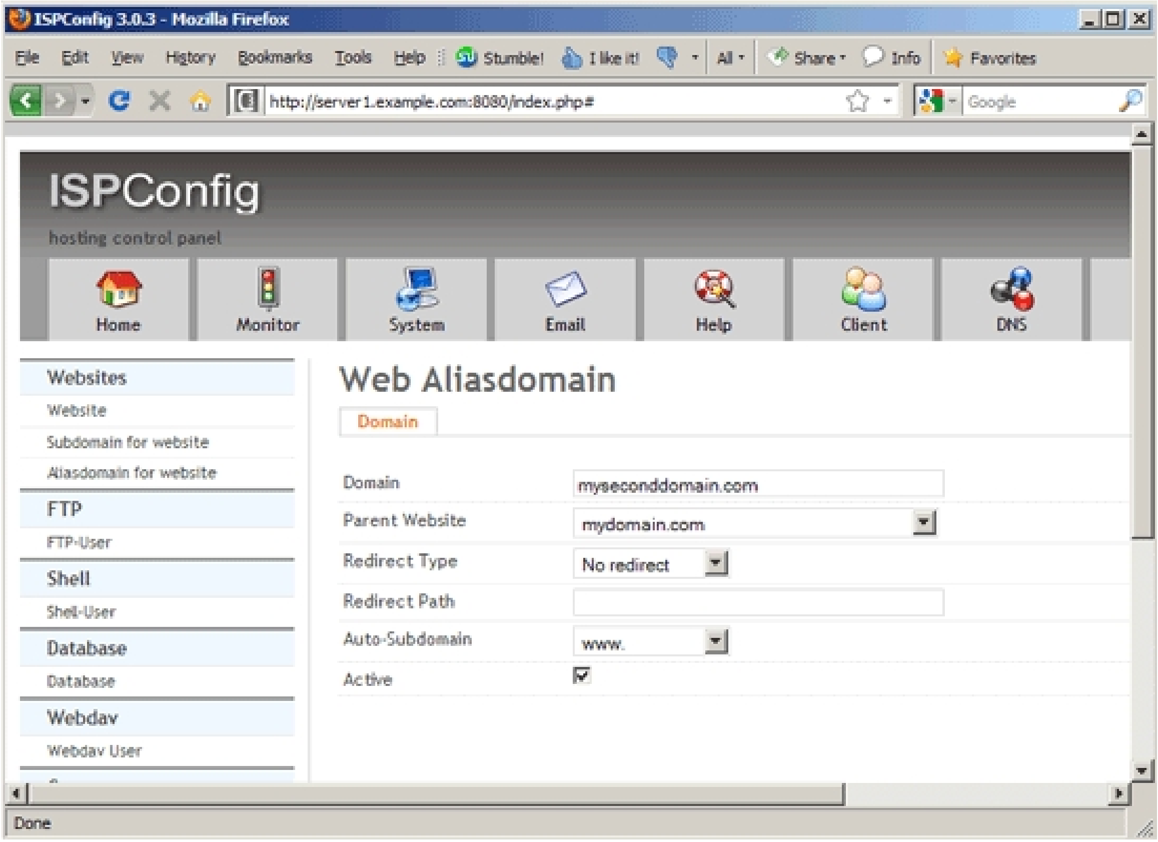
  More details about flags can be found here:

  - Apache: ***http://httpd.apache.org/docs/2.2/rewrite/flags.html***
  - nginx: ***http://wiki.nginx.org/NginxHttpRewriteModule#rewrite***

- *Redirect Path*: This is the target, i.e., the path (full path or path relative to the document root) or URL where the redirect should point to.
- *Auto-Subdomain*: Here you can define whether you want no automatic subdomain for the aliasdomain (in this case you can access the site only by using the domain, e.g. *http://yourseconddomain.com*), an automatic *www* subdomain (you can then access the site using *http://yourseconddomain.com* and *http://www.yourseconddomain.com*), or a wildcard subdomain (**.*) which means you can access the site with any subdomain that does not point to another web site.
- *SEO Redirect*: Here you can do search-engine optimization for your website and configure a redirect to avoid duplicate content. You can redirect your non-www website to your www website (e.g. visitors to *example.com will be redirected permanently to www.example.com*) or vice versa (*www.example.com to example.com*). These are the options:
  - *No Redirect*: Don't use any SEO redirect.
  - *domain.tld => www.domain.tld*: Redirect requests for *example.com* to *www.example.com* .
  - *www.domain.tld => domain.tld*: Redirect requests for *www.example.com* to *example.com*.
  - **.domain.tld => domain.tld*: Redirect all subdomains of *example.com* (including *www.example.com*) to *example.com*.
  - **.domain.tld => www.domain.tld*: Redirect all subdomains (including *example.com* itself) to *www.example.com*.
  - ** => domain.tld*: Redirect everything that is not *example.com* to *example.com* (this includes subdomains and also alias domains).
  - ** => www.domain.tld*:   Redirect everything that is not *www.example.com* to *www.example.com* (this includes *example.com*, subdomains and also alias domains).

Please note that SEO redirects for the parent web site take precedence over SEO redirects for alias domains if they are conflicting. For example, if you have defined the redirect ** => www.domain.tld* for the parent web site, your selection in the SEO Redirect field for the alias domain is ignored.

- *Active*: This defines if the aliasdomain is active or not.



Web Aliasdomain

Domain

Domain	myseconddomain.com
Parent Website	mydomain.com
Redirect Type	No redirect
Redirect Path	
Auto-Subdomain	www.
Active	☑

Options

This tab is displayed only if you select `proxy` in the `Redirect Type` field on the `Domain` tab.

- Proxy Directives: Please refer to chapter ***4.6.1.1 Website*** to learn how to use this field**.**

4.6.2 Database

4.6.2.1 Databases

This is where you can create databases for your web sites. Currently, only MySQL databases are supported.

Databases and database users are now split into two separate forms because this allows you to use one database user for multiple databases, a feature requested by many users.

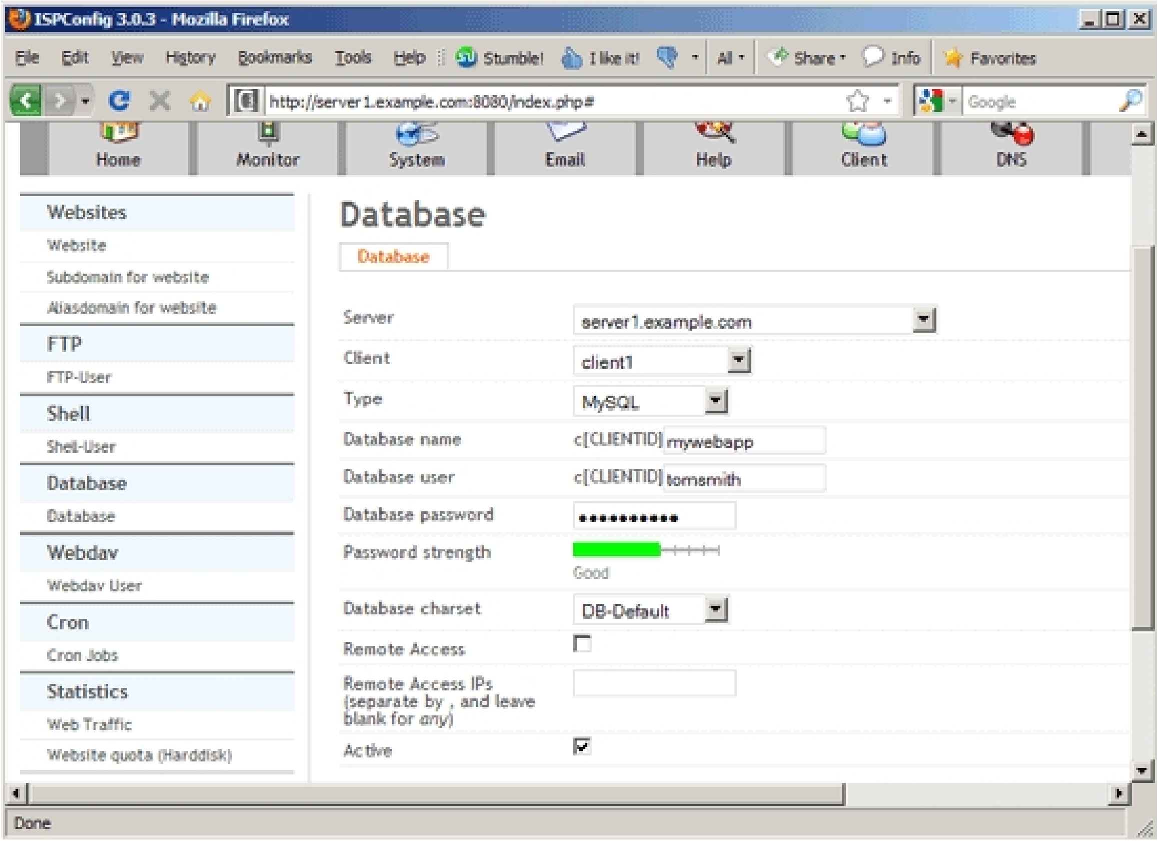
To create a new database, click on the `Add new Database` button. This will lead you to the `Database` form with the tab `Database`.

Database

Database

The form to create/modify a database has the following fields:

- *`Server`*: If more than one server is available, you can select the server on which the database will be created.
- *`Site`*: Select the web site to which this database will belong.
- *`Type`*: Select the database type. Currently only MySQL is supported.
- *`Database name`*: This is the name of the database. The string in square brackets before the database name will be replaced appropriately, for example *`[CLIENTID]`* will be replaced with the ID of the client, i.e., *`1`*, *`2`*, *`3`*, etc. So if the current client is *`client1`*, and you type in *`wordpress`* in the *`Database name`* field, the actual database name will be *`c1wordpress`*. The database name prefix can be defined under *`System > Interface Config`*, however it is not recommended to change the default value. Please note that database names must not be longer than 16 characters - MySQL doesn't support longer database names!
- *`Database user`*: Select the name of the database user. Database users have to be created under *`Sites > Database Users`* (see chapter ***4.6.2.2 Database Users***) first before you can select them here.
- *`Read-only database user`* (optional): In this field you can select another database user that has read-only permissions on the database.
- *`Database charset`*: Select the character set of the database. MySQL includes character set support that enables you to store data using a variety of character sets and perform comparisons according to a variety of collations. You can learn more about ***MySQL's character set support here***.
- *`Remote Access`*: This specifies if the MySQL should allow only local access to the database, or if connections from remote places should be allowed as well (which can be a security risk because intruders don't need access to the local system to connect to the database; all they need is the database username and password).
- *`Remote Access IPs`*: If you've enable remote access and want to allow just a few remote hosts to connect to this database, you can enter the IPs of the remote hosts here. Multiple IPs must be seperated with a comma (*`,`*). To allow connections from all remote hosts, leave this field empty.
- *`Active`*: This defines if this database is active or not.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
Home Monitor System Email Help Client DNS
Websites
Website
Subdomain for website
Aliasdomain for website
FTP
FTP-User
Shell
Shell-User
Database
Database
Webdav
Webdav User
Cron
Cron Jobs
Statistics
Web Traffic
Website quota (Harddisk)
Done

Database

Database

Server	server1.example.com
Client	client1
Type	MySQL
Database name	c[CLIENTID] mywebapp
Database user	c[CLIENTID] tomsmith
Database password	●●●●●●●●●●●
Password strength	Good
Database charset	DB-Default
Remote Access	☐
Remote Access IPs (separate by , and leave blank for any)	
Active	☑

4.6.2.2 Database Users

This is where you create database users. As mentioned before, creating database users independently of the database allows you to use a database user for multiple databases if you wish to do so.

- `Client`: Here you select the client to which the database user belongs.
- `Database user`: This is the name of the database user. The string in square brackets before the database username will be replaced appropriately, for example `[CLIENTID]` will be replaced with the ID of the client, i.e., `1 2`, `3`, etc. So if the current client is `client1`, and you type in `johndoe` in the `Database user` field, the actual database username will be `c1johndoe`. The database user prefix can be defined under `System > Main Config`, however it is not recommended to change the default value. Please do not use underscores (_)in the username.
- `Database password`: Type in a password for the database user (or use the `Generate Password` link to have ISPConfig generate one for you). The Password strength field will show how weak or strong your password is. A strong password should include numbers, symbols, upper and lowercase letters; password length should be 8 characters or more; avoid any password based on repetition, dictionary words, letter or number sequences, usernames, relative or pet names, or biographical information.
- `Repeat Password`: Confirm your password.

4.6.3 Web Access

4.6.3.1 FTP-Accounts

This is where we create new FTP users or modify/delete existing FTP users. FTP users can upload/download/delete files for a website with an FTP client such as ***FileZilla***.

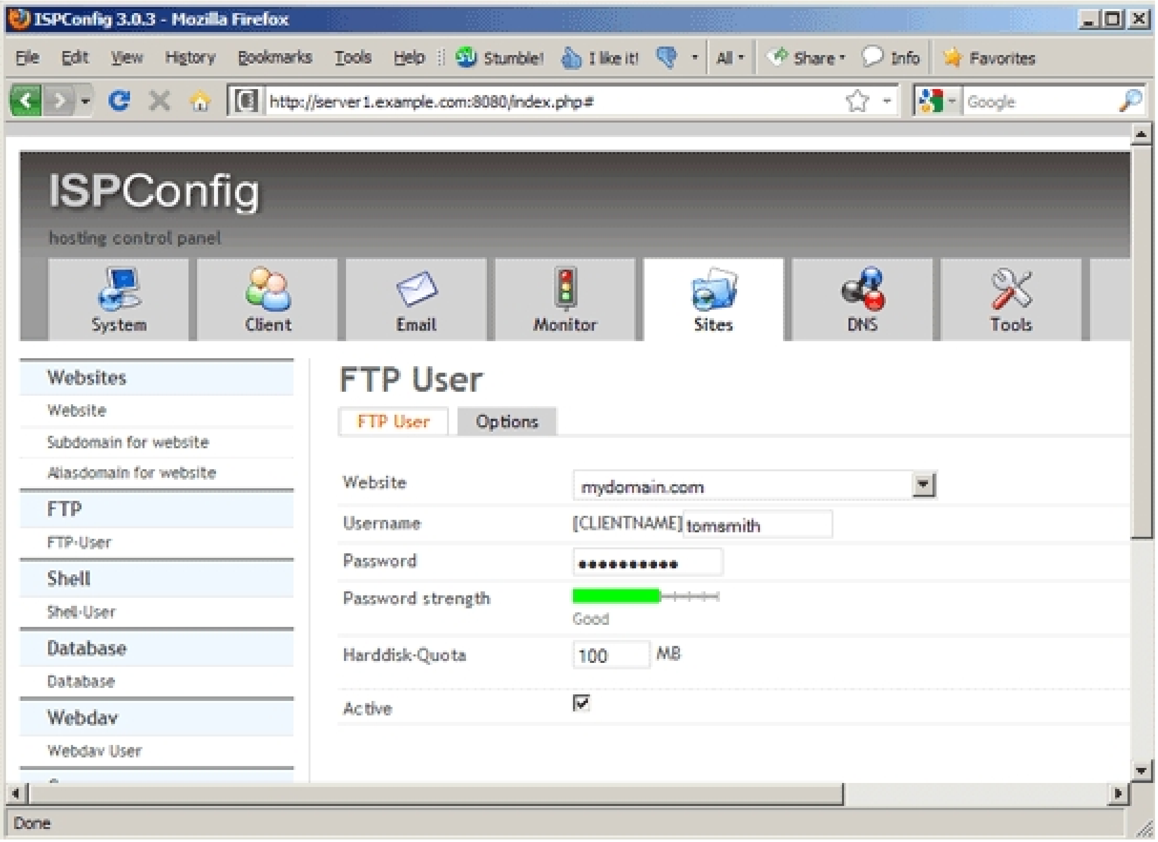
To create a new FTP user, click the *`Add new FTP-User`* button. This will lead you to the *`FTP User`* form with the tabs *`FTP User`* and *`Options`*.

FTP User

FTP User

The form to create/modify an FTP user has the following fields:

- *`Website`*: This is the web site for which you define the FTP user.
- *`Username`*: This is the username of the FTP user. The string in square brackets before the username will be replaced appropriately, for example *`[CLIENT]`* will be replaced with *`client1`*, *`client2`*, etc. So if the current client is *`client1`*, and you type in *`johndoe`* in the *`Username`* field, the actual FTP username will be *`client1johndoe`*. The FTP user prefix can be defined under *`System > Interface Config`*, however it is not recommended to change the default value.
- *`Password`*: Type in a password for the FTP user (or use the *`Generate Password`* link to have ISPConfig generate one for you). The *`Password strength`* field will show how weak or strong your password is. A strong password should include numbers, symbols, upper and lowercase letters; password length should be 8 characters or more; avoid any password based on repetition, dictionary words, letter or number sequences, usernames, relative or pet names, or biographical information.
- *`Repeat Password`*: Confirm your password.
- *`Harddisk-Quota`*: This is the max. amount of disk space (in MB) that is available for the FTP user.
- *`Active`*: This defines if this FTP user account is active or not.



FTP User

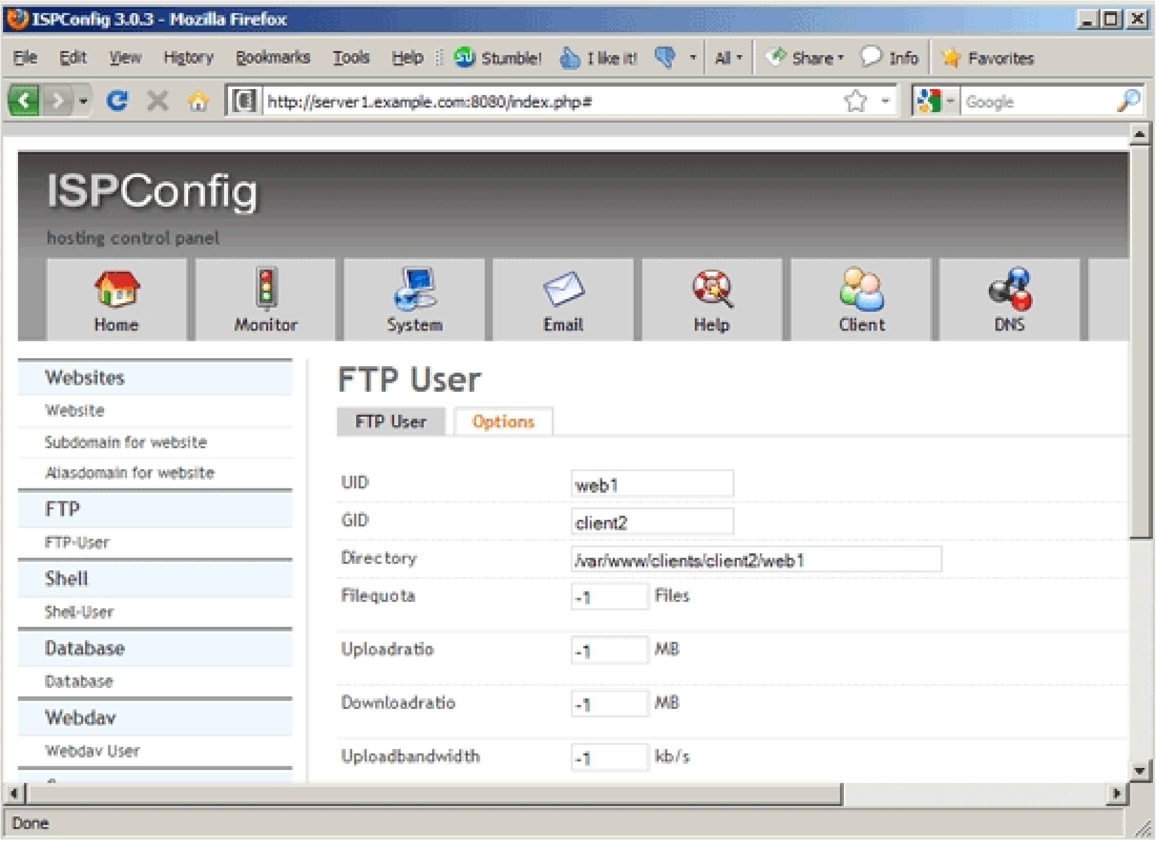
Website	mydomain.com
Username	[CLIENTNAME] tomsmith
Password	•••••••••••
Password strength	Good
Harddisk-Quota	100 MB
Active	☑

Options

On the `Options` tab you can fine-tune the FTP account. The form has the following fields:

- `UID`: The FTP account is a virtual account, i.e., it is no system user, but a user that is stored in a MySQL database. The `UID` field specifies under which system user account the FTP user does uploads and downloads. Normally this should be the same user that is shown in the `Linux User` field on the `Options` tab of the web site.
- `GID`: This is the system group that the (virtual) FTP users uses to do uploads and downloads. Normally this should be the same group that is shown in the `Linux Group` field on the `Options` tab of the web site.
- `Directory`: This is the home directory of the FTP user, i.e., the FTP user can do uploads and downloads in this directory and all subdirectories thereof.
- `Filequota`: This is the amount of files that the FTP user is allowed to upload. `-1` means unlimited.
- `Uploadratio`: This defines the upload ratio in MB. `-1` means unlimited.
- `Downloadratio`: This defines the download ratio in MB. `-1` means unlimited.
- `Uploadbandwidth`: This defines the bandwidth with which the FTP user can upload files (in kb/s). `-1` means unlimited.

- `Downloadbandwidth`: This defines the bandwidth with which the FTP user can download files (in kb/s). `-1` means unlimited.



FTP User

UID	web1	
GID	client2	
Directory	/var/www/clients/client2/web1	
Filequota	-1	Files
Uploadratio	-1	MB
Downloadratio	-1	MB
Uploadbandwidth	-1	kb/s

4.6.3.2 WebDAV-Users

WebDAV stands for ***Web-based Distributed Authoring and Versioning*** and is a set of extensions to the HTTP protocol that allow users to directly edit files on the Apache server so that they do not need to be downloaded/uploaded via FTP. Of course, WebDAV can also be used to upload and download files.

To create a new WebDAV user, click the `Add new WebDAV-User` button. This will lead you to the `WebDAV User` form with the tab `WebDAV User`.

WebDAV User

WebDAV User

The form to create/modify a WebDAV user has the following fields:

- `Website`: This is the web site for which you define the WebDAV user.

- *`Username`*: This is the username of the WebDAV user. The string in square brackets before the username will be replaced appropriately, for example *`[CLIENT]`* will be replaced with *`client1`*, *`client2`*, etc. So if the current client is *`client1`*, and you type in *`johndoe`* in the *`Username`* field, the actual WebDAV username will be *`client1johndoe`*. The WebDAV user prefix can be defined under *`System > Interface Config`*, however it is not recommended to change the default value.

- *`Password`*: Type in a password for the WebDAV user (or use the *`Generate Password`* link to have ISPConfig generate one for you). The *`Password strength`* field will show how weak or strong your password is. A strong password should include numbers, symbols, upper and lowercase letters; password length should be 8 characters or more; avoid any password based on repetition, dictionary words, letter or number sequences, usernames, relative or pet names, or biographical information.

- *`Repeat Password`*: Confirm your password.

- *`Active`*: This defines if this WebDAV user account is active or not.

- *`Directory`*: This defines the subdirectory of your document root that you want to access with WebDAV. If you leave it empty, you can access the whole document root and its subdirectories with the WebDAV URL *`http://example.com:80/webdav`*. If you type in a subdirectory, e.g. *`images`*, you can access the images subdirectory as follows: *`http://example.com:80/webdav/images`*.

This link explains how you can access a WebDAV share from a Windows PC: ***Configure A Windows XP Client To Connect To The WebDAV Share***

This link shows how you can access a WebDAV share from a Linux desktop (GNOME): ***Configure A Linux Client (GNOME) To Connect To The WebDAV Share***

4.6.3.3 Protected Folders

This is where we can password-protect directories inside websites (basic http authentication with *`.htaccess`*/ *`.htpasswd`*).

To password-protect a website folder, click the *`Add new record`* button. This will lead you to the *`Web Folder`* form with the tab *`Folder`*.

Web Folder

Protected Folders

In this form we select a website for which we want password protection, and then we specify the directory inside this website that will be password-protected. The form has the following fields:

- *`Website`*: Select the website in which you want to password-protect a folder.

- *`Path`*: Specify the folder relative to the website's document root that you want to password-protect, e.g.

/files if you want to protect the directory *files* in the website's document root, or */files/secret* to password-protect the directory *files/secret*. You can also password-protect the whole website by specifying */*. If the specified directory does not exist, it will be created by ISPConfig.

- *Active*: Defines whether this folder protection is active or not.

4.6.3.4 Protected Folder Users

Here we specify the users that are allowed to log into a password-protected wbesite directory.

To create a new user, click the *Add new record* button. This will lead you to the *Web folder user* form with the tab *Folder*.

Web folder user

Folder

Here you can create/edit a user. The form has the following fields:

- *Folder*: In this drop-down menu you can select the folder for which you want to create the user. This drop-down menu contains all active folder that were previously created under Folder (see chapter).
- *Username*: Specify the username.
- *Password*: Specify the user's password (or use the *Generate Password* link to have ISPConfig generate one for you).
- *Repeat Password*: Confirm your password.
- *Active*: This defines if the user is active or not.

4.6.4 Command Line

4.6.4.1 Shell-User

This is where we create new shell users (i.e., system users) or modify/delete existing shell users. Shell users can log into the system via SSH (e.g. by using an SSH client such as ***PuTTY***) and do secure uploads/downloads by using an SCP client (such as ***WinSCP***).

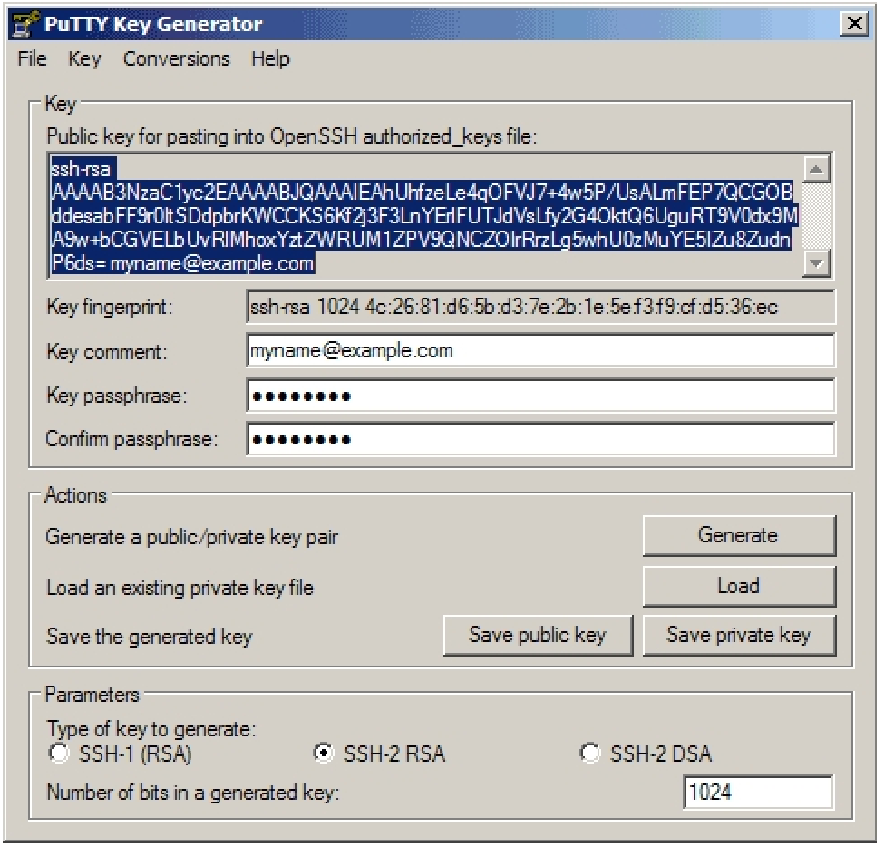
To create a new shell user, click the *Add new Shell-User* button. This will lead you to the *Shell User* form with the tabs *Shell User* and *Options*.

Shell User

Shell User

The form to create/modify a shell user has the following fields:

- `Site`: This is the web site for which you define the shell user.
- `Username`: This is the username of the shell user. The string in square brackets before the username will be replaced appropriately, for example `[CLIENT]` will be replaced with `client1`, `client2`, etc. So if the current client is `client1`, and you type in `johndoe` in the `Username` field, the actual shell username will be `client1johndoe`. The shell user prefix can be defined under `System > Interface Config`, however it is not recommended to change the default value.
- `Password`: Type in a password for the shell user (or use the `Generate Password` link to have ISPConfig generate one for you). The `Password strength` field will show how weak or strong your password is. A strong password should include numbers, symbols, upper and lowercase letters; password length should be 8 characters or more; avoid any password based on repetition, dictionary words, letter or number sequences, usernames, relative or pet names, or biographical information.
- `Repeat Password`: Confirm your password.
- `Chroot Shell`: This defines if this shell user is chrooted or not. If you select `None`, the shell user can browse the whole file system and is limited only by file/directory permissions - this can be a security risk. If you select to chroot the shell user (by selecting `Jalikit` from the drop-down menu), the shell user will be limited to his home directory and can only browse directories inside his home directory.
- `Quota`: This is the max. amount of disk space (in MB) that is available for the shell user.
- `SSH-RSA Public Key (for key-based logins)`: This field allows you to put in one or more public SSH-RSA keys. With such a key you can log into the system without having to provide a password. You can find out more about key-based SSH logins in this tutorial: ***Key-Based SSH Logins With PuTTY***. Generating an SSH-RSA key is described in chapter 5 of that tutorial:



Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAABJQAAAIEAhUhfzeLe4qOFVJ7+4w5P/UsALmFEP7QCGOB
ddesabFF9r0ltSDdpbrKWCCKS6Kf2j3F3LnYErlFUTJdVsLfy2G4OktQ6UguRT9V0dx9M
A9w+bCGVELbUvRIMhoxYztZWRUM1ZPV9QNCZOIrRzLg5whU0zMuYE5IZu8Zudn
P6ds= myname@example.com
```

Key fingerprint: ssh-rsa 1024 4c:26:81:d6:5b:d3:7e:2b:1e:5e:f3:f9:cf:d5:36:ec

Key comment: myname@example.com

Key passphrase: ●●●●●●●●●

Confirm passphrase: ●●●●●●●●●

Generate a public/private key pair

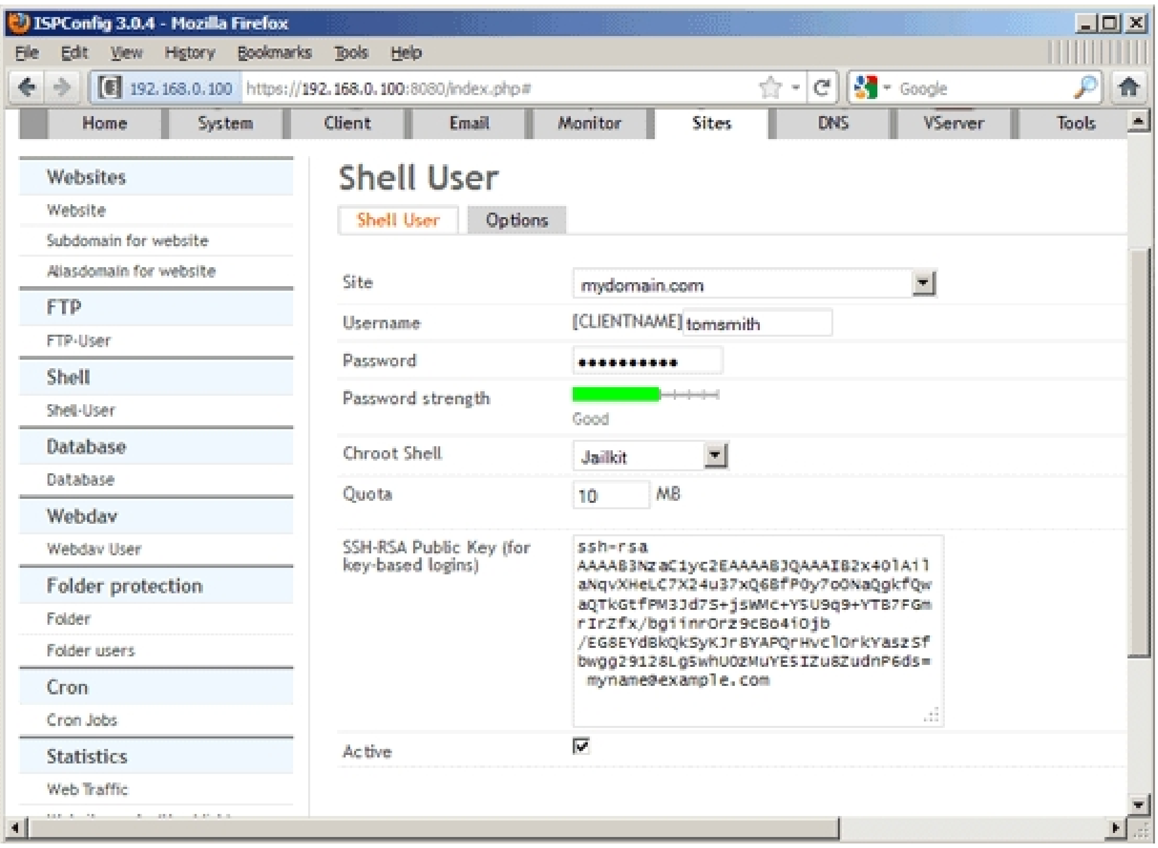
Load an existing private key file

Save the generated key

Type of key to generate:
○ SSH-1 (RSA) ● SSH-2 RSA ○ SSH-2 DSA

Number of bits in a generated key: 1024

- *Active*: This defines if this shell user account is active or not.

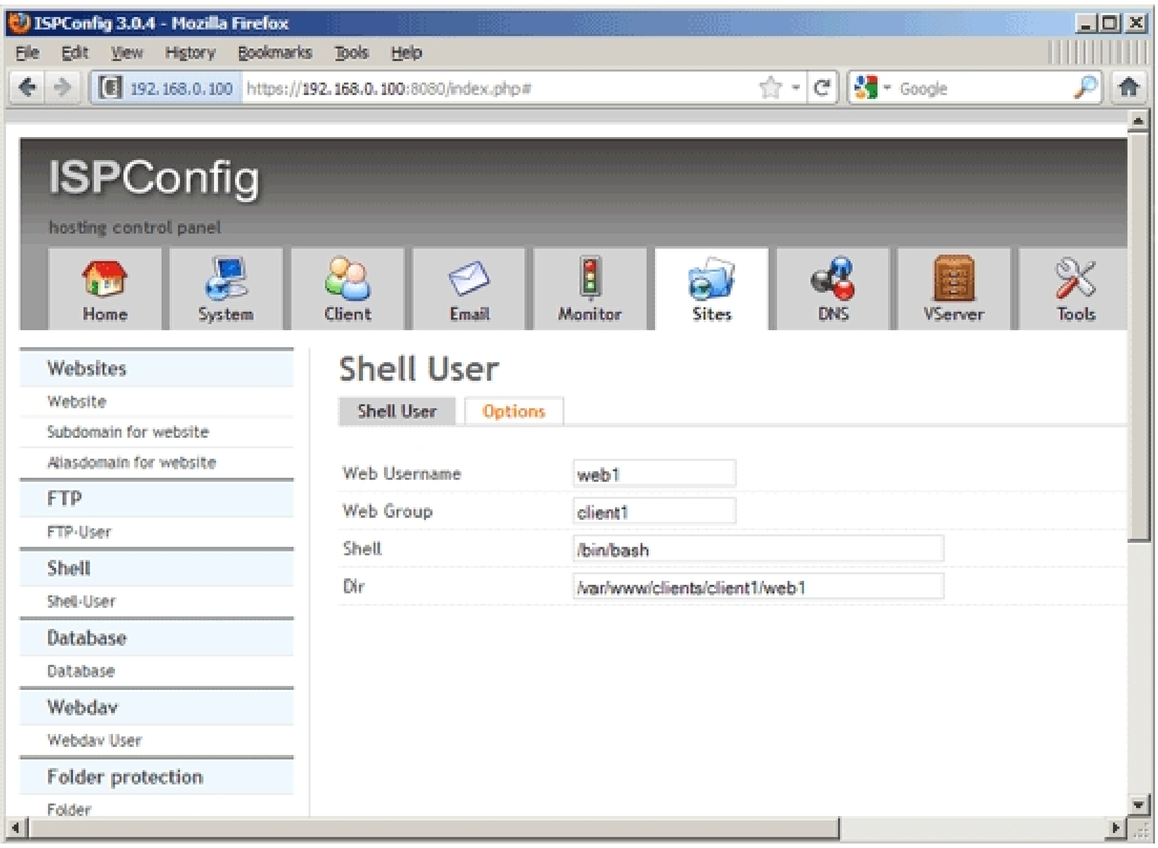


Site	mydomain.com
Username	[CLIENTNAME] tomsmith
Password	●●●●●●●●●●●
Password strength	Good
Chroot Shell	Jailkit
Quota	10 MB
SSH-RSA Public Key (for key-based logins)	ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAIB2x401Ai1aNqvXHeLC7X24u37xQ6BfP0y7oONaQgkfQwaQTkGtfPM3Jd7S+jSWMc+YSU9q9+YTB7FGmrIrZfx/bgiinrOrz9cBo4iOjb/EG8EYdBkQkSyKJr8YAPQrHvclOrkYaszSfbwgg29128LgSwhUOzMuYESIZu8ZudnP6ds= myname@example.com
Active	☑

Options

On the `Options` tab you can fine-tune the shell user account. The form has the following fields:

- *`Web Username`*: The shell user account is a "virtual" account. The *`UID`* field specifies to which system user this virtual account is mapped. Normally this should be the same user that is shown in the *`Linux User`* field on the *`Options`* tab of the web site.
- *`Web Group`*: This is the system group that the (virtual) shell user is mapped to. Normally this should be the same group that is shown in the *`Linux Group`* field on the *`Options`* tab of the web site.
- *`Shell`*: This is the shell that the user uses to log in. Possible values are, for example: *`/bin/bash`* or *`/bin/sh`*. It's also possible to give a shell user a shell that doesn't allow him to log in, sich as *`/bin/false`* or *`/usr/sbin/nologin`*.
- *`Dir`*: This is the home directory of the shell user. If you have chrooted the shell user, he cannot break out of this directory.



Web Username: web1

Web Group: client1

Shell: /bin/bash

Dir: /var/www/clients/client1/web1

4.6.4.2 Cron Jobs

A cron job is a scheduled task that is executed by the system at a specified time/date.

To create a new cron job, click on the `Add new Cron job` button. This will lead you to the `Cron Job` form with the tab `Cron Job`.

Cron Job

Cron Job

The form to create/modify a cron job has the following fields:

- `Parent website`: This is the web site for which you define the cron job.
- `Minutes`: The minute to run the cron job. Allowed values: `0-59`. `*` means every minute.
- `Hours`: The hour to run the cron job. Allowed values: `0-23`. `*` means every hour.
- `Days of month`: The day of the month to run the cron job. Allowed values: `1-31`. `*` means every day of the month.
- `Months`: The month to run the cron job. Allowed values: `1-12` (or names, see ***below***). `*` means every

month.

- `Days of week`: The day of the week to run the cron job. Allowed values: `0-7` (`0` or `7` is `Sun`, or use names). `*` means every day of the week.
- `Command to run`: This is the command to execute. Shell scripts will be run by `/bin/sh`, URLs will be executed by `wget`.
- `Active`: This defines if the cron job is active or not.

When specifying day of week, both day `0` and day `7` will be considered Sunday.

A field may be an asterisk (`*`), which always stands for first-last.

Names can also be used for the "month" and "day of week" fields. Use the first three letters of the particular day or month (case doesn't matter), e.g. `sun` or `SUN` for Sunday or `mar`/`MAR` for March..

Let's take a look at two sample cron jobs:

```
* * * * * /usr/local/ispconfig/server/server.sh > /dev/null 2>>
/var/log/ispconfig/cron.log
```

This means: execute `/usr/local/ispconfig/server/server.sh > /dev/null 2>> /var/log/ispconfig/cron.log` once per minute.

```
30 00 * * * /usr/local/ispconfig/server/cron_daily.sh > /dev/null 2>>
/var/log/ispconfig/cron.log
```

This means: execute `/usr/local/ispconfig/server/cron_daily.sh > /dev/null 2>> /var/log/ispconfig/cron.log` once per day at 00:30h.

The day of a command's execution can be specified by two fields: day of month, and day of week. If both fields are restricted (i.e., aren't `*`), the command will be run when either field matches the current time. For example, `30 4 1,15 * 5` would cause a command to be run at 4:30h on the 1st and 15th of each month, plus every Friday.

You can use ranges to define cron jobs:

Examples:

`1,2,5,9` - means every first, second, fifth, and ninth (minute, hour, month, ...).

`0-4,8-12` - means all (minutes, hours, months,...) from 0 to 4 and from 8 to 12.

`*/5` - means every fifth (minute, hour, month, ...).

`1-9/2` is the same as `1,3,5,7,9`.

Ranges or lists of names are not allowed (if you are using names instead of numbers for months and days - e.g., `Mon-Wed` is not valid).

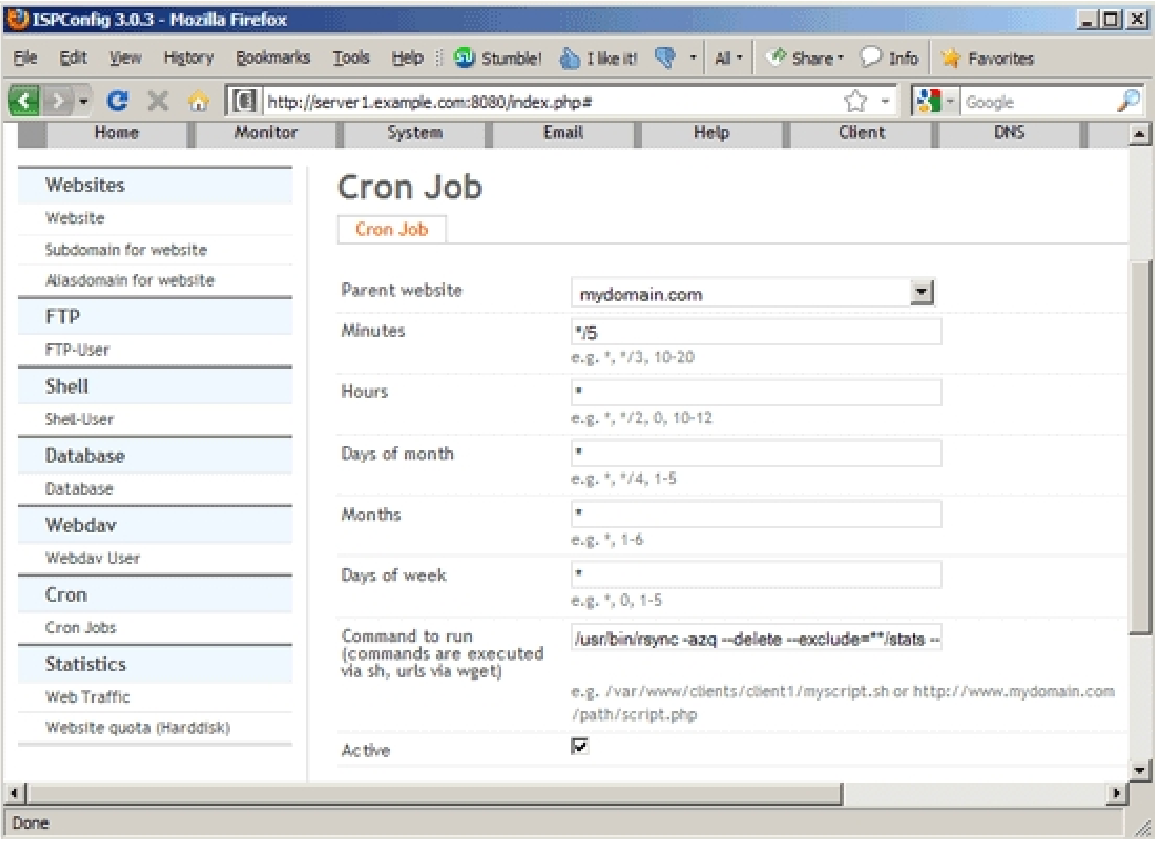
```
1,7,25,47 */2 * * * command
```

means: run command every second hour in the first, seventh, 25th, and 47th minute.

Instead of the first five fields, one of eight special strings may appear:

```
string         meaning
  ------         -------
  @reboot        Run once, at startup.
  @yearly        Run once a year, "0 0 1 1 *".
  @annually      (same as @yearly)
  @monthly       Run once a month, "0 0 1 * *".
  @weekly        Run once a week, "0 0 * * 0".
  @daily         Run once a day, "0 0 * * *".
  @midnight      (same as @daily)
  @hourly        Run once an hour, "0 * * * *".
```

You can learn more about cron jobs here: ***A Short Introduction To Cron Jobs***.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
Home Monitor System Email Help Client DNS
Websites
Website
Subdomain for website
Aliasdomain for website
FTP
FTP-User
Shell
Shell-User
Database
Database
Webdav
Webdav User
Cron
Cron Jobs
Statistics
Web Traffic
Website quota (Harddisk)
Done

Cron Job

Cron Job

Parent website	mydomain.com
Minutes	*/5 e.g. *, */3, 10-20
Hours	* e.g. *, */2, 0, 10-12
Days of month	* e.g. *, */4, 1-5
Months	* e.g. *, 1-6
Days of week	* e.g. *, 0, 1-5
Command to run (commands are executed via sh, urls via wget)	/usr/bin/rsync -azq --delete --exclude=**/stats -- e.g. /var/www/clients/client1/myscript.sh or http://www.mydomain.com/path/script.php
Active	☑

4.6.5 APS Installer

The APS installer is a one-click installer for web applications. It follows the Application Packaging Standard (see ***http://www.apsstandard.org/***) and offers the most common web applications for easy installation into any web site in ISPConfig.

4.6.5.1 Available packages

Here you get a list of all available web applications. Please note that you might have to click on the `Update Packagelist` link (see chapter ***4.6.5.3 Update Packagelist***) first to populate or update the list.

To install a package, simply click on its link. This will bring you to a page with more details about the package (split up into the tabs `Details`, `Screenshots`, `Changelog`, and `Settings` - the `Settings` tab is important because it lists the requirements - such as PHP settings and required PHP and MySQL versions - for running the web application, so before you install an application, make sure the web site where you want to install it fulfills these requirements).

Click on the `Install this package` button to start the install process. This will bring you to a form with a few fields that depend on the application (such as username and password for the web application's backend, blog title, languae, license, etc.). One field is common for all applications:

- `Install location`: Select the web site where the application will be installed, and specify a subdirectory if you want to install the application in a subdirectory. To install it directly in the document root, just leave the subdirectory field empty.

4.6.5.2 Installed packages

This is a list of all installed packages. The list displays the name of the application, its version, its installation location, and its installation status (success, error, removed). By clicking on the name of the application or the name of the client who installed it, you will be redirected to the package details page (split up into the tabs `Details`, `Screenshots`, `Changelog`, and `Settings`). If you click on the Install location link, a new browser window opens with the install location URL.

In the last column of the list you find one or two icons, a no entry sign and a srew wrench, depending on if the application can be reinstalled or not. Clicking on the no entry sign allows you to delete the web application from the web site, while clicking on the screw wrench allows you to reinstall the web application with the same settings as the original installation.

4.6.5.3 Update Packagelist

Under `Update Packagelist` you can reload the local database of available packages. This will take you to a page with an `Update Packagelist` button. Once you click that button the operation starts in the background and can take a few minutes, that's why no output is displayed. It is ok to leave the page while the task is being processed, it will simply continue in the background.

4.6.6 Statistics

The `Statistics` section is a bit special in that there's nothing that you can configure here. This section just displays statistics for your web sites.

4.6.6.1 Web traffic

Under *`Web traffic`* you can see traffic statistics (in MB) for your web sites for the current month, the month before, the current year, and the year before.

These statistics are realtime (updated once per minute).

4.6.6.2 Website quota (Harddisk)

Under *`Website quota (Harddisk)`* you can see the hard disk usage (*`Used Space`*, in MB) for your web sites, as well as the current quota soft limits and hard limits.

Soft limit indicates the maximum amount of disk usage a quota user has on a partition. When combined with "grace period", it acts as the border line, which a quota user is issued warnings about his impending quota violation when passed. Hard limit works only when "grace period" is set. It specifies the absolute limit on the disk usage, which a quota user can't go beyond his "hard limit".

These statistics are near realtime (updated every five minutes).

4.7 Email

On this tab we can create email accounts, define email forwards and spamfilter settings, configure the system to fetch mail from remote POP3 and/or IMAP servers, set up content filters and black- and whitelists, etc.

4.7.1 Email Accounts

4.7.1.1 Domain

Here we can define the domains for which we want to set up email accounts later on.

To create a new email domain, click on the *`Add new Domain`* button. This will lead you to the *`Mail Domain`* form with the tab *`Domain`*.

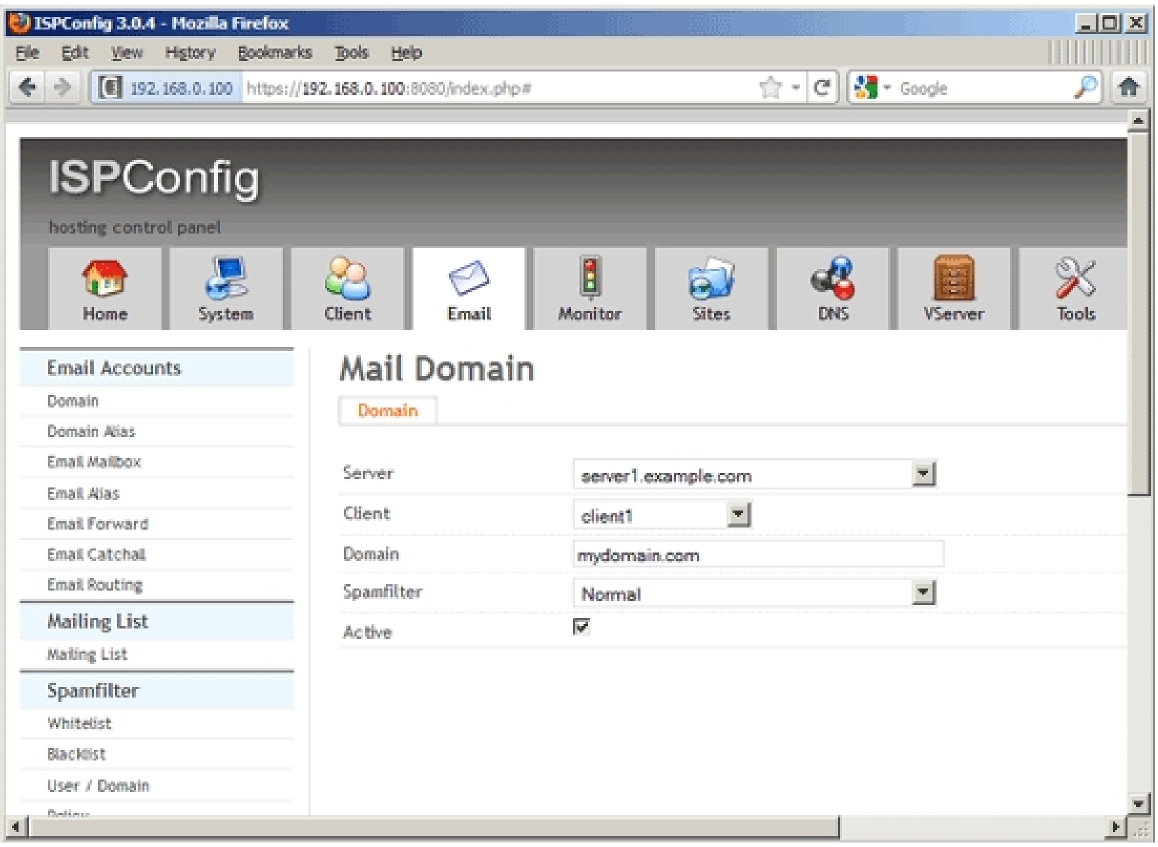
Mail Domain

Domain

This form contains the following fields:

- *`Server`*: If more than one server is available, you can select the server on which the email domain will be located. It is possible that the email domain is located on another server than the web site domain.

- `Client`: Here you select the client that owns the email domain.
- `Domain`: Type in the email domain, e.g. `example.com` (this would lead to email addresses such as `user@example.com`). It is also possible to fill in subdomains, e.g. `sub.example.com`, which would result in email addresses such as `user@sub.example.com`.
- `Spamfilter`: Here you can specify if you want to enable the spamfilter for this domain, and if so, what spamfilter level to use: `Non-Paying`, `Uncensored`, `Wants all spam`, `Wants viruses`, `Normal`, `Trigger happy`, `Permissive`. The settings for each of these levels are defined under `Email > Spamfilter > Policy`.
- `Active`: This defines whether this email domain is active or not.



Mail Domain

Domain

Server: server1.example.com

Client: client1

Domain: mydomain.com

Spamfilter: Normal

Active: ☑

4.7.1.2 Domain Alias

With domain aliases, you can map one email domain to another one. Let's assume you have created the email domains `example.com` and `yourseconddomain.com`, and have also created the email accounts `user1@example.com` and `user2@example.com`. Now you want to use the exact same mail boxes for `yourseconddomain.com` as well, i.e., `user1@example.com` and `user1@yourseconddomain.com` as well as `user2@example.com` and `user2@yourseconddomain.com` should be identical mail boxes. This can be achieved by mapping *yourseconddomain.com* to *example.com* - it can be imagined as a kind of symlink from *yourseconddomain.com* to *example.com*.

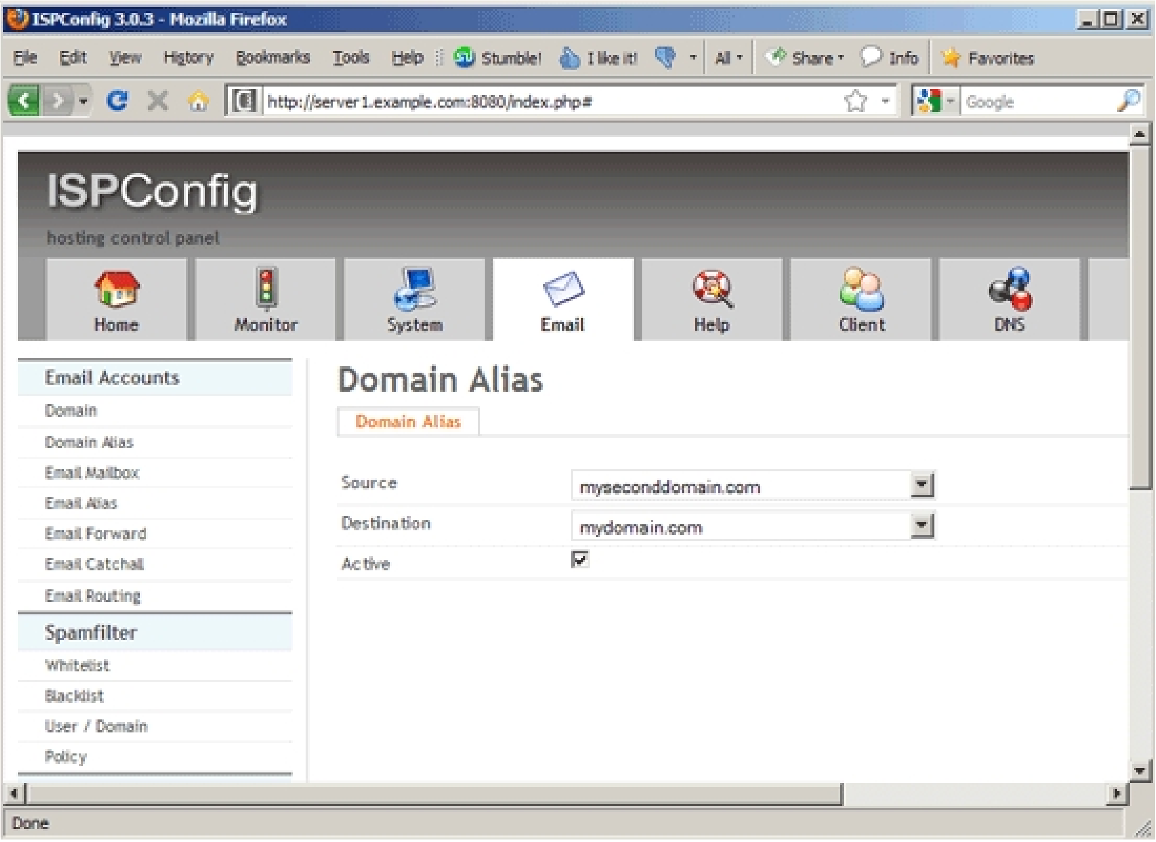
To create a new domain alias, click on the *Add new Domain alias* button. This will lead you to the *Domain Alias* form with the tab *Domain Alias*.

Domain Alias

Domain Alias

This form has the following fields:

- *Source*: This is the domain that you want to map to another email domain. In our above example, this would be *yourseconddomain.com*.
- *Destination*: This is the email domain that the source domain should be mapped to. In our above example, this would be *example.com*.
- *Active*: This defines whether this domain alias is active or not.



Source: myseconddomain.com
Destination: mydomain.com
Active: ☑

4.7.1.3 Email Mailbox

This is where we create/modify/delete email accounts.

To create a new email account, click on the `Add new Mailbox` button. This will lead you to the `Mailbox` form with the tabs `Mailbox`, `Autoresponder`, `Mail Filter`, and `Custom Rules`.

Mailbox

Mailbox

This form has the following fields:

- `Realname`: Type in the real name of the email user, e.g. `John Doe`. This field is optional.
- `Email`: This specification is split up in two fields, `Alias` and `Domain`. `Alias` contains the part in front of the `@` sign (the "local part"), and in the `Domain` drop-down menu, you select the email domain. For example, if you want to create the email account `john.doe@example.com`, you'd fill in `john.doe` in the `Alias` field and select `example.com` from the `Domain` drop-down menu. The email address is also the SMTP/POP3/IMAP username for the email account.

The local-part of an e-mail address may be up to 64 characters long and the domain name may have a maximum of 255 characters. However, the maximum length of a forward or reverse path length of 256 characters restricts the entire e-mail address to be no more than 254 characters. Some mail protocols, such as X.400, may require larger objects, however. The SMTP specification recommends that software implementations impose no limits for the lengths of such objects.

The local-part of the e-mail address may use any of these ASCII characters:

* Uppercase and lowercase English letters (`a&#8211;z`, `A&#8211;Z`)
* Digits `0` to `9`
* Characters `! # $ % & ' * + - / = ? ^ _ ` { | } ~`
* Character `.` (dot, period, full stop) provided that it is not the first or last character, and provided also that it does not appear two or more times consecutively (e.g. `John..Doe@example.com`).

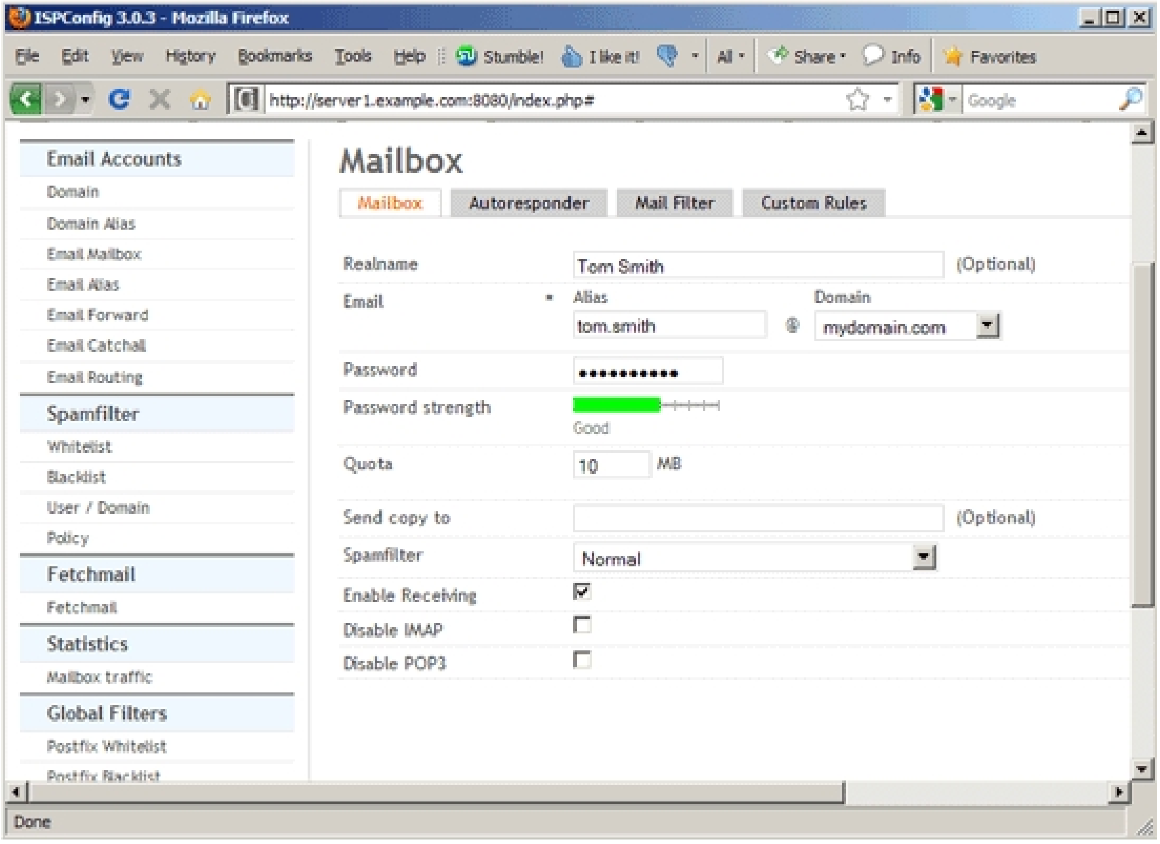
Additionally, quoted-strings (e.g. `"John Doe"@example.com`) are permitted, thus allowing characters that would otherwise be prohibited, however they do not appear in common practice. RFC 5321 also warns that "a host that expects to receive mail SHOULD avoid defining mailboxes where the Local-part requires (or uses) the Quoted-string form".

The local-part is case sensitive, so `"jsmith@example.com"` and `"JSmith@example.com"` may be delivered to different people. This practice is discouraged by RFC 5321. However, only the authoritative mail servers for a domain may make that decision (if you have set up your server according to one of the "Perfect Server" tutorials from HowtoForge.com, then the local part is ***not case sensitive***). The only exception is for a local-part value of "postmaster" which is case insensitive, and should be forwarded to the server's administrator.

Within the rules set out in the RFCs, organisations are free to restrict the forms their own e-mail addresses take however they wish, e.g. many organizations do not use certain characters, e.g. space, `?`, and `^`, and most organizations treat uppercase and lowercase letters as equivalent. Hotmail, for example, only allows creation of e-mail addresses using alphanumerics, dot (`.`), underscore (_) and hyphen (`-`).

Systems that send mail, of course, must be capable of handling outgoing mail for all addresses. Contrary to the relevant standards, some defective systems treat certain legitimate addresses as invalid and fail to handle mail to these addresses. Hotmail, for example, incorrectly refuses to send mail to any address containing any of the following legitimate characters: *`! # $ % * / ? ^ ` { | } ~`*

- *`Password`*: Type in a password for the email account (or use the *`Generate Password`* link to have ISPConfig generate one for you). The *`Password strength`* field will show how weak or strong your password is. A strong password should include numbers, symbols, upper and lowercase letters; password length should be 8 characters or more; avoid any password based on repetition, dictionary words, letter or number sequences, usernames, relative or pet names, or biographical information.
- Repeat Password: Confirm your password.
- *`Quota`*: This is the max. amount of disk space (in MB) that is available for this email account.
- *`Send copy to`*: Here you can specify an email address that should receive a copy of all incoming mails for this email account. This field is optional.
- *`Spamfilter`*: Here you can specify if you want to enable the spamfilter for this email account, and if so, what spamfilter level to use: *`Non-Paying`*, *`Uncensored`*, *`Wants all spam`*, *`Wants viruses`*, *`Normal`*, *`Trigger happy`*, *`Permissive`*. The settings for each of these levels are defined under *`Email > Spamfilter > Policy`*. Please note that this setting overrides the spamfilter setting of the mail domain (no matter what spamfilter level you chose for the mail domain; this is true even if you disabled the spamfilter for the mail domain), with one exception: If you choose to not enable the spamfilter for this email account, but the spamfilter is enabled for the mail domain, then the spamfilter setting of the mail domain is used for this email account. Use *`Uncensored`* to disable the spamfilter.
- *`Enable Receiving`*: If you don't check this box, then incoming emails for this mail account will be rejected. This makes sense if you want to use this account only for sending mail, but not for receiving.
- *`Disable IMAP`*: If you check this box, you cannot use IMAP to access the mails of this mailbox.
- *`Disable POP3`*: If you check this box, you cannot use POP3 to access the mails of this mailbox.



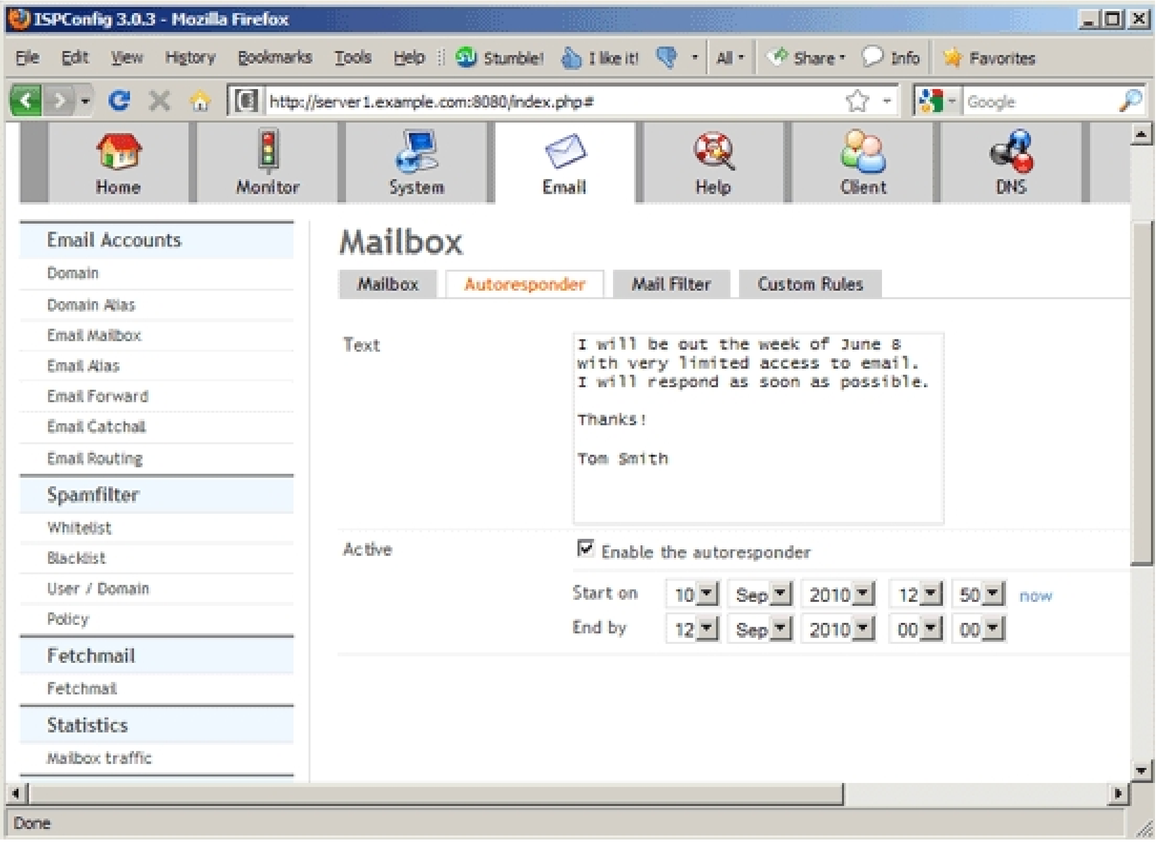
Realname	Tom Smith (Optional)
Email	Alias: tom.smith @ Domain: mydomain.com
Password	●●●●●●●●●●●
Password strength	Good
Quota	10 MB
Send copy to	(Optional)
Spamfilter	Normal
Enable Receiving	☑
Disable IMAP	☐
Disable POP3	☐

Autoresponder

With the autoresponder you have the possibility to automatically send replies to incoming mails (e.g. if you are on holidays).

The form has the following fields:

- Email Subject: Specify the subject to be used for autoresponder messages.
- *`Text`*: Enter your autoresponder message in this field.
- *`Enable the autoresponder`*: This defines whether this autoresponder is currently active or not.
- *`Start on`*: Here you can define when the autoresponder should start (day - month - year - hour - minute). If you don't specify a start date, the autoresponder becomes active immediately. If you click on *now*, ISPConfig will fill in the current start date, and the end date will be the end of the next day.
- *`End by`*: Here you can define when the autoresponder should stop (day - month - year - hour - minute). If you don't specify an end date, the autoresponder will be active forever.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Home Monitor System Email Help Client DNS
Email Accounts
Domain
Domain Alias
Email Mailbox
Email Alias
Email Forward
Email Catchall
Email Routing
Spamfilter
Whitelist
Blacklist
User / Domain
Policy
Fetchmail
Fetchmail
Statistics
Mailbox traffic
Mailbox
Mailbox Autoresponder Mail Filter Custom Rules
now
Done

Text: I will be out the week of June 8 with very limited access to email. I will respond as soon as possible.

Thanks!

Tom Smith

Active: ☑ Enable the autoresponder

Start on: 10 Sep 2010 12 50

End by: 12 Sep 2010 00 00

Mail Filter

On this tab you can define filters for incoming emails. One common filter has already been defined for you:

- *`Move Spam Emails to Junk directory`*: If you check this, emails that are tagged as spam by the spamfilter will automatically be moved to the junk folder. Please note that you can access the junk folder only if you use IMAP. This filter is active only if the spamfilter is active for this email account (i.e., a spamfilter level other than *`Uncensored`* must be selected, either for the whole mail domain or specifically for this email account).

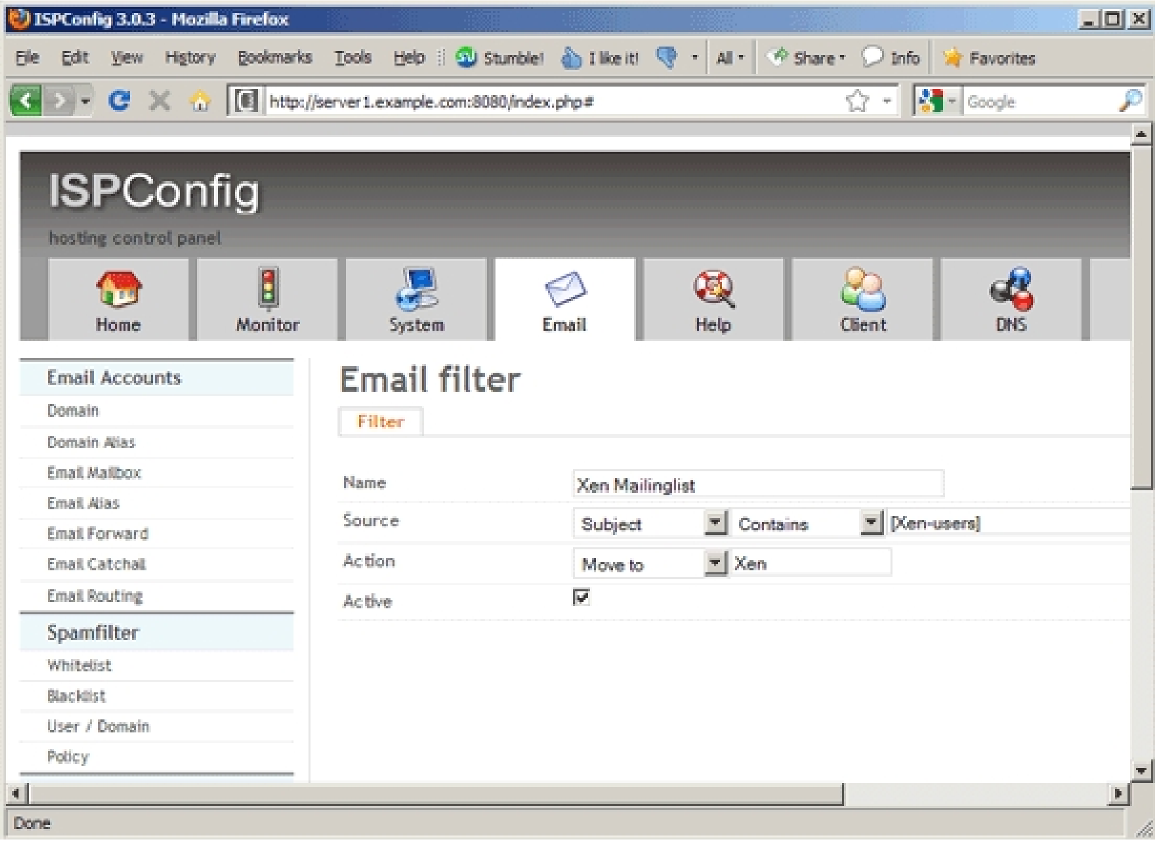
To create custom email filters, click on the *`Add new Filter`* button. This will lead you to the *`Email filter`* form with the tab `Filter`.

Email filter

Filter

The form to create a custom email filter has the following fields:

- *Name*: Specify a name for this filter rule. Examples: *Spam*, *Work*, *Private*, *HowtoForge Newsletter*, *Xen Mailinglist*, etc.
- *Source*: This defines the criteria based on which emails will be filtered. Select the field from the email header that should be examined (*Subject*, *From*, *To*), then select when this filter should be used (if the field *Contains*, *Is*, *Begins with*, *Ends with* the string that you specify), and finally specify a search string. If you select *From* or *To* in the first field and we assume that the email address is specified as *John Doe <john.doe@example.com>*, you can specify an email address here (*john.doe@example.com*) or a name (*John Doe*) - in both cases you should select *Contains* instead of *Is*.
- *Action*: Specify what to do with the emails if the filter applies. If you select *Move to*, you must also specify a folder name in the field right of the drop-down menu. If this folder doesn't exist, it will automatically be created. Please note that you can access this folder only if you use IMAP. If you select *Delete*, the emails will be deleted, and there's no need to specify a folder.
- *Active*: This defines whether this filter rule is currently active or not.



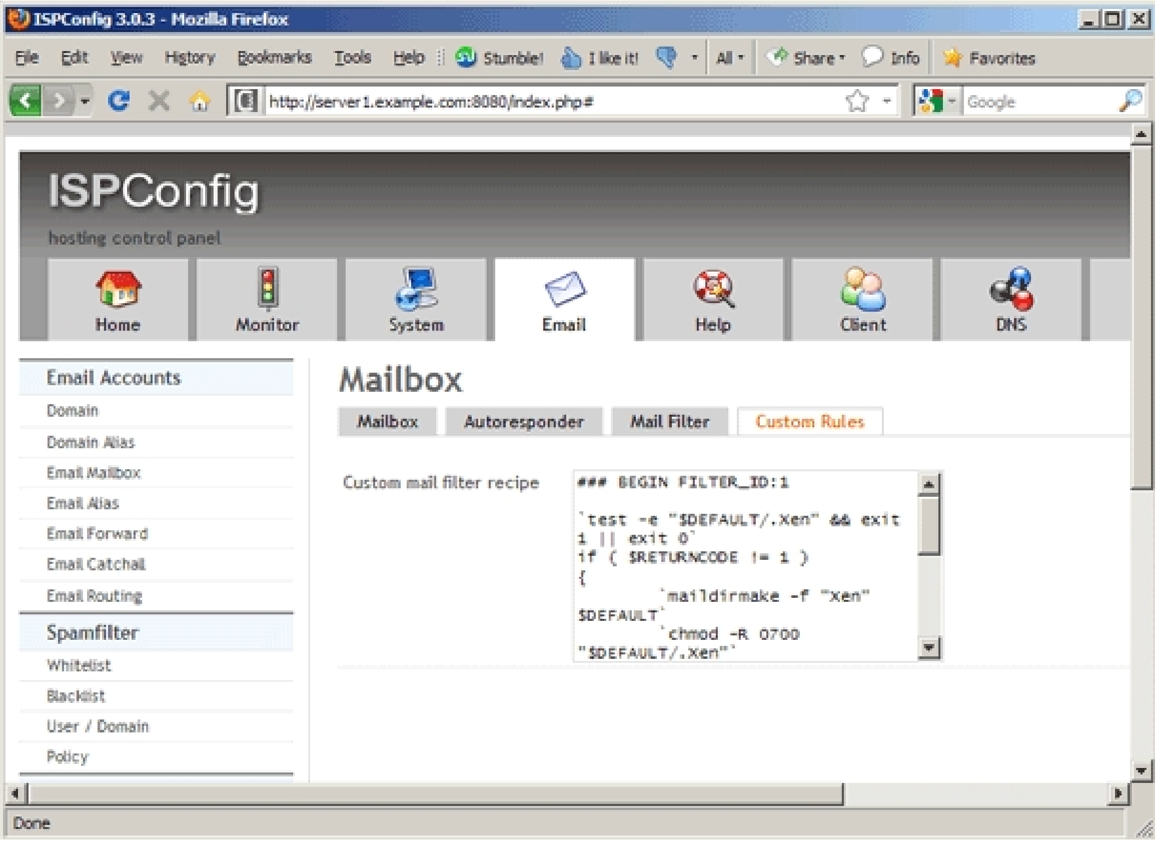
Email filter

Name	Xen Mailinglist
Source	Subject ▾ Contains ▾ [Xen-users]
Action	Move to ▾ Xen
Active	☑

Custom Rules

(This tab is visible only for the ISPConfing *admin* user.)

- `Custom mail filter recipe`: Depending on if you use Courier + Maildrop or Dovecot + Sieve, you can fill in custom directives either in ***Maildrop syntax*** or in ***Sieve syntax***, one directive per line. If you have created a mail filter on the `Mail Filter` tab, you will notice that there are already directives in the text area - that is your mail filter translated into Maildrop or Sieve syntax. You can add further directives, if you like.



Custom mail filter recipe

```
### BEGIN FILTER_ID:1

`test -e "$DEFAULT/.Xen" && exit
1 || exit 0`
if ( $RETURNCODE != 1 )
{
        `maildirmake -f "Xen"
$DEFAULT`
        `chmod -R 0700
"$DEFAULT/.Xen"`
```

4.7.1.4 Email Alias

An email alias is the same as a domain alias, except that it is used to map an email address to another email address instead of mapping a whole email domain to another email domain.

To create a new email alias, click on the `Add new Email alias` button. This will lead you to the `Email Alias` form with the tab `Email Alias`.

Email Alias

Email Alias

This form has the following fields:

- *Email*: This specification is split up in two fields, *Alias* and *Domain*. *Alias* contains the part in front of the *@* sign (the "local part") - it should be an alias that doesn't already exist for this domain -, and in the *Domain* drop-down menu, you select the email domain. For example, if you want to create the email alias *info@example.com*, you'd fill in *info* in the *Alias* field and select *example.com* from the *Domain* drop-down menu.

The local-part of an e-mail address may be up to 64 characters long and the domain name may have a maximum of 255 characters. However, the maximum length of a forward or reverse path length of 256 characters restricts the entire e-mail address to be no more than 254 characters. Some mail protocols, such as X.400, may require larger objects, however. The SMTP specification recommends that software implementations impose no limits for the lengths of such objects.

The local-part of the e-mail address may use any of these ASCII characters:

* Uppercase and lowercase English letters (*a&#8211;z*, *A&#8211;Z*)
* Digits *0* to *9*
* Characters *! # $ % & ' * + - / = ? ^ _ ` { | } ~*
* Character *.* (dot, period, full stop) provided that it is not the first or last character, and provided also that it does not appear two or more times consecutively (e.g. *John..Doe@example.com*).

Additionally, quoted-strings (e.g. *"John Doe"@example.com*) are permitted, thus allowing characters that would otherwise be prohibited, however they do not appear in common practice. RFC 5321 also warns that "a host that expects to receive mail SHOULD avoid defining mailboxes where the Local-part requires (or uses) the Quoted-string form".

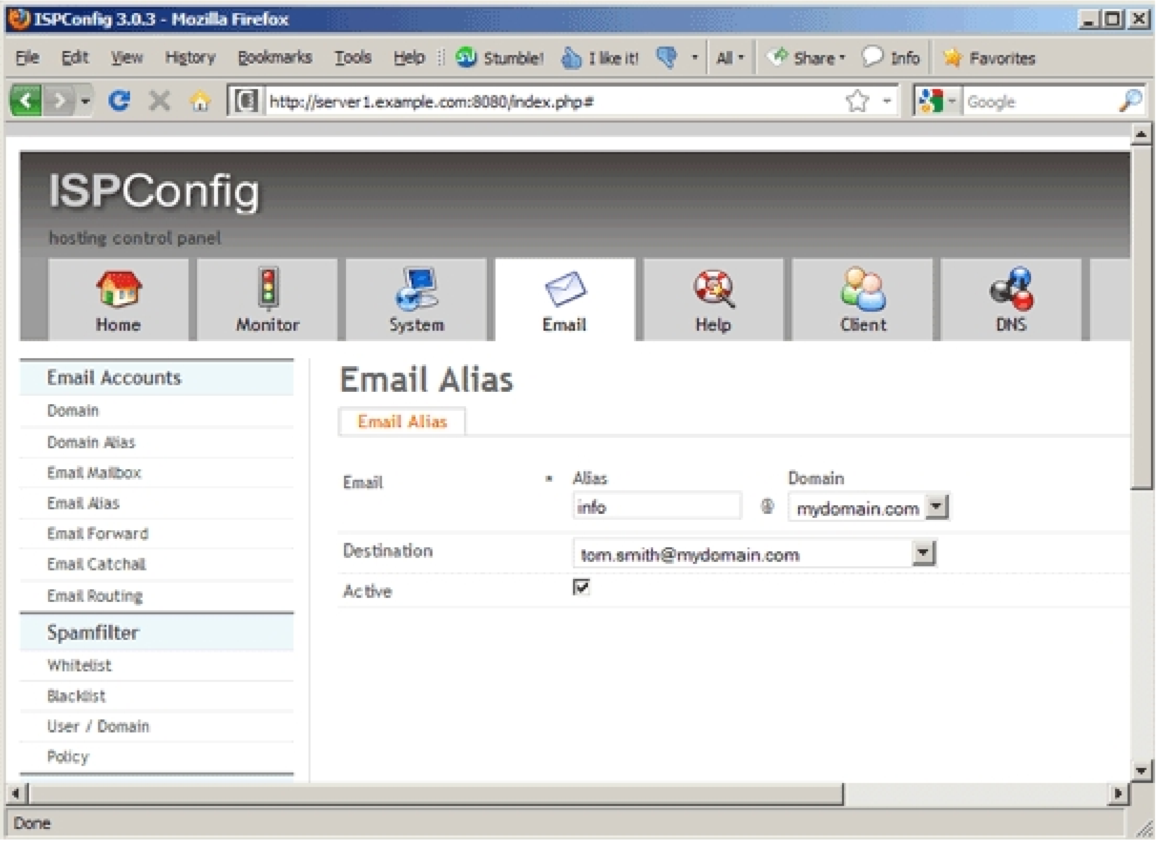
The local-part is case sensitive, so *"jsmith@example.com"* and *"JSmith@example.com"* may be delivered to different people. This practice is discouraged by RFC 5321. However, only the authoritative mail servers for a domain may make that decision (if you have set up your server according to one of the "Perfect Server" tutorials from HowtoForge.com, then the local part is ***not case sensitive***). The only exception is for a local-part value of "postmaster" which is case insensitive, and should be forwarded to the server's administrator.

Within the rules set out in the RFCs, organisations are free to restrict the forms their own e-mail addresses take however they wish, e.g. many organizations do not use certain characters, e.g. space, *?*, and *^*, and most organizations treat uppercase and lowercase letters as equivalent. Hotmail, for example, only allows creation of e-mail addresses using alphanumerics, dot (*.*), underscore (*_*) and hyphen (*-*).

Systems that send mail, of course, must be capable of handling outgoing mail for all addresses. Contrary to the relevant standards, some defective systems treat certain legitimate addresses as invalid and fail to handle mail to these addresses. Hotmail, for example, incorrectly refuses to send mail to any address containing any of the following legitimate characters: *! # $ % * / ? ^ ` { | } ~*

- *Destination*: Select the email account that you want to map this email alias to. If you want to map *info@example.com* to *john.doe@example.com*, you'd select *john.doe@example.com* here. The destination email address is also the SMTP/POP3/IMAP username for the email account.

- *Active*: This defines whether this email alias is active or not.



Email Alias

Email Alias

Email * Alias: info @ Domain: mydomain.com

Destination: tom.smith@mydomain.com

Active: ☑

4.7.1.5 Email Forward

With this feature you can make the mail system automatically forward emails for an email address to one or more other email accounts. For example, you can use this function to define an email address for a group of people, e.g. *dancec lass@mydancestudio.com*, and forward emails to that address to all members of the dance class, like *dancer1@firstdomain.com*, *dancer2@someotherdomain.com*, *dancer3@yetanotherdomain.com*, etc.

To create a new email forward, click on the *Add new Email forward* button. This will lead you to the *Email Forward* form with the tab *Email Forward*.

Email Forward

Email Forward

The form has the following fields:

- *Email*: This specification is split up in two fields, *Alias* and *Domain*. *Alias* contains the part in front of the *@* sign (the "local part") - it should be an alias that doesn't already exist for this domain -, and in the *Domain* drop-down menu, you select the email domain. For example, if you want to create an email forward for the email address *dancec lass@mydancestudio.com*, you'd fill in *dancec lass* in the

`Alias` field and select `mydancestudio.com` from the `Domain` drop-down menu.

The local-part of an e-mail address may be up to 64 characters long and the domain name may have a maximum of 255 characters. However, the maximum length of a forward or reverse path length of 256 characters restricts the entire e-mail address to be no more than 254 characters. Some mail protocols, such as X.400, may require larger objects, however. The SMTP specification recommends that software implementations impose no limits for the lengths of such objects.

The local-part of the e-mail address may use any of these ASCII characters:

* Uppercase and lowercase English letters (*a&#8211;z*, *A&#8211;Z*)
* Digits *0* to *9*
* Characters *! # $ % & ' * + - / = ? ^ _ ` { | } ~*
* Character *.* (dot, period, full stop) provided that it is not the first or last character, and provided also that it does not appear two or more times consecutively (e.g. *John..Doe@example.com*).

Additionally, quoted-strings (e.g. *"John Doe"@example.com*) are permitted, thus allowing characters that would otherwise be prohibited, however they do not appear in common practice. RFC 5321 also warns that "a host that expects to receive mail SHOULD avoid defining mailboxes where the Local-part requires (or uses) the Quoted-string form".

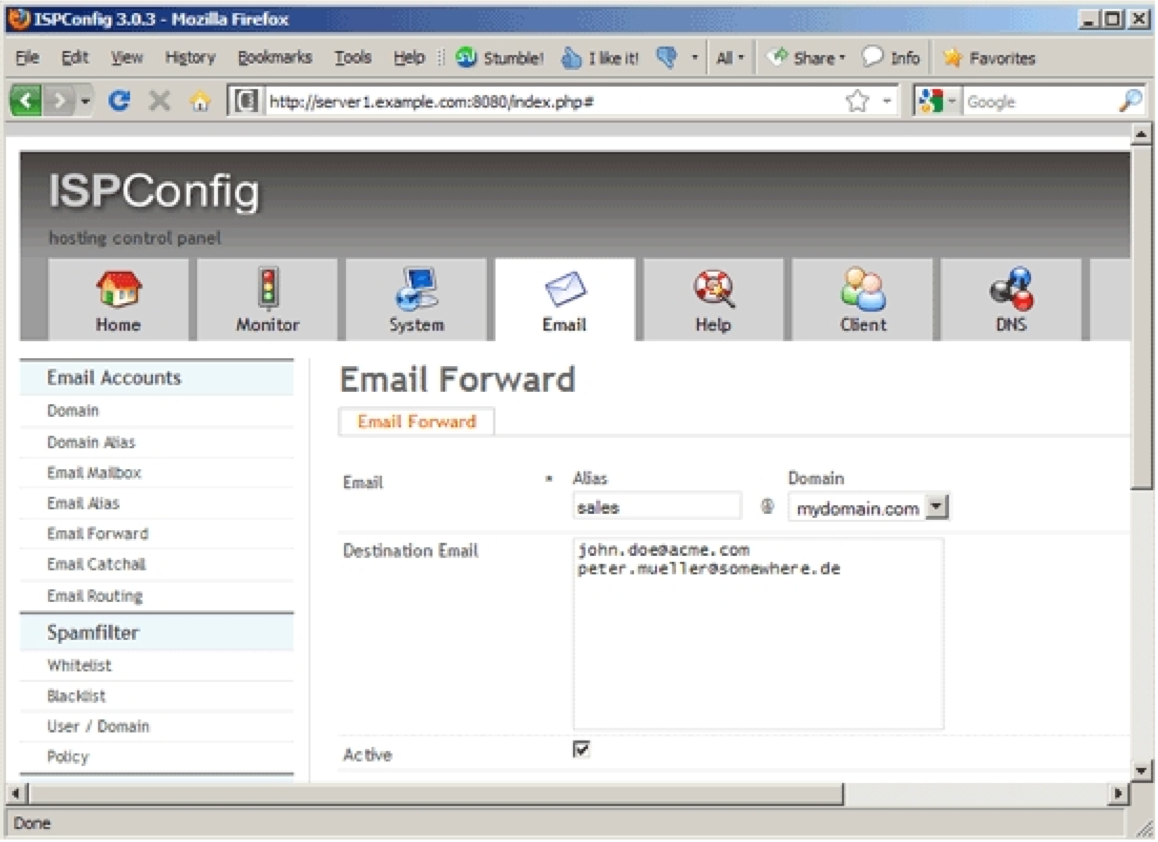
The local-part is case sensitive, so *"jsmith@example.com"* and *"JSmith@example.com"* may be delivered to different people. This practice is discouraged by RFC 5321. However, only the authoritative mail servers for a domain may make that decision (if you have set up your server according to one of the "Perfect Server" tutorials from HowtoForge.com, then the local part is ***not case sensitive***). The only exception is for a local-part value of "postmaster" which is case insensitive, and should be forwarded to the server's administrator.

Within the rules set out in the RFCs, organisations are free to restrict the forms their own e-mail addresses take however they wish, e.g. many organizations do not use certain characters, e.g. space, `?`, and `^`, and most organizations treat uppercase and lowercase letters as equivalent. Hotmail, for example, only allows creation of e-mail addresses using alphanumerics, dot (`.`), underscore (`_`) and hyphen (`-`).

Systems that send mail, of course, must be capable of handling outgoing mail for all addresses. Contrary to the relevant standards, some defective systems treat certain legitimate addresses as invalid and fail to handle mail to these addresses. Hotmail, for example, incorrectly refuses to send mail to any address containing any of the following legitimate characters: *! # $ % * / ? ^ ` { | } ~*

- *`Destination Email`*: Fill in one or more email addresses (one email address per line) that the email should be forwarded to.

- *`Active`*: This defines whether this email forward is active or not.



Email * Alias: sales @ Domain: mydomain.com

Destination Email:
```
john.doe@acme.com
peter.mueller@somewhere.de
```

Active ☑

4.7.1.6 Email Catchall

If you want all emails that are addressed to non-existing mail boxes of a domain to arrive in an existing email box of this domain, you can create a catchAll for this email account. Example: You have configured the email address *`info@example.com`*. Someone sends an email to *`abc@example.com`* which does not exist. If *`info@example.com`* is a catchAll email address the email arrives here. If there is no catchAll email address for this domain the sender of the mail to *`abc@example.com`* gets back an error message ("error: no such user here"). ***Please note:*** Per domain there can be only one catchAll email address.

To create a new email catchAll, click on the *`Add new Catchall`* button. This will lead you to the *`Email Catchall`* form with the tab *`Email Catchall`*.

Email Catchall

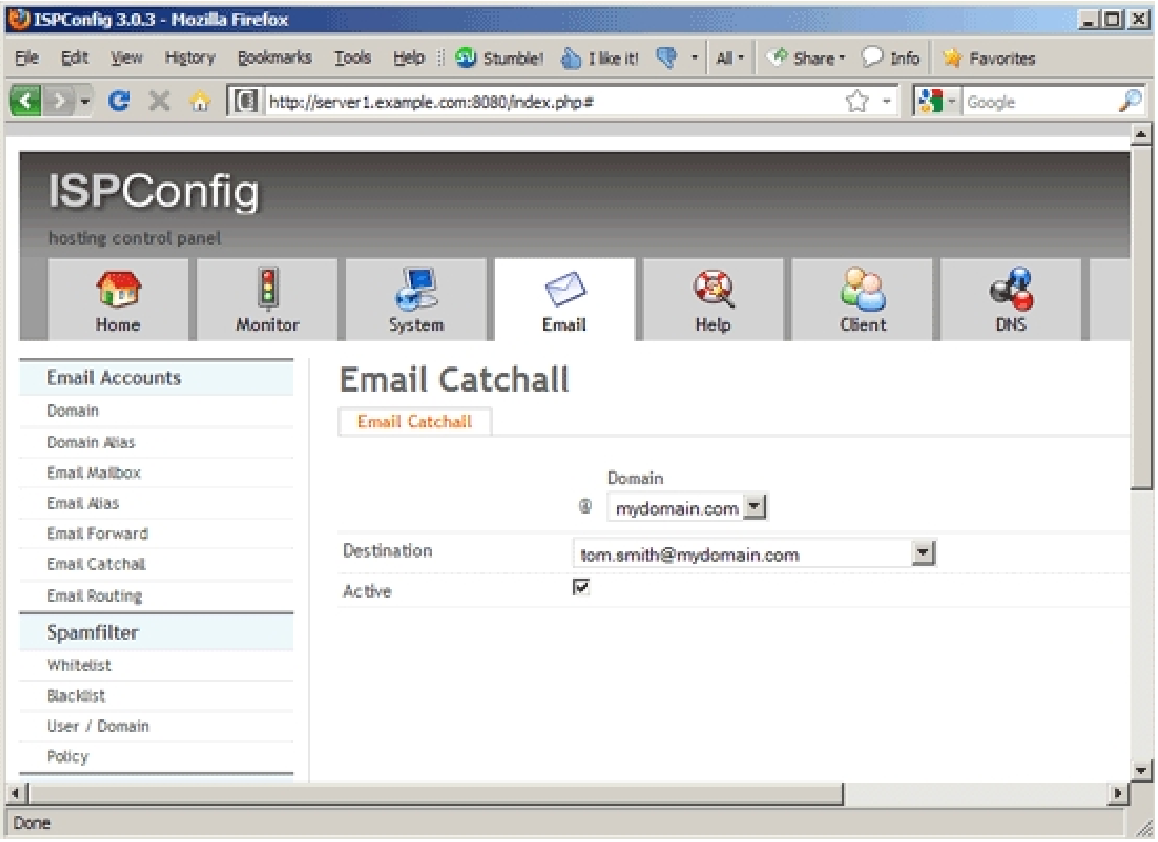
Email Catchall

The form has the following fields:

- *`Source`*: Select the domain for which you want to create a catchAll.
- *`Destination`*: Select the catchAll email account - i.e., the email account that should receive all emails to

non-existing email addresses of this domain.

- *Active*: This defines whether this email catchAll is active or not.



Email Catchall

Email Catchall

Domain: mydomain.com

Destination: tom.smith@mydomain.com

Active: ☑

4.7.1.7 Email Routing

With the email routing feature, you can define what server mail for a given domain will be forwarded to and by what transport. (This feature is based on ***Postfix' transport_maps***.) This makes it possible to route emails for one domain to a totally different server.

Please note that you have create one or more *Relay Recipients* (*Email > Global Filters > Relay Recipients*) for each route that you create so that the system knows it should accept the emails before routing them to another server.

To create a new email route, click on the *Add new transport* button. This will lead you to the *Email Routing* form with the tab *Email transport*.

Email Routing

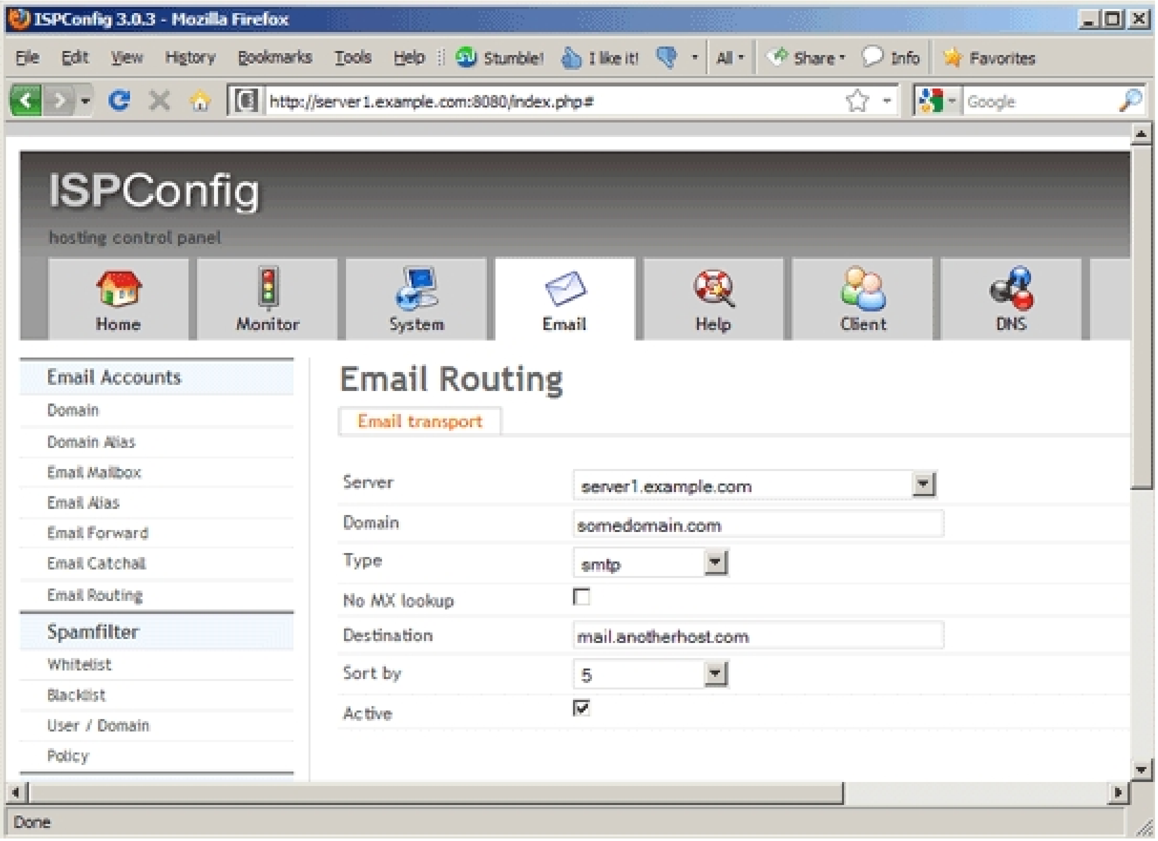
Email transport

This form has the following fields:

- *`Server`*: If more than one server is available, you can select the server on which the email transport will be located. You should select the server that handles emails for the domain that you want to route to another server (i.e., the server that the domain's MX record points to).
- *`Domain`*: Type in the email ***domain*** or email ***address*** that you want to route to another server. You can also use an asterisk (*`*`*) as a wildcard. You can have just one routing rule per domain (ISPConfig will show you an error message if you try to add a second rule with the exact same domain), however if you use an asterisk there can be more than just one routing rule that applies to a domain.
- *`Type`*: Select the transport type (in almost all cases you should use *`smtp`*). Refers to an entry from *`/etc/postfix/master.cf`*, so make sure that what you select here exists in *`/etc/postfix/master.cf`*.
  - *`smtp`*: The Internet standard for transferring email. It uses TCP/IP port 25 and allows for file attachments. You can use the *`Destination`* field to specify the destination host. When no *`Destination`* is specified, the domain name from the *`Domain`* field is used instead.
  - *`uucp`*: A UNIX protocol and set of programs most often used to copy files across serial connections and telephone lines. UUCP was often used to transfer email and Usenet news over phone lines when direct Internet connectivity was scarce in small and medium-sized companies. You can use the *`Destination`* field to specify the UUCP destination host. When no *`Destination`* is specified, the domain name from the *`Domain`* field is used instead.
  - *`slow`*: This transport has to be defined in your *`/etc/postfix/master.cf`* before you can select it. Depending on how your *`slow`* transport looks, you might or might not have to specify a *`Destination`*.
  - *`error`*: The special error transport causes all mail to be rejected. You can use the *`Destination`* field to specify an error message such as *`mail for *.example.com is not deliverable`* (optional).
  - *`custom`*: If you specify a custom transport in *`/etc/postfix/master.cf`*, you can use it for your email routing. Depending on how your *`custom`* transport looks, you might or might not have to specify a *`Destination`*.
  - *`null`*: If you select this transport type, all emails will be deleted. You can leave the *`Destination`* field empty.
- *`No MX lookup`*: This defines whether Postfix will perform an MX lookup for the destination host or not (see the explanation of the next field, *`Destination`*).
- *`Destination`*: The destination host for delivery of messages. The host is used only with inet transports such as SMTP and LMTP. Postfix treats the hostname like any destination domain. It performs an MX lookup to determine where to deliver messages. If there are no MX records, Postfix delivers to the A record IP address. If you know that Postfix should deliver directly to the IP in the A record for the specified host, you can have Postfix skip the check for MX records by checking the *`No MX lookup`* checkbox. If

you use an IP address, it is required that you check the *No MX lookup* checkbox. When no *Destination* is specified, the domain name from the *Domain* field is used instead. IF you use the error transport, you can specify an error message such as *mail for *.example.com is not deliverable* here (optional).

- *Sort by*: Postfix will process all routing rules from top to bottom and use the first one that applies and will stop then. If you have multiple routing rules that might match a certain situation, you can define the order with this field. A higher number means a higher priority, i.e., if you have two rules that apply, and the first has a priority of 8 and the second a priority of 5, then the first rule will be used by Postfix.
- *Active*: This defines whether this email transport is active or not.

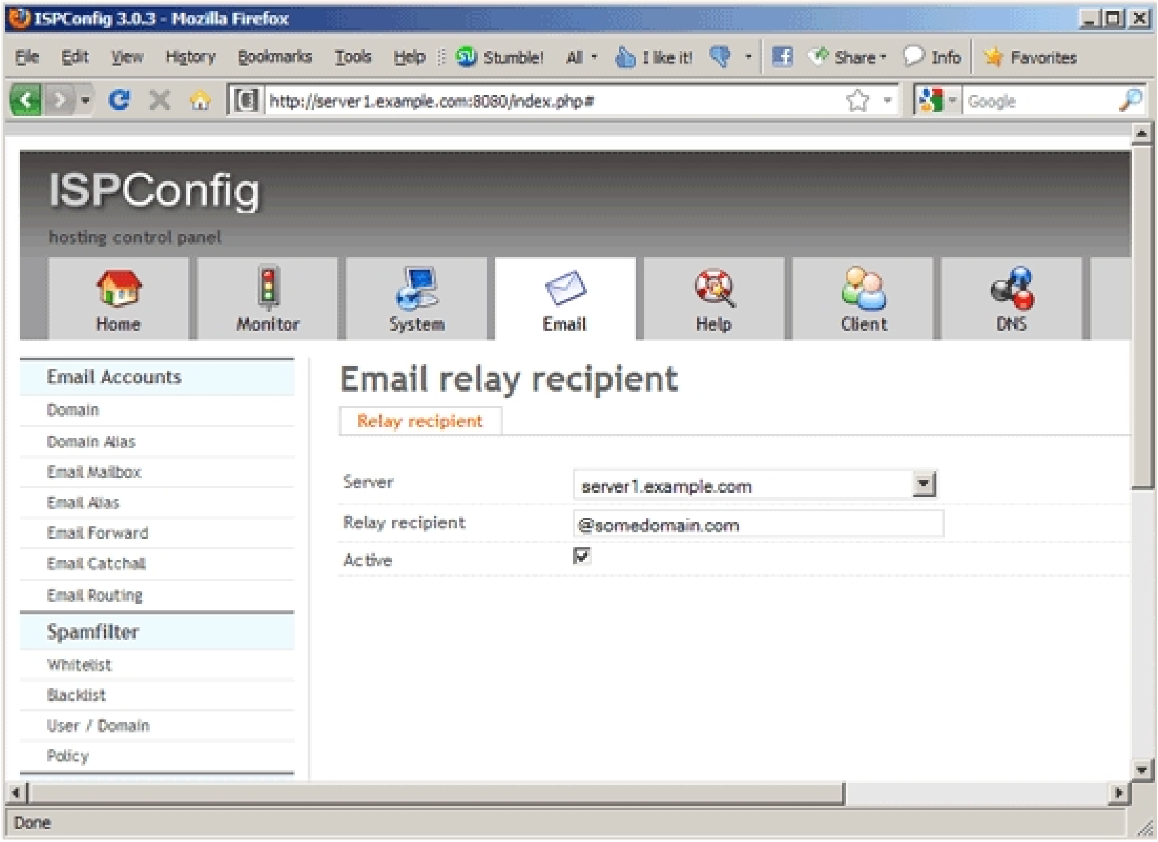


Email Routing

Email transport

Server	server1.example.com
Domain	somedomain.com
Type	smtp
No MX lookup	□
Destination	mail.anotherhost.com
Sort by	5
Active	☑

Please note that you have create one or more *Relay Recipients* (*Email > Global Filters > Relay Recipients*) for each route that you create so that the system knows it should accept the emails before routing them to another server.



Email relay recipient

Server: server1.example.com

Relay recipient: @somedomain.com

Active: ☑

4.7.2 Mailing List

You can create Mailman mailing lists here. Please note that in order to use this feature, Mailman must have been installed and configured according to chapter ***3.1.2 Mailman*** ***before*** you install ISPConfig.

4.7.2.1 Mailing List

To create a new mailing list, click on the `Add new record` button. This will lead you to the `Mailing List` form with the tab `Mailing List`.

Mailing List

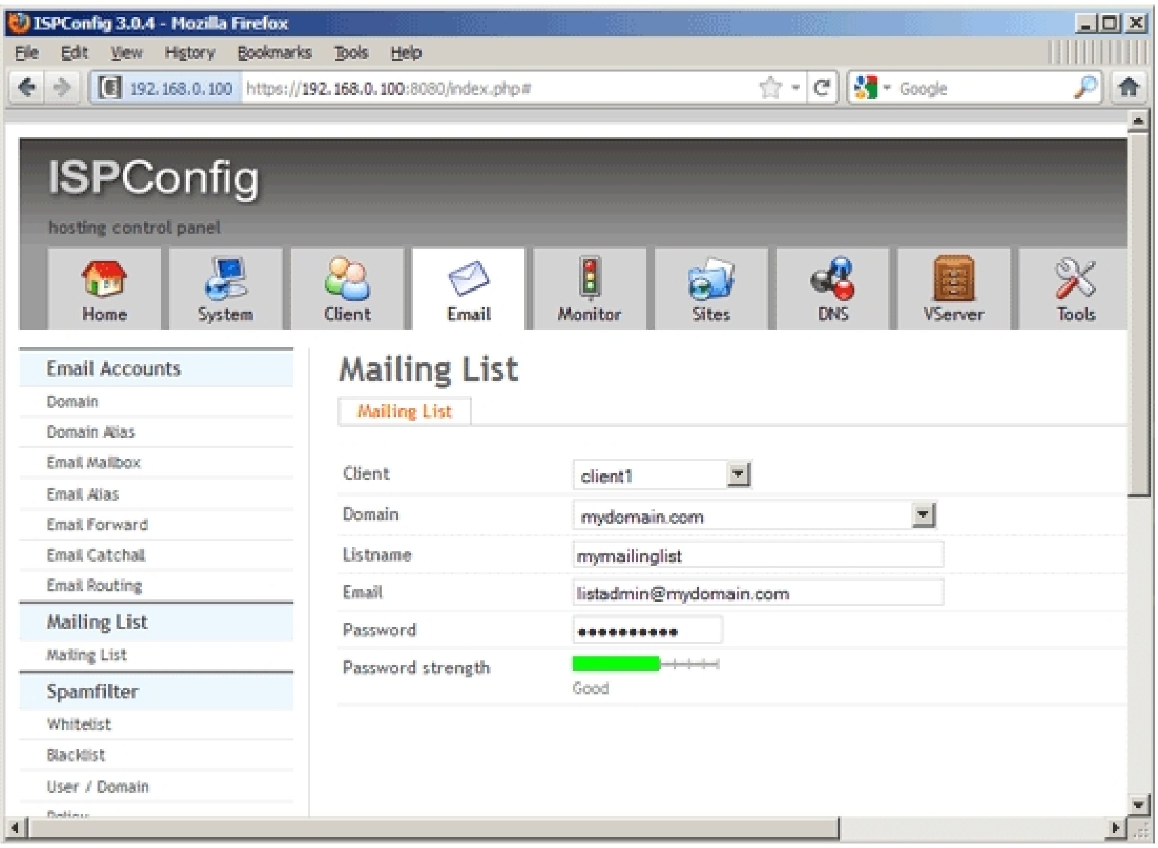
Mailing List

The form has the following fields:

- `Client`: Select the client that owns the mailing list.
- `Domain`: Select the domain to which the mailing list belongs. This is also the list's web interface host name. For example, if you select `example.com` here, and the listname is `testlist`, the administration

interface for testlist will be at *`http://example.com/cgi-bin/mailman/admin/testlist`*, and the web page for users of the mailing list will be at *`http://example.com/cgi-bin/mailman/listinfo/testlist`*.

- *`Listname`*: Specify the name of the list. Note that listnames are forced to lowercase.
- *`Email`*: Specify the email address of the list administrator. The list administrator will receive an email with all important details about the new mailing list after it was created (see below).
- *`Password`*: Please specify the admin password for the mailing list (or use the *`Generate Password`* link to have ISPConfig generate one for you).
- Repeat Password: Confirm your password.



Mailing List

Mailing List

Client: client1

Domain: mydomain.com

Listname: mymailinglist

Email: listadmin@mydomain.com

Password: ●●●●●●●●●●●

Password strength: Good

After the list has been created, the list administrator will receive an email with all important details about the new mailing list, for example as follows:

```
The mailing list `testlist' has just been created for you.  The
  following is some basic information about your mailing list.

