

# ISPConfig

ISPConfig is an open source hosting control panel for Linux which is capable of managing multiple servers from one control panel.

ISPConfig is licensed under the BSD license.

<http://www.ispconfig.org/>

This page is the document optimal configuration for Tiki. For general info about ISPConfig, please link to [ispconfig.org](http://www.ispconfig.org)

## 1.1. CREATE A CLIENT

- As admin, create a client.

You can use a predefined template (or create a new client-template first).

×

### Please Note

It seems ISPConfig allows to omit that and then there is sort-of default client0 but that can cause troubles later with databases e.g. so yes please, create one.

## 1.2. CREATE A WEB SPACE

- Create a website for that client. You need to have a client first, if you don't have one already.
  - Go to tab "sites", create a Website.

# 1.3. CREATE A DB

Create a database user first: "**Sites > Databases : Database Users > Add new User**"

The screenshot shows the ISPConfig interface. At the top, the 'Sites' menu item is highlighted with a red box and a circled '1'. In the left sidebar, the 'Database Users' link is highlighted with a red box and a circled '2'. The main content area is titled 'Database Users' and contains a 'Database Users' tab highlighted with a red box and a circled '3'. The 'Database user' field contains 'tiki01', with a red arrow pointing to it and the text 'db username suffix'. The 'Database password' field is filled with dots, and the 'Repeat Password' field also contains dots. A green message box below the password fields says 'The passwords do match.'. At the bottom, the 'Save' button is highlighted with a red box and a circled '4', and the 'Cancel' button is also visible.

Click to expand

Then, add a database: "**Sites > Databases : Databases > Add new Database**"

ISPConfig  
hosting control panel

Home System Client DNS Help Email Monitor Sites

Websites  
Website  
Subdomain for website  
Aliasdomain for website

**Databases**  
Databases  
Database Users

Web Access  
FTP-Accounts  
WebDAV-Users  
Protected Folders  
Protected Folder Users

Command Line  
Shell-User  
Cron Jobs

APS Installer  
Available packages  
Installed packages  
Update PackageList

Statistics  
Web Traffic  
Website quota (Harddisk)

**Databases**

Server: .com

Site: tiki01. .com

Type: MySQL

Database name: db\_name\_suffix

Database user: c1tik101

Read-only database user: optional

Database charset: UTF-8

Remote Access:

Remote Access IPs (separate by , and leave blank for any):

Active:

Save Cancel

*c1 prefix comes hardcoded from "client1"*

Click to expand

## 1.4. ADDING JAILROOT SSH USERS

- And making sure they have access to SVN

The client needs to have the option ssh-chroot option set to "Jailkit" mode.

Log in as a client (if permissions are granted for such task), or as ISPConfig admin, and go to **Sites > Command line > Shell user > Add new Shell-user**

## 1.5. INSTALLING TIKI

Get your tiki as usual. See, for instance:  
<http://dev.tiki.org/Get+code>

In this example, we will consider that you have fetched Tiki9 LTS into the folder

```
/home/xavi/tiki09svn
```

## 1.6. PHP TWEAKS

Remember that most php modes in ISP Config do not allow tweaking php options in a .htaccess, but they need to be set on a per-directory php.ini file. See the ISP config manual for more information.

In case you want to change some settings for all your sites, like rising the minimum upload size in php for all sizes, you can edit:

**command server side to edit your default php.ini settings for all sites**

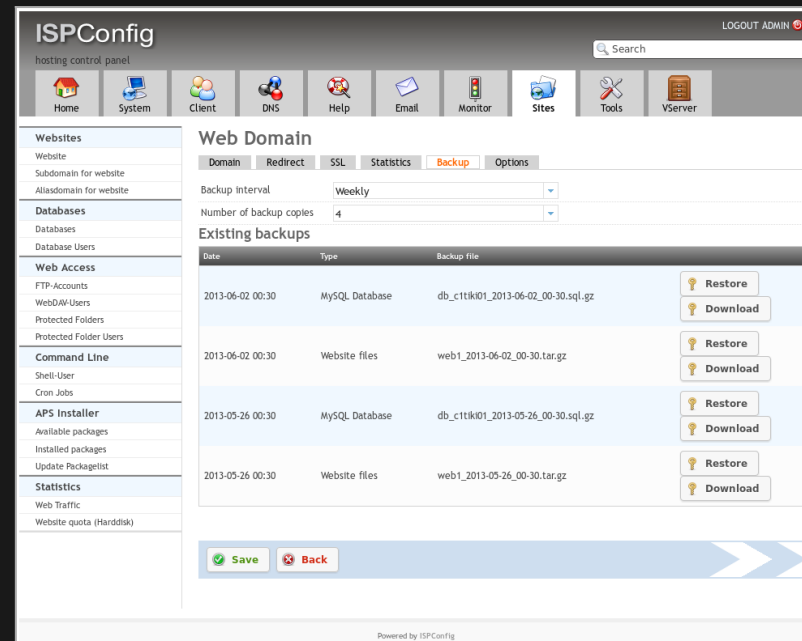
```
pico /etc/php5/cgi/php.ini
```

And change these two settings

```
upload_max_filesize 12M post_max_size 10M
```

# 1.7. BACKUPS

You can setup that per site, at: "**Sites > Websites : Website > Add new website** (or edit a previous one by clicking on its name in the list)", and go to the tab "backup":



Click to expand

## 1.7.1. PATHS INSIDE AND OUTSIDE THE CHROOTED ENVIRONMENT

New sites are associated with clients, and some ssh users can be created associated with that client and site.

ssh users have their chrooted environments in this absolute path in the server:

`/var/www/clients/clientN/webM/`

For instance, for the test case of the tiki01 site (<http://tiki01.example.com>) for user 'john', client 'smith' is #1 (N in the path above) ssh user smithjohn, and the website is #8 (M in the path above). Therefore, his website will be here:

```
/var/www/clients/client1/web8/
```

And when he logs in through ssh, he will be at the apparent path for him:

```
/home/smithjohn/
```

His website <http://tiki01.example.com> will be initially fed with the contents at the file (chrooted, apparently absolute path for him):

```
/web/index.html
```

Which in fact, will be the real paths at the server for his home directory and website are:

```
/var/www/clients/client1/web8/home/smithjohn/ /var/www/clients/client1/web8/web/index.html
```

# 1.8. MULTITIKI SETUP AND PERMISSIONS

Which permissions does the web folder have by default for a site?

For instance, web8 from client 1 will have these permissions at the web folder by default:

```
root@server:/var/www/clients/client1/web8# ls -l total 24 drwxr-xr-x 2 web8 client1 4096 abr 8 08:51 cgi-bin drwxr-xr-x 2 root
root 4096 abr 8 08:51 log drwx--x--- 2 web8 client1 4096 abr 8 08:51 private drwxr-xr-x 2 root root 4096 abr 8 08:51 ssl drwxrwx
- 2 web8 client1 4096 abr 8 08:51 tmp drwx--x--- 4 web8 client1 4096 abr 8 08:51 web drwx--x--- 2 web8 client1 4096 abr 8 08:5
webdav
```

If you check the attributes of those folders, they all seem ok:

```
root@server:/var/www/clients/client1/web8# lsattr ----- ./private ----- ./cgi-bin ----- ./log ----- ./ssl -----
----- ./web ----- ./tmp ----- ./webdav
```

However, if you check the attributes of the parent folder web8, you will see that it has the immutable bit enabled:

```
root@server:/var/www/clients/client1# lsattr (...) ----i----- ./web8 (...)
```

Therefore, you need to change that immutable bit attribute, in order to be able to remove the web subfolder and replace it with a symlink to the multitiki installation.

Do these steps:

## 1. remove the default initial web folder

- if you can't remove it as root, try removing the attribute "immutable" that you might have set in the parent folder ( `/var/www/clients/client1/web8/` ), with this command:

```
sudo chattr -i /var/www/clients/client1/web8/
```

- Then proceed to remove the web subfolder

```
sudo rm /var/www/clients/client1/web8/web/* -R sudo rmdir  
/var/www/clients/client1/web8/web/
```

## 2. create a symlink from the multitiki installation to that web folder.

- If the multitiki installation is at: `/var/www/multitiki/`, then run:

```
sudo ln -s /var/www/multitiki/ /var/www/clients/client1/web8/web/
```

## 3. Set the attributes back for the parent folder, if you changed them

- if you removed the attribute "immutable" for the parent folder, set it back, with this command:



```
sudo chattr +i /var/www/clients/client1/web8/
```

4. run `sh setup.sh` on the multitiki base folder, so that
  - you indicate as user `www-data`,
  - but as group: `client1` instead of the default `www-data`.
  - As sites, provide your domains for each tiki in the multitiki install: `site1.example.com`, `site2.example.com`, ...
5. Then you can proceed with the installation of your tiki at each domain, as usual.

