

# Troubleshooting FAQ

If you're having a problem with your Drupal site, you're almost certainly not alone. Your questions may already have been asked and answered many times. Save yourself some time and start with this list of Frequently Asked Questions (FAQs) before you post an issue in the queue.

- If you don't find your answer here, you can [search the entire Drupal site](#).
- If you still can't find an answer you can ask for help in the [Support Forums](#), or [learn to use IRC](#) and then ask your question in the [#drupal-support](#) IRC channel.
- For other FAQs, see [Drupal FAQs](#).

## Drupal troubleshooting guide

When you encounter issues with your Drupal site, there are a few ways and places you can look for clues to your problem.

1. **Stop and think**
  - Identify what changes you've made recently, including modules you've recently installed or updated.
  - Double check the version of the module(s) you installed matches your version of Drupal.
2. **[Read the error messages](#)**. Drupal-specific error messages may display on the page, or you can find them in Administer/Reports/Recent log entries.

You may want to also check non-Drupal logs, such as Apache, PHP, etc.

Errors may include:

- .htaccess config problems - "Internal server error" - Error 500 [directive not allowed here](#)
  - Memory problems [Fatal error: Allowed memory size of 8388608 bytes exhausted](#)
  - Misbehaving javascript - modules/troublesome/utility.js 404 file not found.
3. **[Validate your page](#)**  
Incomplete markup tags and structural problems may cause your layout problem - alignment, font size, overflowing blocks, etc. Though you may not be able to solve all validation issues, you should know what errors exist, and how important they are. [The web developer toolbar](#) has tools to help you, including the "Validate local HTML" option, which lets you validate non-public sites.
  4. **Check your CSS**  
Some rules will override others, so [Learn about the first 'C' in CSS - the Cascade](#).
    - Check container and child elements - the margin for one item may be the padding for another (particularly in list items).
    - Don't hack core - If you find unwanted CSS coming from a core Drupal file, override it in a new style.css file using the same rule pattern.

You can inspect your CSS with these tools:

- [FireBug](#) - for Firefox
- [Web Developer Toolbar](#) - for Firefox, Flock, and Seamonkey
- [DOM inspector](#) - for Internet Explorer
  - Open DOM inspector.
  - Use the 'Select element by click' pointer and click on the problem element.
  - Navigate the DOM tree to ensure it's the right element.
  - Look at the 'CSS Style Rules' pane.

- Click through the 'Rules' and find out all the places where the element gets its formatting from.

## 5. Read the README

The README.txt file can contain the following:

- General information and advice from the contributor(s)
- Dependencies (such as server PHP extensions)
- Requirements
- Installation instructions
- Cavaets (such as module conflicts and patches)

## 6. Search the proper way.

If the error you see (in the log or the screen) doesn't make sense to you, try a search on it. Use Google, and **quote** the error as best you can. [Google: "Argument #1 is not an array" drupal](#). - A google on a phrase, plus the keyword 'drupal' is probably better than a drupal.org search. Take care to leave out pathnames (but filenames are probably useful). Try different permutations of quotes around what seem to be discreet parts of the message. Try searching both with and without arguments. Strings input by the user won't help much when searching for the problem elsewhere.

## 7. Identify the source of the problem.

You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '(n.nid), n.title FROM category c INNER JOIN category\_node r ON c.cid = r.cid INN' at line 1 ... in /var/www/html/doadance/drupal/includes/database.mysql.inc on line 120.

Errors like the above are **not** caused by drupal database.inc. They are simply found by it because a module has sent some bad instructions to the internals. We need to find where the call came from, unfortunately you won't get a call stack by default. Try to identify the *table* being addressed in this query to see what module may be doing it. In this case it *looks* like *category*, but it may in fact be any other module that's trying to directly access data about categories.

## 8. Dump some diagnostics.

Debugging. If you are really trying to find the causes, it may help to display some of the code.

- [Try the devel.module](#). It's got lots of useful functionality.
- For one-off quick hacks, try some lightweight code, for instance when you just want to know why you are seeing the following:  
warning: in\_array() [function.in-array]: Wrong datatype for second argument in /home/httpd/global/drupal/modules/node.module on line 1303.  
This is an excerpt from [here](#).
- **Go look at that file.** Modify it a bit. It's safe to do so if you remember to undo afterwards.
- On line 1303, node.module we find `in_array('status', $node_options)`.
- on *the line before* try inserting the code: `print("Node options are : ".print_r($node_options,1)."\n");` and see if that gives any clue.
- A useful data dump when debugging, even within node pages themselves is sometimes.  
`print("Node is : ".print_r($node,1)."\n");`, Devel.module provides much better tools, like `dsm()` and even provides a data dump of nodes as an automatic tab.

## 9. Ask the right questions.

When [posting questions on the Drupal forum](#), it's usually better to post more info than less. The more work you can demonstrate has gone into solving it already, the more likely folks are to help you get the rest of the way. [Do ask questions "The smart way"](#). Do follow up in the same thread if you resolve it - even if no-one else posted. Do consider submitting your solution as a comment under [Troubleshooting FAQ](#) or the elsewhere in the handbook. In Drupal, it's usually important to know if you are on a shared hosting provider - mention this!

## 10. Identify the module that's giving you problems.

All Drupal 'pages' are dynamically created to one extent or another. The URL where dynamically created pages are published is usually defined in a module's `hook_menu()` implementation. When installing or investigating new modules, it's usually interesting to open up the code and look at the `***_menu()` function to see what it does. Conversely, if you want to find *which* module is serving which page, you'll have the search there also. To answer a question like "[how is my sites rss.xml created?](#)", a troubleshooting process may be:

- Do a find-in-files for "rss.xml"
- This returns, sure enough, a result in `node.module` : `node_menu()`
- This tells us that the function in question - that serves that page - is `node_feed()`. This is where you want to look for further details.
- All dynamic pages can be debugged by starting like this. Look at the URL, identify the module that serves that URL, trace the callback through the code.