  Your mailing list password is:

      1234567890
```

```
You need this password to configure your mailing list.  You also need
it to handle administrative requests, such as approving mail if you
choose to run a moderated list.

You can configure your mailing list at the following web page:

    http://example.com/cgi-bin/mailman/admin/testlist

The web page for users of your mailing list is:

    http://example.com/cgi-bin/mailman/listinfo/testlist

You can even customize these web pages from the list configuration
page.  However, you do need to know HTML to be able to do this.

There is also an email-based interface for users (not administrators)
of your list; you can get info about using it by sending a message
with just the word `help' as subject or in the body, to:

    testlist-request@example.com

To unsubscribe a user: from the mailing list 'listinfo' web page,
click on or enter the user's email address as if you were that user.
Where that user would put in their password to unsubscribe, put in
your admin password.  You can also use your password to change
member's options, including digestification, delivery disabling, etc.

Please address all questions to mailman-owner@example.com.
```

Please note: if you use **nginx**, the URLs from the email will not work right away. Please take a look at chapter ***5.27.2 nginx*** to find out how to access the Mailman web interface.

4.7.3 Spamfilter

4.7.3.1 Whitelist

The whitelist allows you to "whitelist" email sender addresses, i.e., emails from such addresses will never be tagged as spam.

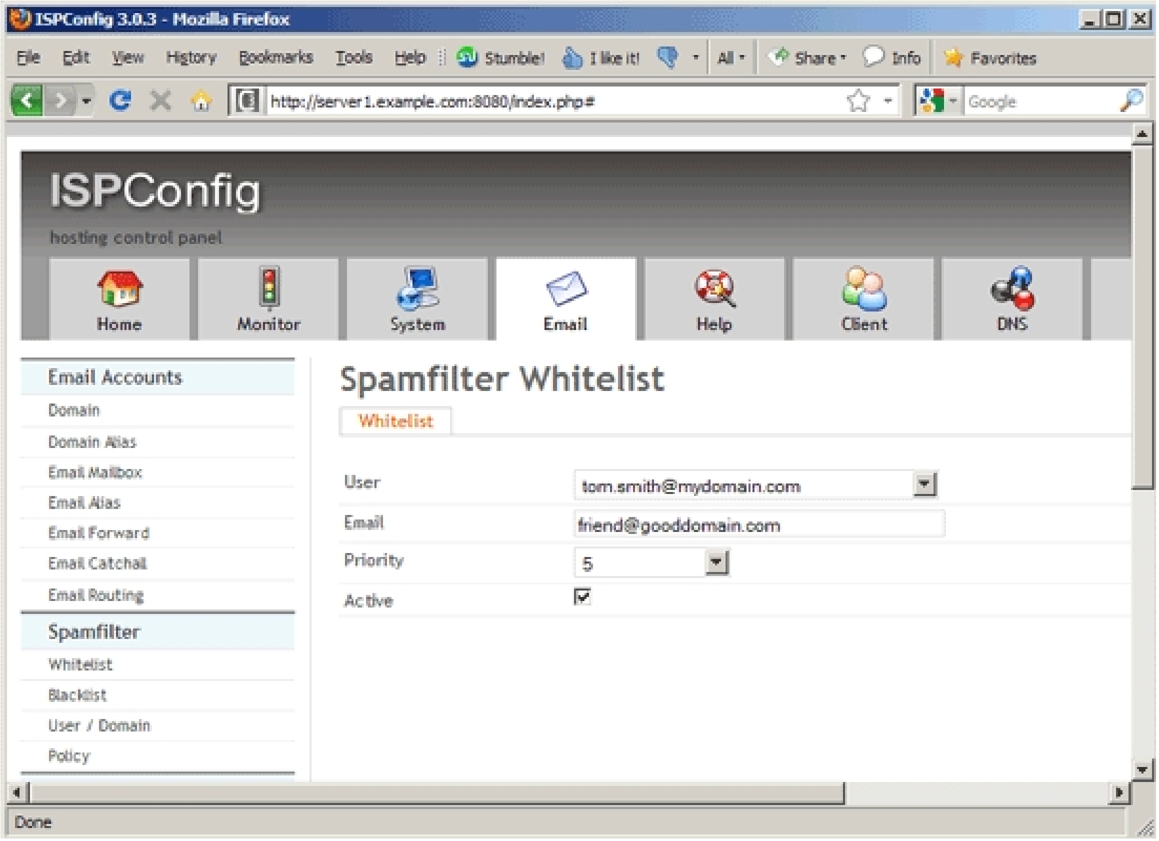
To create a new whitelist, click on the `Add Whitelist record` button. This will lead you to the `Spamfilter Whitelist` form with the tab `Whitelist`.

Spamfilter Whitelist

Whitelist

The form has the following fields:

- *User*: Here you can select the recipient email account or even the whole recipient domain for which this whitelist record will be valid - this whitelist record will not be used for other recipient email accounts or domains.
- *Email*: Specify the email address whose emails should be whitelisted. You can even whitelist a whole domain by leaving out the local part of the email address - i.e., if you want to whitelist emails from the domain *example.com*, type *@example.com* in this field.
- *Priority*: If multiple whitelist/blacklist records apply, this field specifies which rule to use first (*10* = highest priority, *1* = lowest priority). For example, if you blacklist *@example.com* with a priority of *5*, you could whitelist *user@example.com* with a priority of *6* so that *user@example.com*'s mails get through while *@example.com* is blacklisted.
- *Active*: This defines whether this whitelist record is active or not.



Spamfilter Whitelist

Whitelist

User	tom.smith@mydomain.com
Email	friend@gooddomain.com
Priority	5
Active	☑

4.7.3.2 Blacklist

The blacklist allows you to "blacklist" email sender addresses, i.e., emails from such addresses will always be tagged as spam.

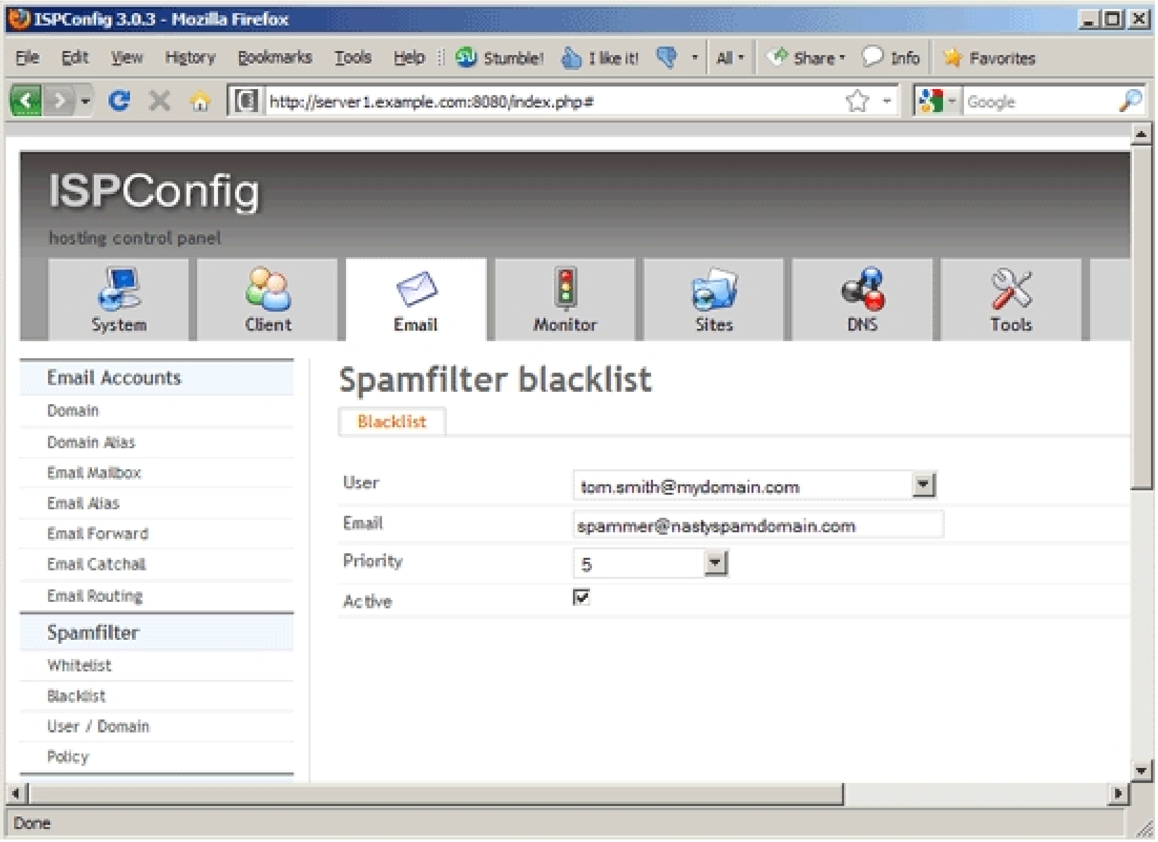
To create a new blacklist, click on the *Add Blacklist record* button. This will lead you to the *Spamfilter blacklist* form with the tab *Blacklist*.

Spamfilter blacklist

Blacklist

The form has the following fields:

- *User*: Here you can select the recipient email account or even the whole recipient domain for which this blacklist record will be valid - this blacklist record will not be used for other recipient email accounts or domains.
- *Email*: Specify the email address whose emails should be blacklisted. You can even blacklist a whole domain by leaving out the local part of the email address - i.e., if you want to blacklist emails from the domain *example.com*, type *@example.com* in this field.
- *Priority*: If multiple whitelist/blacklist records apply, this field specifies which rule to use first (*10* = highest priority, *1* = lowest priority). For example, if you blacklist *@example.com* with a priority of *5*, you could whitelist *user@example.com* with a priority of *6* so that *user@example.com*'s mails get through while *@example.com* is blacklisted.
- *Active*: This defines whether this blacklist record is active or not.



Spamfilter blacklist

Blacklist

User: tom.smith@mydomain.com

Email: spammer@nastyspamdomain.com

Priority: 5

Active: ☑

4.7.3.3 User / Domain

The records that you find here are created automatically by ISPConfig when you create a new email domain or email account (not, when you create a domain alias or an email alias), i.e., for all items that have a `Spamfilter` drop-down menu. These settings tell amavisd when it should scan emails for spam. You can modify these settings here, however, this is usually not necessary. You can also create new records which makes sense for email transports (see chapter ***4.7.1.7 Email Routing***, "Email Routing"), domain aliases, and email aliases.

If you create a record for an email transport, this allows the system to scan emails even if those emails will be forwarded to another server. Normally, such mails would not be scanned.

For domain aliases, there's no automatic record here, and because the record for the target domain doesn't apply to the domain alias, you should create a record if you want emails targetted at the domain alias to be scanned for spam as well.

For email aliases, there's no automatic record here either, and the record for the target email account doesn't apply to the email alias. If there's a record here for the domain of the email alias, then this record applies for the email alias - if there's no record for the domain either, then there's no spam scanning for the email alias at all. If you want spam-scanning settings for the email alias that differ from the domain record or if there's no domain record at all, you can create a record for the email alias here.

What I wrote about the domain aliases and email aliases is true because spam scanning takes place ***before*** addresses are rewritten. So if you have the email account `user@example.com` with spam scanning enabled and the email alias for this mailbox *alias@example.com*, spam scanning would take place before *alias@example.com* is rewritten to *user@example.com*, and because there's no record for *alias@example.com*, no spam scanning takes place for *alias@example.com*, while mails for *user@example.com* are scanned. You can change this behaviour by commenting out or removing the line

receive_override_options = no_address_mappings

from */etc/postfix/main.cf* (don't forget to restart Postfix) - in this case address rewriting takes place before spam scanning, which means you don't need extra rules for aliases because the records for the main domain/main email account apply.

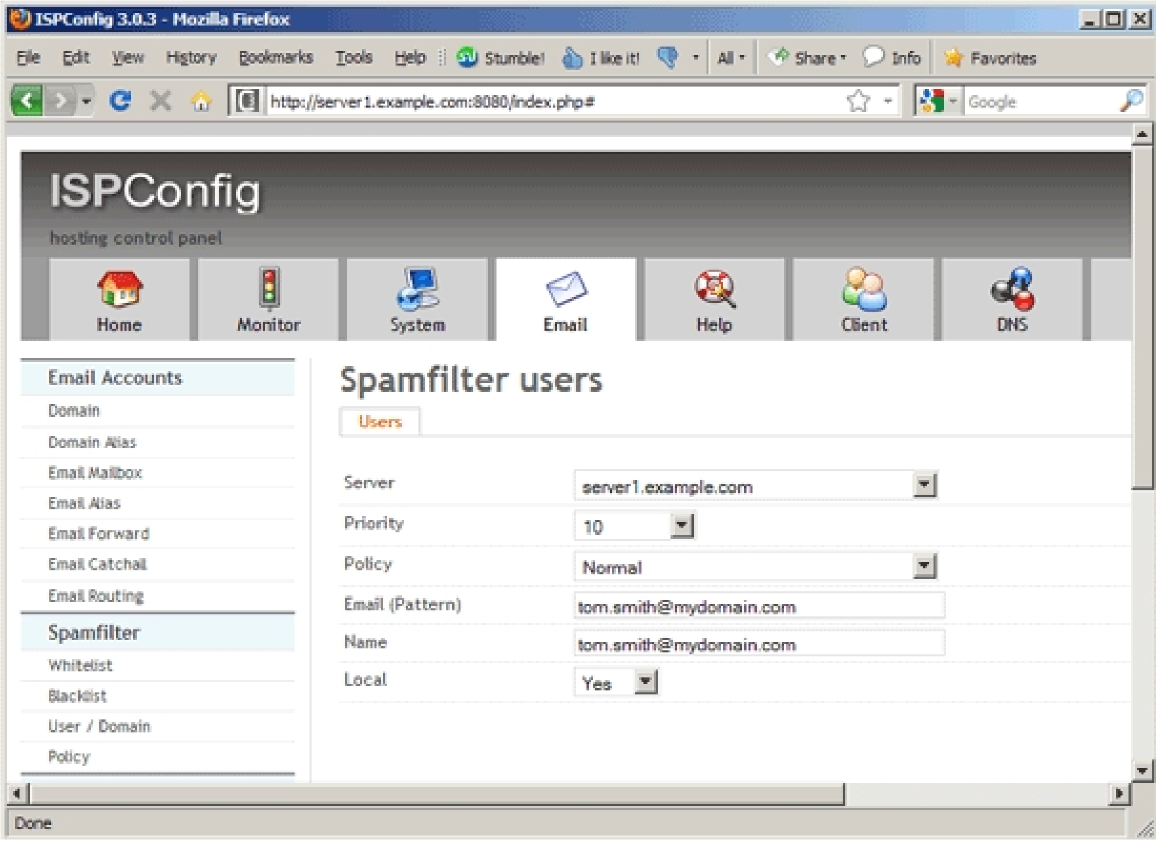
To create a new record, click on the *Add Spamfilter User* button. This will lead you to the *Spamfilter users* form with the tab *Users*.

Spamfilter users

Users

This form has the following fields:

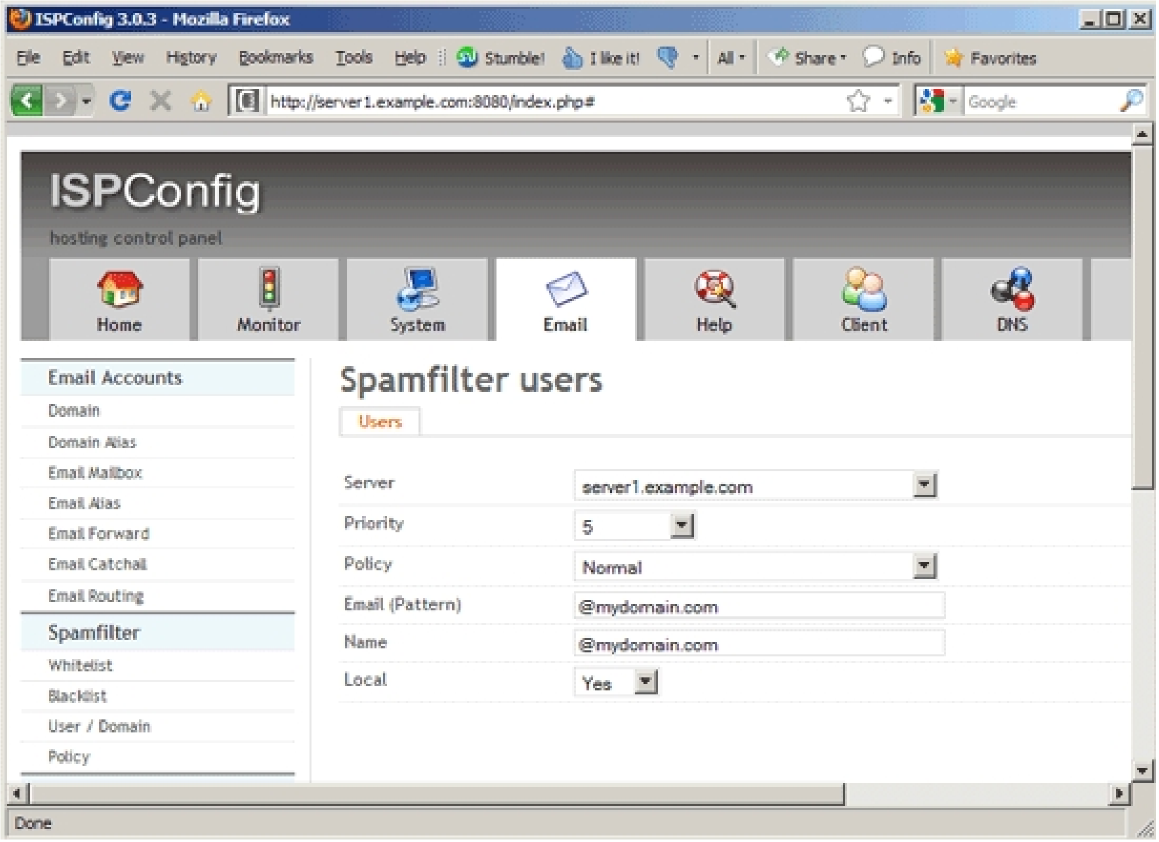
- *Server*: If more than one server is available, you can select the server on which the record will be located.
- *Priority*: If multiple records apply, this field specifies which rule to use first (*10* = highest priority, *1* = lowest priority). For example, if you have a record for a whole domain with the priority *5* and a record for a specific email account (from the same domain) with the priority *10*, Then the record with priority *10* will override the record with priority *5*.
- *Policy*: Here you can specify the spamfilter level to use: *Non-Paying*, *Uncensored*, *Wants all spam*, *Wants viruses*, *Normal*, *Trigger happy*, *Permissive*. The settings for each of these levels are defined under *Email > Spamfilter > Policy*.
- *Email (Pattern)*: Fill in the email address (e.g. *user@example.com*) or the domain (with the *@* in front, e.g. *@example.com*), to which the rule should apply.
- *Name*: Specify a name for the rule. You can use the email address or domain, but you can as well fill in something else, such as *Rule1* etc. This is just for you so that you can distinguish the rules.
- *Local*: This specifies if this record is active (*Yes*) or not (*No*).



Spamfilter users

Users

Server	server1.example.com
Priority	10
Policy	Normal
Email (Pattern)	tom.smith@mydomain.com
Name	tom.smith@mydomain.com
Local	Yes



Spamfilter users

Users

Server	server1.example.com
Priority	5
Policy	Normal
Email (Pattern)	@mydomain.com
Name	@mydomain.com
Local	Yes

4.7.3.4 Policy

Here you can modify existing spam levels (*Non-Paying*, *Uncensored*, *Wants all spam*, *Wants viruses*, *Normal*, *Trigger happy*, *Permissive*) and create new levels, if needed.

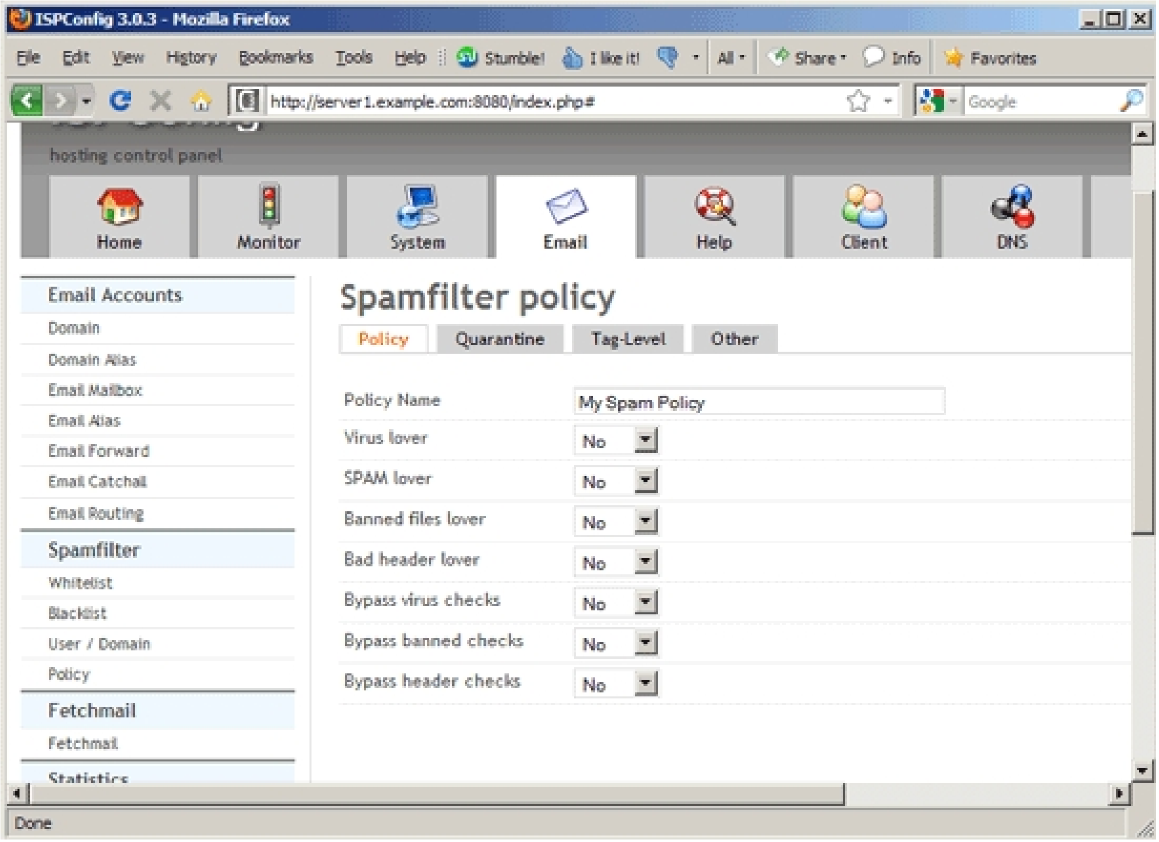
To create a new policy, click on the *Add Policy record* button. This will lead you to the *Spamfilter policy* form with the tabs *Policy*, *Quarantine*, *Tag-Level*, *Other*.

Spamfilter policy

Policy

On this tab you find the following fields:

- *Policy Name*: Specify the name of the rule.
- *Virus lover*: Select if viruses should be allowed through this filter (*Yes*) or not (*No*). Emails will still be scanned for viruses, but results of virus checks are ignored.
- *SPAM lover*: Select if spam should be allowed through this filter (*Yes*) or not (*No*). Emails will still be scanned for spam, but results of spam checks are ignored.
- *Banned files lover*: Select if banned files (like, for example, *.exe*) should be allowed through this filter (*Yes*) or not (*No*). Emails will still be scanned for banned files, but results of banned files checks are ignored. Please note that this setting applies only if banned names and types checks are enabled in your amavisd configuration (see ***http://www.ijs.si/software/amavisd/amavisd-new-docs.html#checks***).
- *Bad header lover*: Select if mails with bad headers should be allowed through this filter (*Yes*) or not (*No*). Emails will still be scanned for bad headers, but results of bad header checks are ignored.
- *Bypass virus checks*: Similar in concept to *Virus lover*, this is used to skip entirely the decoding, unpacking and virus checking.
- *Bypass banned checks*: Similar in concept to *Banned files lover*, this is used to skip entirely the decoding, unpacking and banned files checking.
- *Bypass header checks*: Similar in concept to *Bad header lover*, this is used to skip entirely the decoding, unpacking and bad header checking.



Spamfilter policy

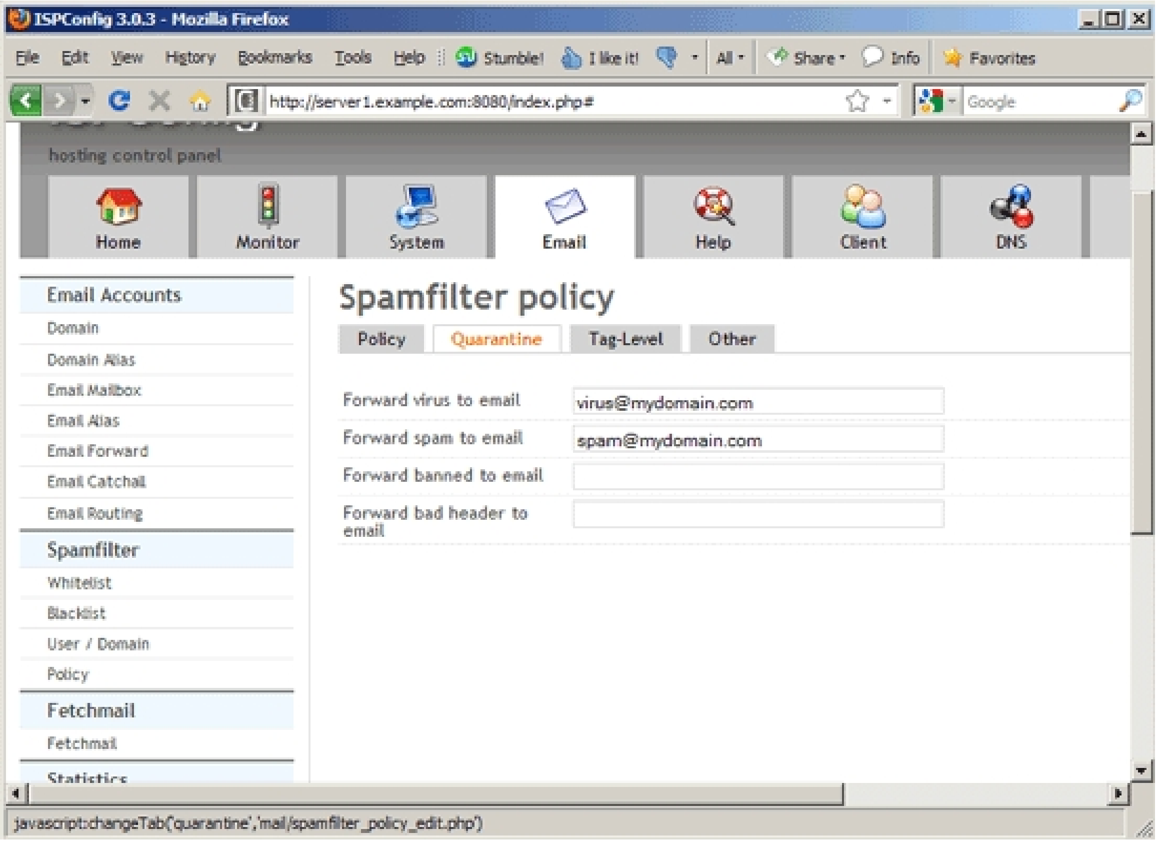
Policy Name	My Spam Policy
Virus lover	No
SPAM lover	No
Banned files lover	No
Bad header lover	No
Bypass virus checks	No
Bypass banned checks	No
Bypass header checks	No

Quarantine

Here you can define quarantine settings for emails containing viruses, spam, banned files, and bad headers.

The form contains the following fields:

- *`Forward virus to email`*: If you want to quarantine virus emails, specify an email address here to which the virus mails will be forwarded.
- *`Forward spam to email`*: If you want to quarantine spam emails, specify an email address here to which the spam mails will be forwarded.
- *`Forward banned to email`*: If you want to quarantine emails that contain banned files, specify an email address here to which these mails will be forwarded.
- *`Forward bad header to email`*: If you want to quarantine emails with bad headers, specify an email address here to which these mails will be forwarded.



Spamfilter policy

Forward virus to email	virus@mydomain.com
Forward spam to email	spam@mydomain.com
Forward banned to email	
Forward bad header to email	

Tag-Level

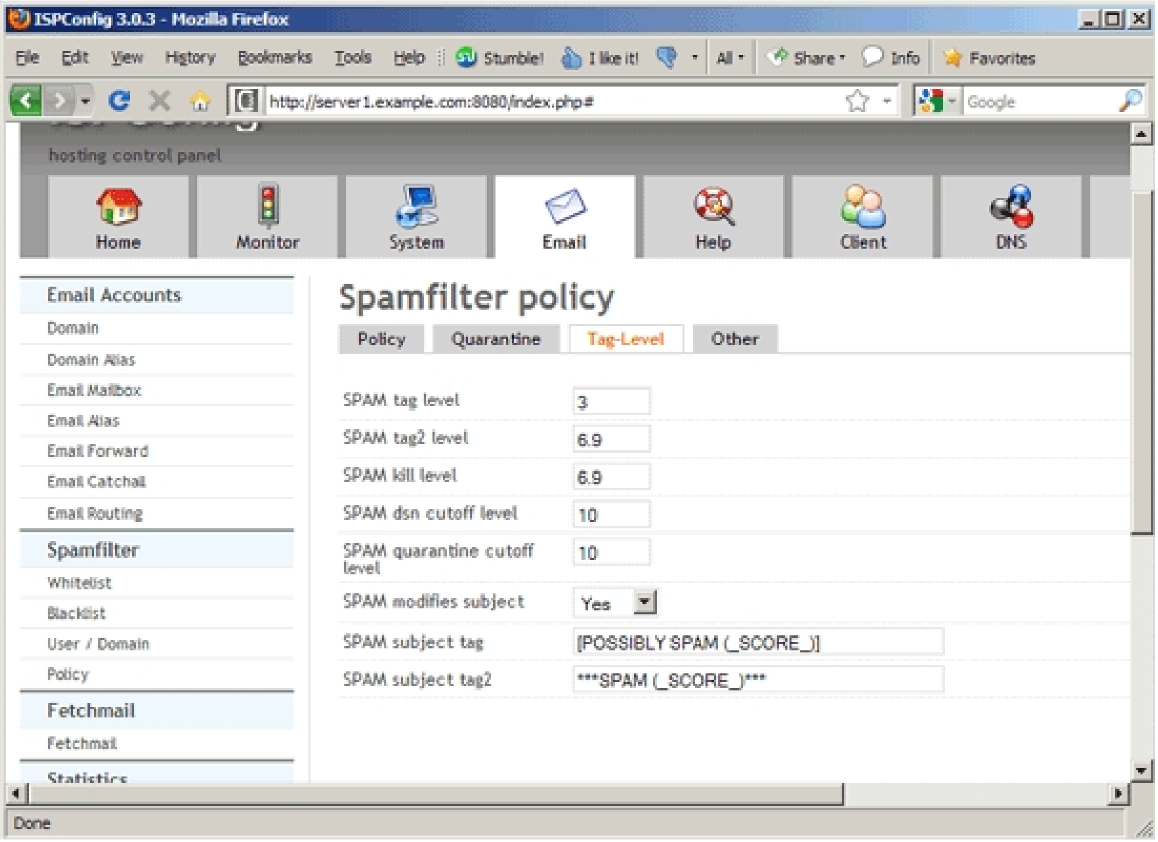
On this tab you can define spam scores and how spam mails will be tagged in the subject line.

The form has the following fields:

- *SPAM tag level*: The system will add spam info headers to the email if at, or above that level. Should be a value > 0; for ISPConfig's *Normal* spam level, the score is 3. Decimal numbers such as 2.4 are allowed.
- *SPAM tag2 level*: The system will add 'spam detected' headers at that level. The value should be > *SPAM tag level*. For ISPConfig's *Normal* spam level, the score is 6.9. Decimal numbers are allowed.
- *SPAM kill level*: The system will trigger spam evasive actions (e.g. blocks mail) at that level. The value should be >= *SPAM tag2 level*. For ISPConfig's *Normal* spam level, the score is 6.9. Decimal numbers are allowed. Important: if *SPAM kill level* = *SPAM tag2 level*, spam will be blocked and not delivered to the user's mailbox, so it doesn't make sense to specify a *SPAM subject tag2* (see **below**).
- *SPAM dsn cutoff level*: This is the spam score beyond which a DSN (Delivery Status Notification) is not sent. Given the fact that almost all spam emails have a fake sender address, it is arguable if you should send a DSN at all. To not send a DSN, specify a low score such as 0.
- *SPAM quarantine cutoff level*: This is the spam score beyond which quarantine is off. Use a low

score (e.g. 0) if you don't want quarantine.

- *SPAM modifies subject*: Select if you want the system to tag the email's subject line with a spam tag if it is categorized as spam. The spam tag can be set in the two below fields, *SPAM subject tag* and *SPAM subject tag2*.
- *SPAM subject tag*: This applies only if the spam score is >= *SPAM tag level*, i.e., if spam info headers are added to the mail, but it is not sure if it is really spam. Normally you leave this field empty. If you don't want to leave this empty, a suitable tag could be [POSSIBLY SPAM]. It is also possible to include the spam score in the spam tag by using _SCORE_, e.g. [POSSIBLY SPAM (_SCORE_)]. In the end it would result in something like [POSSIBLY SPAM (Score: 3.1)].
- *SPAM subject tag2*: This is the field you usually use to tag spam in the subject field. This setting applies if the spam score is >= *SPAM tag2 level*, i.e. if this mail is almost certainly spam. Usual strings are [SPAM] or ***SPAM***. The string will be prepended to the email's subject, for example the subject *Buy Cialis* would become [SPAM] Buy Cialis. You can use this spam tag to filter emails in your email client. It is also possible to include the spam score in the spam tag by using _SCORE_, e.g. ***SPAM (_SCORE_)***. In the end it would result in something like ***SPAM (Score: 7.5)***. Important: if *SPAM kill level* = *SPAM tag2 level*, spam will be blocked and not delivered to the user's mailbox, so it doesn't make sense to specify a *SPAM subject tag2*.



Spamfilter policy

SPAM tag level	3
SPAM tag2 level	6.9
SPAM kill level	6.9
SPAM dsn cutoff level	10
SPAM quarantine cutoff level	10
SPAM modifies subject	Yes
SPAM subject tag	[POSSIBLY SPAM (_SCORE_)]
SPAM subject tag2	***SPAM (_SCORE_)***

Other

On this tab you can configure various other settings, e.g. "plus addressing".

From the ***amavisd-new documentation***:

Amavisd-new can tag passed malware by appending an address extension to a recipient address. An address extension is usually a short string (such as 'spam') appended to the local part of the recipient address, delimited from it by a single character delimiter, often a '+' (or sometimes a '-'). This is why address extensions are also known as "plus addressing". Examples of such mail addresses belonging to user jim@example.com are: jim+spam@example.com , jim+cooking@example.com , jim+health@example.com , jim+postfix@example.com .

Most mailers (MTA), including Postfix and sendmail, have some provision to put address extensions to good use. Similarly, local delivery agents (LDA) such as Cyrus or LDAs that come with MTA, can be configured to recognize and make use of address extensions.

The most common application for address extensions is to provide additional information to LDA to store mail into a separate mail folder. Users may for example choose to use this feature to let LDA automatically file messages from mailing lists to a dedicated subfolder, or to file spam to a spam folder, just by letting LDA simply and quickly examine the envelope recipient address, without having to parse mail header or having to configure and run filters such as procmail or Sieve.

Mailers (MTA and LDA) usually attempt first to examine (to check for validity, to lookup in virtual or aliases maps) a full unmodified recipient address. If the attempt is unsuccessful, they strip away the extension part, and try again. This way a presence of some unknown address extension is simply ignored. For example, a delivery for jim+health@example.com would deliver the mail to the main Jim's inbox if he hasn't provided a subfolder health in his mailbox.

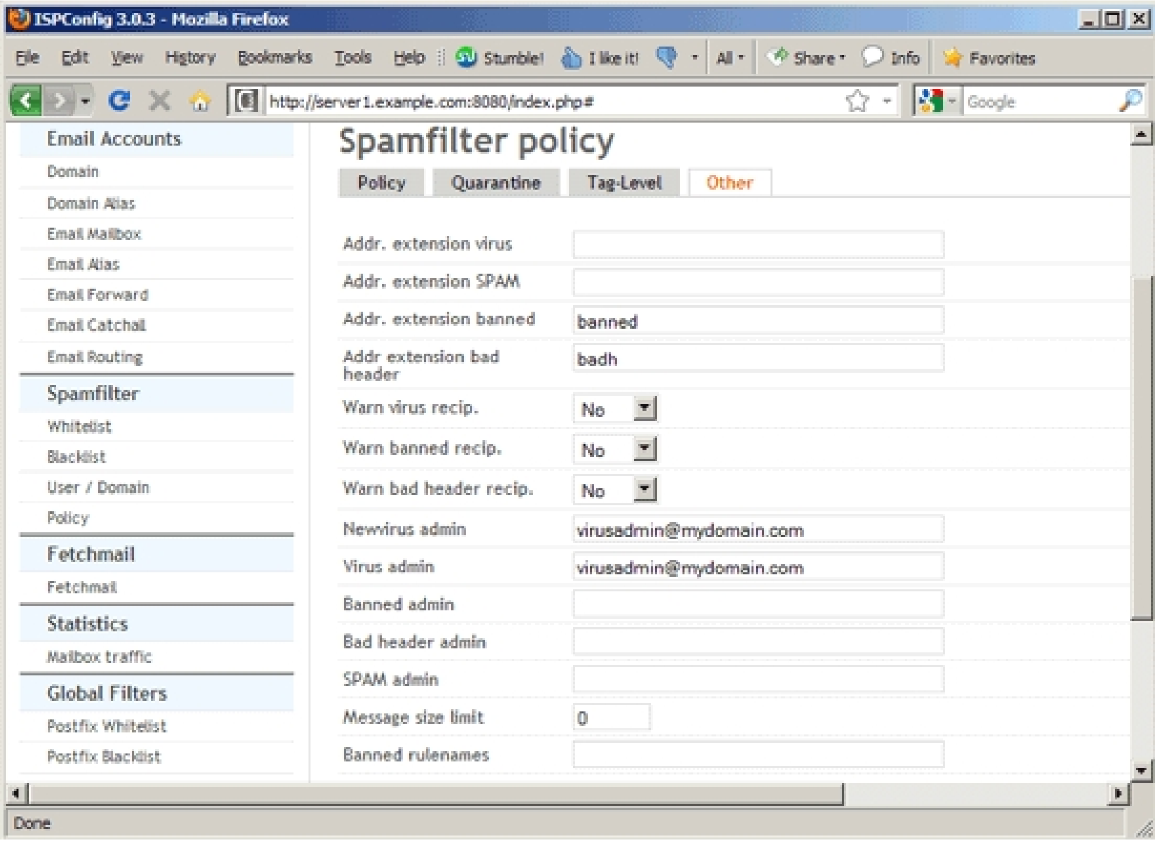
For this fallback to work (to ignore unknown extensions), it is important that all components that need to deal with address extensions (MTA, LDA, content filters) have the same notion of the delimiter in use on the system. For Postfix the configuration option is recipient_delimiter=+ (see also propagate_unmatched_extensions), for amavisd-new the option is $recipient_delimiter='+'.

The form contains the following fields:

- *Addr. extension virus*: Specify an address extension for virus mails. For example, if you specify *virus* (without + at the beginning), the email address would be rewritten to *user+virus@example.com* , and viruses would be delivered to the *virus* folder of the *user@example.com* mailbox. It is possible to access that folder via IMAP. Please note that viruses are delivered only if you've set *Virus lover* or *Bypass virus checks* to *Yes* on the *Policy* tab.
- *Addr. extension SPAM*: Specify an address extension for spam mails. For example, if you specify *spam* (without + at the beginning), the email address would be rewritten to *user+spam@example.com*, and spam would be delivered to the *spam* folder of the *user@example.com* mailbox. It is possible to access that folder via IMAP. Please note that spam is delivered only if you've set *Spam lover* to *Yes* on the *Policy* tab, or if the spam score is > *SPAM tag2 level*and < *SPAM kill level* (see the ***Tag-Level*** tab).
- *Addr. extension banned*: Specify an address extension for mails containing banned files. For example, if you specify *banned* (without + at the beginning), the email address would be rewritten to

user+banned@example.com, and mails containing banned files would be delivered to the *banned* folder of the *user@example.com* mailbox. It is possible to access that folder via IMAP. Please note that mails containing banned files are delivered only if you've set *Banned files lover* or *Bypass banned checks* to *Yes* on the *Policy* tab.

- *Addr extension bad header*: Specify an address extension for mails containing bad headers. For example, if you specify *badh* (without + at the beginning), the email address would be rewritten to *user+badh@example.com*, and mails containing bad headers would be delivered to the *badh* folder of the *user@example.com* mailbox. It is possible to access that folder via IMAP. Please note that mails containing bad headers are delivered only if you've set *Bad header lover* or *Bypass header checks* to *Yes* on the *Policy* tab.
- *Warn virus recip.*: Set this to *Yes* if you want the system to send a warning email to the recipient whenever a virus email is sent.
- *Warn banned recip.*: Set this to *Yes* if you want the system to send a warning email to the recipient whenever an email containing banned files is sent.
- *Warn bad header recip.*: Set this to *Yes* if you want the system to send a warning email to the recipient whenever an email containing bad headers is sent.
- *Newvirus admin*: Here you can specify an email address to which notifications of newly encountered viruses since amavisd startup are sent.
- *Virus admin*: Here you can specify an email address to which notifications of detected viruses are sent.
- *Banned admin*: Here you can specify an email address to which notifications of banned content are sent.
- *Bad header admin*: Here you can specify an email address to which notifications of bad headers are sent.
- *SPAM admin*: Here you can specify an email address to which notifications of received spam are sent.
- *Message size limit*: This is the maximum size of an email (in bytes) beyond which amavisd-new performs no checks (to save system resources). *0* means that amavisd-new does not care about the mail size.
- *Banned rulenames*: In this field you can specify SpamAssassin rules that should not be used to find out if an email is spam or not. Multiple names can be specified comma-separated (or whitespace-separated), e.g. *HTML_MESSAGE, MIME_QP_LONG_LINE*.



Spamfilter policy

Addr. extension virus	
Addr. extension SPAM	
Addr. extension banned	banned
Addr extension bad header	badh
Warn virus recip.	No
Warn banned recip.	No
Warn bad header recip.	No
Newvirus admin	virusadmin@mydomain.com
Virus admin	virusadmin@mydomain.com
Banned admin	
Bad header admin	
SPAM admin	
Message size limit	0
Banned rulenames	

4.7.4 Fetchmail

4.7.4.1 Fetchmail

This feature can be used to retrieve emails from a remote POP3 or IMAP account and put them into a local mailbox. Although this feature is called "Fetchmail" here, ISPConfig uses ***getmail*** instead of ***fetchmail*** under the hood.

To create a new Fetchmail account, click on the *Add new Account* button. This will lead you to the *Get Email* form with the tab *Get Email*.

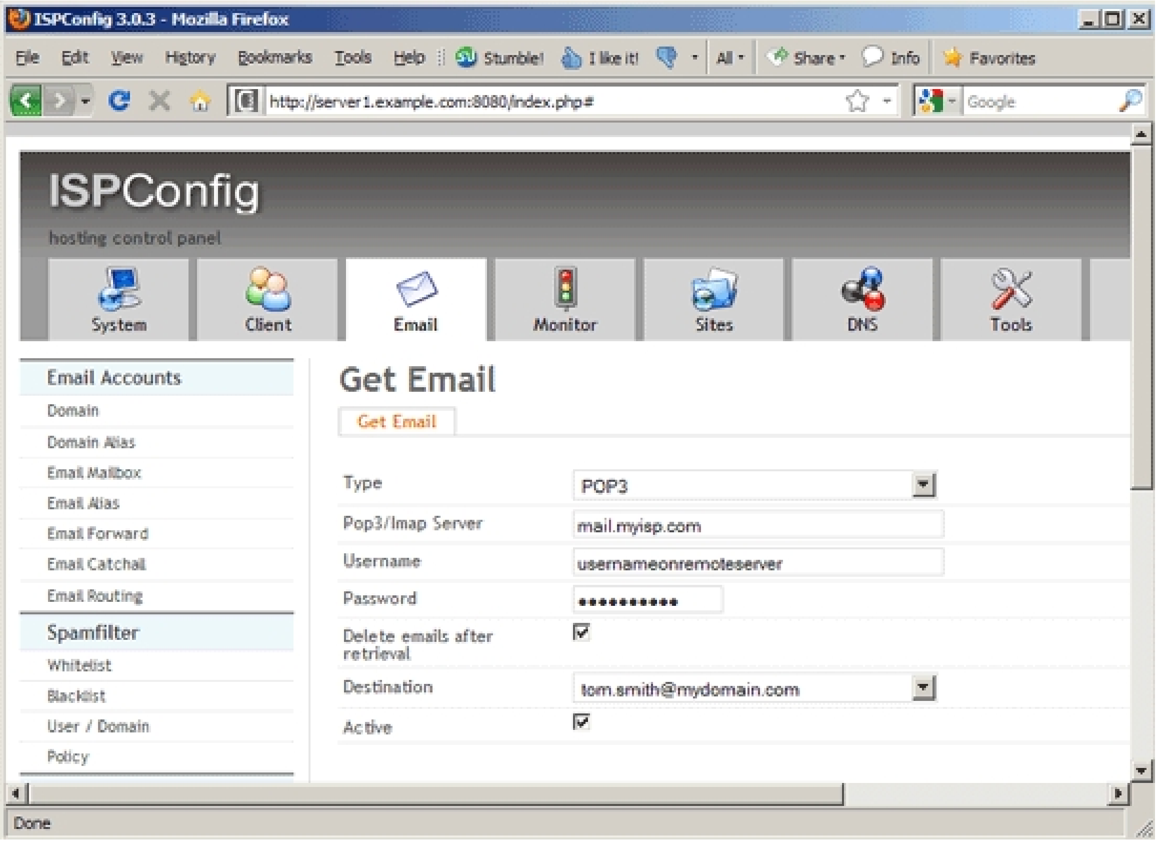
Get Email

Get Email

This form has the following fields:

- *Type*: Select the protocol to use to retrieve emails from the remote account (*POP3*, *IMAP*, *POP3SSL*, *IMAPSSL*).

- `Pop3/Imap Server`: Specify the hostname of the remote mail server, e.g. `mail.example.com`.
- `Username`: Specify the username of the remote email account.
- `Password`: Specify the user's password.
- `Delete emails after retrieval`: Select if you want emails to be automatically deleted on the remote host after they have been retrieved.
- `Retrieve all emails (incl. read mails)`: By default, only new emails will be retrieved from the remote server. If you check this box, read emails will be retrieved as well.
- `Destination`: Select the destination mailbox for the retrieved emails.
- `Active`: This defines whether this Fetchmail account is active or not.



Get Email

Get Email

Type	POP3
Pop3/Imap Server	mail.myisp.com
Username	usernameonremoteserver
Password	●●●●●●●●●●●
Delete emails after retrieval	☑
Destination	tom.smith@mydomain.com
Active	☑

4.7.5 Statistics

The `Statistics` section is a bit special in that there's nothing that you can configure here. This section just displays statistics for your email accounts.

4.7.5.1 Mailbox quota

Under *Mailbox quota* you can see used space (in KB) for your email accounts. This feature is available only if you use Dovecot; the mailbox quota report is not available if you use Courier.

4.7.5.2 Mailbox traffic

Under *Mailbox traffic* you can see traffic statistics (in MB) for your email accounts for the current month, the month before, the current year, and the year before. Please note that this traffic covers only incoming traffic, not outgoing emails. Traffic statistics are available only if you use Courier; traffic cannot be counted if you use Dovecot.

These statistics are updated once per night.

4.7.6 Global Filters

(This tab is visible only for the ISPConfig *admin* user.)

In this section you can define Postfix whitelists, blacklists, content filters (header/body, etc.), and relay recipients.

4.7.6.1 Postfix Whitelist

The whitelist feature must be seen in conjunction with the blacklist feature. If you use the blacklist to block whole domains, for example, you can use the whitelist to allow certain email addresses (for example) from that domain.

To create a new whitelist record, click on the *Add new Whitelist record* button. This will lead you to the *Email Whitelist* form with the tab *Whitelist*.

Email Whitelist

Whitelist

The form has the following fields:

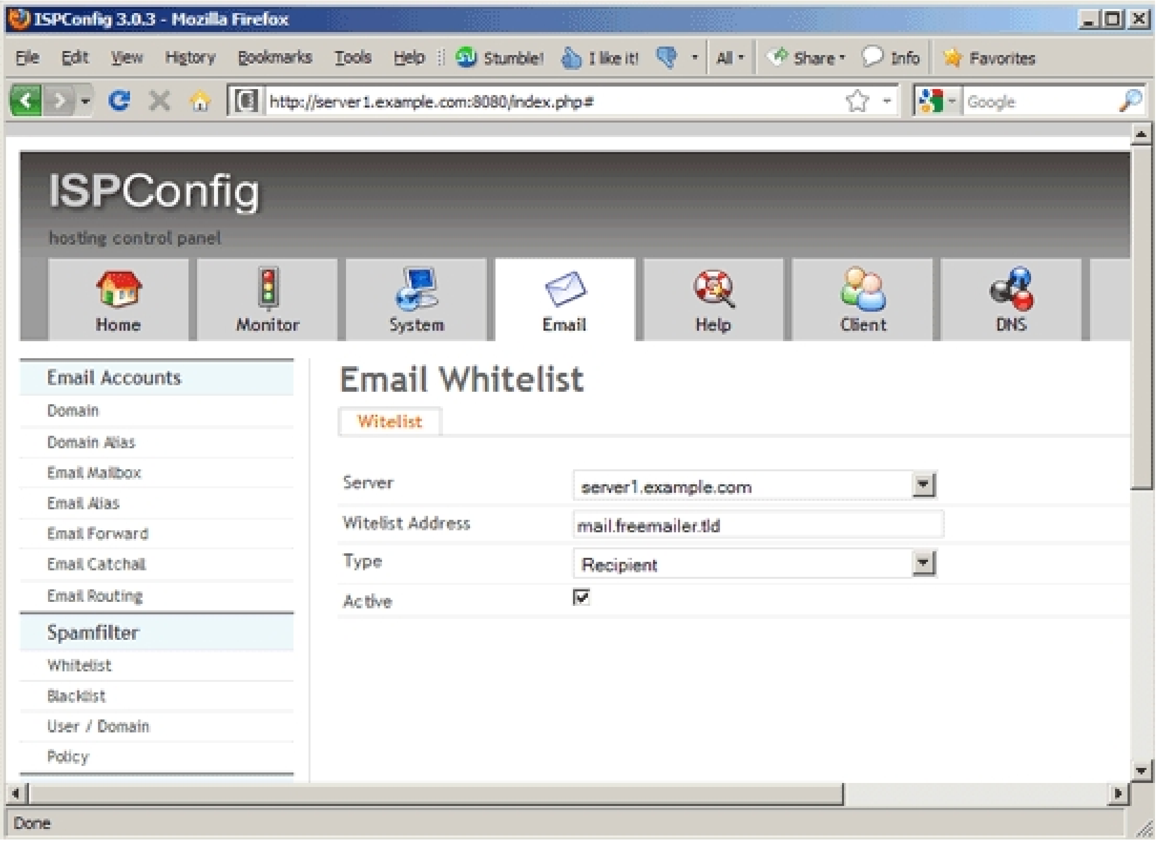
- *Server*: If more than one server is available, you can select the server on which the whitelist record will be located.
- *Whitelist Address*: Specify an email address, domain, parent domains, or localpart@. Exmaples: *user@somedomain.com*, *somedomain.com*, *mail.freemailer.tld*, *1.2.3.4*, *sales@*.
- *Type*: Select between *Recipient* (refers to the Postfix directive *smtpd_recipient_restrictions*) , *Sender* (refers to *smtpd_sender_restrictions*), and *Client* (refers to *smtpd_client_restrictions*).
  - *smtpd_recipient_restrictions*: SMTPD recipient restrictions will put restrictions on what

messages will be accepted into your server based on the recipient email address (RCPT TO:). Postfix will check whether the message sender (email address, domain, mail server) is included in the whitelist table. If the sender is listed, the mail is delivered. If the sender is not listed in the whitelist, the message is rejected with an error code of 554, *`Recipient address rejected: Access denied (in reply to RCPT TO command)`*. To whitelist the sending mail server, you can type in its hostname (e.g. *`mail.freemailer.tld`*) or IP address in the *`Whitelist Address`* field. You can find more details here: [***How To Whitelist Hosts/IP Addresses In Postfix***]

- *`smtpd_sender_restrictions`*: SMTPD sender restrictions will put restrictions on what addresses will be able to send mail through your server based on the sender email address (MAIL FROM:). You can use sender email addresses, domains, and localpart@ in the *`Whitelist Address`* field.

- *`smtpd_client_restrictions`*: SMTPD client restrictions will put restrictions on what systems will be able to send mail through your server based on the client IP and host information (name). For example, if you have a user whose client PC has the IP address *`1.2.3.4`*, you can put *`1.2.3.4`* in the *`Whitelist Address`* field.

- *`Active`*: This defines whether this whitelist record is active or not.



Email Whitelist

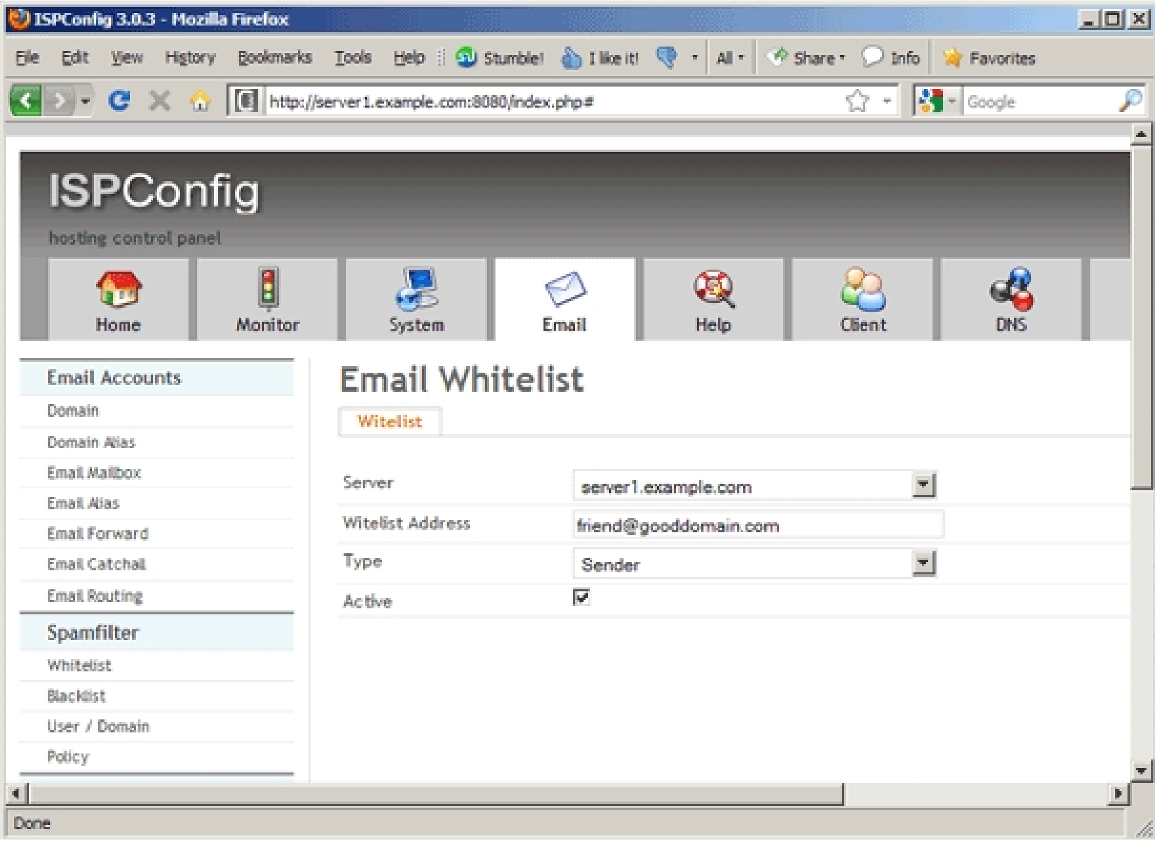
Witelist

Server: server1.example.com

Witelist Address: mail.freemailer.tld

Type: Recipient

Active ☑



Email Whitelist

Witelist

Server	server1.example.com
Witelist Address	friend@gooddomain.com
Type	Sender
Active	☑

4.7.6.2 Postfix Blacklist

The blacklist feature can be used to blacklist email addresses, domains, parent domains, or localpart@.

To create a new blacklist record, click on the `Add new Blacklist record` button. This will lead you to the `Email Blacklist` form with the tab `Blacklist`.

Email Blacklist

Blacklist

The form has the following fields:

- `Server`: If more than one server is available, you can select the server on which the blacklist record will be located.
- `Blacklist Address`: Specify an email address, domain, parent domains, or localpart@. Exmaples: `user@somedomain.com`, `somedomain.com`, `mail.freemailer.tld`, `1.2.3.4`, `sales@`.
- `Type`: Select between `Recipient` (refers to the Postfix directive `smtpd_recipient_restrictions`) , `Sender` (refers to `smtpd_sender_restrictions`), and `Client` (refers to `smtpd_client_restrictions`).

  - *smtpd_recipient_restrictions*: SMTPD recipient restrictions will put restrictions on what messages will be rejected based on the recipient email address (RCPT TO:). Postfix will check whether the message sender (email address, domain, mail server) is included in the blacklist table. If the sender is listed, the mail is rejected with an error code of 554, *Recipient address rejected: Access denied (in reply to RCPT TO command)*. To blacklist the sending mail server, you can type in its hostname (e.g. *mail.freemailer.tld*) or IP address in the *Blacklist Address* field. You can find more details here: **How To Whitelist Hosts/IP Addresses In Postfix**
  - *smtpd_sender_restrictions*: SMTPD sender restrictions will put restrictions on what addresses will be able to send mail through your server based on the sender email address (MAIL FROM:). You can use sender email addresses, domains, and localpart@ in the *Blacklist Address* field.
  - *smtpd_client_restrictions*: SMTPD client restrictions will put restrictions on what systems will be able to send mail through your server based on the client IP and host information (name). For example, if you have a user that sends out viruses (intended or unintended) and you know his client PC has the IP address *1.2.3.4*, you can put *1.2.3.4* in the *Blacklist Address* field.

- *Active*: This defines whether this blacklist record is active or not.

4.7.6.3 Content Filter

The content filter allows you to block emails based on their content, e.g. you can block emails that contain a certain string in the subject or in the body. Postfix supports a built-in filter mechanism that examines message header and message body content, one line at a time, before it is stored in the Postfix queue.

To create a new content filter, click on the *Add new Content Filter* button. This will lead you to the *Mail Content Filter* form with the tab *Filter*.

Mail Content Filter

Filter

The form contains the following fields:

- *Server*: If more than one server is available, you can select the server on which the content filter will be located.
- *Filter*: Select what part of the email message you want to inspect:
  - *Header Filter*: These are applied to initial message headers (except for the headers that are processed with *MIME-Header Filter*).

  - *MIME-Header Filter*: These are applied to MIME related message headers only.
  - *Nested-Header Filter*: These are applied to message headers of attached email messages (except for the headers that are processed with *MIME-Header Filter*).
  - *Body Filter*: These are applied to all other content, including multi-part message boundaries.

Note: message headers are examined one logical header at a time, even when a message header spans multiple lines. Body lines are always examined one line at a time.

- *Regexp. Pattern*: Fill in the search pattern. Usually the best performance is obtained with ***pcre*** (Perl Compatible Regular Expression), but the slower ***regexp*** (POSIX regular expressions) support is more widely available. Use the command *postconf -m* to find out what lookup table types your Postfix system supports - usually it will be ***regexp***. Here are a few examples:

Regexp. Pattern:	***Filter Type:***	***Explanation:***
/^Subject: .*Make Money Fast!/	Header Filter	Searches for the string *Make Money Fast!* in the Subject line.
/name=[^>]*.(bat|com|exe|dll)/	MIME-Header Filter	This will match all messages that have attachments whose files end in *.bat*, *.com*, *.exe* or *.dll*.
/^<iframe src=(3D)?cid:.* height=(3D)?0 width=(3D)?0>$/	Body Filter	Body pattern to stop a specific HTML browser vulnerability exploit.
/^From: joe@example.com/	Header Filter	Matches all messages sent by *joe@example.com*.
/^From: .*@example.com/	Header Filter	Matches all messages sent from the *example.com* domain.
/Real Bad Words/	Body Filter	This matches "real bad words" in any case (upper, lower, or mixed).
/^Date: .* 200[0-2]/	Header Filter	This matches all emails sent in the years 2000 - 2002.
/^Date: .* 19[0-9][0-9]/	Header Filter	This matches all emails sent between 1900 and 1999.
/^To: postmaster@yourdom.ain/	Header Filter	Matches all messages sent to *postmaster@yourdom.ain*.

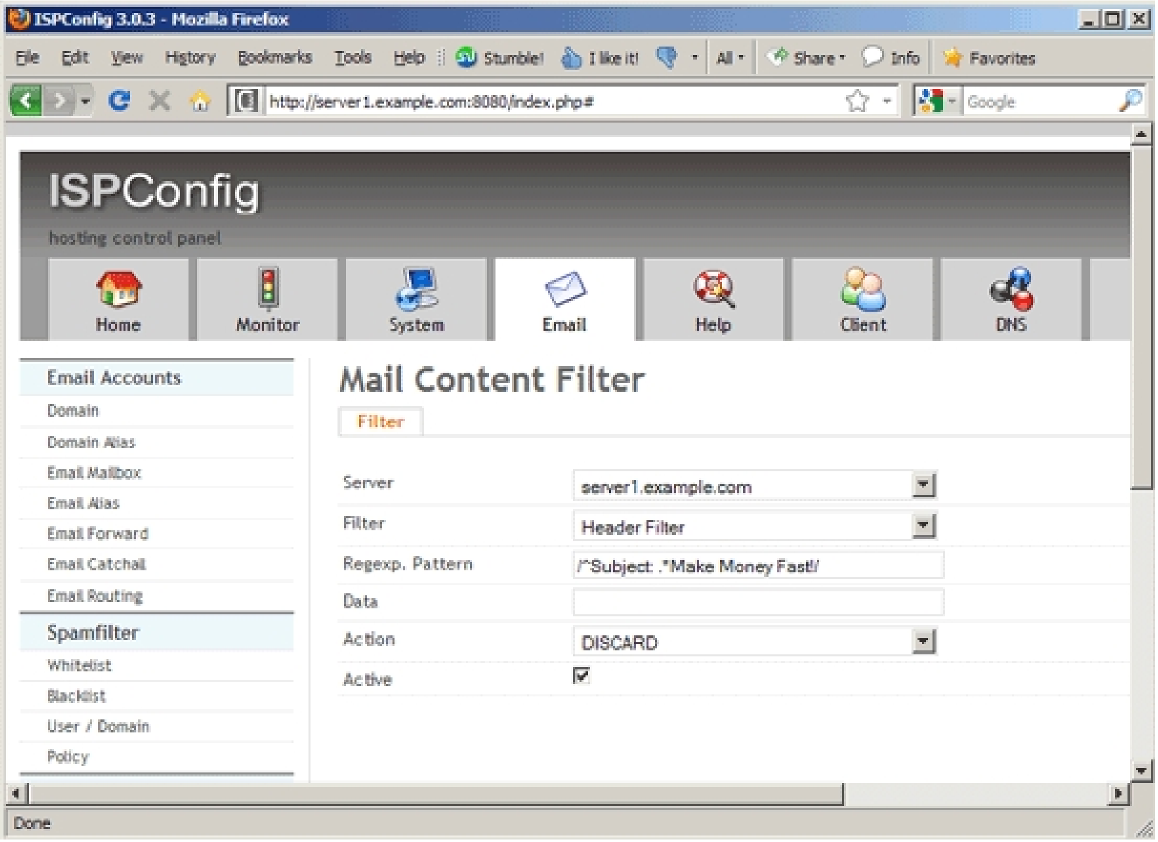
- *Data*: You can specify an action for each filter (see below). Some actions allow or require you to specify an additional text or destination. The *Data* field is where you place this information.
- *Action*: Here you can select what should happen to an email if a filter applies:

  - *DISCARD* (***optional text*** can be specified in the *Data* field): Claim successful delivery and silently discard the message. Log the optional text if specified, otherwise log a generic message.
  - *DUNNO*: Pretend that the input line did not match any pattern, and inspect the next input line. This action can be used to shorten the table search.
  - *FILTER* (***required transport:destination*** must be specified in the *Data* field): After the message is queued, send the entire message through the specified external content filter. The ***transport*** name specifies the first field of a mail delivery agent definition in master.cf; the syntax of the next-hop ***destination*** is described in the manual page of the corresponding delivery agent. More information about external content filters is in the Postfix ***FILTER_README*** file.
  - *HOLD* (***optional text*** can be specified in the *Data* field): Arrange for the message to be placed on the ***hold*** queue, and inspect the next input line. The message remains on ***hold*** until someone

either deletes it or releases it for delivery. Log the optional text if specified, otherwise log a generic message.

- `IGNORE`: Delete the current line from the input, and inspect the next input line.
- `PREPEND` (***required text*** must be specified in the `Data` field): Prepend one line with the specified text, and inspect the next input line.
- `REDIRECT` (***required user@domain*** must be specified in the `Data` field): Write a message redirection request to the queue file, and inspect the next input line. After the message is queued, it will be sent to the specified address instead of the intended recipient(s). Note: this action overrides the `FILTER` action, and affects all recipients of the message. If multiple `REDIRECT` actions fire, only the last one is executed.
- `REPLACE` (***required text*** must be specified in the `Data` field): Replace the current line with the specified text, and inspect the next input line.
- `REJECT` (***optional text*** can be specified in the `Data` field): Reject the entire message. Reply with ***optional text...*** when the optional text is specified, otherwise reply with a generic error message.
- `WARN` (***optional text*** can be specified in the `Data` field): Log a warning with the ***optional text...*** (or log a generic message), and inspect the next input line. This action is useful for debugging and for testing a pattern before applying more drastic actions.

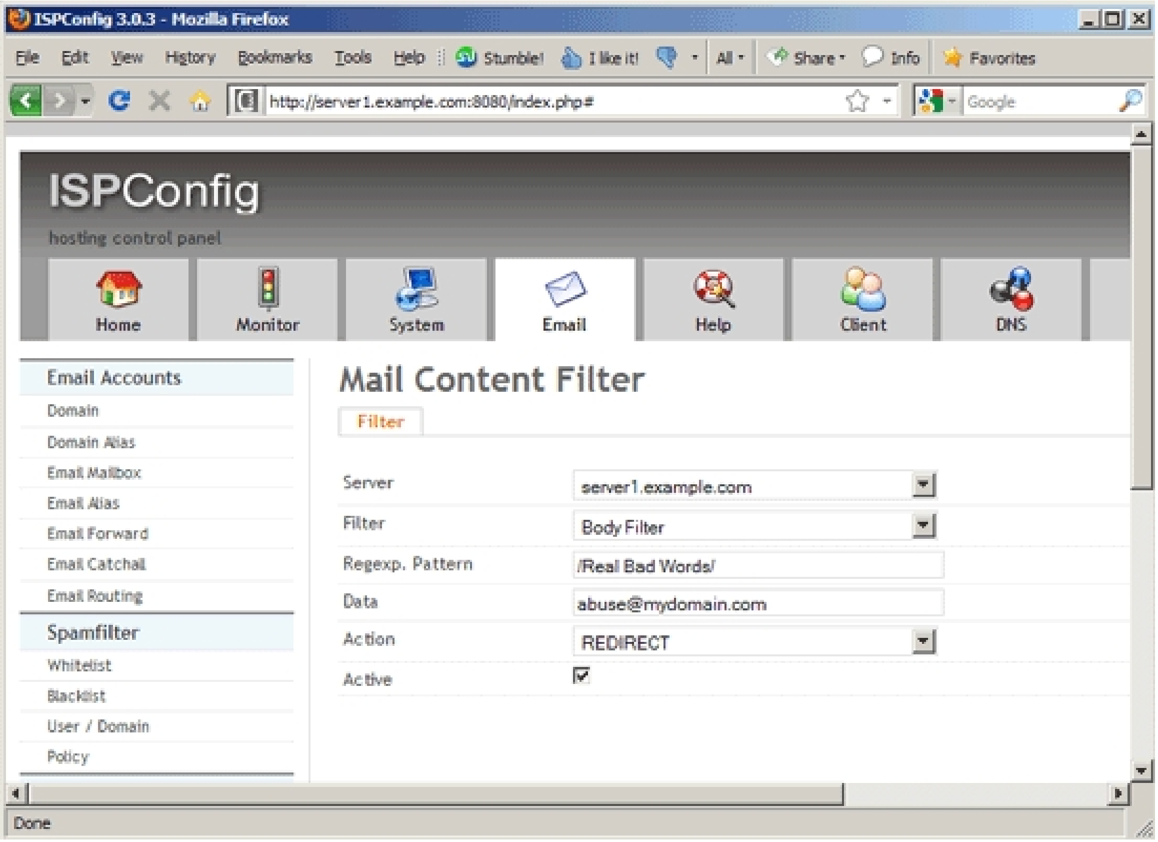
- `Active`: This defines whether this content filter is active or not.



Mail Content Filter

Filter

Server	server1.example.com
Filter	Header Filter
Regexp. Pattern	/^Subject: .*Make Money Fast!/
Data	
Action	DISCARD
Active	☑



Mail Content Filter

Filter

Server	server1.example.com
Filter	Body Filter
Regexp. Pattern	/Real Bad Words/
Data	abuse@mydomain.com
Action	REDIRECT
Active	☑

4.7.6.4 Relay Recipients

If you have created email transports under *Email > Email Accounts > Email Routing*, you must also create *Relay Recipients* so that the server knows that it should accept those emails before routing them to another server. If you have created a route for a single email address, you must create a relay recipient for that email address. If you have create a route for a whole domain, and you know all existing email accounts of that domain, it is recommended to create relay recipients for all these email addresses; if you don't know all the email addresses of a domain, or there are simply too many, you can create a relay recipient for the whole domain, but you should keep in mind that the destination server can become a source of ***backscatter*** in this case because if a mail is sent to a non-existing address of the domain, the forwarding server will route it to its destination server, and because the destination server doesn't know that email address, it might send a bounce.

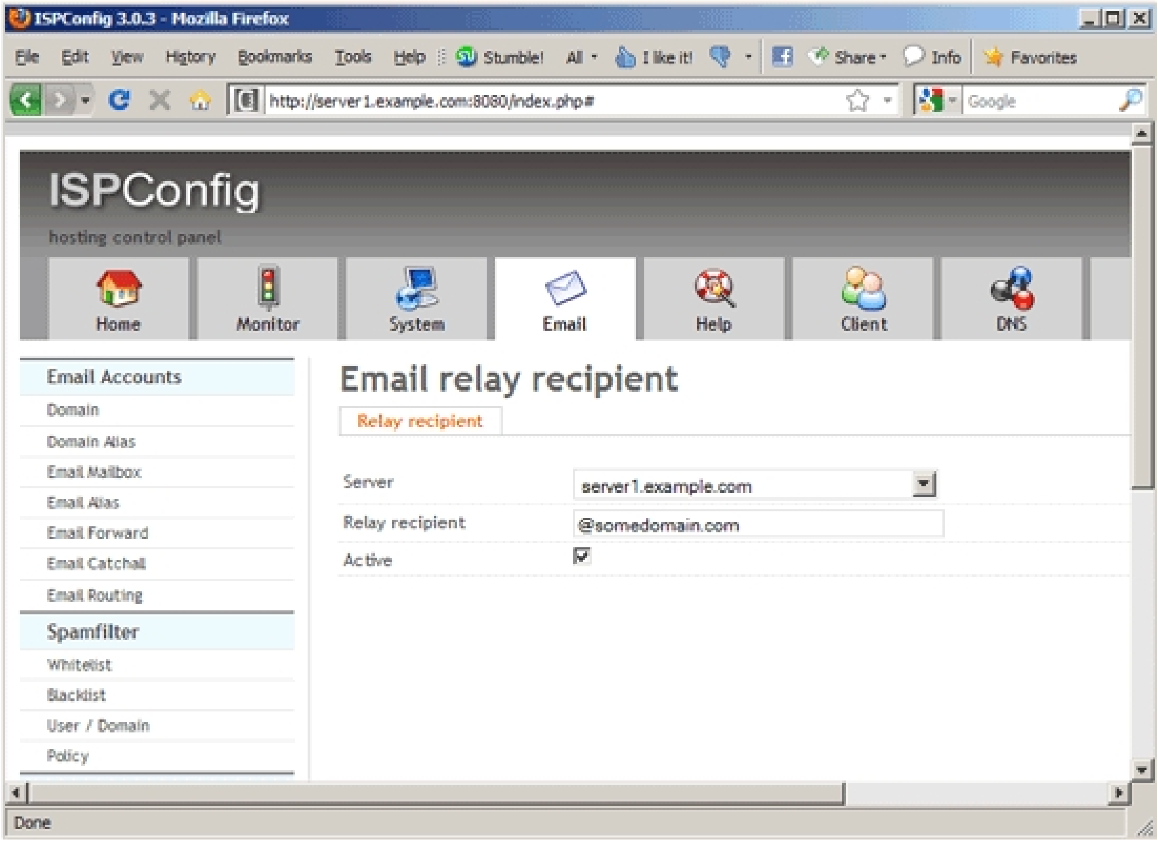
To create a new relay recipient, click on the *Add new relay recipient* button. This will lead you to the *Email relay recipient* form with the tab *Relay recipient*.

Email relay recipient

Relay recipient

The form has the following fields:

- *Server*: If more than one server is available, you can select the server on which the relay recipient will be located.
- *Relay recipient*: Fill in the email address or email domain, e.g. *user@example.com* or *@example.com*.
- *Active*: This defines whether this relay recipient is active or not.



Email relay recipient

Relay recipient

Server	server1.example.com
Relay recipient	@somedomain.com
Active	☑

4.8 DNS

On this tab you can create zones and DNS records for your domains. You can either do this by using the DNS Wizard (*DNS > DNS Wizard > Add DNS Zone*) which will automatically create a set of common DNS records for your domain (like *www*, *mail*, *ns* records, etc.), or you create the zones and records manually under *DNS > DNS > Zones* - you will also have to go there if you want to create further DNS records that are not created by the DNS Wizard.

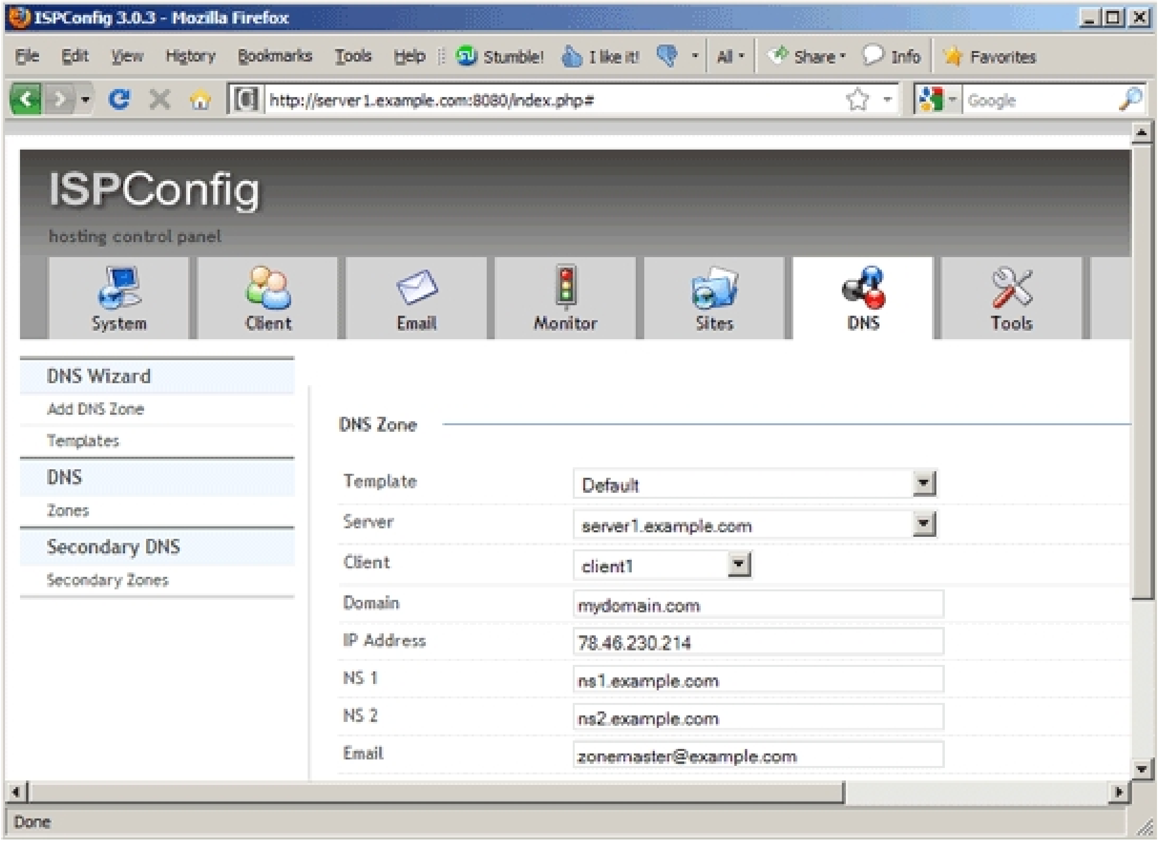
4.8.1 DNS Wizard

4.8.1.1 Add DNS Zone

This is the wizard to create a new DNS zone. The form has the following fields:

- *Template*: This refers to the templates that exist under *DNS > DNS Wizard > Templates*. These templates define what records will be created by default if you use the DNS Wizard. Let's assume we create a zone for the domain *example.com* - the *Default* template will create A records for *example.com*, *www.example.com*, and *mail.example.com*, two NS (nameserver) records, plus an MX (mail exchanger) record for *example.com* that points to *mail.example.com*.

- `Server`: If more than one server is available, you can select the server on which the DNS zone will be located.
- `Client`: Here you select the client that owns the new DNS zone.
- `Domain`: Fill in the domain for which you want to create the zone, e.g. `example.com` - please note that you don't need a dot at the end, i.e., `example.com.` would work as well, but `example.com` (without the trailing dot) is sufficient.
- `IP Address`: Fill in the IP address that `example.com` should point to - please note that `www.example.com` and `mail.example.com` will also point to that IP address (you can change that later on under `DNS > DNS > Zones`).
- `NS 1`: Specify the hostname of the primary nameserver for the domain, e.g. `ns1.somedomain.com`. Again, no trailing dot is needed. `ns1.somedomain.com` must point to the server that you selected in the `Server` field.
- `NS 2`: Specify the hostname of the secondary nameserver for the domain, e.g. `ns2.somedomain.com`. Again, no trailing dot is needed.
- `Email`: Specify the email address of the zone administrator, e.g. `zonemaster@somedomain.com`.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
ISPConfig
hosting control panel
System Client Email Monitor Sites DNS Tools
DNS Wizard
Add DNS Zone
Templates
DNS
Zones
Secondary DNS
Secondary Zones
Done

DNS Zone

Template	Default
Server	server1.example.com
Client	client1
Domain	mydomain.com
IP Address	78.46.230.214
NS 1	ns1.example.com
NS 2	ns2.example.com
Email	zonemaster@example.com

4.8.1.2 Import Zone File

This form allows you to import an existing BIND zone file that you have as a text file on your client computer, like this one:

```
$ORIGIN example.org.
$TTL       4000
@      IN      SOA     ns1.example.com. zonemaster.example.com. (
                       2011101901       ; serial, todays date + todays serial #
                       7200             ; refresh, seconds
                       540              ; retry, seconds
                       604800           ; expire, seconds
                       86400 )          ; minimum, seconds
;

example.org. A        1.2.3.4
example.org. 3600      MX    10   mail.example.org.
example.org. 3600      NS        ns1.example.com.
example.org. 3600   IN   NS        ns2.example.com.
mail 3600 A        1.2.3.4
www 3600 A        1.2.3.4
example.org. TXT "v=spf1 a mx ptr -all"
```

It supports *NS*, *A*, *AAAA*, *CNAME*, *HINFO*, *MX*, *NAPTR*, *PTR*, *RP*, *SRV*, and *TXT* records.

The form has the following fields:

- *Server*: If more than one server is available, you can select the server on which the DNS zone will be located.
- *Client*: Here you select the client that owns the imported DNS zone.
- *Domain*: It is recommended that you fill in the domain for which you want to create the zone, e.g. *example.com* - please note that you don't need a dot at the end, i.e., *example.com.* would work as well, but *example.com* (without the trailing dot) is sufficient. If you don't fill in the domain here, ISPConfig tries to read the domain from the *$ORIGIN* line of your zone file, and if that doesn't exist, from the *SOA* line. If the *SOA* line doesn't contain the domain name (for example because it begins with an *@*), ISPConfig generates the domain name from the zone file name. For example, if your zone file is named *example.com.txt* or *example.com.zone*, the domain name would be *example.com*. But if your zone name is named *pri.example.com.txt* or doesn't even contain your domain name, ISPConfig would get a wrong value from the file name. Therefore it is recommended to fill in the domain name in this field.
- *Zone File*: Select the zone file to upload from your local hard drive.

After a successful import, you can find the zone and its records under *DNS > Zones* (see chapter ***4.8.2.1 Zones***). Please note that ISPConfig replaces the original *NS* records with the correct new values, depending on which server you choose in the *Server* field.

4.8.1.3 Templates

Here you can create templates for the DNS Wizard. A template defines what records will be created by default if a new zone is created with the DNS Wizard.

To create a new template, click on the *Add new record* button. This will lead you to the *DNS Wizard template* form with the tab *DNS Template*.

DNS Wizard template

DNS Template

The form contains the following fields:

- *Name*: Specify a name for the template.
- *Fields*: Here you can select what fields will be visible in the DNS Wizard form (*Domain*, *IP Address*, *NS 1*, *NS 2*, *Email*). For example, if you decide to hard-code the nameservers and the zonemaster email address into the template, it doesn't make sense to show those fields in the DNS Wizard.
- *Template*: Fill in your template. As an example, here is the *Default* template:

```
[ZONE]
          origin={DOMAIN}.
          ns={NS1}.
          mbox={EMAIL}.
          refresh=28800
          retry=7200
          expire=604800
          minimum=86400
          ttl=86400
[DNS_RECORDS]
            A|{DOMAIN}.|{IP}|0|86400
            A|www|{IP}|0|86400
            A|mail|{IP}|0|86400
            NS|{DOMAIN}.|{NS1}.|0|86400
            NS|{DOMAIN}.|{NS2}.|0|86400
            MX|{DOMAIN}.|mail.{DOMAIN}.|10|86400
```

As you see, a template consists out of two stanzas, *[ZONE]* and *[DNS_RECORDS]*.

In the *[ZONE]* stanza, you secify values for *origin*, *ns1*, *mbox*, *refresh*, *retry*, *expire*, *minimum*, and *ttl* in the form *name=value*.

- *origin*: The name of this zone. Make sure you use a trailing dot, e.g. *example.com.* or

{DOMAIN}.

- ns: The name of the name server that is the original or primary source of data for this zone. Make sure you use a trailing dot.
- *mbox*: A name which specifies the mailbox of the person responsible for this zone. If you don't use the *{EMAIL}* placeholder, this should be specified in the mailbox-as-domain-name format where the *@* character is replaced with a dot, e.g. *zonemaster.example.com.* (for *zonemaster@example.com*). Make sure you use a trailing dot.
- *refresh*: The number of seconds after which slave nameservers should check to see if this zone has been changed. If the zone's serial number has changed, the slave nameserver initiates a zone transfer.
- *retry*: This specifies the number of seconds a slave nameserver should wait before retrying if it attmepts to transfer this zone but fails.
- *expire*: If for ***expire*** seconds the primary server cannot be reached, all information about the zone is invalidated on the secondary servers (i.e., they are no longer authoritative for that zone).
- *minimum*: The minimum TTL field that should be exported with any record from this zone. If any record has a lower TTL, this TTL is sent instead.
- *ttl*: The number of seconds that this zone may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the zone should not be cached.

In the *[DNS_RECORDS]* stanza, you specify all records that should be created by default, one record per line. A line has the following format:

type|name|data|aux|ttl

As you see, there are five fields, separated by a pipe character (*|*). This is the meaning of the five fields:

- *type*: The type of record (*A*, *AAAA*, *ALIAS*, *CNAME*, *HINFO*, *MX*, *NS*, *PTR*, *RP*, *SRV*, *TXT*).
  - *A*: An IPv4 host address. The *data* column should contain the IP address (in numbers-and-dots format) associated with the *name*.

Example: *192.168.1.88*

  - *AAAA*: An IPv6 host address. The *data* column should contain the IPv6 address associated with the *name*.

Example: *3ffe:b00:c18:3::a*

  - *ALIAS*: A server side alias. An alias is like a *CNAME*, only it is handled entirely by the server. The *data* column should contain the hostname aliased by *name*. Aliases can be used in place of *A* records. The client will only see *A* records and will not be able to tell that aliases are involved. The hostname specified by *data* must exist in the database. It can be useful to use aliases for everything. Use *A* records for the canonical name of the machine and use aliases for any additional names. This is especially useful when combined with automatic *PTR* records. If a single IP address is only used for one *A* record, then there will never be any confusion over what the *PTR* record should be.

Example: *albuquerque.example.com.* (FQDN)
Example: *albuquerque* (hostname only)

- *CNAME*: The canonical name for an alias. The *data* column should contain the real name of the machine specified by *name*. *data* may be a hostname or an FQDN.

Example: *porcini.example.com.* (FQDN)
Example: *porcini* (hostname only)

- *HINFO*: Host information. The *data* column should contain two strings which provide information about the host specified by *name*. The first string specifies the CPU type, and the second string describes the operating system type. The two strings should be separated by a space. If either string needs to contain a space, enclose it in quotation marks.

Example: *"Pentium Pro" Linux*

- *MX*: Mail exchanger. The *data* column should contain the hostname or FQDN of a mail server which will accept mail for the host specified by *name*. The *aux* column should contain a preference for this mail server. Mail transfer agents prefer MX records with lower values in *aux*.

Example: *mail.example.com.* (FQDN)
Example: *mail* (hostname only)

- *NS*: An authoritative nameserver. The *data* column should contain the hostname or FQDN of a server which should be considered authoritative for the zone listed in *name*.

Example: *ns1.example.com.* (FQDN)
Example: *ns1* (hostname only)

- *PTR*: A domain name pointer. These records, used only with IN-ADDR.ARPA zones, should contain the canonical hostname of the machine referred to by *name* in *data*.

Example: *webserver.example.com.*

- *RP*: A responsible person. The *data* column should contain the DNS-encoded email address of the person responsible for the name requested, then a space, then a hostname that should return a TXT record containing additional information about the responsible person. If there is no such TXT record, the second value should contain a dot (*.*).

Example: *webmaster.example.com. contactinfo.example.com.*

- *SRV*: Server location. Specifies the location of the server(s) for a specific protocol and domain. The *data* column must contain three space-separated values. The first value is a number specifying the ***weight*** for this entry. The second field is a number specifying the ***port*** on the target host of this service. The last field is a name specifying the ***target*** host. The *aux* column should contain the ***priority*** of this target host. Targets with a lower priority are preferred.

For more information, read ***RFC 2782***.

Example: *0 9 server.example.com.* (FQDN)

Example: *0 9 server* (hostname only)

- *TXT*: A text string. The *data* column contains a text string that is returned only when a TXT query is issued for the host specified by *name*. TXT records can be used for ***SPF records***.

Example: *This is a string.*

Example: *v=spf1 a mx ptr -all* (SPF record)

- *name*: The name that this record describes. Wildcard values such as *** or **.sub* are supported, and this field can contain an FQDN or just a hostname. If you specify an FQDN, the name must end with a dot; if you specify just a hostname, it must not end with a dot. It may contain out-of-zone data if this is a glue record.

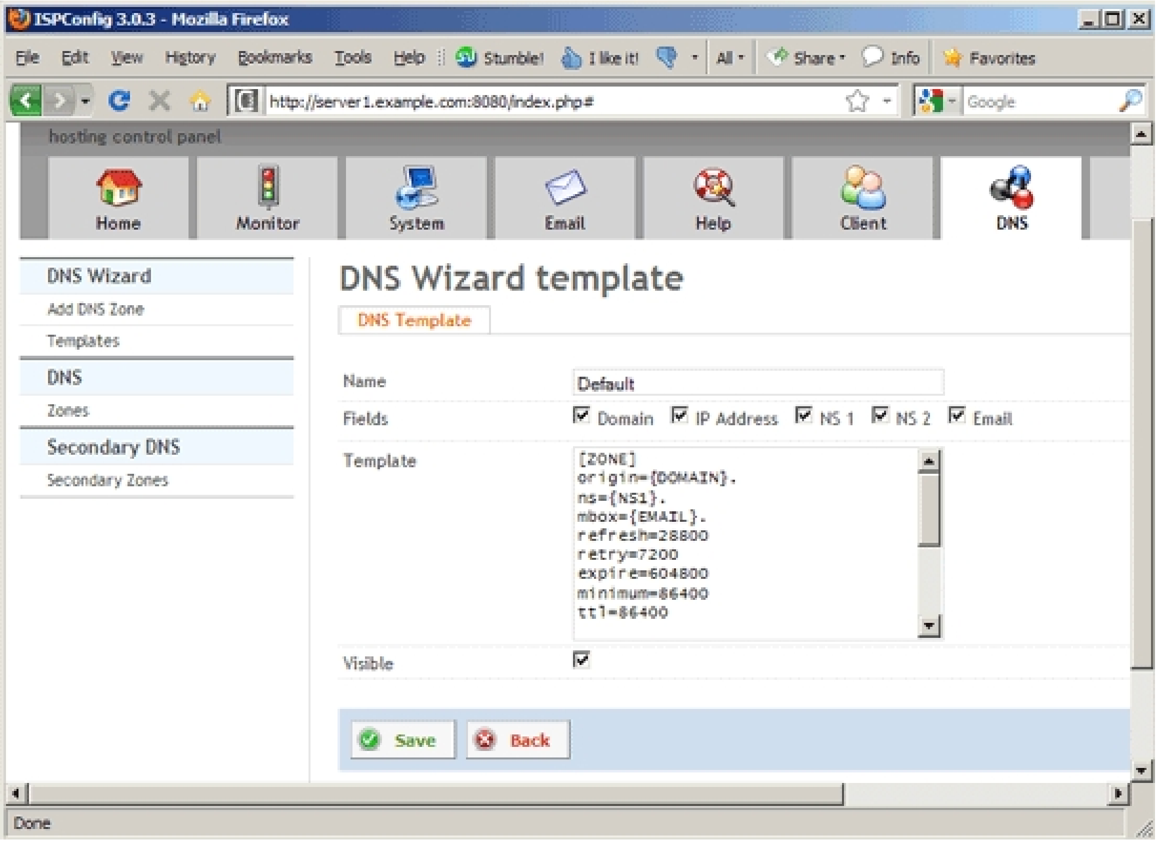
Examples:

- foo
- foo.example.com.
- {DOMAIN}.
- www

- *data*: The data associated with this record, e.g. an IP address for *A* records, a hostname/FQDN for *CNAME*/*MX*/*NS* records, etc. Please note that an *MX* record must always point to a hostname/FQDN that has an *A* record - *CNAME* records are not allowed.
- *aux*: An auxillary numeric value in addition to *data*. For *MX* records, this field specifies the preference. For *SRV* records, this field specifies the priority. Specify *0* for all other records.
- *ttl*: The time interval (in seconds) that this record may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the record can only be used for the transaction in progress, and should not be cached.

The following placeholders are available in a template and will be replaced with the value of the corresponding field in the DNS Wizard: *{DOMAIN}*, *{IP}*, *{NS1}*, *{NS2}*, and *{EMAIL}*.

- *Visible*: This defines whether this template is visible (i.e., can be selected) in the DNS Wizard or not.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://server1.example.com:8080/index.php#
hosting control panel
Home Monitor System Email Help Client DNS
DNS Wizard
Add DNS Zone
Templates
DNS
Zones
Secondary DNS
Secondary Zones
Save Back
Done

DNS Wizard template

DNS Template

Name	Default
Fields	☑ Domain ☑ IP Address ☑ NS 1 ☑ NS 2 ☑ Email
Template	(see below)
Visible	☑

```
[ZONE]
origin={DOMAIN}.
ns={NS1}.
mbox={EMAIL}.
refresh=28800
retry=7200
expire=604800
minimum=86400
ttl=86400
```

4.8.2 DNS

4.8.2.1 Zones

Here you can create DNS zones manually (if you are experienced enough with DNS and don't want to use the DNS Wizard) and modify existing DNS zones (that were created, for example, with the DNS Wizard).

To create a new DNS zone, click on the *`Add new DNS Zone manually`* button. This will lead you to the *`DNS Zone`* form with the tabs *`DNS Zone`* and *`Records`*.

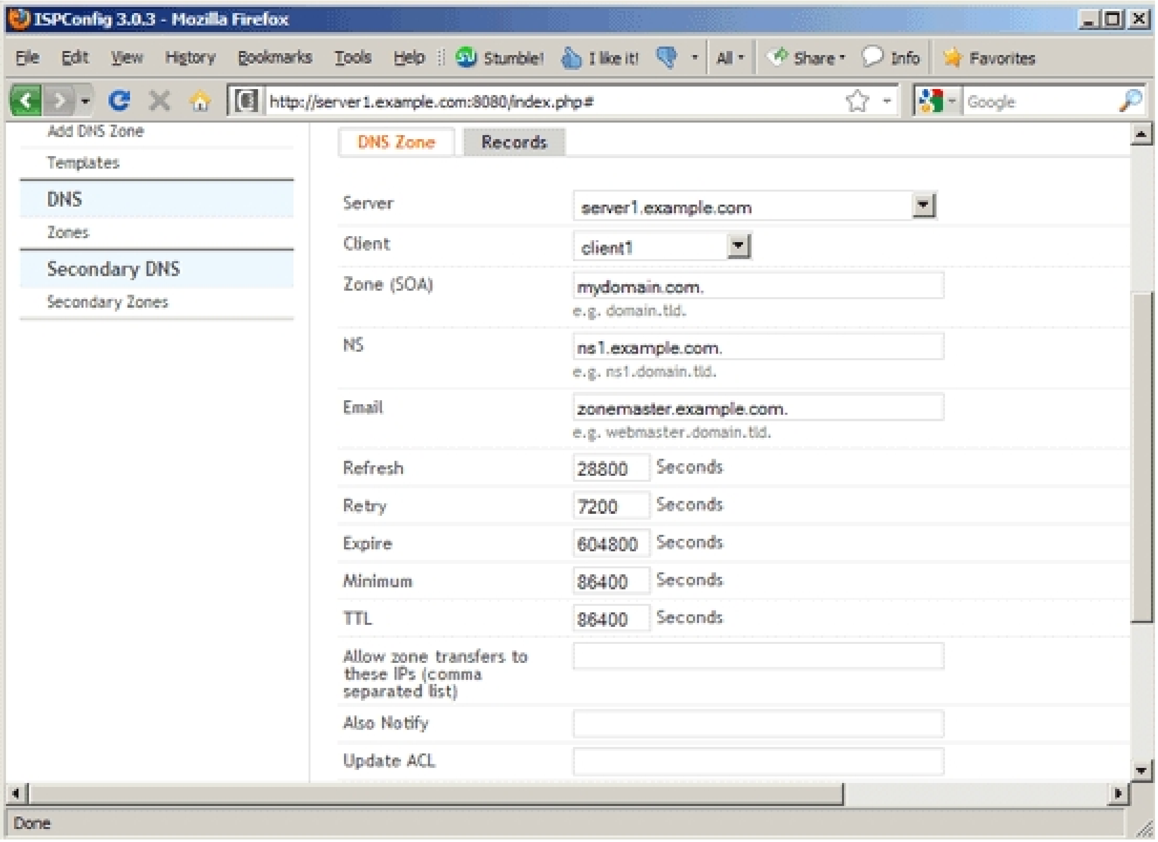
DNS Zone

DNS Zone

On this tab you specify the SOA (***s***tart ***o***f ***a***uthority) record. It contains authoritative information about a DNS zone, including the primary name server, the email of the domain administrator, the domain serial number, and several timers relating to refreshing the zone.

The form contains the following fields:

- *Server*: If more than one server is available, you can select the server on which the DNS zone will be located.
- *Client*: Here you select the client that owns the new DNS zone.
- *Zone (SOA)*: Fill in the domain for which you want to create the zone, e.g. *example.com.* - please note that other than in the DNS Wizard you need a dot at the end.
- *NS*: Specify the hostname of the primary nameserver for the domain, e.g. *ns1.somedomain.com.* - again, a trailing dot is needed. *ns1.somedomain.com* must point to the server that you selected in the *Server* field.
- *Email*: Specify the email address of the zone administrator. This should be specified in the mailbox-as-domain-name format where the *@* character is replaced with a dot, e.g. *zonemaster.somedomain.com.* - again, you need a trailing dot.
- *Refresh*: The number of seconds after which slave nameservers should check to see if this zone has been changed. If the zone's serial number has changed, the slave nameserver initiates a zone transfer.
- *Retry*: This specifies the number of seconds a slave nameserver should wait before retrying if it attmepts to transfer this zone but fails.
- *Expire*: If for expire seconds the primary server cannot be reached, all information about the zone is invalidated on the secondary servers (i.e., they are no longer authoritative for that zone).
- *Minimum*: The minimum TTL field that should be exported with any record from this zone. If any record has a lower TTL, this TTL is sent instead.
- *TTL*: The number of seconds that this zone may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the zone should not be cached.
- *Allow zone transfers to these IPs (comma separated list)*: This field can contain one or more IP addresses separated by commas. These IP addresses will be allowed to connect to the server to transfer the zone. If no IP is specified, no server is allowed to connect. Usually, you should list your slave DNS servers for this zone here.
- *Also Notify*: This optional field should contain one or more IP addresses separated by commas. These IP addresses will be used to send NOTIFY messages to additional name servers. Notification is sent to all name servers that have NS records in the zone plus any mentioned in this field.
- *Update ACL*: This is an optional specifying the ACL (***a***ccess ***c***ontrol ***l***ist) controlling who can update a zone. You can specify one or more IP addresses separated by commas. This field is useful if the zone contains dynamic IP addresses and you want to allow dynamic DNS updates from a client. If no IP is specified, then dynamic DNS updates are disabled.
- *Active*: This defines whether this DNS zone is active or not.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
Add DNS Zone
Templates
DNS
Zones
Secondary DNS
Secondary Zones
DNS Zone
Records
Done

Server	server1.example.com
Client	client1
Zone (SOA)	mydomain.com. e.g. domain.tld.
NS	ns1.example.com. e.g. ns1.domain.tld.
Email	zonemaster.example.com. e.g. webmaster.domain.tld.
Refresh	28800 Seconds
Retry	7200 Seconds
Expire	604800 Seconds
Minimum	86400 Seconds
TTL	86400 Seconds
Allow zone transfers to these IPs (comma separated list)	
Also Notify	
Update ACL	

Records

On this tab you can create the following types of records:

- A
- AAAA
- ALIAS
- CNAME
- HINFO
- MX
- NS
- PTR
- RP
- SRV
- TXT

A Records

An A record is an IPv4 host address. The *`IP-Address`* field should contain the IP address (in numbers-and-dots format) associated with the *`Hostname`*.

Example: *`192.168.1.88`*

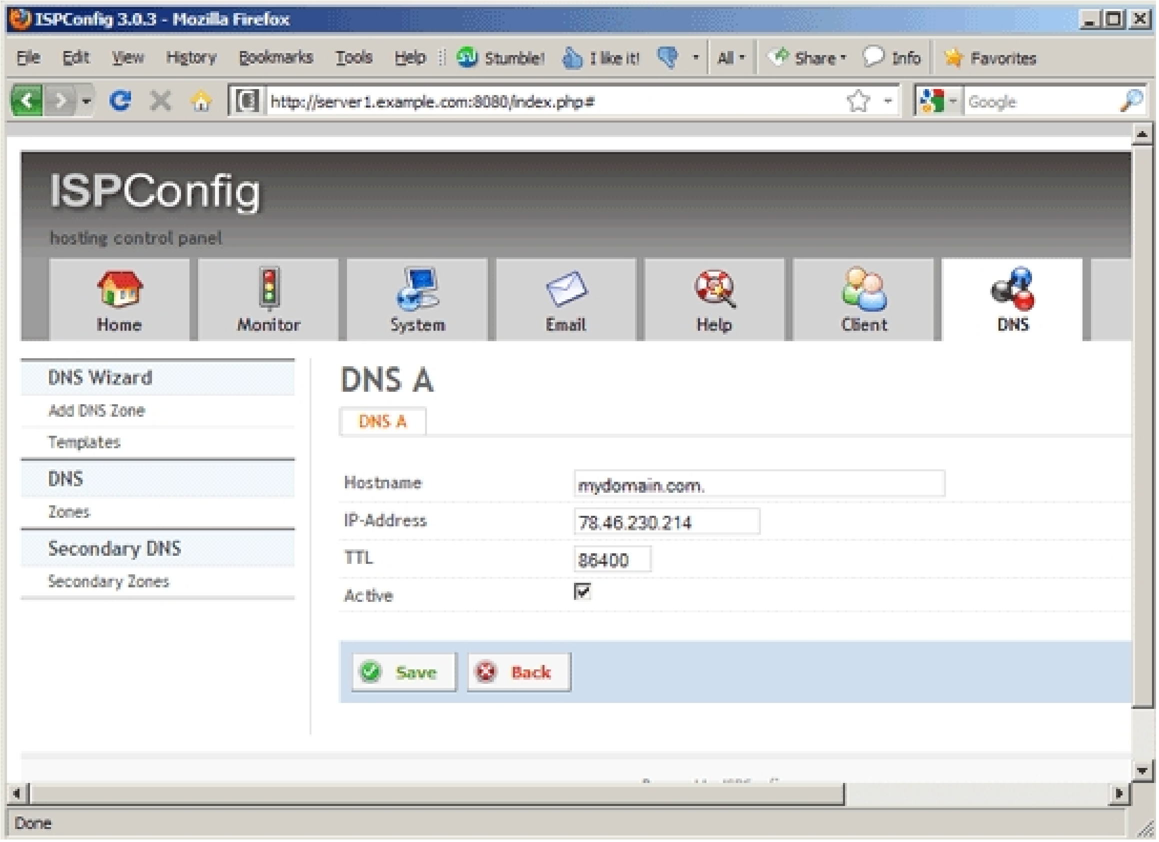
The form contains the following fields:

- *`Hostname`*: The name that this record describes. Wildcard values such as *`*`* or *`*.sub`* are supported, and this field can contain an FQDN or just a hostname. If you specify an FQDN, the name must end with a dot; if you specify just a hostname, it must not end with a dot.

Examples:

  - foo
  - foo.example.com.
  - www
  - example.com.
  - You can also leave the field empty which has the same meaning as if you'd fill in *`example.com.`*

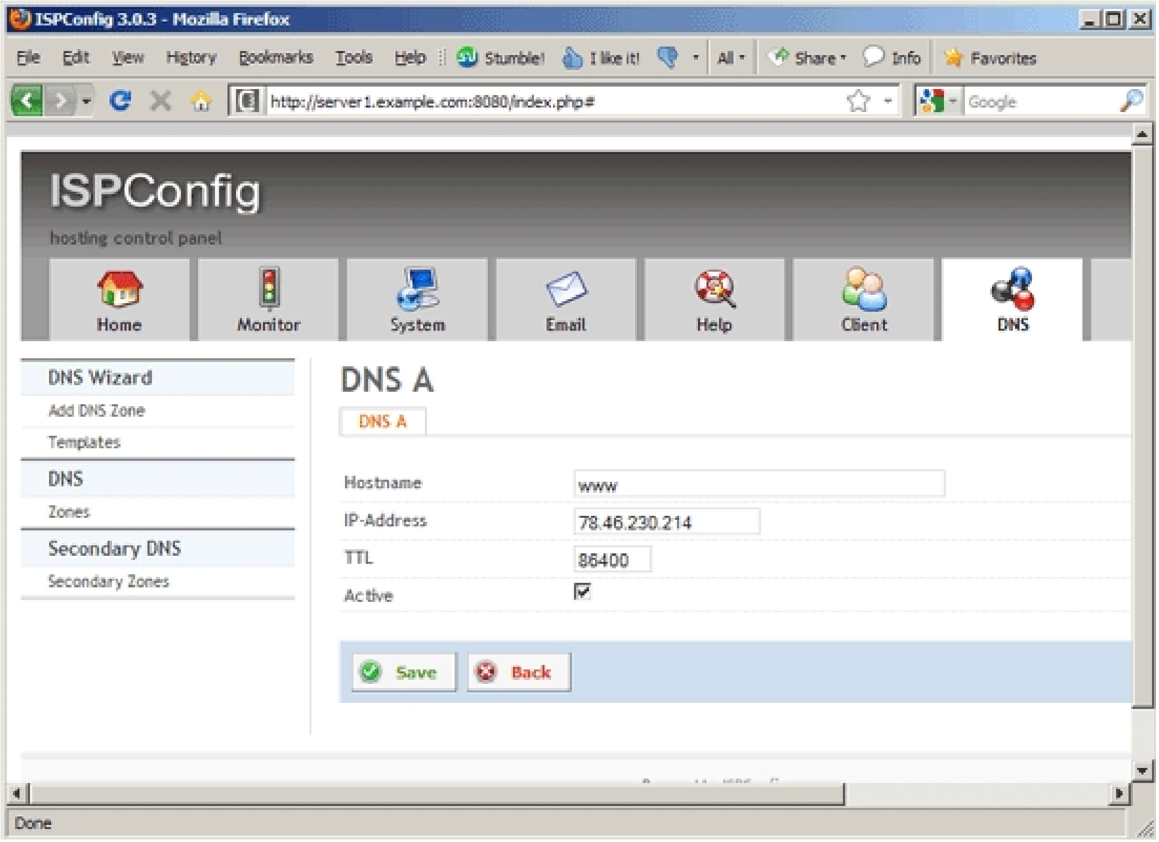
- *`IP-Address`*: Fill in the IPv4 IP address that the hostname should point to. Example: *`192.168.1.88`*
- *`TTL`*: The time interval (in seconds) that this record may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the record can only be used for the transaction in progress, and should not be cached.
- *`Active`*: This defines whether this A record is active or not.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
ISPConfig
hosting control panel
Home
Monitor
System
Email
Help
Client
DNS
DNS Wizard
Add DNS Zone
Templates
DNS
Zones
Secondary DNS
Secondary Zones
Save
Back
Done

DNS A

DNS A

Hostname	mydomain.com.
IP-Address	78.46.230.214
TTL	86400
Active	☑


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
ISPConfig
hosting control panel
Home
Monitor
System
Email
Help
Client
DNS
DNS Wizard
Add DNS Zone
Templates
DNS
Zones
Secondary DNS
Secondary Zones
Save
Back
Done

DNS A

DNS A

Hostname	www
IP-Address	78.46.230.214
TTL	86400
Active	☑

AAAA Records

An AAAA record is an IPv6 host address. The *`IPv6-Address`* field should contain the IPv6 address associated with the *`Hostname`*.

Example: *`3ffe:b00:c18:3::a`*

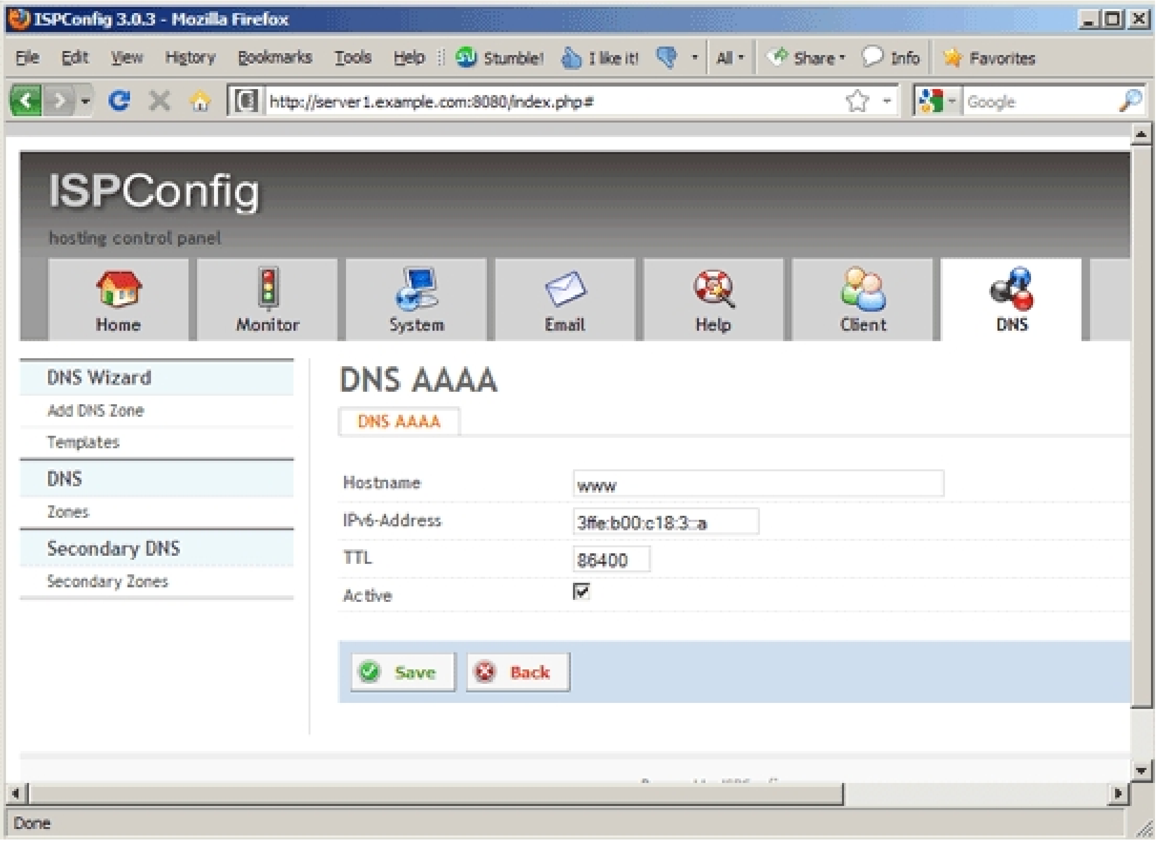
The form contains the following fields:

- *`Hostname`*: The name that this record describes. Wildcard values such as *`*`* or *`*.sub`* are supported, and this field can contain an FQDN or just a hostname. If you specify an FQDN, the name must end with a dot; if you specify just a hostname, it must not end with a dot.

Examples:

- foo
- foo.example.com.
- www
- example.com.
- You can also leave the field empty which has the same meaning as if you'd fill in *`example.com.`*

- *`IPv6-Address`*: Fill in the IPv6 IP address that the hostname should point to. Example: *`3ffe:b00:c18:3::a`*
- *`TTL`*: The time interval (in seconds) that this record may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the record can only be used for the transaction in progress, and should not be cached.
- *`Active`*: This defines whether this AAAA record is active or not.



DNS AAAA

DNS AAAA

Hostname: www

IPv6-Address: 3ffe:b00:c18:3::a

TTL: 86400

Active: ☑

ALIAS Records

(Please note the ALIAS records are supported by the MyDNS name server, but not by the BIND name server. If you use BIND, ALIAS records are identical to CNAME records, i.e., if you create an ALIAS record, actually a CNAME record will be created.)

An ALIAS record is a server side alias. An alias is like a CNAME, only it is handled entirely by the server. The *`Target Hostname`* field should contain the hostname aliased by *`Hostname`*. Aliases can be used in place of A records. The client will only see A records and will not be able to tell that aliases are involved. The target hostname must exist in the database. It can be useful to use aliases for everything. Use A records for the canonical name of the machine and use aliases for any additional names. This is especially useful when combined with automatic PTR records. If a single IP address is only used for one A record, then there will never be any confusion over what the PTR record should be.

Example: *`albuquerque.example.com.`* (FQDN)
Example: *`albuquerque`* (hostname only)

The field contains the following fields:

- *`Hostname`*: The name that this record describes. Wildcard values such as *`*`* or *`*.sub`* are supported, and this field can contain an FQDN or just a hostname. If you specify an FQDN, the name must end with a dot; if you specify just a hostname, it must not end with a dot.

Examples:

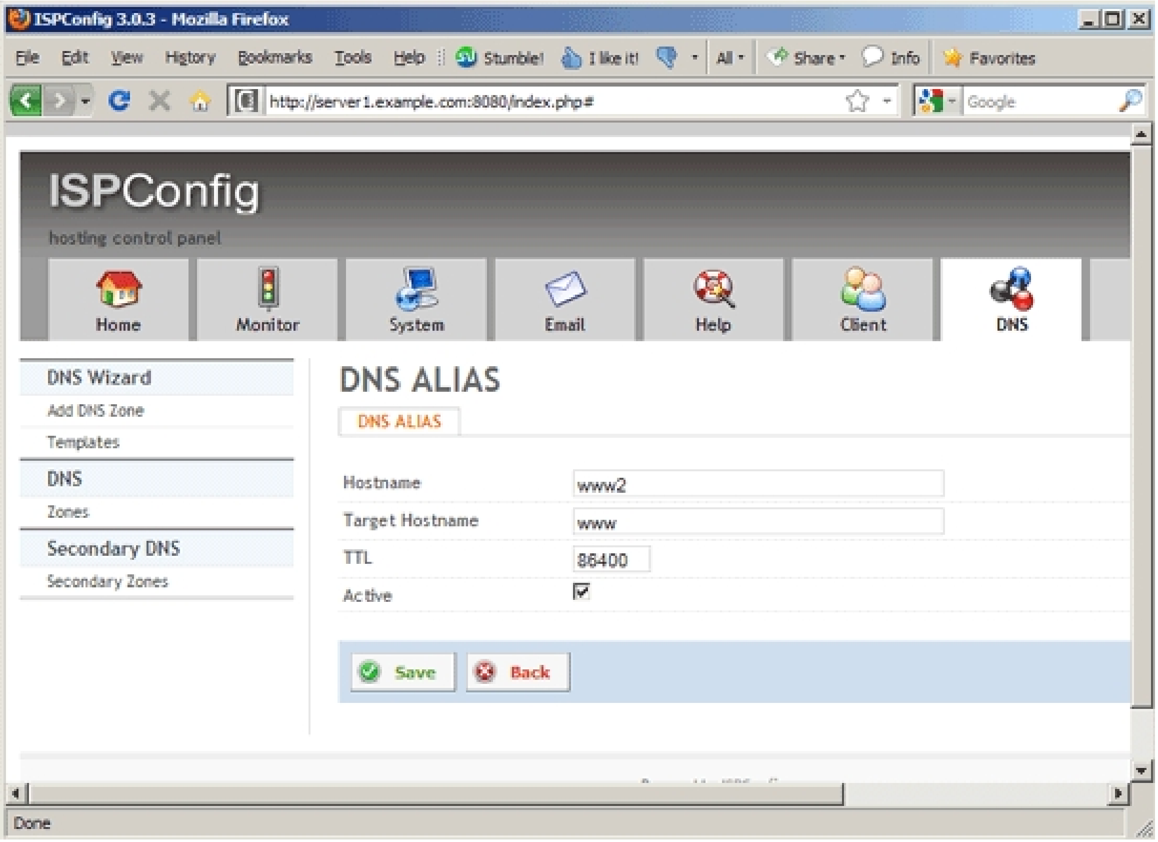
    - foo
    - foo.example.com.
    - www
    - example.com.
    - You can also leave the field empty which has the same meaning as if you'd fill in *`example.com`*.

- *`Target Hostname`*: The hostname that is aliased by the hostname in the *`Hostname`* field. Wildcard values such as *`*`* or *`*.sub`* are supported, and this field can contain an FQDN or just a hostname. If you specify an FQDN, the name must end with a dot; if you specify just a hostname, it must not end with a dot.

Examples:

    - albuquerque
    - albuquerque.example.com.

- *`TTL`*: The time interval (in seconds) that this record may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the record can only be used for the transaction in progress, and should not be cached.
- *`Active`*: This defines whether this ALIAS record is active or not.



DNS ALIAS

DNS ALIAS

Hostname: www2

Target Hostname: www

TTL: 86400

Active: ☑

CNAME Records

A CNAME record is the canonical name for an alias. The *`Target Hostname`* field should contain the real name of the machine specified by *`Hostname`*. *`Target Hostname`* may be a hostname or an FQDN.

Example: *`porcini.example.com.`* (FQDN)
Example: *`porcini`* (hostname only)

The field contains the following fields:

- *`Hostname`*: The name that this record describes. Wildcard values such as *`*`* or *`*.sub`* are supported, and this field can contain an FQDN or just a hostname. If you specify an FQDN, the name must end with a dot; if you specify just a hostname, it must not end with a dot.

Examples:

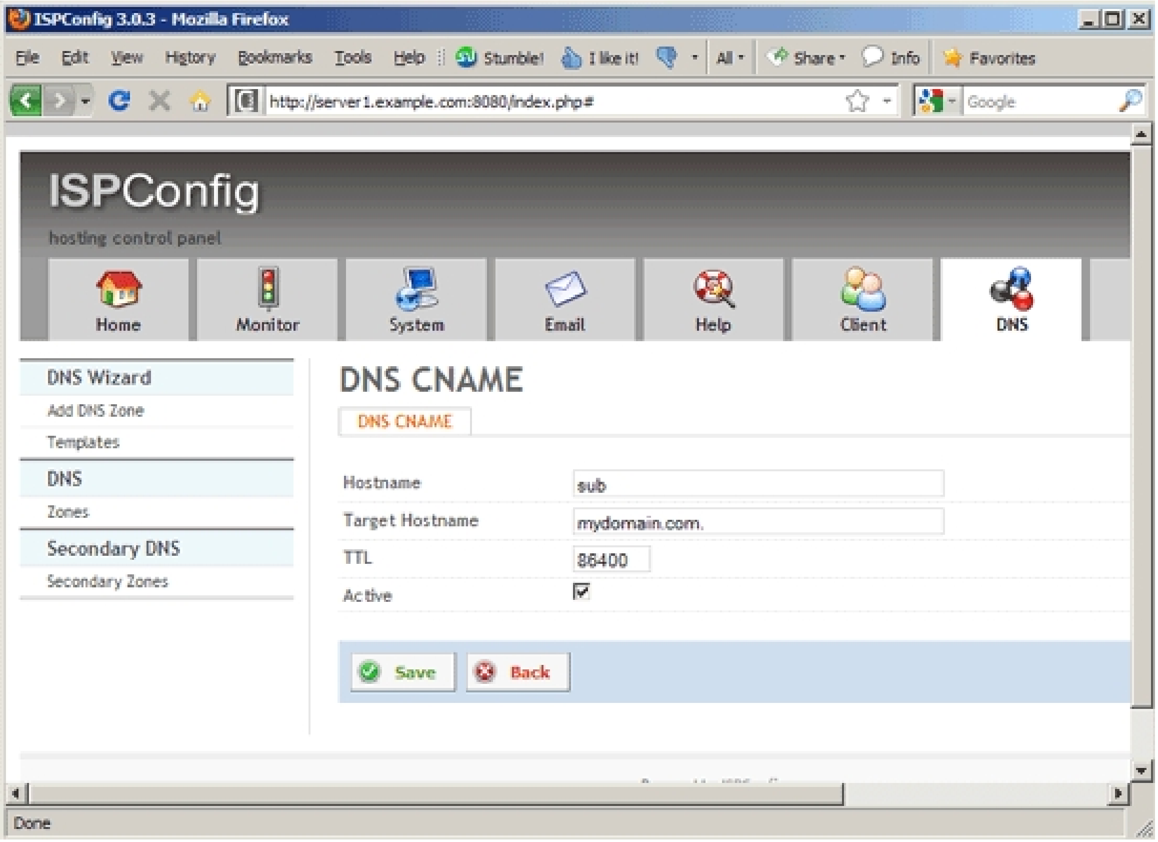
- foo
- foo.example.com.
- www
- example.com.
- You can also leave the field empty which has the same meaning as if you'd fill in *`example.com.`*

- *Target Hostname*: The real name of the machine that the hostname in the *Hostname* field points to. Wildcard values such as *** or **.sub* are supported, and this field can contain an FQDN or just a hostname. If you specify an FQDN, the name must end with a dot; if you specify just a hostname, it must not end with a dot.

Examples:

- porcini
- porcini.example.com.

- *TTL*: The time interval (in seconds) that this record may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the record can only be used for the transaction in progress, and should not be cached.
- *Active*: This defines whether this CNAME record is active or not.



DNS CNAME

DNS CNAME

Hostname	sub
Target Hostname	mydomain.com.
TTL	86400
Active	☑

HINFO Records

A HINFO record contains host information. The *`Host Information`* field should contain two strings which provide information about the host specified by *`Hostname`*. The first string specifies the CPU type, and the second string describes the operating system type. The two strings should be separated by a space. If either string needs to contain a space, enclose it in quotation marks.

Example: *`"Pentium Pro" Linux`*

The form contains the following fields:

- *`Hostname`*: The name that this record describes. Wildcard values such as *`*`* or *`*.sub`* are supported, and this field can contain an FQDN or just a hostname. If you specify an FQDN, the name must end with a dot; if you specify just a hostname, it must not end with a dot.

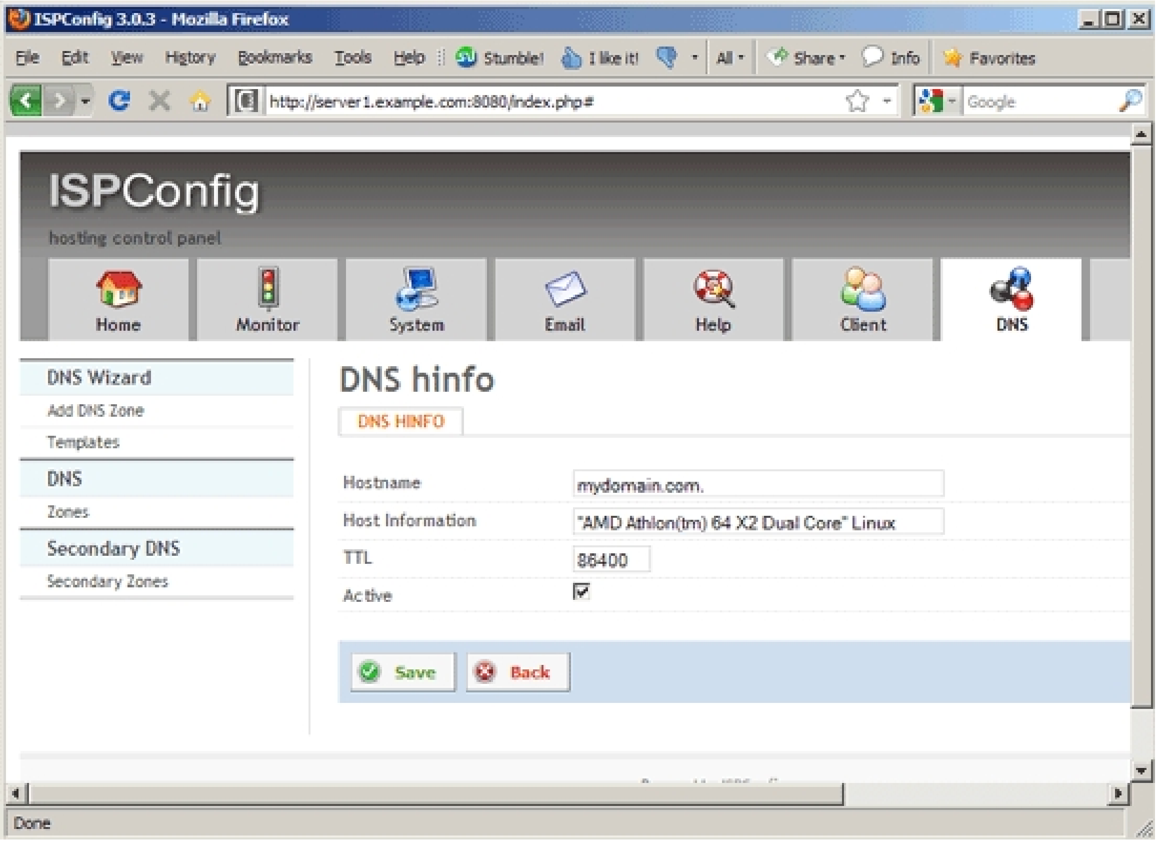
Examples:

  - foo
  - foo.example.com.
  - www
  - example.com.
  - You can also leave the field empty which has the same meaning as if you'd fill in *`example.com`*.

- *`Host Information`*: Specify two strings which provide information about the host specified by *`Hostname`*. The first string specifies the CPU type, and the second string describes the operating system type. The two strings should be separated by a space. If either string needs to contain a space, enclose it in quotation marks.

Example: *`"Pentium Pro" Linux`*

- *`TTL`*: The time interval (in seconds) that this record may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the record can only be used for the transaction in progress, and should not be cached.
- *`Active`*: This defines whether this HINFO record is active or not.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
ISPConfig
hosting control panel
Home
Monitor
System
Email
Help
Client
DNS
DNS Wizard
Add DNS Zone
Templates
DNS
Zones
Secondary DNS
Secondary Zones
DNS HINFO
Save
Back
Done

DNS hinfo

Hostname	mydomain.com.
Host Information	"AMD Athlon(tm) 64 X2 Dual Core" Linux
TTL	86400
Active	☑

MX Records

An MX record describes the mail exchanger for a domain or hostname. The `Mailserver hostname` field should contain the hostname or FQDN of a mail server which will accept mail for the host specified by `Hostname`. The `Priority` field should contain a preference for this mail server. Mail transfer agents prefer MX records with lower values in `Priority`.

Example: `mail.example.com.` (FQDN)
Example: `mail` (hostname only)

The form contains the following fields:

- `Hostname`: The name that this record describes. Wildcard values such as `*` or `*.sub` are supported, and this field can contain an FQDN or just a hostname. If you specify an FQDN, the name must end with a dot; if you specify just a hostname, it must not end with a dot. If you want email addresses of the form `user@example.com`, you must fill in `example.com.` in the `Hostname` field (or leave it empty); if you want email addresses of the form `user@sub.example.com`, you must fill in `sub` or `sub.example.com.` in the `Hostname` field.

Examples:

- foo

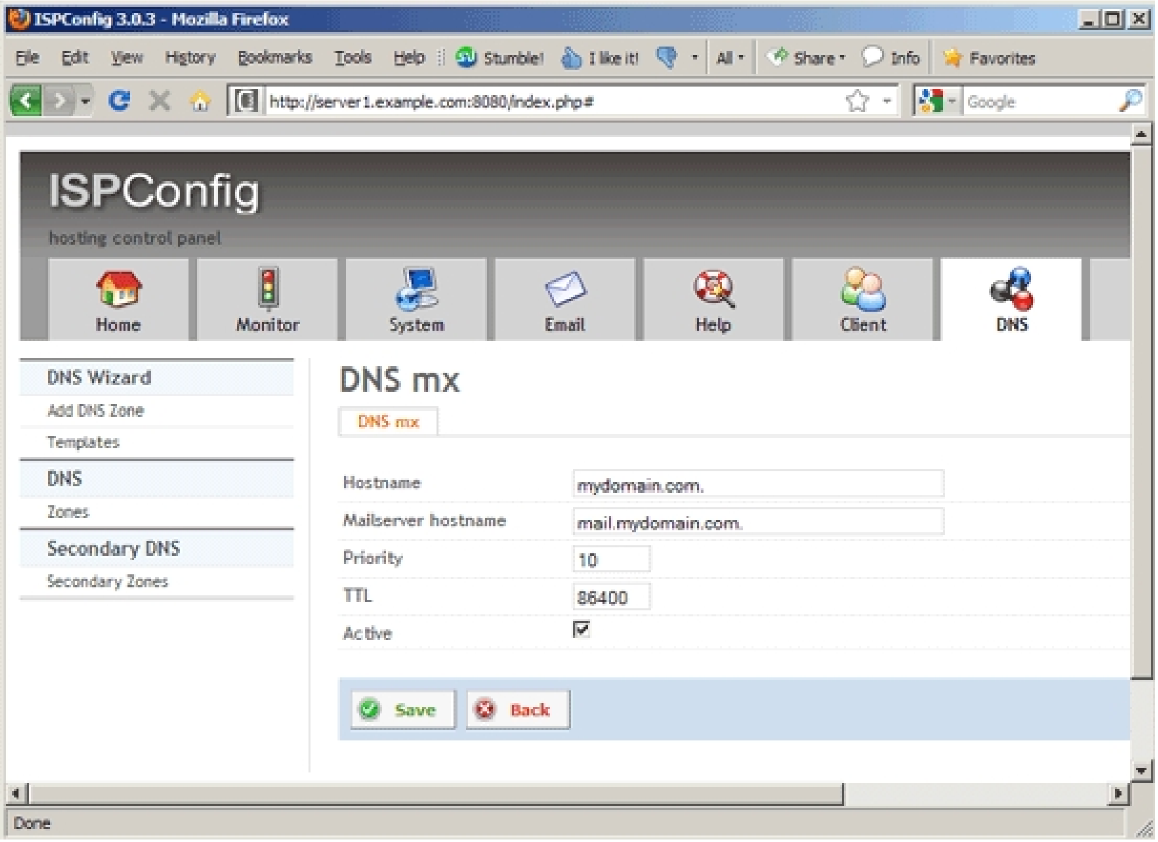
- foo.example.com.
- www
- example.com.
- You can also leave the field empty which has the same meaning as if you'd fill in *example.com.*

- *Mailserver hostname*: The *Mailserver hostname* field should contain the hostname or FQDN of a mail server which will accept mail for the host specified by *Hostname*. Please note that this *Mailserver hostname* must always be an A record - CNAME records are not allowed.

Examples:

- *mail.example.com.* (FQDN)
- *mail* (hostname only)

- *Priority*: The *Priority* field should contain a preference for this mail server, usually between *0* and *100*. Mail transfer agents prefer MX records with lower values in *Priority*.
- *TTL*: The time interval (in seconds) that this record may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the record can only be used for the transaction in progress, and should not be cached.
- *Active*: This defines whether this MX record is active or not.



Hostname	mydomain.com.
Mailserver hostname	mail.mydomain.com.
Priority	10
TTL	86400
Active	☑

NS Records

An NS record describes an authoritative nameserver of a zone. A zone can have more than one authoritative nameserver (usually it has at least two so that if one nameserver fails, the zone can still be resolved from the other nameserver), so there can be multiple NS records. The *Nameserver Hostname* field should contain the hostname or FQDN of a server which should be considered authoritative for the zone listed in *Zone*.

Example: *ns1.example.com.* (FQDN)
Example: *ns1* (hostname only)

The form contains the following fields:

- *Zone*: Fill in the name of the zone, i.e., the domain.

Examples:

- example.com.
- You can also leave the field empty which has the same meaning as if you'd fill in *example.com.*

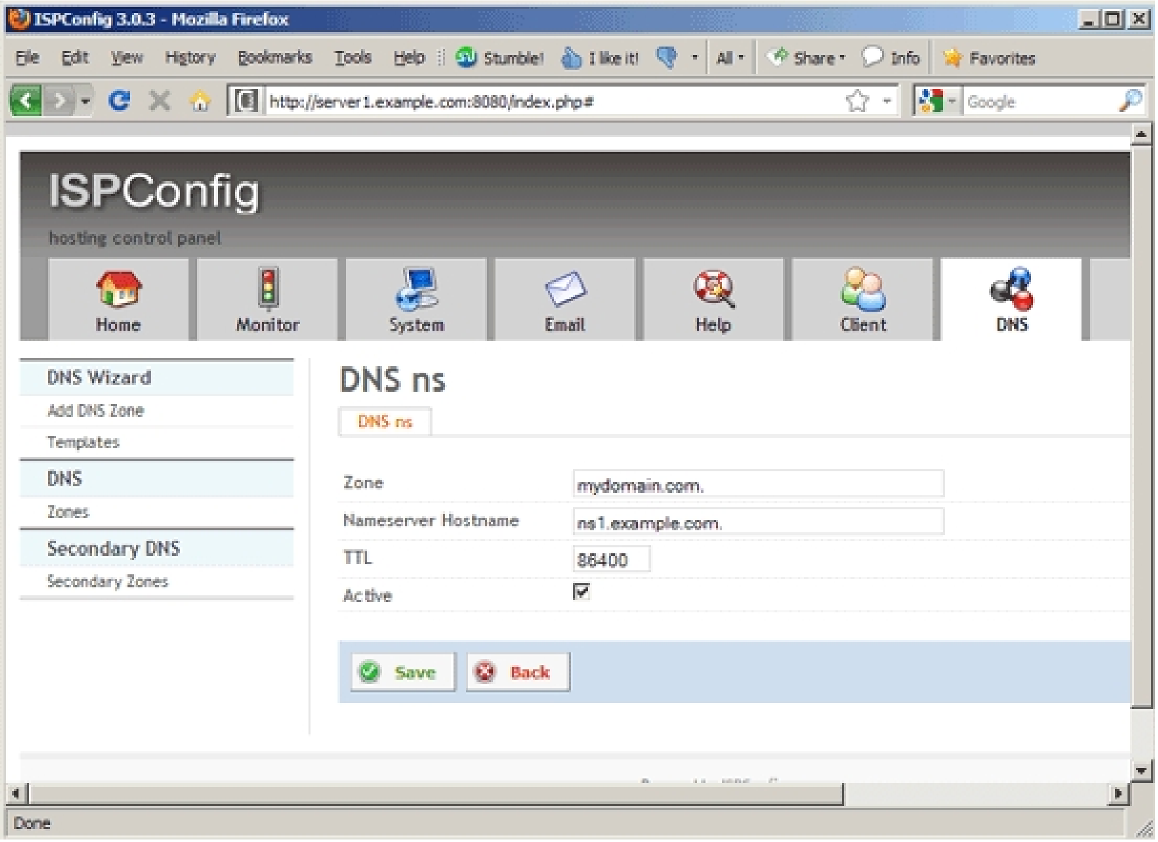
- *Nameserver Hostname*: The *Nameserver Hostname* field should contain the hostname or FQDN of a server which should be considered authoritative for the zone listed in *Zone*.

Examples:

- `ns1.somedomain.com.` (FQDN)
- `ns1.example.com.` (FQDN)
- `ns1` (hostname only)

If the nameserver is in the same zone (i.e., if the zone is `example.com.` and you fill in `ns1.example.com.` or just `ns1` in the `Nameserver Hostname` field), you also need a [***glue record***]() which you can usually create at your domain registrar.

- `TTL`: The time interval (in seconds) that this record may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the record can only be used for the transaction in progress, and should not be cached.
- `Active`: This defines whether this NS record is active or not.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
ISPConfig
hosting control panel
Home
Monitor
System
Email
Help
Client
DNS
DNS Wizard
Add DNS Zone
Templates
DNS
Zones
Secondary DNS
Secondary Zones
DNS ns
DNS ns
Save
Back
Done

Zone: mydomain.com.

Nameserver Hostname: ns1.example.com.

TTL: 86400

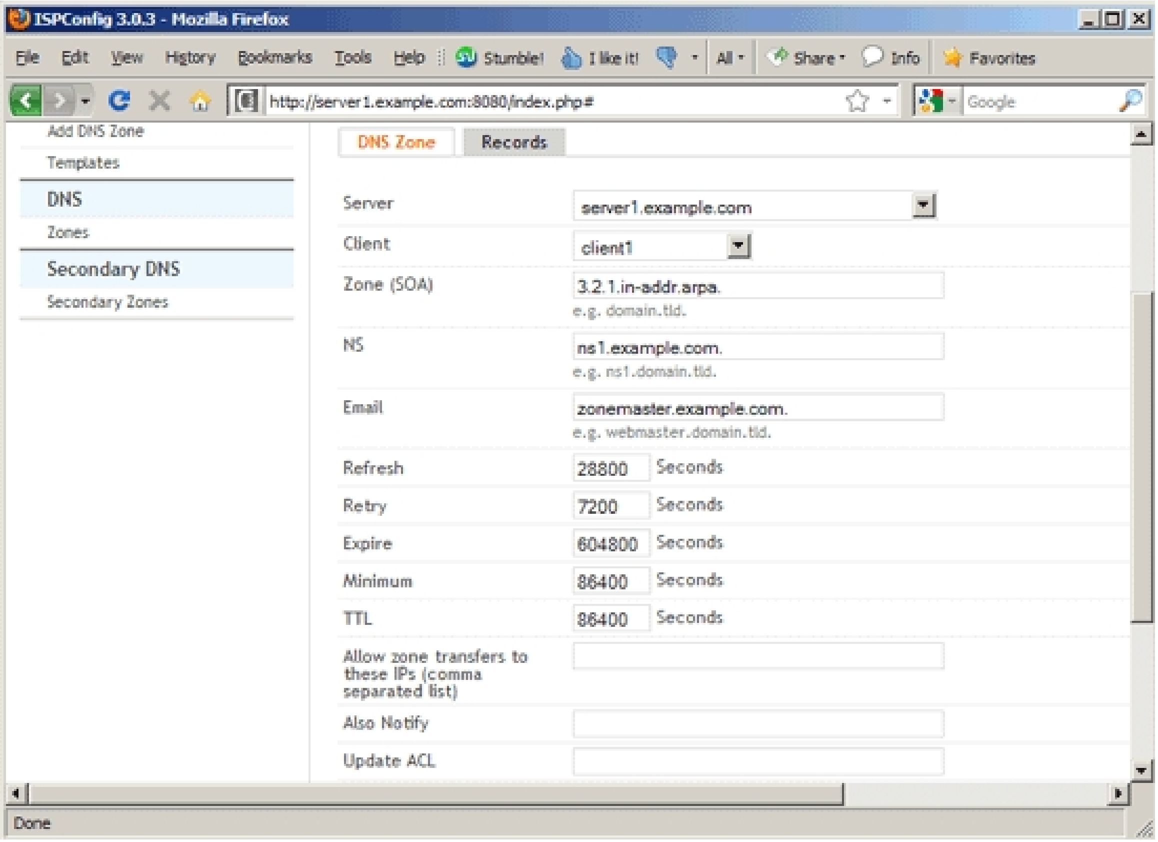
Active: ☑

PTR Records

A PTR record is a domain name pointer, i.e., it is used to point from an IP address to a domain or hostname. This is used for [***reverse DNS lookups***](). These records, used only with IN-ADDR.ARPA zones, should contain the canonical hostname of the machine referred to in the `Canonical Hostname` field. Usually the administrator of an IP address/subnet (i.e., your ISP or hoster) creates these for you (or gives you a web interface where you can configure this yourself), so in most cases you can ignore this feature in ISPConfig (unless you're the administrator of your own IP addresses).

Example: *`webserver.example.com.`*

Now let's assume you're the administrator of the IP subnet *`1.2.3/255.255.255.0`* and want to create a PTR record for the IP address *`1.2.3.4`* that should point to *`www.example.com`*. First you create the DNS zone *`3.2.1.in-addr.arpa`* (*`3.2.1`* is our *`1.2.3`* subnet in reverse order) in ISPConfig...


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
Add DNS Zone
Templates
DNS
Zones
Secondary DNS
Secondary Zones
DNS Zone
Records
Done

Server	server1.example.com
Client	client1
Zone (SOA)	3.2.1.in-addr.arpa. e.g. domain.tld.
NS	ns1.example.com. e.g. ns1.domain.tld.
Email	zonemaster.example.com. e.g. webmaster.domain.tld.
Refresh	28800 Seconds
Retry	7200 Seconds
Expire	604800 Seconds
Minimum	86400 Seconds
TTL	86400 Seconds
Allow zone transfers to these IPs (comma separated list)	
Also Notify	
Update ACL	

... and in this DNS zone you create a PTR record for the *`Name 4`* (which is our IP address *`1.2.3.4`*) which points to *`www.example.com`*.

The form contains the following fields:

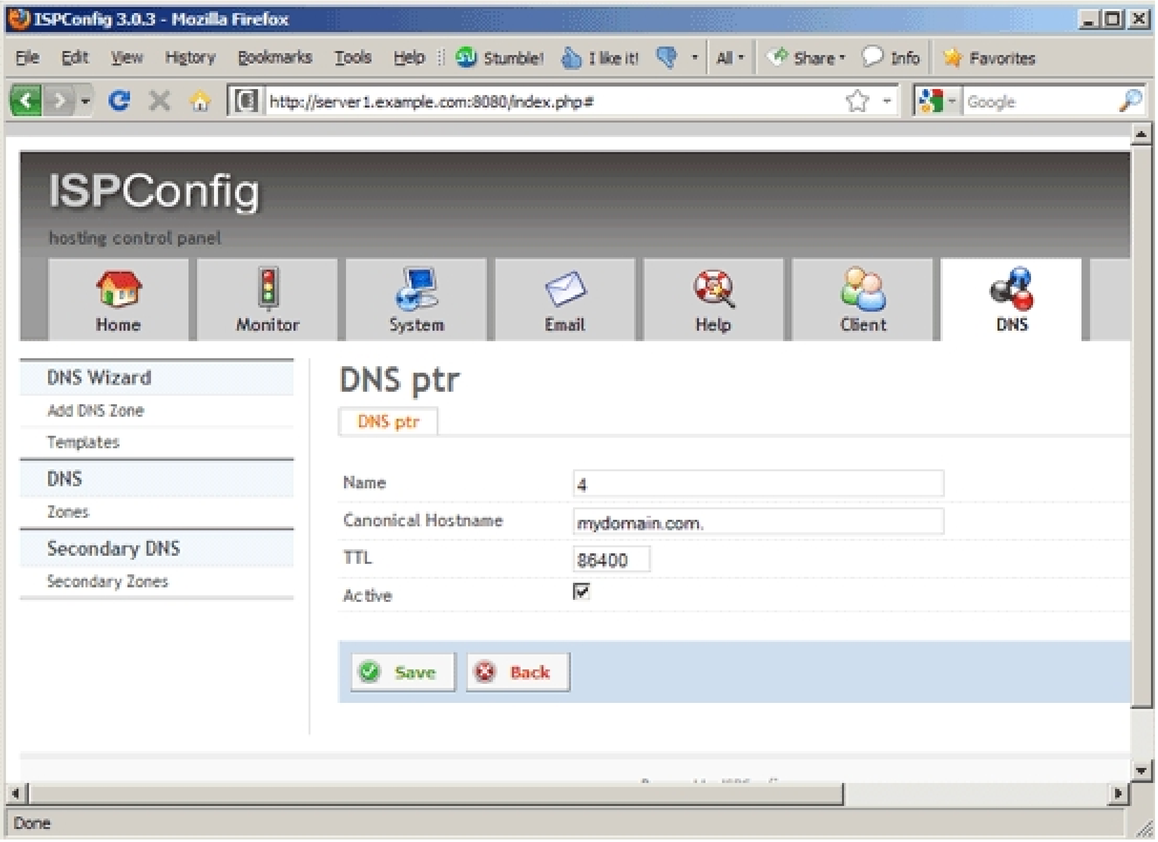
- *`Name`*: Fill in the last part of your IP address. In our example of the *`1.2.3.4`* IP address, this would be *`4`* (without any dots).
- *`Canonical Hostname`*: Fill in the domain or hostname that this PTR record should point to. You must use fully qualified domain names here:

Examples:

- *`example.com.`* (FQDN)

- `www.example.com.` (FQDN)

- `TTL`: The time interval (in seconds) that this record may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the record can only be used for the transaction in progress, and should not be cached.
- `Active`: This defines whether this NS record is active or not.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
ISPConfig
hosting control panel
Home
Monitor
System
Email
Help
Client
DNS
DNS Wizard
Add DNS Zone
Templates
DNS
Zones
Secondary DNS
Secondary Zones
Save
Back
Done

DNS ptr

DNS ptr

Name	4
Canonical Hostname	mydomain.com.
TTL	86400
Active	☑

RP Records

An RP record describes a responsible person for a hostname. The `Responsible Person` field contains the DNS-encoded email address of the person responsible for the `Hostname` requested, then a space, then a hostname that should return a TXT record containing additional information about the responsible person. If there is no such TXT record, the second value should contain a dot (`.`).

Example: *`webmaster.example.com. contactinfo.example.com.`*

The form contains the following fields:

- `Hostname`: The name that this record describes. Wildcard values such as `*` or `*.sub` are supported, and this field can contain an FQDN or just a hostname. If you specify an FQDN, the name must end with a dot; if you specify just a hostname, it must not end with a dot. If you want email addresses of the form `user@example.com`, you must fill in `example.com.` in the `Hostname` field (or leave it empty); if you want email addresses of the form `user@sub.example.com`, you must fill in `sub` or `sub.example.com.` in the `Hostname` field.

Examples:

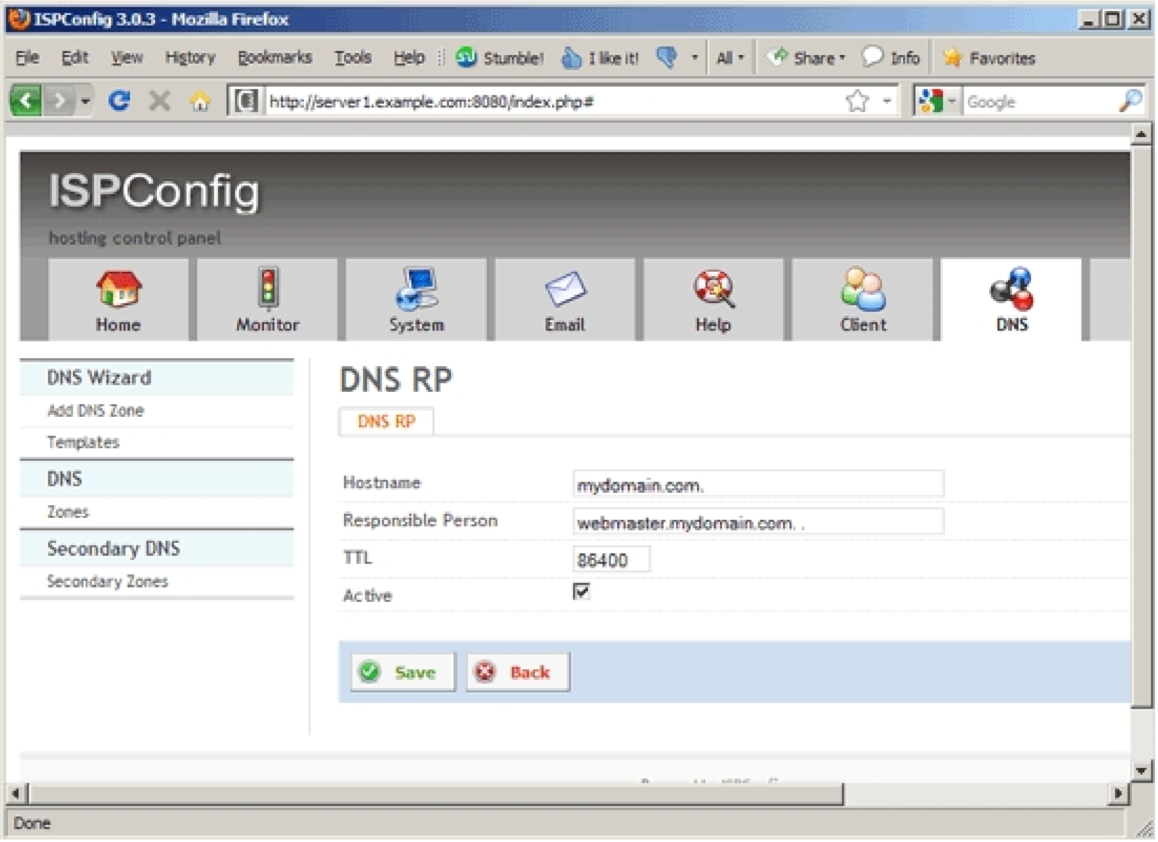
  - foo
  - foo.example.com.
  - www
  - example.com.
  - You can also leave the field empty which has the same meaning as if you'd fill in `example.com.`

- `Responsible Person`: The `Responsible Person` field contains the DNS-encoded email address of the person responsible for the `Hostname` requested, then a space, then a hostname that should return a TXT record containing additional information about the responsible person. If there is no such TXT record, the second value should contain a dot (`.`).

Examples:

  - `webmaster.example.com. contactinfo.example.com.` (This means the responsible person is `webmaster@example.com`, and there is a TXT record for the hostname `contactinfo.example.com` which contains additional information about `webmaster@example.com`. If no TXT record for `contactinfo.example.com` exists, create one.)
  - `webmaster.example.com. .` (If no such TXT record exists or you don't want to create one, just fill in a dot for the hostname.)

- `TTL`: The time interval (in seconds) that this record may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the record can only be used for the transaction in progress, and should not be cached.
- `Active`: This defines whether this RP record is active or not.



DNS RP

Hostname	mydomain.com.
Responsible Person	webmaster.mydomain.com. .
TTL	86400
Active	☑

SRV Records

Server location. SRV records specify the location of the server(s) for a specific protocol and domain. The `Server Record` field must contain three space-separated values. The first value is a number specifying the weight for this entry. The second field is a number specifying the port on the target host of this service. The last field is a name specifying the target host. The `Priority` field should contain the priority of this target host. Targets with a lower priority are preferred.

Some protocols such as SIP and XMPP require SRV records. SRV records have the form

```
_service._proto.name TTL class SRV priority weight port target
```

- *`service`*: The symbolic name of the desired service.
- *`proto`*: The transport protocol of the desired service; this is usually either TCP or UDP.
- *`name`*: The domain name for which this record is valid.
- *`TTL`*: The time interval (in seconds) that this record may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the record can only be used for the transaction in progress, and should not be cached.
- *`class`*: Standard DNS class field (this is always *`IN`*).
- *`priority`*: The priority of the target host, lower value means more preferred (similar to MX records).

- *weight*: A relative weight for records with the same priority.
- *port*: The TCP or UDP port on which the service is to be found.
- *target*: The canonical hostname of the machine providing the service.

E.g.

```
 _sip._tcp.example.com. 86400 IN SRV 0 5 5060 sipserver.example.com.
```

SRV records allow you to achieve a basic form of high-availability and load-balancing (basic because information is static, i.e., current server loads are not taken into account). The priority field is similar the the one of MX record - clients use the server with the lowest priority value first and use other servers only if this server fails. This means oyu can have multiple SRV records and define a fallback server that is used only if the primary server fails by giving the fallback server a higher priority value than the primary server.

If there are multiple SRV records with the same priority, clients use the weight field to find out which host to use. The weight value is relevant only in among records with the same priority.

Here's an example of basic high-availability and load-balancing with SRV records:

```
_sip._tcp.example.com. 86400 IN SRV 10 60 5060 server1.example.com.
_sip._tcp.example.com. 86400 IN SRV 10 40 5060 server2.example.com.
_sip._tcp.example.com. 86400 IN SRV 20 0 5060 server3.example.com.
```

In the above example, both *server1.example.com* and *server2.example.com* have a priority value of *10*, so all requests will be shared by them, where *server1.example.com* gets 60% of the requests and *server2.example.com* gets the remaining 40% of the requests (because *server1.example.com* has a weight value of *60* and *server2.example.com* has a weight value of *40*). If *server1.example.com* fails, all requests will go to *server2.example.com*. If both *server1.example.com* and *server2.example.com* fail, all requests will go to *server3.example.com* which has a priority value of *20*.

For more information, read ***RFC 2782*** and ***SRV Records on Wikipedia***.

The form has the following fields:

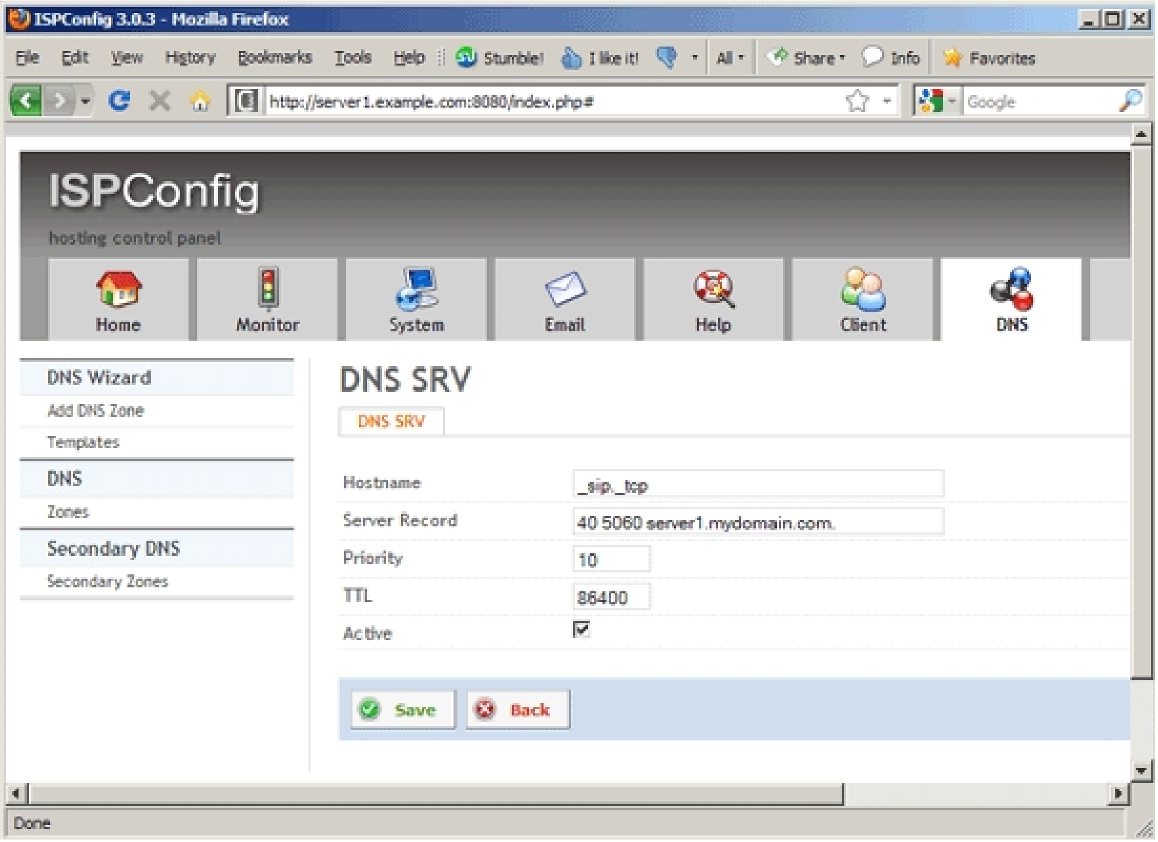
- *Hostname*: The name that this record describes. This field can contain an FQDN or just a hostname. If you specify an FQDN, the name must end with a dot; if you specify just a hostname, it must not end with a dot.

Examples:

- _sip._tcp.example.com.
- _sip._tcp

- Target: Specify the target host in this field. The target host can be an FQDN or just a hostname. If you specify an FQDN, the name must end with a dot; if you specify just a hostname, it must not end with a dot.

- Weight: Specify the weight for this entry.
- Port: Specify the port on the target host of this service.
- `Priority`: The `Priority` field should contain a preference for this SRV record, usually between `0` and `100`. Records with lower values are preferred.
- `TTL`: The time interval (in seconds) that this record may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the record can only be used for the transaction in progress, and should not be cached.
- `Active`: This defines whether this SRV record is active or not.



Hostname	_sip._tcp
Server Record	40 5060 server1.mydomain.com.
Priority	10
TTL	86400
Active	☑

TXT Records

TXT records are used to give additional information about a hostname. The `Text` field contains a text string that is returned only when a TXT query is issued for the host specified by `Hostname`. TXT records can be used for ***SPF records***.

The form contains the following fields:

- *`Hostname`*: The name that this record describes. This field can contain an FQDN or just a hostname. If you specify an FQDN, the name must end with a dot; if you specify just a hostname, it must not end with a dot.

Examples:

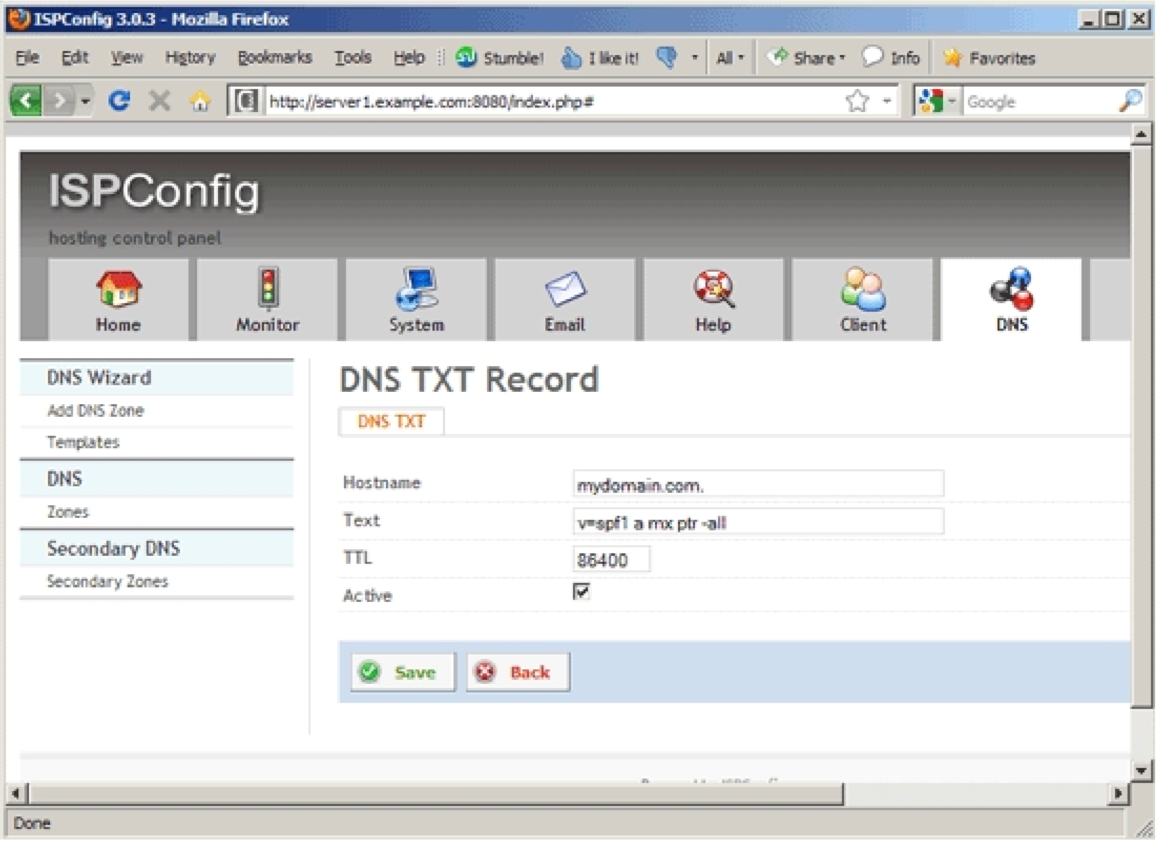
- server1.example.com.
- server1

- *`Text`*: The *`Text`* field contains a text string that is returned only when a TXT query is issued for the host specified by *`Hostname`*. TXT records can be used for ***SPF records***. It must not end with a dot.

Examples:

- This is a string.
- *`v=spf1 a mx ptr -all`* (SPF record)

- *`TTL`*: The time interval (in seconds) that this record may be cached before the source of the information should again be consulted. Zero values are interpreted to mean that the record can only be used for the transaction in progress, and should not be cached.
- *`Active`*: This defines whether this TXT record is active or not.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
ISPConfig
hosting control panel
Home
Monitor
System
Email
Help
Client
DNS
DNS Wizard
Add DNS Zone
Templates
DNS
Zones
Secondary DNS
Secondary Zones
DNS TXT
Save
Back
Done

DNS TXT Record

Hostname	mydomain.com.
Text	v=spf1 a mx ptr -all
TTL	86400
Active	☑

4.8.3 Secondary DNS

4.8.3.1 Secondary Zones

(This feature is supported only if you use the BIND name server. If you use MyDNS, database replication will be used to transfer data to the secondary DNS server.)

Here you can create secondary (slave) zones, i.e., zones for which another server is the primary (master) nameserver. A slave zone will then automatically be transferred from the master to the slave, so that both servers hold the same information about the zone. If the master fails, the slave can still answer DNS requests.

To create a new slave zone, click on the `Add new secondary DNS Zone` button. This will lead you to the `Secondary DNS Zone` form with the tab `Secondary DNS Zone`.

Secondary DNS Zone

Secondary DNS Zone

The form has the following fields:

- *`Server`*: If more than one server is available, you can select the server on which the secondary DNS zone will be located.

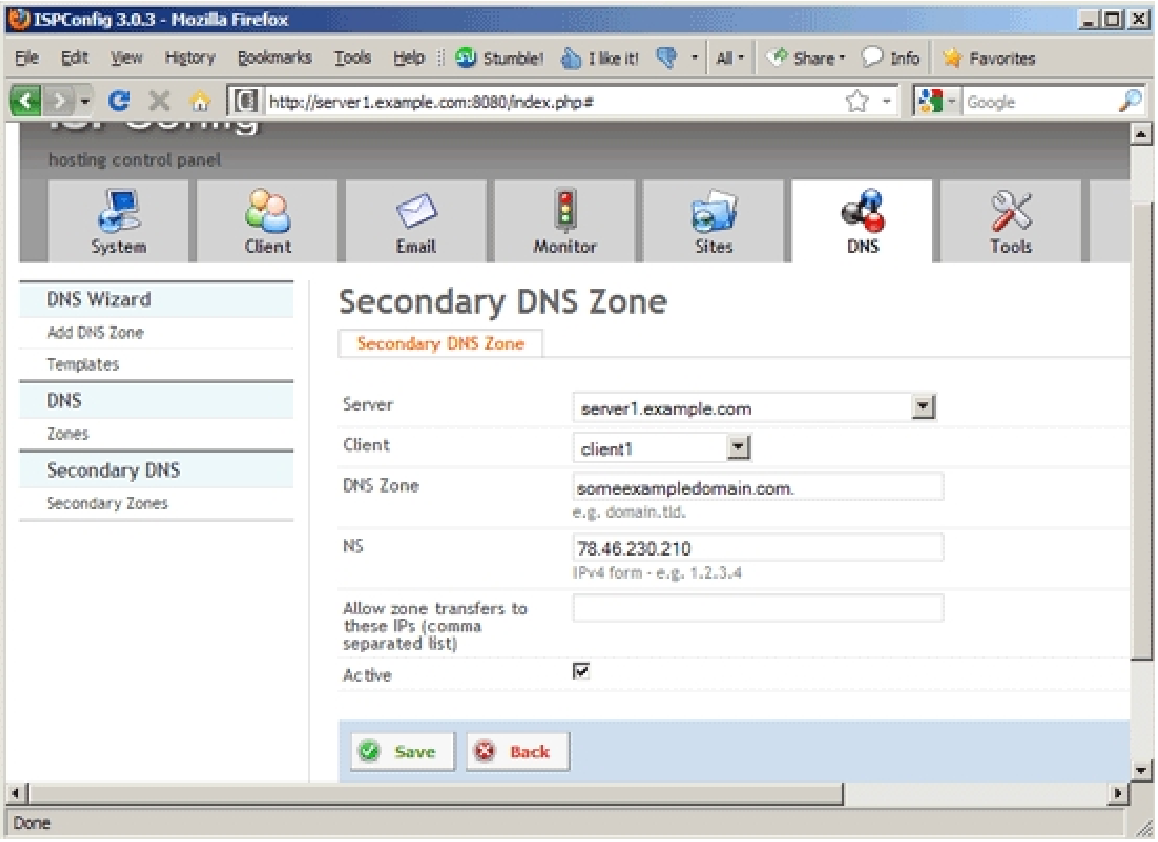
- *`Client`*: Here you select the client that owns the new secondary DNS zone.

- *`DNS Zone`*: Fill in the domain for which you want to create the secondary zone, e.g. *`example.com.`* - please note that you need a dot at the end.

- *`NS`*: Specify the IPv4 address of the primary nameserver for the domain, e.g. *`1.2.3.4`*.

- *`Allow zone transfers to these IPs (comma separated list)`*: This field can contain one or more IP addresses separated by commas. These IP addresses will be allowed to connect to the server to transfer the zone. If no IP is specified, any server is allowed to connect. Usually, you can leave this field empty because all slave DNS servers for this zone should contact the master DNS server for the zone, not another slave server.

- *`Active`*: This defines whether this secondary DNS zone is active or not.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
hosting control panel
System
Client
Email
Monitor
Sites
DNS
Tools
DNS Wizard
Add DNS Zone
Templates
DNS
Zones
Secondary DNS
Secondary Zones
Save
Back
Done

Secondary DNS Zone

Secondary DNS Zone

Server	server1.example.com
Client	client1
DNS Zone	someexampledomain.com. e.g. domain.tld.
NS	78.46.230.210 IPv4 form - e.g. 1.2.3.4
Allow zone transfers to these IPs (comma separated list)	
Active	☑

4.9 System

This is where you define the basic settings of the ISPConfig control panel (creating users, configuring services, IP addresses, firewall records, updating the system, etc.).

4.9.1 User Management

Here you can create and modify users of the ISPConfig control panel. Please note that you should use these functions only to create or modify admin users. To create/edit normal ISPConfig users, use the client- and reseller settings in the Client module instead because modifying users or groups here may cause data loss. If you change modules or groups of existing users, these users might not be able to access their web site settings, email settings, etc. in ISPConfig anymore.

4.9.1.1 CP Users

Here you can create new ISPConfig users. The *Users* form has the tabs *Users* and *Groups*.

Users

Users

The form has the following fields:

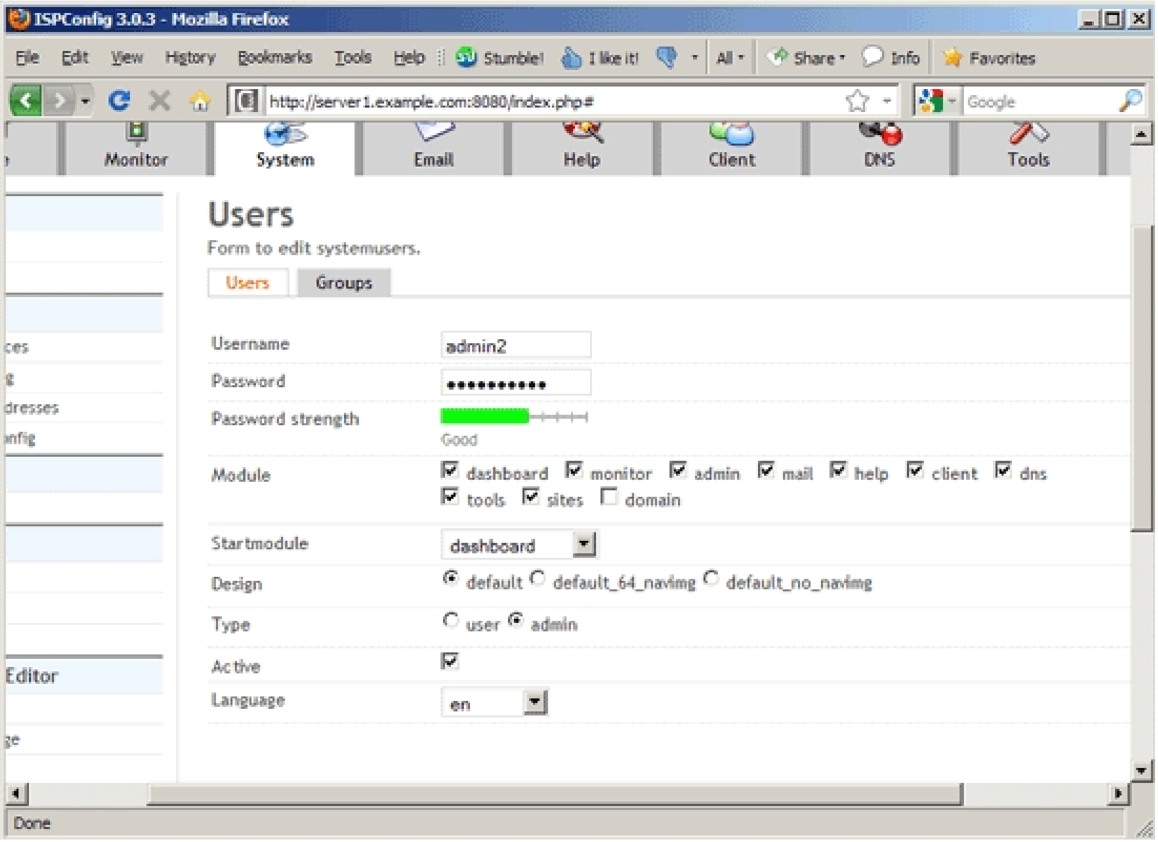
- *Username*: Fill in the username of the new user.
- *Password*: Type in a password for the ISPConfig user (or use the *Generate Password* link to have ISPConfig generate one for you). The *Password strength* field will show how weak or strong your password is. A strong password should include numbers, symbols, upper and lowercase letters; password length should be 8 characters or more; avoid any password based on repetition, dictionary words, letter or number sequences, usernames, relative or pet names, or biographical information.
- Repeat Password: Confirm your password.
- *Module*: Select the modules that will be available for the user:
  - *sites*: This enables the *Sites* module.
  - *mail*: This activates the *Email* module.
  - *monitor*: This refers to the *Monitor* module.
  - *admin*: This is the *System* module (i.e., the module that we are currently in).
  - *dashboard*: This refers to *Home*.
  - *dns*: This is the *DNS* module.
  - *help*: This activates the *Help* module.
  - *domain*: This enables the *Domain* module. This makes sense only if you also check the *Use the domain-module to add new domains* checkbox on the *Domains* tab under *System > System > Interface Config*. If you use this module, your customers can only select one of

the domains the admin creates for them. They can not freely edit the domain field.

- *`client`*: This enables the *`Client`* module.
- *`tools`*: This is the *`Tools`* module.
- *`vm`*: This enables the *`VServer`* module.

You can select multiple modules for each user.

- *`Startmodule`*: Select the module that will automatically be loaded when the user logs into ISPConfig.
- *`Design`*: Select the theme of the ISPConfig interface.
- *`Type`*: Please select if this is a normal *`user`* account or an *`admin`* account.
- *`Active`*: This defines whether this ISPConfig user account is active or not.
- *`Language`*: Select the language in which ISPConfig will be loaded for the user.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
Monitor
System
Email
Help
Client
DNS
Tools
Editor
Done

Users

Form to edit systemusers.

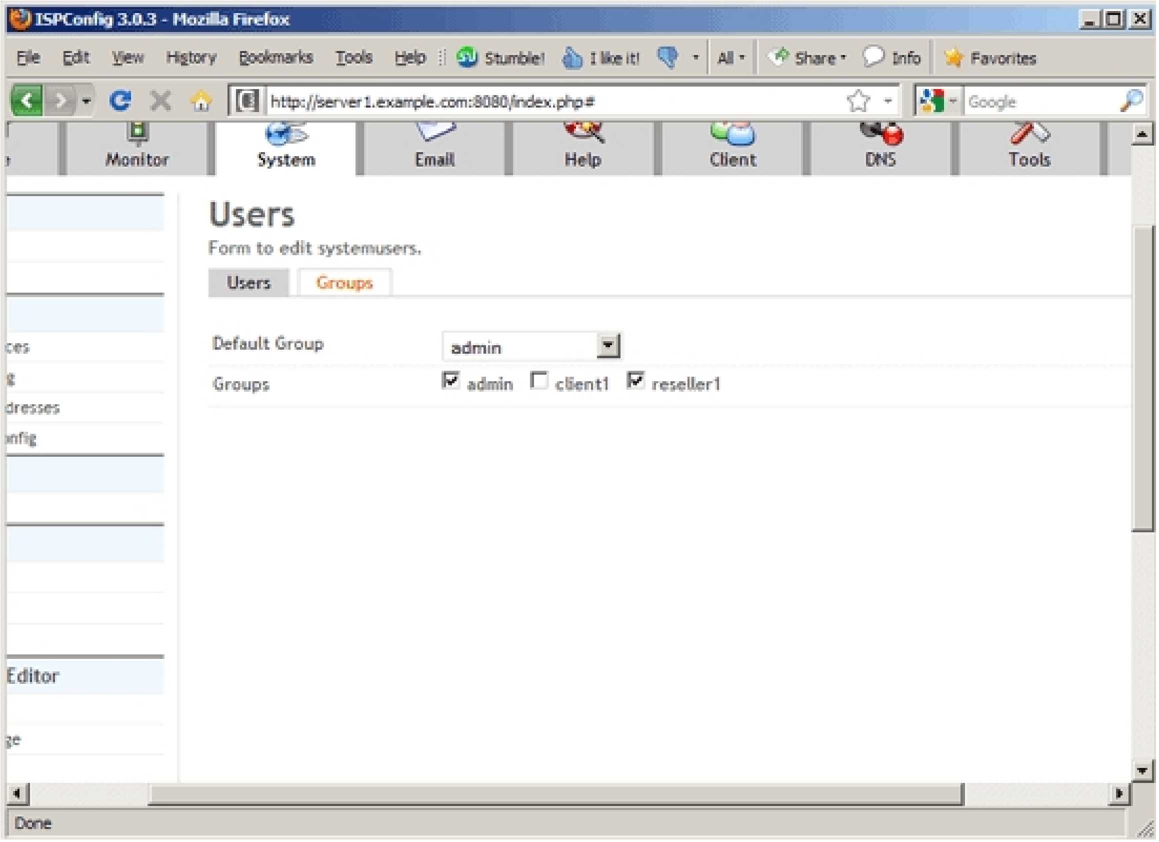
Users | Groups

Username	admin2
Password	••••••••••
Password strength	Good
Module	☑ dashboard ☑ monitor ☑ admin ☑ mail ☑ help ☑ client ☑ dns ☑ tools ☑ sites □ domain
Startmodule	dashboard
Design	● default ○ default_64_navimg ○ default_no_navimg
Type	○ user ● admin
Active	☑
Language	en

Groups

The form has the following fields:

- `Default Group`: This defines the group to which items created by the user (web sites, email accounts, etc.) will belong (unless a different group is selected when the item is created). Selecting a default group does not necessarily mean that the user is also a member of the group - you must check that group in the following form item, `Groups`, to make the user also a member of the default group.
- `Groups`: Check all groups that the user account should be a member of. Make sure that you also check the group that you selected under `Default Group` to make the user a member of that group.



Users

Form to edit systemusers.

Users | Groups

Default Group: admin

Groups: ☑ admin ☐ client1 ☑ reseller1

4.9.1.2 Remote Users

This feature is for ISPConfig developers only. ISPConfig has an API that allows to access all ISPConfig functions from other applications or remote places (the API documentation is not part of this manual). For example, an ISP could build a web interface and allow his customers to create web sites from this web interface.

Access to this API is password protected. To allow access to the API, you must create a user and password here first and use these login credentials in the application that uses the API.

Remote user

Remote User

The form has the following fields:

- `Username`: Fill in the username of the new API user.
- `Password`: Type in a password for the API user (or use the `Generate Password` link to have ISPConfig generate one for you). The `Password Strength` field will show how weak or strong your password is.
- Repeat Password: Confirm your password.
- `Functions`: Please check all functions that the API user will be allowed to use.

4.9.2 System

4.9.2.1 Server Services

All servers that are listed here are added by the ISPConfig installer, i.e., you cannot add new servers here yourself. ISPConfig allows you to control multiple servers from just one control panel, and all servers that are listed here are controlled by ISPConfig. If you want to add another server to ISPConfig, you have to run the ISPConfig installer in ***expert*** mode on the remote server and tell the installer that the server will be a slave.

Although you cannot add servers here yourself, you can modify them from here by selecting a server. This will bring you to the `Server` form with the tab `Services`.

Server

Services

The form has the following fields:

- `Servername`: Specify the hostname of the server. Example: `server1.example.com`
- `Mailserver`: This specifies if this server acts as a mail server (i.e., you can use the `Email` module to create email accounts etc. on this server).
- `Webserver`: This specifies if this server acts as a web server (i.e., you can use the `Sites` module to create web sites etc. on this server).
- `DNS-Server`: This specifies if this server acts as a DNS server (i.e., you can use the `DNS` module to create DNS zones etc. on this server).
- `Fileserver`: If this server acts as a web server, you should also enable `Fileserver` for this server so

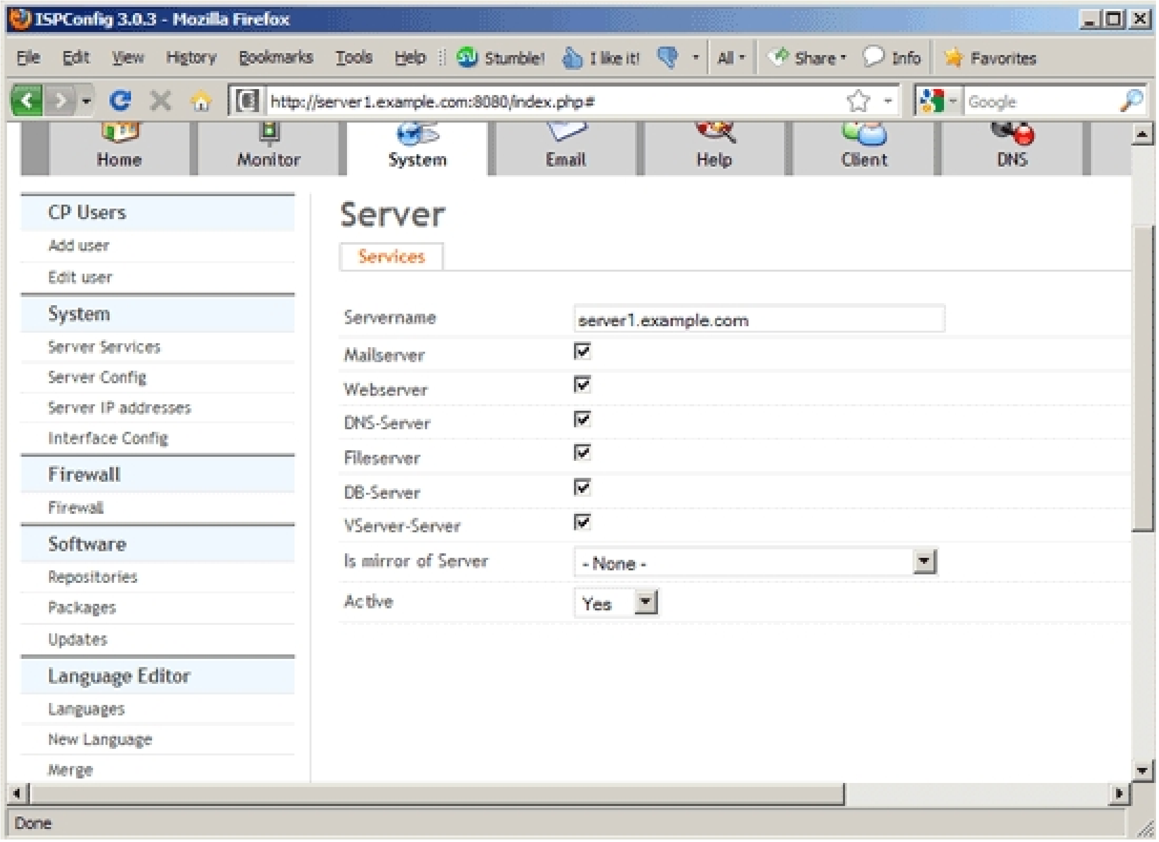
that FTP access is possible.

- *`DB-Server`*: This specifies if it will be able to create databases (in the `Sites` module) on this server.

- *`VServer-Server`*: If you check this, it will be possible to create OpenVZ virtual machines on this server (this will be possible from version 3.0.4 of ISPConfig).

- *`Is mirror of Server`*: If you have specified that this server is a slave of another server during the ISPConfig installation, this server can have two roles: it can act as a full-fledged server, i.e., you can create web sites, email accounts, etc. on this server just like on the main server, or it can act as a mirror of another server - in this case you cannot create any items on that server (this server cannot be selected when you create a new item), but instead the configuration (web site configuration, email configuration, etc.) will be copied to the mirror (just the configuration, not any web site contents, etc. - if you want this, you can achieve this by using ***rsync*** or using a cluster filesystem like ***GlusterFS*** or some kind of network-attached storage, and you'd have to use one of these techniques on the directories */var/www* for the web sites' contents and */var/vmail* for the emails - for MySQL databases, you'd have to use ***MySQL master-master replication***). If you select a master server in the *`Is mirror of Server`* field, the server for which you select the master will act as a mirror, not as a full-fledged server. If you have a failover-IP address that you can switch between the master and the mirror (e.g. automatically with ***heartbeat***/***keepalived***/etc. or manually, e.g. from your hoster's control panel), you can achieve high-availability because if the master fails, the mirror can take over.

- *`Active`*: This defines whether this server is active or not.



Server

Services

Servername	server1.example.com
Mailserver	☑
Webserver	☑
DNS-Server	☑
Fileserver	☑
DB-Server	☑
VServer-Server	☑
Is mirror of Server	- None -
Active	Yes

4.9.2.2 Server Config

All servers that are listed here are added by the ISPConfig installer, i.e., you cannot add new servers here yourself. ISPConfig allows you to control multiple servers from just one control panel, and all servers that are listed here are controlled by ISPConfig. If you want to add another server to ISPConfig, you have to run the ISPConfig installer in ***expert*** mode on the remote server and tell the installer that the server will be a slave.

Although you cannot add servers here yourself, you can modify them from here by selecting a server. This will bring you to the *Server Config* form with the tabs *Server*, *Mail*, *Getmail*, *Web*, *DNS*, *FastCGI*, *Jailkit*, *vlogger*, and *Cron*.

Please note that you shouldn't modify these settings unless you know exactly what you're doing - changes in paths etc. might stop the system from working!

Server Config

Server

On this tab you can configure some basic network settings for the server plus the loglevel for the ISPConfig log (under *Monitor > System State (All Servers) > Show System-Log*) plus the backup directory for web site backups.

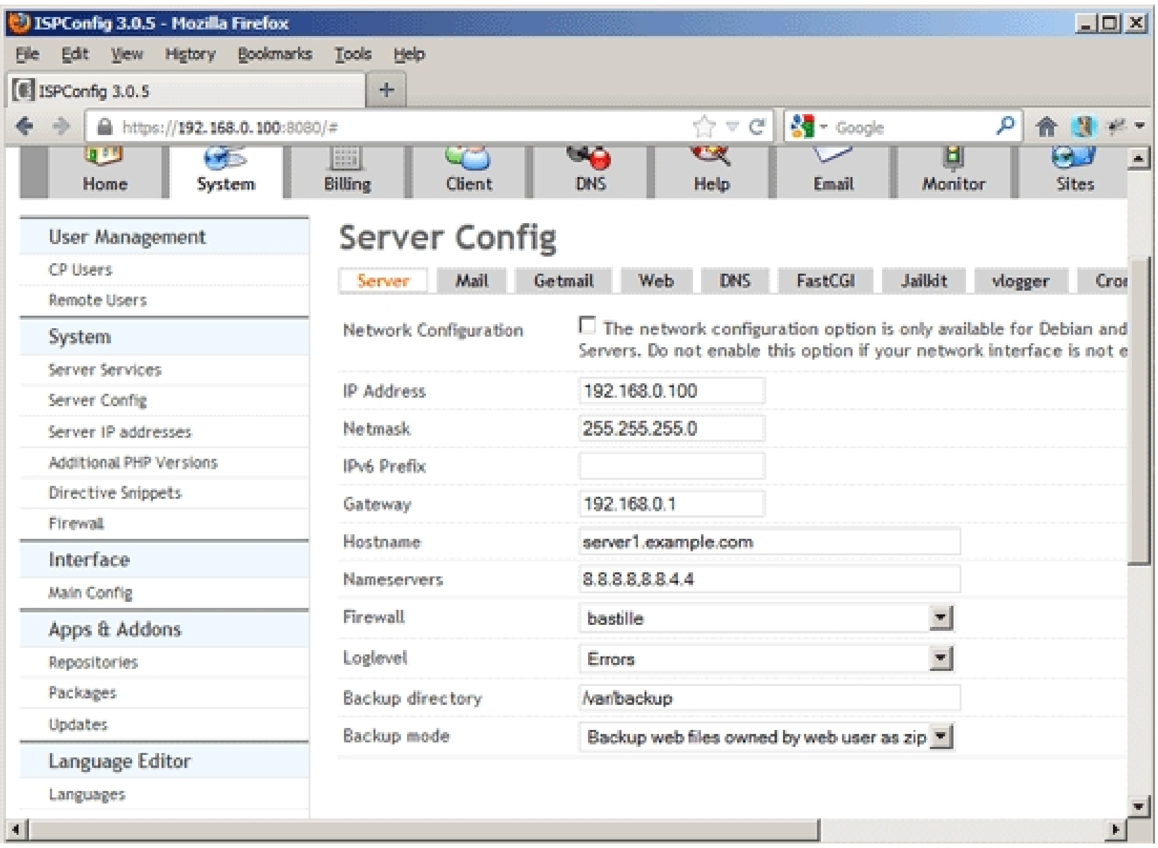
The form has the following fields:

- *Network Configuration*: If you check this, ISPConfig will automatically configure your system with the network settings from the *IP Address*, *Netmask*, *Gateway*, *Hostname*, and *Nameservers* fields. It will also automatically configure all IP addresses that are defined under *System > System > Server IP addresses*. Please note that this automatic network configuration works only on Debian/Ubuntu and only if you have one network card which must be eth0. It is recommended to not check this checkbox and configure your network settings manually.
- *IP Address*: Specify the IPv4 address of this server. Example: *1.2.3.4*
- *Netmask*: Type in the server's netmask. Example: *255.255.255.0*
- IPv6 Prefix: If you have an IPv6 subnet, you can specify the IPv6 prefix here. ISPConfig uses this field for mirror servers where it generates the IPv6 address from this IPv6 prefix and the web ID.
- *Gateway*: Fill in the server's gateway.
- *Hostname*: Type in the server's fully-qualified hostname. Example: *server1.example.com*
- *Nameservers*: Fill in the IP addresses of nameservers that this server will use to do DNS lookups. You can specify multiple nameservers by separating them with a comma. These should be the nameservers from */etc/resolv.conf*. Example: *145.253.2.75,8.8.8.8*
- Firewall: Select which firewall you want to use if you want ISPConfig to set up the firewall (see chapter ***4.9.2.6 Firewall***). You can choose between *bastille* (comes with ISPConfig) and *ufw* (must be installed

manually).

- `Loglevel`: Select the loglevel for the ISPConfig log (under `Monitor > System State (All Servers) > Show System-Log`).
    - `Debug`: This loglevel will log all output from ISPConfig, including warnings and errors. As the name says, this is usefull for debugging.
    - `Warnings`: This loglevel will log ISPConfig warnings and errors.
    - `Errors`: This loglevel will just log ISPConfig errors. Recommended for production systems.

- `Backup directory`: This is the directory where web site backups will be stored. The default directory is `/var/backup`.
- `Backup mode`: There are two backup modes. The recommended and secure option is `Backup web files owned by web user as zip` which creates a zip file that contains only files and directories owned by the web site user, while `Backup all files in web directory as root user` creates a tar.gz file with all files of a web site, no matter who owns them. This is potentially dangerous and therefore not recommended because a user could hard-link or symlink system files such as `/etc/passwd` and `/etc/shadow` which would then be part of the backup. MySQL database dumps are automatically included in backups, regardless of the option you select here.



Server Config

Network Configuration	□ The network configuration option is only available for Debian and Servers. Do not enable this option if your network interface is not e
IP Address	192.168.0.100
Netmask	255.255.255.0
IPv6 Prefix	
Gateway	192.168.0.1
Hostname	server1.example.com
Nameservers	8.8.8.8,8.8.4.4
Firewall	bastille
Loglevel	Errors
Backup directory	/var/backup
Backup mode	Backup web files owned by web user as zip

Mail

On this tab you can configure the general mail settings for the server.

The form has the following fields:

- *Module*: Select the mail module that the server should use. Currently only *postfix_mysql* is supported.

- *Maildir Path*: This defines where users' mailboxes will be located. The default path is */var/vmail/[domain]/[localpart]*. *[domain]* is a placeholder for the mail domain and *[localpart]* is a placeholder for the local part of an email address. Example: if your email address is *user@example.com*, the Maildir path would be */var/vmail/example.com/user*. Please note that *Maildir Path* should be a subdirectory of *Homedir Path* - otherwise the mail system will probably stop to work.

- *Homedir Path*: This is the home directory of *Mailuser Name*. The default directory is */var/vmail*. If you use maildrop, this is the directory where the mailfilter file will be located.

- *POP3/IMAP Daemon*: Select your POP3/IMAP daemon. Supported POP3/IMAP daemons are Courier and Dovecot.

- *Mailfilter Syntax*: Select the mailfilter to use. If you use Courier, you must select *Maildrop*; if you use Dovecot, you must select *Sieve*. Depending on what you select, you must use Maildrop or Sieve syntax if you define custom filter rules for an email mailbox (*Email > Email Mailbox > Custom Rules*). If you create mailfilters under *Email > Email Mailbox > Mail Filter*, the system will automatically translate them into Maildrop or Sieve syntax depending on your selection here.

- *Mailuser UID*: This is the user ID of the system user defined under *Mailuser Name*.

- *Mailuser GID*: This is the group ID of the system group defined under *Mailuser Group*.

- *Mailuser Name*: This is the system user name of the user under which the virtual mail setup runs. Default value: *vmail*

- *Mailuser Group*: This is the system group name of the group under which the virtual mail setup runs. Default value: *vmail*

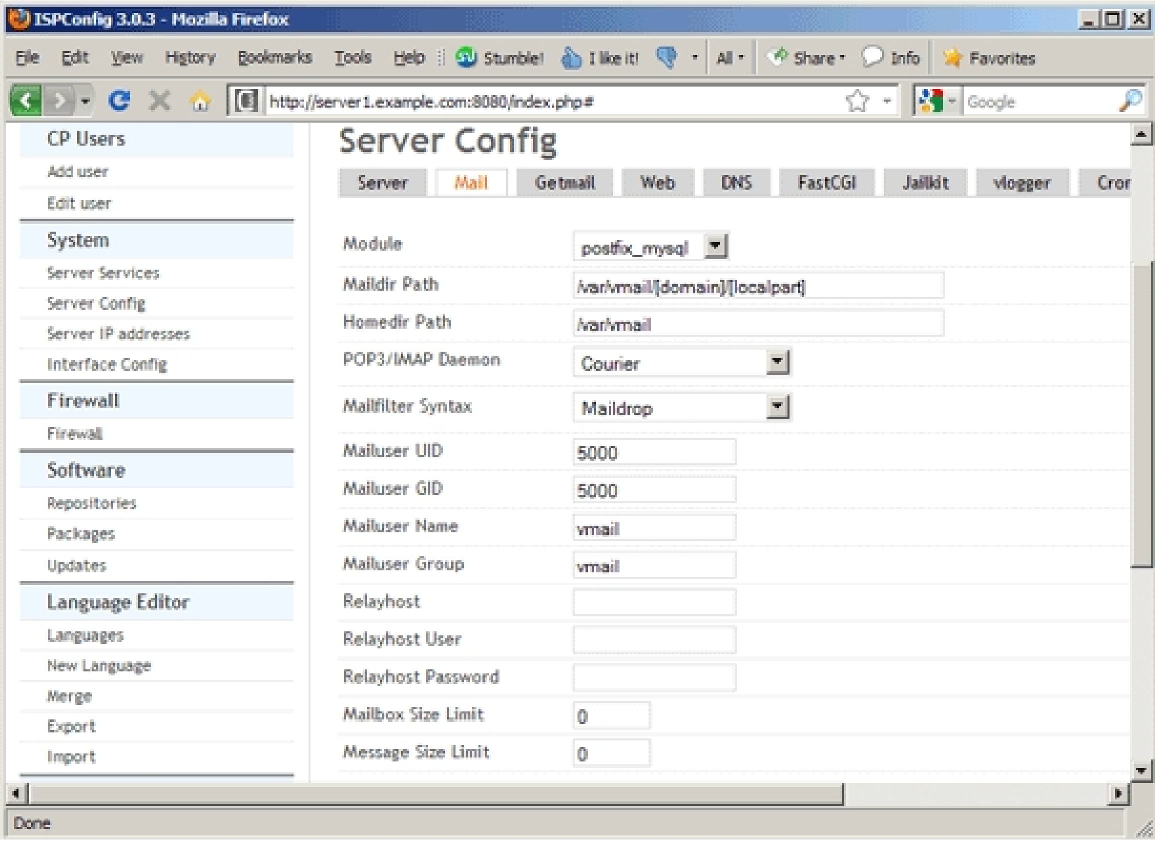
- *Relayhost*: If you want to ***relay*** outgoing mails through another mailserver (for example, because your server is on a dynamic IP and therefore blacklisted), you can use the *Relayhost*, *Relayhost User*, and *Relayhost Password* fields for this. Fill in the hostname or IP address of the server through which you want to relay in the *Relayhost* field. If you use an IP address, put it in square brackets (*[]*) to prevent DNS lookups. Examples: *mail.yourisp.com*, *[1.2.3.4]*. Leave the field empty if you don't want to relay.

- *Relayhost User*: Fill in the username that can be used to log in on the relayhost .

- *Relayhost Password*: Fill in the password of the relayhost user on the relayhost.

- *Mailbox Size Limit*: This defines the max. size (in MB) that a single mailbox can have on this server. *0* means unlimited.

- *`Message Size Limit`*: This defines the max. size (in MB) that a single email can have on this server. *`0`* means unlimited.
- *`Mailbox quota statistics`*: If you check this field, ISPConfig will create mailbox quota statistics for each mail box.
- *`Real-time Blackhole List`*: This field allows you to specify blacklists for Postfix, so that spam can be blocked at MTA level. To specify multiple blacklists, separate them by comma. Example: *`zen.spamhaus.org,cbl.abuseat.org,ix.dnsbl.manitu.net`*



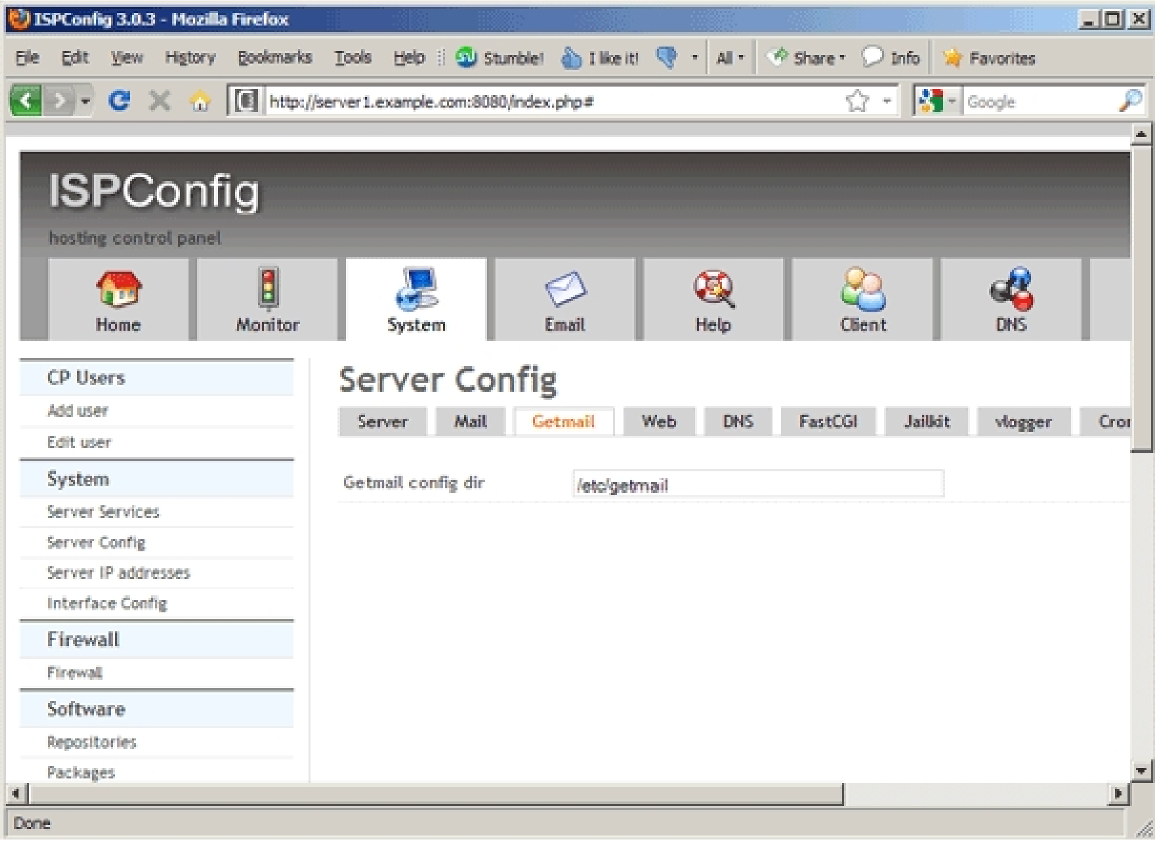
Module	postfix_mysql
Maildir Path	/var/vmail/[domain]/[localpart]
Homedir Path	/var/vmail
POP3/IMAP Daemon	Courier
Mailfilter Syntax	Maildrop
Mailuser UID	5000
Mailuser GID	5000
Mailuser Name	vmail
Mailuser Group	vmail
Relayhost	
Relayhost User	
Relayhost Password	
Mailbox Size Limit	0
Message Size Limit	0

Getmail

Here you can configure getmail. Getmail is the service that fetches emails from remote servers; it is used if you define accounts under *`Email > Fetchmail > Fetchmail`*.

The form has the following field:

- *`Getmail config dir`*: This is the directory where getmail expects its configuration.



Getmail config dir: /etc/getmail

Web

On this tab you can configure various settings for Apache, nginx, PHP, AWStats, etc. Some fields are relevant to Apache only, others to nginx and are only shown if the appropriate http server is installed.

The form has the following fields:

- *`Server Type`*: This is a readonly field and shows if the server uses Apache2 or nginx.
- *`Website basedir`*: This is the directory where all web sites will be created (in subdirectories). Usually this is the value of *AP_DOC_ROOT* from the output of *`suexec -V`* or *`suexec2 -V`* so that suEXEC can be used in the web sites (*`/var/www`* on Debian/Ubuntu/Fedora/CentOS, *`/srv/www`* on OpenSUSE). (The suEXEC feature provides Apache users the ability to run CGI and SSI programs under user IDs different from the user ID of the calling web-server.)
- *`Website path`*: This is the actual path where new web sites will be created (this is not the actual document root of the web site - this will be the subdirectory *web* in *`Website path`*). This should be a subdirectory of *`Website basedir`*. You can use the placeholders *`[client_id]`* and *`[website_id]`* which will be replaced by the IDs of the client and web site respectively.
- *`Website symlinks`*: ISPConfig can create symlinks to *`Website path`* so that it is easier to navigate to *`Website Path`* on the command line. You can use the placeholder *`[website_domain]`* which will be replaced by the domain of the web site (e.g. *`example.com`*). You can define multiple symlinks by

separating them with a colon (`:`) (don't use spaces).

- `Make relative symlinks`: By default, ISPConfig uses full paths for the symlinks it creates, like `example.com -> /var/www/clients/client0/web1/` in the `/var/www` directory or `example.com -> /var/www/clients/client0/web1/` in the `/var/www/clients/client0` directory. If you check this box, ISPConfig creates relative symlinks like `example.com -> clients/client0/web1/` in the `/var/www` directory.

- `Website auto alias`: In this field you can specify a hostname that clients can use to visit a web site before the real web site domain is connected with this web site (for example, because the real domain has not been registered yet, or its DNS records are still pointing to another server). You can use the placeholders `[client_id]`, `[client_username]`, `[website_id]`, and `[website_domain]`. Example: `client[client_username].example.com`. The client with the client ID 34 would then be able to access his web site under the address `client34.example.com`.

- `Vhost config dir` ***(Apache only)***: This is the directory where ISPConfig will place the vhost configuration files for each web site. This does not automatically enable the vhost because Apache doesn't read that directory. To enable a vhost, it must be symlinked to another directory which is read by Apache (see `Vhost config enabled dir`).

- `Vhost config enabled dir` ***(Apache only)***: This is a directory that is read by Apache and to which vhost configuration files must be symlinked to enable the vhost.

- `Nginx Vhost config dir` ***(nginx only)***: This is the directory where ISPConfig will place the vhost configuration files for each web site. This does not automatically enable the vhost because nginx doesn't read that directory. To enable a vhost, it must be symlinked to another directory which is read by nginx (see `Nginx Vhost config enabled dir`).

- `Nginx Vhost config enabled dir` ***(nginx only)***: This is a directory that is read by nginx and to which vhost configuration files must be symlinked to enable the vhost.

- `Security level`: This defines how permissions and ownerships are set for the `Website path` directory.

  - `Medium`: The directory is owned by root and readable for all users.

  - `High`: The directory is owned by the web site user and cannot be read by other users. It is recommended to choose `High`.

- `Test apache configuration on restart` ***(Apache only)***: If checked, ISPConfig will test if the Apache configuration is syntactically ok after it has written changes to vhosts. If errors are encountered, the new configuration is abandoned (the vhost file is renamed to `example.com.vhost.`**`err`** in the `sites-available` directory), and the last working configuration is used instead to make sure that Apache restarts successfully.

- `Apache user` ***(Apache only)***: This is the user under which the Apache web server runs.

- `Apache group` ***(Apache only)***: This is the group under which the Apache web server runs.

- `Nginx user` ***(nginx only)***: This is the user under which the nginx web server runs.

- *Nginx group* ***(nginx only)***: This is the group under which the nginx web server runs.
- *Nginx CGI Socket* ***(nginx only)***: This defines the location of the FCGIwrap socket. This socket is needed if you want to serve CGI/Perl scripts through nginx.
- *Enable IP wildcard (*)*: If you check this box, it is possible to use the IPv4 address wildcard * in the IPv4 address field in the web site form.
- *Send overtraffic notification to admin*: If a web site exceeds its traffic limit, an email notification is sent to ISPConfig's admin (in addition to taking the site offline which is independent from your choice here). The admin email address can be specified under *System > Main Config* on the *Mail* tab (see chapter ***4.9.3.1 Main Config***).
- *Send overtraffic notification to client*: If a web site exceeds its traffic limit, an email notification is sent to to client that owns the website (in addition to taking the site offline which is independent from your choice here).
- *Enable SNI*: By default, you need one IP address per SSL website. SNI is short for *Server Name Indication* and allows you to run multiple SSL vhosts on one IP address. Please note that currently SNI is not supported by all browsers/operating systems. Browsers/clients with support for TLS server name indication:
    - Opera 8.0 and later (the TLS 1.1 protocol must be enabled)
    - Internet Explorer 7 or later (under Windows Vista and later only, not under Windows XP)
    - Firefox 2.0 or later
    - Curl 7.18.1 or later (when compiled against an SSL/TLS toolkit with SNI support)
    - Chrome 6.0 or later (on all platforms - releases up to 5.0 only on specific OS versions)
    - Safari 3.0 or later (under OS X 10.5.6 or later and under Windows Vista and later)

  To find out if your browser supports SNI, you can go to ***https://alice.sni.velox.ch/***.
- *CA Path*: If you are your own certificate authority (CA), you can automatically sign certificates created for web sites through ISPConfig. (This is useful, for example, in big companies where all browsers have your CA certificate installed.) To do this, please specify the path to the directory where your *openssl.cnf* file is located (e.g. */usr/local/ssl*).
- *CA passphrase*: Specify your CA's key password here if you are your own CA. This works only in conjunction with the *CA Path field.*
- *Set folder permissions on update*: If this option is checked, ISPConfig sets folder permissions of a web site to their original state (i.e., if you have changed permissions manually, those changes will be lost) when it updates a web site.
- *Make web folders immutable (extended attributes)*: If this option is checked, ISPConfig will set the immutable bit on the web site root (e.g. */var/www/example.com*) which means that no files/folders can be created or deleted in */var/www/example.com* (with the exception of the folder */var/www/example.com/private* that ISPConfig creates so that you can upload files outside of the web site document root */var/www/example.com/web*). That way users canot accidentally delete

important folders, but they are still able to upload files and folders to the web site's document root *`/var/www/example.com/web`*. If you have this option checked and need to make some manual changes, you can delete the immutable bit like this:

```
chattr -i /var/www/example.com/
```

And to enable it again afterwards, you run:

```
chattr +i /var/www/example.com/
```

- *`Add web users to -sshusers- group`*: This option adds a web site user to the *`sshusers`* group.
- *`Connect Linux userid to webid`*: This option is needed only for mirror setups. It creates a Linux UID that is connected to the ID of the web site.
- *`Start ID for userid/webid connect`*: If you have enabled *`Connect Linux userid to webid`* , you must specify a start UID here to which the web id is added. For example, if you specify *`10000`* here, and the web ID is 23, the Linux UID will be 10023. Make sure to use a start ID above which no UIDs are in use (*`10000`* is normally a good value).
- *`Apache php.ini path`* ***(Apache only)***: This is the full *`php.ini`* path for the *`php.ini`* file used by Apache's mod_php.
- *`CGI php.ini path`* ***(Apache only)***: This is the full *`php.ini`* path for the *`php.ini`* used by FastCGI, CGI, and suPHP.
- *`PHP-FPM init script`* ***(nginx only)***: This is the filename of the PHP-FPM init script from the *`/etc/init.d/`* directory.
- *`PHP-FPM php.ini path`* ***(nginx only)***: This is the full *`php.ini`* path for the *`php.ini`* file used by PHP-FPM.
- *`PHP-FPM pool directory`* ***(nginx only)***: This is the directory from which PHP-FPM reads the pool definitions. ISPConfig creates a pool definition for each vhost that has PHP-FPM enabled; this pool defines under which user PHP runs, if a socket or a TCP connection is used, and it can also contain individual PHP settings.
- *`PHP-FPM start port`* ***(nginx only)***: This defines the minimum TCP port that ISPConfig can allocate to a PHP-FPM pool for a vhost. If you want to change the value, you should do so ***before*** you create the first vhost, because it is otherwise possible that two or more websites use the same PHP-FPM port which can cause security and permissions issues.
- *`PHP-FPM socket directory`* ***(nginx only)***: Instead of using TCP ports, it's also possible to configure nginx vhosts in ISPConfig that use PHP-FPM sockets. This field defines the directory where these sockets will be created.
- *`PHP open_basedir`*: This setting limits the files that can be opened by PHP to the specified directory-tree, including the file itself. This directive is NOT affected by whether Safe Mode is turned On or Off. You can use the placeholder *`[website_path]`* which will be replaced by the path that is set in the *`Website path`* field. You can define multiple directories by separating them with a colon (*`:`*) (don't use spaces).
- *`.htaccess AllowOverride`* ***(Apache only)***: This setting specifies what types of directives are allowed

in *.htaccess* files. Possible values: *All|None|AuthConfig|FileInfo|Indexes|Limit|Options[= Option ,...]* See **http://httpd.apache.org/docs/2.2/mod/core.html#allowoverride** for more details.

- *Apps-vhost port*: ISPConfig allows to install software packages ("apps" - applications) such as phpMyAdmin or Roundcube via the ISPConfig Package Installer (*System > Software > Packages*). These apps will be installed in the */var/www/apps* directory and can be accessed over their own vhost. Specify the port that you want to use for this vhost (default is *8081* - the vhost could then be accessed over *http://example.com:8081*). Please do not use a port that is already in use (such as *80* (http) or *443* (https)).

- *Apps-vhost IP*: Specify an IPv4 address that is configured on your server on which the vhost will listen. It is also possible to use *_default_* (meaning a request to an unspecified address on the *Apps-vhost port* is served from the apps vhost) or a wildcard (*** - meaning requests on all addresses on the port specified by *Apps-vhost port* will be served by the apps vhost).

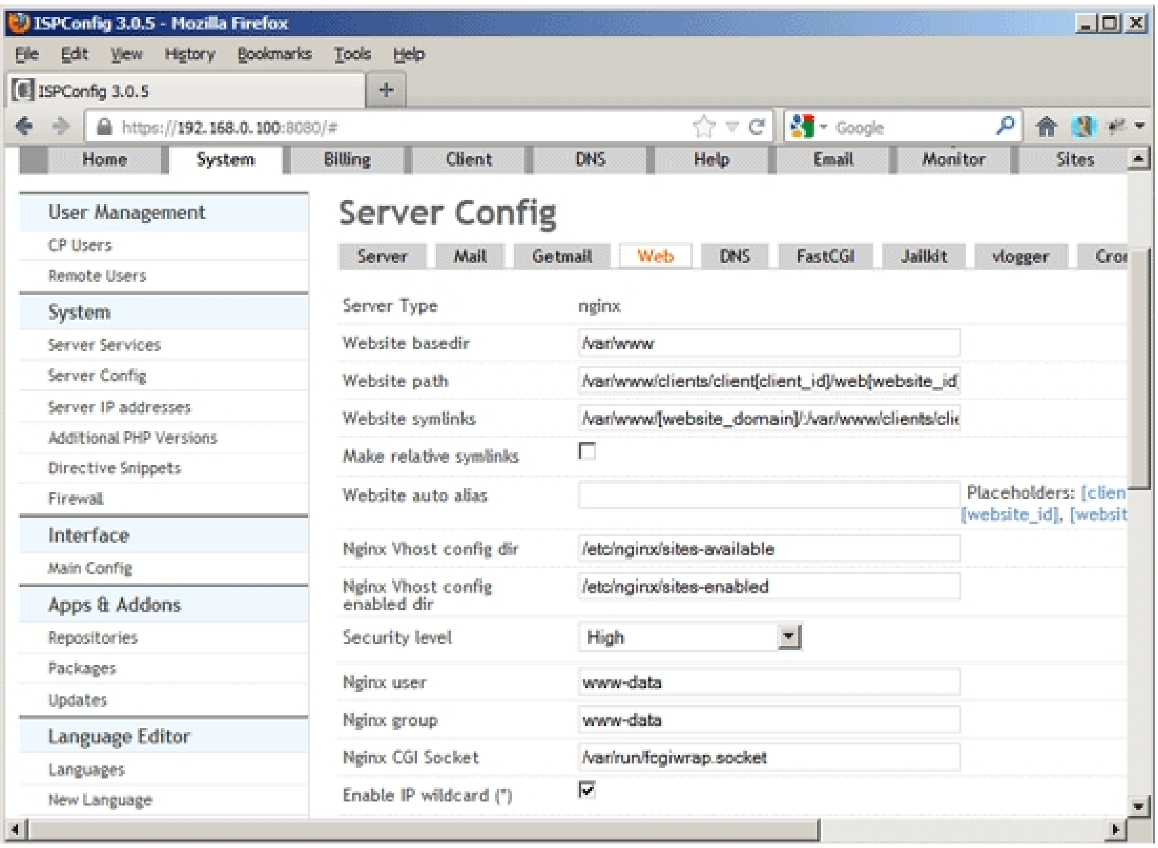
- *Apps-vhost Domain*: Specify the domain that you want to use to access the apps vhost. Examples: *example.com* (-> *http://example.com:8081*), *apps.example.com* (-> *http://apps.example.com:8081*), *www.example.com* (-> *http://www.example.com:8081*). Leave this field empty to use any address (domain, hostname, IP address) that points to the server.

- *awstats conf folder*: This specifies the directory where the web site statistics package ***AWStats*** expects its configuration files. This field is meaningless if you use ***Webalizer*** instead of AWStats.

- *awstats data folder*: This specifies the directory where AWStats creates its data files (from which the reports will be created).

- *awstats.pl script*: This specifies the location of the *awstats.pl* script on the server.

- *awstats_buildstaticpages.pl script*: This specifies the location of the *awstats_buildstaticpages.pl* script on the server. This script creates static HTML pages with statistics - these will be generated once a day (at 0.30h) and are available in the */stats* folder of your web site (e.g. *http://www.example.com/stats*).



Server Config

Server | Mail | Getmail | Web | DNS | FastCGI | Jailkit | vlogger | Cro

Server Type: nginx

Website basedir: /var/www

Website path: /var/www/clients/client[client_id]/web[website_id]

Website symlinks: /var/www/[website_domain]/:/var/www/clients/clie

Make relative symlinks: □

Website auto alias: Placeholders: [clien [website_id], [websit

Nginx Vhost config dir: /etc/nginx/sites-available

Nginx Vhost config enabled dir: /etc/nginx/sites-enabled

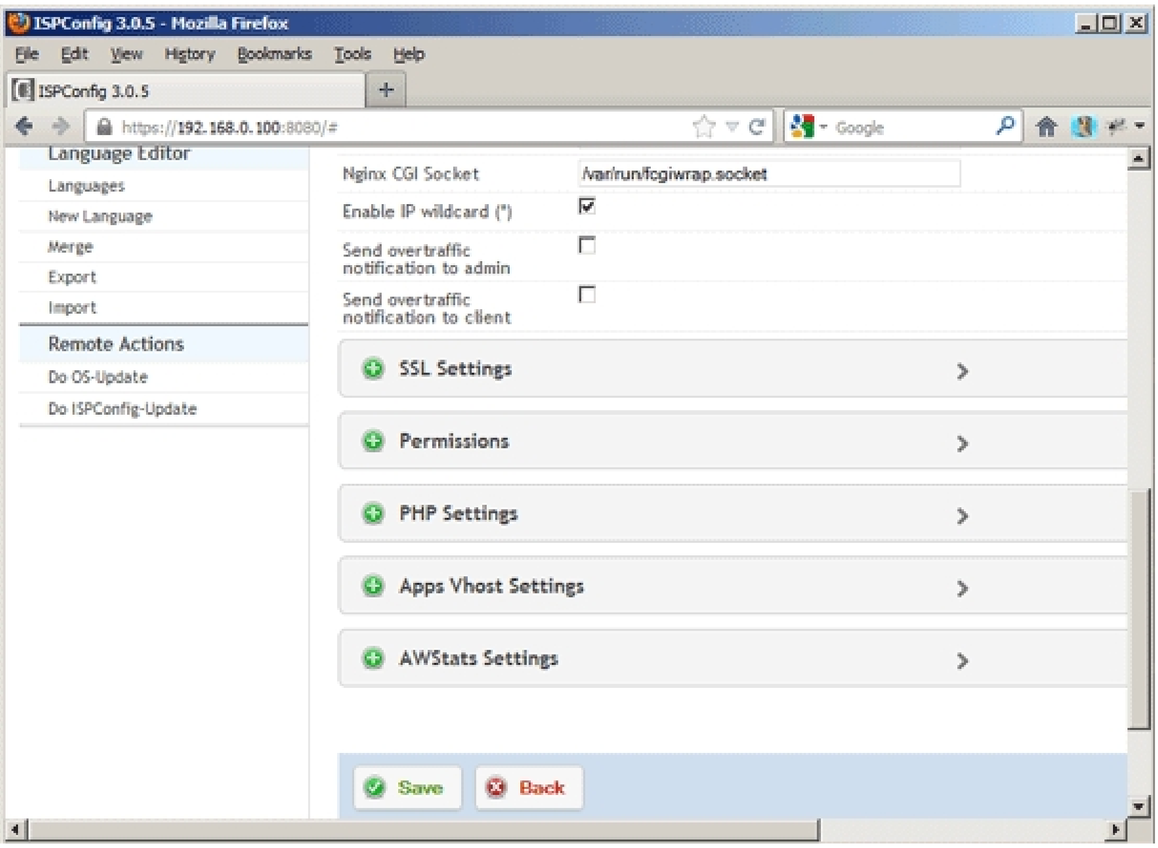
Security level: High

Nginx user: www-data

Nginx group: www-data

Nginx CGI Socket: /var/run/fcgiwrap.socket

Enable IP wildcard (*): ☑



Nginx CGI Socket: /var/run/fcgiwrap.socket

Enable IP wildcard (*): ☑

Send overtraffic notification to admin: □

Send overtraffic notification to client: □

SSL Settings

Permissions

PHP Settings

Apps Vhost Settings

AWStats Settings

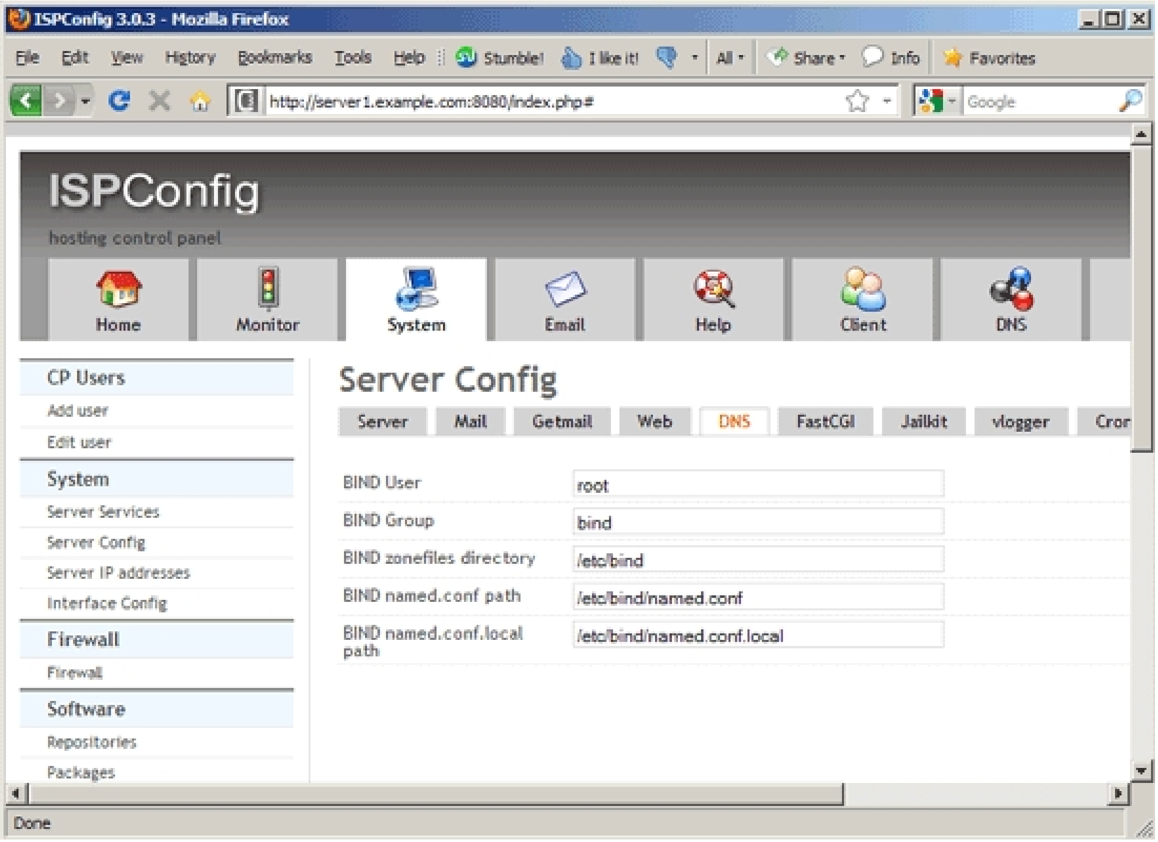
Save | Back

DNS

If you use the BIND nameserver (instead of MyDNS), you can configure basic BIND settings on this tab.

The form has the following fields:

- *`BIND User`*: This is the system user that BIND runs under.
- *`BIND Group`*: This is the system group that BIND runs under.
- *`BIND zonefiles directory`*: This is the directory where BIND will place its zone files (Debian: *`/etc/bind`*).
- *`BIND named.conf path`*: This is the location where BIND expects its configuration file *`named.conf`* (Debian: *`/etc/bind/named.conf`*).
- *`BIND named.conf.local path`*: This is the location of the *`named.conf.local`* file that is included in *`named.conf`* and which includes the zone files created by ISPConfig.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
ISPConfig
hosting control panel
Home
Monitor
System
Email
Help
Client
DNS
CP Users
Add user
Edit user
System
Server Services
Server Config
Server IP addresses
Interface Config
Firewall
Firewall
Software
Repositories
Packages
Server Mail Getmail Web DNS FastCGI Jailkit vlogger Cron
Done

Server Config

BIND User	root
BIND Group	bind
BIND zonefiles directory	/etc/bind
BIND named.conf path	/etc/bind/named.conf
BIND named.conf.local path	/etc/bind/named.conf.local

FastCGI

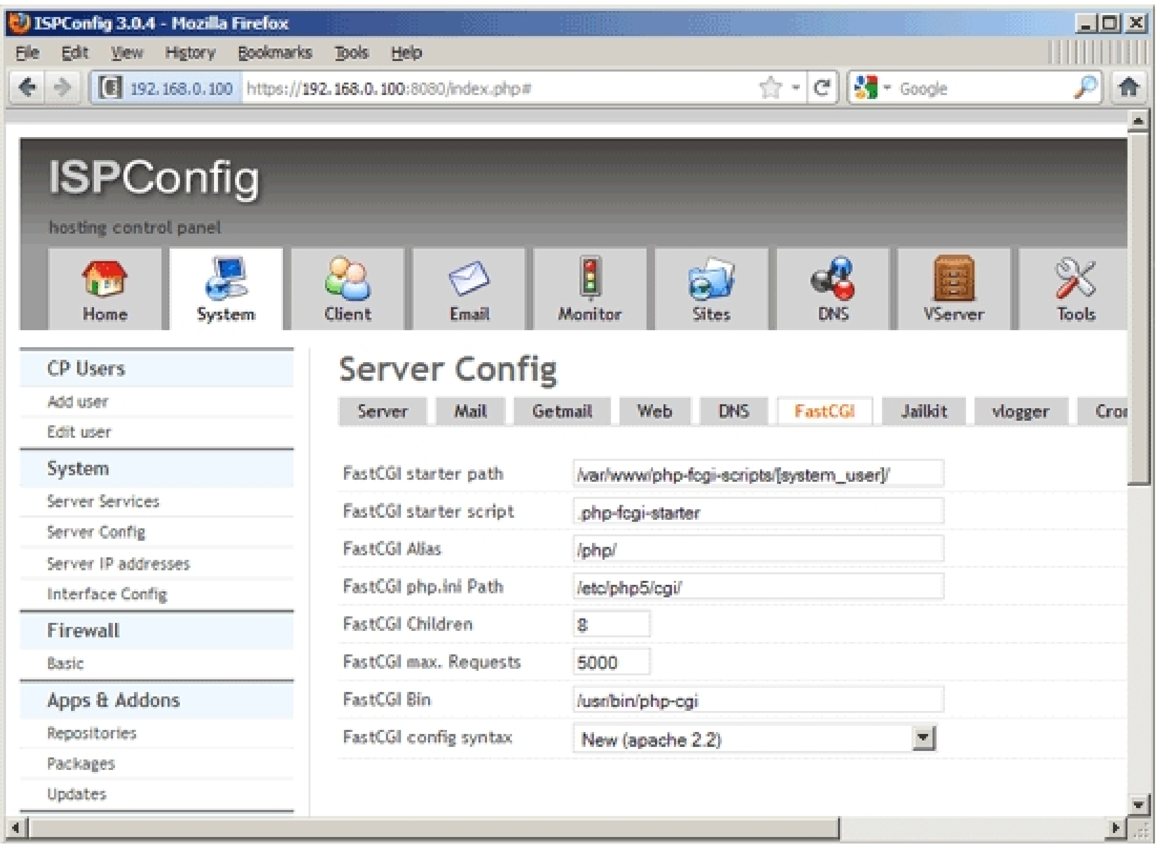
On this tab you can configure basic FastCGI settings that are relevant if you use PHP via FastCGI. ***Please note that these settings are relevant only if you use Apache2.*** They are ignored by nginx.

The form has the following fields:

- *`FastCGI starter path`*: We will run PHP using suExec; suExec's document root is *`/var/www`* (Debian/Ubuntu/Fedora/CentOS) or *`/srv/www`* (OpenSUSE). Therefore we cannot call the PHP binary directly because it is located outside suExec's document root. As suExec does not allow symlinks, the only way to solve the problem is to create a wrapper script for each web site in a subdirectory of *`/var/www`* or *`/srv/www`*; the wrapper script will then call the PHP binary. In this field you can specify the directory (should be a subdirectory of *`/var/www`* or *`/srv/www`*) where the wrapper script will be located. You can use the placeholder *`[system_user]`* which will be replaced by the system user that owns the web site, e.g. *`web1`*.
- *`FastCGI starter script`*: This is the name of the FastCGI wrapper script. Example: *`.php-fcgi-starter`*
- *`FastCGI Alias`*: (not in use right now; see ***http://www.fastcgi.com/docs/faq.html#FastCGIExternalServer*** for more details.) Since all FastCGI directives are global (they are not configured in a server context), all FastCGI paths map to the filesystem. In the case of external servers, this path does not have anything to do with the file system; it is a virtual file system path. Since the connection between mod_fastcgi and the FastCGI app is by a socket (unix or tcp), mod_fastcgi does not care where the program is (it could be on a completely different machine). However, mod_fastcgi needs to know when a hit is calling for an external app, so it uses this path as if it were a local filesystem path. Apache translates a request URI to a filesystem path.

Example: *`FastCGIExternalServer /var/www/htdocs/extprog -host 127.0.0.1:9000`*

- *`FastCGI php.ini Path`*: This is the full *`php.ini`* path for the *`php.ini`* used by FastCGI.
- *`FastCGI Children`*: This defines the number of PHP children that will be launched. (This variable is onyl useful for lighttpd or nginx as Apache mod_fcgi will control the number of children itself and never use the additional processes.)
- *`FastCGI max. Requests`*: This is the maximum number of requests before an fcgid process is stopped and a new one is launched.
- *`FastCGI Bin`*: This is the path to the FastCGI PHP binary.
- *`FastCGI config syntax`*: Please choose your Apache version here (*`Old (apache 2.0)`* or *`New (apache 2.2)`*). This defines how the FastCGI directives are written into vhost configuration files because the syntax has changed between Apache 2.0 and 2.2 (see ***http://httpd.apache.org/mod_fcgid/mod/mod_fcgid.html#upgrade***).



FastCGI starter path	/var/www/php-fcgi-scripts/[system_user]/
FastCGI starter script	.php-fcgi-starter
FastCGI Alias	/php/
FastCGI php.ini Path	/etc/php5/cgi/
FastCGI Children	8
FastCGI max. Requests	5000
FastCGI Bin	/usr/bin/php-cgi
FastCGI config syntax	New (apache 2.2)

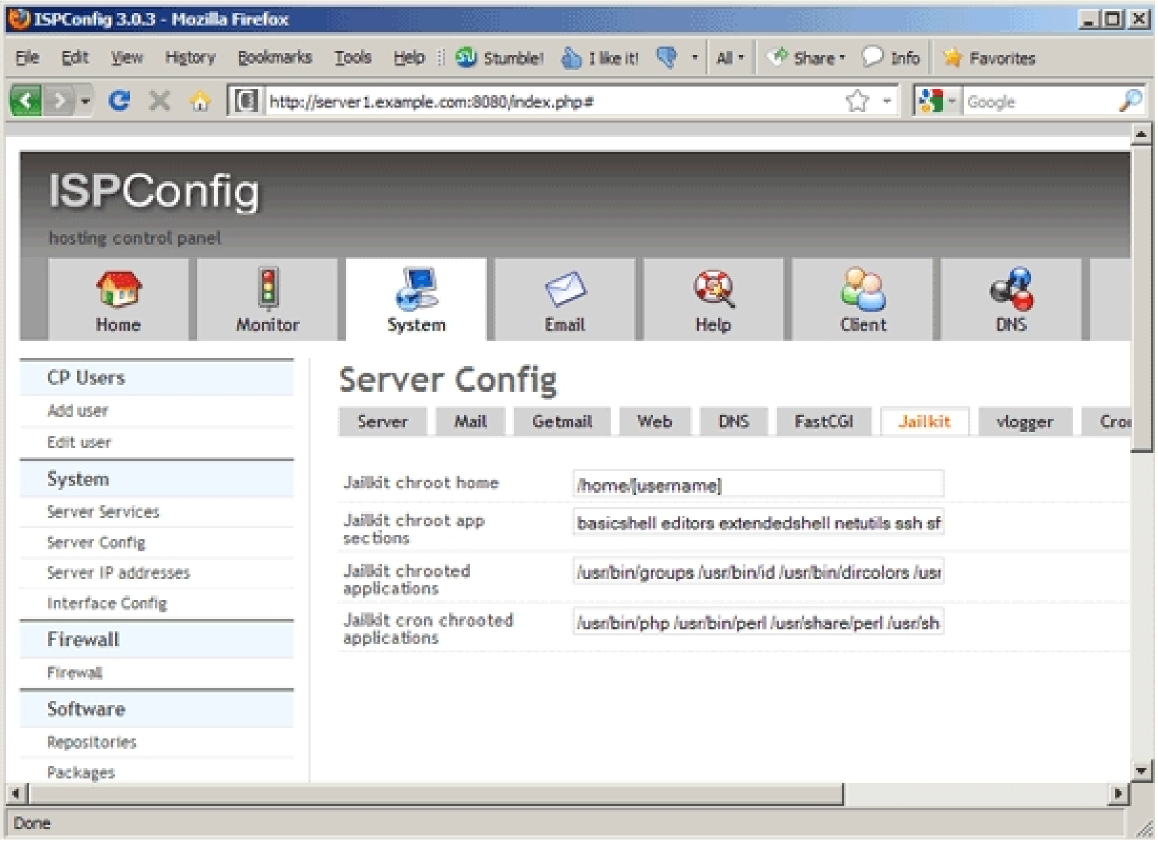
Jailkit

Here you can configure the basic ***Jailkit*** behaviour. Jailkit is a set of utilities to limit user accounts to specific files using chroot() and or specific commands. You can make a shell user use Jailkit by selecting it in the Chroot Shell drop-down menu of the shell user under *Sites > Shell > Shell-User*.

The form has the following fields:

- *Jailkit chroot home*: This is the directory where jailkit users will be chrooted. The placeholder *[username]* will be replaced with the actual system user name. Example: */home/[username]*
- *Jailkit chroot app sections*: These are predefinded sets of applications/programs that chrooted users can use. These sets are defined in */etc/jailkit/jk_init.ini*. Separate multiple entries with a space. Example: *basicshell editors extendedshell netutils ssh sftp scp groups jk_lsh*
- *Jailkit chrooted applications*: In this field you can explicitly list single applications/programs that chrooted users will be able to use (it is possible that these applications/programs are already part of the predefined sets of applications that you've enabled in the *Jailkit chroot app sections* field). Separate multiple entries with a space. Example: */usr/bin/groups /usr/bin/id /usr/bin/dircolors /usr/bin/lesspipe /usr/bin/basename /usr/bin/dirname /usr/bin/nano /usr/bin/pico*

- `Jailkit cron chrooted applications`: Under `Sites > Cron > Cron Jobs` you can define cron jobs. If `Chrooted Cron` is selected in the limits of the client that owns the cron job, the cron jobs are chrooted (using Jailkit). In this field you can explicitly list single applications/programs that chrooted cron jobs will be able to use. Separate multiple entries with a space. Example: `/usr/bin/php /usr/bin/perl /usr/share/perl /usr/share/php`



Jailkit chroot home: /home/[username]

Jailkit chroot app sections: basicshell editors extendedshell netutils ssh sf

Jailkit chrooted applications: /usr/bin/groups /usr/bin/id /usr/bin/dircolors /usr

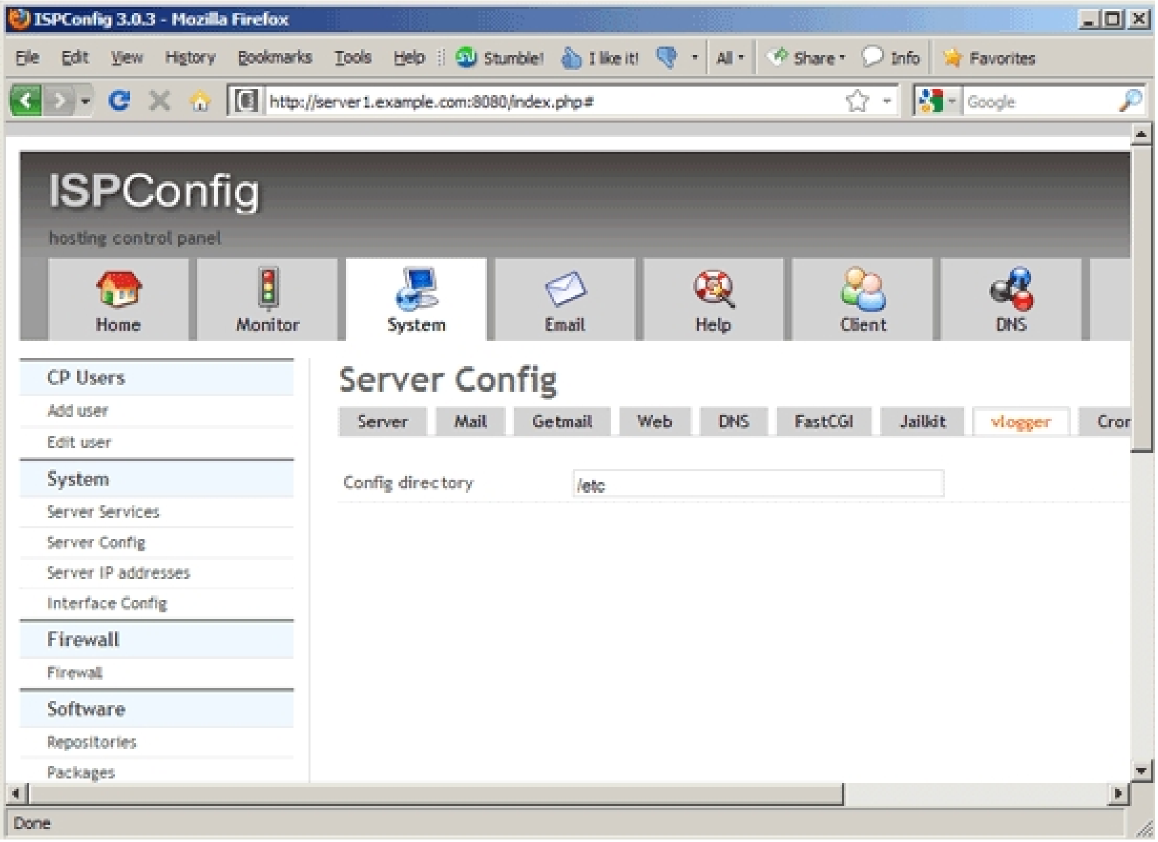
Jailkit cron chrooted applications: /usr/bin/php /usr/bin/perl /usr/share/perl /usr/sh

vlogger

vlogger is a little tool that takes the burden of creating Apache virtual host logfiles off of Apache so that Apache doesn't have to deal with open logfiles. ***Please note that these settings are relevant only if you use Apache2.*** They are ignored by nginx.

The form has the following field:

- `Config directory`: This defines the directory where vlogger expects its configuration file.



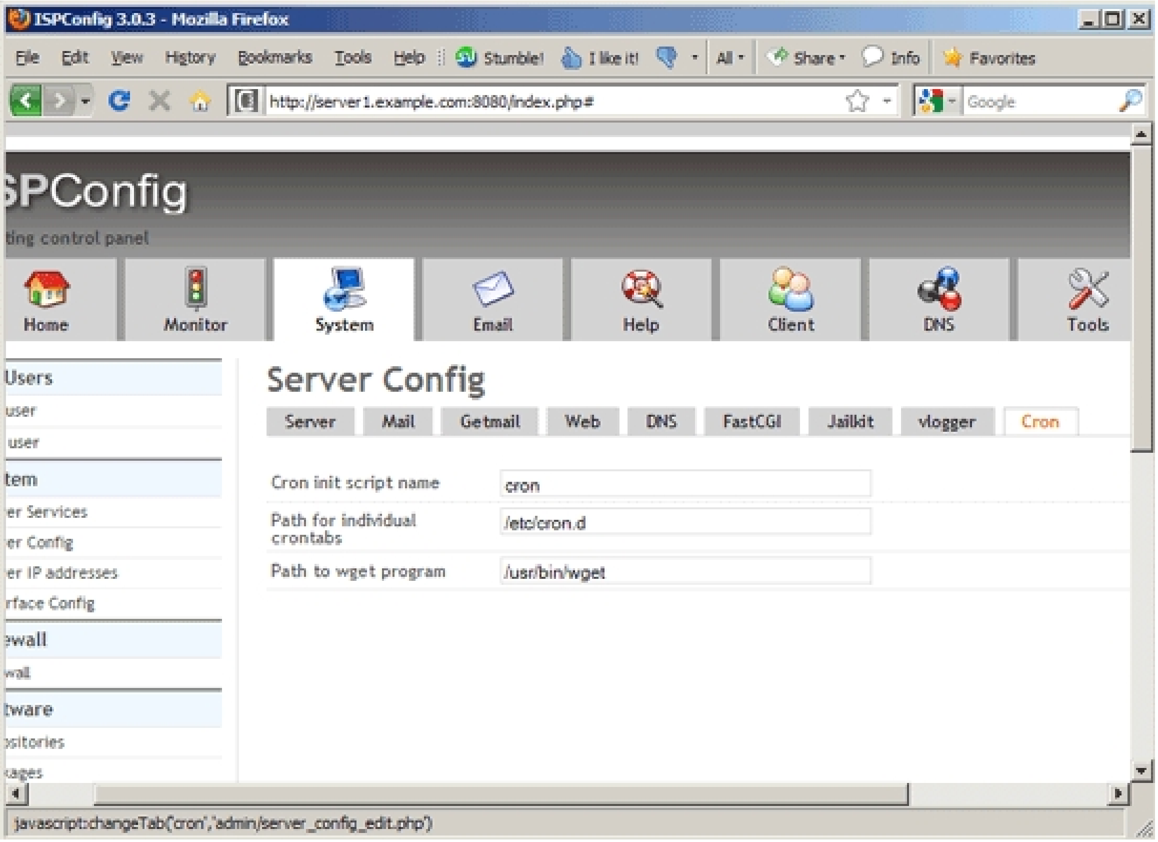
Config directory: /etc

Cron

On this tab you can configure a few settings for cron.

The form has the following fields:

- *Cron init script name*: This is the name of the cron init script that is located in the */etc/init.d/* directory.
- *Path for individual crontabs*: This is the directory where cron jobs will be created by ISPConfig. This must be a directory where cron expects to find cron jobs, e.g. */etc/cron.d*.
- *Path to wget program*: This is the path to the *wget* program, e.g. */usr/bin/wget*. If you specify a URL in the *Command to run* field under *Sites > Cron > Cron Jobs*, it will automatically be executed via wget, that's why cron needs to know the exact path.



Cron init script name: cron

Path for individual crontabs: /etc/cron.d

Path to wget program: /usr/bin/wget

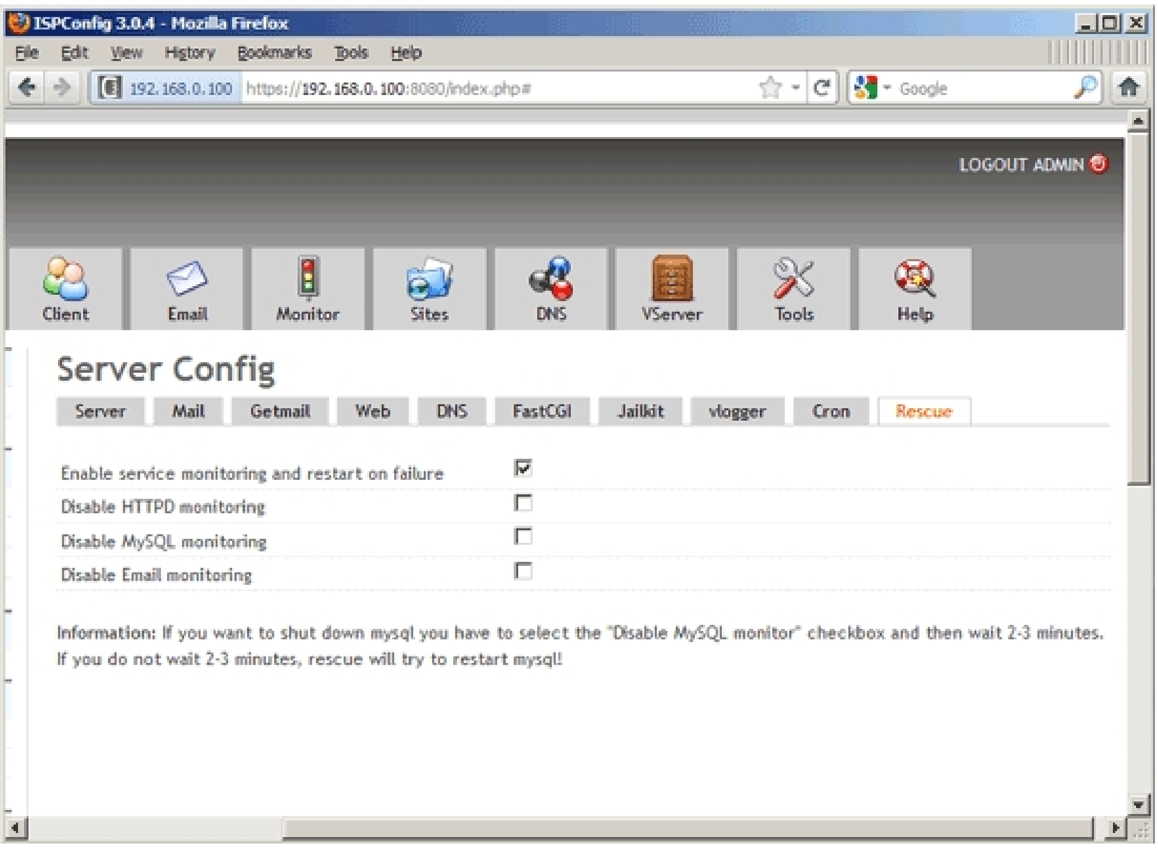
javascript:changeTab('cron','admin/server_config_edit.php')

Rescue

On this tab you can configure ISPConfig to check the services HTTPD (Apache2/nginx), MySQL, and email (Postfix/Dovecot/Courier) if they are up and running and to restart them if they are not (similar to ***monit***).

The form has the following fields:

- *`Enable service monitoring and restart on failure`*: If this box is checked, ISPConfig will check the services http, MySQL, and email each minute and restart them if they are down.
- *`Disable HTTPD monitoring`*: Check this box if you do not want ISPConfig to monitor/restart HTTPD (Apache2/nginx).
- *`Disable MySQL monitoring`*: Check this box if you do not want ISPConfig to monitor/restart MySQL. If you want to shut down MySQL you have to select the "Disable MySQL monitoring" checkbox and then wait 2-3 minutes. If you do not wait 2-3 minutes, rescue will try to restart MySQL!
- *`Disable Email monitoring`*: Check this box if you do not want ISPConfig to monitor/restart email services (Postfix/Dovecot/Courier).


ISPConfig 3.0.4 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
192.168.0.100 https://192.168.0.100:8080/index.php#
Google
LOGOUT ADMIN
Client
Email
Monitor
Sites
DNS
VServer
Tools
Help
Server Config
Server
Mail
Getmail
Web
DNS
FastCGI
Jailkit
vlogger
Cron
Rescue

Enable service monitoring and restart on failure ☑

Disable HTTPD monitoring □

Disable MySQL monitoring □

Disable Email monitoring □

Information: If you want to shut down mysql you have to select the "Disable MySQL monitor" checkbox and then wait 2-3 minutes. If you do not wait 2-3 minutes, rescue will try to restart mysql!

4.9.2.3 Server IP addresses

Here you can add additional IP addresses to your server. If you've enabled automatic network configuration for your server (field `Network Configuration` on the `Server` tab under `Server Config`), these additional IP addresses will be configured automatically (please note that this works only on Debian/Ubuntu servers and if you have one network card which is named `eth0`). However, it is recommended to configure additional IP addresses manually (see chapter ***5.18 How Do I Manually Configure New IP Addresses On My System?***) and then add them here so that ISPConfig knows that they exist.

To create a new IP address, click on the `Add new IP Address` button. This will lead you to the `IP Addresses` form with the tab `IP Address`.

IP Addresses

IP Address

The form has the following fields:

- `Server`: If more than one server is available, you can select the server on which the IP address is/will be located.

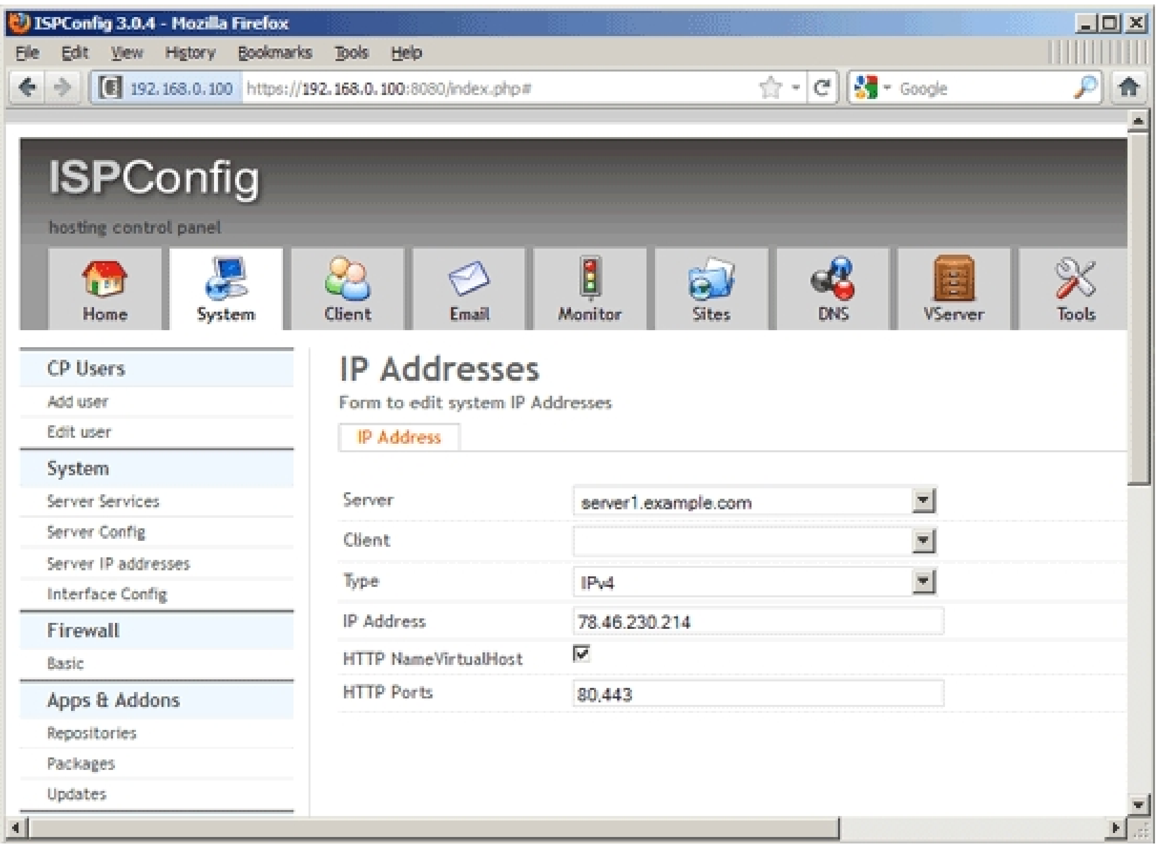
- *Client*: If you select a client here, the IP address is available only to this client, i.e., he can select it if he creates a new website.

- *Type*: Please select if this is an IPv4 or an IPv6 address.

- *IP Address*: Type in the IP address (IPv4 or IPv6, depending on what you selected in the *Type* field). This should be an IP address that you see in the output of the shell command *ifconfig*. If your server is in a data center, this is probably a public IP address. If your server is in a local network, this should be a local IP address and not your router's public IP address. Example: *1.2.3.4*

- *HTTP NameVirtualHost*: If you check this field, you can select this IP address for a new web sites in the *Sites* module; otherwise it cannot be used for Apache vhosts.

- *HTTP Ports*: Please specify the ports (comma-separated) to be used in Apache's *NameVirtualHost* directives for this IP address. The default value is *80,443*.



IP Addresses

Form to edit system IP Addresses

IP Address

Server	server1.example.com
Client	
Type	IPv4
IP Address	78.46.230.214
HTTP NameVirtualHost	☑
HTTP Ports	80,443

4.9.2.4 Additional PHP Versions

As of version 3.0.5, ISPConfig supports multiple PHP versions for the PHP modes FastCGI and PHP-FPM (for PHP-FPM, PHP versions must be >= 5.3.0 because prior to that version, PHP-FPM's configuration syntax was different; for FastCGI, there is no such limitation). This means that you can now have, for example, PHP 5.3 and

5.4 installed in parallel on your servers and select the PHP version that suits your web application best.

The PHP version that comes with your distribution and that you have installed through your distribution's package manager is the default PHP version (and it isn't limited to the PHP modes FastCGI and PHP-FPM), and you don't have to define it here. All other PHP versions that you have built manually must be defined here so that they are available in the PHP Version drop-down menu in the web site settings.

I will show now how to build PHP 5.3.18 on Ubuntu 12.04/12.10 so that it can be used on the same server while the default PHP is installed. I will install PHP 5.3.18 in the `/opt/php-5.3.18` directory.

Download and extract PHP 5.3.18:

```
mkdir /opt/php-5.3.18

mkdir /usr/local/src/php5-build

cd /usr/local/src/php5-build

wget http://de.php.net/get/php-5.3.18.tar.bz2/from/this/mirror -O php-5.3.18.tar.bz2

tar jxf php-5.3.18.tar.bz2
```

```
cd php-5.3.18/
```

Install the prerequisites for building PHP5:

```
apt-get build-dep php5
```

```
apt-get install libfcgi-dev libfcgi0ldbl libjpeg62-dbg libmcrypt-dev libssl-dev
```

Configure an build PHP 5.3.18 as follows (you can adjust the `./configure` command to your needs, take a look at

```
./configure --help
```

to see all available options; if you use a different `./configure` command, it is possible that additional libraries are required, or the build process will fail) - PHP-FPM and FastCGI are mutually exclusive in PHP 5.3, that's why I show two ways of building PHP 5.3, one for PHP-FPM, one for FastCGI, however you can compile PHP twice with both configuration options to get both:

PHP-FPM

```
./configure \

--prefix=/opt/php-5.3.18 \
--with-pdo-pgsql \

--with-zlib-dir \

--with-freetype-dir \

--enable-fpm \

--enable-mbstring \

--with-libxml-dir=/usr \

--enable-soap \

--enable-calendar \

--with-curl --with-mcrypt \

--with-zlib \

--with-gd \

--with-pgsql \

--disable-rpath \

--enable-inline-optimization \

--with-bz2 \

--with-zlib \

--enable-sockets \

--enable-sysvsem \

--enable-sysvshm \

--enable-pcntl \

--enable-mbregex \

--with-mhash \

--enable-zip   \
--with-pcre-regex \

--with-mysql \

--with-pdo-mysql \

--with-mysqli \

--with-jpeg-dir=/usr \

--with-png-dir=/usr \

--enable-gd-native-ttf \

--with-openssl \

--with-fpm-user=www-data \

--with-fpm-group=www-data \

--with-libdir=/lib/x86_64-linux-gnu
```

```
make

make install
```

Copy *php.ini* and *php-fpm.conf* (if you've compiled PHP with FPM) to the correct locations:

```
cp /usr/local/src/php5-build/php-5.3.18/php.ini-production /opt/php-5.3.18/lib/php.ini
```

```
cp /opt/php-5.3.18/etc/php-fpm.conf.default /opt/php-5.3.18/etc/php-fpm.conf
```

Open */opt/php-5.3.18/etc/php-fpm.conf* and adjust the following settings - in the *listen* line you must use an unused port (e.g. *8999*; port *9000* might be in use by Ubuntu's default PHP-FPM already), and you must add the line *include=/opt/php-5.3.18/etc/pool.d/*.conf* at the end:

```
vi /opt/php-5.3.18/etc/php-fpm.conf
```

```
[...]
pid = run/php-fpm.pid
[...]
user = www-data
group = www-data
[...]
listen = 127.0.0.1:8999
```

```
[...]
include=/opt/php-5.3.18/etc/pool.d/*.conf
```

Create the pool directory for PHP-FPM:

```
mkdir /opt/php-5.3.18/etc/pool.d
```

Next create an init script for PHP-FPM:

```
vi /etc/init.d/php-5.3.18-fpm
```

```
#! /bin/sh
### BEGIN INIT INFO
# Provides:          php-5.3.18-fpm
# Required-Start:    $all
# Required-Stop:     $all
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
# Short-Description: starts php-5.3.18-fpm
# Description:       starts the PHP FastCGI Process Manager daemon
### END INIT INFO
php_fpm_BIN=/opt/php-5.3.18/sbin/php-fpm
php_fpm_CONF=/opt/php-5.3.18/etc/php-fpm.conf
php_fpm_PID=/opt/php-5.3.18/var/run/php-fpm.pid
php_opts="--fpm-config $php_fpm_CONF"

wait_for_pid () {
        try=0
        while test $try -lt 35 ; do
                case "$1" in
                        'created')
                        if [ -f "$2" ] ; then
                                try="
                                break
                        fi
                        ;;
                        'removed')
                        if [ ! -f "$2" ] ; then
                                try="
                                break
                        fi
                        ;;
                esac
                echo -n .
                try=`expr $try + 1`
                sleep 1
        done
}
case "$1" in
        start)
                echo -n "Starting php-fpm "
                $php_fpm_BIN $php_opts
                if [ "$?" != 0 ] ; then
                        echo " failed"
                        exit 1
                fi
                wait_for_pid created $php_fpm_PID
                if [ -n "$try" ] ; then
                        echo " failed"
                        exit 1
                else
                        echo " done"
                fi
        ;;
        stop)
                echo -n "Gracefully shutting down php-fpm "
                if [ ! -r $php_fpm_PID ] ; then
                        echo "warning, no pid file found - php-fpm is not running ?"
                        exit 1
                fi
                kill -QUIT `cat $php_fpm_PID`
                wait_for_pid removed $php_fpm_PID
                if [ -n "$try" ] ; then
                        echo " failed. Use force-exit"
                        exit 1
                else
                        echo " done"
                       echo " done"
                fi
        ;;
        force-quit)
                echo -n "Terminating php-fpm "
                if [ ! -r $php_fpm_PID ] ; then
                        echo "warning, no pid file found - php-fpm is not running ?"
                        exit 1
                fi
                kill -TERM `cat $php_fpm_PID`
                wait_for_pid removed $php_fpm_PID
                if [ -n "$try" ] ; then
                        echo " failed"
                        exit 1
                else
                        echo " done"
        fi
    ;;
    restart)
        $0 stop
        $0 start
    ;;
    reload)
        echo -n "Reload service php-fpm "
        if [ ! -r $php_fpm_PID ] ; then
            echo "warning, no pid file found - php-fpm is not running ?"
            exit 1
        fi
        kill -USR2 `cat $php_fpm_PID`
        echo " done"
    ;;
    *)
        echo "Usage: $0 {start|stop|force-quit|restart|reload}"
        exit 1
    ;;
esac
```

Make the init script executable and create the system startup links:

```
chmod 755 /etc/init.d/php-5.3.18-fpm

insserv php-5.3.18-fpm
```

Finally start PHP-FPM:

```
/etc/init.d/php-5.3.18-fpm start
```

PHP With FastCGI

```
./configure \

--prefix=/opt/php-5.3.18 \

--with-pdo-pgsql \

--with-zlib-dir \

--with-freetype-dir \

--enable-cgi \
--enable-mbstring \

--with-libxml-dir=/usr \

--enable-soap \

--enable-calendar \

--with-curl --with-mcrypt \

--with-zlib \

--with-gd \

--with-pgsql \

--disable-rpath \

--enable-inline-optimization \

--with-bz2 \

--with-zlib \

--enable-sockets \

--enable-sysvsem \

--enable-sysvshm \

--enable-pcntl \

--enable-mbregex \

--with-mhash \

--enable-zip   \

--with-pcre-regex \

--with-mysql \

--with-pdo-mysql \

--with-mysqli \

--with-jpeg-dir=/usr \
--with-png-dir=/usr \

--enable-gd-native-ttf \

--with-openssl \

--with-libdir=/lib/x86_64-linux-gnu
```

```
make

make install
```

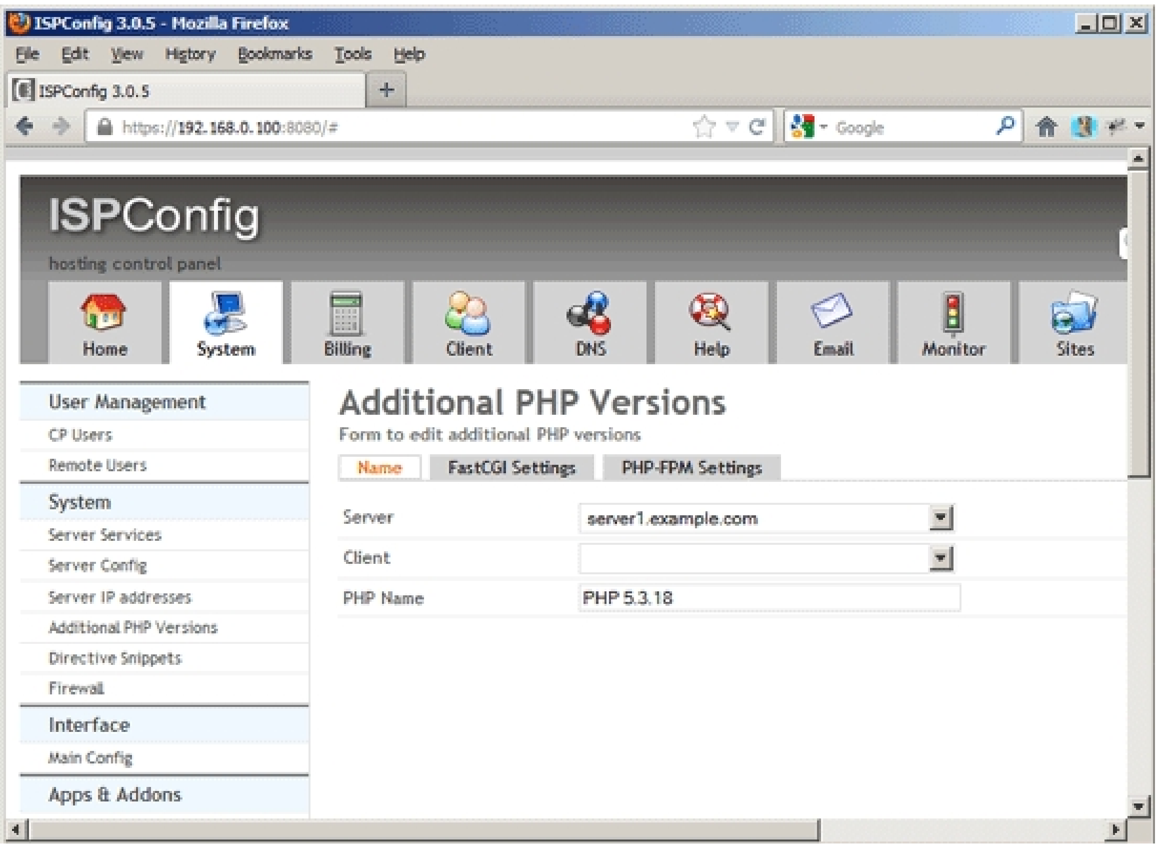
Copy *php.ini* to the correct location:

```
cp /usr/local/src/php5-build/php-5.3.18/php.ini-production /opt/php-5.3.18/lib/php.ini
```

ISPConfig Configuration

In ISPConfig 3.0.5, you can configure the new PHP version under *System > Additional PHP Versions*. On the *Name* tab, you just fill in a name for the PHP version (e.g. *PHP 5.3.18*) - this PHP version will be listed under this name in the website settings in ISPConfig:

- *Server*: Select the server where the PHP version is installed.
- *Client*: If this PHP version should be available for just for one specific client, select that client here. Otherwise, this PHP version will be available for all clients.
- *PHP Name*: Specify a name for this PHP version (e.g. *PHP 5.3.18*).



Additional PHP Versions

Form to edit additional PHP versions

Name | FastCGI Settings | PHP-FPM Settings

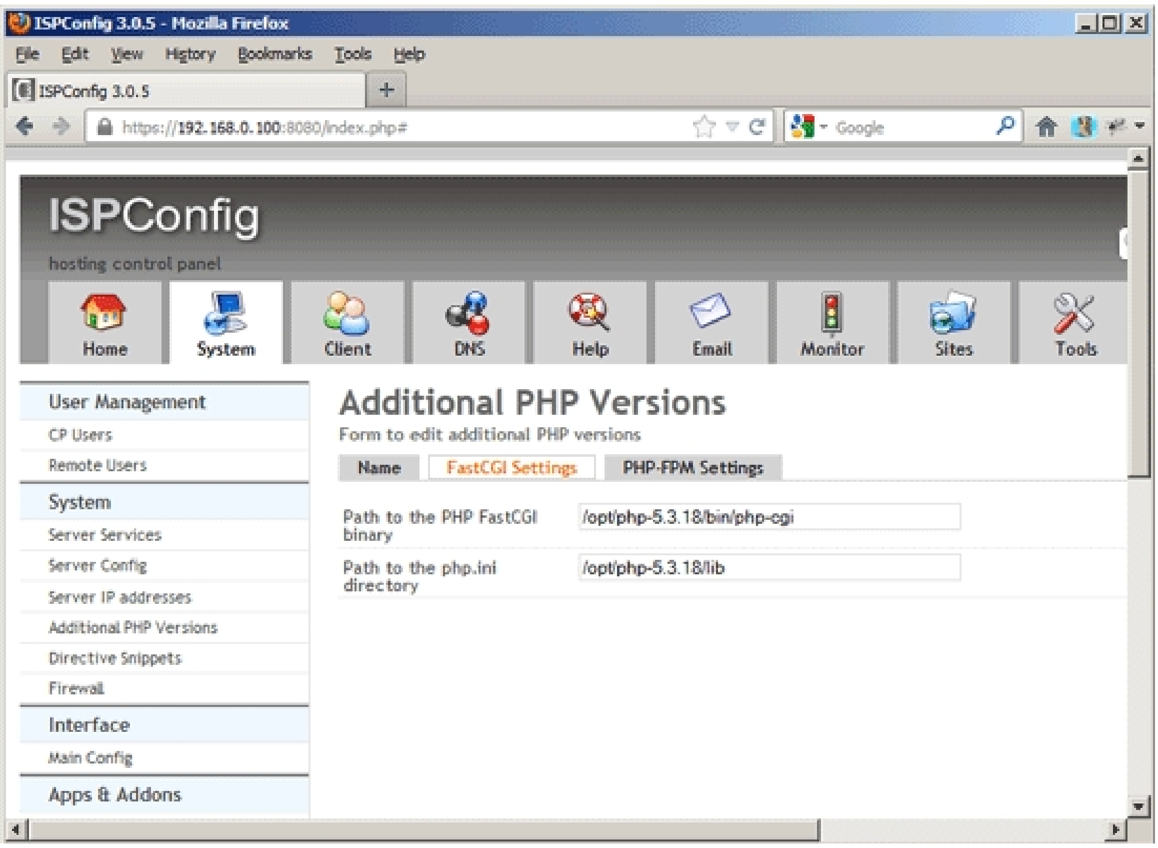
Server: server1.example.com

Client:

PHP Name: PHP 5.3.18

If you want to use this PHP version with FastCGI, go to the `FastCGI Settings` tab (the *PHP-FPM Settings* tab can be left empty) and fill out the fields as follows:

- *Path to the PHP FastCGI binary*: Specify the path to the PHP FastCGI binary (e.g. */opt/php-5.3.18/bin/php-cgi*).
- *Path to the php.ini directory*: Specify the path of the directory where the php.ini file for this PHP version is located (e.g. */opt/php-5.3.18/lib*).



Additional PHP Versions

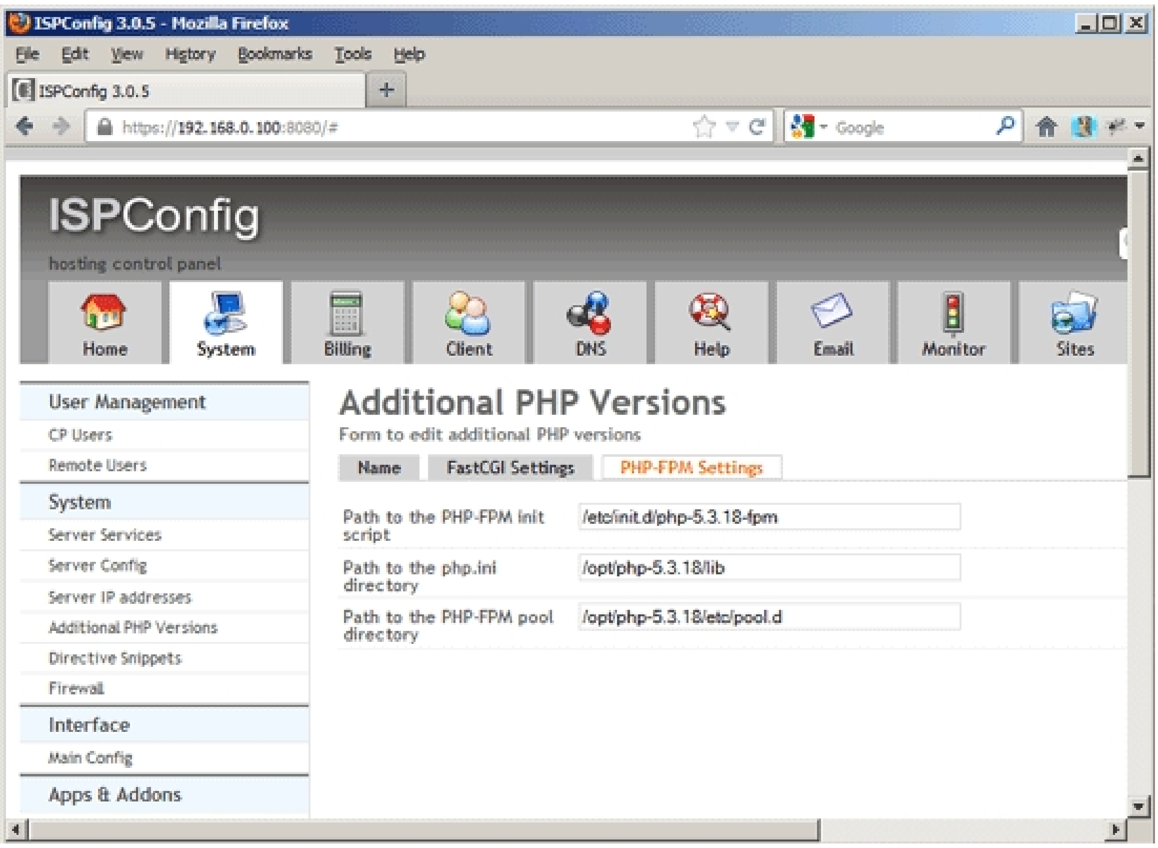
Form to edit additional PHP versions

Path to the PHP FastCGI binary: /opt/php-5.3.18/bin/php-cgi

Path to the php.ini directory: /opt/php-5.3.18/lib

If you want to use this PHP version with PHP-FPM, go to the *PHP-FPM Settings* tab (the *FastCGI Settings* tab can be left empty) and fill out the fields as follows:

- *Path to the PHP-FPM init script*: Type in the path to the PHP-FPM init script (e.g. */etc/init.d/php-5.3.18-fpm*).
- *Path to the php.ini directory*: Specify the path of the directory where the php.ini file for this PHP version is located (e.g. */opt/php-5.3.18/lib*).
- *Path to the PHP-FPM pool directory*: Specify the path of the directory where ISPConfig will store the PHP-FPM pool definitions (e.g. */opt/php-5.3.18/etc/pool.d*).



Additional PHP Versions

Form to edit additional PHP versions

Path to the PHP-FPM init script	/etc/init.d/php-5.3.18-fpm
Path to the php.ini directory	/opt/php-5.3.18/lib
Path to the PHP-FPM pool directory	/opt/php-5.3.18/etc/pool.d

4.9.2.5 Directive Snippets

On the *`Options`* tab of a web site in ISPConfig, you find text areas for advanced configurations for PHP (*`PHP Directives`*), Apache/nginx (*`Apache Directives`*/*`nginx Directives`*), and nginx proxy settings (*`Proxy Directives`*). It is possible that lots of the settings that you use there are the same or nearly the same for multiple web sites. To make your life easier so that you don't have to type the same directives again and again, the idea of *`Directive Snippets`* was born. *`Directive Snippets`* allow you to save configuration snippets for PHP, Apache, nginx, and nginx proxy settings, and these saved snippets are available for the suitable field on the web site's *`Options`* tab. The name of the snippets is displayed as a link next to the text area, and when you click on that link, the snippet is inserted into the text area.

Directive Snippets

Directive Snippets

The form has the following fields:

- *`Name of Snippet`*: Specify a name for the snippet (e.g. *`PHP for Hosting Package Gold`* or *`nginx WordPress Configuration`*).
- *`Type`*: Select the type of the snippet (*`Apache`*, *`nginx`*, *`PHP`*, *`Proxy`*).

- *Snippet*: Type in your configuration settings/directives. Examples:

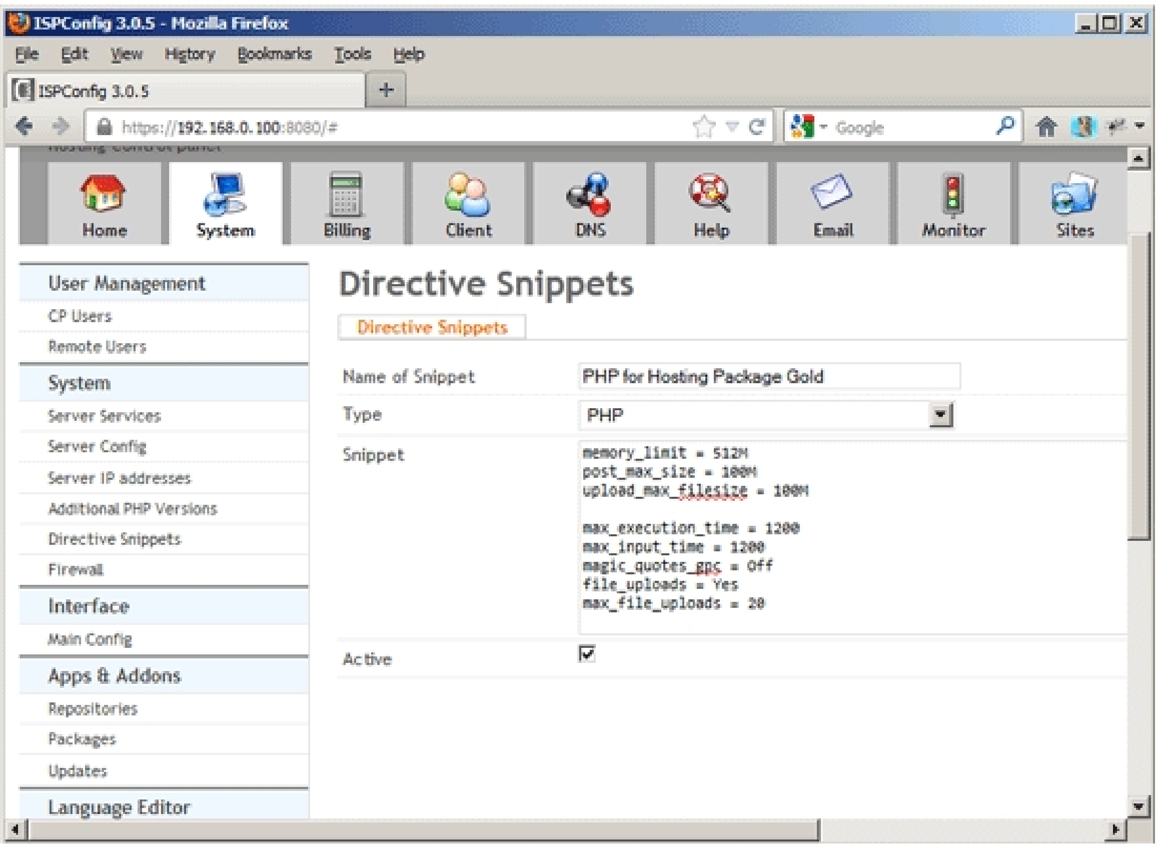
PHP:

```
 memory_limit = 512M
  post_max_size = 100M
  upload_max_filesize = 100M
  max_execution_time = 1200
  max_input_time = 1200
  magic_quotes_gpc = Off
  file_uploads = Yes
max_file_uploads = 20
```

nginx:

```
  client_max_body_size 20M;
    location / {
         try_files $uri $uri/ /index.php?$args;
}