## 1.9. WHERE TO STORE FILES OUTSIDE THE WEB ROOT

See below "**Open\_basedir restrictions for storing uploaded files on disk**"

Or add an `.htaccess` to avoid browsing

## 1.10. OPEN\_BASEDIR RESTRICTIONS FOR STORING UPLOADED FILES ON DISK

If you plan to setup file galleries, wiki attachments, tracker file uploads, etc., to use storage on disk instead of in database, then you are recommended to indicate a subfolder with the base path **private** of the site folder, with the absolute path:

---

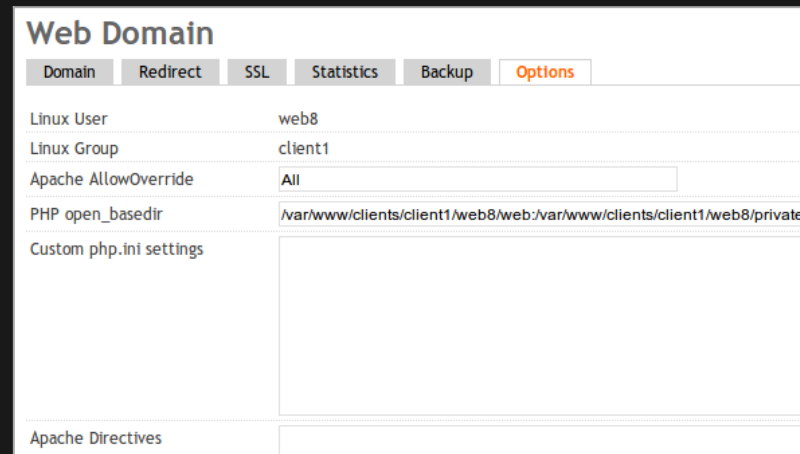
```
/var/www/clients/client1/web8/private/
```

Therefore, for file galleries, we could use the folder `"tiki_file_gals"`, with an ending slash, and therefore, the absolute path would be

```
/var/www/clients/client1/web8/private/tiki_file_gals/
```

This seems to be necessary to comply with the few `open_basedir` folders allowed by ISPConfig.

Alternatively, you could add new folders to the PHP `open_basedir` param in the ISPConfig Admin Panel: "**ISPConfig > Sites > (click on a web site name) > Options (tab) > PHP open\_basedir**"



Web Domain					
Domain	Redirect	SSL	Statistics	Backup	Options
Linux User	web8				
Linux Group	client1				
Apache AllowOverride	All				
PHP open_basedir	/var/www/clients/client1/web8/web:/var/www/clients/client1/web8/private:/'				
Custom php.ini settings	<div style="border: 1px solid #ccc; height: 100px;"></div>				
Apache Directives	<div style="border: 1px solid #ccc; height: 30px;"></div>				

Click to expand

Permissions of the folders and files under this folder can be set to 750 for user web8 and client1, following with the previous example.

```
drwxr-x--- 8 web8 client1 4096 des 12 17:06 tiki_file_gals/
```

But if you plan to setup several sites based on a multitiki install, and having two different sites (different domains, and site folders on disk, except that web folder will be a symlink to the real multitiki installation, etc), then you have to set permissions of the `tiki_file_gals` and subfolders to 770 for user web8 and client1, following with the previous example. The second site could be

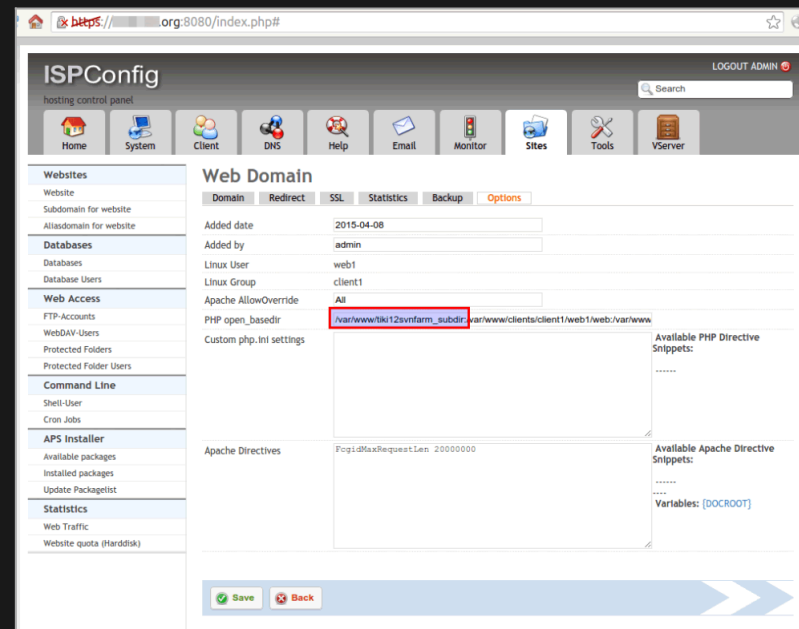
web9 and client1, and since both have the same client (unix group), you have to set that the group can write to those folders (and that's why the second 7 in 770, or rwx for group instead of r-x).

```
drwxrwx--- 8 web8 client1 4096 des 12 17:06 tiki_file_gals/
```