## Report a problem

If you've found a problem with Drupal, here's what to do:

1. Try to solve the problem yourself.
  - Look for hints in the [Troubleshooting FAQ](#). Chances are that your question has been asked and answered many times already.
  - Read the appropriate sections of [the Drupal Handbooks](#).
  - [Search the issue queues](#) for earlier reports of your problem. [Learn about the issue queues](#).
  - [Search the Drupal site](#).
2. Report problems with the Drupal software, the documentation, or the drupal.org site to the community using the issue queues.
  - [Learn about Drupal's issue queues](#). Learn [how to report bugs effectively](#).
  - Report an [issue with the Drupal software](#).
  - Report an [issue with the documentation](#).
  - Report an [issue with the Drupal.org website](#) to the webmasters. Note that this is for problems with Drupal.org and not your own website.

Note that Drupal is built by a community of volunteers, that there are more things to be done than there are people to do them, and that even "simple" changes take time. See: [how to enact change within the Drupal community](#).

3. Ask the community for assistance. Please do not simply post a Support Forum request without at least trying to find an answer using the FAQ or the docs (see above). The online docs will probably offer you a faster resolution for your problem.
  - You should begin by [getting an account on Drupal.org](#) if you haven't already.
  - For real-time answers to brief questions you can ask the [#drupal-support](#) channel on [IRC](#).
  - Ask your question in the [Drupal Support Forums](#). To use the forums effectively, read these [tips for posting to the Drupal forums](#).
  - Other ways to [talk with the Drupal community](#).

## Webhosting issues

A lot of new Drupal users run into issues getting Drupal to work properly on their webhost. These problems are not Drupal problems as such, they are usually the way the webserver is configured. Complex and functional web applications like Drupal will naturally require a lot more from a web server than just hosting a static HTML site would.

Unfortunately there is usually a knowledge gap between the Drupal installation documents and the exact environment a new user is confronted with when attempting to install and configure Drupal. For more experienced users that have a little Unix and Apache (a typical webhost setup) knowledge under their belts, bridging the gap between their webhosting environment and the general purpose Drupal installation guide is no problem. Things are more confusing for less experienced users though. The following info is intended to help newbies come to grips with the documentation and their webhosts environment.

There are a wide range of web site configuration tools (eg CPanel, Plesk, webmin etc) and near infinite number of ways a webhost could configure a webserver. What this means is that a lot of the general Drupal docs you read will have to use the lowest common denominator (ie Unix shell commands) to describe configuration steps.

Less experienced users will not fully understand what these commands do or why they should do them, and definitely won't know when to deviate from them. To make matters worse most webhosts don't even offer shell access, so the instructions need to be *'translated'* to whatever control panel they offer.

The following topics will hopefully provide just enough insight into Apache and Unix that new users will be able to better understand the installation documentation and 'tune' their sites for running Drupal without too much pain. As a new user you might also gain an understanding of why most of the docs are written the way they are.

Note: There are webhosts that offer Drupal specific hosting for those without the time or inclination to do it themselves. See <http://drupal.org/services> and <http://drupal.org/forum/34> for more info.

## A mod\_rewrite bug causing occasional corruption of the query string

It seems that using mod\_rewrite to set or modify the query string can corrupt it. One case where it certainly fails is if the URL has %2B (a urlencoded "+" character) in it.

As a demonstration, try searching for the string "alice + bob". You can see on <http://drupal.org/search/node/alice+%2B+bob> that the rewrite rules change the %2B to a space, while on <http://drupal.org/?q=search/node/alice+%2B+bob> it work correctly without clean URLs.

The reason I am posting it here as a warning and not as a bug report is that it's not a Drupal issue, but a mod\_rewrite issue. (It would be worth a bug report for mod\_rewrite, but they have such bug reporting guidelines that require a few hours' work to follow...)

## Brief intro to Unix file permissions

Unix is a multiuser operating system. That means that it was designed for multiple users to be logged in at once and each running their own programs without getting in the way of each other.

Every program that runs on a Unix machine runs as a specific user account. This includes the web server itself, any [command line shell](#) you are running, and whatever software you use to access the servers file system and admin interfaces (eg FTP software, CPanel interface etc). When you log in with a username and password through an FTP client or a control panel you are now operating on the server using that user account.

Note: this doesn't include any Drupal user accounts you have set up. They only exist in your Drupal database, not as any operating system user accounts.

All files on a Unix server have an owner and a group assigned to them. Whenever a file is created on the server it is automatically owned by the user account running the program that created it. Each user account also has a primary group associated with it, and this group also gets assigned to the files group.

Each file and directory also has a set of permission 'bits' assigned to it as well. These permission bits

determine what access various users get to a file. The owner of a file is allowed to change these permissions, but all other users can't change them (with the exception of the root user).

The file permission bits are arranged into three sets: 'user owner', 'group owner', and 'other'. These three sets can also be referred to as 'user', 'group' and 'world' respectively. 'world' or 'other' refers to the permissions that apply for any user that isn't the owner and isn't in the files group. Each of these can have its own combination of three basic permissions.

The three basic permissions are 'read', 'write', and 'execute' and are abbreviated as 'rwx'. When you see dashes replacing a letter that means that the permission is absent eg 'r--' means that only read access is present.

When all three sets of permission bits are combined you get a setting like 'rwxr-xr-x' which represents 'rwx' for the owner, 'r-x' for the group, and 'r-x' for everyone else.