# Add trailing slash to */wp-admin requests.
rewrite /wp-admin$ $scheme://$host$uri/ permanent;

location ~*  .(jpg|jpeg|png|gif|css|js|ico)$ {
         expires max;
         log_not_found off;
}
```

- *Active*: Specify whether this snippet is active or not.



Directive Snippets

Directive Snippets

Name of Snippet	PHP for Hosting Package Gold
Type	PHP

Snippet	memory_limit = 512M
	post_max_size = 100M
	upload_max_filesize = 100M
	
	max_execution_time = 1200
	max_input_time = 1200
	magic_quotes_gpc = Off
	file_uploads = Yes
	max_file_uploads = 20

Active	☑

4.9.2.6 Firewall

This is where we can enable the firewall for a server. For each server controlled by ISPConfig, there can be just one firewall record. If there's no firewall record for a server, the firewall is not active on that server.

To create a new firewall record, click on the *`Add Firewall record`* button. This will lead you to the *`Firewall`* form with the tab *`Firewall`*.

Firewall

Firewall

The form has the following fields:

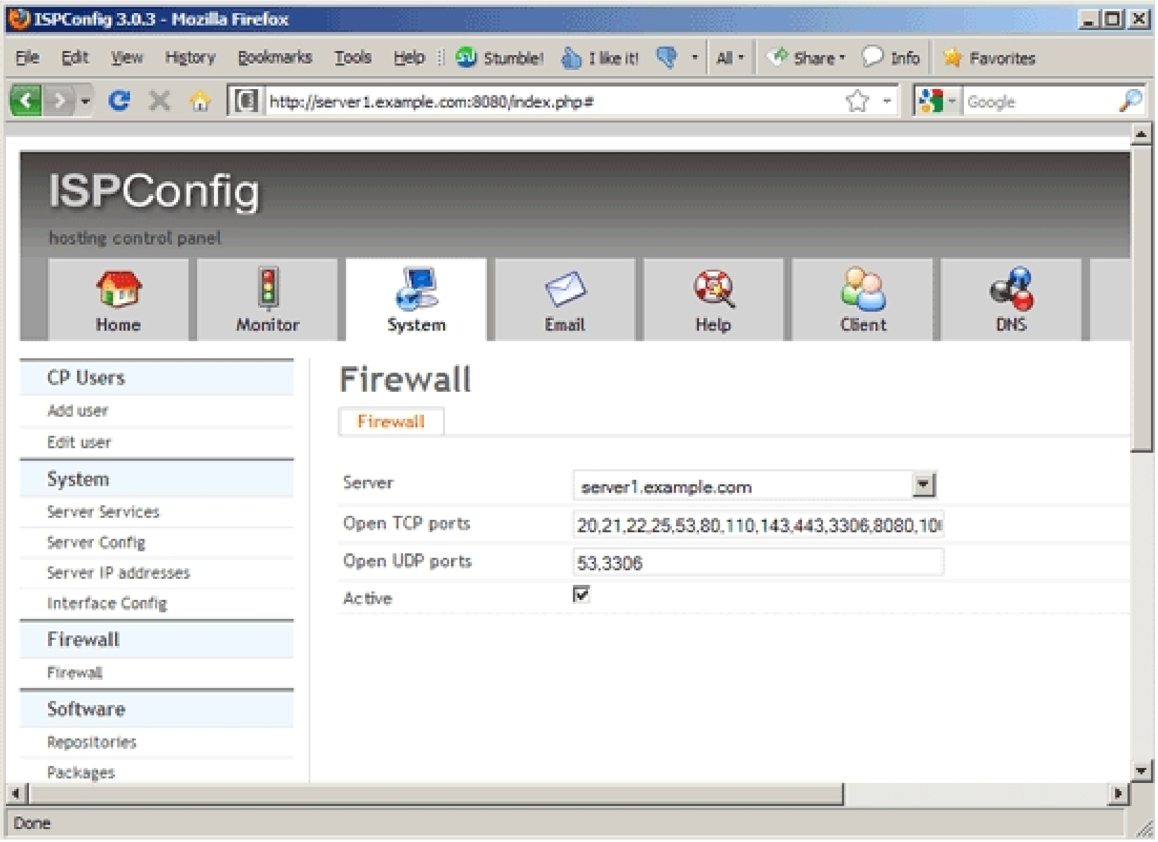
- *`Server`*: Select the server on which you want to enable the firewall.
- *`Open TCP ports`*: Specify the TCP ports that should be open in the firewall. Separate multiple ports by comma (no space), e.g. *`20,21,22,25,53,80,110,143,443,3306,8080,8081,10000`*. Specify port ranges with a colon, e.g. *`60:70`* or *`21,22,25,30:40,53,80`*. TCP ports that you don't list here will automatically be closed by the firewall. Common TCP ports are:

  - *20*: FTP
  - *21*: FTP
  - *22*: SSH
  - *25*: SMTP
  - *53*: DNS
  - *80*: HTTP
  - *110*: POP3
  - *143*: IMAP
  - *443*: HTTPS
  - *3306*: MySQL
  - *8080*: ISPConfig, HTTP-Proxies
  - *8081*: ISPConfig apps vhost
  - *10000*: Webmin

- *Open UDP ports*: Specify the UDP ports that should be open in the firewall. Separate multiple ports by comma (no space), e.g. *53,3306*. Specify port ranges with a colon, e.g. *60:70* or *21,22,25,30:40,53,80*. UDP ports that you don't list here will automatically be closed by the firewall. Common UDP ports are:

  - *53*: DNS
  - *3306*: MySQL

- *Active*: This defines whether the firewall is active or not.



Server	server1.example.com
Open TCP ports	20,21,22,25,53,80,110,143,443,3306,8080,10
Open UDP ports	53,3306
Active	☑

4.9.3 Interface

4.9.3.1 Main Config

Under `Main Config` you can configure the behaviour of the ISPConfig control panel itself.

You can find the following tabs here: `Sites`, `Mail`, `Domains`, `Misc`.

Sites

This tab allows you to configure a few settings for the `Sites` module.

The form has the following fields:

- `Database name prefix`: This defines the prefix that will be used for databases that you create under `Sites > Database > Database`. You can use the placeholders `[CLIENTID]` (which will be replaced with the ID of the client, e.g. `1` or `58`) and `[CLIENTNAME]` (which will be replaced with the client's username). Please note that database names must not be longer than 16 characters - MySQL doesn't support longer database names! Therefore it is stronlgy recommended to use `[CLIENTID]` here instead of `[CLIENTNAME]`. Examples: `c[CLIENTID]`, `[CLIENTID]` (MySQL database names can begin with a number).

- *Database user prefix*: This defines the prefix that will be used for database users of databases that you create under *Sites > Database > Database*. You can use the placeholders *[CLIENTID]* (which will be replaced with the ID of the client, e.g. *1* or *58*) and *[CLIENTNAME]* (which will be replaced with the client's username). You must not use underscores (_)! Example: *c[CLIENTID]*

- *FTP user prefix*: This defines the prefix that will be used for FTP users that you create under *Sites > FTP > FTP-User*. You can use the placeholders *[CLIENTID]* (which will be replaced with the ID of the client, e.g. *1* or *58*) and *[CLIENTNAME]* (which will be replaced with the client's username). Example: *[CLIENTNAME]*

- *Shell user prefix*: This defines the prefix that will be used for shell users that you create under *Sites > Shell > Shell-User*. You can use the placeholders *[CLIENTID]* (which will be replaced with the ID of the client, e.g. *1* or *58*) and *[CLIENTNAME]* (which will be replaced with the client's username). Example: *[CLIENTNAME]*

- *Webdav user prefix*: This defines the prefix that will be used for WebDAV users that you create under *Sites > Webdav> Webdav User*. You can use the placeholders *[CLIENTID]* (which will be replaced with the ID of the client, e.g. *1* or *58*) and *[CLIENTNAME]* (which will be replaced with the client's username). Example: *[CLIENTNAME]*

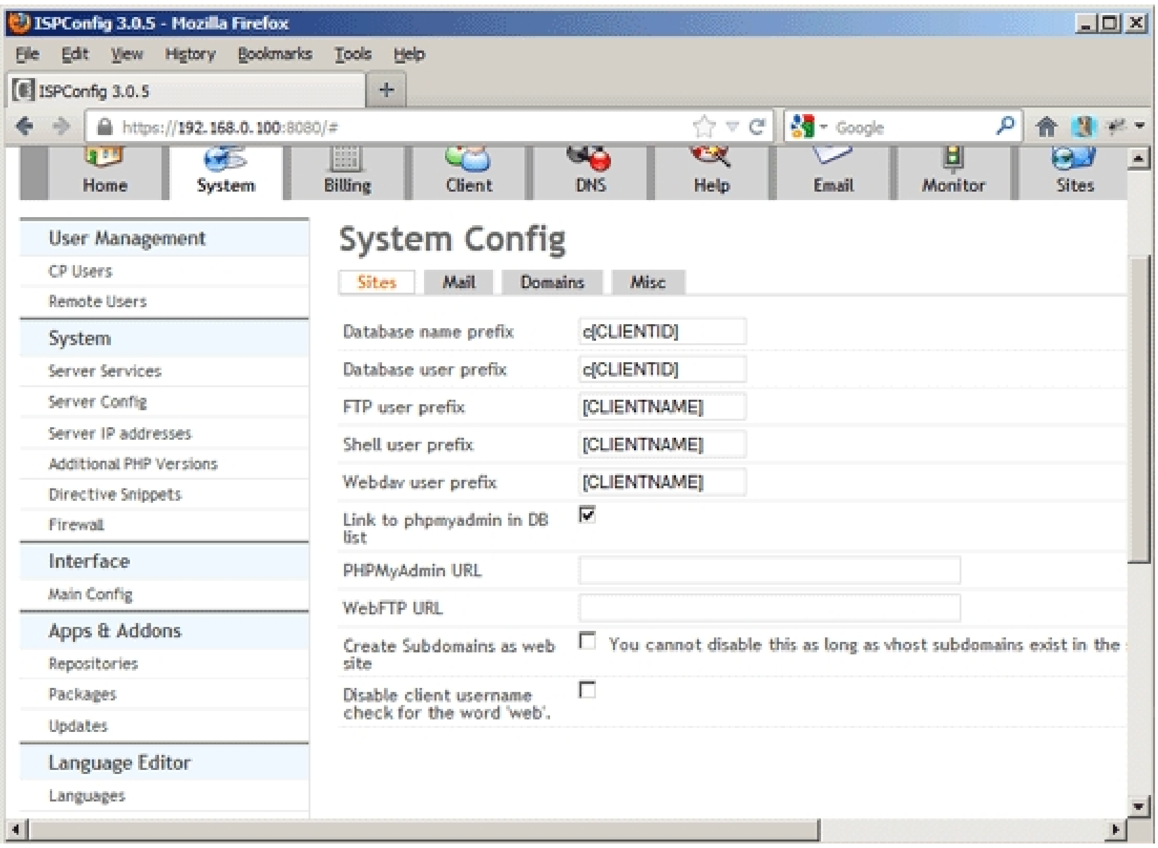
- *Link to phpmyadmin in DB list*: If you check this checkbox, an icon with a link to phpMyAdmin will be added to each database in the database list under *Sites > Database > Database*.

- *PHPMyAdmin URL*: If you have checked the *Link to phpmyadmin in DB list* checkbox, specify your phpMyAdmin URL here - otherwise an icon with a link to the default phpMyAdmin location will be displayed. This also means phpMyAdmin must already be installed somewhere on your server. Example: *http://www.example.com/phpmyadmin*

- *WebFTP URL*: If you specify your WebFTP URL here, a WebFTP icon with a link to your WebFTP application will be displayed in the FTP user list. This also means that a WebFTP application such as **net2ftp** must already be installed somewhere on your server. Example: *http://www.example.com/webftp*

- *Create Subdomains as web site*: If you enable this, it will be possible to create subdomains as Vhostsubdomains (see chapter ***4.6.1.3 Subdomain (Vhost)***). Please note that it is not possible to deactivate this feature as long as there are Vhostsubdomains in the system.

- *Disable client username check for the word 'web'*: Normally, ISPConfig does not allow client usernames to begin with the string web because this can lead to serious permissions issues (imagine you create a client with the username *web1* and then an SSH user *[CLIENTNAME]0* for that client which translates to *web10*; if you have a web site with the ID 10, this web site is owned by the user *web10* which means the SSH user *web10* can access/read/write/delete that web site although it might be owned by another client). You should therefore check this only if you know exactly what you are doing! It is stronly recommended to leave this box unchecked!



System Config

Database name prefix: c[CLIENTID]

Database user prefix: c[CLIENTID]

FTP user prefix: [CLIENTNAME]

Shell user prefix: [CLIENTNAME]

Webdav user prefix: [CLIENTNAME]

Link to phpmyadmin in DB list: ☑

PHPMyAdmin URL:

WebFTP URL:

Create Subdomains as web site: □ You cannot disable this as long as vhost subdomains exist in the

Disable client username check for the word 'web'.: □

Mail

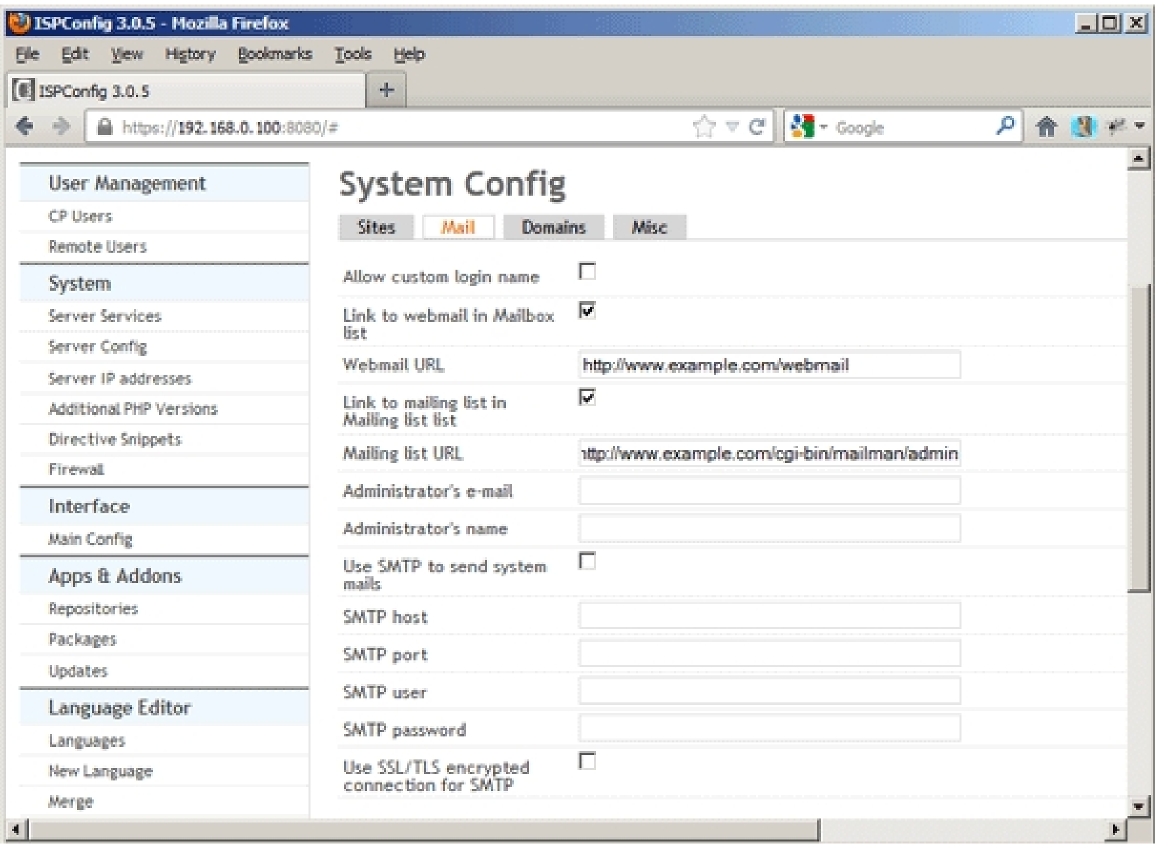
This tab allows you to configure a few settings for the `Email` module.

The form has the following fields:

- *`Allow custom login name`*: If checked, an additional username will have to be specified when an email account is created. This is helpful, for example, if you want to migrate email accounts from an ISPConfig 2 system to ISPConfig 3. By default, the email address is the username.
- *`Link to webmail in Mailbox list`*: If you check this checkbox, an icon with a link to your webmail application will be added to each mailbox in the mailbox list under *`Email > Email Accounts > Email Mailbox`*.
- *`Webmail URL`*: If you have checked the *`Link to webmail in Mailbox list`* checkbox, specify your webmail URL here - otherwise an icon with a link to the default webmail location will be displayed. This also means a webmail application must already be installed somewhere on your server. Example: *`http://www.example.com/webmail`*
- *`Link to mailing list in Mailing list list`*: If you check this checkbox, an icon with a link to your Mailman admin web interface will be added to each mailing list in the mailing list overview under *`Email > Mailing List > Mailing List`*.
- *`Mailing list URL`*: If you have checked the *`Link to mailing list in Mailing list list`*

checkbox, specify your Mailman admin web interface URL here - otherwise an icon with a link to the default Mailman admin web interface location will be displayed (*`/cgi-bin/mailman/admin/<listname>`*).

- *`Administrator's e-mail`*: When a new email account is created, a welcome message is sent to the new account. You can specify the sender address of that welcome email here.
- *`Administrator's name`*: Specify the name from which welcome emails are sent.
- *`Use SMTP to send system mails`*: Usually, ISPConfig uses PHP's *`mail()`* function to send mails from the local server. If you check this option, ISPConfig will use SMTP to send emails. This makes it possible to use remote servers for sending emails, for example.
- *`SMTP host`*: Specify the hostname of the server that will be used to send emails.
- *`SMTP port`*: Specify the port (e.g. *`25`*).
- *`SMTP user`*: Specify the username, if needed (leave it empty if no authentication is needed).
- *`SMTP password`*: Specify the password, if needed (leave it empty if no authentication is needed).
- *`Use SSL/TLS encrypted connection for SMTP`*: Check this to use an encrypted connection to the SMTP server (which must support SSL/TLS).



System Config

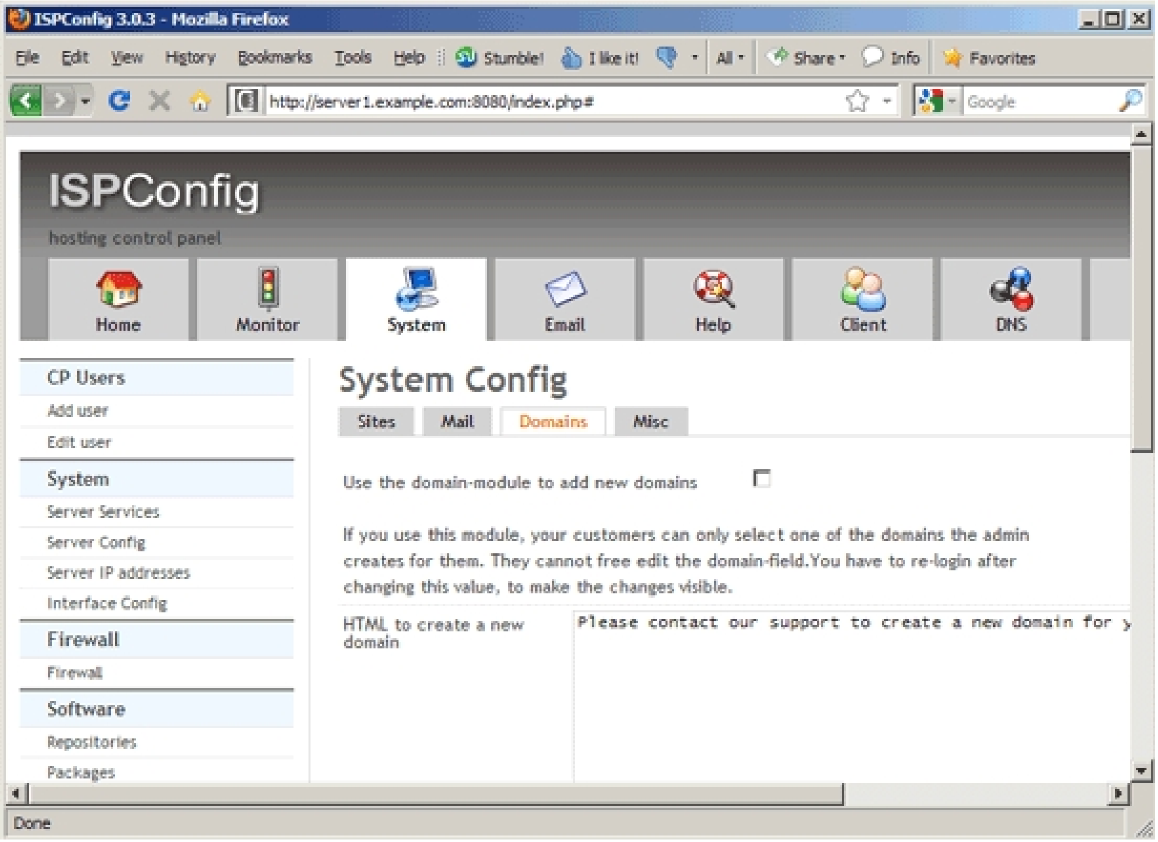
Allow custom login name	□
Link to webmail in Mailbox list	☑
Webmail URL	http://www.example.com/webmail
Link to mailing list in Mailing list list	☑
Mailing list URL	http://www.example.com/cgi-bin/mailman/admin
Administrator's e-mail	
Administrator's name	
Use SMTP to send system mails	□
SMTP host	
SMTP port	
SMTP user	
SMTP password	
Use SSL/TLS encrypted connection for SMTP	□

Domains

This tab is relevant only if you've enabled the `domain` module under `System > CP Users`. If you use this module, your customers can only select one of the domains the admin creates for them. They can not freely edit the domain field.

The form has the following fields:

- *`Use the domain module to add new domains`*: If you check this field (and the *`domain`* module is enabled), your customers can only select one of the domains that you create for them. They can not freely edit the domain field. You have to re-login after changing this value to make the changes visible.
- *`HTML to create a new domain`*: This text area can contain some HTML that will be shown to a customer if the domain module is enabled for the customer and he tries to create a new domain.



Use the domain-module to add new domains □

If you use this module, your customers can only select one of the domains the admin creates for them. They cannot free edit the domain-field.You have to re-login after changing this value, to make the changes visible.

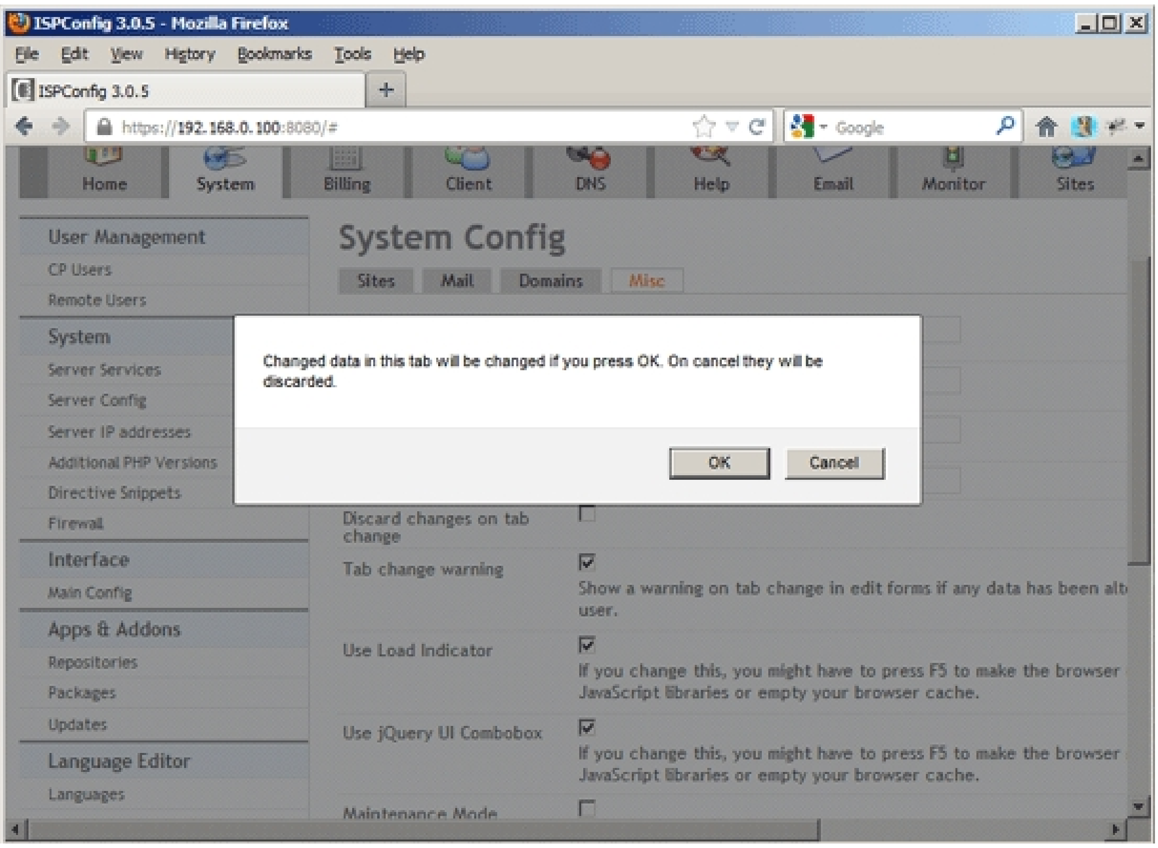
HTML to create a new domain: Please contact our support to create a new domain for y

Misc

You can configure some miscellaneous settings here.

The form has the following fields:

- `Dashboard atom feed URL (admin)`: If you want to display a certain news feed on the dashboard (Home) of the ISPConfig administrator, you can specify the URL of the Atom feed here (RSS feeds are not supported). By default, the latest ISPConfig news are displayed (***http://www.ispconfig.org/atom***).
- `Dashboard atom feed URL (reseller)`: If you want to display a certain news feed on the resellers' dashboard (Home), you can specify the URL of the Atom feed here (RSS feeds are not supported). By default, the latest ISPConfig news are displayed (***http://www.ispconfig.org/atom***).
- `Dashboard atom feed URL (client)`: If you want to display a certain news feed on the clients' dashboard (Home), you can specify the URL of the Atom feed here (RSS feeds are not supported). By default, the latest ISPConfig news are displayed (***http://www.ispconfig.org/atom***).
- `Monitor keyword`: If you use the ***ISPConfig Monitor App*** on your Android phone, you must specify a token in the App so that the App can connect to ISPConfig's Monitoring module (see chapter ***4.10 Monitor***). In the `Monitor keyword` field, you define this token; if you use a different token in your ISPConfig Monitor App, the App will not be allowed to fetch details from the Monitoring module.
- `Discard changes on tab change`: Usually, ISPConfig saves changes when you change the tab. If you check this checkbox, ISPConfig will not save changes when you change the tab.
- `Tab change warning`: Check this to make ISPConfig display a warning when you change the tab: `Changed data in this tab will be changed if you press OK. On cancel they will be discarded.` This allows you to decide whether changes should be saved or discarded for each tab change (regardless of what you selected under `Discard changes on tab change`).



System Config

Changed data in this tab will be changed if you press OK. On cancel they will be discarded.

Discard changes on tab change □

Tab change warning ☑
Show a warning on tab change in edit forms if any data has been alt
user.

Use Load Indicator ☑
If you change this, you might have to press F5 to make the browser
JavaScript libraries or empty your browser cache.

Use jQuery UI Combobox ☑
If you change this, you might have to press F5 to make the browser
JavaScript libraries or empty your browser cache.

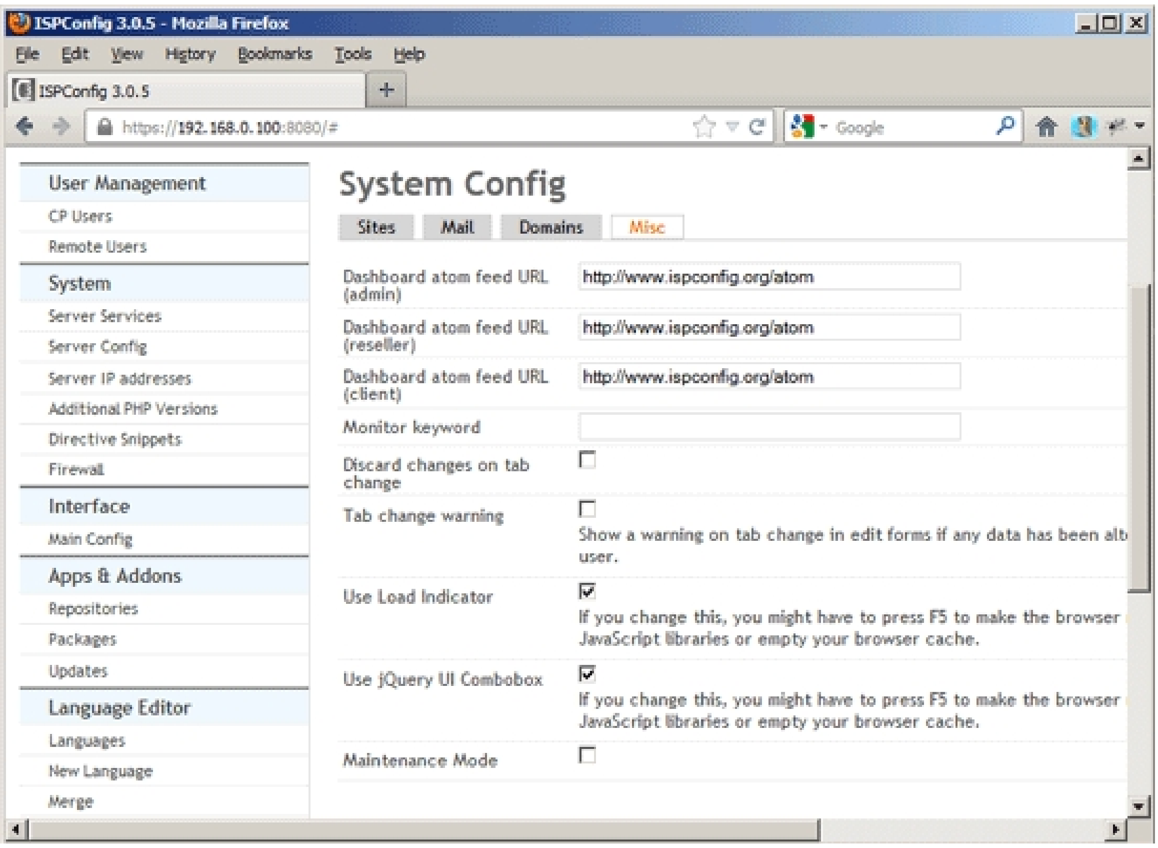
Maintenance Mode □

- *Use Load Indicator*: Uncheck this to not use the JavaScript load indicator when ISPconfig loads a page. This might make ISPConfig load pages a bit faster.
- *Use jQuery UI Combobox*: By default, ISPConfig uses a JavaScript combobox for drop-down menus (in list views and forms) that allows you to type something in the box so that the box can make suggestions. Uncheck this to use normal drop-down menus.
- *Maintenance Mode*: Check this to put ISPConfig into maintencnace mode and log out all currently logged-in users (except the administrator). It is strongly recommended to use *Maintencnae Mode* when you upgrade ISPConfig so that noone can make any changes in ISPConfig while the system is being upgraded. Uncheck this checkbox when the upgrade is finished.

About the ISPConfig Monitor App:

The ISPConfig Monitor App is for all servers, not only for servers running ISPConfig. With the ISPConfig Monitor App, you can check your server status and find out if all services are running as expected. You can check TCP and UDP ports and ping your servers. In addition to that you can use this app to request details from servers that have ISPConfig installed; these details include everything you know from the Monitoring module in the ISPConfig Control Panel (e.g. services, mail and system logs, mail queue, CPU and memory info, disk usage, quota, OS details, RKHunter log, etc.), and of course, as ISPConfig is multiserver-capable, you can check all servers that are controlled from your ISPConfig master server.

You can find download instructions on ***http://www.ispconfig.org/ispconfig-3/ispconfig-monitor-app-for-android/*** or use this QR code to install the ISPConfig Monitor App for Android (to read this code, you must have a barcode scanner app installed, e.g. like the free ***Barcode Scanner***):



System Config

Dashboard atom feed URL (admin)	http://www.ispconfig.org/atom
Dashboard atom feed URL (reseller)	http://www.ispconfig.org/atom
Dashboard atom feed URL (client)	http://www.ispconfig.org/atom
Monitor keyword	
Discard changes on tab change	□
Tab change warning	□ Show a warning on tab change in edit forms if any data has been alt user.
Use Load Indicator	☑ If you change this, you might have to press F5 to make the browser JavaScript libraries or empty your browser cache.
Use jQuery UI Combobox	☑ If you change this, you might have to press F5 to make the browser JavaScript libraries or empty your browser cache.
Maintenance Mode	□

4.9.4 Apps & Addons

Under *Software* you can define ISPConfig application repositories, install ISPConfig application packages (such as phpMyAdmin) and install updates of such packages, if available.

4.9.4.1 Repositories

Here you can add ISPConfig application repositories to your system.

To create a new repository, click on the *Add new record* button. This will lead you to the *Software Repository* form with the tab *Repository*.

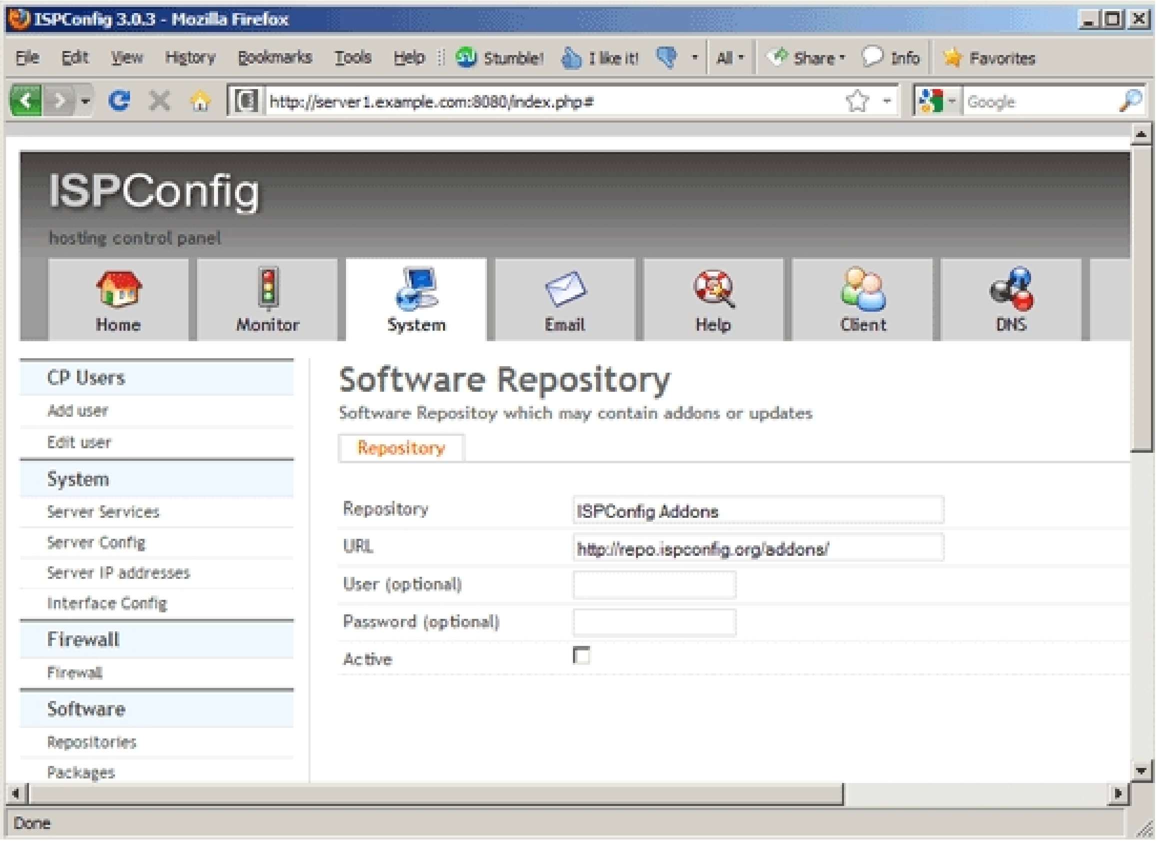
Software Repository

Repository

The form has the following fields:

- *Repository*: Type in a name for the repository, e.g. *ISPConfig Addons*.

- *`URL`*: Specify the URL of the repository. Example: *`http://repo.ispconfig.org/addons/`*
- *`User (optional)`*: If the whole repository or single packages of the repository (e.g. packages that need testing and should be available only to developers) are password-protected, type in the repository username here. Leave the field empty if the repository isn't password-protected.
- *`Password (optional)`*: If the whole repository or single packages of the repository (e.g. packages that need testing and should be available only to developers) are password-protected, type in the repository password here. Leave the field empty if the repository isn't password-protected.
- *`Active`*: This defines whether the repository is active or not.



Software Repository

Software Repositoy which may contain addons or updates

Repository	ISPConfig Addons
URL	http://repo.ispconfig.org/addons/
User (optional)	
Password (optional)	
Active	□

4.9.4.2 Packages

Here you can find a list of available packages from the active repositories. For each server that is controlled by ISPConfig, you can see if the package is already installed (it then reads *`Installed version ...`*) or if it can be installed (it reads *`Install now`*). To install a package, simply click on the *`Install now`* link. Installation can take two or three minutes; reload the page to see if it has been installed successfully (it should then read *`Installed version ...`*).

4.9.4.3 Updates

Under *Updates* you can find a list of all installed packages for which updates are available (select the server under *Select server* first). You can install the updates from here by clicking on the *Install update* link.

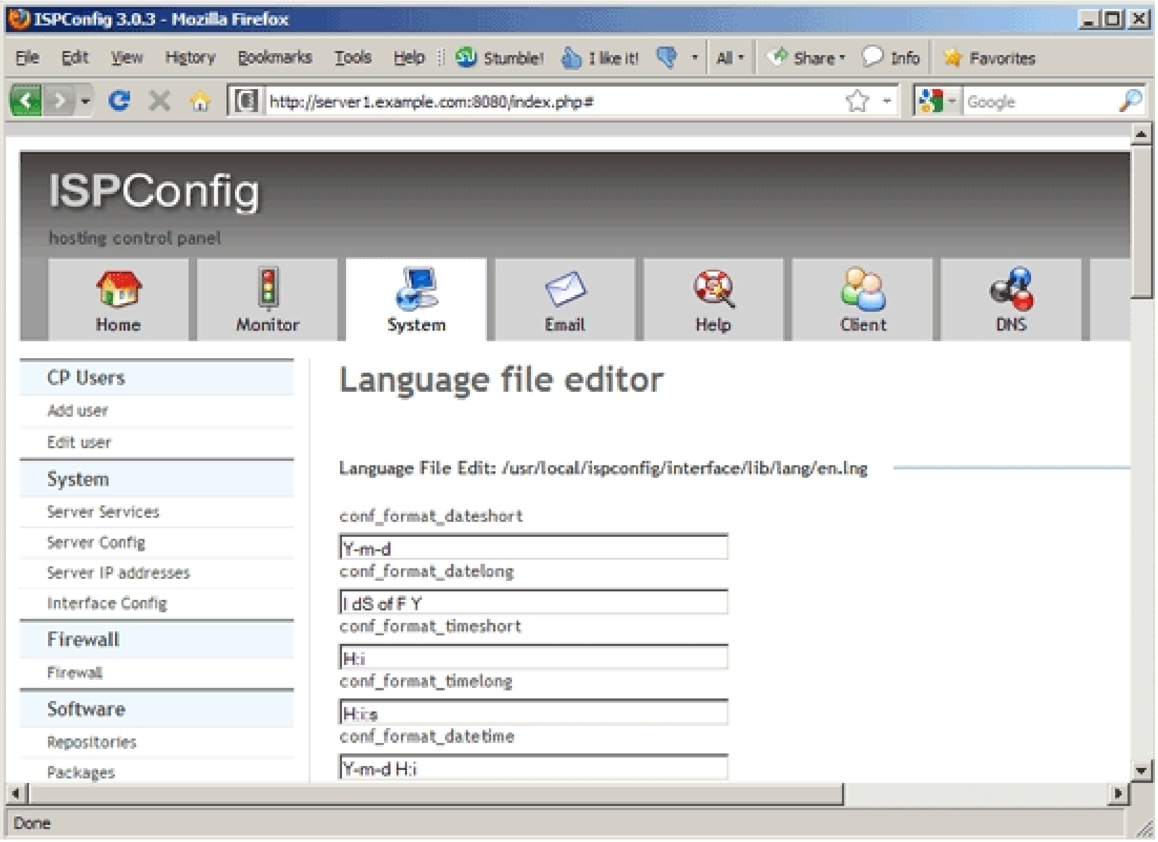
4.9.5 Language Editor

The *Language Editor* allows you to add new ISPConfig translations or modify existing ones. For example, if the ISPConfig interface isn't available in your language, you can create a translation here.

4.9.5.1 Languages

Here you can find the *Language file editor* which allows you to modify all existing translations. Select the translation that you want to modify in the *Select language* drop-down menu; this will bring up a list of all available language files (extension *.lng*) for that language, together with a note to which ISPConfig module the language file belongs and the last modification date. Click on the file that you want to modify - this will bring you to a form with all strings that can be translated. Make your modifications and click on the *Save* button afterwards.

Hint: You can change the text of the welcome email that is sent to new email accounts under *mail > en_mail_user.lng* (the fields are *welcome_mail_subject* and *welcome_mail_message*).



Language file editor

Language File Edit: /usr/local/ispconfig/interface/lib/lang/en.lng

conf_format_dateshort: Y-m-d

conf_format_datelong: l dS of F Y

conf_format_timeshort: H:i

conf_format_timelong: H:i:s

conf_format_datetime: Y-m-d H:i

4.9.5.2 New Language

If you want to add a new translation (for example, because ISPConfig isn't available in your language), you can do this here. The `Add new language` form has the following fields:

- `Select language basis`: Select one of the existing translations here, e.g. `en`. Your new language files will use this existing translation first so that you have a basis to start with, and you can then use the `Language file editor` under `Languages` to translate the strings to the new language.
- `New language`: Type in the two characters ISO 639-1 language code (see **http://en.wikipedia.org/wiki/List_of_ISO_639-1_codes**) of your new translation.

After you have created the new language, you can use the `Language file editor` under `Languages` to translate the strings to the new language.

4.9.5.3 Merge

The `Merge` function adds missing strings and even missing language files from the English master language files to the selected language. This is useful for the following two scenarios:

- You've created your own language in an old ISPConfig version, and now you update ISPConfig, and the new ISPConfig version has a lot of new functions that are missing in your language files. You can use the `Merge` function to merge the new/missing translations into your language files, and then you can use the `Language file editor` under `Languages` to translate the strings to the new language.
- The second scenario is for the ISPConfig developers only. A lot of translations were contributed by ISPConfig users, but of course the developers don't speak all these languages. If the developers add new functions, they add the English translations and merge these English translations into all the other supported languages (so that a native speaker and ISPConfig contributor can translate them using the `Language file editor`).

To merge new English strings into a translation, just select the language in the `Select language` drop-down menu and click on the `Merge files now` button.

4.9.5.4 Export

Here you can export existing translations. Just select the language that you want to export and click on the `Export the selected language file set` button. This will display a link to the exported file (e.g. `Exported language file to: /temp/en.lng`); click on that link, and the exported file will be displayed in a new browser window (from where you can save it on your computer).

You must not use the `Export` function to manually edit exported translations in a text editor - always use

ISPConfig's *Language file editor* for that! The *Export* funtion is useful if

- you've created a translation on one ISPConfig installation and want to use the same translation on another ISPConfig installation (where you can use the *Import* function to import that translation).
- you've created a translation and want to send it to the ISPConfig developers (*dev@ispconfig.org*).

4.9.5.5 Import

You can use the *Import* function to import translations that you've previously exported on another ISPConfig server. Please not that you must not import language files that have been manually modified in a text editor - always use ISPConfig's *Language file editor* to modify translations!

The *Import language file* form has the following fields:

- *Select language file*: Select the language file to import from your local computer. ISPConfig will automatically detect the language from the contents of the selected file.
- *Overwrite file, if exists*: Check this if you want to overwrite any existing files of this translation on the ISPConfig server.
- *Skip ISPConfig version check*: Usually ISPConfig performs a version check to find out if the translation that is to be imported matches the version number of the ISPConfig installation, and displays an error mesage if the versions don't match (i.e., ISPConfig refuses to import the translation). By checking this checkbox you can skip this version check.

4.9.6 Remote Actions

Here you can initiate operating system updates and ISPConfig updates on all servers controlled by ISPConfig.

4.9.6.1 Do OS-Update

The *Do OS-Update* function allows you to start an operating system update on the selected server, i.e., the latest updates will be installed. Please note that this function supports only Debian and Ubuntu. It will perform

```
aptitude -y upgrade
```

on the selected server. This works also on remote servers that are controlled by this ISPConfig installation. To update all servers controlled by ISPConfig, select *All servers*.

As this is an unattended update and you don't see what packages are updated, you should use this function at your own risk. At this point, it is strongly recommended to run your updates manually on the command line!

4.9.6.2 Do ISPConfig-Update

The *`Do ISPConfig-Update`* function allows you to update ISPConfig on the selected server. This works also on remote servers that are controlled by this ISPConfig installation. To update all servers controlled by ISPConfig, select *`All servers`*.

This funtion is experimental! At this point, it is strongly recommended to run your updates manually on the command line!

4.10 Monitor

The *`Monitor`* module allows you to take a look at the logs, CPU, memory, disk usage, etc. of all servers controlled by ISPConfig. Under *`System State (All Servers)`* you can find information about all servers controlled by ISPConfig, whereas the details in the other menu items refer to just one (the selected) server.

4.10.1 System State (All Servers)

Here you can find details about all servers that are controlled by this ISPconfig installation.

4.10.1.1 Show Overview

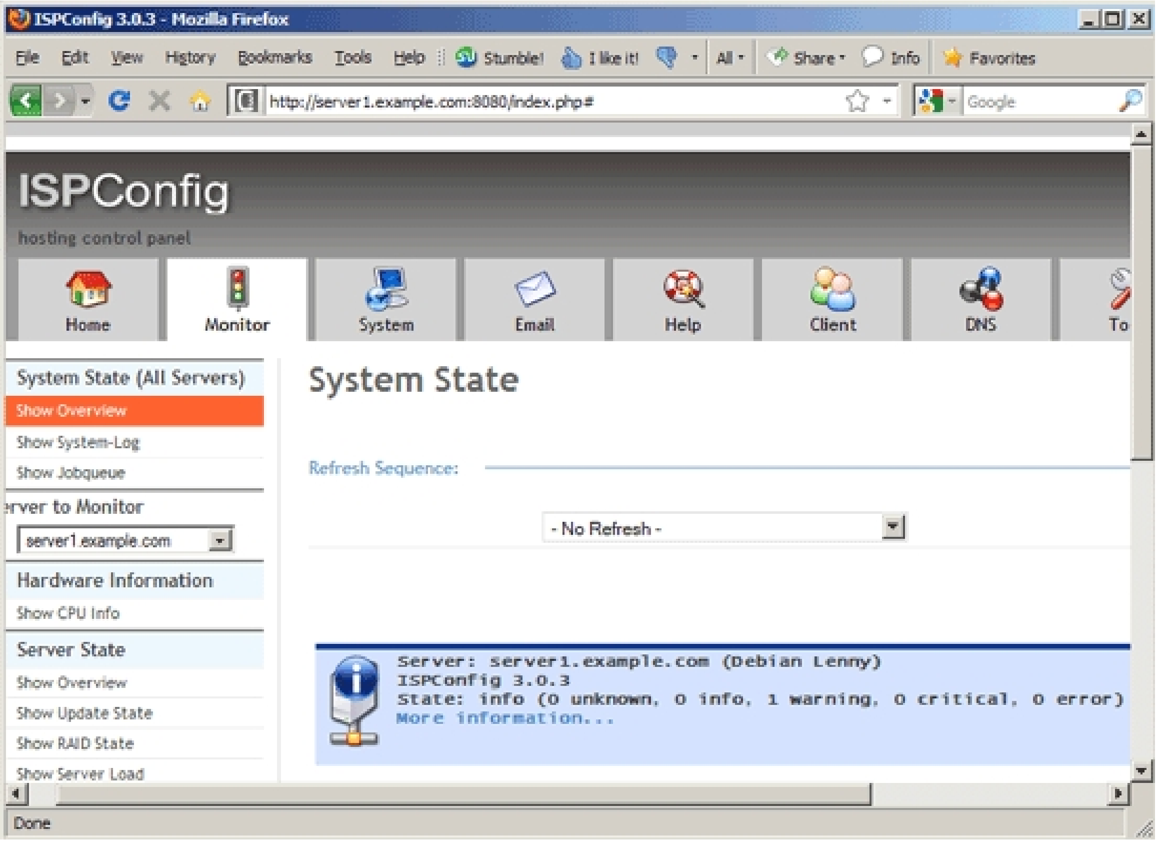
Here you can find an overview of all your servers that are controlled by ISPConfig. Details that are displayed here are the general state of the server (if there have been warnings, errors, etc.), the state of the hard drive space, mail queue, server load, if all services are online, and if updates are available (some of these details will only be displayed if you click on the *`More information...`* link).

Under *`Refresh Sequence`* you can select if the information should be refreshed automatically while you are on this page (by default it is not refreshed), and in which interval.

By clicking on the *`[More...]`* link that is displayed next to each status, you can find out more details about that item - the same details can be accessed under *`Server State`* in the menu (but then make sure you select the correct server under *`Server to Monitor`*).

Each server's overview is displayed with one of the background colours green, orange, or red:

- Green: everything is ok - no warnings or errors, no updates are available, all services are online, etc.
- Blue: there are warnings in your logs, or updates are available, but there's nothing system-critical on the server.
- Red: this marks some kind of failure, e.g. errors in the logs, needed services aren't running, a script failed to execute, etc. This is system-critical, and immediate action should be taken by your side (e.g. log onto your server's shell and check the logs in the *`/var/log/`* directory).



```
Server: server1.example.com (Debian Lenny)
ISPConfig 3.0.3
State: info (0 unknown, 0 info, 1 warning, 0 critical, 0 error)
More information...
```

4.10.1.2 Show System-Log

Here you can take a look at the ISPConfig log - this log shows what ISPConfig does in the background, and if there have been warnings or errors. This log is for all servers controlled by ISPConfig (you can use the filter to display log entries from a specific server); what is getting logged depends on the log level that you set for each server on the *Server* tab under *System > System > Server Config* (*Debug*, *Warnings*, or *Errors*).

4.10.1.3 Show Jobqueue

Here you can find a list of background tasks that ISPConfig has to carry out on the nodes that are controlled by ISPConfig. If the list is empty, ISPConfig has completed all tasks.

4.10.2 Server to Monitor

This refers to all following menu items, i.e., the following menu items will display information about the server that you select here.

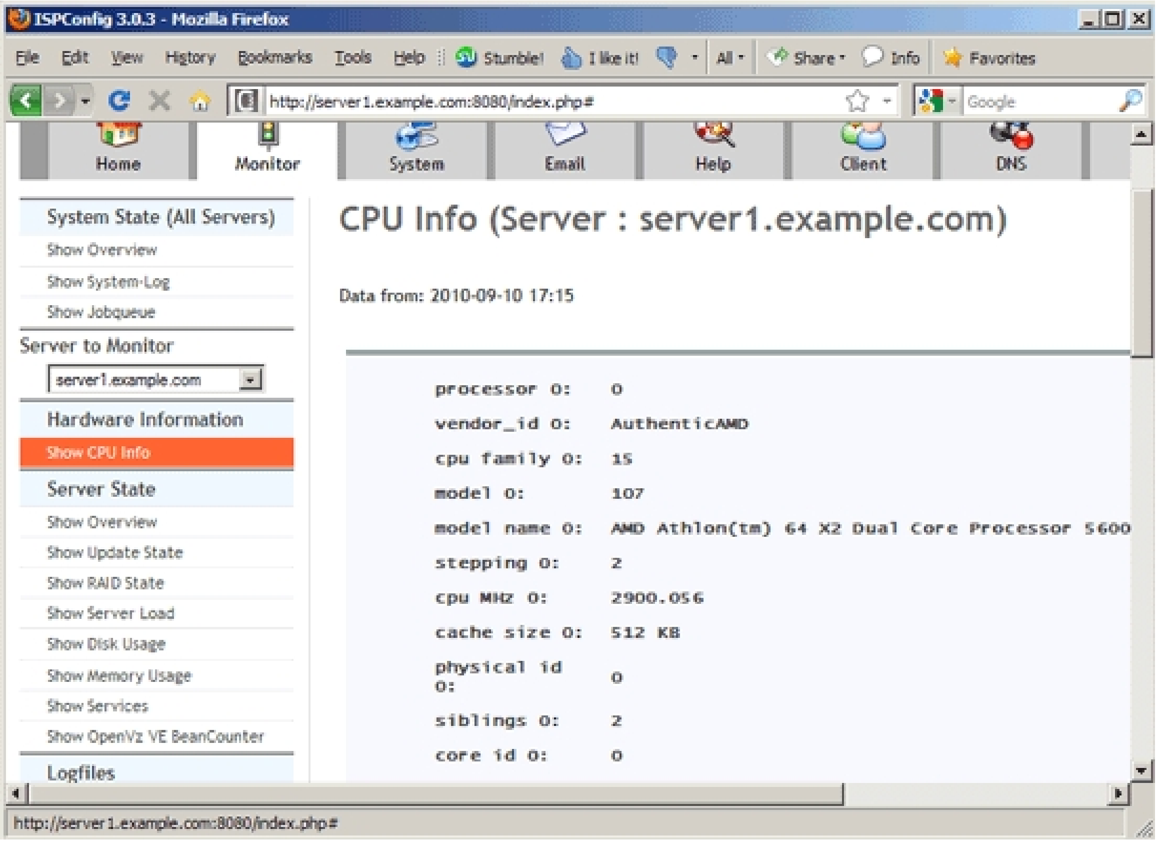
4.10.3 Hardware Information

4.10.3.1 Show CPU Info

You can find details about the CPU of the selected server here. This is the same as if you run

```
cat /proc/cpuinfo
```

on the server.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
Home Monitor System Email Help Client DNS
System State (All Servers)
Show Overview
Show System-Log
Show Jobqueue
Server to Monitor
server1.example.com
Hardware Information
Show CPU Info
Server State
Show Overview
Show Update State
Show RAID State
Show Server Load
Show Disk Usage
Show Memory Usage
Show Services
Show OpenVz VE BeanCounter
Logfiles
http://server1.example.com:8080/index.php#

CPU Info (Server : server1.example.com)

Data from: 2010-09-10 17:15

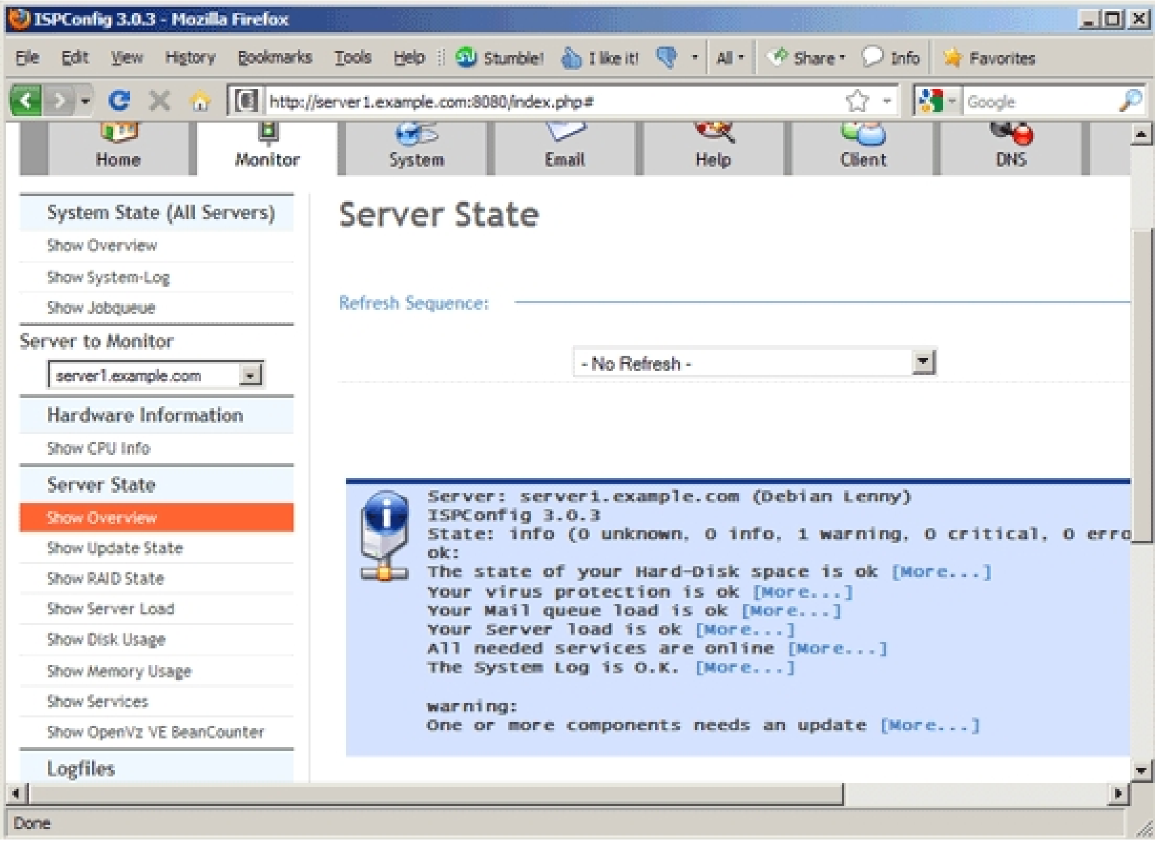
```
processor 0:      0
vendor_id 0:      AuthenticAMD
cpu family 0:     15
model 0:          107
model name 0:     AMD Athlon(tm) 64 X2 Dual Core Processor 5600
stepping 0:       2
cpu MHz 0:        2900.056
cache size 0:     512 KB
physical id
0:                0
siblings 0:       2
core id 0:        0
```

4.10.4 Server State

4.10.4.1 Show Overview

Here you can find the same details as under `Monitor > System State (All Servers) > Show Overview`, except that the details here refer to just one server (the one you selected under `Monitor > Server to Monitor`).

Under `Refresh Sequence` you can select if the information should be refreshed automatically while you are on this page (by default it is not refreshed), and in which interval.



Server State

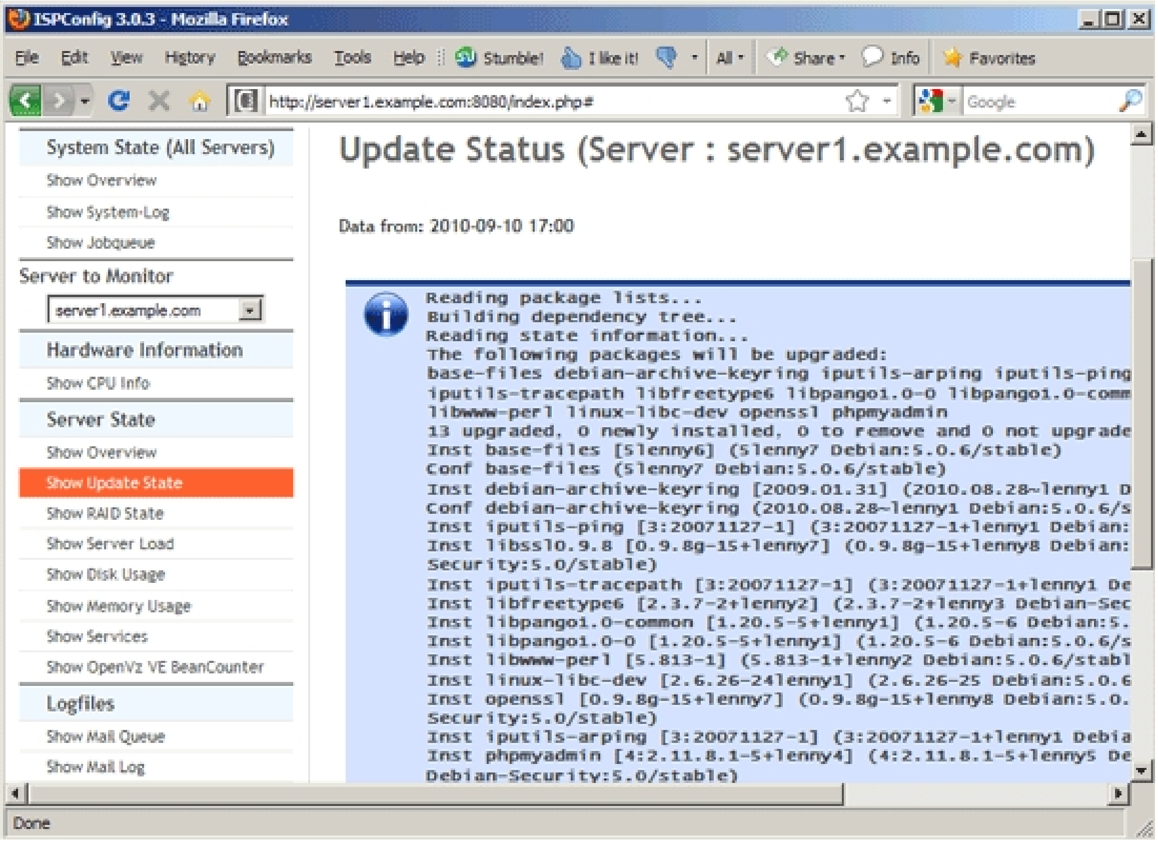
Refresh Sequence: - No Refresh -

```
Server: server1.example.com (Debian Lenny)
ISPConfig 3.0.3
State: info (0 unknown, 0 info, 1 warning, 0 critical, 0 erro
ok:
The state of your Hard-Disk space is ok [More...]
Your virus protection is ok [More...]
Your Mail queue load is ok [More...]
Your Server load is ok [More...]
All needed services are online [More...]
The System Log is O.K. [More...]

warning:
One or more components needs an update [More...]
```

4.10.4.2 Show Update State

This page displays if update packages are available for the operating system and the installed packages. If there are, you should bring your server up to date.

If you see the warning `WARNING: Your ClamAV installation is OUTDATED!` - this sounds more dramatic than it actually is, and it is usually not necessary to take any action. This just means that a newer ClamAV version is available than the one that is installed - your current version is still ok. It does not mean that the virus signature database is not up to date - it actually is, and protection is still guaranteed. You can check if your distribution offers an updated ClamAV package - if it does, you can install it, but if it doesn't, you should avoid installing ClamAV from the sources - wait until your distribution provides an updated package.



Update Status (Server : server1.example.com)

Data from: 2010-09-10 17:00

```
Reading package lists...
Building dependency tree...
Reading state information...
The following packages will be upgraded:
base-files debian-archive-keyring iputils-arping iputils-ping
iputils-tracepath libfreetype6 libpango1.0-0 libpango1.0-comm
libwww-perl linux-libc-dev openssl phpmyadmin
13 upgraded, 0 newly installed, 0 to remove and 0 not upgrade
Inst base-files [5lenny6] (5lenny7 Debian:5.0.6/stable)
Conf base-files (5lenny7 Debian:5.0.6/stable)
Inst debian-archive-keyring [2009.01.31] (2010.08.28~lenny1 D
Conf debian-archive-keyring (2010.08.28~lenny1 Debian:5.0.6/s
Inst iputils-ping [3:20071127-1] (3:20071127-1+lenny1 Debian:
Inst libssl0.9.8 [0.9.8g-15+lenny7] (0.9.8g-15+lenny8 Debian:
Security:5.0/stable)
Inst iputils-tracepath [3:20071127-1] (3:20071127-1+lenny1 De
Inst libfreetype6 [2.3.7-2+lenny2] (2.3.7-2+lenny3 Debian-Sec
Inst libpango1.0-common [1.20.5-5+lenny1] (1.20.5-6 Debian:5.
Inst libpango1.0-0 [1.20.5-5+lenny1] (1.20.5-6 Debian:5.0.6/s
Inst libwww-perl [5.813-1] (5.813-1+lenny2 Debian:5.0.6/stabl
Inst linux-libc-dev [2.6.26-24lenny1] (2.6.26-25 Debian:5.0.6
Inst openssl [0.9.8g-15+lenny7] (0.9.8g-15+lenny8 Debian:5.0.
Security:5.0/stable)
Inst iputils-arping [3:20071127-1] (3:20071127-1+lenny1 Debia
Inst phpmyadmin [4:2.11.8.1-5+lenny4] (4:2.11.8.1-5+lenny5 De
Debian-Security:5.0/stable)
```

It is recommended to do this manually with your distribution's package manager, e.g. apt/aptitude on Debian/Ubuntu, yum on Fedora/CentOS, and yast/zypper on OpenSUSE.

Debian/Ubuntu:

```
aptitude update
```

```
aptitude safe-upgrade
```

Fedora/CentOS:

```
yum update
```

OpenSUSE:

```
zypper refresh
zypper update
```

If you are on Debian/Ubuntu, you could also go to *System > Remote Actions > Do OS-Update*, but this method is not recommended!

4.10.4.3 Show RAID State

If the selected server uses RAID, you can find details about the RAID arrays here. Basically, these are the same details that the command

```
cat /proc/mdstat
```

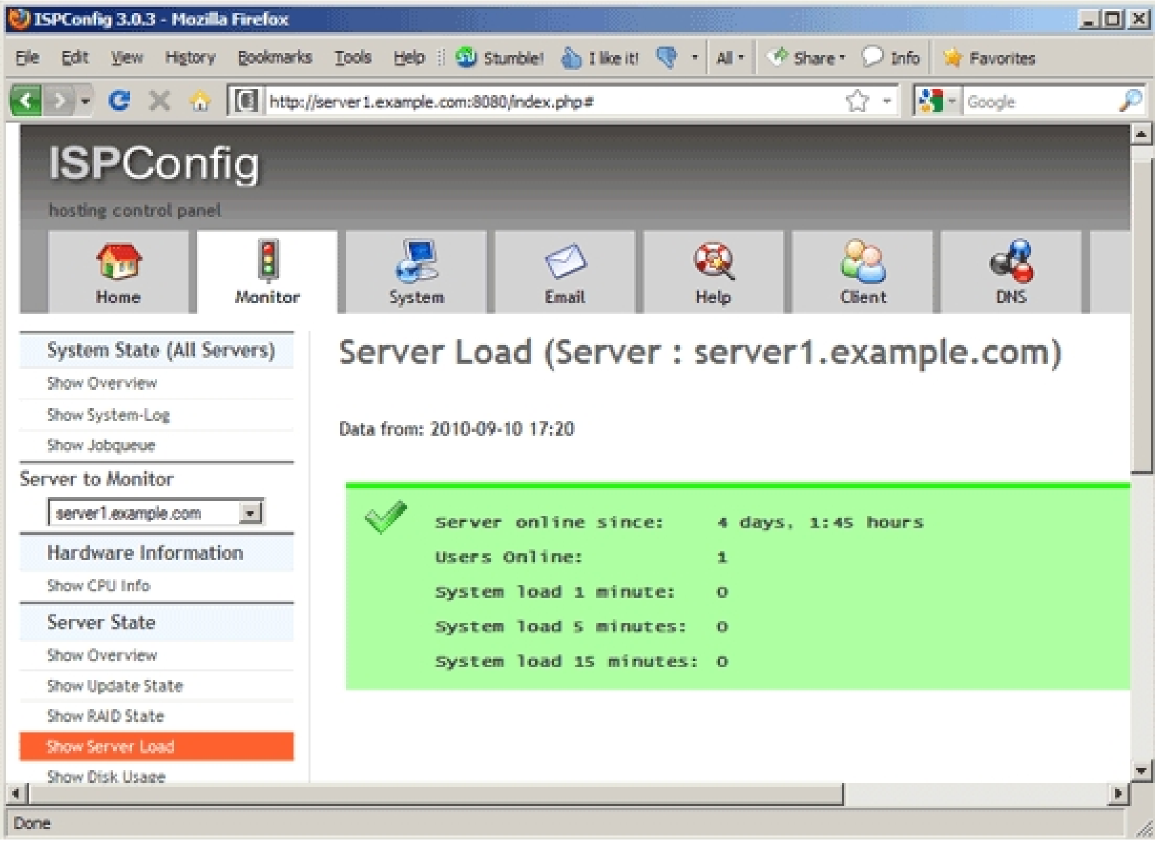
would show.

4.10.4.4 Show Server Load

Here you can find details about the server load. Basically, these are the same details that the command

```
uptime
```

would show.



Server Load (Server : server1.example.com)

Data from: 2010-09-10 17:20

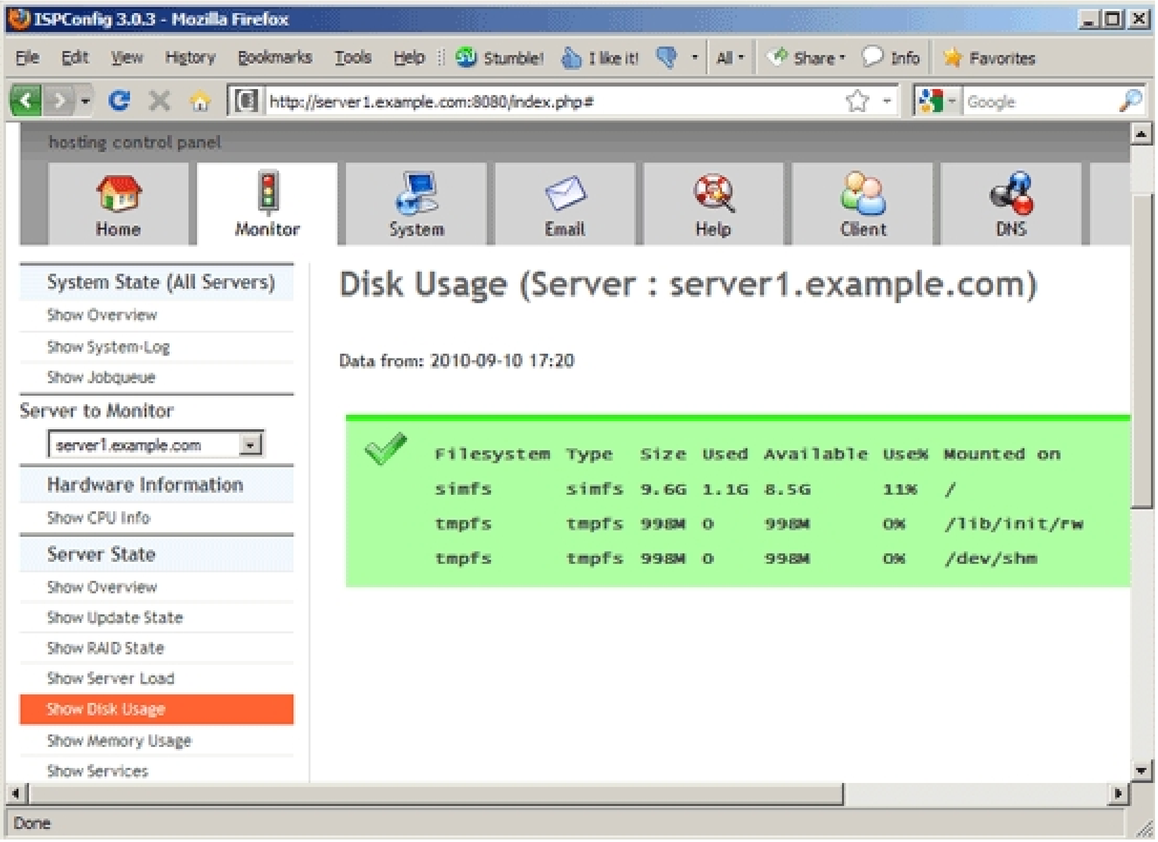
```
Server online since:     4 days, 1:45 hours
Users Online:            1
System load 1 minute:    0
System load 5 minutes:   0
System load 15 minutes:  0
```

4.10.4.5 Show Disk Usage

Here you can find details about the server's disk usage. Basically, these are the same details that the command

```
df -h
```

would show.



Disk Usage (Server : server1.example.com)

Data from: 2010-09-10 17:20

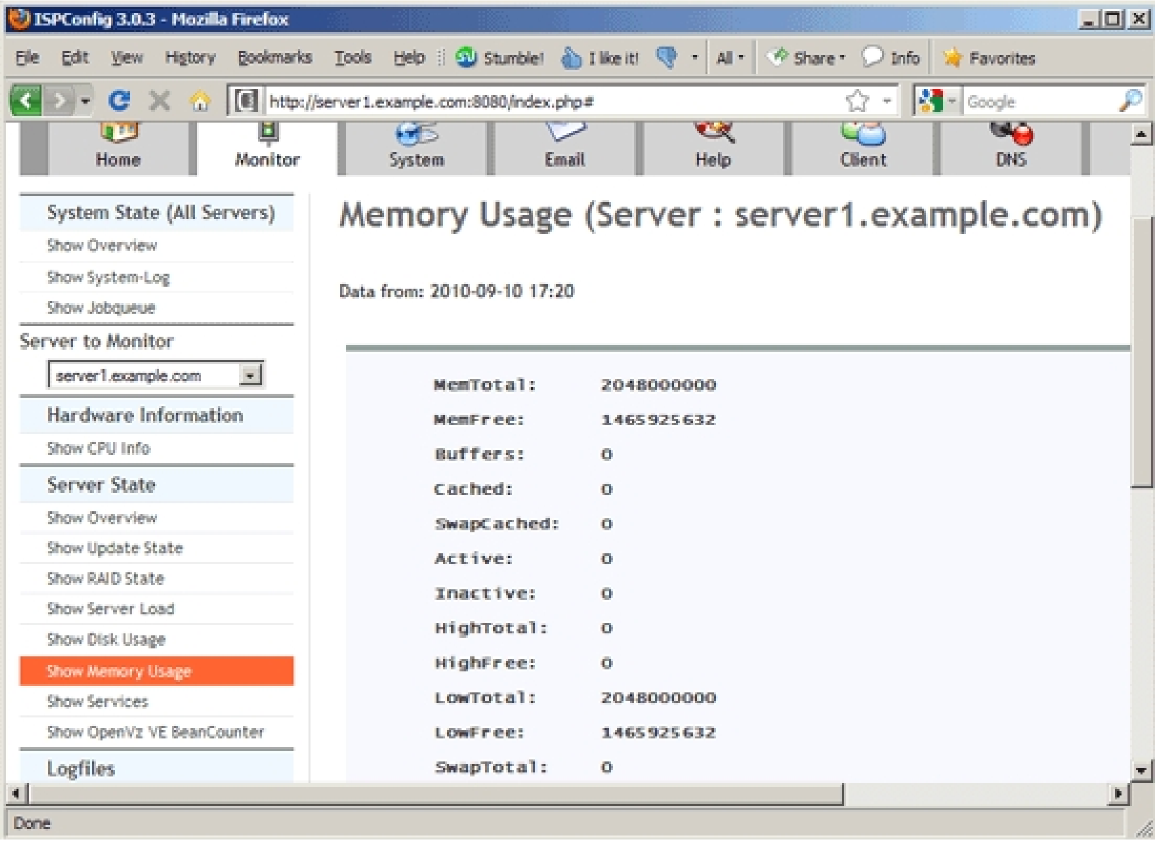
```
Filesystem  Type   Size  Used  Available  Use%  Mounted on
simfs       simfs  9.6G  1.1G  8.5G       11%   /
tmpfs       tmpfs  998M  0     998M       0%    /lib/init/rw
tmpfs       tmpfs  998M  0     998M       0%    /dev/shm
```

4.10.4.6 Show Memory Usage

Here you can find details about the server's memory usage. Basically, these are the same details that the command

```
cat /proc/meminfo
```

would show.



Memory Usage (Server : server1.example.com)

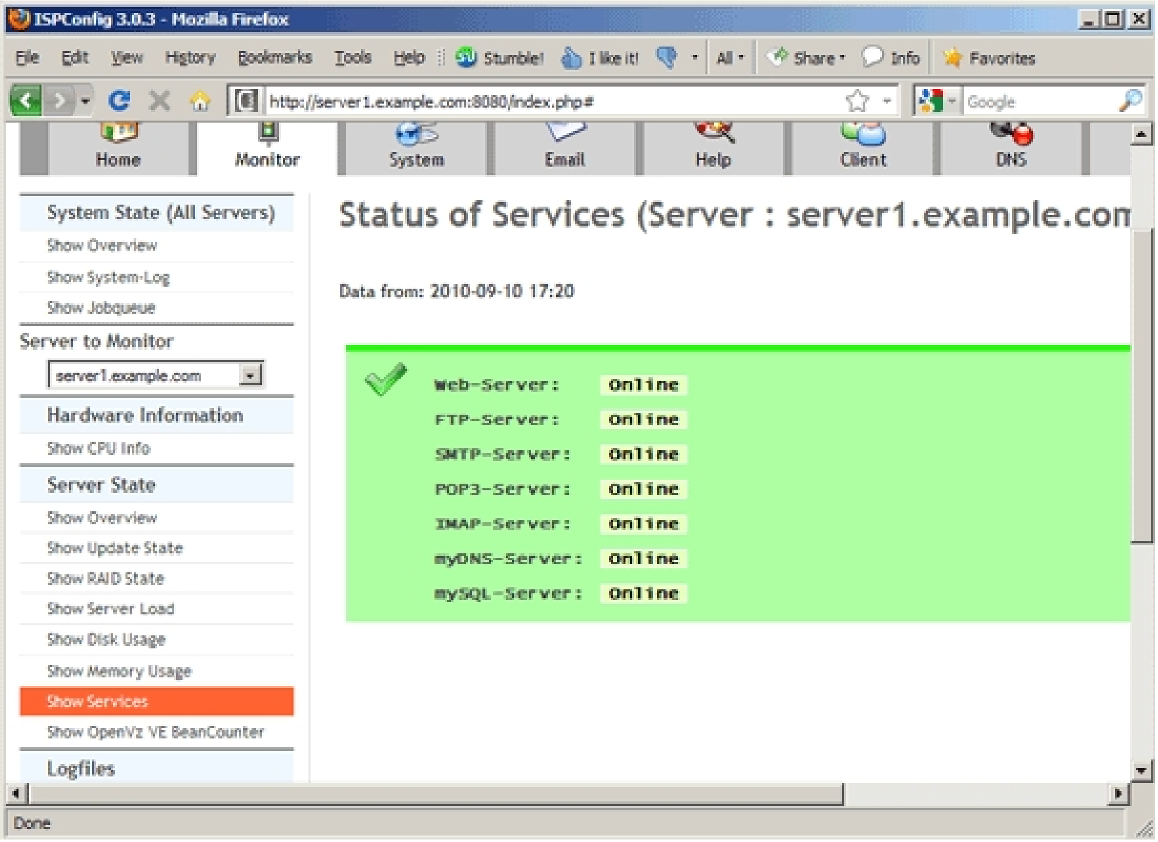
Data from: 2010-09-10 17:20

```
MemTotal:     2048000000
MemFree:      1465925632
Buffers:      0
Cached:       0
SwapCached:   0
Active:       0
Inactive:     0
HighTotal:    0
HighFree:     0
LowTotal:     2048000000
LowFree:      1465925632
SwapTotal:    0
```

4.10.4.7 Show Services

Under this menu item you can find information if the following services are running or not:

- Web-Server
- FTP-Server
- SMTP-Server
- POP3-Server
- IMAP-Server
- myDNS-Server (this refers to your DNS server in general, no matter if you use MyDNS, BIND, or PowerDNS)
- mySQL-Server



Status of Services (Server : server1.example.com

Data from: 2010-09-10 17:20

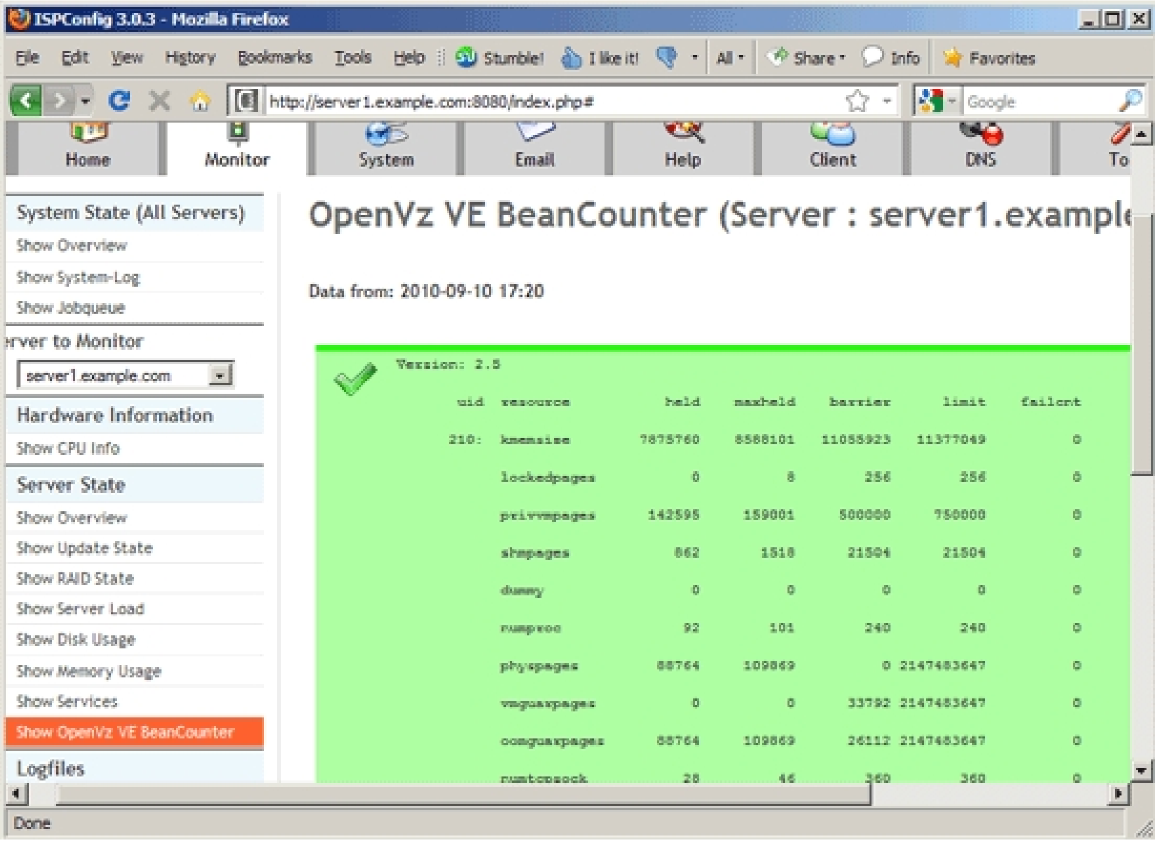
```
Web-Server:     Online
FTP-Server:     Online
SMTP-Server:    Online
POP3-Server:    Online
IMAP-Server:    Online
myDNS-Server:   Online
mySQL-Server:   Online
```

4.10.4.8 Show OpenVz VE BeanCounter

If the selected server is an OpenVZ container (virtual machine), you can find details about the OpenVZ beancounter here (it displays details about the allocated resources and limits of the virtual machine). Basically, these are the same details that the command

```
cat /proc/user_beancounters
```

would show.



OpenVz VE BeanCounter (Server : server1.example

Data from: 2010-09-10 17:20

```
Version: 2.5
       uid  resource           held    maxheld    barrier      limit    failcnt
      210:  kmemsize        7875760    8588101   11055923   11377049          0
            lockedpages           0          8        256        256          0
            privvmpages      142595     159001     500000     750000          0
            shmpages            862       1518      21504      21504          0
            dummy                 0          0          0          0          0
            numproc              92        101        240        240          0
            physpages         88764     109869          0 2147483647          0
            vmguarpages           0          0      33792 2147483647          0
            oomguarpages      88764     109869      26112 2147483647          0
            numtcpsock           28         46        360        360          0
```

4.10.5 Logfiles

4.10.5.1 Show Mail Queue

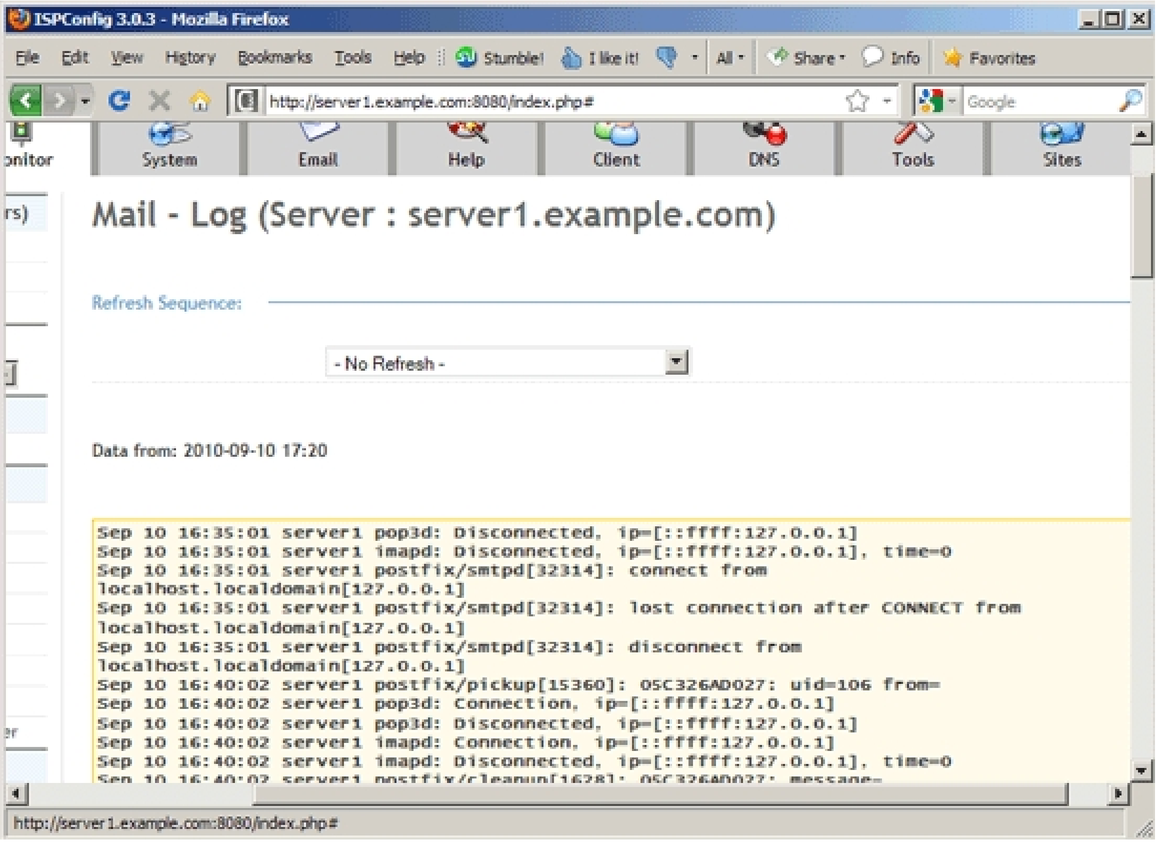
Here you can find details about the server's mail queue. Basically, these are the same details that the command

```
postqueue -p
```

would show.

4.10.5.2 Show Mail Log

You can find the last 100 lines of the selected server's mail log (`/var/log/mail.log` on Debian/Ubuntu) here. Under `Refresh Sequence` you can select if the information should be refreshed automatically while you are on this page (by default it is not refreshed), and in which interval.



Mail - Log (Server : server1.example.com)

Refresh Sequence:

- No Refresh -

Data from: 2010-09-10 17:20

```
Sep 10 16:35:01 server1 pop3d: Disconnected, ip=[::ffff:127.0.0.1]
Sep 10 16:35:01 server1 imapd: Disconnected, ip=[::ffff:127.0.0.1], time=0
Sep 10 16:35:01 server1 postfix/smtpd[32314]: connect from
localhost.localdomain[127.0.0.1]
Sep 10 16:35:01 server1 postfix/smtpd[32314]: lost connection after CONNECT from
localhost.localdomain[127.0.0.1]
Sep 10 16:35:01 server1 postfix/smtpd[32314]: disconnect from
localhost.localdomain[127.0.0.1]
Sep 10 16:40:02 server1 postfix/pickup[15360]: 05C326AD027: uid=106 from=
Sep 10 16:40:02 server1 pop3d: Connection, ip=[::ffff:127.0.0.1]
Sep 10 16:40:02 server1 pop3d: Disconnected, ip=[::ffff:127.0.0.1]
Sep 10 16:40:02 server1 imapd: Connection, ip=[::ffff:127.0.0.1]
Sep 10 16:40:02 server1 imapd: Disconnected, ip=[::ffff:127.0.0.1], time=0
Sep 10 16:40:02 server1 postfix/cleanup[1628]: 05C326AD027: message=
```

4.10.5.3 Show Mail Warn-Log

You can find the last 100 lines of the selected server's mail.warn log (*`/var/log/mail.warn`* on Debian/Ubuntu) here. Under *`Refresh Sequence`* you can select if the information should be refreshed automatically while you are on this page (by default it is not refreshed), and in which interval.

4.10.5.4 Show Mail Error-Log

You can find the last 100 lines of the selected server's mail.error log (*`/var/log/mail.err`* on Debian/Ubuntu) here. Under *`Refresh Sequence`* you can select if the information should be refreshed automatically while you are on this page (by default it is not refreshed), and in which interval.

4.10.5.5 Show System-Log

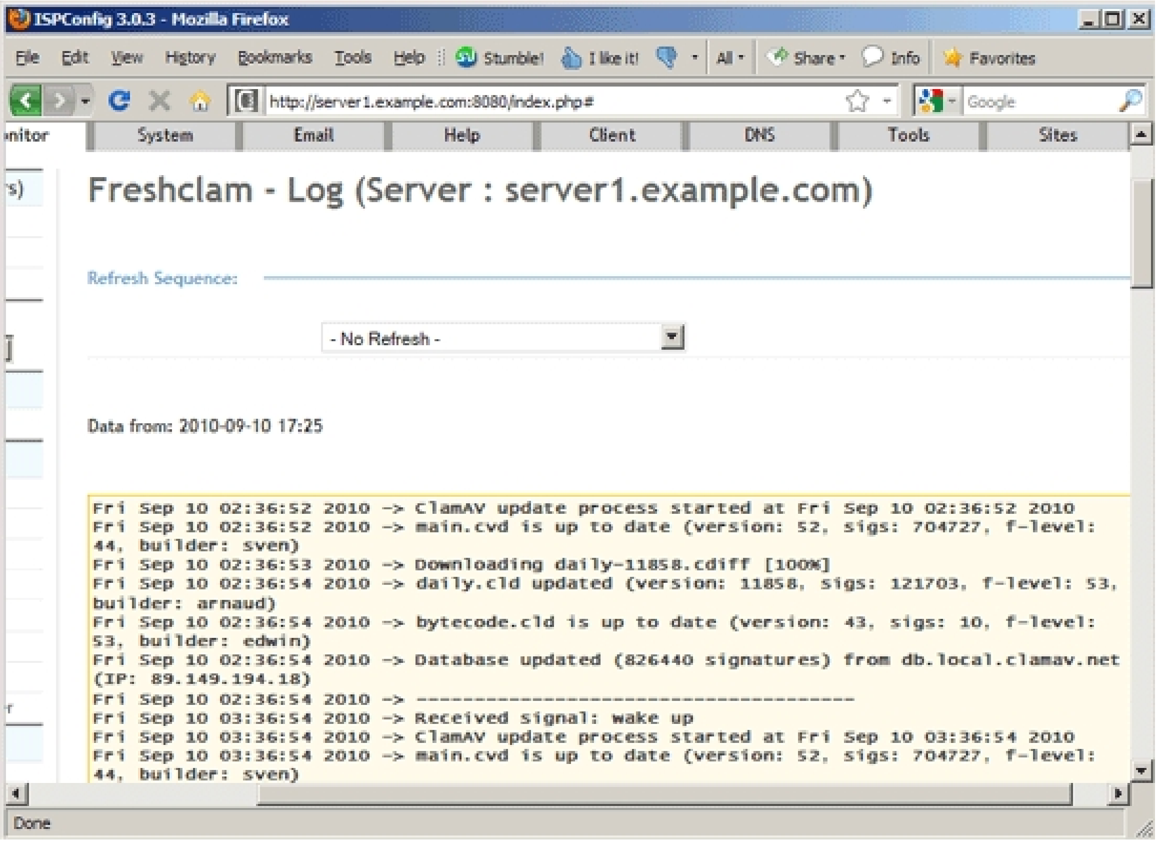
You can find the last 100 lines of the selected server's system log (*`/var/log/messages`* on Debian/Ubuntu) here. Under *`Refresh Sequence`* you can select if the information should be refreshed automatically while you are on this page (by default it is not refreshed), and in which interval.

4.10.5.6 Show ISPC Cron-Log

You can find the last 100 lines of the selected server's ISPConfig cron log (`/var/log/ispconfig/cron.log`) here - the ISPConfig background tasks are run by cron, and therefore this log contains information about what happened behind the scenes. Under *Refresh Sequence* you can select if the information should be refreshed automatically while you are on this page (by default it is not refreshed), and in which interval.

4.10.5.7 Show Freshclam-Log

You can find the last 100 lines of the selected server's freshclam log (`/var/log/clamav/freshclam.log` on Debian/Ubuntu) here - this log contains information regarding the virus signature updates of the server's virus scanner, ClamAV. Under *Refresh Sequence* you can select if the information should be refreshed automatically while you are on this page (by default it is not refreshed), and in which interval.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
nitor System Email Help Client DNS Tools Sites
Done

Freshclam - Log (Server : server1.example.com)

Refresh Sequence:

- No Refresh -

Data from: 2010-09-10 17:25

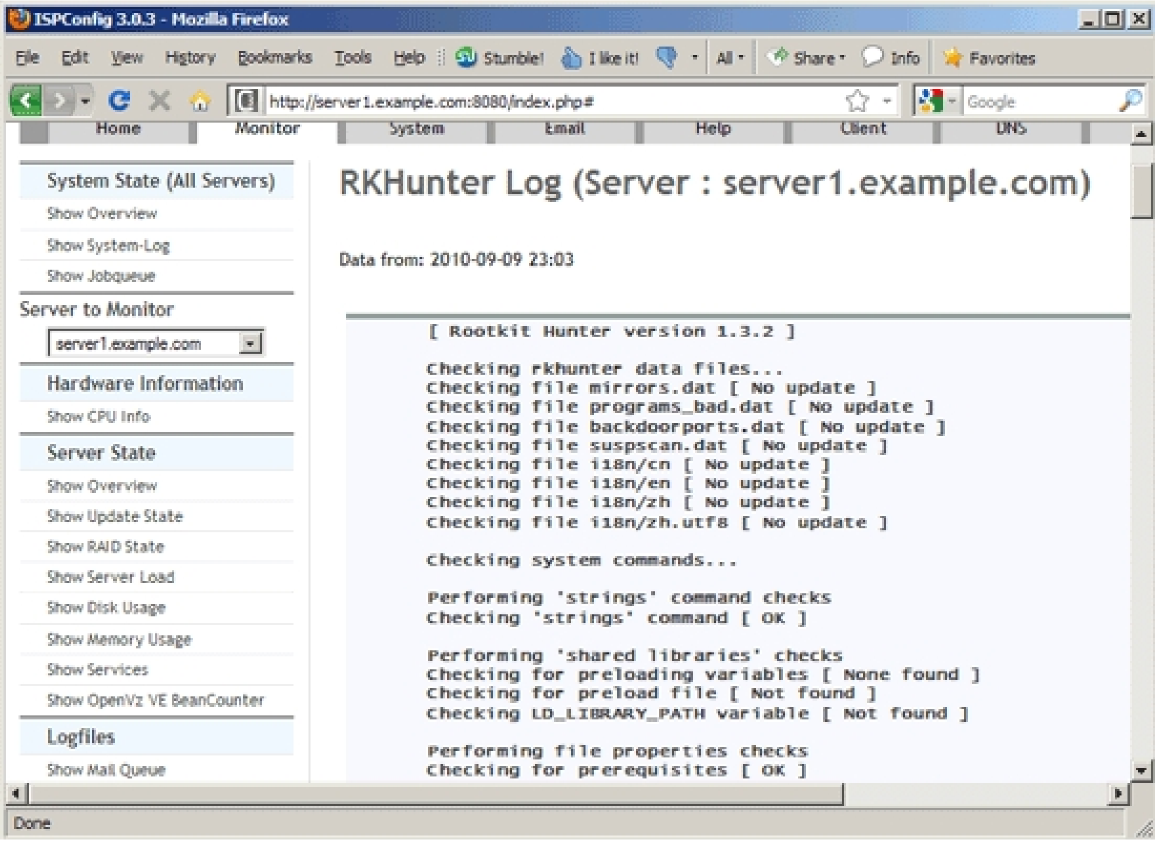
```
Fri Sep 10 02:36:52 2010 -> ClamAV update process started at Fri Sep 10 02:36:52 2010
Fri Sep 10 02:36:52 2010 -> main.cvd is up to date (version: 52, sigs: 704727, f-level:
44, builder: sven)
Fri Sep 10 02:36:53 2010 -> Downloading daily-11858.cdiff [100%]
Fri Sep 10 02:36:54 2010 -> daily.cld updated (version: 11858, sigs: 121703, f-level: 53,
builder: arnaud)
Fri Sep 10 02:36:54 2010 -> bytecode.cld is up to date (version: 43, sigs: 10, f-level:
53, builder: edwin)
Fri Sep 10 02:36:54 2010 -> Database updated (826440 signatures) from db.local.clamav.net
(IP: 89.149.194.18)
Fri Sep 10 02:36:54 2010 -> --------------------------------------
Fri Sep 10 03:36:54 2010 -> Received signal: wake up
Fri Sep 10 03:36:54 2010 -> ClamAV update process started at Fri Sep 10 03:36:54 2010
Fri Sep 10 03:36:54 2010 -> main.cvd is up to date (version: 52, sigs: 704727, f-level:
44, builder: sven)
```

4.10.5.8 Show Clamav-Log

You can find the last 100 lines of the selected server's clamav log (`/var/log/clamav/clamav.log` on Debian/Ubuntu) here - this log contains information regarding the server's virus scanner, ClamAV. Under *Refresh Sequence* you can select if the information should be refreshed automatically while you are on this page (by default it is not refreshed), and in which interval.

4.10.5.9 Show RKHunter-Log

You can find the last 100 lines of the selected server's rkhunter log (*`/var/log/rkhunter.log`* on Debian/Ubuntu) here - rkhunter is run by cron (usually once per night) and scans the server for malware/rootkits/trojans. The result of such a scan is logged in the rkhunter log file. Under *`Refresh Sequence`* you can select if the information should be refreshed automatically while you are on this page (by default it is not refreshed), and in which interval.



RKHunter Log (Server : server1.example.com)

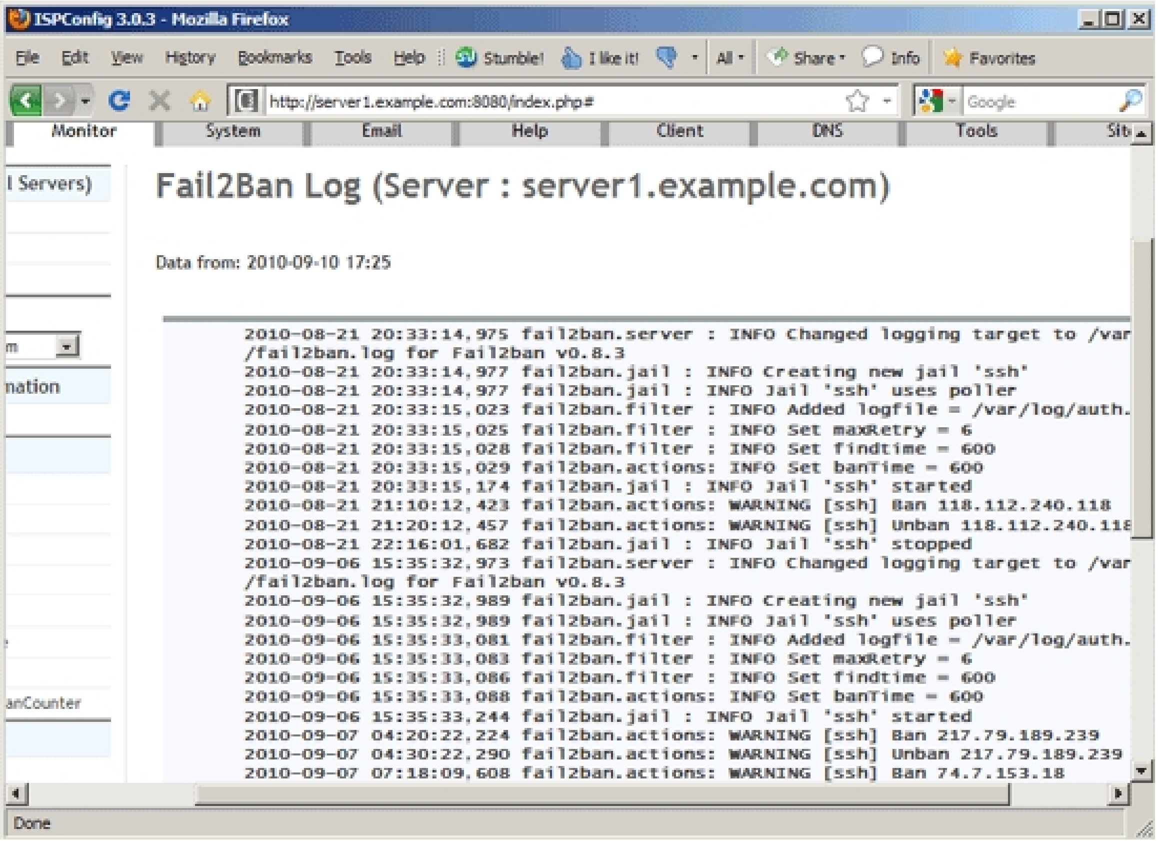
Data from: 2010-09-09 23:03

```
    [ Rootkit Hunter version 1.3.2 ]

Checking rkhunter data files...
Checking file mirrors.dat [ No update ]
Checking file programs_bad.dat [ No update ]
Checking file backdoorports.dat [ No update ]
Checking file suspscan.dat [ No update ]
Checking file i18n/cn [ No update ]
Checking file i18n/en [ No update ]
Checking file i18n/zh [ No update ]
Checking file i18n/zh.utf8 [ No update ]

Checking system commands...