## 1.11. FIX 'NO INPUT FILE SPECIFIED.' FROM WEB SITES IN SUBFOLDERS OF A PREVIOUS SITE

If the website to be allowed in a subfolder of another one, is a symlink to another folder in the different path, you need to add that other path in the Open\_Basedir directive of PHP. You can do so through the ISP\_Config interface for that parent website (mysite.example.com, in this case) which hosts a web site in a subfolder (mysite.com/mysubsite/ for instance), because `mysubsite` is a symlink to `/var/www/tiki12svnfarm_subdir`.

In this case, we had to add: `/var/www/tiki12svnfarm_subdir:` to **PHP open\_basedir** parameter for the site `mysite.example.com`



Another approach, if the site is not in a multitiki installation, like the mrbs (for the `mysite.example.com/bookings/` ) is to move the external file tree inside the path of the parent site, so that you avoid using symlinks. this way, no need to tweak the open base dir directive.

## 1.12. ISPCONFIG MANUAL NOTES

### 1.12.1. DOCUMENTATION PAGES ABOUT JAILROOTED SSH USERS

[+]

## 1.13. ADDING TIKI TO CLIENT WEBSITES

For instance, to copy the svn installation of tiki12 under the user 'xavi' home folder over the website of a client (lets say: client1 (smith) web8 (john) (i.e. <http://tiki01.example.com> ), you can do that with:

---

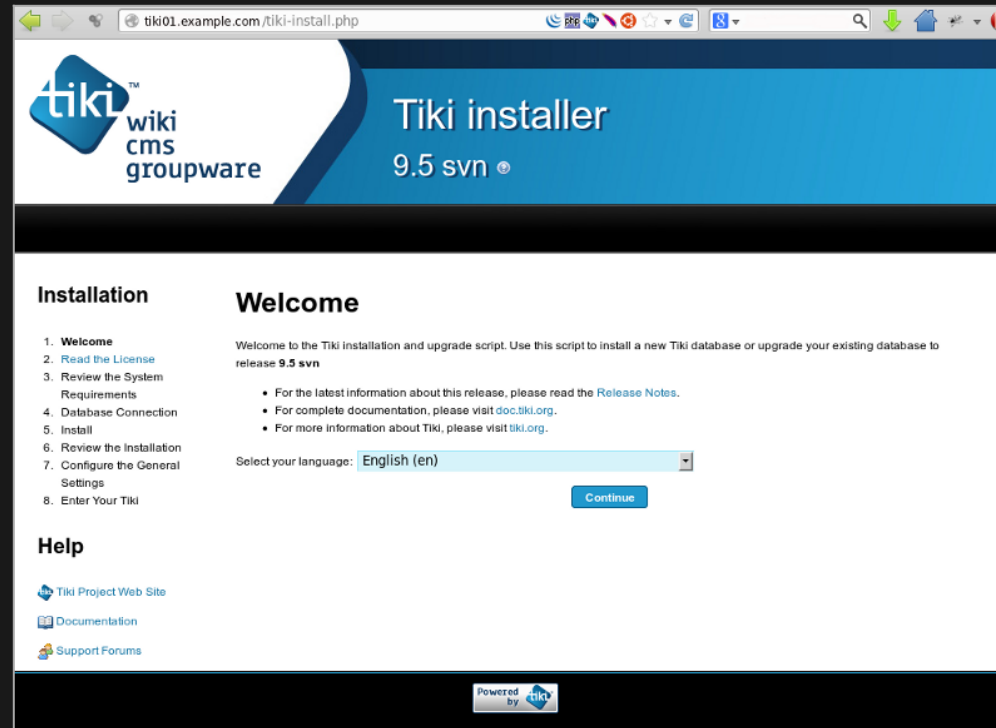
```
xavi@r:~# sudo su root@r:~# cd /home/xavi/tiki12svn root@r:~/tiki12svn# svn export --force .
/var/www/clients/client1/web8/web/ Export complete. root@r:~/tiki12svn# cd /var/www/clients/client1/web8/web/
root@r:/var/www/clients/client1/web8/web# mv .htaccess .htaccess_old root@r:/var/www/clients/client1/web8/web# cp _htaccess
.htaccess root@r:/var/www/clients/client1/web8/web# sh setup.sh User [www-data]: web8 > Group [www-data]: client1 > Multi[
Checking dirs : db ... ok. dump ... ok. img/wiki ... ok. img/wiki_up ... ok. img/trackers ... ok. modules/cache ... ok. temp ... ok.
temp/cache ... ok. temp/public ... ok. templates_c ... ok. templates ... ok. styles ... ok. maps ... ok. whelp ... ok. mods ... Creating
directory ok. files ... ok. tiki_tests/tests ... ok. temp/unified-index ... Creating directory ok. Fix global perms ... Change user to web
and group to client1... done. Fix normal dirs ... done. Fix special dirs ... done.
```