You will also see permissions represented as a numerical shorthand eg 755 or 644 etc. In this case the value of 'r' = 4, 'w' = 2, and 'x' = 1, and the digits are determined by adding up these numbers for each set.

examples:

- 755 is shorthand for 'rwxr-xr-x'. Translation: full access for the owner, everyone else has read and execute access
- 664 is shorthand for 'rw-rw-r--'. Translation: the owner and the group get read and write access, all other users get read access.

For files these permissions settings are quite straight forward. 'read' allows accessing the contents of a file, 'write' allows the file to be changed or deleted, and 'execute' allows the file to be run as a program from the command line shell. Note that the execute bit isn't really required for PHP files as they don't generally get run from the shell.

Permissions on directories are a little different from those on files. 'read' allows the contents of a directory to be listed, 'write' means that you can add or delete files in the directory, and 'execute' allows direct access to files in the directory (if you already know their names). On most directories read and execute bits tend to go together ie typically directories will either have both bits set or neither set.

There is a friendly tutorial here if you need more information:

<http://www.perlfect.com/articles/chmod.shtml>

If you didn't see the link above about [command line shell](#), here it is again.

See also permissions: <http://drupal.org/node/34025> in this handbook.

## Configure settings.php and .htaccess to redirect subfolders properly

When you're on a shared host like Lunarpages, you'll sometimes need to install drupal in subfolder. Often, you'll want to make it look like the subfolder is the base URL. To do that, edit .htaccess in your domain's public HTML root to redirect requests for pages, as instructed here:

[http://support.lunarpages.com/knowledge\\_bases/article/549](http://support.lunarpages.com/knowledge_bases/article/549)

You'll need to make sure that the redirect-to-subfolder command in your domain's .htaccess file isn't redundant with the settings.php (in sites/default/) \$base\_url line. If they're redundant, you'll get an

infinite loop during certain tasks.

In sites/default/settings.php, you will have:

```
$base_url = 'http://www.MY-SITE.com'; // NO trailing slash!
```

DON'T list the base URL as

```
$base_url = 'http://www.MY-SITE.com/subfolder';
```

To prevent an infinite redirect loop with sophisticated external connection applications and other errors, let the .htaccess file in the public HTML root (one folder above the subfolder) do the redirecting. Don't include your subfolder in your settings.php Base URL. This may seem obvious, but it is easy to miss because the redundancy doesn't cause problems until you use sophisticated programs like the fb modules.

## Enabling and disabling phpinfo() for security reasons

Some server administrators may choose to disable the PHP function `phpinfo()` for security reasons, because it displays information which can be used to compromise the server that your site is running on. In cases where `phpinfo()` is disabled, debugging problems in Drupal (and PHP in general) is much more difficult but the server is also more secure.

If `phpinfo()` is **disabled** and you want to enable it, try the following:

- If you have access to the server's `php.ini` file and the line that includes the `disable_functions` directive says `disable_functions = phpinfo` then change it to `disable_functions =`
- If you don't have access to your server's `php.ini` file, you may be able to [create your own php.ini file](#) and change the `disable_functions` directive that way. If that doesn't work, please contact your server administrator.

If `phpinfo()` is **enabled** and you want to disable it, try the following:

- If you have access to the server's `php.ini` file, change the line that includes the `disable_functions` directive so that it says `disable_functions = phpinfo`
- If you don't have access to your server's `php.ini` file, you may be able to [create your own php.ini file](#) and change the `disable_functions` directive that way. If that doesn't work, you may be able to disable `phpinfo()` by configuring Drupal's `.htaccess` or `settings.php` files.
  - In `.htaccess`, add a line that says `php_value disable_functions phpinfo`
  - In `settings.php`, add a line that says `ini_set('disable_functions', 'phpinfo');`

## Host-specific error messages

Some web hosts generate inexplicable errors when you install, upgrade, or even use certain contrib modules in Drupal. The host may not know they're the one causing the problem, mostly because you assume you're the one who doesn't understand the inner workings of Drupal. Here are a few host-specific error messages we've encountered, and how you can ask your host to solve them.

Hosting Company	Error Message	Solution
	While editing a View, you receive the following error	Ask the host to make a

Webmasters.com	message: <b>Forbidden. You don't have permission to access /admin/build/views/edit/1 on this server.</b>	mod_security entry for your <a href="http://www.example.com">www.example.com</a> account.
Webmasters.com	While editing a Content Template (Contemplate), you receive the following error message: <b>Forbidden. You don't have permission to access /admin/content/templates/yourtemplatename on this server.</b>	Ask the host to make a mod_security entry for your <a href="http://www.example.com">www.example.com</a> account.
not specified	running the install.php script on a shared host, You got this message: <b>The following error must be resolved before you can continue the installation process: The Drupal installer requires write permissions to ./sites/default/settings.php during the installation process..</b>	change the permissions for folders to 755 and files to 644. Just right click on the settings.php file which is in "your site root /sites/default/settings.php" , if you are using an FTP like go to properties and change permission to 777. Drupal will remind you to change them back later on

If you have something to add to this list, please post a comment, and one of the document maintainers will add it for you, then erase your comment.

## Increase PHP memory limit

A PHP memory limit of 16MB is the minimum [requirement for Drupal 6](#) and 32MB is recommended. Some sites may need more than 32MB if they are using certain contributed modules such as CCK and Views. Memory limits of 64MB and higher are not unusual. There are several techniques to increase the PHP memory limit and you only need to use one of them. The right one for you depends on your system configuration.

### You may try to install a module

List of modules which allow you increase your PHP memory limit without editing any files:

[Drupal Tweaks](#)

### php.ini

This is the recommended approach if you have access to php.ini. This may not be possible in many shared hosting environments. Note that this change will affect all websites and PHP scripts on the server.