Performing 'strings' command checks
Checking 'strings' command [ OK ]

Performing 'shared libraries' checks
Checking for preloading variables [ None found ]
Checking for preload file [ Not found ]
Checking LD_LIBRARY_PATH variable [ Not found ]

Performing file properties checks
Checking for prerequisites [ OK ]
```

4.10.5.10 Show fail2ban-Log

Fail2ban is a tool that observes login attempts to various services, e.g. SSH, FTP, SMTP, Apache, etc., and if it finds failed login attempts again and again from the same IP address or host, fail2ban stops further login attempts from that IP address/host byblocking it with an iptables firewall rule.

You can find the last 100 lines of the selected server's fail2ban log (*`/var/log/fail2ban.log`* on Debian/Ubuntu) here - it contains details about what services are monitored and what IP addresses got blocked due to a tried break-in attempt. Under *`Refresh Sequence`* you can select if the information should be refreshed automatically while you are on this page (by default it is not refreshed), and in which interval.


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
Monitor System Email Help Client DNS Tools Sit
l Servers)
m
nation
anCounter
Done

Fail2Ban Log (Server : server1.example.com)

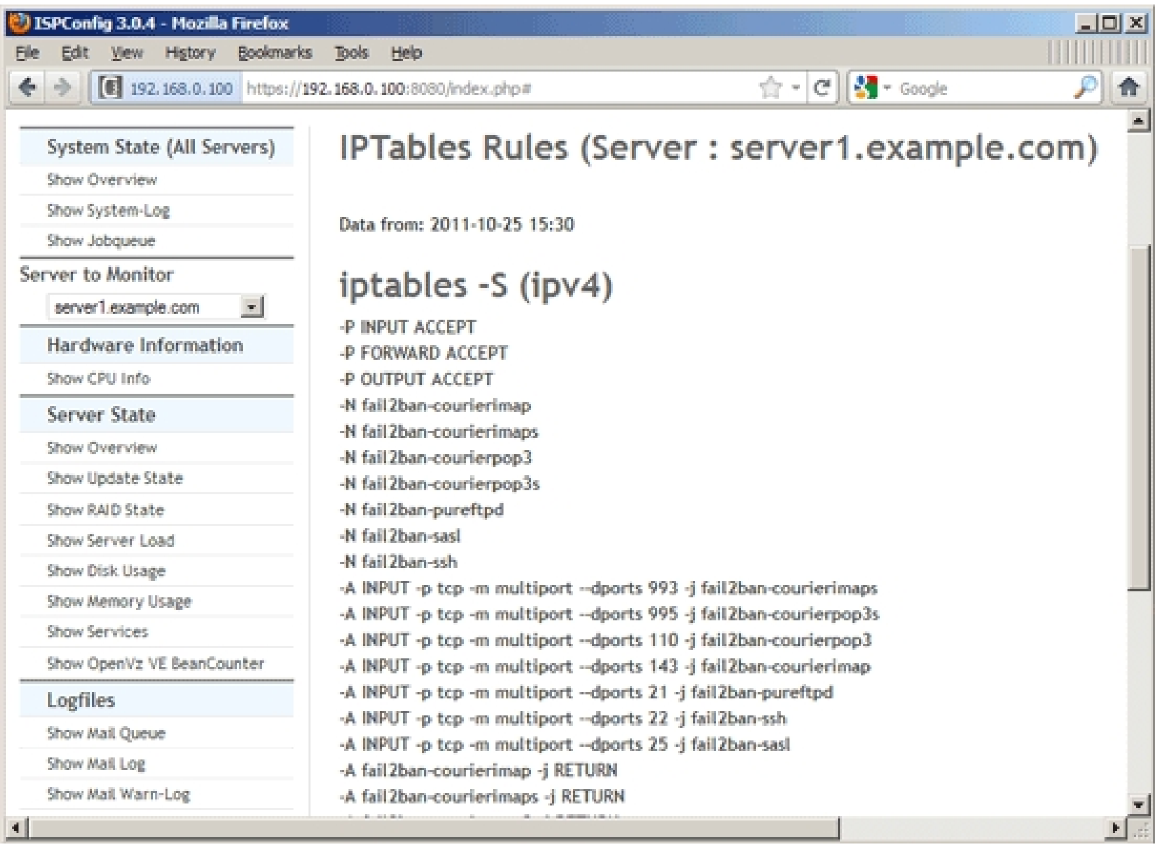
Data from: 2010-09-10 17:25

```
2010-08-21 20:33:14,975 fail2ban.server : INFO Changed logging target to /var
/fail2ban.log for Fail2ban v0.8.3
2010-08-21 20:33:14,977 fail2ban.jail : INFO Creating new jail 'ssh'
2010-08-21 20:33:14,977 fail2ban.jail : INFO Jail 'ssh' uses poller
2010-08-21 20:33:15,023 fail2ban.filter : INFO Added logfile = /var/log/auth.
2010-08-21 20:33:15,025 fail2ban.filter : INFO Set maxRetry = 6
2010-08-21 20:33:15,028 fail2ban.filter : INFO Set findtime = 600
2010-08-21 20:33:15,029 fail2ban.actions: INFO Set banTime = 600
2010-08-21 20:33:15,174 fail2ban.jail : INFO Jail 'ssh' started
2010-08-21 21:10:12,423 fail2ban.actions: WARNING [ssh] Ban 118.112.240.118
2010-08-21 21:20:12,457 fail2ban.actions: WARNING [ssh] Unban 118.112.240.118
2010-08-21 22:16:01,682 fail2ban.jail : INFO Jail 'ssh' stopped
2010-09-06 15:35:32,973 fail2ban.server : INFO Changed logging target to /var
/fail2ban.log for Fail2ban v0.8.3
2010-09-06 15:35:32,989 fail2ban.jail : INFO Creating new jail 'ssh'
2010-09-06 15:35:32,989 fail2ban.jail : INFO Jail 'ssh' uses poller
2010-09-06 15:35:33,081 fail2ban.filter : INFO Added logfile = /var/log/auth.
2010-09-06 15:35:33,083 fail2ban.filter : INFO Set maxRetry = 6
2010-09-06 15:35:33,086 fail2ban.filter : INFO Set findtime = 600
2010-09-06 15:35:33,088 fail2ban.actions: INFO Set banTime = 600
2010-09-06 15:35:33,244 fail2ban.jail : INFO Jail 'ssh' started
2010-09-07 04:20:22,224 fail2ban.actions: WARNING [ssh] Ban 217.79.189.239
2010-09-07 04:30:22,290 fail2ban.actions: WARNING [ssh] Unban 217.79.189.239
2010-09-07 07:18:09,608 fail2ban.actions: WARNING [ssh] Ban 74.7.153.18
```

If you want to unblock an IP address/host, take a look at chapter ***5.16 How Do I Unblock An IP Address That Got Blocked By fail2ban?***.

4.10.5.11 Show IPTables

Here you can see the firewall rules (for IPv4 and IPv6) that are currently active on the selected system. These are the outputs of the commands `iptables -S` (for IPv4) and `ip6tables -S` (for IPv6).



IPTables Rules (Server : server1.example.com)

Data from: 2011-10-25 15:30

iptables -S (ipv4)

```
-P INPUT ACCEPT
-P FORWARD ACCEPT
-P OUTPUT ACCEPT
-N fail2ban-courierimap
-N fail2ban-courierimaps
-N fail2ban-courierpop3
-N fail2ban-courierpop3s
-N fail2ban-pureftpd
-N fail2ban-sasl
-N fail2ban-ssh
-A INPUT -p tcp -m multiport --dports 993 -j fail2ban-courierimaps
-A INPUT -p tcp -m multiport --dports 995 -j fail2ban-courierpop3s
-A INPUT -p tcp -m multiport --dports 110 -j fail2ban-courierpop3
-A INPUT -p tcp -m multiport --dports 143 -j fail2ban-courierimap
-A INPUT -p tcp -m multiport --dports 21 -j fail2ban-pureftpd
-A INPUT -p tcp -m multiport --dports 22 -j fail2ban-ssh
-A INPUT -p tcp -m multiport --dports 25 -j fail2ban-sasl
-A fail2ban-courierimap -j RETURN
-A fail2ban-courierimaps -j RETURN
```

4.11 Help

If this module isn't enabled for a normal user, you can enable it on the `Users` tab under `System > CP Users > Edit user`.

4.11.1 Support

This is a ticket system where users can send messages to their reseller or the server administrator if they need help.

4.11.1.1 Send message

You can create a new ticket here. You will see the `Support Message` form with the tab `Message`.

Support Message

Message

The form has the following fields:

- *Recipient ID*: Normal users cannot select a recipient here because ISPConfig determines the recipient itself - it is the ISPConfig administrator. Only if you are logged in as the ISPConfig administrator can you select the recipient (because the administrator is allowed to send messages to all ISPConfig users).
- *Subject*: Fill in the subject of your request.
- *Message*: Fill in your message.

4.11.1.2 View messages

Here you can see a list of all tickets opened by you (answered or unanswered).

4.11.2 FAQ

The FAQ module allows you to define FAQ sections (= categories) (like "General", "Technical", "Billing", "Contract", etc.) and FAQ entries that you can allocate to the FAQ sections you defined before. These FAQ can be seen by all resellers and clients for who the *Help* module is activated.

4.11.2.1 Manage Sections

Here you can add FAQ sections.

To add a new section, click on the *Add a new section* button. This will lead you to the *FAQ Sections* form with the tab *FAQ*.

FAQ Sections

FAQ

The form has the following field:

- *Section Name*: Type in a name for an FAQ section/category. You might want to create categories like "General", "Technical", "Billing", "Contract", etc.

4.11.2.2 Manage Questions

Here you can add FAQ questions and allocate them to the sections/categories you defined before.

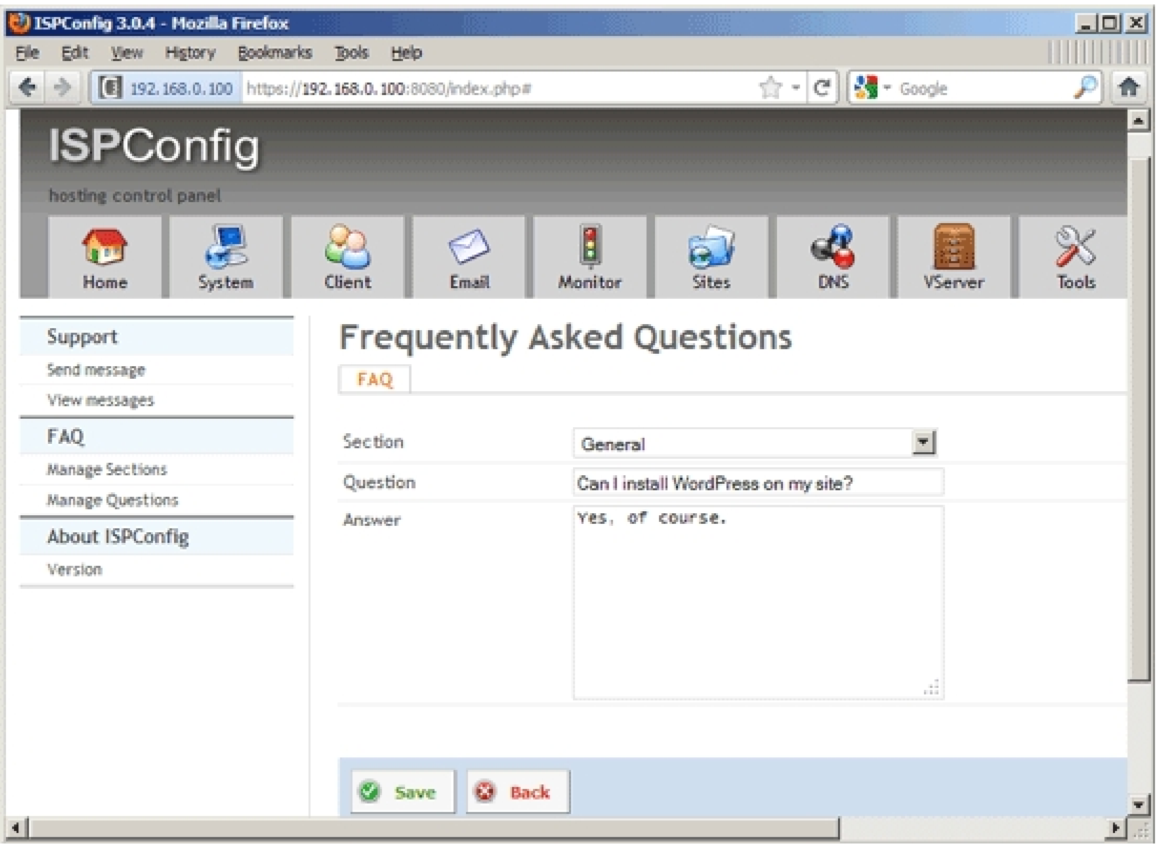
To add a new question, click on the *Add a new question & answer pair* button. This will lead you to the *Frequently Asked Questions* form with the tab *FAQ*.

Frequently Asked Questions

FAQ

The form has the following fields:

- *Section*: Select the FAQ section to which you want to allocate this question.
- *Question*: Type in the question in this field.
- *Answer*: Provide the answer in this textarea.



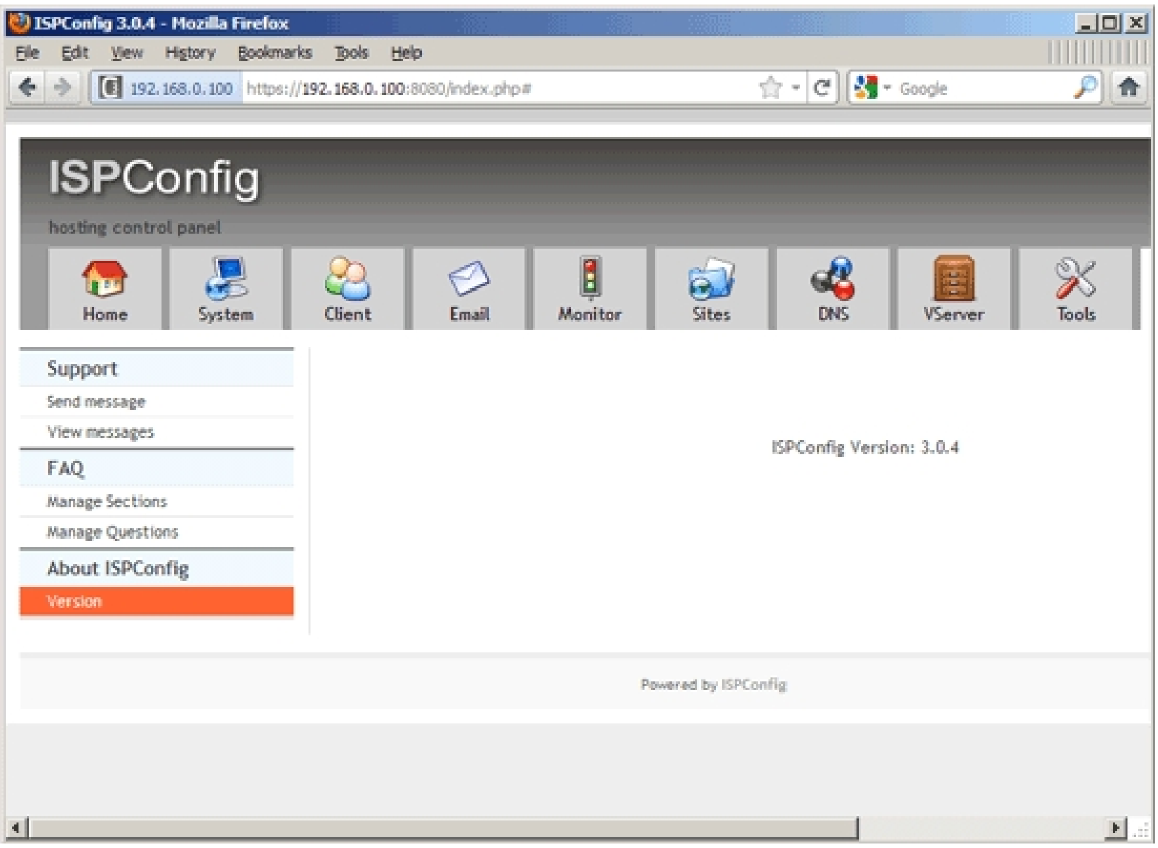
Frequently Asked Questions

Section	General
Question	Can I install WordPress on my site?
Answer	Yes, of course.

4.11.3 About ISPConfig

4.11.3.1 Version

Shows the currently installed ISPConfig 3 version:



ISPConfig Version: 3.0.4

4.12 Domains

If you use this module, your customers can only select one of the domains the admin creates for them. They can not freely edit the domain field.

This module is active only if you also check the *Use the domain-module to add new domains* checkbox on the *Domains* tab under *System > System > Interface Config*.

4.12.1 Domains

4.12.1.1 Domains

Here you can add domains to your server that clients can later on select when they create a new web site.

To add a new domain, click on the *Add new Domain* button. This will lead you to the *Domain* form with the tab *Domain*.

Domain

Domain

The form has the following fields:

- *`Domain`*: Type in a domain name that you want to allocate to a client, e.g. *`example.com`* (without any subdomain like *`www`*).
- *`Client`*: Select the client to which you want to allocate the domain from the drop-down menu. This client will then be able to select the domain from a drop-down menu when he creates a web site.

4.13 VServer

On this tab we can create and manage virtual machines (currently the only supported virtualization technique is OpenVZ). To make use of this module, you need at least one ISPConfig server (master or slave) where OpenVZ is installed, and ISPConfig must be installed on the host system, not inside an OpenVZ container.

4.13.1 OpenVZ

4.13.1.1 Virtual Servers

This is where we can create new and edit/delete existing virtual machines.

To create a new virtual machine, click the *`Add new record`* button. This will lead you to the *`Openvz virtual server`* form with the tabs *`Virtual server`* and *`Advanced`*.

Openvz virtual server

Virtual server

This is where the virtual machine is actually created. The form has the following fields:

- *`Hostserver`*: Select the ISPConfig server where the virtual machine will be created. Please note that OpenVZ must be installed on that server (this means that this server must not be a virtual machine itself). This field lists only the ISPConfig servers for which you have checked the *`VServer-Server`* checkbox under *`System > System > Server Services`* (see chapter ***4.9.2.1 Server Services***).
- *`Client`*: Here you select the client that owns the new virtual machine.

- *OSTemplate*: Select the operating system template for the new virtual machine. This template must previously have been defined under *VServer > OpenVZ > OS Templates* (see chapter ***4.13.1.2 OS Templates***).

- *Template*: Select a basic configuration template for this virtual machine. This applies a predefined OpenVZ configuration to the new virtual machine; you can adjust these values on the *Advanced* tab. This basic configuration template must previously have been defined under *VServer > OpenVZ > VM Templates* (see chapter ***4.13.1.3 VM Templates***).

- *IP address*: Select the IP address for this virtual machine. The available IP addresses must previously have been defined under *VServer > OpenVZ > IP addresses* (see chapter ***4.13.1.4 IP addresses***).

- *Hostname*: Type in the fully qualified hostname of the virtual machine (e.g. *vm1234.example.com*).

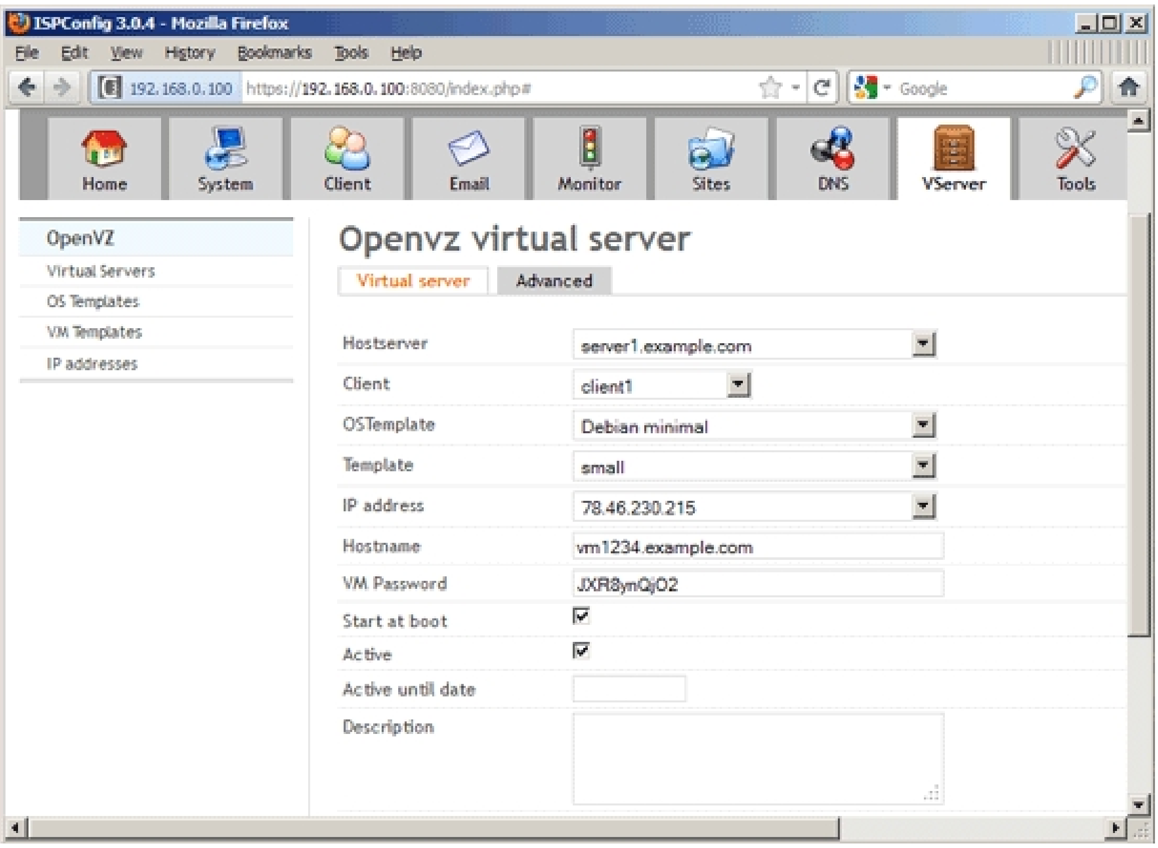
- *VM Password*: This is the root password of the virtual machine. This field is filled automatically with a password generated by ISPConfig, but you can type in your own.

- *Start at boot*: Check this if you want to have the virtual machine started automatically when the system boots.

- *Active*: Defines whether this virtual machine is active or not.

- *Active until Date*: This defines a date up to which the virtual machine is active. After that date, the virtual machine is closed down. Leave empty to let the virtual machine run forever.

- *Description* (optional): Fill in a description for this virtual machine if needed.



Openvz virtual server

Hostserver	server1.example.com
Client	client1
OSTemplate	Debian minimal
Template	small
IP address	78.46.230.215
Hostname	vm1234.example.com
VM Password	JXR8ynQjO2
Start at boot	☑
Active	☑
Active until date	
Description	

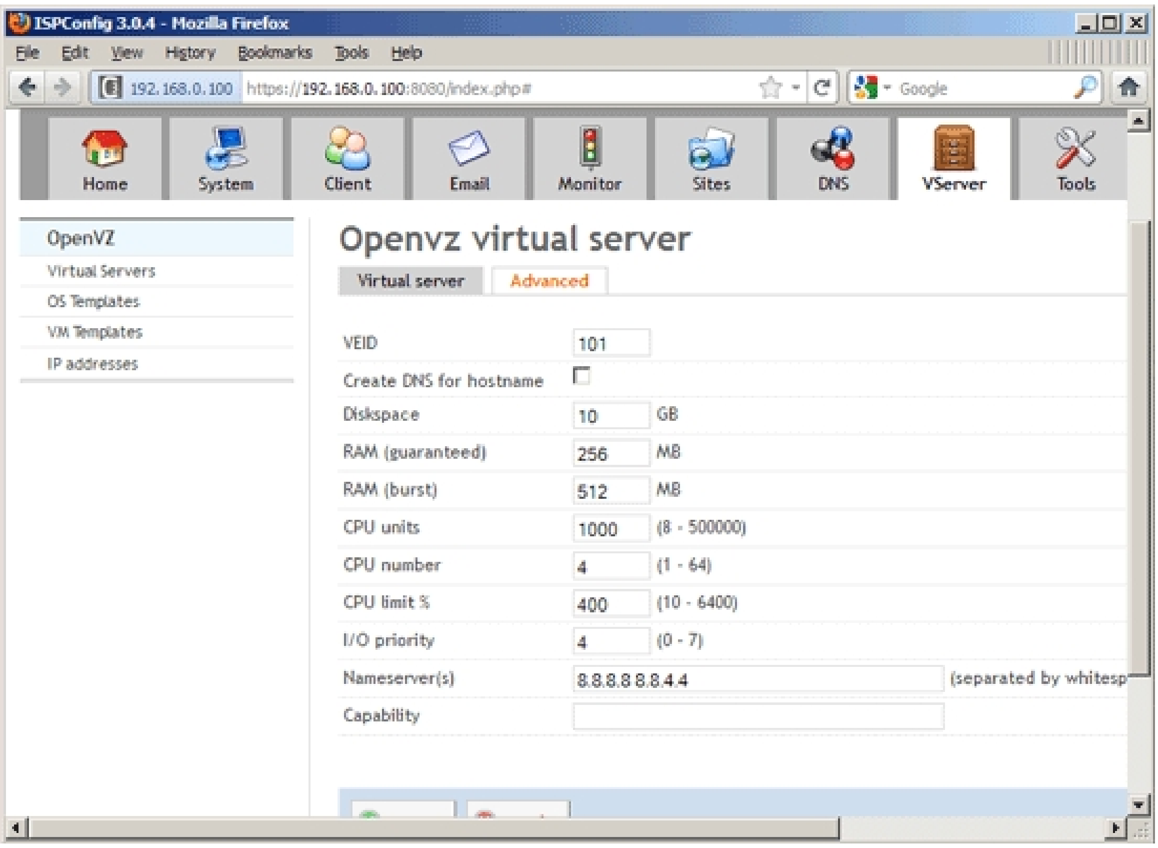
Advanced

On this tab you can fine-tune the virtual machine's settings. Most of the values you see here have been filled in by the VM Template you've chosen on the `Virtual server` tab in the field `Template`.

- *VEID*: This is the ID of the virtual machine. Each virtual machine must have its own unique ID. Note that VEID <= 100 are reserved for OpenVZ internal purposes.
- *Create DNS for hostname*: Check this if you want ISPConfig to add a DNS record (A record) for the hostname of the virtual machine to the appropriate DNS zone. Please note that the zone must already exist in the *DNS* module, or otherwise ISPConfig will not create the hostname DNS record.
- *Diskspace*: Specify the diskspace of the virtual machine (in GB).
- *RAM (guaranteed)*: This is the minimum amount of RAM that is allocated to this virtual machine (guaranteed). If the host system has unused RAM available, more RAM will be allocated to the virtual machine (up to the value specified in *RAM (burst)*).
- *RAM (burst)*: This is the maximum amount of RAM that can be allocated to the virtual machine.
- *CPU units*: CPU weight for a virtual machine. Argument is a positive non-zero number, which is passed to and used in kernel fair scheduler. The larger the number is, the more CPU time this virtual machine gets. Maximum value is 500000, minimal is 8. Number is relative to weights of all the other running virtual machines. If not specified, the default value 1000 is used.
- *CPU number*: This is the number of CPUs that this virtual machine can use. Do not specify more CPUs than your system has.
- *CPU limit %*: Limit of CPU usage for the virtual machine, in %. Note if the computer has 2 CPUs, it has a total of 200% CPU time. Default CPU limit is 0 (no CPU limit).
- *I/O priority*: Assigns an I/O priority to the virtual machine. Priority range is 0-7. The greater the priority is, the more time for I/O activity the virtual machine has. By default each virtual machine has a priority of 4.
- *Nameserver(s)*: Set DNS server IP address(es) for a virtual machine. If you want to set several nameservers, separate them with spaces (e.g. *8.8.8.8 8.8.4.4*).
- *Capability*: Format: *capname:on|off* Sets capability inside a virtual machine. Note a virtual machine has a default set of capabilities, thus any operation on capabilities is "logical and" with the default capability mask. You can use the following values: *chown, dac_override, dac_read_search, fowner, fsetid, kill, setgid, setuid, setpcap, linux_immutable, net_bind_service, net_broadcast, net_admin, net_raw, ipc_lock, ipc_owner, sys_module, sys_rawio, sys_chroot, sys_ptrace, sys_pacct, sys_admin, sys_boot, sys_nice, sys_resource, sys_time, sys_tty_config, mknod, lease, setveid, ve_admin*.

For example, if you have problems running a Pure-FTPd server inside a Debian/Ubuntu virtual machine, you can set the following capabilities to solve the problem: *CHOWN:on DAC_READ_SEARCH:on SETGID:on SETUID:on NET_BIND_SERVICE:on NET_ADMIN:on SYS_CHROOT:on SYS_NICE:on*

WARNING: setting some of those capabilities may have far reaching security implications, so do not do it unless you know what you are doing. Also note that setting `setpcap:on` for a virtual machine will most probably lead to inability to start it.



Openvz virtual server

VEID	101	
Create DNS for hostname	□	
Diskspace	10	GB
RAM (guaranteed)	256	MB
RAM (burst)	512	MB
CPU units	1000	(8 - 500000)
CPU number	4	(1 - 64)
CPU limit %	400	(10 - 6400)
I/O priority	4	(0 - 7)
Nameserver(s)	8.8.8.8 8.8.4.4	(separated by whitesp
Capability		

4.13.1.2 OS Templates

Here we tell ISPConfig what OpenVZ operating system templates are available on the OpenVZ servers managed by ISPConfig. Please note that the templates that you define here must exist on the appropriate OpenVZ server in the `/vz/template/cache` directory.

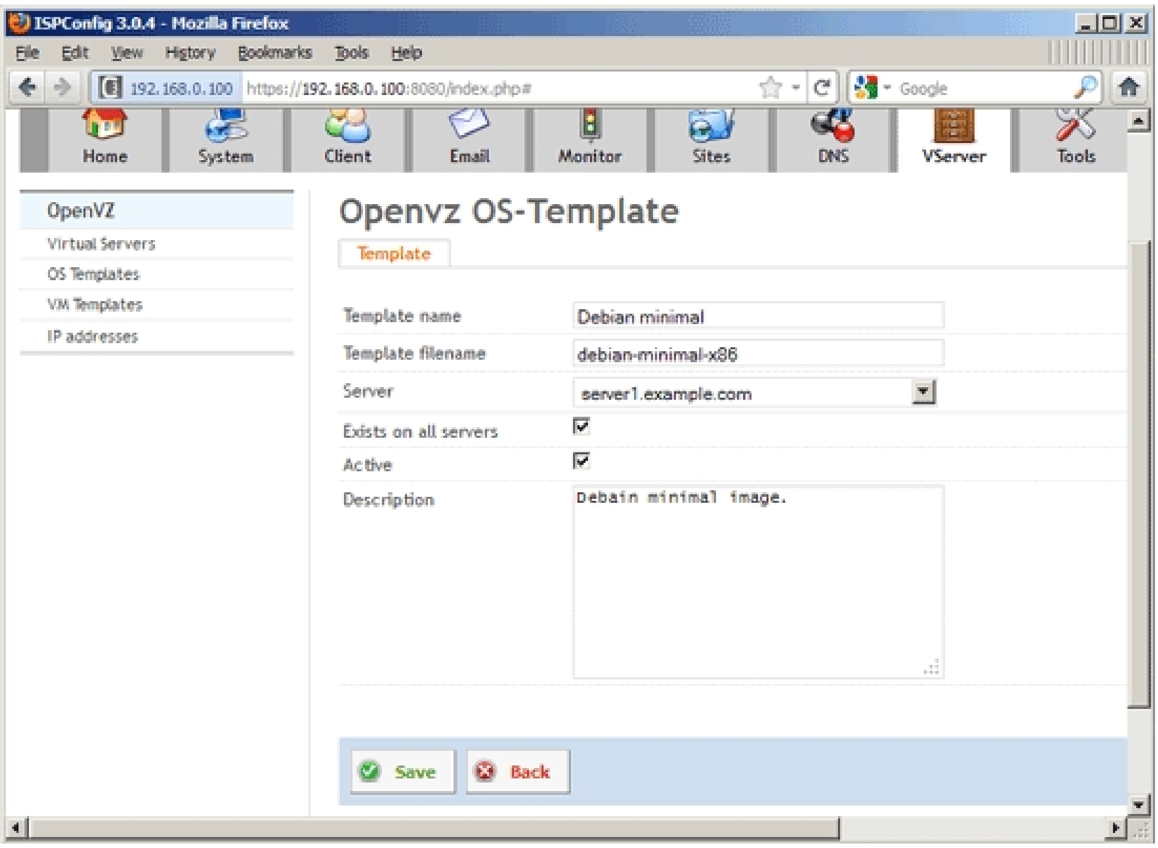
To create a new OS template, click the `Add new record` button. This will lead you to the `Openvz OS-Template` form with the tab `Template`.

Openvz OS-Template

Template

The form has the following fields:

- `Template name`: Fill in a name for the OS template, like `Debian 6.0 minimal` or `Ubuntu 11.04 LAMP`.
- `Template filename`: Specify the filename (without the `.tar.gz` extension), as it resides in the `/vz/template/cache` directory of the appropriate server, e.g. `debian-6.0-minimal-x86` or `ubuntu-11.04-lamp-x86`.
- `Server`: Select the ISPConfig server where this OS template is located. This field lists only the ISPConfig servers for which you have checked the `VServer-Server` checkbox under `System > System > Server Services` (see chapter **4.9.2.1 Server Services**).
- `Exists on all servers`: Check this if this OS template exists on all your OpenVZ servers.
- `Active`: Defines whether this OS template is active or not.
- `Description` (optional): Fill in a description for this OS template if needed.


ISPConfig 3.0.4 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
192.168.0.100 https://192.168.0.100:8080/index.php#
Google
Home System Client Email Monitor Sites DNS VServer Tools
OpenVZ
Virtual Servers
OS Templates
VM Templates
IP addresses
Template
Save
Back

Openvz OS-Template

Template name	Debian minimal
Template filename	debian-minimal-x86
Server	server1.example.com
Exists on all servers	☑
Active	☑
Description	Debain minimal image.

4.13.1.3 VM Templates

Here we can create configuration templates for virtual machines. This applies a predefined OpenVZ configuration to a virtual machine, such as RAM (guaranteed) or CPU units.

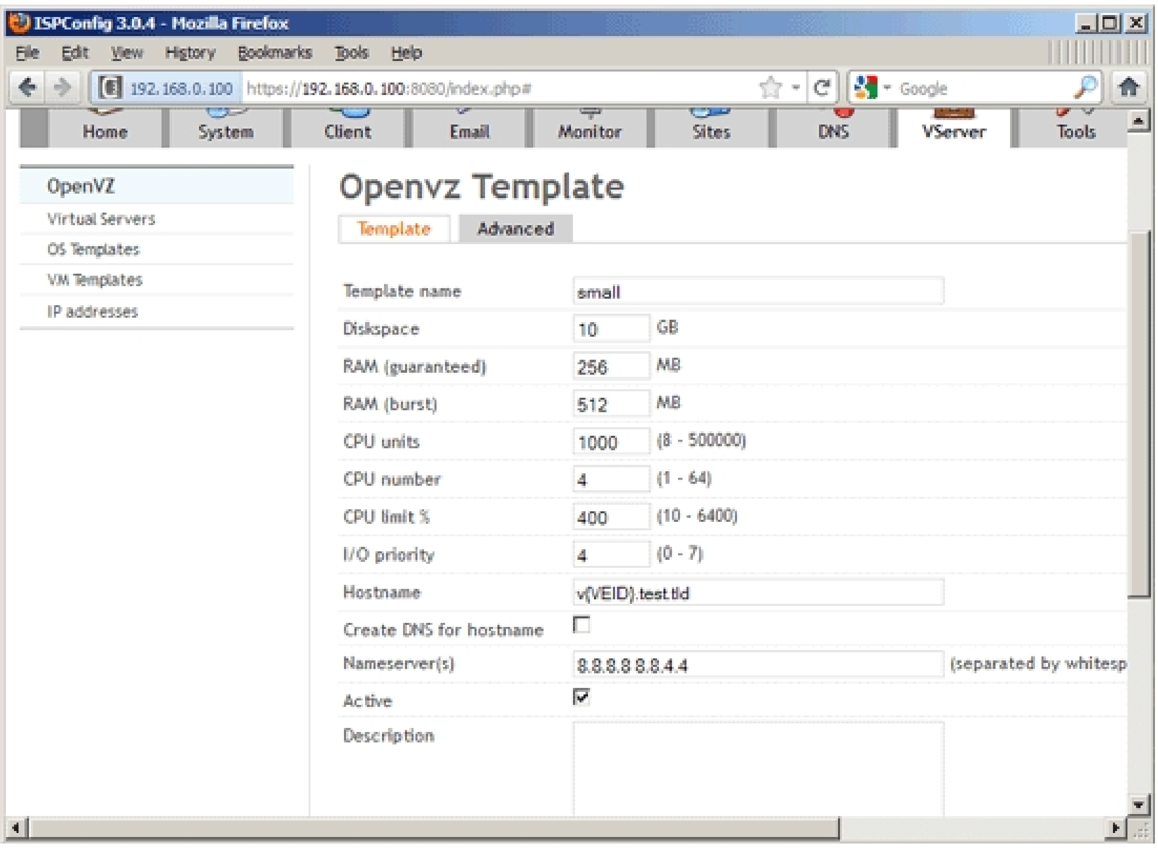
To create a new VM template, click the *Add new record* button. This will lead you to the *Openvz Template* form with the tabs *Template* and *Advanced*.

Openvz Template

Template

On this tab you can configure the basic configuration settings for the VM template. The form has the following fields:

- *Template name*: Please specify a name for the VM template.
- *Diskspace*: Specify the diskspace (in GB).
- *RAM (guaranteed)*: This is the minimum amount of RAM that is allocated to the virtual machine (guaranteed). If the host system has unused RAM available, more RAM will be allocated to the virtual machine (up to the value specified in *RAM (burst)*).
- *RAM (burst)*: This is the maximum amount of RAM that can be allocated to the virtual machine.
- *CPU units*: CPU weight for a virtual machine. Argument is a positive non-zero number, which is passed to and used in kernel fair scheduler. The larger the number is, the more CPU time this virtual machine gets. Maximum value is 500000, minimal is 8. Number is relative to weights of all the other running virtual machines. If not specified, the default value 1000 is used.
- *CPU number*: This is the number of CPUs that this virtual machine can use. Do not specify more CPUs than your system has.
- *CPU limit %*: Limit of CPU usage for the virtual machine, in %. Note if the computer has 2 CPUs, it has a total of 200% CPU time. Default CPU limit is 0 (no CPU limit).
- *I/O priority*: Assigns an I/O priority to the virtual machine. Priority range is 0-7. The greater the priority is, the more time for I/O activity the virtual machine has. By default each virtual machine has a priority of 4.
- *Hostname*: Here you can specify a hostname pattern to be applied to a new virtual machine, like *v{VEID}.test.tld*. *{VEID}* will be replaced with the virtual machine's actual VEID, so the virtual machine with the VEID 101 would get the hostname *v101.test.tld*.
- *Create DNS for hostname*: Check this if you want ISPConfig to add a DNS record (A record) for the hostname of the virtual machine to the appropriate DNS zone. Please note that the zone must already exist in the *DNS* module, or otherwise ISPConfig will not create the hostname DNS record.
- *Nameserver(s)*: Set DNS server IP address(es) for a virtual machine. If you want to set several nameservers, separate them with spaces (e.g. *8.8.8.8 8.8.4.4*).
- *Active*: This defines whether this VM template is active or not.
- *Description* (optional): Fill in a description for this VM template if needed.


ISPConfig 3.0.4 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
192.168.0.100 https://192.168.0.100:8080/index.php#
Google
Home System Client Email Monitor Sites DNS VServer Tools
OpenVZ
Virtual Servers
OS Templates
VM Templates
IP addresses
Template Advanced

Openvz Template

Template name	small
Diskspace	10 GB
RAM (guaranteed)	256 MB
RAM (burst)	512 MB
CPU units	1000 (8 - 500000)
CPU number	4 (1 - 64)
CPU limit %	400 (10 - 6400)
I/O priority	4 (0 - 7)
Hostname	v{VEID}.test.tld
Create DNS for hostname	□
Nameserver(s)	8.8.8.8 8.8.4.4 (separated by whitesp
Active	☑
Description	

Advanced

On this tab you can fine-tune your VM template and configure expert settings. Some of these settings have certain interdependencies. To find out if your configuration is consistent, please visit:

- UBC: ***http://wiki.openvz.org/Category:UBC***
- UBC systemwide configuration: ***http://wiki.openvz.org/UBC_systemwide_configuration***
- UBC consistency check: ***http://wiki.openvz.org/UBC_consistency_check***
- UBC primary parameters: ***http://wiki.openvz.org/UBC_primary_parameters***
- UBC secondary parameters: ***http://wiki.openvz.org/UBC_secondary_parameters***
- UBC auxiliary parameters: ***http://wiki.openvz.org/UBC_auxiliary_parameters***

- *`Numproc`*: Format: *`items[:items]`* Maximum number of processes and kernel-level threads. Setting the barrier and the limit to different values does not make practical sense.
- *`Numtcpsock`*: Format: *`items[:items]`* Maximum number of TCP sockets. This parameter limits the number of TCP connections and, thus, the number of clients the server application can handle in parallel. Setting the barrier and the limit to different values does not make practical sense.

- *`Numothersock`*: Format: *`items[:items]`* Maximum number of non-TCP sockets (local sockets, UDP and other types of sockets). Setting the barrier and the limit to different values does not make practical sense.

- *`Vmguarpages`*: Format: *`pages[:pages]`* Memory allocation guarantee. This parameter controls how much memory is available to a virtual machine. The barrier is the amount of memory that virtual machine's applications are guaranteed to be able to allocate. The meaning of the limit is currently unspecified; it should be set to 2,147,483,647.

- *`Kmemsize`*: Format: *`bytes[:bytes]`* Maximum amount of kernel memory used. This parameter is related to *`Numproc`*. Each process consumes certain amount of kernel memory - 16 KB at leas, 30-50 KB typically. Very large processes may consume a bit more. It is important to have a certain safety gap between the barrier and the limit of this parameter: equal barrier and limit may lead to the situation where the kernel will need to kill virtual machine applications to keep the kmemsize usage under the limit.

- *`Tcpsndbuf`*: Format: *`bytes[:bytes]`* Maximum size of TCP send buffers. Barrier should be not less than 64 KB, and difference between barrier and limit should be equal to or more than value of *`Numtcpsock`* multiplied by 2.5 KB.

- *`Tcprcvbuf`*: Format: *`bytes[:bytes]`* Maximum size of TCP receive buffers. Barrier should be not less than 64 KB, and difference between barrier and limit should be equal to or more than value of *`Numtcpsock`* multiplied by 2.5 KB.

- *`Othersockbuf`*: Format: *`bytes[:bytes]`* Maximum size of other (non-TCP) socket send buffers. If virtual machine processes need to send very large datagrams, the barrier should be set accordingly. Increased limit is necessary for high performance of communications through local (UNIX-domain) sockets.

- *`Dgramrcvbuf`*: Format: *`bytes[:bytes]`* Maximum size of other (non-TCP) socket receive buffers. If virtual machine processes need to send very large datagrams, the barrier should be set accordingly. The difference between the barrier and the limit is not needed.

- *`Oomguarpages`*: Format: *`pages[:pages]`* Guarantees against OOM kill. Under this beancounter the kernel accounts the total amount of memory and swap space used by the virtual machine processes. The barrier of this parameter is the out-of-memory guarantee. If the oomguarpages usage is below the barrier, processes of this virtual machine are guaranteed not to be killed in out-of-memory situations. The meaning of limit is currently unspecified; it should be set to 2,147,483,647.

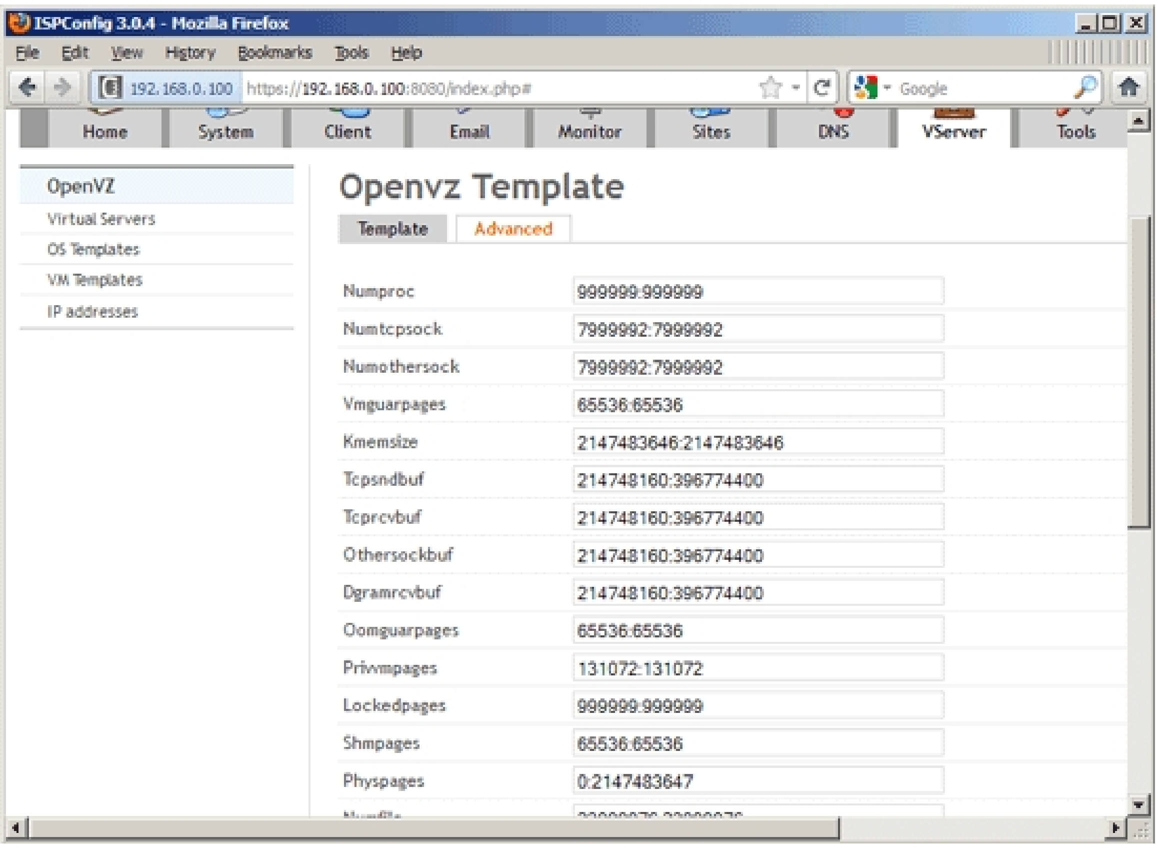
- *`Privvmpages`*: Format: *`pages[:pages]`* Allows controlling the amount of memory allocated by the applications. For shared (mapped as MAP_SHARED) pages, each virtual machine really using a memory page is charged for the fraction of the page (depending on the number of others using it). For "potentially private" pages (mapped as MAP_PRIVATE), the virtual machine is charged either for a fraction of the size or for the full size if the allocated address space. In the latter case, the physical pages associated with the allocated address space may be in memory, in swap or not physically allocated yet. The barrier and the limit of this parameter control the upper boundary of the total size of allocated memory. Note that this upper boundary does not guarantee that virtual machine will be able to allocate that much memory. The primary mechanism to control memory allocation is the *`Vmguarpages`* guarantee.

- *`Lockedpages`*: Format: *`pages[:pages]`* Maximum number of pages acquired by mlock(2).

- *`Shmpages`*: Format: *`pages[:pages]`* Maximum IPC SHM segment size. Setting the barrier and the limit to different values does not make practical sense.
- *`Physpages`*: Format: *`pages[:pages]`* This is currently an accounting-only parameter. It shows the usage of RAM by this virtual machine. Barrier should be set to 0, and limit should be set to 2,147,483,647.
- *`Numfile`*: Format: *`items[:items]`* Maximum number of open files. Setting the barrier and the limit to different values does not make practical sense.
- *`Avnumproc`*: Format: *`items[:items]`* The expected average number of processes.
- *`Numflock`*: Format: *`items[:items]`* Maximum number of file locks. Safety gap should be between barrier and limit.
- *`Numpty`*: Format: *`items[:items]`* Number of pseudo-terminals (PTY). Note that in OpenVZ each virtual machine can have not more than 255 PTYs. Setting the barrier and the limit to different values does not make practical sense.
- *`Numsiginfo`*: Format: *`items[:items]`* Number of siginfo structures. Setting the barrier and the limit to different values does not make practical sense.
- *`Dcachesize`*: Format: *`bytes[:bytes]`* Maximum size of filesystem-related caches, such as directory entry and inode caches. Exists as a separate parameter to impose a limit causing file operations to sense memory shortage and return an errno to applications, protecting from memory shortages during critical operations that should not fail. Safety gap should be between barrier and limit.
- *`Numiptent`*: Format: *`num[:num]`* Number of iptables (netfilter) entries. Setting the barrier and the limit to different values does not make practical sense.
- *`Swappages`*: Format: *`pages[:pages]`* The amount of swap space to show in container. The configuration of this parameter doesn't affect security and stability of the whole system or isolation between containers. Its configuration only affects the way OpenVZ kernel reports about available swap in a container. This is needed for some applications which refuse to run inside a container unless the kernel report that no less than some specific amount of swap is available.
- *`Capability`*: Format: *`capname:on|off`* Sets capability inside a virtual machine. Note a virtual machine has a default set of capabilities, thus any operation on capabilities is "logical and" with the default capability mask. You can use the following values: *`chown, dac_override, dac_read_search, fowner, fsetid, kill, setgid, setuid, setpcap, linux_immutable, net_bind_service, net_broadcast, net_admin, net_raw, ipc_lock, ipc_owner, sys_module, sys_rawio, sys_chroot, sys_ptrace, sys_pacct, sys_admin, sys_boot, sys_nice, sys_resource, sys_time, sys_tty_config, mknod, lease, setveid, ve_admin`*.

For example, if you have problems running a Pure-FTPd server inside a Debian/Ubuntu virtual machine, you can set the following capabilities to solve the problem: *`CHOWN:on DAC_READ_SEARCH:on SETGID:on SETUID:on NET_BIND_SERVICE:on NET_ADMIN:on SYS_CHROOT:on SYS_NICE:on`*

WARNING: setting some of those capabilities may have far reaching security implications, so do not do it unless you know what you are doing. Also note that setting *`setpcap:on`* for a virtual machine will most probably lead to inability to start it.


ISPConfig 3.0.4 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
192.168.0.100 https://192.168.0.100:8080/index.php#
Google
Home System Client Email Monitor Sites DNS VServer Tools
OpenVZ
Virtual Servers
OS Templates
VM Templates
IP addresses
Template Advanced

Openvz Template

Numproc	999999:999999
Numtcpsock	7999992:7999992
Numothersock	7999992:7999992
Vmguarpages	65536:65536
Kmemsize	2147483646:2147483646
Tcpsndbuf	214748160:396774400
Tcprcvbuf	214748160:396774400
Othersockbuf	214748160:396774400
Dgramrcvbuf	214748160:396774400
Oomguarpages	65536:65536
Privvmpages	131072:131072
Lockedpages	999999:999999
Shmpages	65536:65536
Physpages	0:2147483647

4.13.1.4 IP addresses

Here you can define IP addresses that can then be allocated to virtual machines (see chapter ***4.13.1.1 Virtual Servers***). If you do not define any IP addresses here, you cannot create virtual machines. Please note that you can allocate one IP to just one virtual machine, so make sure you define enough free IP addresses here that can then be used by your virtual machines.

To create a new IP address, click the *Add new record* button. This will lead you to the *Openvz IP address* form with the tab *IP address*.

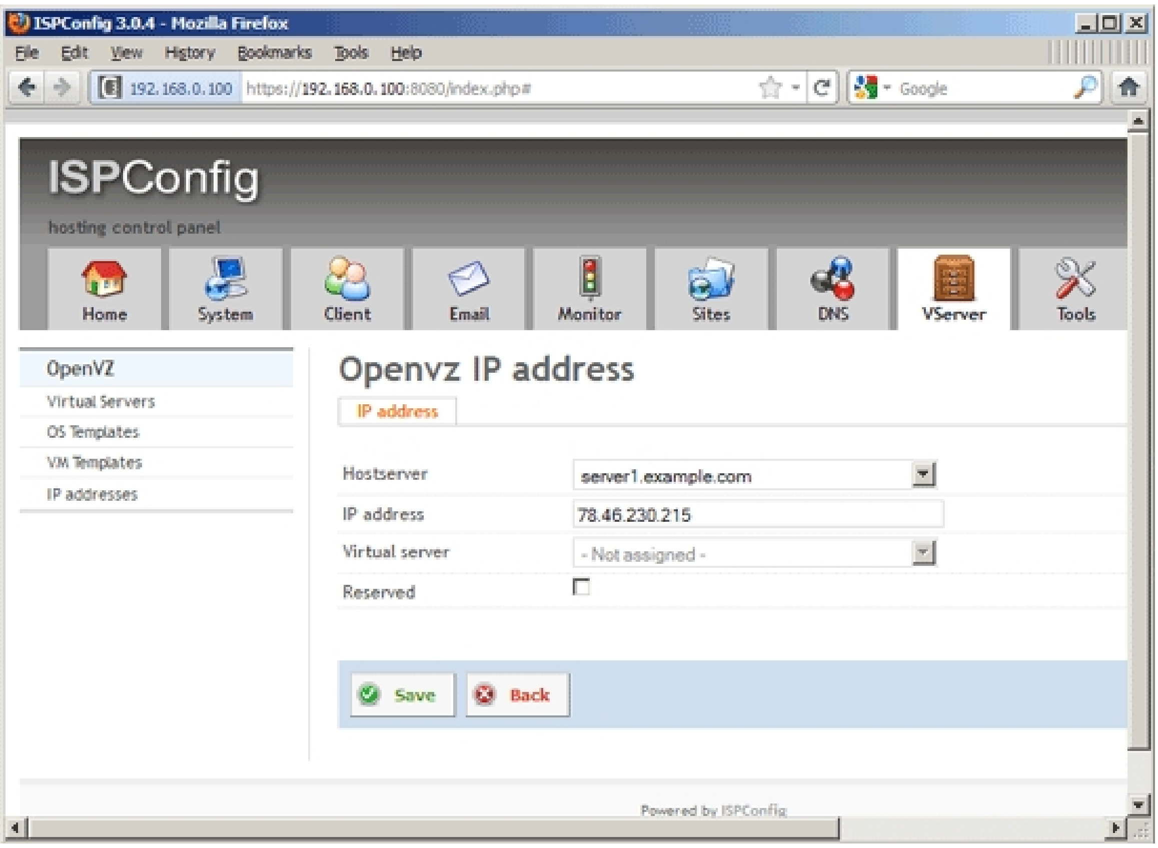
Openvz IP address

IP address

The form has the following fields:

- *Hostserver*: Select the OpenVZ host on which this IP address is located. This field lists only the ISPConfig servers for which you have checked the *VServer-Server* checkbox under *System > System > Server Services* (see chapter ***4.9.2.1 Server Services***).
- *IP address*: Type in the IP address.

- `Virtual server`: This field is readonly. Later on when you create a new virtual machine and allocate this IP address to the virtual machine, ISPConfig will fill in the virtual machine name here.
- `Reserved`: If you check this box, the IP address cannot be allocated to any virtual machine inside ISPConfig. This is useful if you want to allocate this IP address manually, e.g. on the command line.

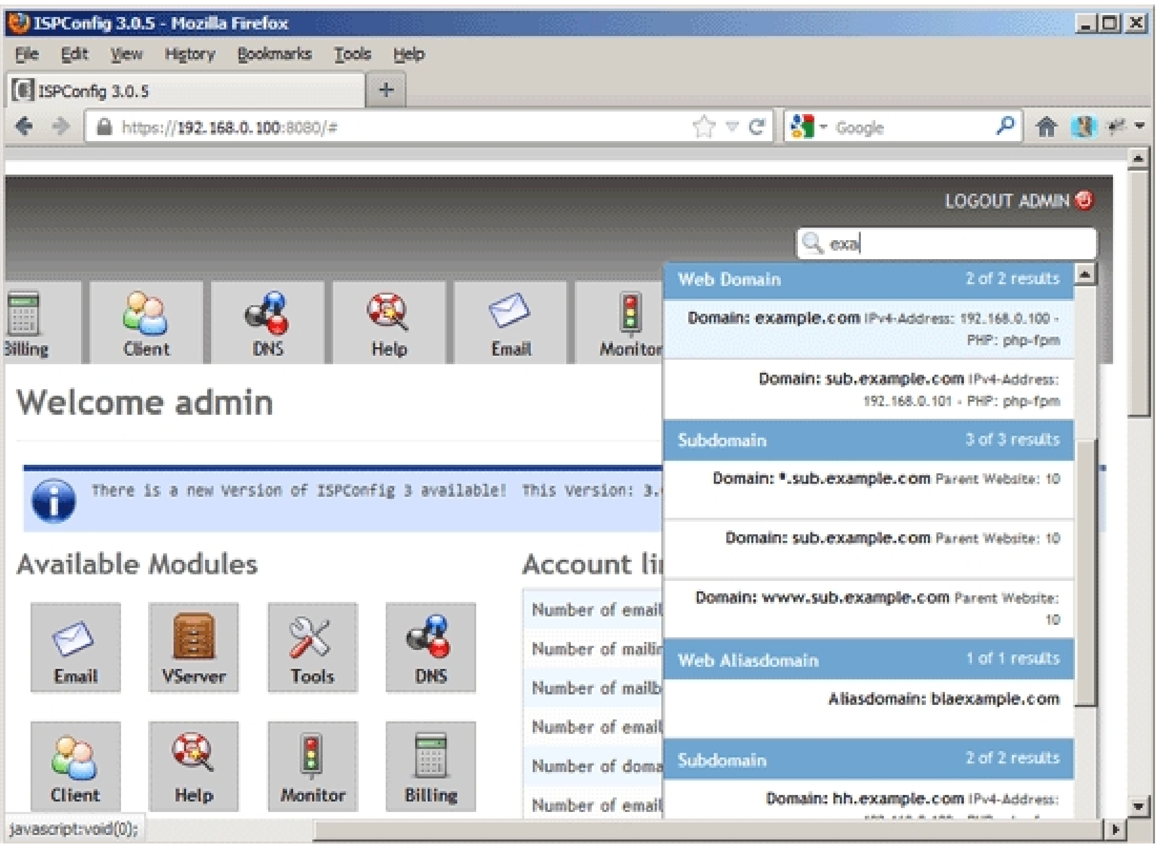


Openvz IP address

Hostserver	server1.example.com
IP address	78.46.230.215
Virtual server	- Not assigned -
Reserved	□

4.14 Global Search

The global search is not a module, but for completeness it is listed here. Global search is a new feature in ISPConfig 3.0.5. It is an AJAX powered search function - you have to type in at least two characters, and ISPConfig will try to find matching items for the search string in all modules (clients, resellers, web sites, subdomains, alias domains, databases, email accounts, etc.). When you click on a search result, ISPConfig will redirect you to the appropriate form where you can modify the item.



Welcome admin

There is a new Version of ISPConfig 3 available! This Version: 3.

Number of email
Number of mailin
Number of mailb
Number of email
Number of doma
Number of email

Web Domain 2 of 2 results

Domain: example.com IPv4-Address: 192.168.0.100 - PHP: php-fpm

Domain: sub.example.com IPv4-Address: 192.168.0.101 - PHP: php-fpm

Subdomain 3 of 3 results

Domain: *.sub.example.com Parent Website: 10

Domain: sub.example.com Parent Website: 10

Domain: www.sub.example.com Parent Website: 10

Web Aliasdomain 1 of 1 results

Aliasdomain: blaexample.com

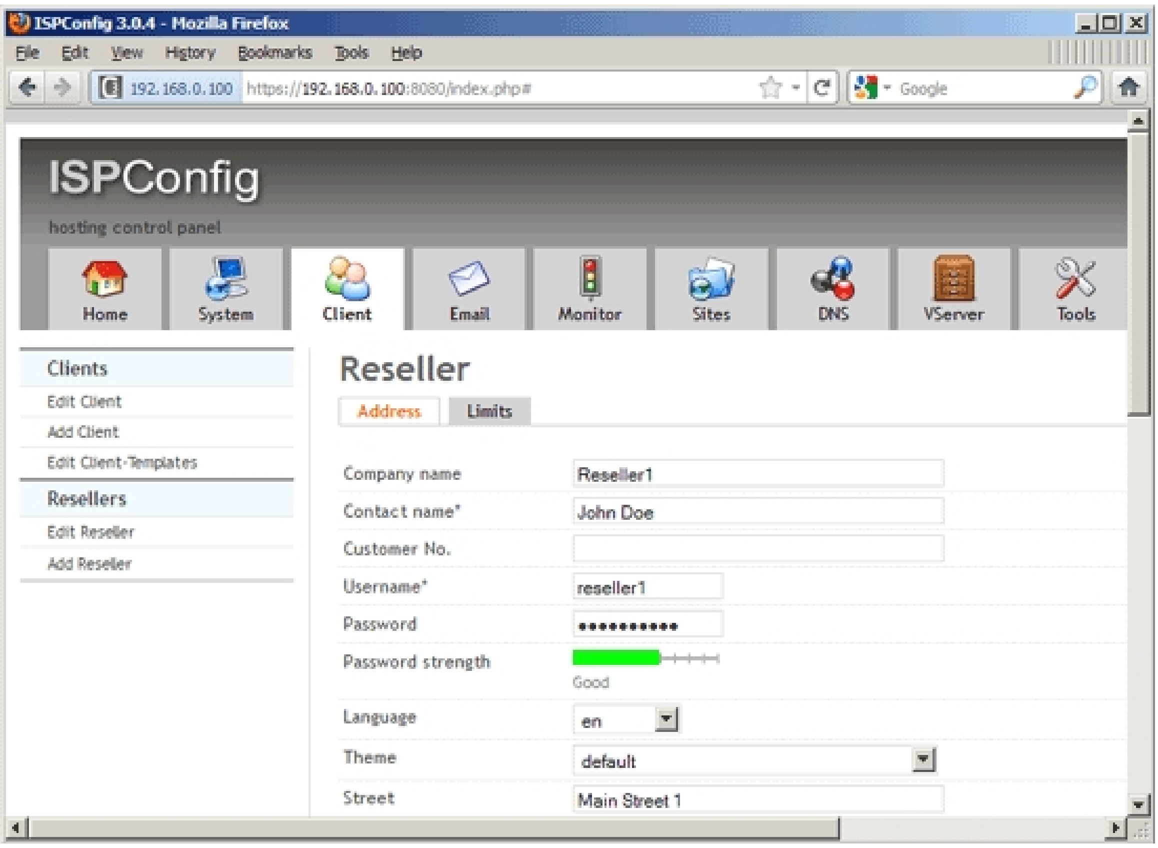
Subdomain 2 of 2 results

Domain: hh.example.com IPv4-Address:

5 Howtos

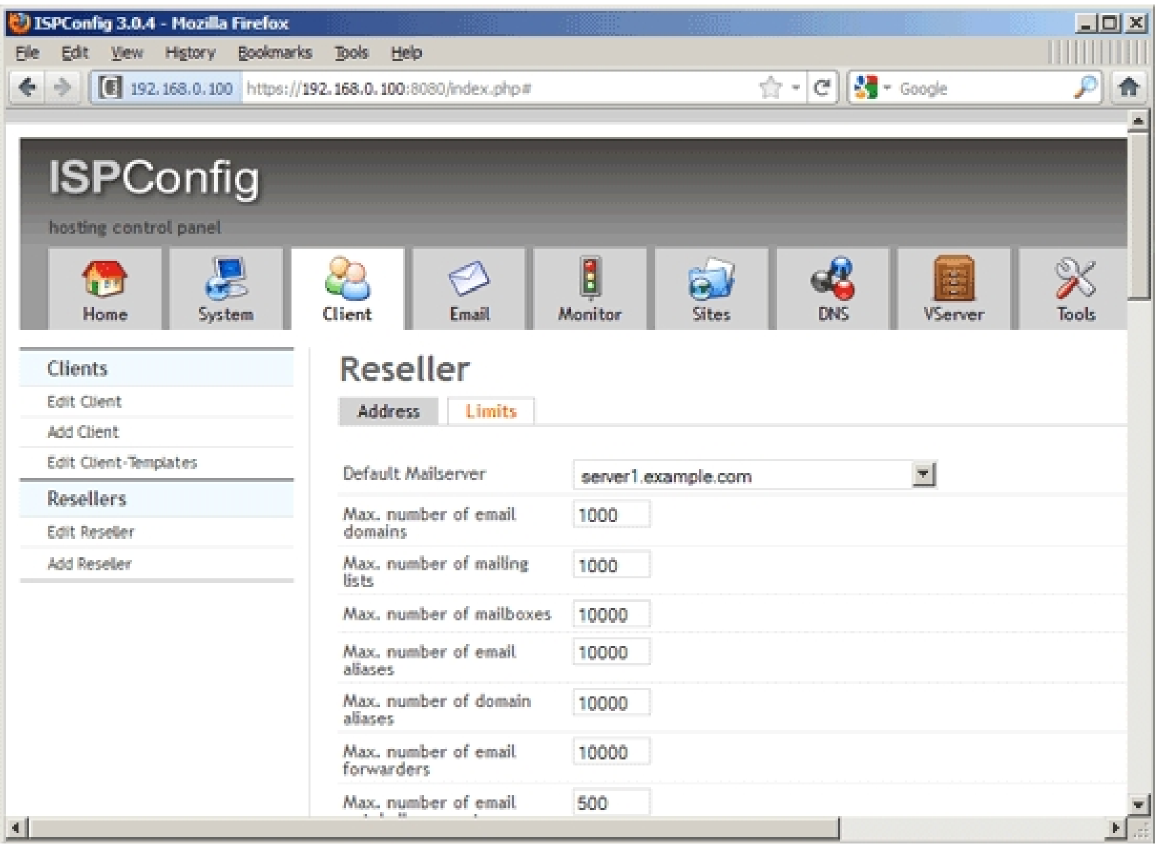
5.1 How Do I Create A Reseller?

Log in as admin and go to `Client > Resellers > Add Reseller` (see chapter 4.5.2.1 Add Reseller). Fill in the address of the reseller on the `Address` tab...



Company name	Reseller1
Contact name*	John Doe
Customer No.	
Username*	reseller1
Password	●●●●●●●●●●●
Password strength	Good
Language	en
Theme	default
Street	Main Street 1

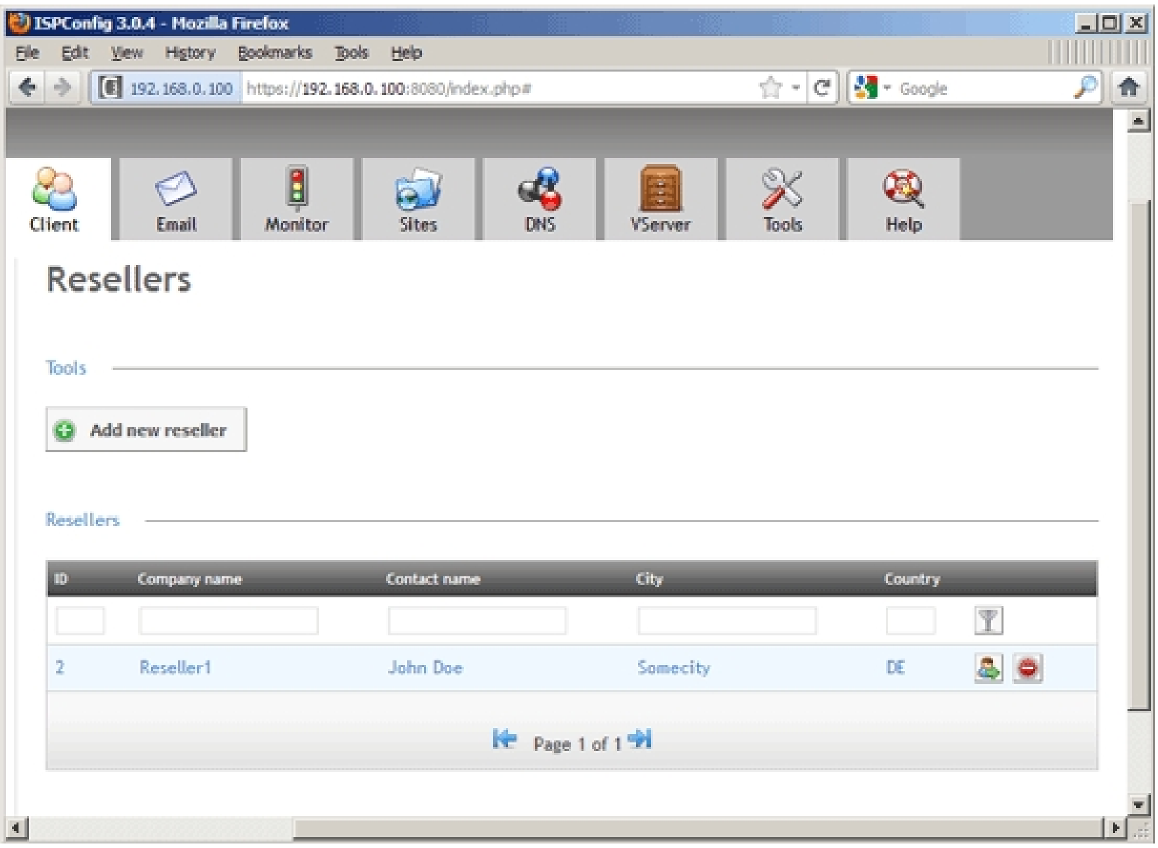
... and then go to the `Limits` tab to specify limits for the reseller. An important field is the `Max. number of Clients` field as it specifies how many clients the reseller can create.



Reseller

Default Mailserver	server1.example.com
Max. number of email domains	1000
Max. number of mailing lists	1000
Max. number of mailboxes	10000
Max. number of email aliases	10000
Max. number of domain aliases	10000
Max. number of email forwarders	10000
Max. number of email	500

After you have created the reseller, you can find it in the list under `Client > Resellers > Edit Reseller`:


ISPConfig 3.0.4 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
192.168.0.100 https://192.168.0.100:8080/index.php#
Google
Client
Email
Monitor
Sites
DNS
VServer
Tools
Help
Resellers
Tools
Add new reseller
Resellers
Page 1 of 1

ID	Company name	Contact name	City	Country
2	Reseller1	John Doe	Somecity	DE

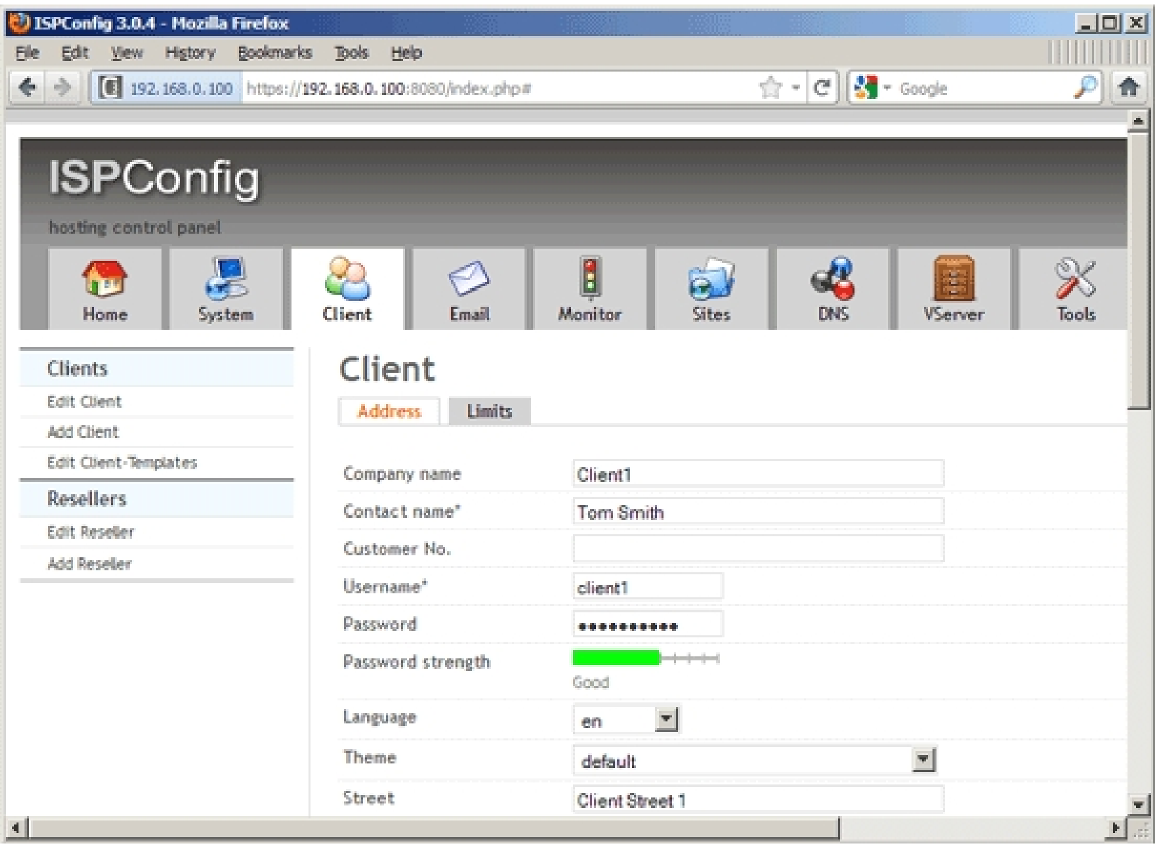
If you want to modify the reseller, you can pick it from that list and change the reseller's settings. From the list view, it is also possible to directly log in as the reseller (just click on the button) and to delete the reseller (click on the button) (see chapter ***4.5.2.2 Edit Reseller***).

5.2 How Do I Create A Client?

Now we have to differentiate between two scenarios: 1) the client belongs to the admin 2) the client belongs to a reseller.

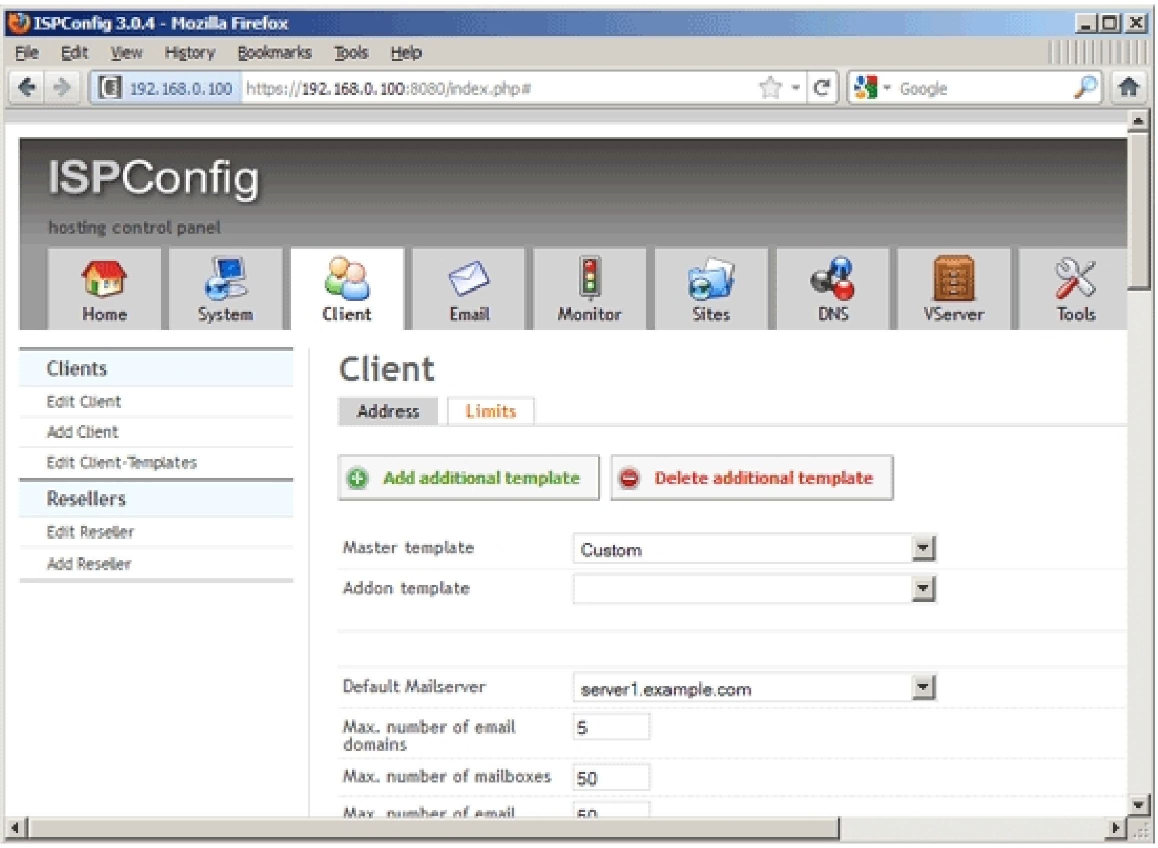
In the first case you must log in as admin and create the client from the admin account, in the second case you must log in as the reseller and create the client from the reseller account.

Then go to *`Client > Clients > Add Client`* (see chapter ***4.5.1.1 Add Client***). Fill in the address of the client on the *`Address`* tab...



Company name	Client1
Contact name*	Tom Smith
Customer No.	
Username*	client1
Password	●●●●●●●●●●●
Password strength	Good
Language	en
Theme	default
Street	Client Street 1

... and then go to the `Limits` tab to specify limits for the client:



Master template: Custom

Addon template:

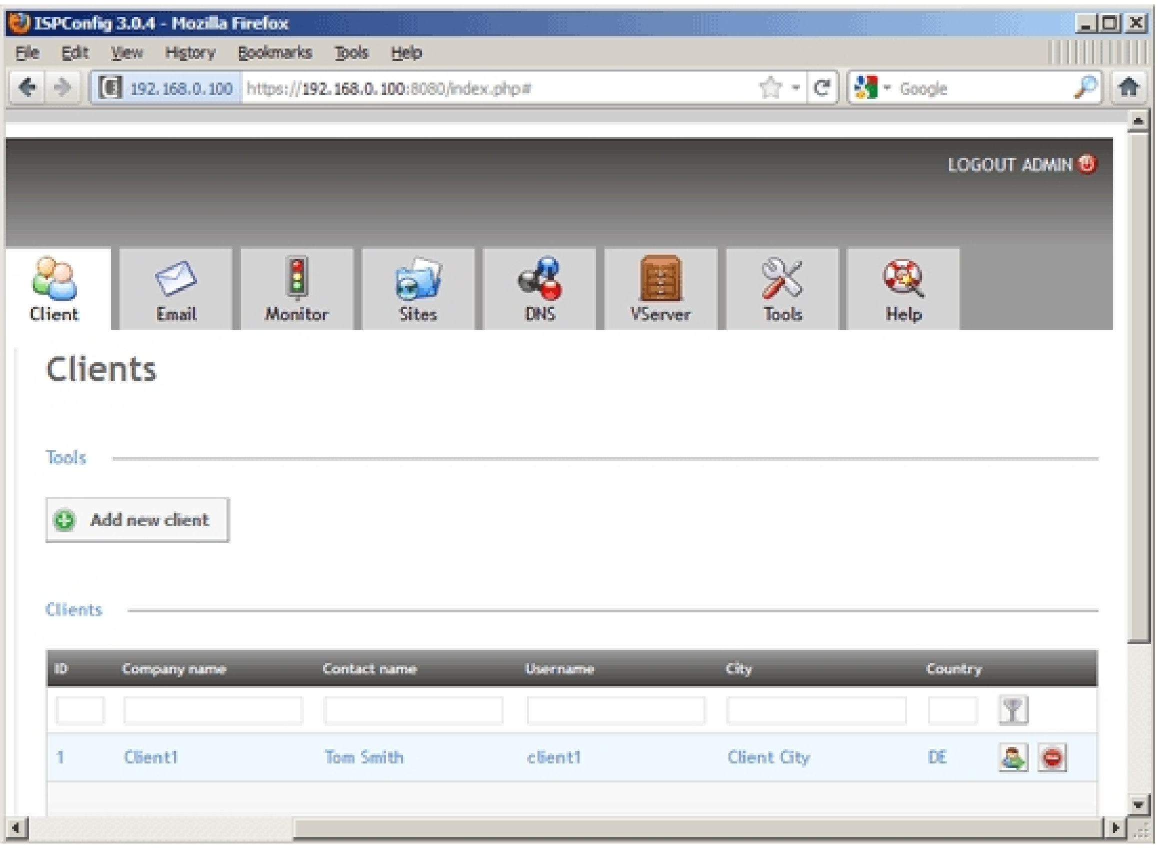
Default Mailserver: server1.example.com

Max. number of email domains: 5

Max. number of mailboxes: 50

Max. number of email: 50

After you have created the client, you can find it in the list under `Client > Clients > Edit Client`:


ISPConfig 3.0.4 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
192.168.0.100 https://192.168.0.100:8080/index.php#
Google
LOGOUT ADMIN
Client
Email
Monitor
Sites
DNS
VServer
Tools
Help
Clients
Tools
Add new client
Clients

ID	Company name	Contact name	Username	City	Country
1	Client1	Tom Smith	client1	Client City	DE

If you want to modify the client, you can pick it from that list and change the client's settings. From the list view, it is also possible to directly log in as the client (just click on the

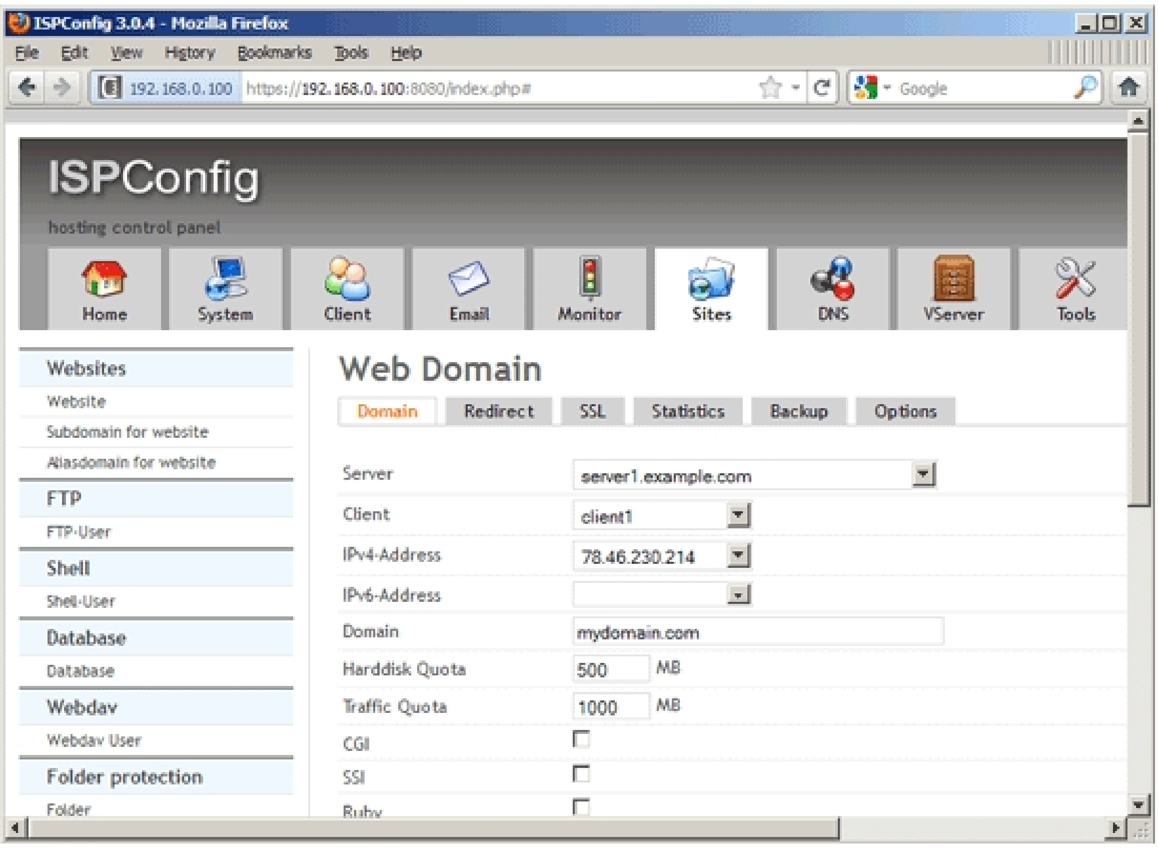
button) and to delete the client (click on the

button) (see chapter ***4.5.1.2 Edit Client***).

5.3 How Do I Create A Web Site?

It is important that you create a client first before you create a web site, so that you can assign the web site to that client (a client can own multiple web sites).

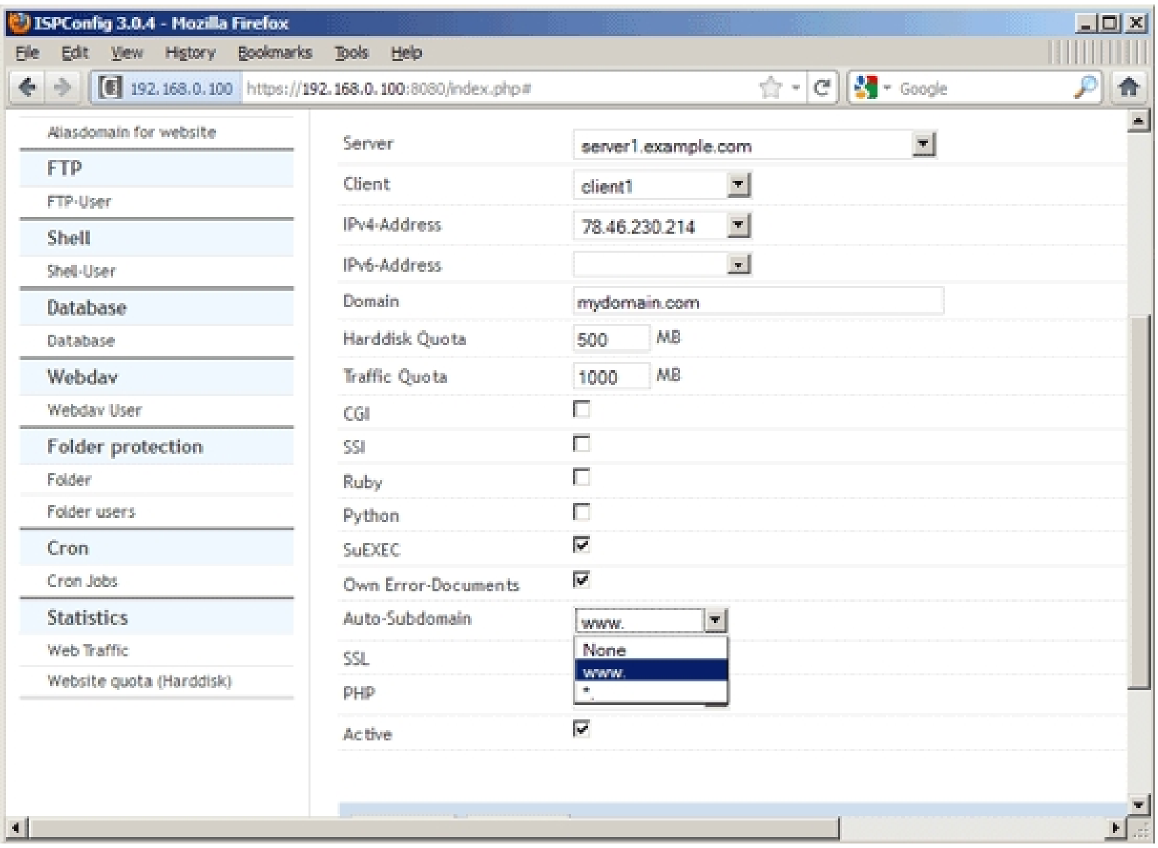
Then log in as admin or as the reseller to which that client belongs and go to *`Sites > Websites > Website`* (see chapter ***4.6.1.1 Website***). To create a web site, you just need to fill out the *`Domain`* tab (the other tabs contain special configurations that you usually don't need). Make sure that you select the correct client in the *`Client`* drop-down menu (if you are logged in as admin, you can select all clients that exist on the system; if you are logged in as a reseller, you can select only the clients that belong to the reseller):



Web Domain

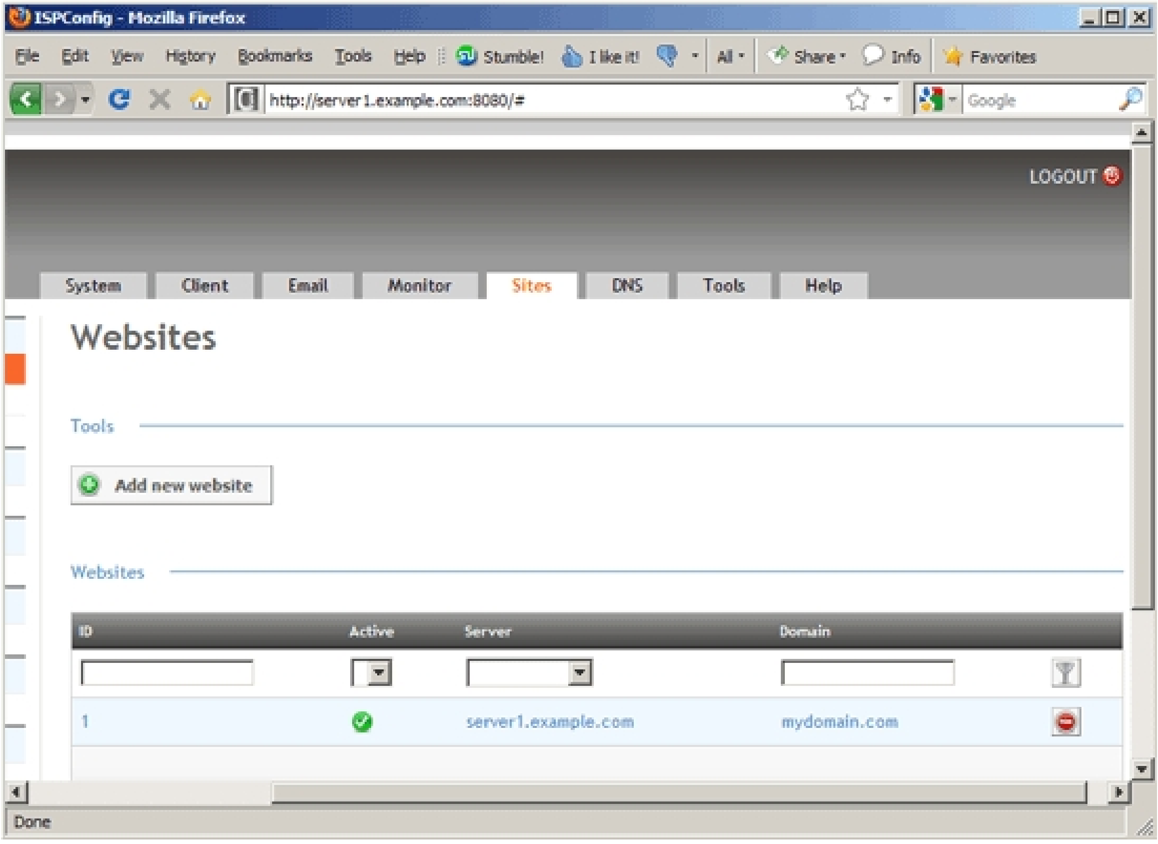
Server	server1.example.com
Client	client1
IPv4-Address	78.46.230.214
IPv6-Address	
Domain	mydomain.com
Harddisk Quota	500 MB
Traffic Quota	1000 MB
CGI	□
SSI	□
Ruby	□

Use the `Auto-Subdomain` field to define whether you want no automatic subdomain for the web site (in this case you can access the site only by using the domain, e.g. *`http://example.com`*), an automatic *`www`* subdomain (recommended) (you can then access the site using *`http://example.com`* and *`http://www.example.com`*), or a wildcard subdomain (*`*.`*) which means you can access the site with any subdomain that does not point to another web site:



Server: server1.example.com
Client: client1
IPv4-Address: 78.46.230.214
IPv6-Address:
Domain: mydomain.com
Harddisk Quota: 500 MB
Traffic Quota: 1000 MB
CGI: □
SSI: □
Ruby: □
Python: □
SuEXEC: ☑
Own Error-Documents: ☑
Auto-Subdomain: www. (None / www. / *.)
SSL:
PHP:
Active: ☑

After you have created the web site, you can find it in the list under *`Sites > Websites > Website`*:

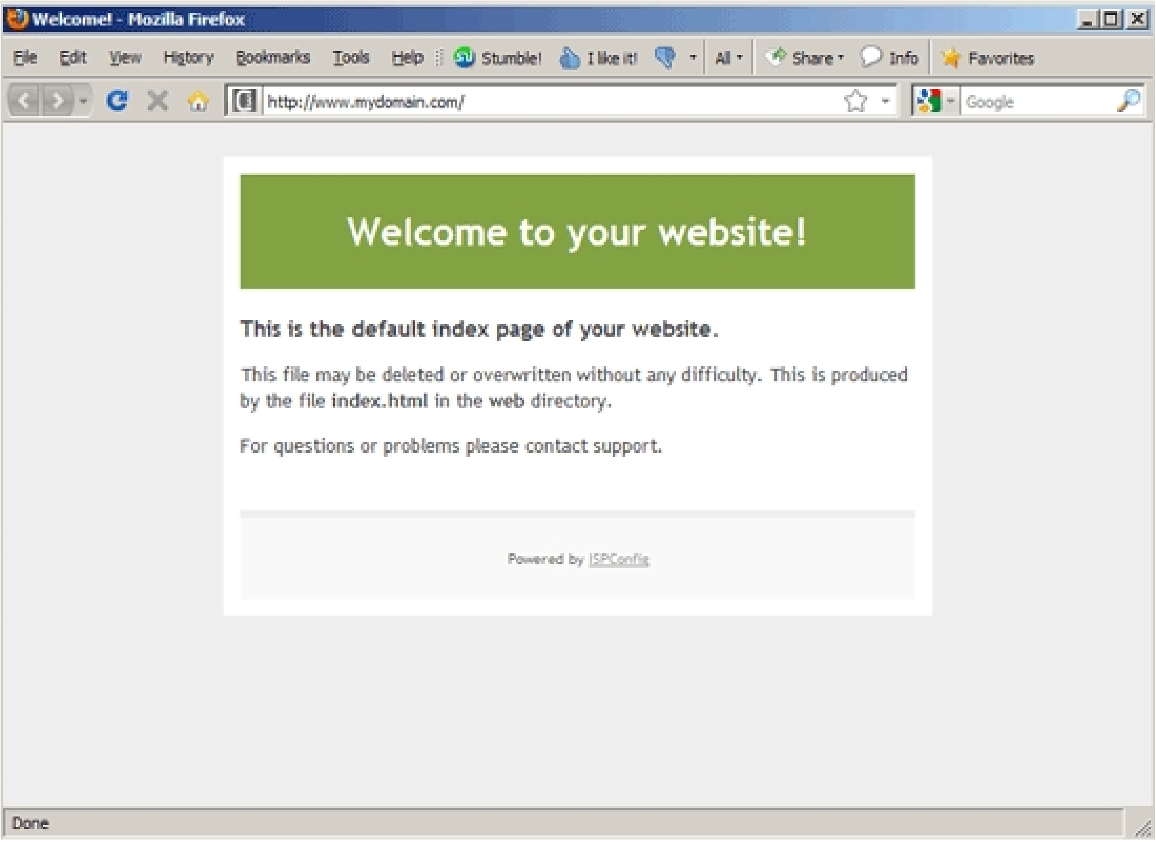

ISPConfig - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/#
Google
LOGOUT
System Client Email Monitor Sites DNS Tools Help
Tools
Add new website
Done

Websites

ID	Active	Server	Domain	
1	✓	server1.example.com	mydomain.com	⊖

From the list view, it is possible to delete the web site (click on the ⊖ button).

If the DNS records for the new web site exist and point to the correct server, you can now go to the new web site in a browser, and you should see the default ISPConfig 3 welcome page:



Welcome to your website!

This is the default index page of your website.

This file may be deleted or overwritten without any difficulty. This is produced by the file **index.html** in the web directory.

For questions or problems please contact support.

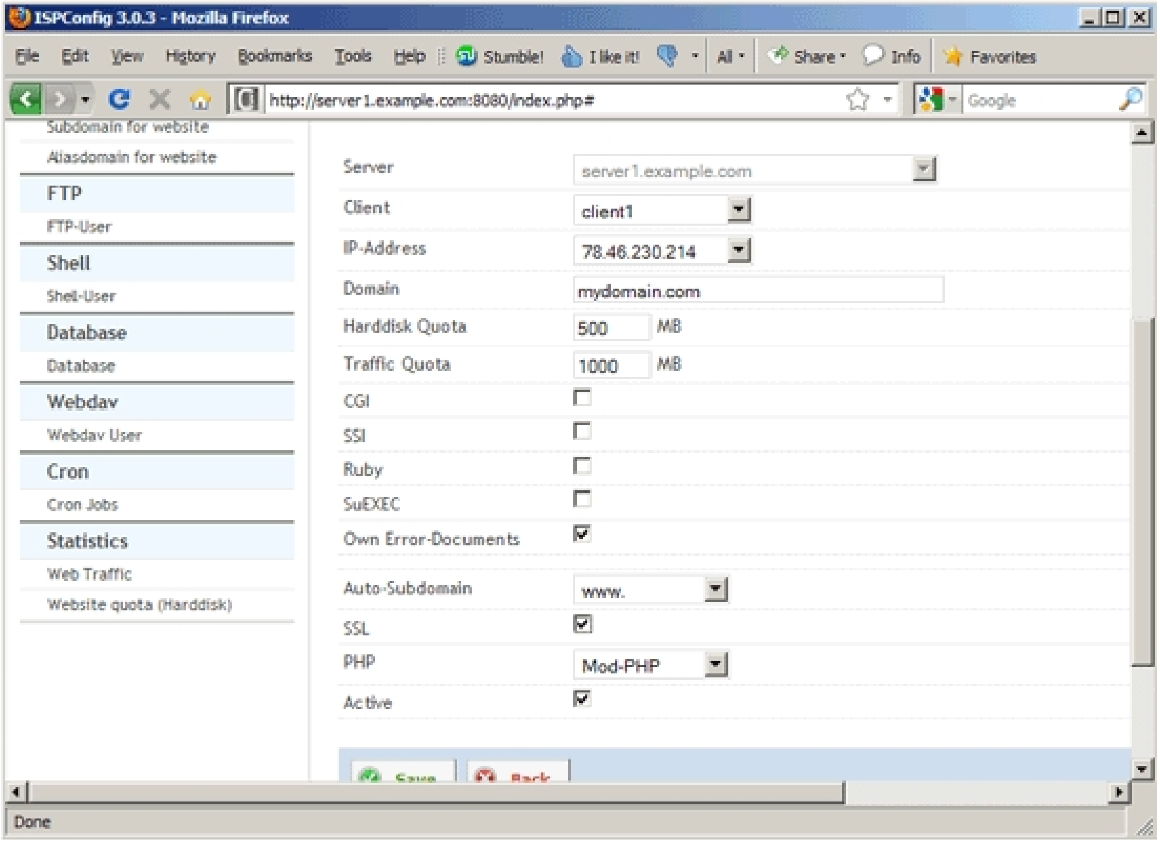
Powered by ISPConfig

Welcome! - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://www.mydomain.com/
Google
Done

Important: if a client creates a web site himself, he has the permissions to modify the web site settings in ISPConfig. If the admin or a reseller creates a web site for a client, then the web site settings cannot be modified by the client in ISPConfig, only by the admin or by the reseller that created the web site.

5.4 How Do I Create An SSL Web Site?

To make a web site SSL-capable, please make sure that the SSL checkbox is checked on the web site's *Domain* tab (please note that you can have only one SSL web site per IP address). Important: you must select a specific IP address from the *IP-Address* drop-down menu; you must not select the wildcard (*)!



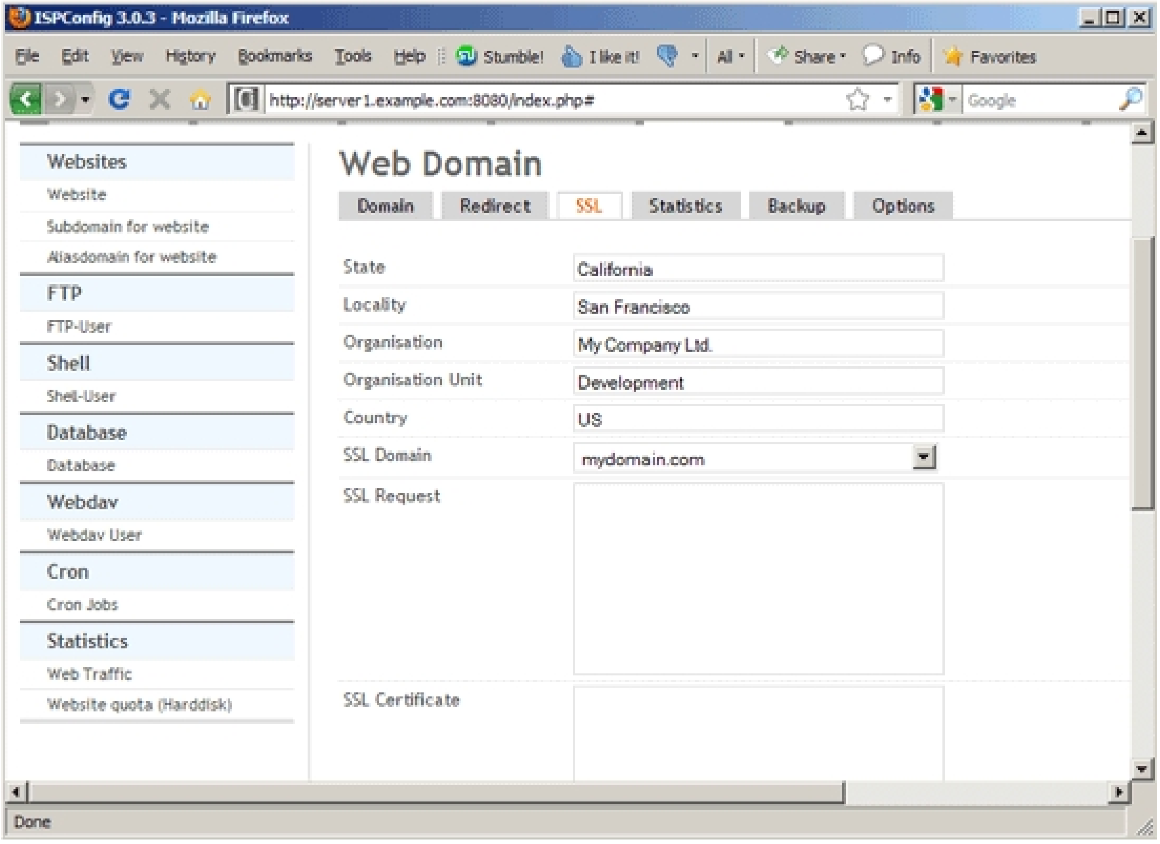
Server	server1.example.com
Client	client1
IP-Address	78.46.230.214
Domain	mydomain.com
Harddisk Quota	500 MB
Traffic Quota	1000 MB
CGI	□
SSI	□
Ruby	□
SuEXEC	□
Own Error-Documents	☑
Auto-Subdomain	www.
SSL	☑
PHP	Mod-PHP
Active	☑

Then go to the *SSL* tab (see chapter ***4.6.1.1 Website***).

On the *SSL* tab you can create a self-signed SSL certificate together with a certificate signing request (CSR) that you can use to apply for an SSL certificate that is signed by a trusted certificate authority (CA) such as Verisign, Comodo, Thawte, etc. It's not necessary to buy such a trusted SSL certificate, but you should note that if you use a self-signed SSL certificate, browsers will display a warning to your visitors.

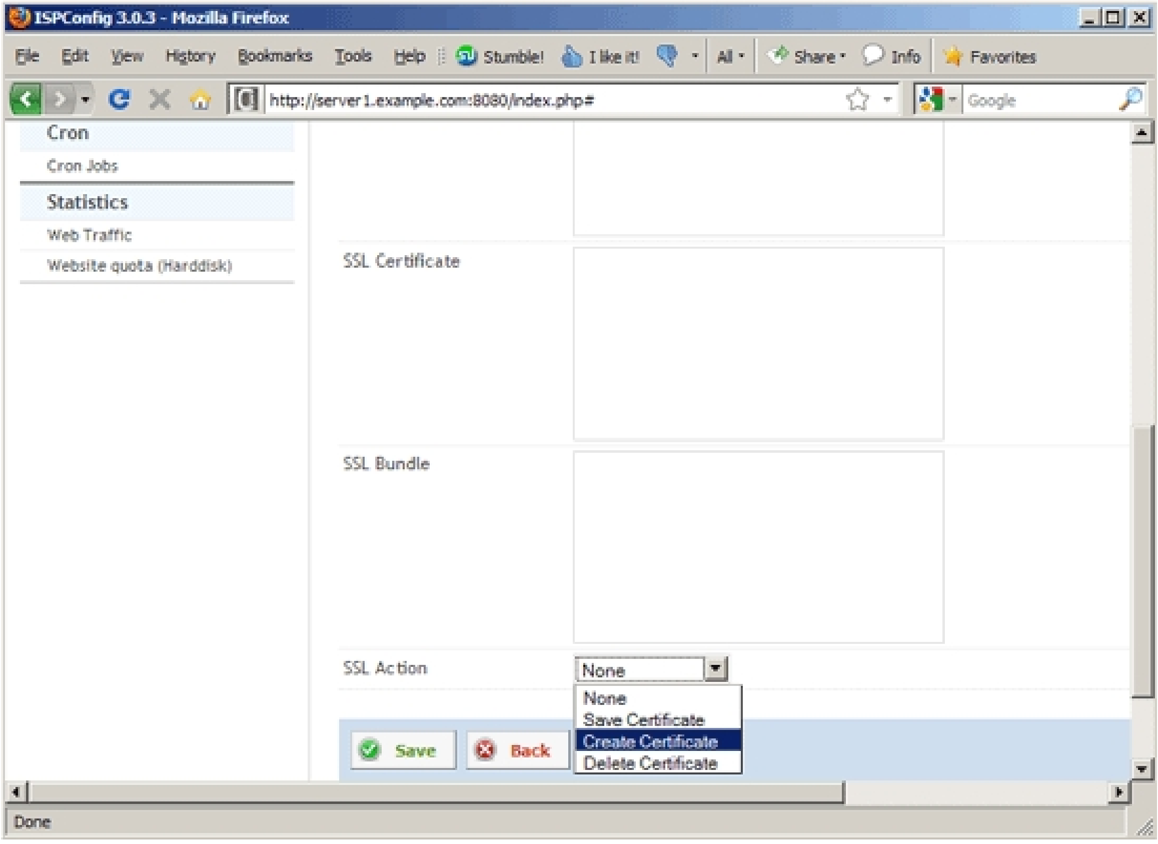
Please note that you can have just one SSL web site per IP address.

To create a self-signed certificate, please fill out the fields *State*, *Locality*, *Organisation*, *Organisation Unit*, *Country*, and *SSL Domain*, and then select *Create Certificate* from the *SSL Action* drop-down menu, and click on *Save*. Leave the fields *SSL Request*, *SSL Certificate*, and *SSL Bundle* empty - the fields *SSL Request* and *SSL Certificate* will be filled out by the system.



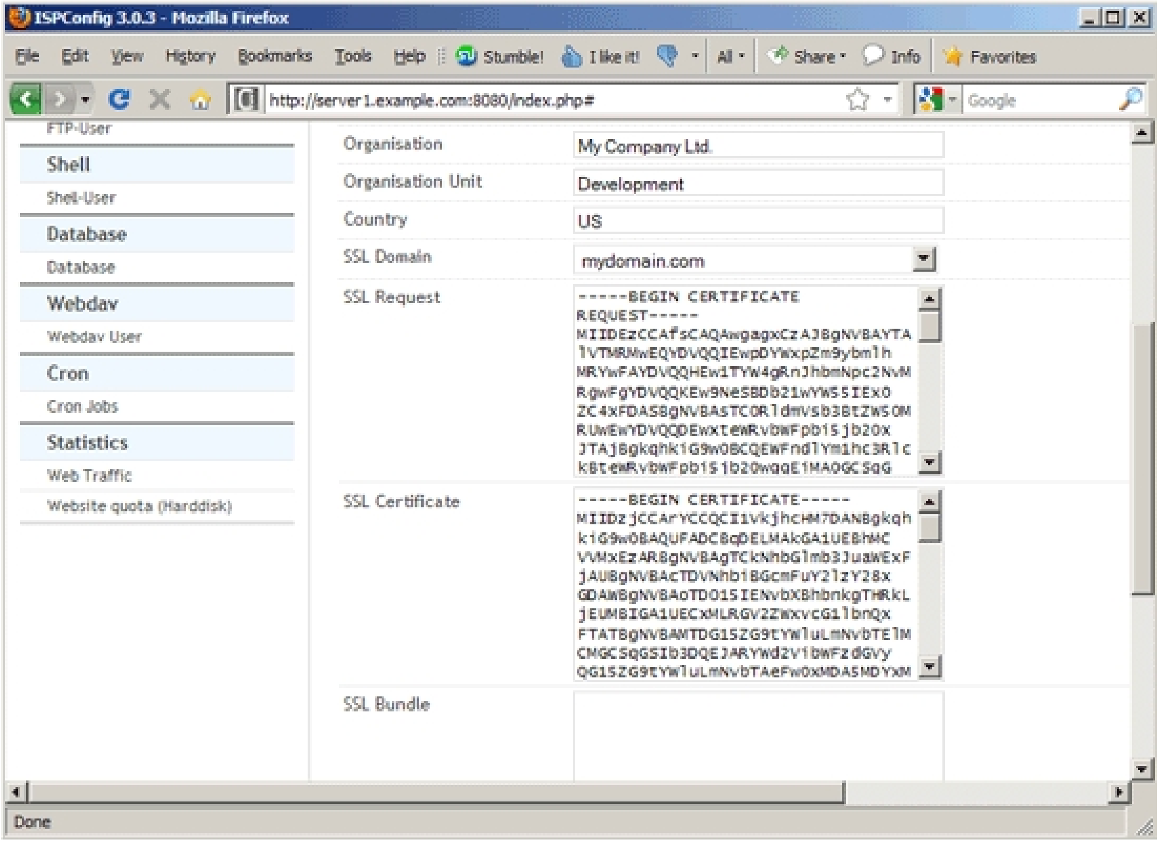
Web Domain

State	California
Locality	San Francisco
Organisation	My Company Ltd.
Organisation Unit	Development
Country	US
SSL Domain	mydomain.com
SSL Request	
SSL Certificate	


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
Cron
Cron Jobs
Statistics
Web Traffic
Website quota (Harddisk)
None
Save Certificate
Create Certificate
Delete Certificate
Save
Back
Done

SSL Certificate	
SSL Bundle	
SSL Action	None

After the self-signed certificate was created, you will find data in the `SSL Request` and `SSL Certificate` fields (it can take one or two minutes until the data appears in the fields):


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
FTP-User
Shell
Shell-User
Database
Database
Webdav
Webdav User
Cron
Cron Jobs
Statistics
Web Traffic
Website quota (Harddisk)
Done

Organisation	My Company Ltd.
Organisation Unit	Development
Country	US
SSL Domain	mydomain.com

SSL Request

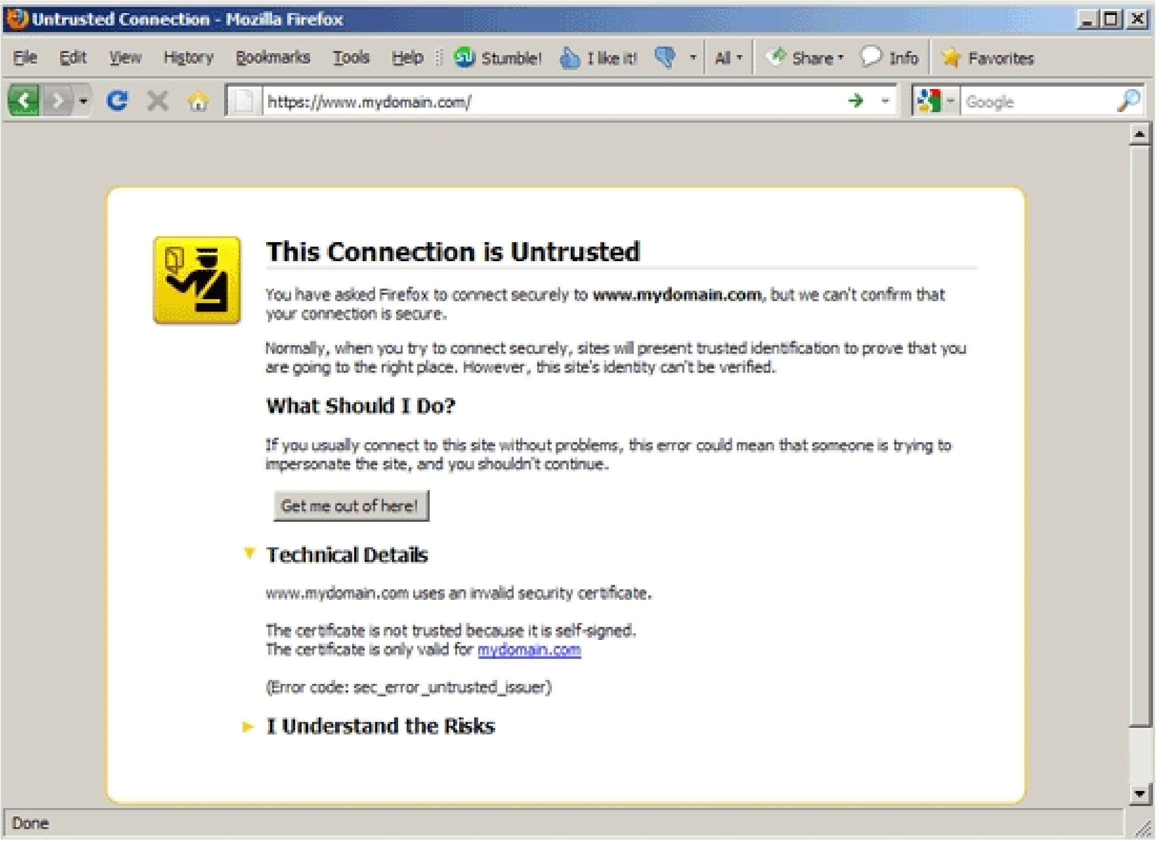
```
-----BEGIN CERTIFICATE
REQUEST-----
MIIDEzCCAfsCAQAwgagxCzAJBgNVBAYTA
lVTMRMwEQYDVQQIEwpDYWxpZm9ybmlh
MRYwFAYDVQQHEw1TYW4gRnJhbmNpc2NvM
RgwFgYDVQQKEw9NeSBDb21wYW55IEx0
ZC4xFDASBgNVBAsTC0RldmVsb3BtZW50M
RUwEwYDVQQDEwxteWRvbWFpbi5jb20x
JTAjBgkqhkiG9w0BCQEWFndlYmlhc3Rlc
kBteWRvbWFpbi5jb20wggEiMA0GCSqG
```

SSL Certificate