The force option is needed since the destination folder already exists.

And the svn export is preferred (if no svn is needed) because of the space savings reducing it down to aprox. 40% of the initial size

on disk (As a simple reference, 453 Mb for the svn-enabled version of tiki09svn, 181 Mb for the non-svn-enabled version).

At this point, the client user can proceed with the installation of Tiki as usual through the web interface that he will see when he goes to his website url; in this example: <http://tiki01.example.com>



Click to expand

Later on in step4 of the installation, he will need to provide the db credentials that were create for him in earlier step by the isprconfig admin (see above; in this example, db name: c1tiki01, db user: c1tiki01, and the db password you used at that step).

## 1.14. GRANT SVN TO CHROOTED USERS

Run this:

```
root@server:/# jk_cp /var/www/clients/client1/web8 /usr/bin/svn
```

Some users have reported to get error messages reporting that some folders are not owned by root:root, like this one below,

```
ERROR: /var/www/clients/client1/web8/etc is not owned by root:root!
```

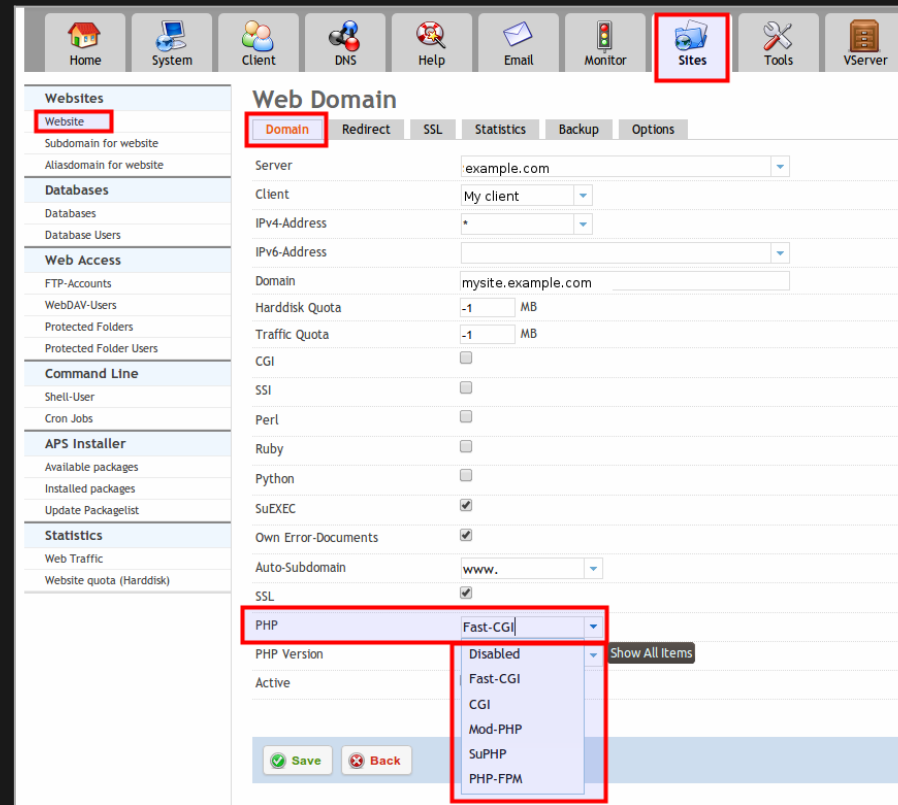
In such case, the problem was solved by changing ownership to root for these folders like etc, and running the first command `jk_cp` again:

```
root@server:/var/www/clients/client1/web8# chown -R root:root etc usr var bin dev root@server:/var/www/clients/client1/web8#
jk_cp /var/www/clients/client1/web8 /usr/bin/svn Copying /usr/bin/svn to /var/www/clients/client1/web8/usr/bin/svn Creating