1. Locate the php.ini file used by your web server. You can use the [phpinfo\(\)](#) PHP function to find it. During installation Drupal checks the PHP Memory Limit and if it is less than 16M an error message also provides the path to the php.ini file.
2. Edit the memory\_limit parameter in the php.ini file (usually in a section called Resource Limits)

```
memory_limit = 32M ; Maximum amount of memory a script may consume (32MB)
```

 If there is no section already for this, place the above line at the end of the file.
3. Restart Apache.

Note: If you are using XAMPP/WAMP, there may be two PHP.ini files (one under the PHP directory and the other under Apache/bin). To change your memory limit, edit the file in the XAMPP/Apache/bin directory.

*The next two solutions are more restricted in scope and, in some cases, may be more appropriate choices than affecting all sites.*

## .htaccess

Edit the .htaccess file in the Drupal root directory. Look for the section:

```
# Override PHP settings. More in sites/default/settings.php  
# but the following cannot be changed at runtime.
```

and immediately after this add the following line:

```
php_value memory_limit 32M
```

## settings.php

If Drupal is already installed, you can edit sites/default/settings.php. This method will affect only the site using this file.

Locate the PHP settings section and add the following line at the end of that section:

```
ini_set('memory_limit', '32M');
```

## Shared Hosting

In some shared hosting environments, access to the PHP memory limit setting is restricted. If you cannot make the change yourself, please ask your hosting provider to adjust it for you, or look for a new host that allows more flexibility.

## Check your change has taken effect

In all cases, it pays to ensure that your change is actually working. Use [phpinfo](#) to verify that your memory actually is what you want it to be. If your change doesn't seem to be working, *double-check the location of php.ini* displayed in the phpinfo page. Some systems have multiple copies of that file in different places. Only one is used and the others are red herrings.

## Finding php.ini on your Local Server

- For MAMP see <http://drupal.org/node/66187> particularly the "Optional: Adjust PHP's memory limit for scripts" section (note the difference for MAMP Pro).

# Modifying Linux, Unix, and Mac file permissions

## Modifying Linux, Unix, and Mac file permissions

To solve write permission errors, you need to modify the permissions on the file or directory so that it can be writable by the web server process that is running Drupal. More information about why you have to do this can be found in [Brief intro to Unix file permissions](#).

To change file or directory permissions, log in to the system, and follow the directions below:

## From command line:

To GRANT write access:

1. Browse to the parent directory using `cd [path]`
2. Change permissions using `chmod a+w [file-or-folder]`

To REMOVE write access:

1. Browse to the parent directory using `cd [path]`
2. Change permissions using `chmod a-w [file-or-folder]`

Example:

To make Drupal's sites/default folder writeable...

```
cd /home/exampleuser/www.example.com/sites
chmod a+w default
```

For security, you should revoke write permissions on the files or folders that Drupal no longer needs permissions for when the installation is complete.

```
cd /home/exampleuser/www.example.com/sites
chmod a-w default
```

## Through an FTP client:

After connecting to the server and browsing to the parent directory, you usually need to right-click the file or directory and use some sort of "change permissions" option, depending on the exact application you're using. Some FTP software will only let you use a numeric shorthand for file permissions and in that case you should, when adding write permissions, change the permissions to '777' for folders, and '666' for files.

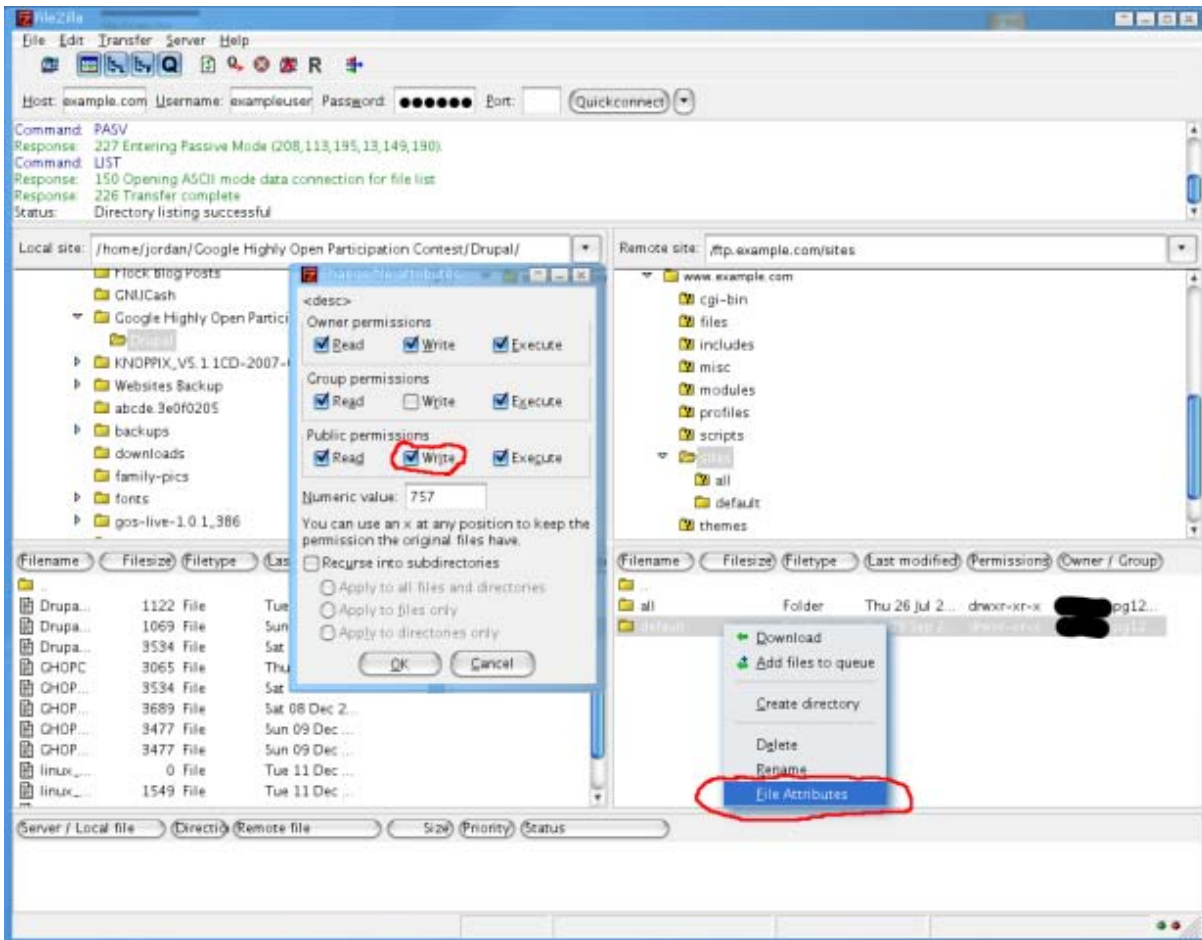
Using the [FileZilla client](#):