```
-----BEGIN CERTIFICATE-----
MIIDzjCCArYCCQCI1VkjhCHM7DANBgkqh
kiG9w0BAQUFADCBqDELMAkGA1UEBhMC
VVMxEzARBgNVBAgTCkNhbGlmb3JuaWExF
jAUBgNVBAcTDVNhbiBGcmFuY2lzY28x
GDAWBgNVBAoTD015IENvbXBhbnkgTHRkL
jEUMBIGA1UECxMLRGV2ZWxvcG1lbnQx
FTATBgNVBAMTDG15ZG9tYWluLmNvbTElM
CMGCSqGSIb3DQEJARYWd2VibWFzdGVy
QG15ZG9tYWluLmNvbTAeFw0xMDA5MDYxM
```

SSL Bundle

It is already possible to access the web site using `https://` now with the self-signed certificate, but your visitors will see a warning. For example, Firefox will complain about the self-signed certificate, therefore you must tell Firefox to accept the certificate - to do this, click on the `I Understand the Risks` link:


Untrusted Connection - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
https://www.mydomain.com/
Google
Get me out of here!
Done

This Connection is Untrusted

You have asked Firefox to connect securely to **www.mydomain.com**, but we can't confirm that your connection is secure.

Normally, when you try to connect securely, sites will present trusted identification to prove that you are going to the right place. However, this site's identity can't be verified.

What Should I Do?

If you usually connect to this site without problems, this error could mean that someone is trying to impersonate the site, and you shouldn't continue.

Technical Details

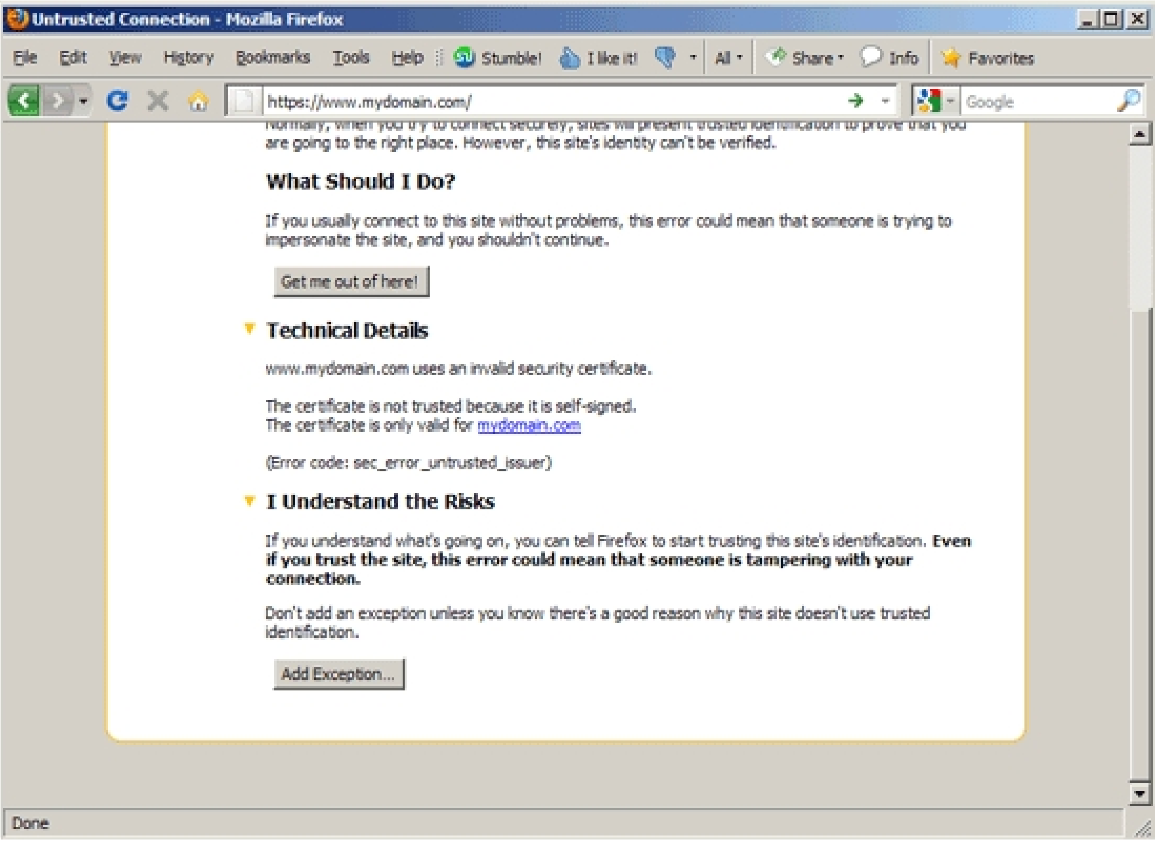
www.mydomain.com uses an invalid security certificate.

The certificate is not trusted because it is self-signed.
The certificate is only valid for mydomain.com

(Error code: sec_error_untrusted_issuer)

I Understand the Risks

Click on *`Add Exception...`*:


Untrusted Connection - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
https://www.mydomain.com/
Google
Get me out of here!
Add Exception...
Done

Normally, when you try to connect securely, sites will present trusted identification to prove that you are going to the right place. However, this site's identity can't be verified.

What Should I Do?

If you usually connect to this site without problems, this error could mean that someone is trying to impersonate the site, and you shouldn't continue.

Technical Details

www.mydomain.com uses an invalid security certificate.

The certificate is not trusted because it is self-signed.
The certificate is only valid for mydomain.com

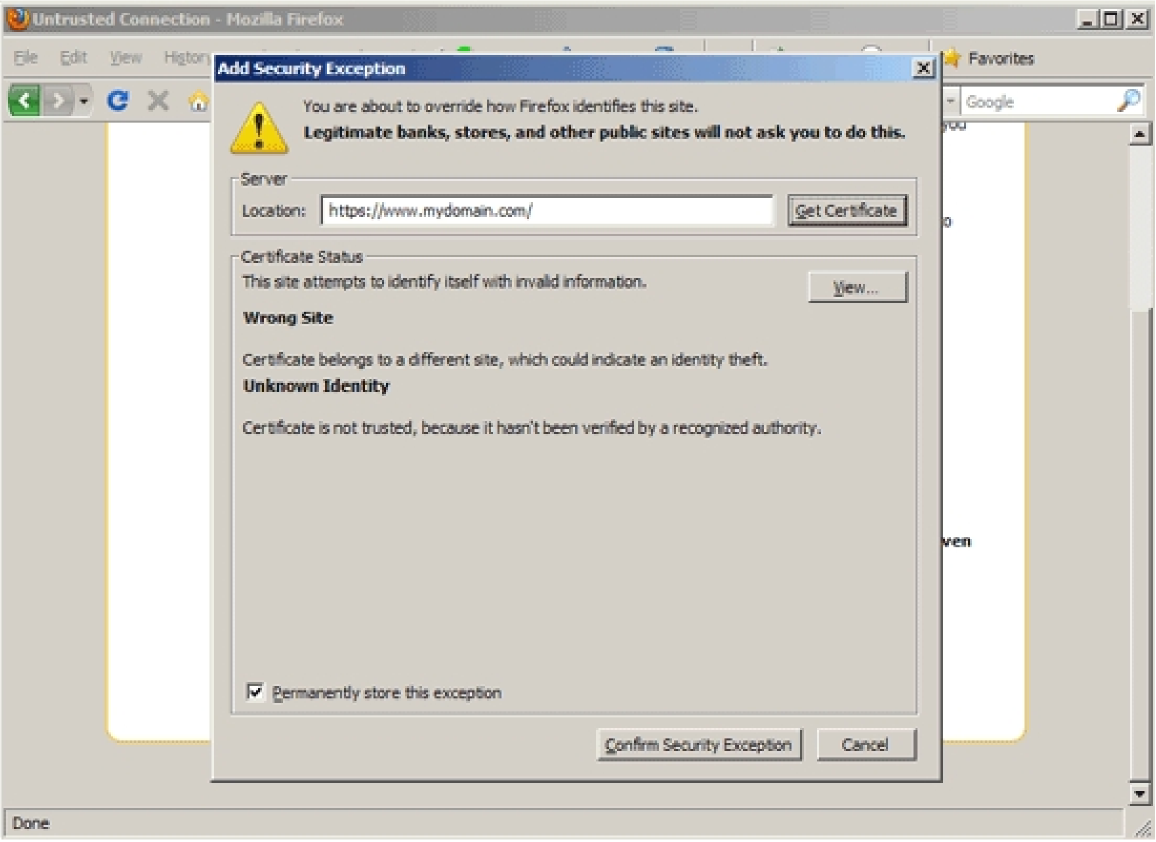
(Error code: sec_error_untrusted_issuer)

I Understand the Risks

If you understand what's going on, you can tell Firefox to start trusting this site's identification. **Even if you trust the site, this error could mean that someone is tampering with your connection.**

Don't add an exception unless you know there's a good reason why this site doesn't use trusted identification.

The *Add Security Exception* window opens. In that window, click on the *Get Certificate* button first and then on the *Confirm Security Exception* button:



You are about to override how Firefox identifies this site.

Legitimate banks, stores, and other public sites will not ask you to do this.

Server

Location: https://www.mydomain.com/

Certificate Status

This site attempts to identify itself with invalid information.

Wrong Site

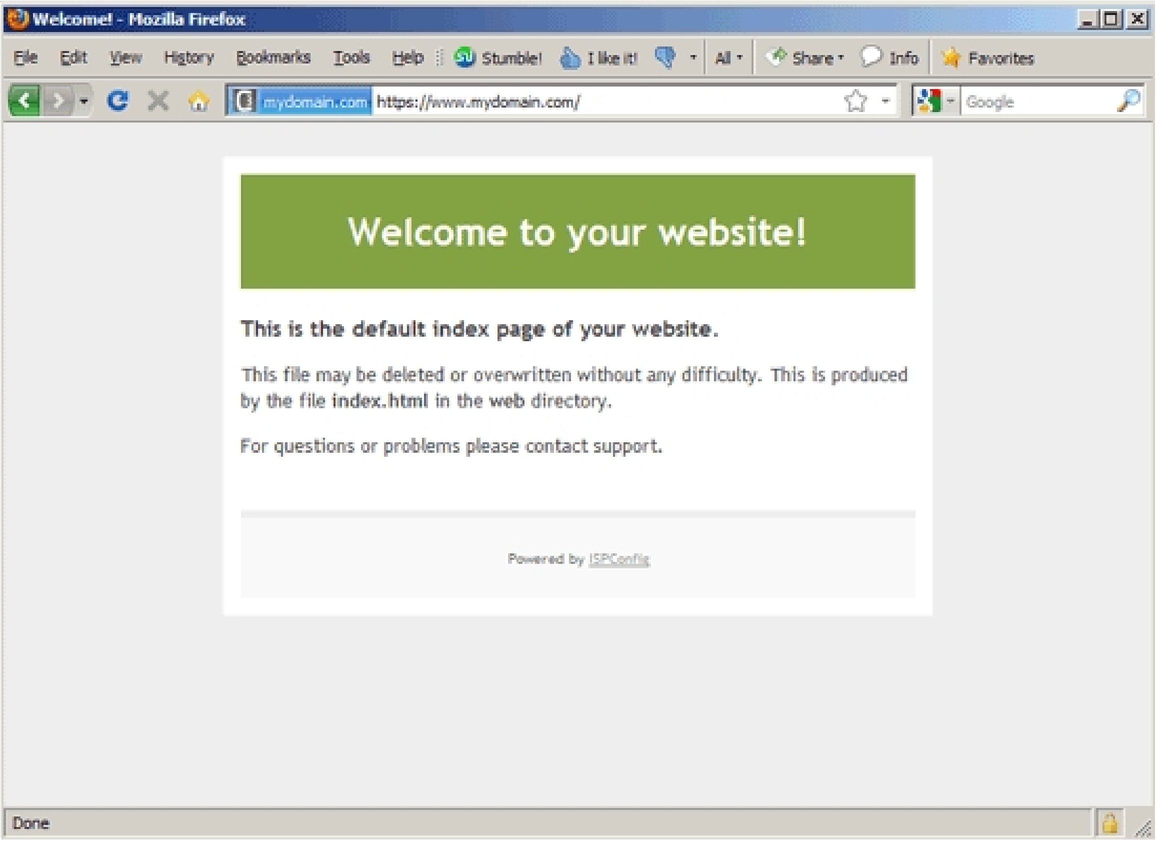
Certificate belongs to a different site, which could indicate an identity theft.

Unknown Identity

Certificate is not trusted, because it hasn't been verified by a recognized authority.

☑ Permanently store this exception

Afterwards you should be able to see the `https://` web site:


Welcome! - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
mydomain.com https://www.mydomain.com/
Google
Done

Welcome to your website!

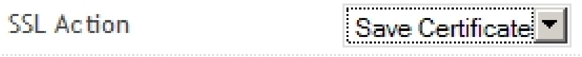
This is the default index page of your website.

This file may be deleted or overwritten without any difficulty. This is produced by the file index.html in the web directory.

For questions or problems please contact support.

Powered by ISPConfig

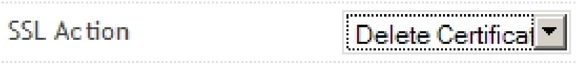
If you want to buy an SSL certificate from a trusted CA, you have to copy the data from the *SSL Request* field - this is the certificate signing request (CSR). With this CSR, you can apply for a trusted SSL certificate at your CA - the CA will create an SSL certificate from this CSR, and you can paste the trusted SSL certificate into the *SSL Certificate* field. Sometimes your CA will also give you an SSL bundle - paste this into the *SSL Bundle* field. Select *Save Certificate* from the *SSL Action* drop-down menu and click on the *Save* button:



SSL Action: Save Certificate

You have just replaced your self-signed certificate with a trusted SSL certificate.

To delete a certificate, select *Delete Certificate* from the *SSL Action* drop-down menu and click on the *Save* button.



SSL Action: Delete Certificat

5.4.1 How Do I Import An Existing SSL Certificate Into A Web Site That Was Created Later In ISPConfig?

This is very easy with ISPConfig 3.0.5. Just copy your certificate, the key, the bundle certificate (if necessary) and the CSR (optional) into the respective text areas on the SSL tab of the web site in ISPConfig and select `Save Certificate` under `SSL Action` and click on `Save` (see chapter ***4.6.1.1 Website***).

5.5 How Do I Redirect My Web Site To Another Web Site Or To A Specific Directory On The Server?

Go to the `Redirect` tab of your web site in ISPConfig (see chapter ***4.6.1.1 Website***). In the `Redirect Type` field, please select the flag that you want to use for the redirect:

Flags (Apache):

- `R`: Use of the `[R]` flag causes a HTTP redirect to be issued to the browser. If a fully-qualified URL is specified (that is, including `http://servername/`) then a redirect will be issued to that location. Otherwise, the current servername will be used to generate the URL sent with the redirect.
- `L`: The `[L]` flag causes mod_rewrite to stop processing the rule set. In most contexts, this means that if the rule matches, no further rules will be processed.
- `R,L`: You will almost always want to use `[R]` in conjunction with `[L]` (that is, use `[R,L]`) because on its own, the `[R]` flag prepends `http://thishost[:thisport]` to the URI, but then passes this on to the next rule in the ruleset, which can often result in 'Invalid URI in request' warnings.

More details about Apache rewrite flags can be found here: ***http://httpd.apache.org/docs/2.2/rewrite/flags.html***

If you want to do a URL redirect, you should use the R,L flags, while for a directory redirect it is recommended to just use the L flag.

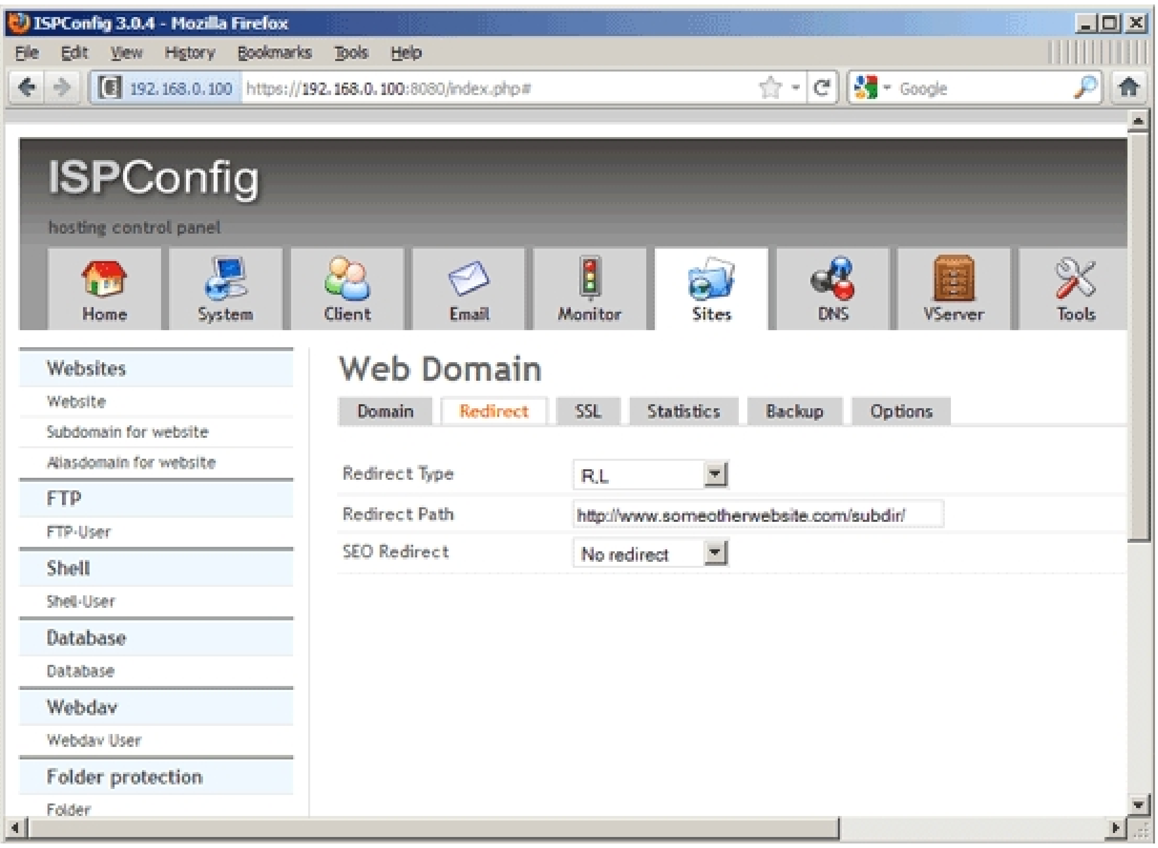
Flags (nginx):

- `last`: Completes processing of rewrite directives, after which searches for corresponding URI and location.
- `break`: Completes processing of rewrite directives and breaks location lookup cycle by not doing any location lookup and internal jump at all.
- `redirect`: Returns temporary redirect with code 302; it is used if the substituting line begins with `http://`.
- `permanent`: Returns permanent redirect with code 301.

More details about Apache rewrite flags can be found here:
http://wiki.nginx.org/NginxHttpRewriteModule#rewrite

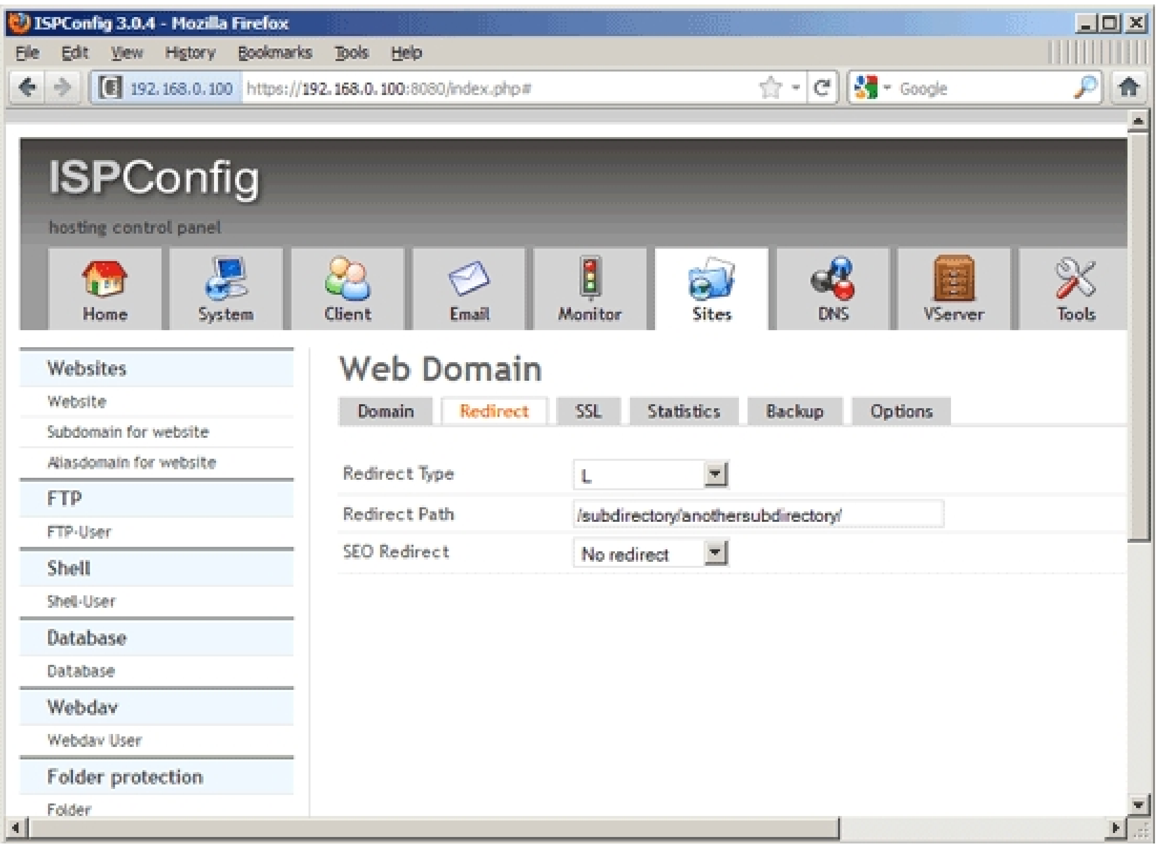
If you want to do a URL redirect, you should use the permanent flag.

If you want to do a URL redirect, please specify the redirect target URL in the Redirect Path field (e.g. `http://www.someotherwebsite.com/subdir/` or `http://www.someotherwebsite.com/`). Please note that the URL should have a trailing slash:



Redirect Type: R,L
Redirect Path: http://www.someotherwebsite.com/subdir/
SEO Redirect: No redirect

If you want to do a redirect to a subdirectory of your web site, please specify the subdirectory or the path to the subdirectory (relative to the document root of your web site) in the `Redirect Path` field. Please note that the path must begin and end with a slash (e.g. `/subdirectory/anothersubdirectory/`):



Web Domain

Redirect Type: L

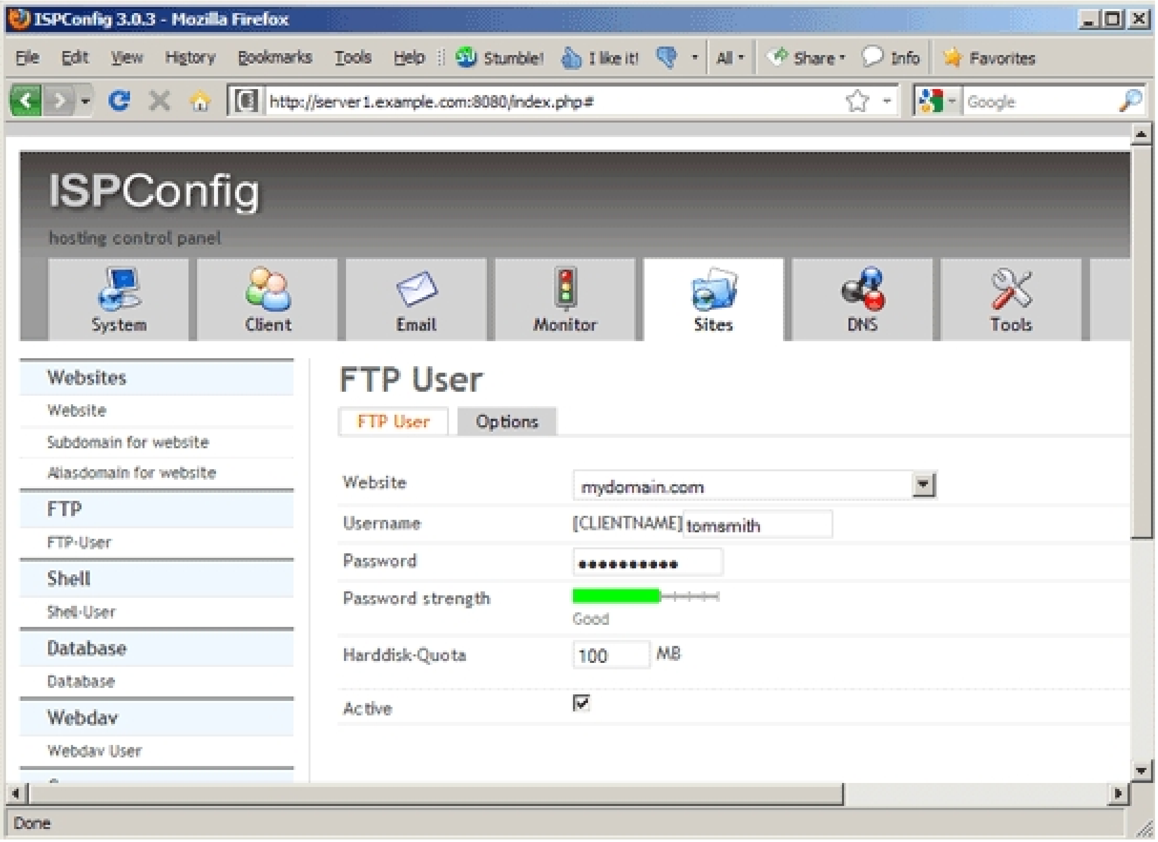
Redirect Path: /subdirectory/anothersubdirectory/

SEO Redirect: No redirect

5.6 How Do I Create An FTP Account So That I Can Upload Files To My Web Site?

Go to `Sites > FTP > FTP-User` and click on the `Add new FTP-User` button (see chapter **4.6.2.1 Databases**).

Select the web site for which you want to create the FTP user, then define a username for the FTP account (`[CLIENTNAME]` is a placeholder and will be replaced by ISPConfig; you can see the final username in the FTP user list) and a password and specify a hard disk quota in MB (`-1` means unlimited):

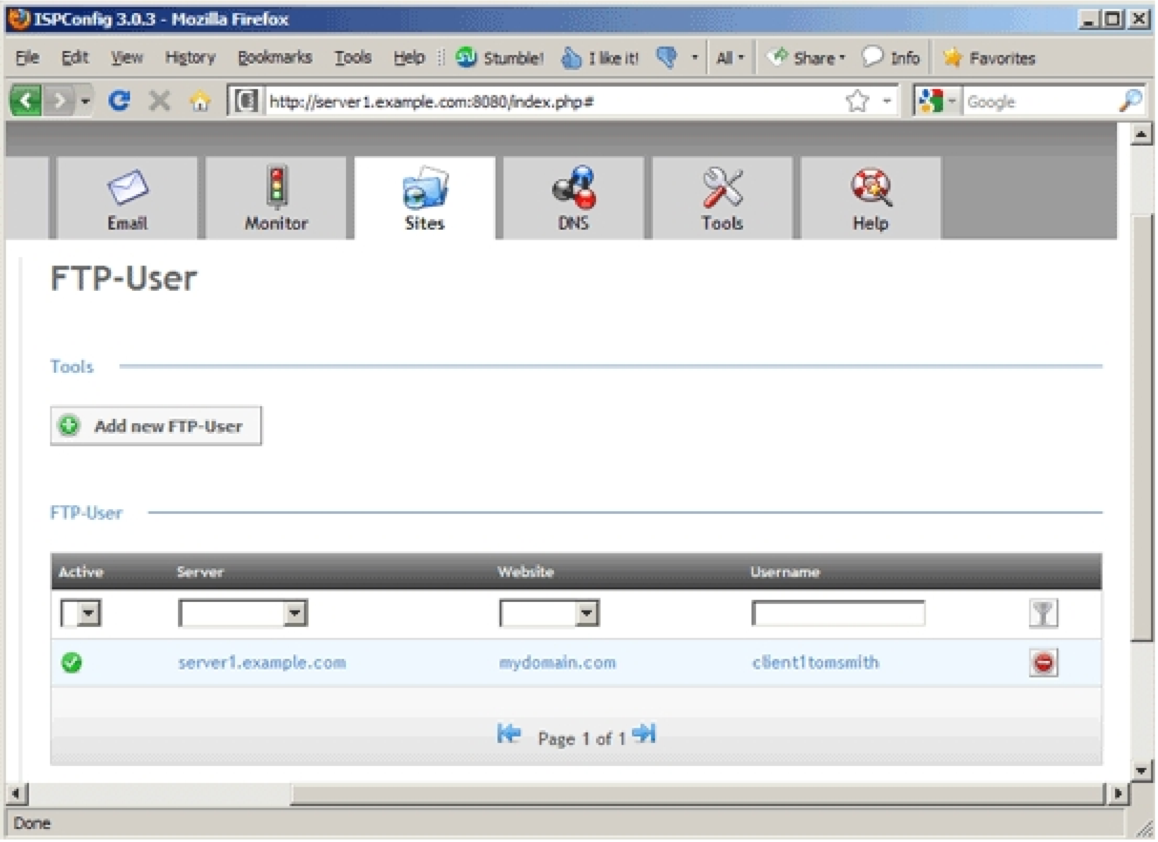


FTP User

FTP User | Options

Website	mydomain.com
Username	[CLIENTNAME] tomsmith
Password	●●●●●●●●●●●
Password strength	Good
Harddisk-Quota	100 MB
Active	☑

Afterwards you can find the new FTP user in the list under `Sites > FTP > FTP-User` (where you can also see the final username of the FTP user, `client1tomsmith` in this case which means that `[CLIENTNAME]` was replaced with `client1`):



ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
Email
Monitor
Sites
DNS
Tools
Help
Add new FTP-User
Page 1 of 1
Done

FTP-User

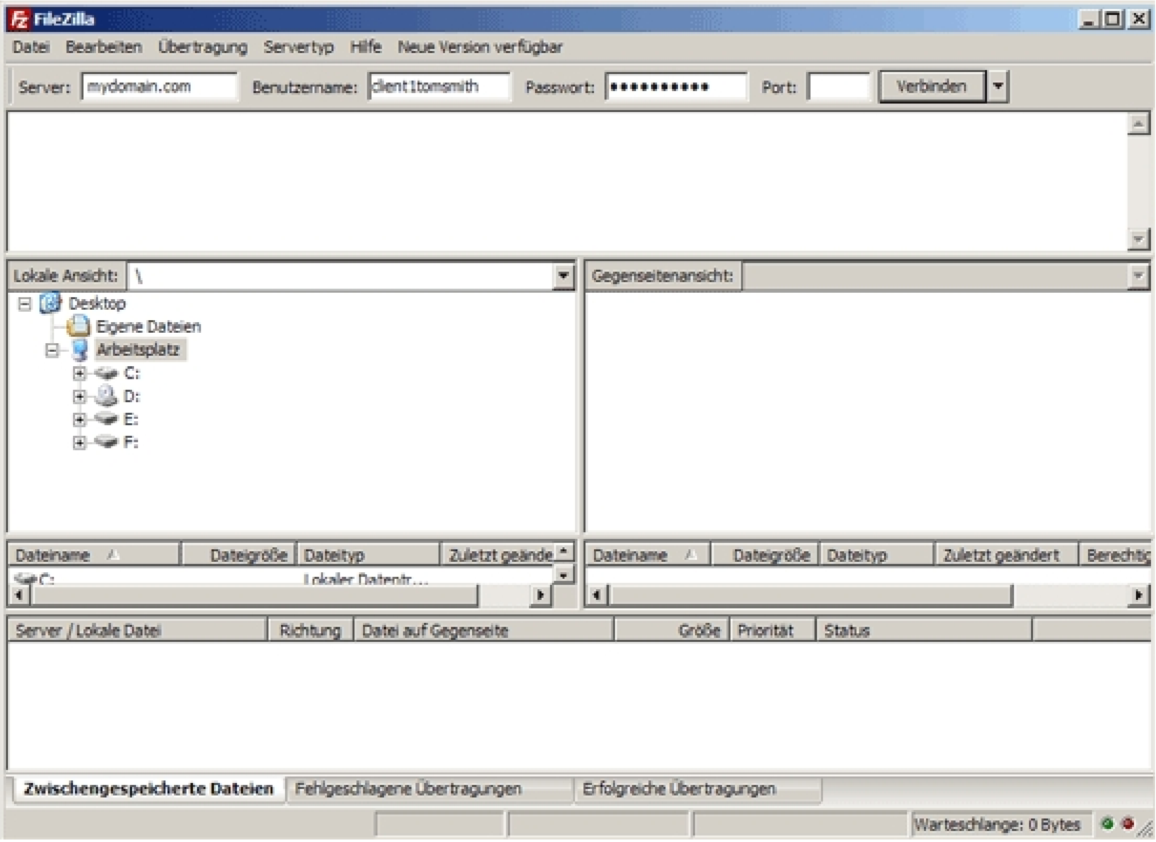
Tools

FTP-User

Active	Server	Website	Username
✓	server1.example.com	mydomain.com	client1tomsmith

From the list view, it is possible to delete the FTP user (click on the ⊖ button).

You can now use the new FTP account to log into your web site (using an FTP client such as ***FileZilla***) - use your web site domain (without *`http://`* or *`https://`*) in the *`Server`* or *`Hostname`* field of your FTP client and then your FTP username and password to log in:


FileZilla
Datei Bearbeiten Übertragung Servertyp Hilfe Neue Version verfügbar
Verbinden
Gegenseitenansicht:
Desktop
Eigene Dateien
Arbeitsplatz
C:
D:
E:
F:
Dateiname Dateigröße Dateityp Zuletzt geände
Dateiname Dateigröße Dateityp Zuletzt geändert Berechtig
Server / Lokale Datei Richtung Datei auf Gegenseite Größe Priorität Status
Zwischengespeicherte Dateien
Fehlgeschlagene Übertragungen
Erfolgreiche Übertragungen

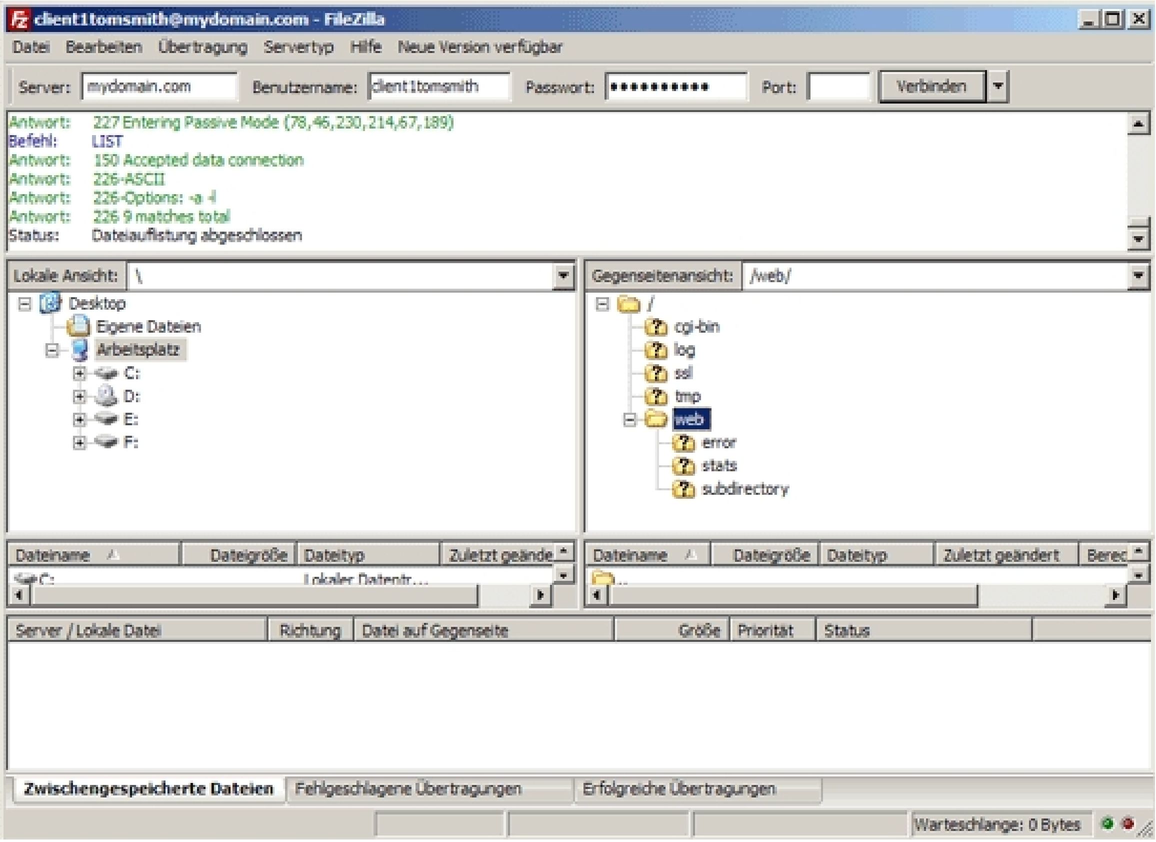
Server: mydomain.com Benutzername: client1tomsmith Passwort: •••••••••• Port:

Lokale Ansicht: \

C: Lokaler Datentr...

Warteschlange: 0 Bytes

After you've logged in, you can now see the directory structure of your web site. You must upload web site contents into the *web/* directory (or subdirectories of it) as this is the document root of your web site; Perl or CGI scripts must go into the *cgi-bin/* directory:



client1tomsmith@mydomain.com - FileZilla

Server: mydomain.com Benutzername: client1tomsmith Passwort: •••••••••• Port:

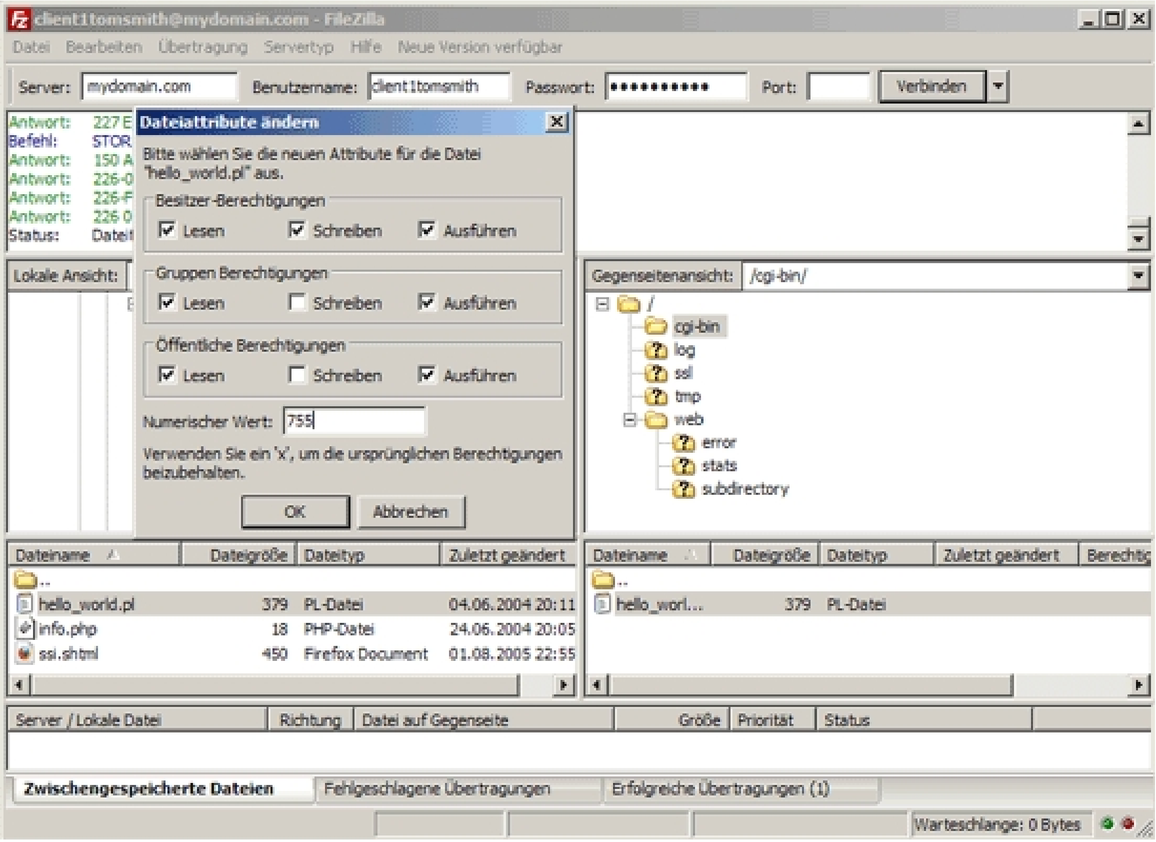
```
Antwort:   227 Entering Passive Mode (78,46,230,214,67,189)
Befehl:    LIST
Antwort:   150 Accepted data connection
Antwort:   226-ASCII
Antwort:   226-Options: -a -l
Antwort:   226 9 matches total
Status:    Dateiauflistung abgeschlossen
```

Lokale Ansicht: \

Gegenseitenansicht: /web/

Warteschlange: 0 Bytes

Please note that Perl or CGI scripts that you upload into the `cgi-bin/` directory must be executable; you can make them executable by changing the file attributes through your FTP client:



Server: mydomain.com Benutzername: client1tomsmith Passwort: •••••••••• Port:

Dateiattribute ändern

Bitte wählen Sie die neuen Attribute für die Datei "hello_world.pl" aus.

Besitzer-Berechtigungen: ☑ Lesen ☑ Schreiben ☑ Ausführen

Gruppen Berechtigungen: ☑ Lesen □ Schreiben ☑ Ausführen

Öffentliche Berechtigungen: ☑ Lesen □ Schreiben ☑ Ausführen

Numerischer Wert: 755

Verwenden Sie ein 'x', um die ursprünglichen Berechtigungen beizubehalten.

OK Abbrechen

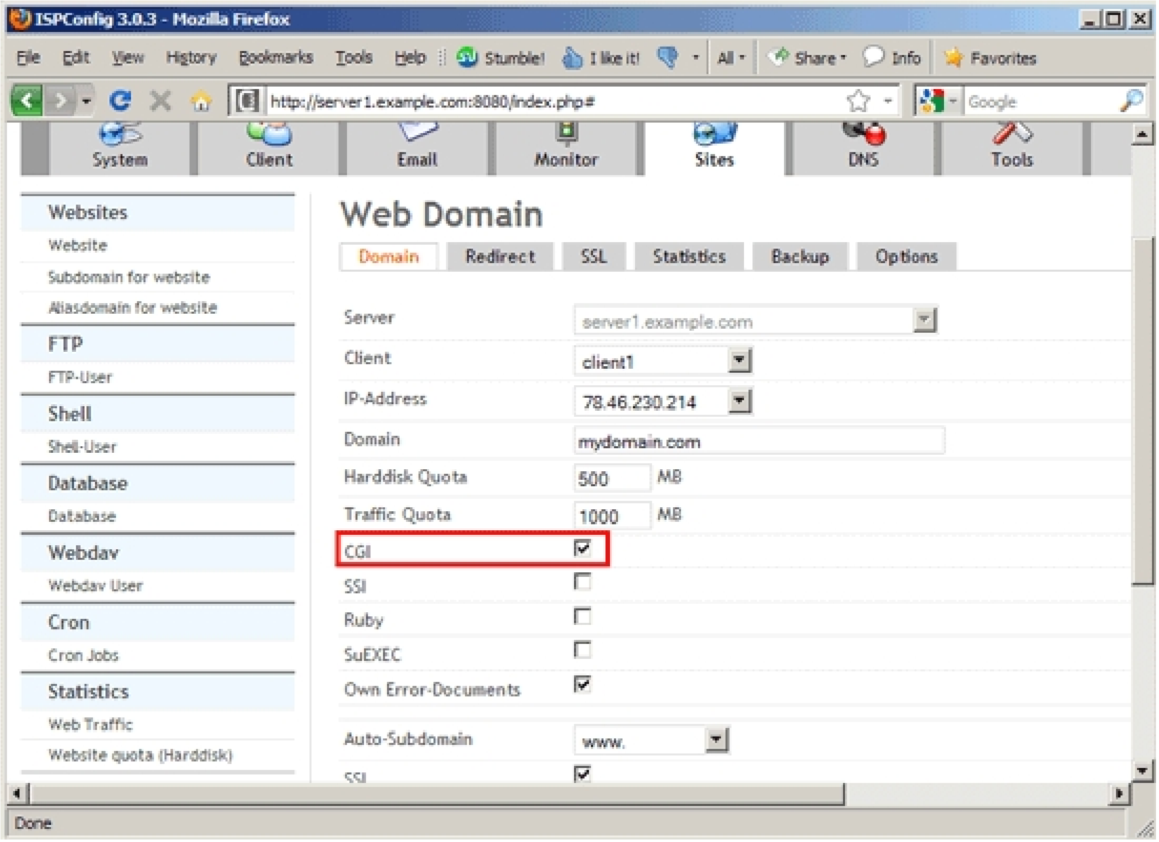
Gegenseitenansicht: /cgi-bin/

Dateiname	Dateigröße	Dateityp	Zuletzt geändert
..			
hello_world.pl	379	PL-Datei	04.06.2004 20:11
info.php	18	PHP-Datei	24.06.2004 20:05
ssi.shtml	450	Firefox Document	01.08.2005 22:55

Dateiname	Dateigröße	Dateityp	Zuletzt geändert	Berechtig
..				
hello_worl...	379	PL-Datei		

5.7 How Can I Use Perl/CGI Scripts With My Web Site?

First you must check the CGI checkbox for your web site on the Domain tab in ISPConfig:

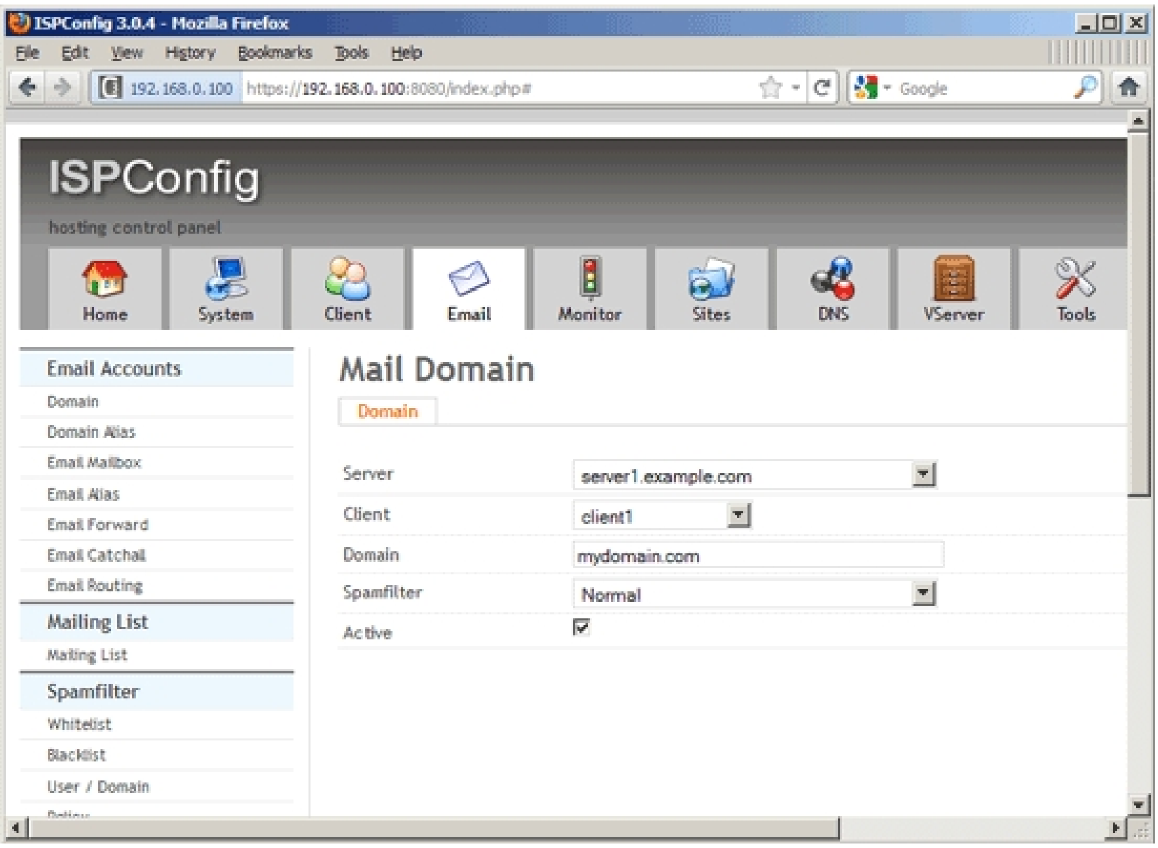

ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
System Client Email Monitor Sites DNS Tools
Websites
Website
Subdomain for website
Aliasdomain for website
FTP
FTP-User
Shell
Shell-User
Database
Database
Webdav
Webdav User
Cron
Cron Jobs
Statistics
Web Traffic
Website quota (Harddisk)
Web Domain
Domain Redirect SSL Statistics Backup Options
Done

Server	server1.example.com
Client	client1
IP-Address	78.46.230.214
Domain	mydomain.com
Harddisk Quota	500 MB
Traffic Quota	1000 MB
CGI	☑
SSI	□
Ruby	□
SuEXEC	□
Own Error-Documents	☑
Auto-Subdomain	www.
SSL	☑

Afterwards, you can upload your Perl and CGI scripts to the `cgi-bin/` directory of your web site (they will only work in that directory). Please note that you must make your Perl and CGI scripts executable (e.g. through your FTP client, see chapter ***5.6 How Do I Create An FTP Account So That I Can Upload Files To My Web Site?***) because otherwise they will not work. Also, if you have enabled suExec (Apache) for your web site, the scripts must be owned by the correct user and group (which is already the case if you uploaded them through FTP).

5.8 How Do I Create An Email Account?

The first thing we have to do is to add the domain of the email account to the system. To do this, go to *`Email > Email Accounts > Domain`* and click on the *`Add new Domain`* button. Fill in the domain name, select the correct client and enable the spamfilter for the domain, if desired:

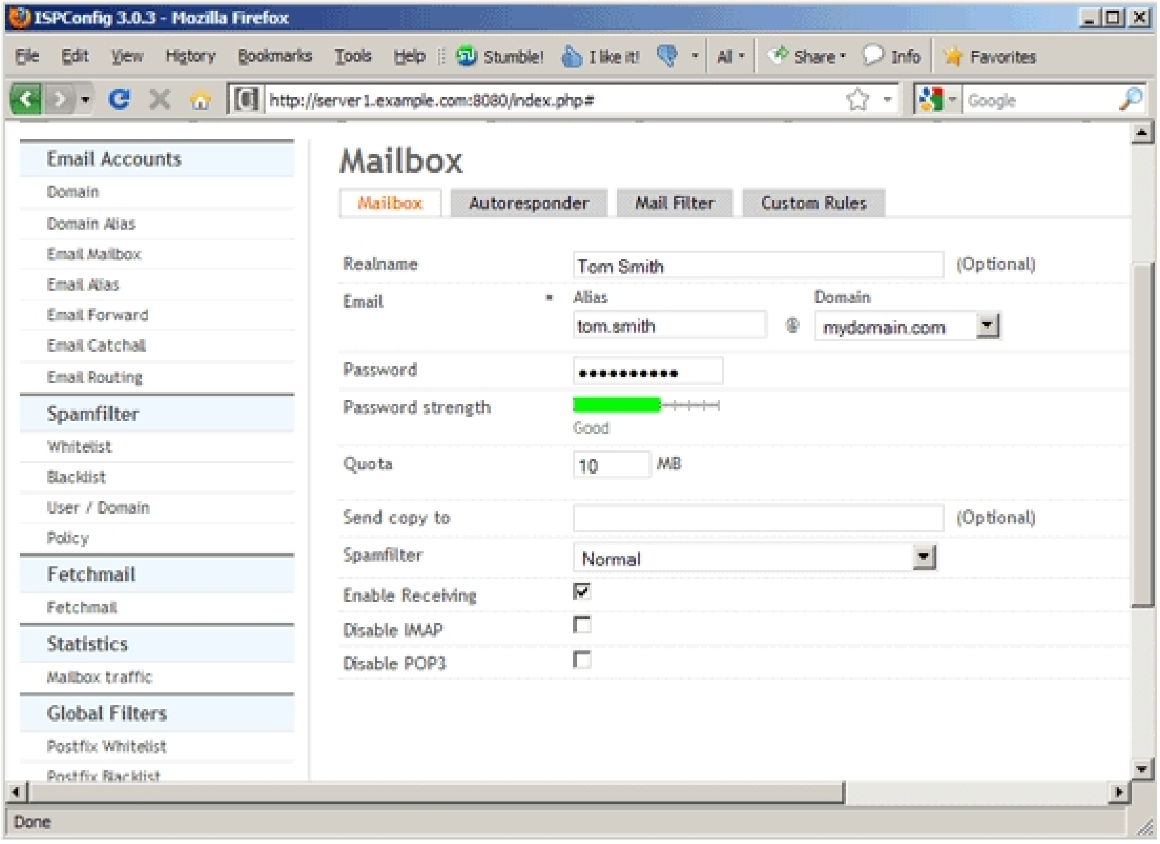


Mail Domain

Domain

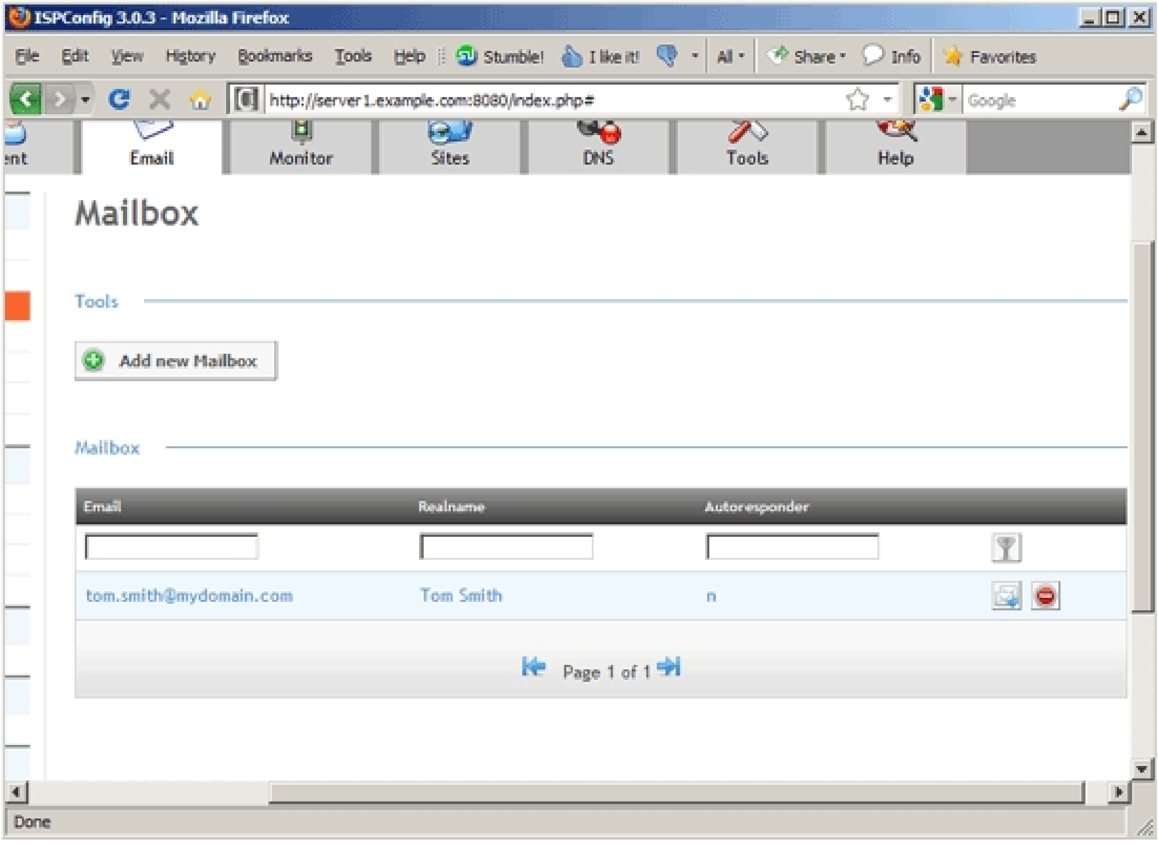
Server	server1.example.com
Client	client1
Domain	mydomain.com
Spamfilter	Normal
Active	☑

Now we can create an email account for that domain. Go to `Email > Email Accounts > Email Mailbox` and click on the `Add new Mailbox` button. Select the domain and fill in an alias (i.e., the local part or the part before the @ sign). The `Realname` and `Send copy to` fields are optional. Fill in a password for the account, set a quota in MB (`-1` means unlimited) and select a spamfilter level to use: `Non-Paying`, `Uncensored`, `Wants all spam`, `Wants viruses`, `Normal`, `Trigger happy`, `Permissive`. The settings for each of these levels are defined under `Email > Spamfilter > Policy`. Please note that this setting overrides the spamfilter setting of the mail domain (no matter what spamfilter level you chose for the mail domain; this is true even if you disabled the spamfilter for the mail domain), with one exception: If you choose to not enable the spamfilter for this email account, but the spamfilter is enabled for the mail domain, then the spamfilter setting of the mail domain is used for this email account. Use `Uncensored` to disable the spam-/virusfilter (see chapter ***4.7.1.3 Email Mailbox***):



Realname	Tom Smith	(Optional)
Email *	Alias: tom.smith @ Domain: mydomain.com	
Password	•••••••••••	
Password strength	Good	
Quota	10 MB	
Send copy to		(Optional)
Spamfilter	Normal	
Enable Receiving	☑	
Disable IMAP	□	
Disable POP3	□	

After you have created the email account, you can find it in the list under `Email > Email Accounts > Email Mailbox`:


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
ent Email Monitor Sites DNS Tools Help
Tools
Add new Mailbox
Done

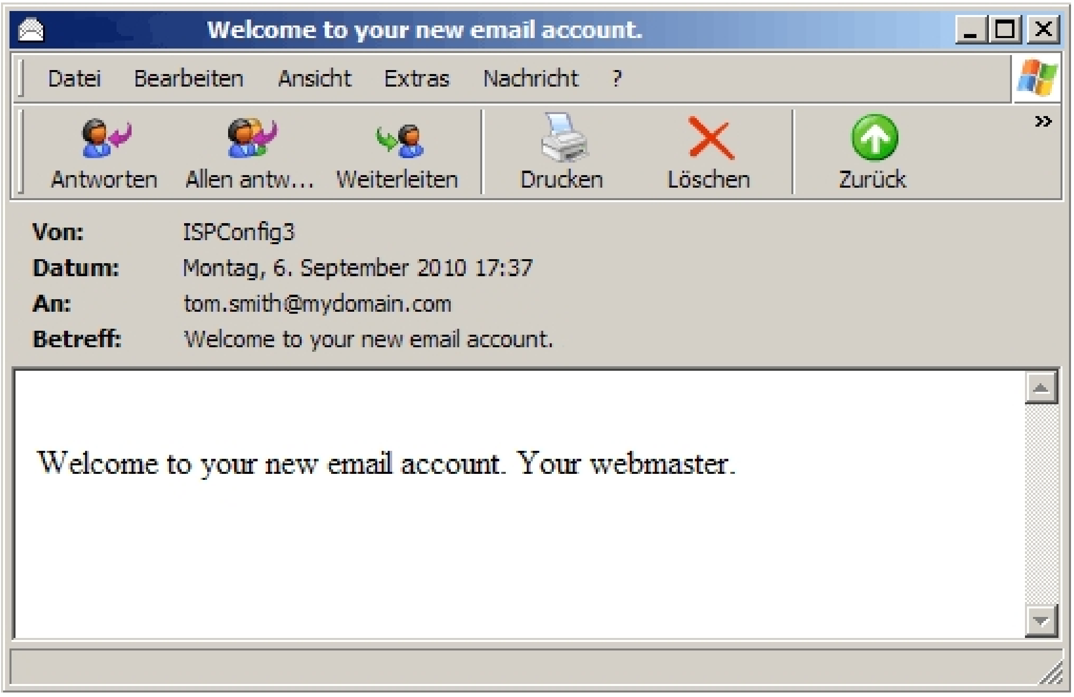
Mailbox

Email	Realname	Autoresponder
tom.smith@mydomain.com	Tom Smith	n

Page 1 of 1

From the list view, it is possible to access the email account using a webmail application (click on the ✉ button; please note that you must have installed a webmail application yourself and defined the webmail URL in the system configuration, as described in chapter 4.9.2.4) or to delete the email account (click on the ⛔ button).

Every new email account will automatically receive a welcome email from the ISPConfig 3 system:

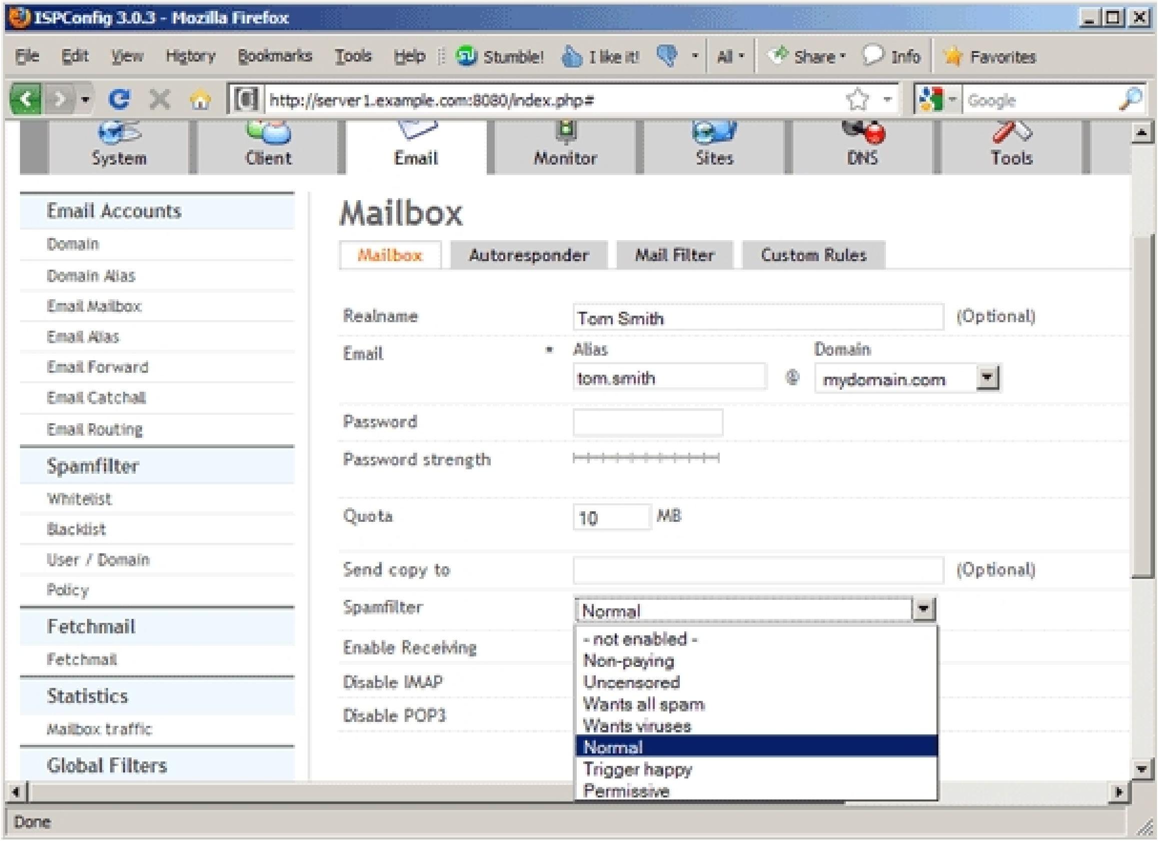


Von: ISPConfig3
Datum: Montag, 6. September 2010 17:37
An: tom.smith@mydomain.com
Betreff: Welcome to your new email account.

Welcome to your new email account. Your webmaster.

5.9 How Do I Activate The Spamfilter/Virus Scanner For An Email Account?

When you create or edit an email account, you can select a spamfilter level to use: *Non-Paying*, *Uncensored*, *Wants all spam*, *Wants viruses*, *Normal*, *Trigger happy*, *Permissive*. The settings for each of these levels are defined under *Email > Spamfilter > Policy*. Please note that this setting overrides the spamfilter setting of the mail domain (no matter what spamfilter level you chose for the mail domain; this is true even if you disabled the spamfilter for the mail domain), with one exception: If you choose to not enable the spamfilter for this email account, but the spamfilter is enabled for the mail domain, then the spamfilter setting of the mail domain is used for this email account. Use *Uncensored* to disable the spamfilter (see chapter ***4.7.1.3 Email Mailbox***).



Realname: Tom Smith (Optional)

Email *: Alias tom.smith @ Domain mydomain.com

Password:

Password strength:

Quota: 10 MB

Send copy to: (Optional)

Spamfilter: Normal

Enable Receiving

Disable IMAP

Disable POP3

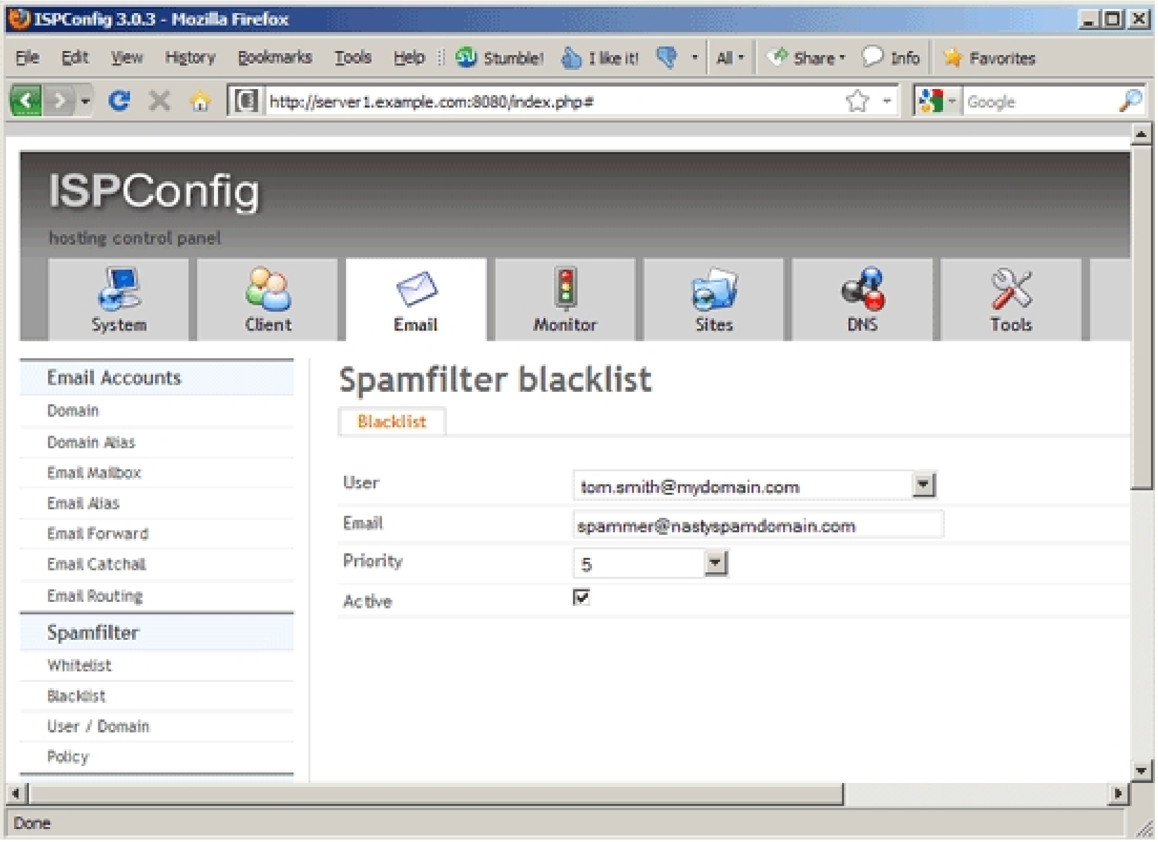
5.10 How Do I Blacklist/Whitelist Email Addresses In The Spamfilter?

To blacklist an email address in the spamfilter (which means that emails originating from that email address will always be considered spam), go to *Email > Spamfilter > Blacklist* and click on the *Add Blacklist record* button (see chapter ***4.7.3.2 Blacklist***).

Select the user or the whole domain that will benefit from this blacklist record in the *User* drop-down menu, and then fill in the email address that you want to blacklist in the *Email* field.

If multiple whitelist/blacklist records apply, the *Priority* field specifies which rule to use first (*10* = highest priority, *1* = lowest priority). For example, if you blacklist *@nastyspamdomain.com* with a priority of *5*, you could whitelist *gooduser@nastyspamdomain.com* with a priority of 6 so that *gooduser@nastyspamdomain.com*'s mails get through while *@nastyspamdomain.com* is blacklisted. In most cases you can disregard the *Priority* field.

Make sure that the *Active* checkbox is checked and click on *Save*.

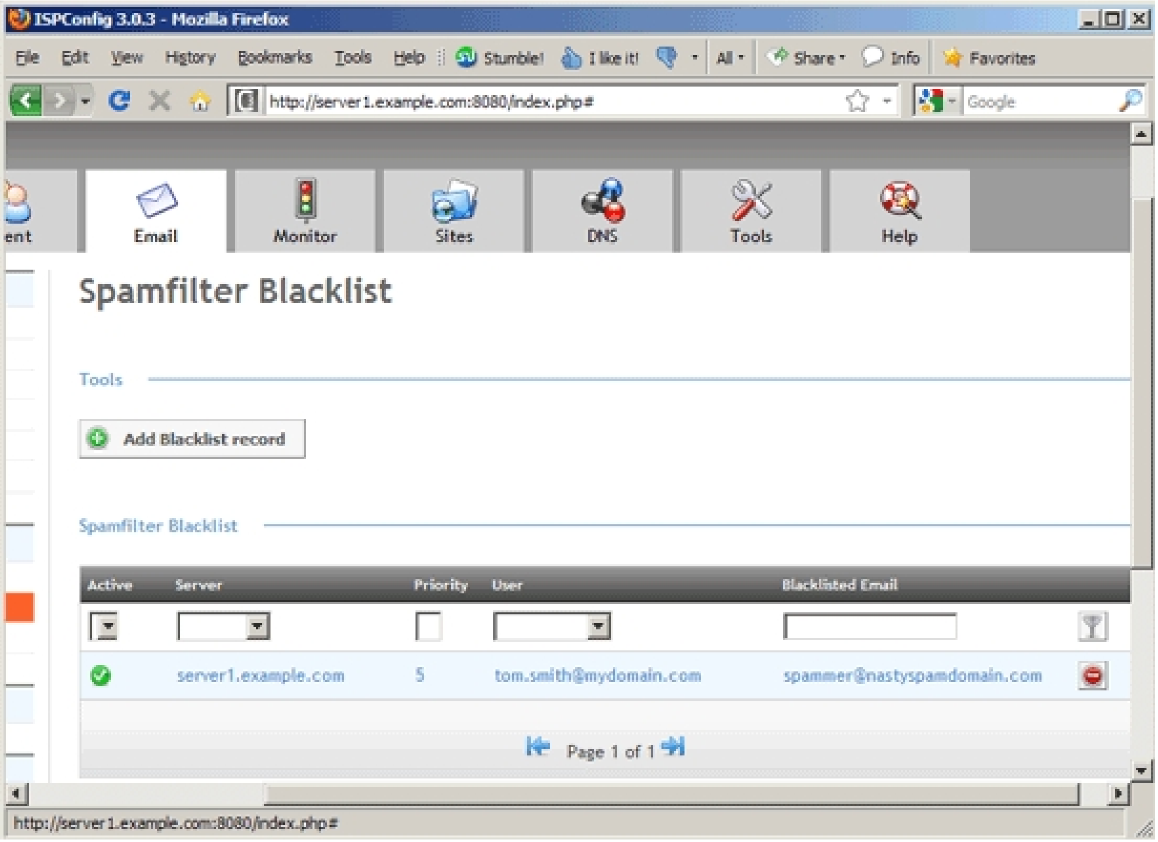


Spamfilter blacklist

Blacklist

User	tom.smith@mydomain.com
Email	spammer@nastyspamdomain.com
Priority	5
Active	☑

Afterwards you can find the new blacklist record in the list under `Email > Spamfilter > Blacklist`:


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
ent Email Monitor Sites DNS Tools Help
http://server1.example.com:8080/index.php#

Spamfilter Blacklist

Tools

Add Blacklist record

Spamfilter Blacklist

Active	Server	Priority	User	Blacklisted Email
✓	server1.example.com	5	tom.smith@mydomain.com	spammer@nastyspamdomain.com

Page 1 of 1

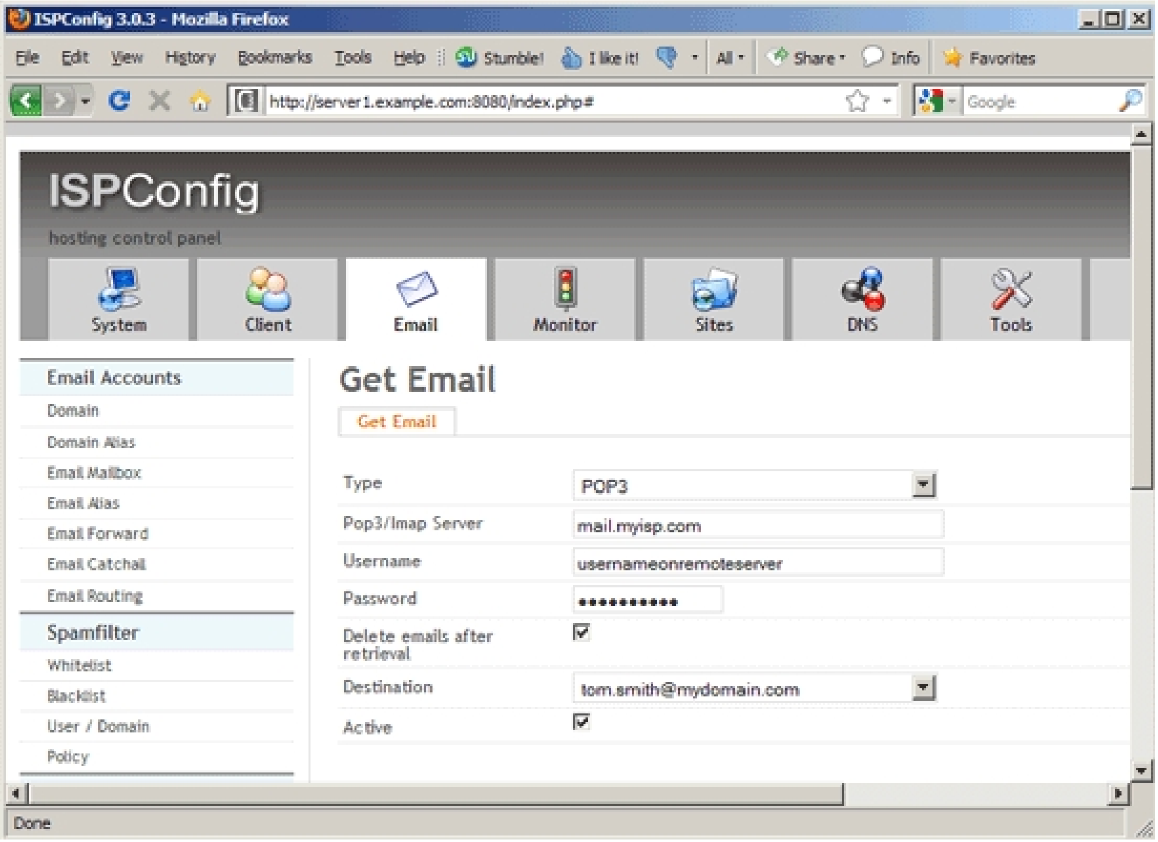
From the list view, it is possible to delete the blacklist record (click on the ⛔ button).

Creating whitelist records works the same as for blacklist records - just go to `Email > Spamfilter > Whitelist` (see chapter ***4.7.3.1 Whitelist***).

5.11 How Do I Fetch Emails From A Remote Server With ISPConfig And Put The Emails In A Local Email Account?

You can use ISPConfig to retrieve emails from a remote POP3 or IMAP account and put them into a local mailbox (see chapter ***4.7.4.1 Fetchmail***). To create such a Fetchmail account, go to *`Email > Fetchmail > Fetchmail`* and click on the *`Add new Account`* button.

Select the protocol that should be used to retrieve emails from the remote server (*`POP3`*, *`IMAP`*, *`POP3SSL`*, *`IMAPSSL`*), then specify the hostname of the remote mail server, the username of the mailbox on the remote server together with the password, and select the local mailbox (in the *`Destination`* field) where mails retrieved from the remote server should be put. If you want emails to be automatically deleted on the remote host after they have been retrieved, check the *`Delete emails after retrieval`* checkbox:



Type: POP3

Pop3/Imap Server: mail.myisp.com

Username: usernameonremoteserver

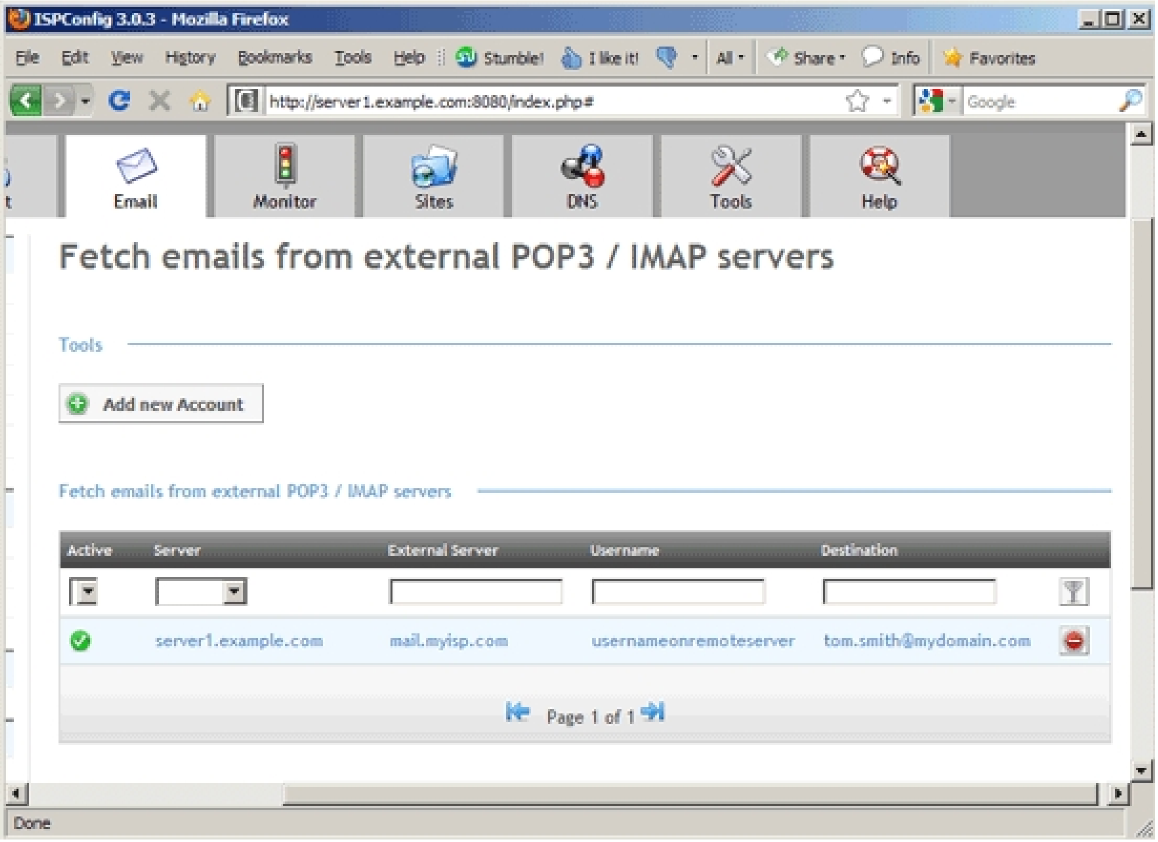
Password: •••••••••••

Delete emails after retrieval: ☑

Destination: tom.smith@mydomain.com

Active: ☑

Afterwards you can find the new Fetchmail account in the list under *`Email > Fetchmail > Fetchmail`*:


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
Email
Monitor
Sites
DNS
Tools
Help
Tools
Add new Account
Done

Fetch emails from external POP3 / IMAP servers

Fetch emails from external POP3 / IMAP servers

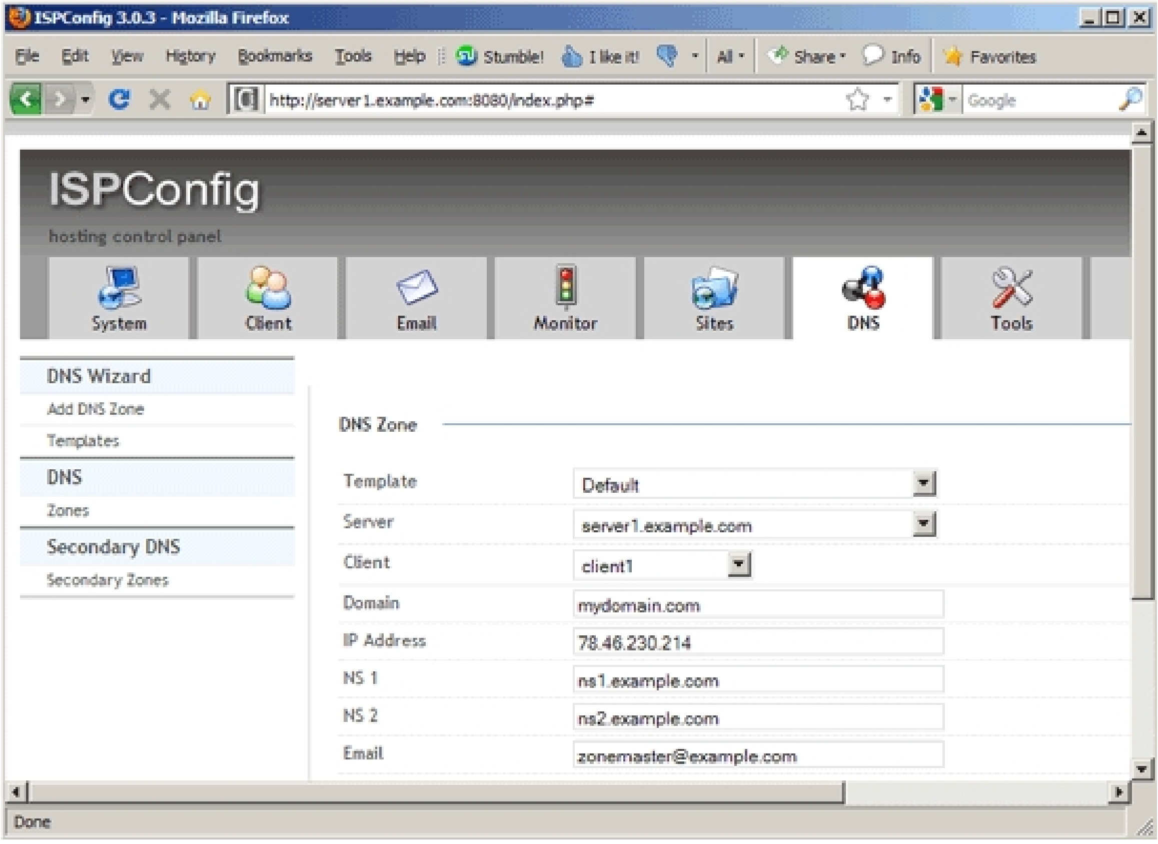
Active	Server	External Server	Username	Destination
✓	server1.example.com	mail.myisp.com	usernameonremoteserver	tom.smith@mydomain.com

Page 1 of 1

From the list view, it is possible to delete the Fetchmail account (click on the ⛔ button).

5.12 How Do I Create A DNS Zone?

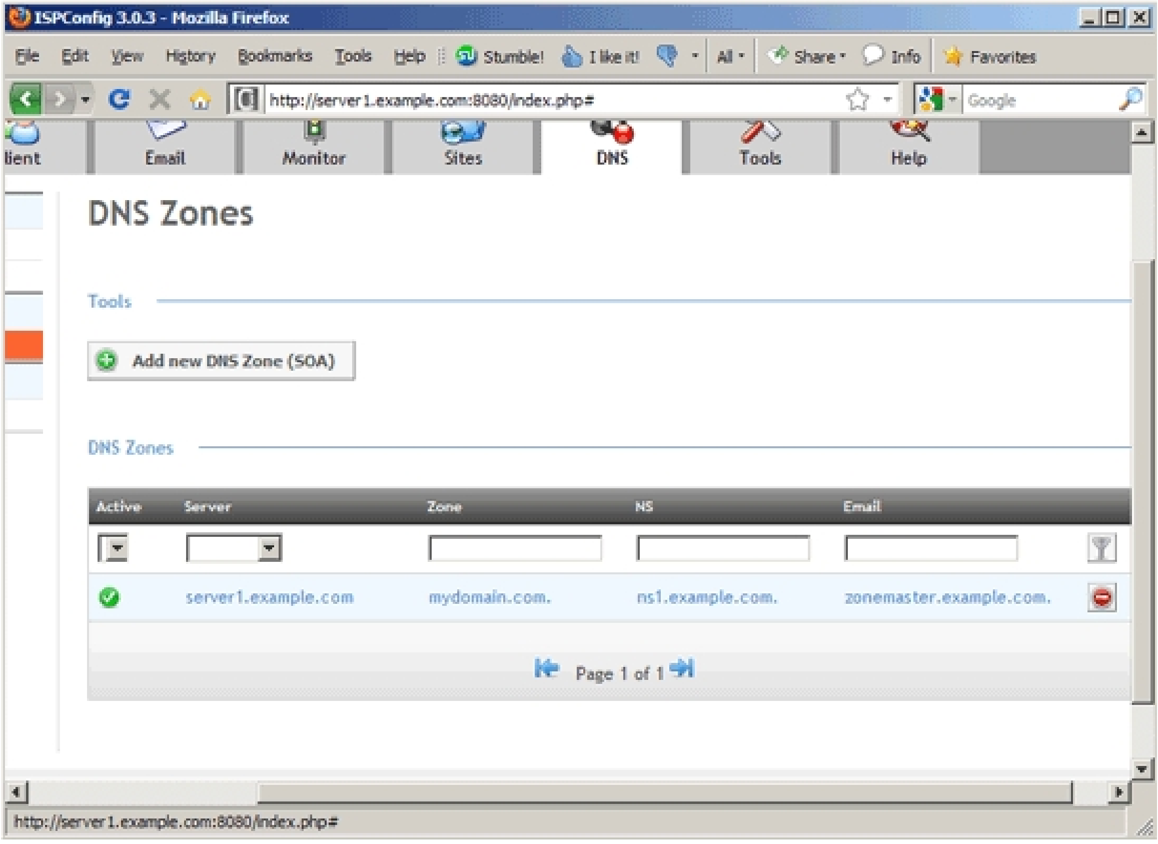
To create a DNS zone, it is recommended to use the DNS Wizard (*DNS > DNS Wizard > Add DNS Zone*) which will automatically create a set of common DNS records for your domain (like *www*, *mail*, *ns* records, etc.) (see chapter ***4.8.1.1 Add DNS Zone***).



DNS Zone

Template	Default
Server	server1.example.com
Client	client1
Domain	mydomain.com
IP Address	78.46.230.214
NS 1	ns1.example.com
NS 2	ns2.example.com
Email	zonemaster@example.com

Afterwards you can find the new zone in the list under `DNS > DNS > Zones`:


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
lient
Email
Monitor
Sites
DNS
Tools
Help

DNS Zones

Tools

Add new DNS Zone (SOA)

DNS Zones

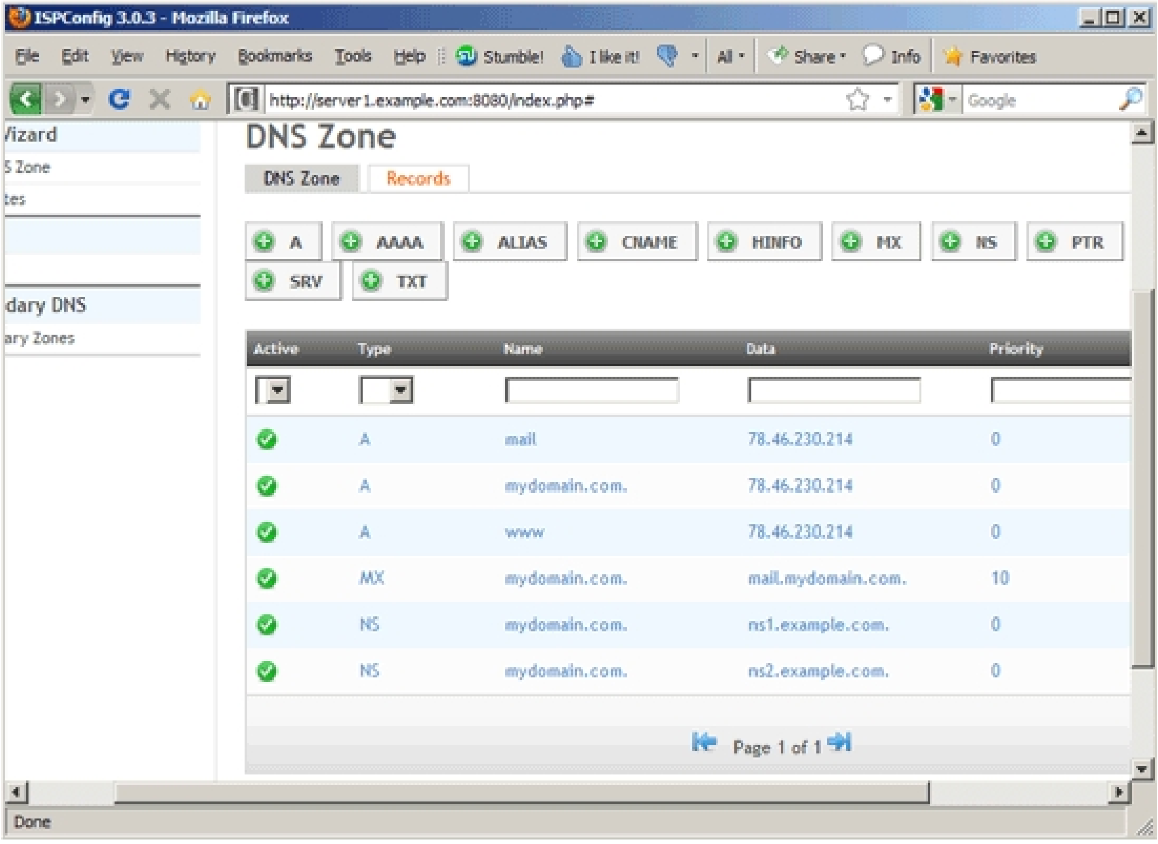
Active	Server	Zone	NS	Email
✓	server1.example.com	mydomain.com.	ns1.example.com.	zonemaster.example.com.

Page 1 of 1

http://server1.example.com:8080/index.php#

From the list view, it is possible to delete the DNS zone (click on the ⊖ button).

If you edit the zone and go to the *Records* tab, you will see the records that have automatically been created by the DNS Wizard (the *Default* template will create A records for *mydomain.com*, *www.mydomain.com*, and *mail.mydomain.com*, two NS (nameserver) records, plus an MX (mail exchanger) record for *mydomain.com* that points to *mail.mydomain.com*):


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
/izard
S Zone
tes
dary DNS
ary Zones
DNS Zone
DNS Zone
Records
A
AAAA
ALIAS
CNAME
HINFO
MX
NS
PTR
SRV
TXT
Page 1 of 1
Done

Active	Type	Name	Data	Priority
✓	A	mail	78.46.230.214	0
✓	A	mydomain.com.	78.46.230.214	0
✓	A	www	78.46.230.214	0
✓	MX	mydomain.com.	mail.mydomain.com.	10
✓	NS	mydomain.com.	ns1.example.com.	0
✓	NS	mydomain.com.	ns2.example.com.	0

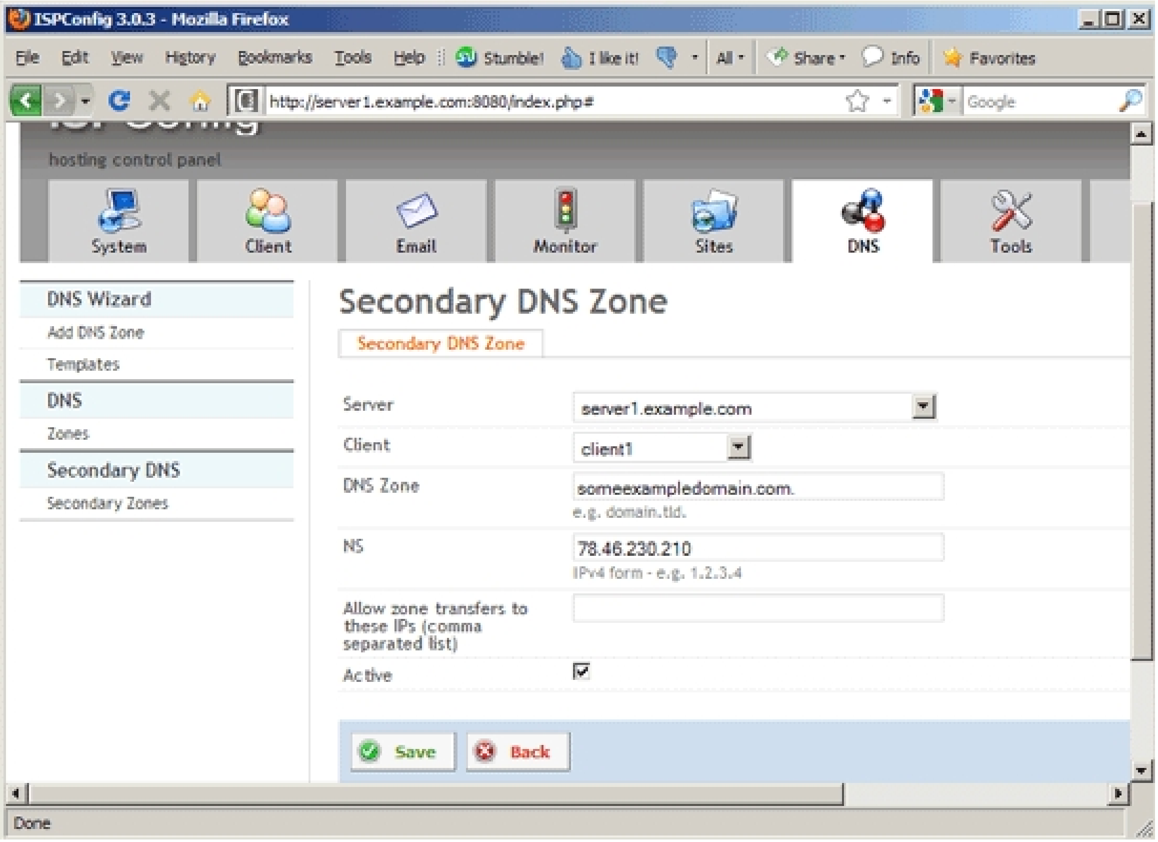
On the *Records* tab, you can edit or delete existing records and add further ones.

5.13 How Do I Create A Secondary DNS Zone?

(This feature is supported only if you use the BIND name server. If you use MyDNS, database replication will be used to transfer data to the secondary DNS server.)

If you've already created the master DNS zone for a domain on another server and would like to use ISPConfig to create the secondary zone for the domain on one of the servers controlled by ISPConfig, go to *DNS > Secondary DNS > Secondary Zones* and click on the *Add new secondary DNS Zone* button (see chapter ***4.8.3.1 Secondary Zones***).

Select the server and the client for the secondary zone, then fill in the domain for which you want to create the secondary zone in the *DNS Zone* field, e.g. *someexampledomain.com.* - please note that you need a dot at the end. Then specify the IPv4 address of the primary nameserver for the domain in the *NS* field, e.g. *1.2.3.4*. Make sure that the *Active* checkbox is checked and click on *Save*:

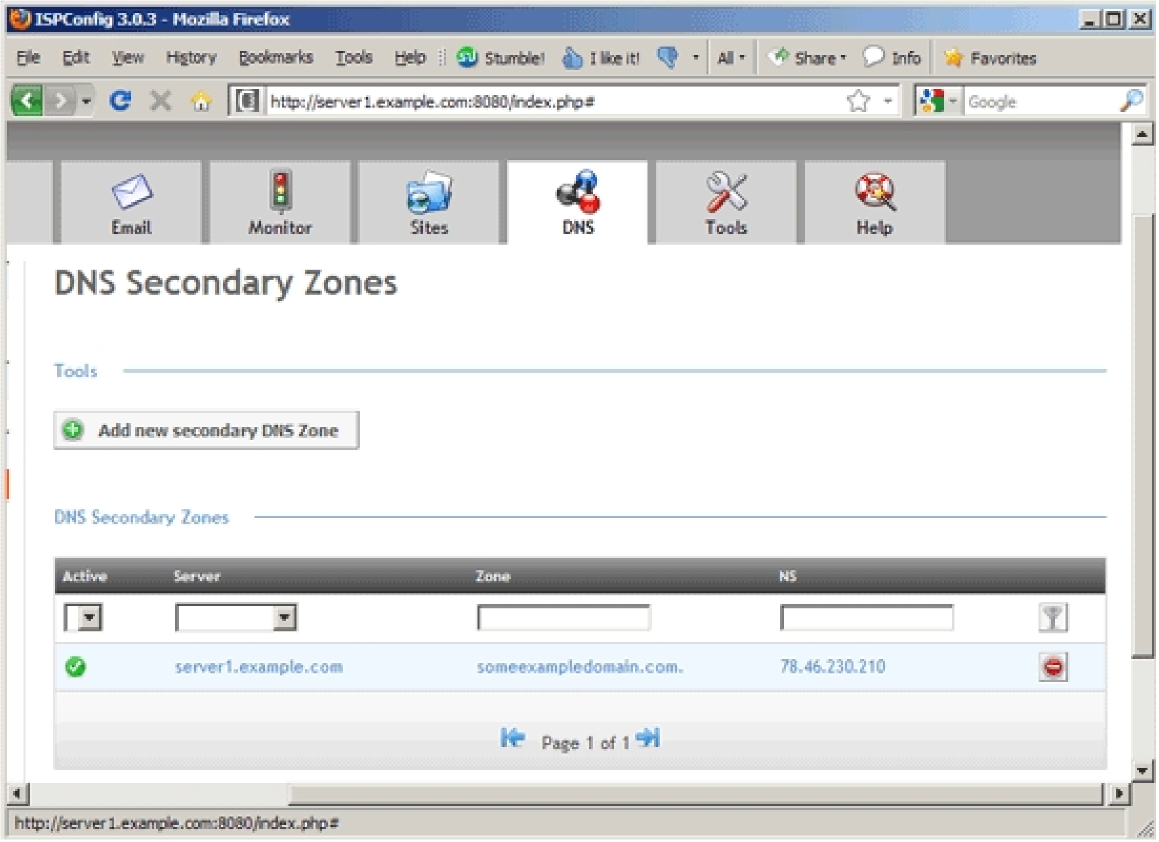

ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://server1.example.com:8080/index.php#
hosting control panel
System Client Email Monitor Sites DNS Tools
DNS Wizard
Add DNS Zone
Templates
DNS
Zones
Secondary DNS
Secondary Zones
Save
Back
Done

Secondary DNS Zone

Secondary DNS Zone

Server	server1.example.com
Client	client1
DNS Zone	someexampledomain.com. e.g. domain.tld.
NS	78.46.230.210 IPv4 form - e.g. 1.2.3.4
Allow zone transfers to these IPs (comma separated list)	
Active	☑

Afterwards you can find the new zone in the list under `DNS > Secondary DNS > Secondary Zones`:


ISPConfig 3.0.3 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! All Share Info Favorites
http://server1.example.com:8080/index.php#
Google
Email
Monitor
Sites
DNS
Tools
Help
http://server1.example.com:8080/index.php#

DNS Secondary Zones

Tools

Add new secondary DNS Zone

DNS Secondary Zones

Active	Server	Zone	NS	
✓	server1.example.com	someexampledomain.com.	78.46.230.210	⊖

Page 1 of 1

From the list view, it is possible to delete the secondary DNS zone (click on the ⊖ button).

5.14 How Do I Create A Mirror?

Please take a look at chapter ***3.3 Mirror Setup***.

5.15 How Do I Split Up Services Between Multiple Servers?

Please take a look at chapter ***3.2 Multiserver Setup***.

5.16 How Do I Unblock An IP Address That Got Blocked By fail2ban?

If you want to unblock an IP address that got blocked by fail2ban, first run