symlink /var/www/clients/client1/web8/usr/lib/x86_64-linux-gnu/libsvn_client-1.so.1 to libsvn_client-1.so.1.0.0 Copying
/usr/lib/x86_64-linux-gnu/libsvn_client-1.so.1.0.0 to /var/www/clients/client1/web8/usr/lib/x86_64-linux-gnu/libsvn_client-1.so.1.0.0
Creating symlink /var/www/clients/client1/web8/usr/lib/x86_64-linux-gnu/libsvn_wc-1.so.1 to libsvn_wc-1.so.1.0.0 Copying
/usr/lib/x86_64-linux-gnu/libsvn_wc-1.so.1.0.0 to /var/www/clients/client1/web8/usr/lib/x86_64-linux-gnu/libsvn_wc-1.so.1.0.0
Creating symlink /var/www/clients/client1/web8/usr/lib/x86_64-linux-gnu/libsvn_ra-1.so.1 to libsvn_ra-1.so.1.0.0 Copying
/usr/lib/x86_64-linux-gnu/libsvn_ra-1.so.1.0.0 to /var/www/clients/client1/web8/usr/lib/x86_64-linux-gnu/libsvn_ra-1.so.1.0.0
Creating symlink /var/www/clients/client1/web8/usr/lib/x86_64-linux-gnu/libsvn_delta-1.so.1 to libsvn_delta-1.so.1.0.0 Copying
/usr/lib/x86_64-linux-gnu/libsvn_delta-1.so.1.0.0 to /var/www/clients/client1/web8/usr/lib/x86_64-linux-gnu/libsvn_delta-1.so.1.0.0
Creating symlink /var/www/clients/client1/web8/usr/lib/x86_64-linux-gnu/libsvn_diff-1.so.1 to libsvn_diff-1.so.1.0.0 Copying
/usr/lib/x86_64-linux-gnu/libsvn_diff-1.so.1.0.0 to /var/www/clients/client1/web8/usr/lib/x86_64-linux-gnu/libsvn_diff-1.so.1.0.0
Creating symlink /var/www/clients/client1/web8/usr/lib/x86_64-linux-gnu/libsvn_subr-1.so.1 to libsvn_subr-1.so.1.0.0 Copying
/usr/lib/x86_64-linux-gnu/libsvn_subr-1.so.1.0.0 to /var/www/clients/client1/web8/usr/lib/x86_64-linux-gnu/libsvn_subr-1.so.1.0.0
Creating symlink /var/www/clients/client1/web8/usr/lib/libapr-1.so.0 to libapr-1.so.0.4.6 Copying /usr/lib/libapr-1.so.0.4.6 to
```