When adding write permission:

1. After logging in to the server, navigate to the the directory or file that needs its permissions changed.
2. Right-click on the directory/file.
3. In the resulting menu, click on "File Attributes".
4. Check all of the "Write" check boxes.

To remove write permissions:

1. After logging in to the server, navigate to the the directory or file that needs its permissions changed.
2. Right-click on the directory/file.
3. In the resulting menu, click on "File Attributes".
4. Uncheck all of the "Write" checkboxes.



## [FTP uploads and file permissions using Transmit](#)

There are also other ways, such as simply using the desktop GUI on a test install, if available.

## In Ubuntu (7.04) via GUI

For files:

1. right-click on a file and click on properties
2. then go to the permissions tab and there, you can change "Access:" to "Read and write" for Owner, Group, and Others.

To remove write privileges, just change the Owner, Group, and Others permissions to "Read-only".

For folders:

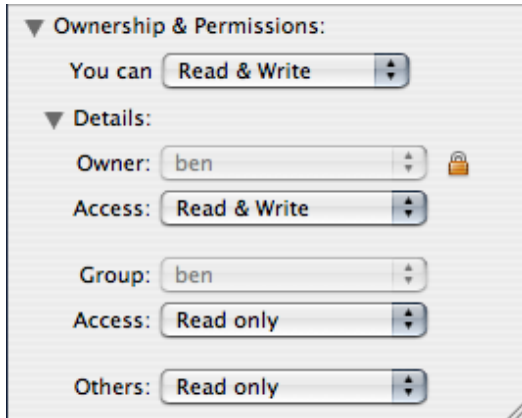
1. right-click on a folder and click on properties
2. then go to the permissions tab and there, change folder access to "Create and delete" for Owner, Group, and Others.

To remove write privileges to a folder, just change the Owner, Group, and Others permissions to "Access files".

## On Mac OS X 10.4 and earlier

1. Open a Finder window and navigate to the file or folder

2. You can see and set permissions by pressing Command-I on your file or directory or...
3. By control-clicking and selecting Get Info.

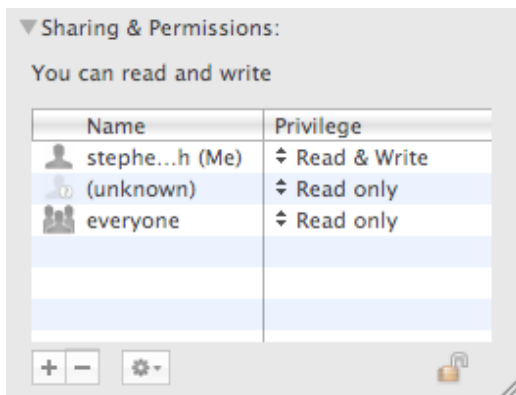


## On Mac OS X 10.5 and later

It's recommended that you modify permissions via the command-line by opening Terminal and using the instructions [above](#).

If you still need to edit permissions via the GUI:

1. Open a Finder window and navigate to the file or folder
2. You can see and set permissions by pressing Command-I on your file or directory or...
3. By control-clicking and selecting Get Info.
4. Click the lock in the lower right-hand corner of the window to authenticate.



## Explanation

Each file or directory has its owner. If you created the file yourself, you're the owner, and you can do virtually anything to the file, including change of permissions. Drupal, in the other hand, is an automated script acting on the system as "apache", "webserver" or some different identity, which is not the same as yours. So to allow Drupal to write, you need to grant write permissions to everyone, which can't be done by Drupal automatically.

## When you don't own the file

The same also works the other way: Any files created by Drupal (such as settings.php, or any uploaded and/or temporary files) are owned by the webserver process, and you may be unable to change/remove them if Drupal didn't give the permission. Sometimes it helps to manipulate the

parent directory, but mostly in such a case you need to remove the files as root user, if you have access to that, you might want to use [some helper PHP scripts](#) to execute the operation through PHP, which runs under the webserver process who owns the file. To avoid security risks, don't forget to remove any such helper scripts immediately after use!

## Cleaning up

It's important to note that the only folder in your Drupal install that needs write permissions is the "files" folder. It is *especially* important that you ensure that settings.php is set to read-only.

# Modifying Windows file permissions

This page will tell you how to change the file permissions on a Windows server.