```
iptables -L
```

Output could be as follows:

```
root@server1:~# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
fail2ban-ssh  tcp  --  anywhere             anywhere            tcp dpt:ssh

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain fail2ban-ssh (1 references)
target     prot opt source               destination
DROP       0    --  some.remote.host  anywhere
RETURN     0    --  anywhere             anywhere
root@server1:~#
```

Notice *some.remote.host* is currently being blocked here. You can tell iptables to drop that rule. The syntax is *iptables -D <rulename> <rule line>*. To unblock *some.remote.host*, run

```
iptables -D fail2ban-ssh 1
```

Run *iptables -L* again, and you should see that the rule is gone, and *some.remote.host* should be able to log in via SSH again.

5.17 How Do I Create A Subdomain And Redirect It To A Different Folder/Web Site?

Go to *Sites > Websites > Subdomain for website* (see chapter **4.6.1.2 Subdomain for website**). In the *Redirect Type* field, please select the flag that you want to use for the redirect:

Flags (Apache):

- *R*: Use of the *[R]* flag causes a HTTP redirect to be issued to the browser. If a fully-qualified URL is specified (that is, including *http://servername/*) then a redirect will be issued to that location. Otherwise, the current servername will be used to generate the URL sent with the redirect.
- *L*: The *[L]* flag causes mod_rewrite to stop processing the rule set. In most contexts, this means that if the rule matches, no further rules will be processed.
- *R,L*: You will almost always want to use *[R]* in conjunction with *[L]* (that is, use *[R,L]*) because on its own, the *[R]* flag prepends *http://thishost[:thisport]* to the URI, but then passes this on to the next rule in the ruleset, which can often result in 'Invalid URI in request' warnings.

More details about flags can be found here: ***http://httpd.apache.org/docs/2.2/rewrite/flags.html***

If you want to do a URL redirect, you should use the R,L flags, while for a directory redirect it is recommended to just use the L flag.

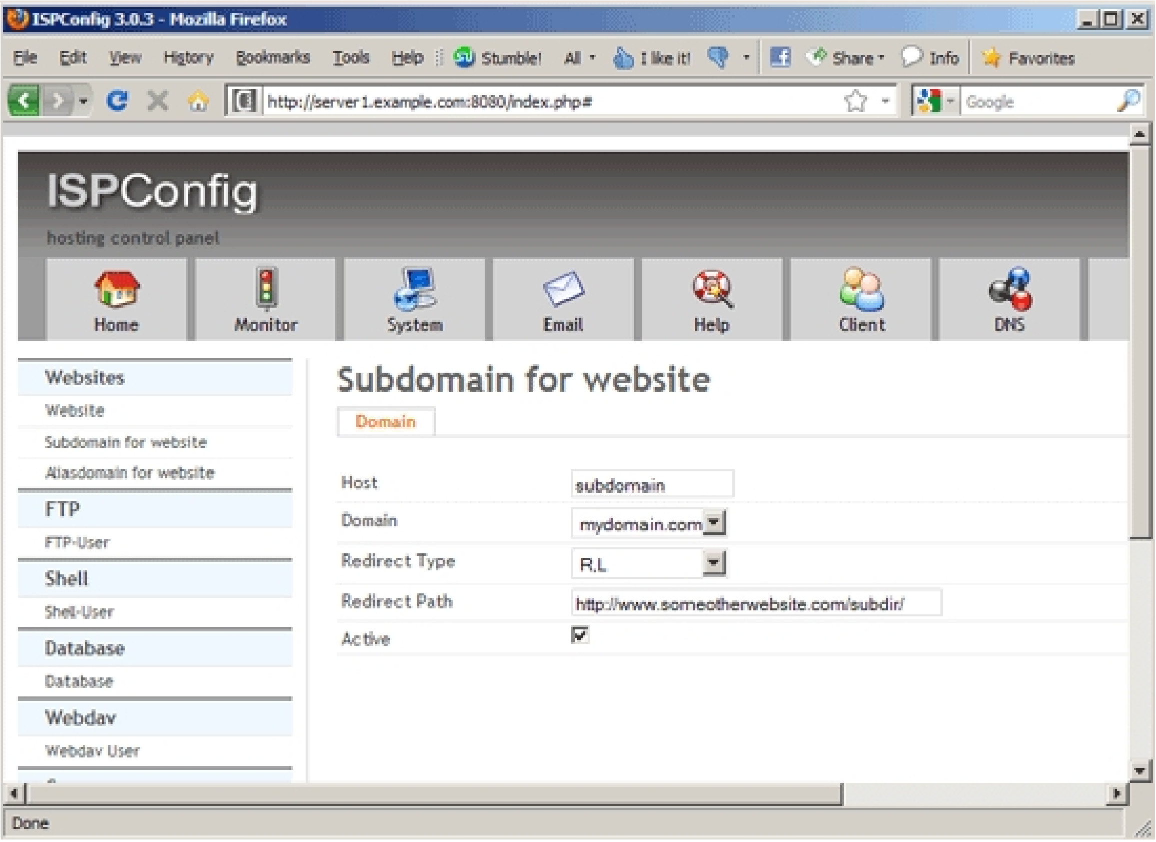
Flags (nginx):

- `last`: Completes processing of rewrite directives, after which searches for corresponding URI and location.
- `break`: Completes processing of rewrite directives and breaks location lookup cycle by not doing any location lookup and internal jump at all.
- `redirect`: Returns temporary redirect with code 302; it is used if the substituting line begins with `http://`.
- `permanent`: Returns permanent redirect with code 301.

More details about Apache rewrite flags can be found here: ***http://wiki.nginx.org/NginxHttpRewriteModule#rewrite***

If you want to do a URL redirect, you should use the permanent flag.

If you want to do a URL redirect, please specify the redirect target URL in the Redirect Path field (e.g. `http://www.someotherwebsite.com/subdir/` or `http://www.someotherwebsite.com/`). Please note that the URL should have a trailing slash:

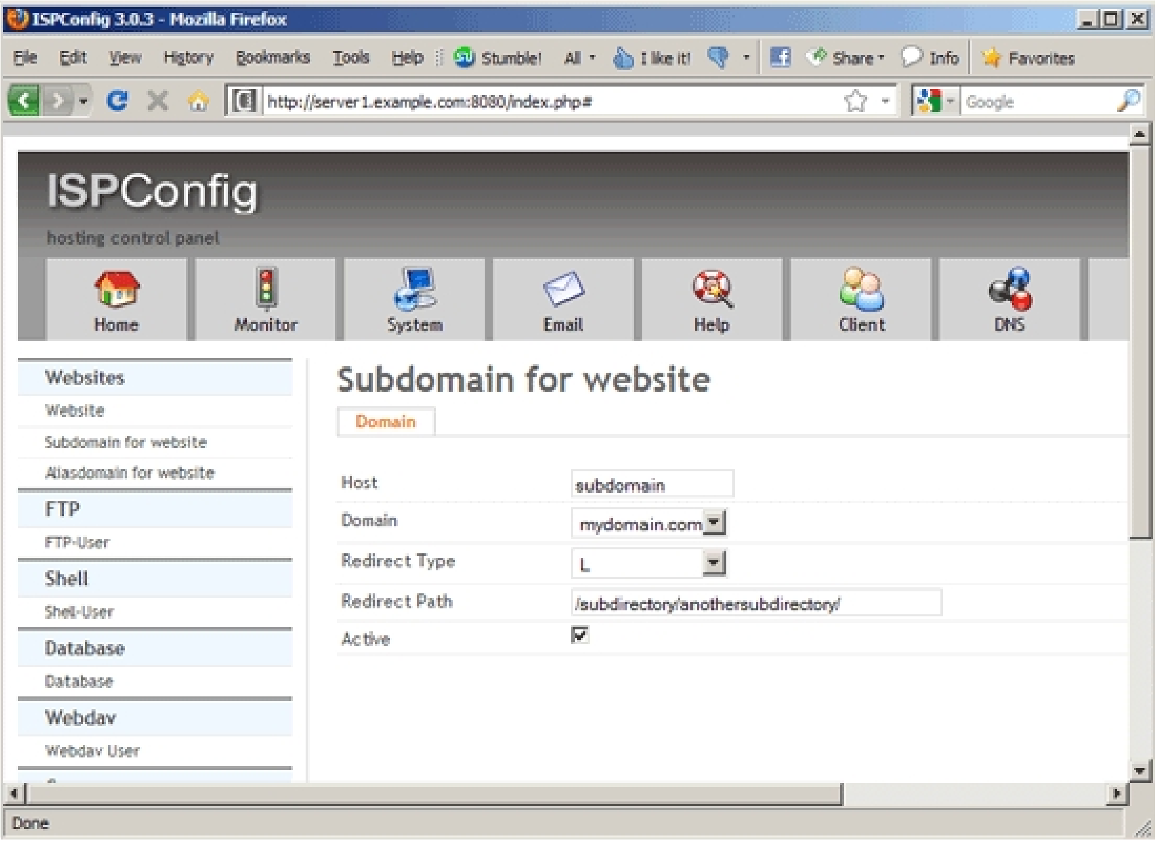


Subdomain for website

Domain

Host	subdomain
Domain	mydomain.com
Redirect Type	R,L
Redirect Path	http://www.someotherwebsite.com/subdir/
Active	☑

If you want to do a redirect to a subdirectory of your web site, please specify the subdirectory or the path to the subdirectory (relative to the document root of your web site) in the `Redirect Path` field. Please note that the path must begin and end with a slash (e.g. `/subdirectory/anothersubdirectory/`):



Subdomain for website

Host: subdomain
Domain: mydomain.com
Redirect Type: L
Redirect Path: /subdirectory/anothersubdirectory/
Active: ☑

5.18 How Do I Manually Configure New IP Addresses On My System?

I'm assuming that your system uses the static IP address `192.168.0.100` on the network interface `eth0`, and that you want to add the IP address `192.168.0.101` to that interface.

Debian/Ubuntu:

Open `/etc/network/interfaces`:

```
vi /etc/network/interfaces
```

It will probably look like this:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
        address 192.168.0.100
        netmask 255.255.255.0
        network 192.168.0.0
        broadcast 192.168.0.255
        gateway 192.168.0.1
```

What we do now is duplicate the `eth0` stanza, but instead of `eth0` we use `eth0:0` (a virtual network device), and in the address line we use the new IP address `192.168.0.101` instead of `192.168.0.100`. All other settings remain the same. In the end the complete file looks as follows:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
        address 192.168.0.100
        netmask 255.255.255.0
        network 192.168.0.0
        broadcast 192.168.0.255
        gateway 192.168.0.1

auto eth0:0
iface eth0:0 inet static
        address 192.168.0.101
        netmask 255.255.255.0
        network 192.168.0.0
        broadcast 192.168.0.255
        gateway 192.168.0.1
```

(If you want to use a third, fourth, etc. IP address, use the virtual interfaces `eth0:1`, `eth0:2`, and so on. If you are unsure about the network settings, you can use this network calculator: **http://subnetmask.info/**.)

Restart the network afterwards:

```
/etc/init.d/networking restart
```

The command

```
ifconfig
```

should show the new interface afterwards:

```
server1:~# ifconfig
  eth0      Link encap:Ethernet  HWaddr 00:0C:29:FD:78:BE
          inet addr:192.168.0.100  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fefd:78be/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:130 errors:0 dropped:0 overruns:0 frame:0
          TX packets:137 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:12592 (12.2 KiB)  TX bytes:31876 (31.1 KiB)
          Base address:0x1070 Memory:ec820000-ec840000

  eth0:0    Link encap:Ethernet  HWaddr 00:0C:29:FD:78:BE
          inet addr:192.168.0.101  Bcast:192.168.0.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          Base address:0x1070 Memory:ec820000-ec840000

  lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:560 (560.0 b)  TX bytes:560 (560.0 b)

  server1:~#
```

Fedora/CentOS:

The file `/etc/sysconfig/network-scripts/ifcfg-eth0` contains the settings for `eth0`. We can use this as a sample for our new virtual network interface `eth0:0` (which we use for our additional IP address `192.168.0.101`):

```
cp /etc/sysconfig/network-scripts/ifcfg-eth0 /etc/sysconfig/network-scripts/ifcfg-eth0:0
```

Now we want to use the IP address `192.168.0.101` on the virtual interface `eth0:0`. Therefore we open the file `/etc/sysconfig/network-scripts/ifcfg-eth0:0` and modify it as follows (use `eth0:0` in the `DEVICE` line and `192.168.0.101` in the `IPADDR` line; the other settings should remein the same; we can leave out the `HWADDR` line as it is the same physical network card):

```
vi /etc/sysconfig/network-scripts/ifcfg-eth0:0
```

```
DEVICE=eth0:0
BOOTPROTO=static
BROADCAST=192.168.0.255
IPADDR=192.168.0.101
NETMASK=255.255.255.0
NETWORK=192.168.0.0
ONBOOT=yes
```

(If you want to use a third, fourth, etc. IP address, do the same steps again, but use the virtual interfaces *eth0:1*, *eth0:2*, and so on. If you are unsure about the network settings, you can use this network calculator: **http://subnetmask.info/**.)

Restart the network afterwards:

```
/etc/init.d/network restart
```

Now run

```
ifconfig
```

You should now see your new IP address in the output:

```
[root@server1 ~]# ifconfig
  eth0      Link encap:Ethernet  HWaddr 00:0C:29:FD:78:BE
          inet addr:192.168.0.100  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fefd:78be/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:130 errors:0 dropped:0 overruns:0 frame:0
          TX packets:137 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:12592 (12.2 KiB)  TX bytes:31876 (31.1 KiB)
          Base address:0x1070 Memory:ec820000-ec840000

  eth0:0    Link encap:Ethernet  HWaddr 00:0C:29:FD:78:BE
          inet addr:192.168.0.101  Bcast:192.168.0.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          Base address:0x1070 Memory:ec820000-ec840000

  lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:560 (560.0 b)  TX bytes:560 (560.0 b)
```

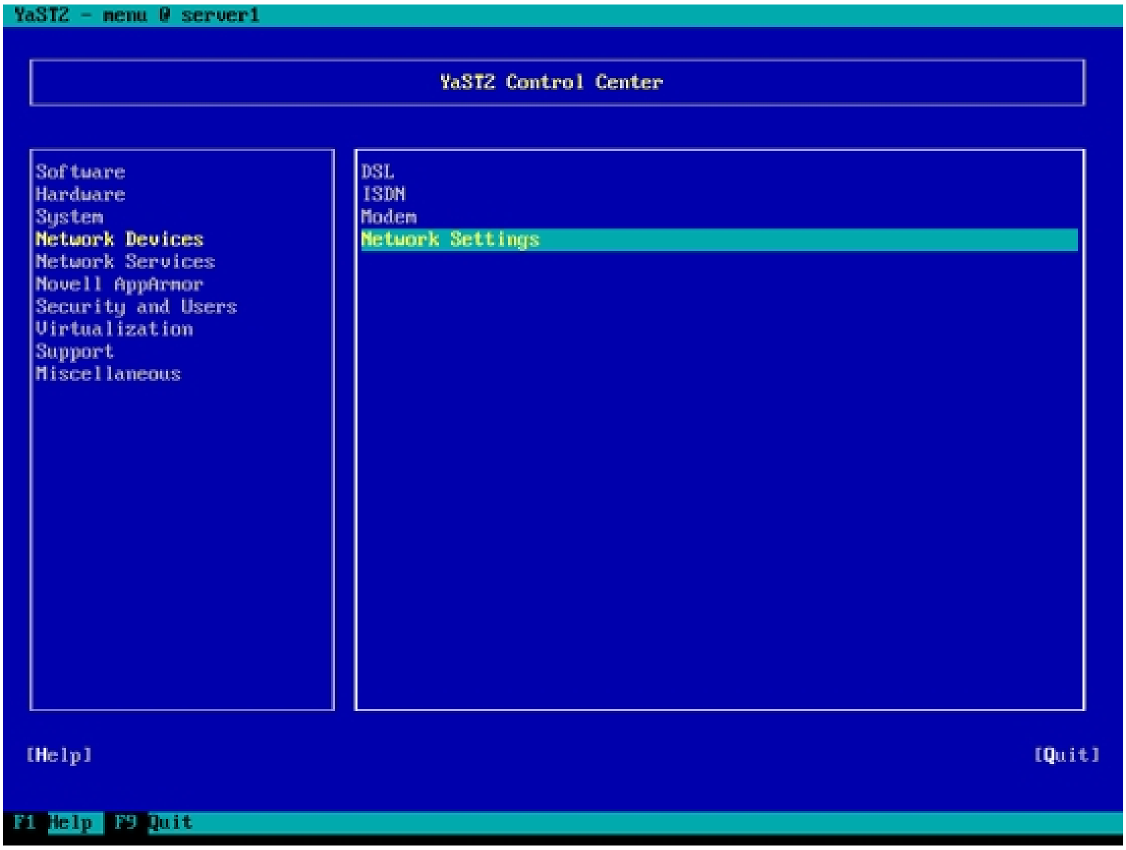
```
[root@server1 ~]#
```

OpenSUSE:

Start YaST:

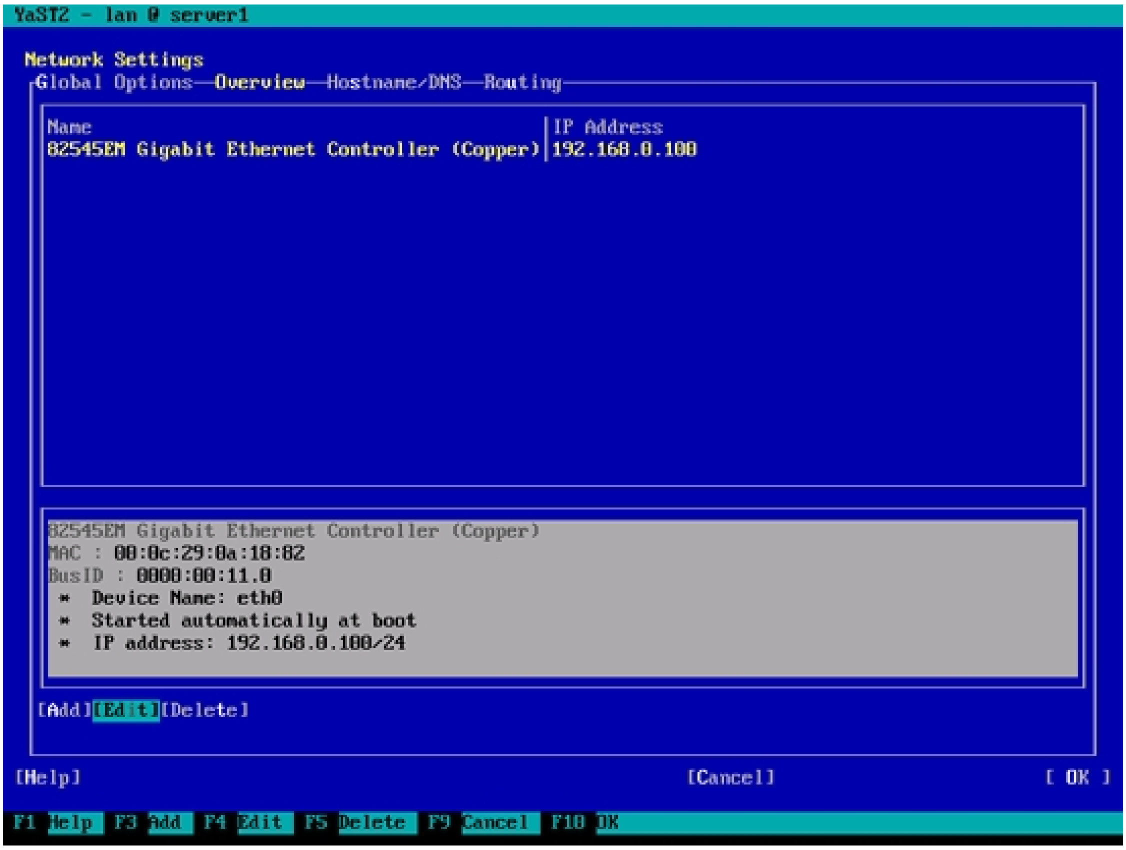
```
yast2
```

Go to `Network Devices > Network Settings`:



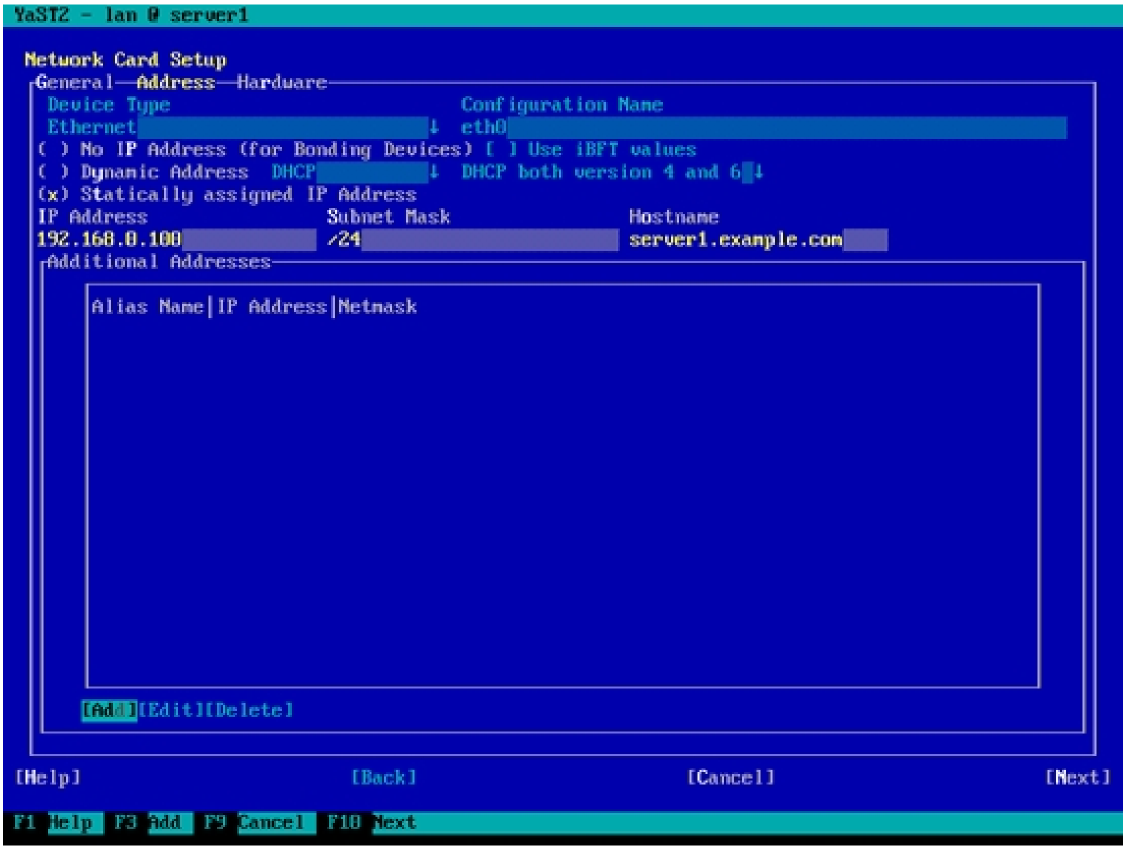
```
YaST2 - menu @ server1

YaST2 Control Center

Software                 DSL
Hardware                 ISDN
System                   Modem
Network Devices          Network Settings
Network Services
Novell AppArmor
Security and Users
Virtualization
Support
Miscellaneous

[Help]                                          [Quit]

F1 Help F9 Quit
```

Mark the current network interface and select `[Edit]`:



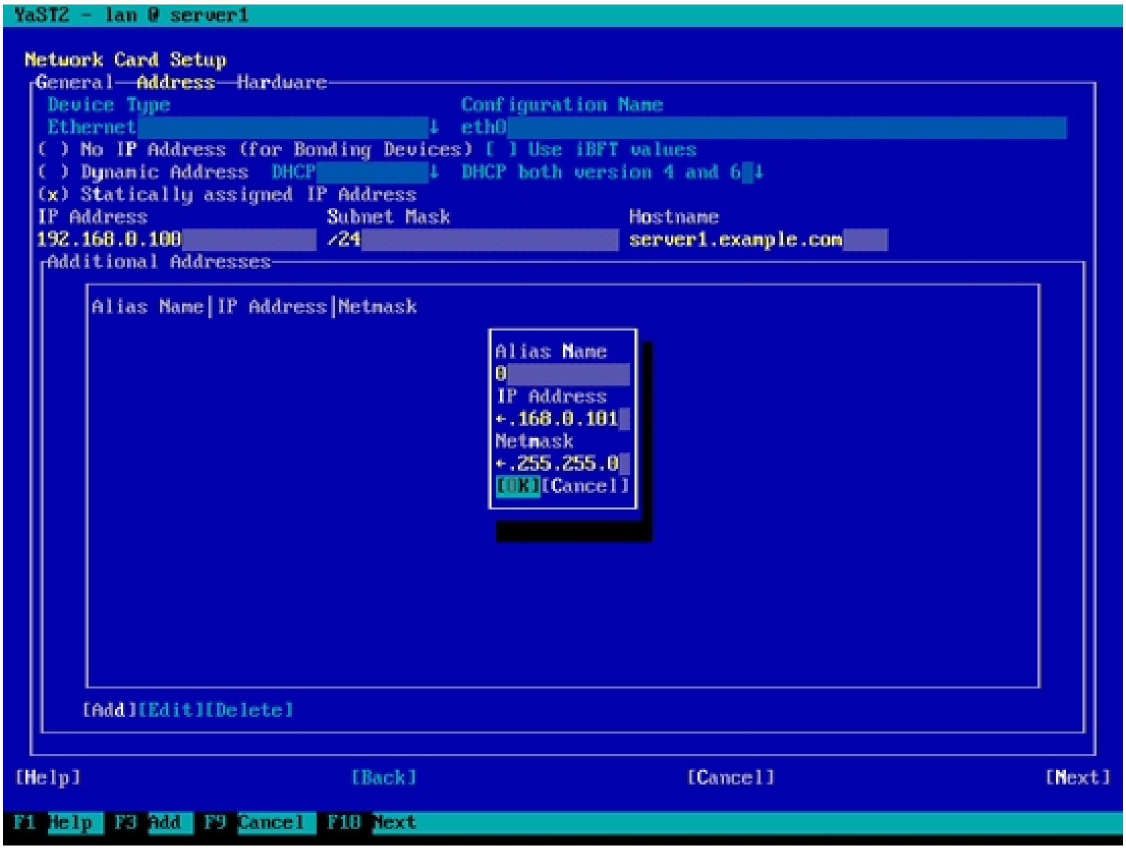
```
YaST2 - lan @ server1

 Network Settings
 Global Options—Overview—Hostname/DNS—Routing

  Name                                       |IP Address
  82545EM Gigabit Ethernet Controller (Copper)|192.168.0.100

  82545EM Gigabit Ethernet Controller (Copper)
  MAC : 00:0c:29:0a:18:02
  BusID : 0000:00:11.0
   *  Device Name: eth0
   *  Started automatically at boot
   *  IP address: 192.168.0.100/24

 [Add][Edit][Delete]

[Help]                                         [Cancel]                        [ OK ]
F1 Help  F3 Add  F4 Edit  F5 Delete  F9 Cancel  F10 OK
```

In the *Additional Addresses* box, select *[Add]*:



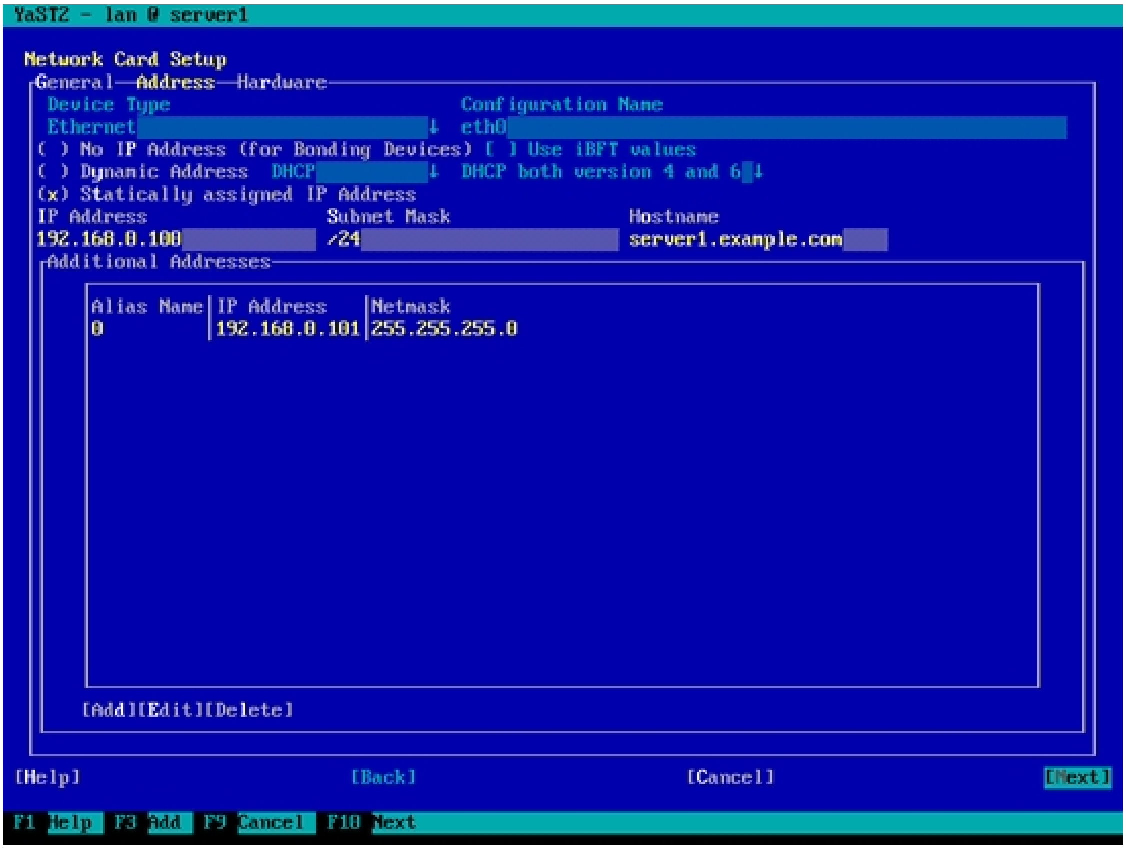
```
YaST2 - lan @ server1

 Network Card Setup
 ┌General─Address─Hardware──────────────────────────────────────────────────┐
 │ Device Type                          Configuration Name                    │
 │ Ethernet                        ↓    eth0                                  │
 │( ) No IP Address (for Bonding Devices) [ ] Use iBFT values                 │
 │( ) Dynamic Address  DHCP        ↓    DHCP both version 4 and 6 ↓           │
 │(x) Statically assigned IP Address                                          │
 │IP Address               Subnet Mask              Hostname                  │
 │192.168.0.100            /24                      server1.example.com       │
 │┌Additional Addresses─────────────────────────────────────────────────────┐ │
 ││   ┌──────────────────────────────────────────────────────────────────┐  │ │
 ││   │Alias Name│IP Address│Netmask                                     │  │ │
 ││   │                                                                  │  │ │
 ││   └──────────────────────────────────────────────────────────────────┘  │ │
 ││   [Add][Edit][Delete]                                                   │ │
 │└─────────────────────────────────────────────────────────────────────────┘ │
 └────────────────────────────────────────────────────────────────────────────┘
[Help]                        [Back]                     [Cancel]                   [Next]
F1 Help  F3 Add  F9 Cancel  F10 Next
```

Fill in *0* in the *Alias Name* field (this translates to the virtual network interface *eth0:0*; if you want to add a third, fourth, etc. IP address later on, you'd use *1*, *2*, etc. in this field - this would translate to *eth0:1*, *eth0:2*, and so on), *192.168.0.101* in the *IP Address* field, and *255.255.255.0* in the *Netmask* field (in most cases the netmask is the same as for *eth0*; if you are unsure about the network settings, you can use this network calculator: ***http://subnetmask.info/***). Then select *[OK]*:



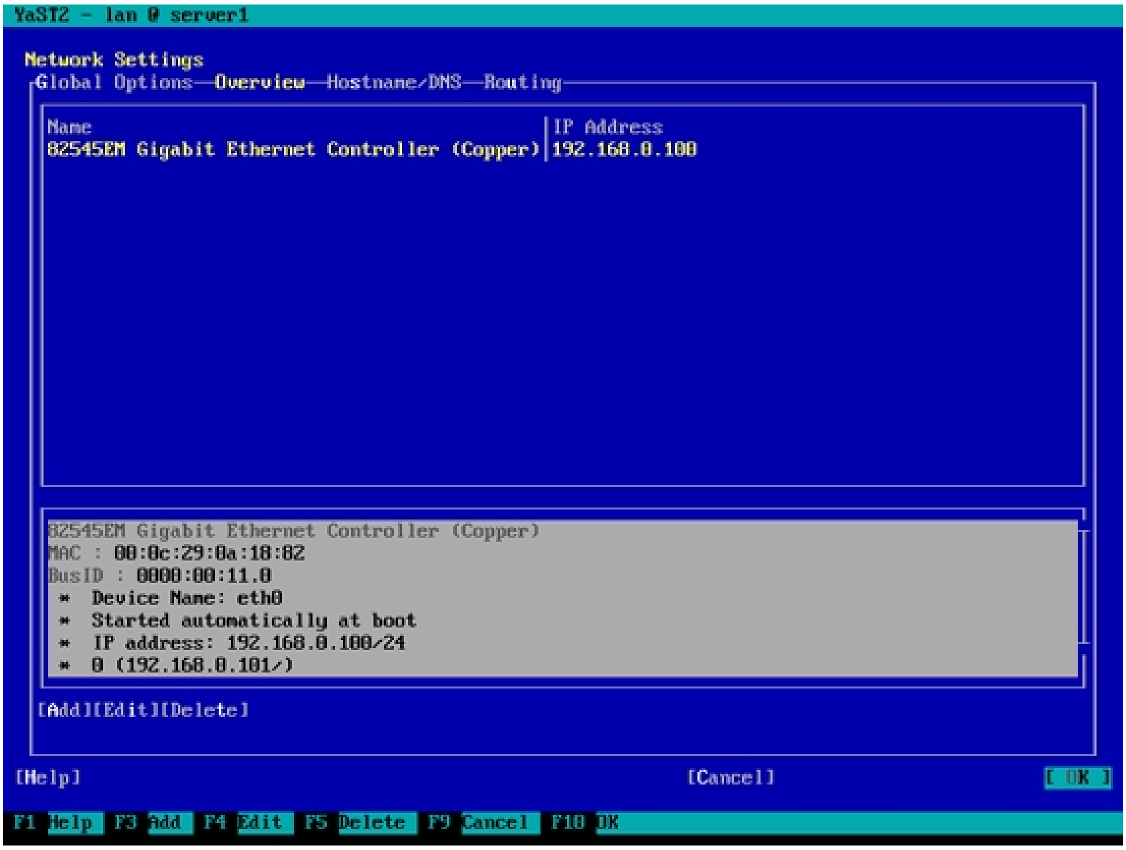
```
YaST2 - lan @ server1

Network Card Setup
┌General──Address──Hardware──────────────────────────────────────────────┐
 Device Type                            Configuration Name
 Ethernet                          ↓    eth0
( ) No IP Address (for Bonding Devices) [ ] Use iBFT values
( ) Dynamic Address  DHCP          ↓    DHCP both version 4 and 6 ↓
(x) Statically assigned IP Address
IP Address               Subnet Mask                 Hostname
192.168.0.100            /24                         server1.example.com
┌Additional Addresses──────────────────────────────────────────────────┐
    Alias Name│IP Address│Netmask
                                          Alias Name
                                          0
                                          IP Address
                                          ←.168.0.101
                                          Netmask
                                          ←.255.255.0
                                          [OK][Cancel]

   [Add][Edit][Delete]

[Help]                  [Back]                  [Cancel]                  [Next]
F1 Help  F3 Add  F9 Cancel  F10 Next
```

Select *[NEXT]* on the following screen:



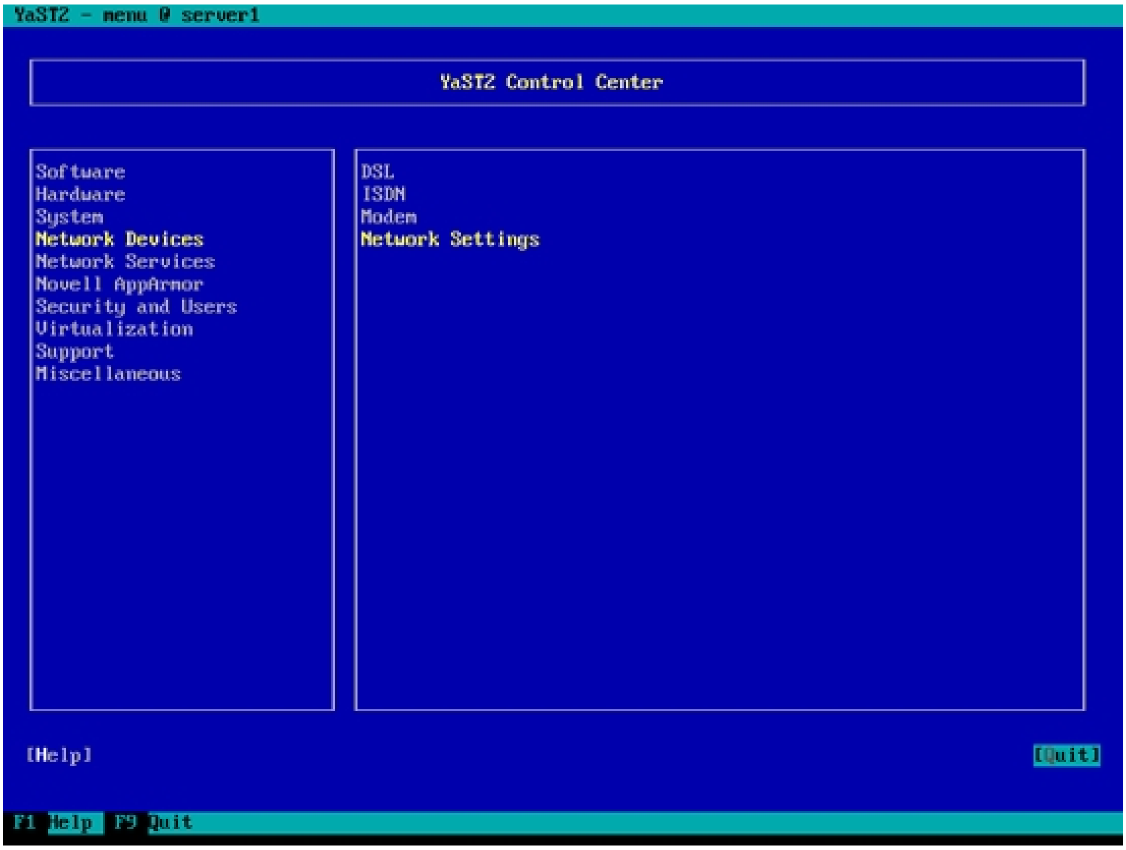
```
YaST2 - lan @ server1

 Network Card Setup
 ┌General──Address──Hardware─────────────────────────────────────────────────────┐
   Device Type                          Configuration Name
   Ethernet                          ↓  eth0
  ( ) No IP Address (for Bonding Devices) [ ] Use iBFT values
  ( ) Dynamic Address  DHCP          ↓  DHCP both version 4 and 6 ↓
  (x) Statically assigned IP Address
  IP Address                Subnet Mask               Hostname
  192.168.0.100             /24                       server1.example.com
  ┌Additional Addresses───────────────────────────────────────────────────────┐
      ┌──────────────────────────────────────────────────────────────────┐
      │Alias Name│IP Address   │Netmask
      │0         │192.168.0.101│255.255.255.0
      │
      └──────────────────────────────────────────────────────────────────┘
      [Add][Edit][Delete]
  └───────────────────────────────────────────────────────────────────────────┘
 └───────────────────────────────────────────────────────────────────────────────┘
[Help]                    [Back]                    [Cancel]                [Next]
F1 Help  F3 Add  F9 Cancel  F10 Next
```

Select *[OK]*:



```
YaST2 - lan @ server1

Network Settings
Global Options─Overview─Hostname/DNS─Routing

Name                                        │IP Address
82545EM Gigabit Ethernet Controller (Copper)│192.168.0.100

82545EM Gigabit Ethernet Controller (Copper)
MAC : 00:0c:29:0a:18:02
BusID : 0000:00:11.0
 *  Device Name: eth0
 *  Started automatically at boot
 *  IP address: 192.168.0.100/24
 *  0 (192.168.0.101/)

[Add][Edit][Delete]

[Help]                                      [Cancel]                [ OK ]
F1 Help  F3 Add  F4 Edit  F5 Delete  F9 Cancel  F10 OK
```

Now you can leave YaST by selecting `[Quit]`:



Now run

```
ifconfig
```

You should now see your new IP address in the output:

```
server1:~ # ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0C:29:0A:18:82
          inet addr:192.168.0.100  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe0a:1882/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:326 errors:0 dropped:0 overruns:0 frame:0
          TX packets:67 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:33800 (33.0 Kb)  TX bytes:7555 (7.3 Kb)

eth0:0    Link encap:Ethernet  HWaddr 00:0C:29:0A:18:82
          inet addr:192.168.0.101  Bcast:192.168.0.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:100 (100.0 b)  TX bytes:100 (100.0 b)

server1:~ #
```

5.19 How To Build A PureFTPd Debian Package For OpenVZ Virtual Machines (Without Capabilities Enabled)

The PureFTPd package that comes with Debian/Ubuntu does not start in an OpenVZ virtual machine as it is compiled with "capabilities". This tutorial describes the steps to build a PureFTPd Debian package with capabilities disabled:

Make a temporary directory:

```
mkdir /usr/src/pure-ftpd

cd /usr/src/pure-ftpd
```

Download the source package for PureFTPd:

```
apt-get source pure-ftpd
```

```
apt-get build-dep pure-ftpd
```

Edit the `rules` file and add the switch `-&#8211;without-capabilities`:

```
cd pure-ftpd-1.0.21/debian

vi rules
```

Change the line:

```
optflags=--with-everything --with-largefile --with-pam --with-privsep --with-tls
```

to (one line!):

```
optflags=--with-everything --with-largefile --with-pam --with-privsep --with-tls --without-capabilities
```

Build the Debian (.deb) package...

```
cd ..
```

```
dpkg-buildpackage -uc -b
```

... and install it:

```
cd ..

dpkg -i pure-ftpd-common_1.0.21-11.4_all.deb pure-ftpd-mysql_1.0.21-11.4_i386.deb

/etc/init.d/pure-ftpd-mysql restart
```

To prevent that apt overwrites these manually compiled packages with the default packages from the Debian repositories, execute these commands:

```
echo 'pure-ftpd-common hold' | dpkg --set-selections

echo 'pure-ftpd-mysql hold' | dpkg --set-selections
```

If you have root access to the OpenVZ host system, ***instead of compiling a new PureFTPd package***, you can do this on the host system (I'm assuming that the ID of the OpenVZ container is `101` - replace it with the correct `VPSID` on your system):

```
VPSID=101

for CAP in CHOWN DAC_READ_SEARCH SETGID SETUID NET_BIND_SERVICE NET_ADMIN SYS_CHROOT SYS_NICE
CHOWN DAC_READ_SEARCH SETGID SETUID NET_BIND_SERVICE NET_ADMIN SYS_CHROOT SYS_NICE



do

  vzctl set $VPSID --capability ${CAP}:on --save

done
```

5.20 How To Display Hidden Files With PureFTPd On Debian And Ubuntu Linux

If hidden files (files that start with a dot like .htaccess, .bash_history, .profile or .ssh) are not displayed in your FTP client, then they are most likely disabled in the FTP server. To enable hidden files in PureFTPd on Debian and Ubuntu Linux, execute this command...

```
echo "yes" > /etc/pure-ftpd/conf/DisplayDotFiles
```

... and then restart PureFTPd:

```
/etc/init.d/pure-ftpd-mysql restart
```

5.21 PureFTPd Does Not Show More Than 2,000 Files On Debian And Ubuntu

The PureFTPd daemon by default has a recursion limit of 2,000 files, this prevents the server from showing more than 2,000 files when you browse a directory with an FTP client. To expand this limit to e.g. 5,000 files, create or edit the file `/etc/pure-ftpd/conf/LimitRecursion` and add the line `5000 500`:

```
echo "5000 500" > /etc/pure-ftpd/conf/LimitRecursion
```

Then restart PureFTPd:

```
/etc/init.d/pure-ftpd-mysql restart
```

5.22 How To Speed Up Logins In PureFTPd On Debian Or Ubuntu Linux By Disabling Name Resolving

If you experience problems with slow logins in PureFTPd, this is often caused by a problem with the resolving of the client's hostname. This happens e.g. when you run an FTP server in your intranet and the hostname of the client computer does not exist in DNS. To disable name resolving in PureFTPd, run the command:

```
echo 'yes' > /etc/pure-ftpd/conf/DontResolve
```

Then restart PureFTPd:

```
/etc/init.d/pure-ftpd-mysql restart
```

Disabling name resolving also fixes the following error message:

```
Jul 24 16:26:28 ispconfig pure-ftpd: (?@?) [ERROR] Sorry, invalid address given
```

5.23 How To Enable Verbose Logging In PureFTPd On Debian And Ubuntu Linux

To turn on verbose logging (e.g. to debug FTP connection or authentication problems) in PureFTPd FTP server on Debian and Ubuntu Linux, execute the following command as root user on the shell:

```
echo 'yes' > /etc/pure-ftpd/conf/VerboseLog
```

Then restart PureFTPd:

```
/etc/init.d/pure-ftpd-mysql restart
```

The debug output will be logged to syslog. To view the log content, execute:

```
tail -n 100 /var/log/syslog
```

To disable verbose logging, execute these commands:

```
rm -f /etc/pure-ftpd/conf/VerboseLog

/etc/init.d/pure-ftpd-mysql restart
```

5.24 How To Enable FTPS For PureFTPd On Debian And Ubuntu Linux

To enable FTPS for PureFTPd on Debian and Ubuntu, run:

```
echo 1 > /etc/pure-ftpd/conf/TLS
```

```
mkdir -p /etc/ssl/private/

openssl req -x509 -nodes -days 7300 -newkey rsa:2048 -keyout /etc/ssl/private/pure-ftpd.pem
-out /etc/ssl/private/pure-ftpd.pem
```

```
chmod 600 /etc/ssl/private/pure-ftpd.pem
```

Then restart PureFTPd:

```
/etc/init.d/pure-ftpd-mysql restart
```

5.25 How Can I Access SquirrelMail From My Web Sites?

5.25.1 Apache

This guide explains how to configure SquirrelMail on a Debian/Ubuntu server so that you can use it from within your web sites (created through ISPConfig).

We will configure SquirrelMail so that you can use it from within your web sites by using the `/squirrelmail` or

/webmail aliases. So if your website is *www.example.com*, you will be able to access SquirrelMail using *www.example.com/squirrelmail* or *www.example.com/webmail*.

SquirrelMail's Apache configuration is in the file */etc/squirrelmail/apache.conf*, but this file isn't loaded by Apache because it is not in the */etc/apache2/conf.d/* directory. Therefore we create a symlink called *squirrelmail.conf* in the */etc/apache2/conf.d/* directory that points to */etc/squirrelmail/apache.conf* and reload Apache afterwards:

```
cd /etc/apache2/conf.d/

ln -s ../../squirrelmail/apache.conf squirrelmail.conf

/etc/init.d/apache2 reload
```

Now open */etc/apache2/conf.d/squirrelmail.conf*...

```
vi /etc/apache2/conf.d/squirrelmail.conf
```

... and add the following lines to the *<Directory /usr/share/squirrelmail></Directory>* container that make sure that mod_php is used for accessing SquirrelMail, regardless of what PHP mode you select for your website in ISPConfig:

```
[...]
<Directory /usr/share/squirrelmail>
 Options FollowSymLinks
 <IfModule mod_php5.c>
  AddType application/x-httpd-php .php
  php_flag magic_quotes_gpc Off

  php_flag track_vars On
  php_admin_flag allow_url_fopen Off
  php_value include_path .
  php_admin_value upload_tmp_dir /var/lib/squirrelmail/tmp
  php_admin_value open_basedir /usr/share/squirrelmail:/etc/squirrelmail:/var/lib/squirrelmail:/etc/hostname:/etc/mailname
  php_flag register_globals off
 </IfModule>

 <IfModule mod_dir.c>
  DirectoryIndex index.php
 </IfModule>

 # access to configtest is limited by default to prevent information leak
 <Files configtest.php>
  order deny,allow
  deny from all
  allow from 127.0.0.1
 </Files>
```

```
</Directory>

[...]
```

Create the directory `/var/lib/squirrelmail/tmp`...

```
mkdir /var/lib/squirrelmail/tmp
```

... and make it owned by the user `www-data`:

```
chown www-data /var/lib/squirrelmail/tmp
```

Reload Apache again:

```
/etc/init.d/apache2 reload
```

That's it already - `/etc/apache2/conf.d/squirrelmail.conf` defines an alias called `/squirrelmail` that points to SquirrelMail's installation directory `/usr/share/squirrelmail`.

You can now access SquirrelMail from your web site as follows:

`http://www.example.com/squirrelmail`

You can also access it from the ISPConfig control panel vhost as follows (this doesn't need any configuration in ISPConfig):

`http(s)://server1.example.com:8080/squirrelmail`

If you'd like to use the alias `/webmail` instead of `/squirrelmail`, simply open `/etc/apache2/conf.d/squirrelmail.conf`...

```
vi /etc/apache2/conf.d/squirrelmail.conf
```

... and add the line `Alias /webmail /usr/share/squirrelmail`:

```
Alias /squirrelmail /usr/share/squirrelmail
Alias /webmail /usr/share/squirrelmail
[...]
```

Then reload Apache:

```
/etc/init.d/apache2 reload
```

Now you can access Squirrelmail as follows:

`http://www.example.com/webmail`
`http(s)://server1.example.com:8080/webmail`

If you'd like to define a vhost like *webmail.example.com* where your users can access SquirrelMail, you'd have to add the following vhost configuration to */etc/apache2/conf.d/squirrelmail.conf*:

```
vi /etc/apache2/conf.d/squirrelmail.conf
```

```
[...]
<VirtualHost 1.2.3.4:80>

  DocumentRoot /usr/share/squirrelmail
  ServerName webmail.example.com
</VirtualHost>
```

Make sure you replace *1.2.3.4* with the correct IP address of your server. Of course, there must be a DNS record for *webmail.example.com* that points to the IP address that you use in the vhost configuration. Also make sure that the vhost *webmail.example.com* does not exist in ISPConfig (otherwise both vhosts will interfere with each other!).

Now reload Apache...

```
/etc/init.d/apache2 reload
```

... and you can access SquirrelMail under *http://webmail.example.com*!

5.25.2 nginx

The ISPConfig apps vhost on port 8081 for nginx comes with a SquirrelMail configuration, so you can use *http://server1.example.com:8081/squirrelmail* or *http://server1.example.com:8081/webmail* to access SquirrelMail.

If you want to use a */webmail* or */squirrelmail* alias that you can use from your web sites, this is a bit more complicated than for Apache because nginx does not have global aliases (i.e., aliases that can be defined for all vhosts). Therefore you have to define these aliases for **each** vhost from which you want to access SquirrelMail.

To do this, paste the following into the *nginx Directives* field on the *Options* tab of the web site in ISPConfig (see chapter ***4.6.1.1 Website***):

```
        location /squirrelmail {
               root /usr/share/;
               index index.php index.html index.htm;
               location ~ ^/squirrelmail/(.+\.php)$ {
                       try_files $uri =404;
                       root /usr/share/;
                       fastcgi_pass 127.0.0.1:9000;
                       fastcgi_index index.php;
                       fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
                       include /etc/nginx/fastcgi_params;
```

```
                fastcgi_buffer_size 128k;
                fastcgi_buffers 256 4k;
                fastcgi_busy_buffers_size 256k;
                fastcgi_temp_file_write_size 256k;
                fastcgi_intercept_errors on;
        }
        location ~* ^/squirrelmail/(.+\.(jpg|jpeg|gif|css|png|js|ico|html|xml|txt))$ {
                root /usr/share/;
        }
}
location /webmail {
        rewrite ^/* /squirrelmail last;
}
```

If you use http**s** instead of http for your vhost, you should add the line *`fastcgi_param HTTPS on;`* to your SquirrelMail configuration like this:

```
location /squirrelmail {
        root /usr/share/;
        index index.php index.html index.htm;
        location ~ ^/squirrelmail/(.+\.php)$ {
                try_files $uri =404;
                root /usr/share/;
                fastcgi_pass 127.0.0.1:9000;
                fastcgi_param HTTPS on; # <-- add this line
                fastcgi_index index.php;
                fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
                include /etc/nginx/fastcgi_params;
                fastcgi_buffer_size 128k;
                fastcgi_buffers 256 4k;
                fastcgi_busy_buffers_size 256k;
                fastcgi_temp_file_write_size 256k;
                fastcgi_intercept_errors on;
        }
        location ~* ^/squirrelmail/(.+\.(jpg|jpeg|gif|css|png|js|ico|html|xml|txt))$ {
                root /usr/share/;
        }
}
location /webmail {
        rewrite ^/* /squirrelmail last;
}
```

If you use both http and https for your vhost, you need to add the following section to the *`http {}`* section in *`/etc/nginx/nginx.conf`* (before any include lines) which determines if the visitor uses http or https and sets the *`$fastcgi_https`* variable (which we will use in our SquirrelMail configuration) accordingly:

```
vi /etc/nginx/nginx.conf
```

```
[...]
http {
[...]
    ## Detect when HTTPS is used
    map $scheme $fastcgi_https {
      default off;
      https on;

    }
[...]
}
[...]
```

Don't forget to reload nginx afterwards:

```
/etc/init.d/nginx reload
```

Then go to the `nginx Directives` field again, and instead of `fastcgi_param HTTPS on;` you add the line `fastcgi_param HTTPS $fastcgi_https;` so that you can use SquirrelMail for both http and https requests:

```
    location /squirrelmail {
        root /usr/share/;
        index index.php index.html index.htm;
        location ~ ^/squirrelmail/(.+\.php)$ {
            try_files $uri =404;
            root /usr/share/;
            fastcgi_pass 127.0.0.1:9000;
            fastcgi_param HTTPS $fastcgi_https; # <-- add this line
            fastcgi_index index.php;
            fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
            include /etc/nginx/fastcgi_params;
            fastcgi_buffer_size 128k;
            fastcgi_buffers 256 4k;
            fastcgi_busy_buffers_size 256k;
            fastcgi_temp_file_write_size 256k;
            fastcgi_intercept_errors on;
        }
        location ~* ^/squirrelmail/(.+\.(jpg|jpeg|gif|css|png|js|ico|html|xml|txt))$ {
            root /usr/share/;
        }
    }
    location /webmail {
        rewrite ^/* /squirrelmail last;
    }
```

5.26 How Can I Access phpMyAdmin From My Web Sites?

5.26.1 Apache

This guide explains how to configure phpMyAdmin on a Debian/Ubuntu server so that you can use it from within your web sites (created through ISPConfig).

We will configure phpMyAdmin so that you can use it from within your web sites by using the */phpmyadmin* or */phpMyAdmin* aliases. So if your website is *www.example.com*, you will be able to access phpMyAdmin using *www.example.com/phpmyadmin* or *www.example.com/phpMyAdmin*.

phpMyAdmin's Apache configuration is in the file */etc/phpmyadmin/apache.conf*. Normally, there should be a symlink called *phpmyadmin.conf* in the */etc/apache2/conf.d/* directory that points to */etc/phpmyadmin/apache.conf*:

```
ls -l /etc/apache2/conf.d/
```

```
root@server1:~# ls -l /etc/apache2/conf.d/
total 20
-rw-r--r-- 1 root root  237 2011-09-01 11:26 apache2-doc
-rw-r--r-- 1 root root  269 2011-09-01 11:30 charset
lrwxrwxrwx 1 root root   45 2011-09-16 12:22 javascript-common.conf -> /etc/javascri
pt-common/javascript-common.conf
-rw-r--r-- 1 root root 3296 2011-09-01 11:30 localized-error-pages
lrwxrwxrwx 1 root root   24 2011-10-18 12:04 mailman.conf -> /etc/mailman/apache.con
f
-rw-r--r-- 1 root root  143 2011-09-01 11:30 other-vhosts-access-log
lrwxrwxrwx 1 root root   28 2011-09-16 12:23 phpmyadmin.conf -> ../../phpmyadmin/apa
che.conf
-rw-r--r-- 1 root root 1424 2011-09-01 11:30 security
root@server1:~#
```

If that symlink does not exist, create it as follows and reload Apache afterwards:

```
cd /etc/apache2/conf.d/

ln -s ../../phpmyadmin/apache.conf phpmyadmin.conf

/etc/init.d/apache2 reload
```

Now open */etc/apache2/conf.d/phpmyadmin.conf*...

```
vi /etc/apache2/conf.d/phpmyadmin.conf
```

... and add the line *Alias /phpMyAdmin /usr/share/phpmyadmin*:

```
# phpMyAdmin default Apache configuration

Alias /phpmyadmin /usr/share/phpmyadmin
Alias /phpMyAdmin /usr/share/phpmyadmin
[...]
```

Then reload Apache:

```
/etc/init.d/apache2 reload
```

Now you can access phpMyAdmin as follows:

http://www.example.com/phpmyadmin
http://www.example.com/phpMyAdmin
http(s)://server1.example.com:8080/phpmyadmin
http(s)://server1.example.com:8080/phpMyAdmin

5.26.2 nginx

The ISPConfig apps vhost on port 8081 for nginx comes with a phpMyAdmin configuration, so you can use *http://server1.example.com:8081/phpmyadmin* or *http://server1.example.com:8081/phpMyAdmin* to access phpMyAdmin.

If you want to use a */phpmyadmin* or */phpMyAdmin* alias that you can use from your web sites, this is a bit more complicated than for Apache because nginx does not have global aliases (i.e., aliases that can be defined for all vhosts). Therefore you have to define these aliases for ***each*** vhost from which you want to access phpMyAdmin.

To do this, paste the following into the *nginx Directives* field on the *Options* tab of the web site in ISPConfig (see chapter ***4.6.1.1 Website***):

```
    location /phpmyadmin {
        root /usr/share/;
        index index.php index.html index.htm;
        location ~ ^/phpmyadmin/(.+\.php)$ {
            try_files $uri =404;
            root /usr/share/;
            fastcgi_pass 127.0.0.1:9000;
            fastcgi_index index.php;
            fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
            include /etc/nginx/fastcgi_params;
            fastcgi_buffer_size 128k;
            fastcgi_buffers 256 4k;
            fastcgi_busy_buffers_size 256k;
            fastcgi_temp_file_write_size 256k;
```

```
                fastcgi_intercept_errors on;
        }
        location ~* ^/phpmyadmin/(.+\.(jpg|jpeg|gif|css|png|js|ico|html|xml|txt))$ {
                root /usr/share/;
        }
}
location /phpMyAdmin {
        rewrite ^/* /phpmyadmin last;
}
```

If you use http**s** instead of http for your vhost, you should add the line *`fastcgi_param HTTPS on;`* to your phpMyAdmin configuration like this:

```
location /phpmyadmin {
        root /usr/share/;
        index index.php index.html index.htm;
        location ~ ^/phpmyadmin/(.+\.php)$ {
                try_files $uri =404;
                root /usr/share/;
                fastcgi_pass 127.0.0.1:9000;
                fastcgi_param HTTPS on; # <-- add this line
                fastcgi_index index.php;
                fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
                include /etc/nginx/fastcgi_params;
                fastcgi_buffer_size 128k;
                fastcgi_buffers 256 4k;
                fastcgi_busy_buffers_size 256k;
                fastcgi_temp_file_write_size 256k;
                fastcgi_intercept_errors on;
        }
        location ~* ^/phpmyadmin/(.+\.(jpg|jpeg|gif|css|png|js|ico|html|xml|txt))$ {
                root /usr/share/;
        }
}
location /phpMyAdmin {
        rewrite ^/* /phpmyadmin last;
}
```

If you use both http and https for your vhost, you need to add the following section to the *`http {}`* section in *`/etc/nginx/nginx.conf`* (before any include lines) which determines if the visitor uses http or https and sets the *`$fastcgi_https`* variable (which we will use in our phpMyAdmin configuration) accordingly:

```
vi /etc/nginx/nginx.conf
```

```
[...]
```

```
http {
[...]
        ## Detect when HTTPS is used
        map $scheme $fastcgi_https {
          default off;
          https on;

        }
[...]
}
[...]
```

Don't forget to reload nginx afterwards:

```
/etc/init.d/nginx reload
```

Then go to the *nginx Directives* field again, and instead of *fastcgi_param HTTPS on;* you add the line *fastcgi_param HTTPS $fastcgi_https;* so that you can use phpMyAdmin for both http and https requests:

```
        location /phpmyadmin {
               root /usr/share/;
               index index.php index.html index.htm;
               location ~ ^/phpmyadmin/(.+\.php)$ {
                       try_files $uri =404;
                       root /usr/share/;
                       fastcgi_pass 127.0.0.1:9000;
                       fastcgi_param HTTPS $fastcgi_https; # <-- add this line
                       fastcgi_index index.php;
                       fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
                       include /etc/nginx/fastcgi_params;
                       fastcgi_buffer_size 128k;
                       fastcgi_buffers 256 4k;
                       fastcgi_busy_buffers_size 256k;
                       fastcgi_temp_file_write_size 256k;
                       fastcgi_intercept_errors on;
               }
               location ~* ^/phpmyadmin/(.+\.(jpg|jpeg|gif|css|png|js|ico|html|xml|txt))$ {
                       root /usr/share/;
               }
        }
        location /phpMyAdmin {
               rewrite ^/* /phpmyadmin last;
        }
```

5.27 How Can I Access Mailman From My Web Sites?

5.27.1 Apache

For Debian/Ubuntu, the required Apache configuration to access Mailman from your web sites is described in chapter ***3.1.2 Mailman***:

Enable the Mailman Apache configuration:

```
ln -s /etc/mailman/apache.conf /etc/apache2/conf.d/mailman.conf
```

This defines the alias */cgi-bin/mailman/* for all Apache vhosts, which means you can access the Mailman admin interface for a list at *http://<vhost>/cgi-bin/mailman/admin/<listname>*, and the web page for users of a mailing list can be found at *http://<vhost>/cgi-bin/mailman/listinfo/<listname>*.

Restart Apache afterwards:

```
/etc/init.d/apache2 restart
```

5.27.2 nginx

The ISPConfig apps vhost on port 8081 for nginx comes with a Mailman configuration, so you can use *http://server1.example.com:8081/cgi-bin/mailman/admin/<listname>* or *http://server1.example.com:8081/cgi-bin/mailman/listinfo/<listname>* to access Mailman.

If you want to use Mailman from your web sites, this is a bit more complicated than for Apache because nginx does not have global aliases (i.e., aliases that can be defined for all vhosts). Therefore you have to define these aliases for ***each*** vhost from which you want to access Mailman.

To do this, paste the following into the *nginx Directives* field on the *Options* tab of the web site in ISPConfig (see chapter ***4.6.1.1 Website***):

```
location /cgi-bin/mailman {
       root /usr/lib/;
       fastcgi_split_path_info (^/cgi-bin/mailman/[^/]*)(.*)$;
       include /etc/nginx/fastcgi_params;
       fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
       fastcgi_param PATH_INFO $fastcgi_path_info;
       fastcgi_param PATH_TRANSLATED $document_root$fastcgi_path_info;
       fastcgi_intercept_errors on;
       fastcgi_pass unix:/var/run/fcgiwrap.socket;
}

location /images/mailman {
       alias /usr/share/images/mailman;
}
```

```
location /pipermail {
      alias /var/lib/mailman/archives/public;
      autoindex on;
}
```

This defines the alias `/cgi-bin/mailman/` for your vhost, which means you can access the Mailman admin interface for a list at `http://<vhost>/cgi-bin/mailman/admin/<listname>`, and the web page for users of a mailing list can be found at `http://<vhost>/cgi-bin/mailman/listinfo/<listname>`.

Under `http://<vhost>/pipermail` you can find the mailing list archives.

6 Security Considerations

6.1 How Do I Disable Certain PHP Functions?

Debian/Ubuntu:

Debian and Ubuntu systems come with multiple php.ini files (`/etc/php5/apache2/php.ini` for mod_php, `/etc/php5/cgi/php.ini` for Fast-CGI and CGI, and `/etc/php5/cli/php.ini` for command-line PHP). You can use the `disable_functions` = directive to disable potentially dangerous PHP functions such as `exec` , `passthru`, `popen`, `ini_set`, `system`, but only in `/etc/php5/apache2/php.ini` and `/etc/php5/cgi/php.ini`, e.g. as follows:

```
[...]
; This directive allows you to disable certain functions for security reasons.
; It receives a comma-delimited list of function names. This directive is
; *NOT* affected by whether Safe Mode is turned On or Off.
disable_functions = exec,passthru,popen,ini_set,system,show_source,shell_exec,proc_open,phpinfo
[...]
```

If you modify `/etc/php5/apache2/php.ini`, please do not forget to restart Apache afterwards:

```
/etc/init.d/apache2 restart
```

Please note that you must not disable any functions in the php.ini file for the command line, `/etc/php5/cli/php.ini`, because if you do, ISPConfig will not work correctly anymore!

Fedora/CentOS/OpenSUSE:

These distributions come with just one php.ini file which is used by mod_php, Fast-CGI/CGI, and command-line PHP. Therefore we cannot disable PHP functions in that php.ini file because that would also affect command.line PHP, and ISPConfig would not work anymore.

But you can disable functions individually for each web site in ISPConfig, either through the *Custom php.ini settings* field (if you use Fast-CGI, CGI, or SuPHP), or through the *Apache directives* field (if you use Mod-PHP), both on the Options tab of a web site (see chapter ***4.6.1.1 Website***).

In the *Custom php.ini settings* field, you can place something like

```
disable_functions =
exec,passthru,popen,ini_set,system,show_source,shell_exec,proc_open,phpinfo
```

In the *Apache directives* field, you can use the *php_flag disable_functions* directive, one directive per function, e.g. as follows:

```
php_flag disable_functions exec
php_flag disable_functions passthru
php_flag disable_functions popen
php_flag disable_functions ini_set
php_flag disable_functions system
php_flag disable_functions show_source
php_flag disable_functions shell_exec
php_flag disable_functions proc_open
php_flag disable_functions phpinfo
```

6.2 Enabling SSL For The ISPConfig Web Interface

(These instructions are for Debian/Ubuntu.)

Since ISPConfig 3.0.4, you can enable SSL for the ISPConfig web interface during installation - just press *ENTER* when you see this question:

```
Do you want a secure (SSL) connection to the ISPConfig web interface (y,n) [y]:
<-- ENTER
```

For earlier ISPConfig versions (or if you told the ISPConfig 3.0.4 installer you want to use http, but have changed your mind and want to use https now) do this to enable SSL for the ISPConfig web interface:

Make the directory for the SSL certificate:

```
mkdir /etc/apache2/ssl

cd /etc/apache2/ssl
```

Create the SSL certificate files:

```
openssl genrsa -des3 -out ispserver.key 4096

openssl req -new -key ispserver.key -out ispserver.csr

openssl x509 -req -days 3650 -in ispserver.csr \
-signkey ispserver.key -out ispserver.crt

openssl rsa -in ispserver.key -out ispserver.key.insecure

mv ispserver.key ispserver.key.secure

mv ispserver.key.insecure ispserver.key
```

Enable the mod_ssl module:

```
a2enmod ssl
```

Edit the ISPConfig vhost file...

```
vi /etc/apache2/sites-available/ispconfig.vhost
```

... and insert the following lines between the `<VirtualHost ...></VirtualHost>` tags:

```
SSLEngine On
SSLCertificateFile /etc/apache2/ssl/ispserver.crt
SSLCertificateKeyFile /etc/apache2/ssl/ispserver.key
```

Restart Apache2:

```
/etc/init.d/apache2 restart
```

The ISPConfig control panel login is now reachable on port 8080 by https.

6.3 Using SuExec For The ISPConfig Web Interface

(These instructions are for Debian/Ubuntu.)

Before you do this, you should close all browser windows where you use ISPConfig because afterwards the current sessions will be invalid.

Open `/etc/apache2/sites-available/ispconfig.vhost`...

```
vi /etc/apache2/sites-available/ispconfig.vhost
```

... and comment out the `<IfModule mod_php5.c>...</IfModule>` section:

```
[...]
#  <IfModule mod_php5.c>
#    DocumentRoot /usr/local/ispconfig/interface/web/
#    AddType application/x-httpd-php .php
#   <Directory /usr/local/ispconfig/interface/web>
#     Options FollowSymLinks
#     AllowOverride None
#     Order allow,deny
#     Allow from all
#        php_value magic_quotes_gpc       0
#   </Directory>
# </IfModule>
[...]
```

... and restart Apache:

```
/etc/init.d/apache2 restart
```

6.4 What Are Secure Settings For Web Sites Created Through ISPConfig?

- Use Fast-CGI, CGI, or SuPHP instead of Mod-PHP.
- Always activate suExec if you use Fast-CGI or CGI.
- Enable only the features that you really need. For example, if you don't need SSI for a web site, then don't enable it.

6.5 How Do I Make fail2ban Monitor Additional Services?

(These instructions are for Debian/Ubuntu.)

By default, fail2ban monitors the SSH service and tries to block users with too many failed login attempts for this service. But fail2ban can also be used to monitor additional services and block users with too many failed login attempts. This tutorial has more details about it: ***http://www.howtoforge.com/fail2ban_debian_etch***

6.5.1 PureFTPd

Open `/etc/fail2ban/jail.local`:

```
vi /etc/fail2ban/jail.local
```

Add the following section at the end:

```
[...]
```

```
[pureftpd]

enabled  = true
port     = ftp
filter   = pureftpd
logpath  = /var/log/syslog
maxretry = 3
```

Then create the file `/etc/fail2ban/filter.d/pureftpd.conf`:

```
vi /etc/fail2ban/filter.d/pureftpd.conf
```

```
[Definition]
failregex = .*pure-ftpd: \(.*@<HOST>\) \[WARNING\] Authentication failed for user.*
ignoreregex =
```

Restart fail2ban:

```
/etc/init.d/fail2ban restart
```

6.5.2 SASL

Open `/etc/fail2ban/jail.local`...

```
vi /etc/fail2ban/jail.local
```

... and make sure you have the following section in it:

```
[...]
[sasl]

enabled  = true
port     = smtp
filter   = sasl
logpath  = /var/log/mail.log
maxretry = 5
[...]
```

Restart fail2ban:

```
/etc/init.d/fail2ban restart
```

6.5.3 Courier

Open */etc/fail2ban/jail.local*...

```
vi /etc/fail2ban/jail.local
```

... and make sure you have the following four sections in it:

```
[...]
[courierpop3]

enabled  = true
port     = pop3
filter   = courierpop3
logpath  = /var/log/mail.log
maxretry = 5



[courierpop3s]

enabled  = true
port     = pop3s
filter   = courierpop3s
logpath  = /var/log/mail.log
maxretry = 5



[courierimap]

enabled  = true
port     = imap2
filter   = courierimap
logpath  = /var/log/mail.log
maxretry = 5



[courierimaps]

enabled  = true
port     = imaps
filter   = courierimaps
logpath  = /var/log/mail.log
maxretry = 5
[...]
```

Next create the following four files:

```
vi /etc/fail2ban/filter.d/courierpop3.conf
```

```
# Fail2Ban configuration file
#
# $Revision: 100 $
#

[Definition]

# Option:  failregex
# Notes.:  regex to match the password failures messages in the logfile. The
#          host must be matched by a group named "host". The tag "<HOST>" can
#          be used for standard IP/hostname matching and is only an alias for
#          (?:::f{4,6}:)?(?P<host>\S+)
# Values:  TEXT
#
failregex = pop3d: LOGIN FAILED.*ip=\[.*:<HOST>\]

# Option:  ignoreregex
# Notes.:  regex to ignore. If this regex matches, the line is ignored.
# Values:  TEXT
#
ignoreregex =
```

```
vi /etc/fail2ban/filter.d/courierpop3s.conf
```

```
# Fail2Ban configuration file
#
# $Revision: 100 $
#

[Definition]

# Option:  failregex
# Notes.:  regex to match the password failures messages in the logfile. The
#          host must be matched by a group named "host". The tag "<HOST>" can
#          be used for standard IP/hostname matching and is only an alias for
#          (?:::f{4,6}:)?(?P<host>\S+)
# Values:  TEXT
#
failregex = pop3d-ssl: LOGIN FAILED.*ip=\[.*:<HOST>\]

# Option:  ignoreregex
# Notes.:  regex to ignore. If this regex matches, the line is ignored.
# Values:  TEXT
```

```
#
ignoreregex =
```

```
vi /etc/fail2ban/filter.d/courierimap.conf
```

```
# Fail2Ban configuration file
#
# $Revision: 100 $
#

[Definition]

# Option:  failregex
# Notes.:  regex to match the password failures messages in the logfile. The
#          host must be matched by a group named "host". The tag "<HOST>" can
#          be used for standard IP/hostname matching and is only an alias for
#          (?:::f{4,6}:)?(?P<host>\S+)
# Values:  TEXT
#
failregex = imapd: LOGIN FAILED.*ip=\[.*:<HOST>\]

# Option:  ignoreregex
# Notes.:  regex to ignore. If this regex matches, the line is ignored.
# Values:  TEXT
#
ignoreregex =
```

```
vi /etc/fail2ban/filter.d/courierimaps.conf
```

```
# Fail2Ban configuration file
#
# $Revision: 100 $
#

[Definition]

# Option:  failregex
# Notes.:  regex to match the password failures messages in the logfile. The
#          host must be matched by a group named "host". The tag "<HOST>" can
#          be used for standard IP/hostname matching and is only an alias for
#          (?:::f{4,6}:)?(?P<host>\S+)
# Values:  TEXT
#
failregex = imapd-ssl: LOGIN FAILED.*ip=\[.*:<HOST>\]
```

```
# Option:  ignoreregex
# Notes.:  regex to ignore. If this regex matches, the line is ignored.
# Values:  TEXT
#
ignoreregex =
```

Restart fail2ban:

```
/etc/init.d/fail2ban restart
```

6.5.4 Dovecot

Open `/etc/fail2ban/jail.local`:

```
vi /etc/fail2ban/jail.local
```

Add the following section at the end:

```
[...]
[dovecot-pop3imap]

enabled = true
filter = dovecot-pop3imap
action = iptables-multiport[name=dovecot-pop3imap, port="pop3,imap", protocol=tcp]
logpath = /var/log/mail.log
maxretry = 20
findtime = 1200
bantime = 1200
```

Then create the file `/etc/fail2ban/filter.d/dovecot-pop3imap.conf`:

```
vi /etc/fail2ban/filter.d/dovecot-pop3imap.conf
```

```
[Definition]
failregex = (?: pop3-login|imap-login): (?:Authentication failure|Aborted login \(auth failed|Aborted login \(tried to use disabled|Disconnected \(auth failed).*rip=(?P<host>\S*),.*
ignoreregex =
```

Restart fail2ban:

```
/etc/init.d/fail2ban restart
```

7 Troubleshooting

7.1 How Do I Find Out What Is Wrong If ISPConfig Does Not Work?

1) Did all jobs finish? Take a look at the job queue (`Monitor > System State (All Servers) > Show Jobqueue`) (see chapter ***4.10.1.3 Show Jobqueue***). Jobs that are listed there are either ***not yet*** completed (i.e., ISPConfig is still working on them) or did not complete because of some kind of problem.

2) If there are open jobs, please check if there are messages with the status "error" in the system log (`Monitor > System State (All Servers) > Show System-Log`) (see chapter ***4.10.1.2 Show System-Log***). If there are, please try to fix the error. After you have fixed the error, please delete the error message from the system log in ISPConfig, so that ISPConfig can continue to process the open jobs.

3) If it is not clear what is causing the error, please set the log level to `Debug` under `System > System > Server Config` (see chapter ***4.9.2.2 Server Config***). After one or two minutes, there should be more detailed messages in ISPConfig's system log (`Monitor > System State (All Servers) > Show System-Log`).

4) If this still doesn't help, then go to the command line of the server on which the error happens (on multiserver systems, it is often the slave and not the master) and run (as root):

```
crontab -e
```

Comment out the `server.sh` cron job:

```
#* * * * * /usr/local/ispconfig/server/server.sh > /dev/null 2>> /var/log/ispconfig/cron.log
```

Then run the command:

```
/usr/local/ispconfig/server/server.sh
```

This will display any errors directly on the command line which should help you to fix the error. If you have fixed the error, please don't forget to uncomment the `server.sh` cron job again.