/var/www/clients/client1/web8/usr/lib/libapr-1.so.0.4.6 Creating symlink /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_ra\_local-1.so.1 to libsvn\_ra\_local-1.so.1.0.0 Copying /usr/lib/x86\_64-linux-gnu/libsvn\_ra\_local-1.so.1.0.0 to /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_ra\_local-1.so.1.0.0 Creating symlink /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_ra\_svn-1.so.1 to libsvn\_ra\_svn-1.so.1.0.0 Copying /usr/lib/x86\_64-linux-gnu/libsvn\_ra\_svn-1.so.1.0.0 to /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_ra\_svn-1.so.1.0.0 Creating symlink /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_ra\_neon-1.so.1 to libsvn\_ra\_neon-1.so.1.0.0 Copying /usr/lib/x86\_64-linux-gnu/libsvn\_ra\_neon-1.so.1.0.0 to /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_ra\_neon-1.so.1.0.0 Creating symlink /var/www/clients/client1/web8/usr/lib/libaprutil-1.so.0 to libaprutil-1.so.0.3.12 Copying /usr/lib/libaprutil-1.so.0.3.12 to /var/www/clients/client1/web8/usr/lib/libaprutil-1.so.0.3.12 Creating symlink /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsqlite3.so.0 to libsqlite3.so.0.8.6 Copying /usr/lib/x86\_64-linux-gnu/libsqlite3.so.0.8.6 to /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsqlite3.so.0.8.6 Creating symlink /var/www/clients/client1/web8/lib/x86\_64-linux-gnu/libuuid.so.1 to libuuid.so.1.3.0 Copying /lib/x86\_64-linux-gnu/libuuid.so.1.3.0 to /var/www/clients/client1/web8/lib/x86\_64-linux-gnu/libuuid.so.1.3.0 Creating symlink /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_repos-1.so.1 to libsvn\_repos-1.so.1.0.0 Copying /usr/lib/x86\_64-linux-gnu/libsvn\_repos-1.so.1.0.0 to /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_repos-1.so.1.0.0 Creating symlink /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_fs-1.so.1 to libsvn\_fs-1.so.1.0.0 Copying /usr/lib/x86\_64-linux-gnu/libsvn\_fs-1.so.1.0.0 to /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_fs-1.so.1.0.0 Creating symlink /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsasldb2.so.2 to libsasldb2.so.2.0.25 Copying /usr/lib/x86\_64-linux-gnu/libsasldb2.so.2.0.25 to /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsasldb2.so.2.0.25 Creating symlink /var/www/clients/client1/web8/usr/lib/libneon-gnutls.so.27 to libneon-gnutls.so.27.2.6 Copying /usr/lib/libneon-gnutls.so.27.2.6 to /var/www/clients/client1/web8/usr/lib/libneon-gnutls.so.27.2.6 Creating symlink /var/www/clients/client1/web8/lib/x86\_64-linux-gnu/libcrypt.so.1 to libcrypt-2.15.so Copying /lib/x86\_64-linux-gnu/libcrypt-2.15.so to /var/www/clients/client1/web8/lib/x86\_64-linux-gnu/libcrypt-2.15.so Creating symlink /var/www/clients/client1/web8/lib/x86\_64-linux-gnu/libexpat.so.1 to libexpat.so.1.5.2 Copying /lib/x86\_64-linux-gnu/libexpat.so.1.5.2 to /var/www/clients/client1/web8/lib/x86\_64-linux-gnu/libexpat.so.1.5.2 Creating symlink /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_fs\_fs-1.so.1 to libsvn\_fs\_fs-1.so.1.0.0 Copying /usr/lib/x86\_64-linux-gnu/libsvn\_fs\_fs-1.so.1.0.0 to /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_fs\_fs-1.so.1.0.0 Creating symlink /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_fs\_base-1.so.1 to libsvn\_fs\_base-1.so.1.0.0 Copying /usr/lib/x86\_64-linux-gnu/libsvn\_fs\_base-1.so.1.0.0 to /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_fs\_base-1.so.1.0.0 Creating symlink /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_fs\_util-1.so.1 to libsvn\_fs\_util-1.so.1.0.0 Copying /usr/lib/x86\_64-linux-gnu/libsvn\_fs\_util-1.so.1.0.0 to /var/www/clients/client1/web8/usr/lib/x86\_64-linux-gnu/libsvn\_fs\_util-1.so.1.0.0 Copying /usr/lib/x86\_64-linux-gnu/libdb-4.8.so to /var/www/clients/client1/web8/usr/lib/x86\_64-

## 1.15. PHP MODES

PHP-FCGI is the default PHP mode used in ISPConfig3 admin panels. But you can change it to other PHP modes if desired.



Click to expand

The other PHP modes are:

- FastCGI
- CGI



- Mod-PHP
- SuPHP
- PHP-FPM

We'll show how to fix some usual errors with some of them.