## Through Windows Explorer

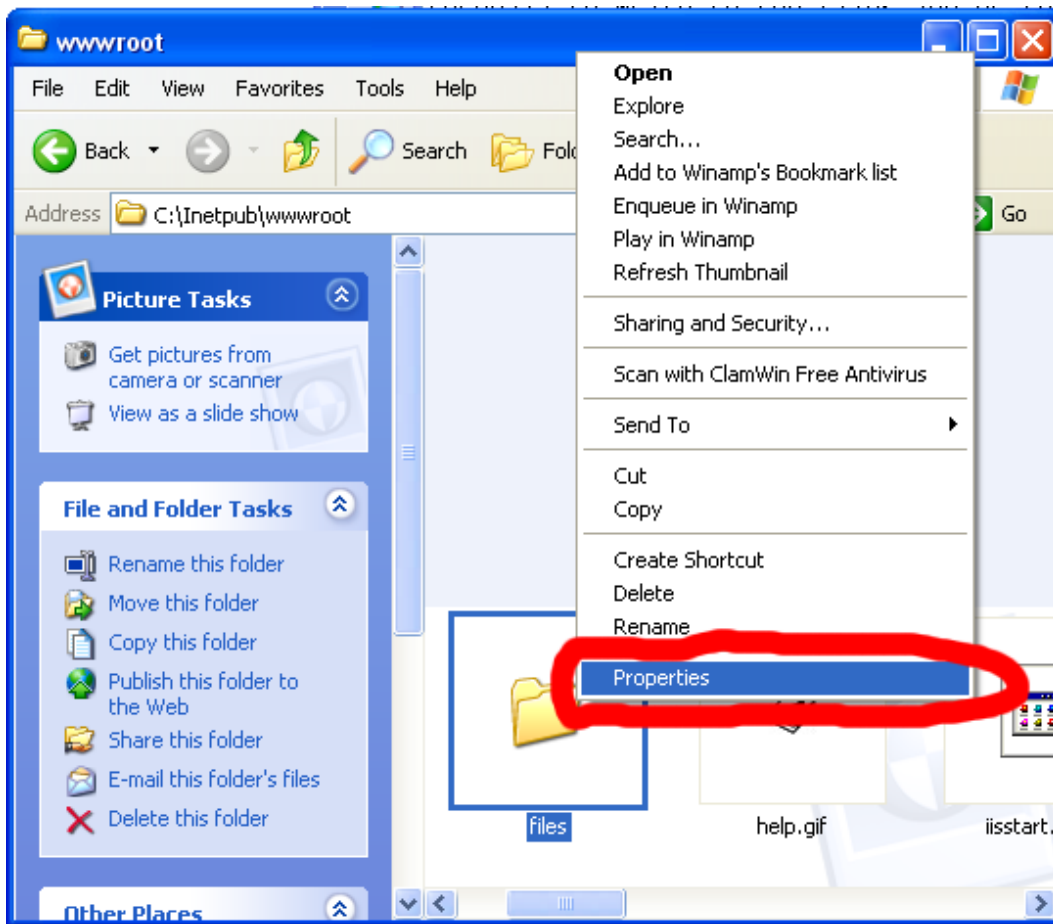
### *On Windows 2003 with IIS*

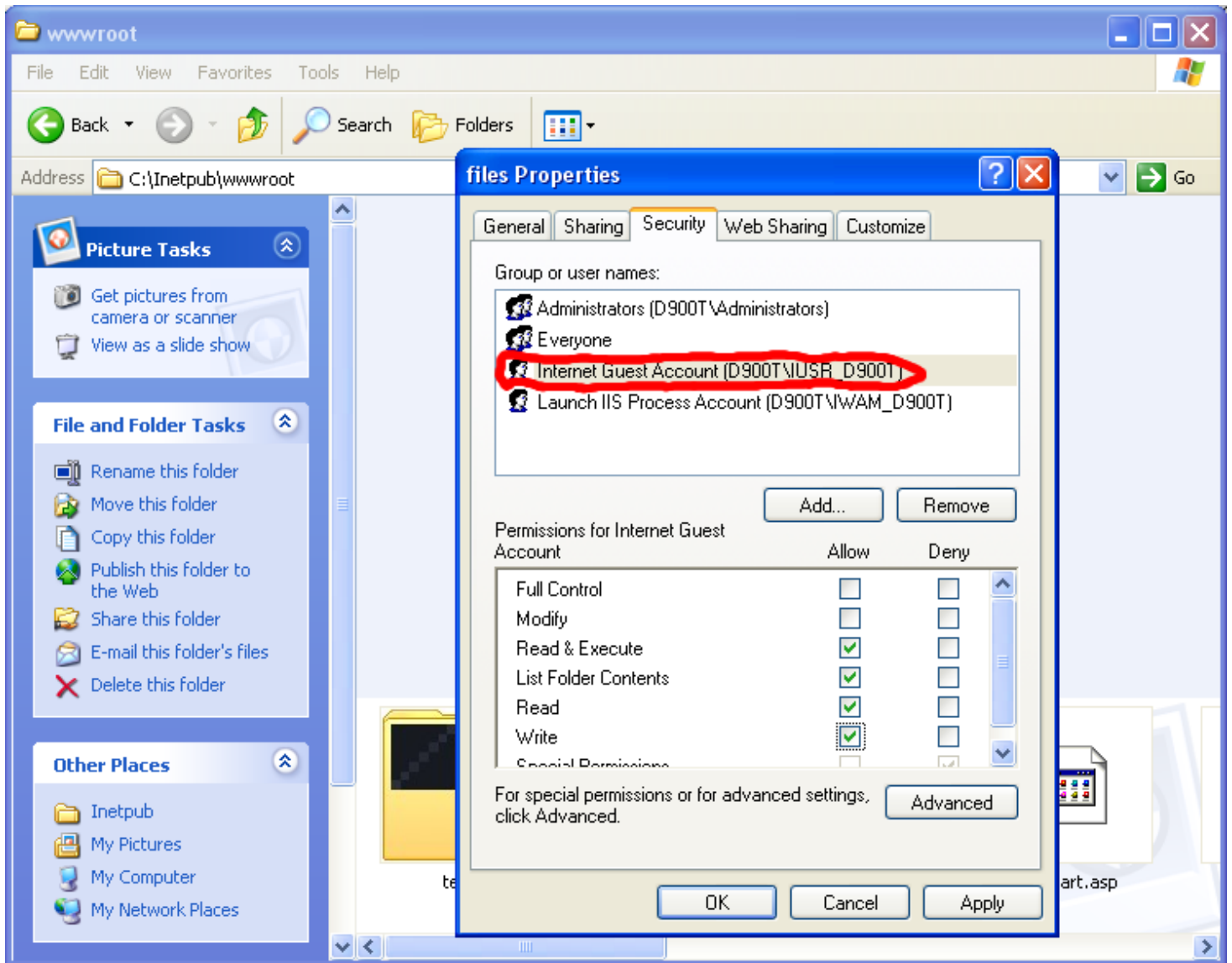
1. Navigate to the file or folder that needs its permissions changed.
2. Right-click on the file or folder and click on *Properties*.
3. Select the *Security* tab.
4. Click on the *Internet Guest Account* and make sure that the *Allow* checkbox is set for *Write* permissions.
5. If you are changing these permissions for the Drupal install process, when the installation is complete, for security reasons, you should revoke write permissions on the files or folders that Drupal no longer needs to write to.
6. To revoke write permissions, just uncheck it.

### *On Windows XP with IIS*

Windows XP's security tab is hidden by default, so before the permissions can be set you need to:

1. Go to *My Computer*.
2. Click on the *Tools>Folder Options* menu item.
3. Select the *View* tab.
4. Under *Advanced settings*, scroll down to the bottom and uncheck *Use simple file sharing (Recommended)*.
5. Click *Ok*.
6. Navigate to the file or folder that needs its permissions changed.
7. Right-click on the file or folder and click on *Properties*.
8. Select the *Security* tab.
9. Click on the *Internet Guest Account* and make sure that the *Allow* checkbox is set for *Write* permissions.
10. If you are changing these permissions for the Drupal install process, when the installation is complete, for security reasons, you should revoke write permissions on the files or folders that Drupal no longer needs to write to.
11. To revoke write permissions, just uncheck it.





NOTE: Be aware that certain IIS configuration options can disable file permissions regardless of the NTFS file system settings. For more information about IIS, see [this](#).

### Other access methods

If you do not have access to the windows GUI, then your web host has probably set up a web-based interface to enable you to add write permissions. Otherwise, contact your web host and ask them to allow PHP scripts (Drupal) write access on *specify files here*.

### External links

Here are links to either more information or alternate instructions to change permissions.

1. [How to set required NTFS permissions and user rights for an IIS 5.0 Web server - Microsoft.com](#)
2. [How to configure Web server permissions for Web content in IIS - Microsoft.com](#)
3. [Instructions with screenshots, from the Coppermine website](#)
4. [A blog post on with instructions for changing permissions on Windows 2003 or XP](#)

## Register globals should be disabled

## Background

*N.B.: need to check whether it is possible to upgrade existing sites to 6.x/7.x when `register_globals` is enabled.*

PHP's deprecated [Register Globals feature](#) is a general security risk, as discussed on that page. Under certain server configurations it can give rise to a specific cross site scripting vulnerability with Drupal core. The vulnerability is described in security announcement [SA-2008-007](#).

Therefore, since versions 5.6 and 6.x, Drupal won't install on a server which has `register_globals` enabled; nor will you be able to upgrade an existing site to 6.x (*need to check this*). Instead, you will be presented with a message such as:

Incompatible environment

The following error must be resolved before you can continue the installation process: `register_globals` is enabled. Drupal requires this configuration directive to be disabled.

Your site may not be secure when `register_globals` is enabled. The PHP manual has instructions for how to change configuration settings. (Currently using PHP `register_globals` Enabled ('1'))

*Note that the actual value of the `register_globals` setting under which Drupal is running is reported at the end of the message: 1 in this example.*

Existing 5.x and 4.7.x sites will continue to run but will display a warning message in the admin area when upgraded to 5.6 and higher or 4.7.11 and higher.

## How to disable `register_globals`

If you use shared hosting it may be best to try persuading your host to turn this feature off. Failing that, or if using your own server or VPS or localhost installation, you can try these approaches. Note that the configuration of the server may prevent them from having the desired effect, in which case you will need to speak to your host.

### If PHP is running as CGI ([how can I tell?](#))

You can try using a custom `php.ini` file located in Drupal's root folder (i.e. the folder containing Drupal's `index.php`). This will only work if your host has enabled the use of custom `php.ini` files.

So, create a file named `php.ini` in Drupal's root folder with the following line:

```
register_globals = off
```

If `php.ini` already exists then add the above line to it.

If this works, and if you created a new `php.ini` file, you may want to follow the instructions on the page [Creating a custom php.ini using the server default php.ini and configuration settings](#) in order to avoid inadvertently changing some of the server's PHP configuration options.

### If PHP is running as an Apache module ([how can I tell?](#))

Make sure that Drupal's main `.htaccess` file (the one in Drupal's root folder) includes the line:

```
php_value register_globals 0
```

This directive has been there since Drupal 4.2 (June 2003). You may want to add it again at the top of the file in case any customizations made to `.htaccess` are preventing the existing directive from

working properly.