## 1.15.1. USING PHP MODE PHP-FCGI

### 1.15.1.1. ERROR 500 IN PHP MODE PHP-FCGI: ALLOW UPLOADING BIGGER FILES THAN 1MB

If you hit error 500 when attempting to upload files bigger than 1 Mb, and error.log shows something like:

---

```
mod_fcgid: HTTP request length 131665 (so far) exceeds MaxRequestLen
```

then you need to increase this directive in your vhost, to something like 2MB (131Kb by default)

---

```
FcgidMaxRequestLen 2000000
```

More in:

[http://www.howtoforge.com/apache2-mod\\_fcgid-http-request-length-exceeds-maxrequestlen](http://www.howtoforge.com/apache2-mod_fcgid-http-request-length-exceeds-maxrequestlen)

## 1.15.2. USING PHP MODE PHP-FPM

You might want to use another PHP mode for your website. For instance, PHP-FPM (FastCGI Process Manager), which is an alternative PHP FastCGI implementation with some additional features useful for sites of any size, especially busier sites..

According to the ISPConfig3 pdf manual, you need to install these packages:

### Commands in a terminal

```
sudo apt-get install php5-fpm sudo /etc/init.d/php5-fpm restart sudo apt-get install fcgiwrap
```

#### 1.15.2.1. ERROR 500 IN PHP MODE PHP-FPM: FILE IS NOT IN DOCUMENT ROOT OF VHOST

You may choose another PHP mode for your site. For instance, suPHP, or PHP-FPM. In such case, some features like replacing a file in a file gallery might produce this type of error 500:

```
root@example:/# tail /var/log/ispconfig/httpd/mysite.example.com/error.log (...) SoftException in Application.cpp:221: File "/var/www/c1tiki12farm/tiki-list_file_gallery.php" is not in document root of Vhost "/var/www/clients/client1/web6/web", referer: http://example.com/tiki-list_file_gallery.php
```

As indicated [here](#), the solution to this is to pop open the suphp config file (/etc/suphp/suphp.conf) and tell it to stop checking that scripts are under the document root like this:

```
File edited: /etc/suphp/suphp.conf
```

```
check_vhost_docroot=false
```

And restart Apache.

```
sudo service apache2 restart
```

You will get then this other error when visiting any url of the tiki site:

```
root@example:/# tail /var/log/ispconfig/httpd/mysite.example.com/error.log (...) SoftException in Application.cpp:350: UID of script "/var/www/clients/client1/web6/web/tiki-list_file_gallery.php" is smaller than min_uid Premature end of script headers: tiki-list_file_gallery.php
```

As indicated [here](#), Suphp, by default, won't allow any scripts to run with a user or group ID under 100. Since Tiki has all its files installed owned by the user www-data (UID 33) when it's installed, this poses quite a problem. One solution is to set the min\_uid and min\_gid values in the suphp config file to 33 which allows the scripts to run as www-data.

```
File edited: /etc/suphp/suphp.conf
```

```
; Minimum UID ;min_uid=100 min_uid=33 ; Minimum GID ;min_gid=100 min_gid=33
```

Restart Apache, and you'll be able to replace files in file galleries again.

## ERRORS

### 1.15.2.2. UNSOLVED ERROR: HANDLER FOR (NULL) RETURNED INVALID RESULT CODE 70007/14

These errors are produced on some pages in one ISPCongig3-powered Tiki 12.x site, on intervals, and it doesn't seem an easily reproducible scenario to reproduce the errors:

---

Handler for (null) returned invalid result code 70007

or

---

Handler for (null) returned invalid result code 70014

Looking at the apache log, they seem to be triggered by tiki-register.php (when we had real users attempting to register to a course site), and edits to simple pages:

---

tiki-editpage.php?page=Session%2018&page\_ref\_id=860 tiki-editpage.php?page=Students

Those wiki pages contain some plugins, but they have been edited dozens of times, in a server without ISPConfig3, and in Tiki versions  $\leq 11.x$ , and they never produced an error 500.

PHP mode for that site is PHP-FPM.

Googling for these type of errors, nothing conclusive was found; just this type of information (e.g., from [here](#)):

“

*7 is "time up" and 14 is "EOF", so the root cause of this is probably a read timeout during the upload.*

*The 70000 prefix means they are APR status codes.*

*They are logged poorly here because they the core of apache is expecting a different type of error code then what is returned — these are APR or "filter" return codes but the core of httpd is expecting a HTTP status code or 0 for OK.*

The most odd, is that, just waiting 5 minutes or so, and doing an F5 (browser refresh) of that edition of a wiki page, produced a normal output: page edited saved and shown, with no issues at all.

So something related to server load? Internal Timeout? ...

Hints welcome at [xavi \(a\) tiki.org](mailto:xavi@tiki.org)

Alias names for this page:

[ISPConfig3](#)