In 7.x you should find the following line in `.htaccess`

```
php_flag register_globals 0
```

This 2nd form is [preferred](#) but in practical terms shouldn't make any difference from the first form.

Note that the configuration of some servers restricts what you can do in `.htaccess`; however, while this directive may not work on the cheapest hosting packages it should work fine on all reasonable quality packages, provided that PHP is running as an Apache module.

## If you are using your own server or localhost installation

The best approach would be to change the configuration of `register_globals` in the main `php.ini` configuration file. You can find out the location of this file by running `phpinfo()` (see below).

## If the above don't help

If your server is running PHP 4 by default then another option is to try to force Drupal to use PHP 5. This sometimes fixes the problem because `register_globals` is disabled by default in PHP 5, whereas it was enabled by default in PHP 4.

In Drupal's main `.htaccess` file, try adding the following line:

```
AddType x-mapp-php5 .php
```

If that doesn't help then speak to your host since if they do have PHP 5 available as well as PHP 4 then there will be some way of enabling it, but the details will vary from one host to another.

Finally, note that you can't use `ini_get()` (e.g. in `settings.php`) to change the `register_globals` setting since [it can't be modified at runtime](#).

## How can I tell if PHP is running as CGI or as an Apache module?

Create a file named `phpinfo.php` in Drupal's root folder (the file **must** be located here to guarantee accurate results), containing the following:

```
<?php
phpinfo();
?>
```

Then visit <http://example.com/phpinfo.php> (where <http://example.com> is the full URL of your Drupal installation). Near the top, look for Server API. If PHP is running as CGI then this should report something like "CGI" or "CGI/FastCGI "; if running as an Apache module it should report something like "Apache", "Apache handler" or "Apache 2.0 handler".

When finished you may want to remove the `phpinfo.php` file to prevent the possibility of revealing information about your server configuration.

## More information

For more information about how to change PHP configuration settings, see "[How to change configuration settings](#)" in the [PHP Manual](#).

# Creating a custom php.ini using the server default php.ini and configuration settings

## Introduction

This explains how to make a copy of the default php.ini file, so you can create a custom php.ini file for your Drupal site and make changes, such as increase memory limit, disable register\_globals, switch off safe mode, increase max file upload size etc.

The reason I created a custom php.ini file for my site was because:

- My shared host had register\_globals switched on for some reason and was slow to respond to tech.support
- I'm unable to access and edit the default php.ini for my sites
- adding in switches in the .htaccess file didn't work for me (because PHP is running as CGI, not as an Apache module)
- I am able to upload a custom **php.ini** file to the root folder of my Drupal installation (i.e. the folder containing Drupal's index.php), but, I don't know what else to put in it besides the register\_globals switch. In other words, when I upload a custom **php.ini**, it replaces the default **php.ini**..so it's missing all the other settings and breaks the site.

## Step by step solution

This is what I did...(thanks to Dublin Drupaller for putting me straight on this)

1. Make a copy of the default **php.ini** by creating a php file using the snippet below, editing it to suit your server paths (e.g. on some servers you might want to use `/usr/local/lib/php.ini` for the 1st path and `/home/YOURUSERNAME/public_html/drupal/php.ini` for the 2nd path) and uploading it to the root folder of your drupal site. Call it something like `gettheini.php`.

```
<?php system("cp /usr/local/php5/lib/php.ini /home/YOURUSERNAME/php.ini"); ?>
```

- If you are unsure where on the server your default php.ini is (there may be a number of different versions on your system, and it is important to get the right one), create a php file called **phpinfo.php** with the following in it `<?php phpinfo(); ?>`.

Upload it to your root folder and in your browser, go to **www.example.com/phpinfo.php**. Scroll down a bit and it should indicate the exact path for where your default **php.ini** is located on the server.

2. In your browser, go to **www.example.com/gettheini.php** (which should display a blank screen). All it does is create a copy of the default **php.ini** and puts in the folder you specified.
  - If you see a message like *Warning: system() has been disabled for security reasons in /home/YOURUSERNAME/get\_the\_ini.php on line 1* then you may need to try an alternative command to PHP's `system`, such as `exec`, `shell_exec` or `passthru`, or ask your host for help.
3. Now, when you go to the root folder of your Drupal site, using FTP, you should see a new **php.ini** file. That's a copy of the default **php.ini**. Open it in a text editor like [PS PAD](#), search for the string **register\_globals** and set it to off.
  - If you cannot see the new php.ini file then possibly the permissions on the directory which should contain it did not permit PHP to copy the file there. You may see an error message such as `cp: cannot create regular file`

`~/home/YOURUSERNAME/public_html/drupal/php.ini`: *Permission denied* in your error log. You may need to temporarily change the permissions on the directory from 755 to 777 (don't forget to change them back later), or ask your host for help.

4. Upload your new custom **php.ini** to the root folder of your Drupal site.
5. Add the following lines to your **.htaccess** file to keep prying eyes from looking at your **php.ini**

```
<Files php.ini>
order allow,deny
deny from all
</Files>
```

## SELinux may cause mysterious permission problems

Security Enhanced Linux (SELinux) is a relatively new, powerful mechanism for fine-grained access control on Linux systems. Properly configured and maintained, it offers much better protection from misbehaving programs and exploitable security weaknesses of server application stacks than conventional Unix systems can provide. Many distributions now come with SELinux support enabled by default, or at least make it available for installation. SELinux is installed/available on Red Hat, Fedora, Debian, Gentoo, SuSE, Slackware, and Ubuntu, among others.

If you (or your ISP) are running Drupal under an operating system which has SELinux installed and enabled, you may find that certain operations fail mysteriously. Symptoms include, for example, files not being written or read though the webserver has permissions; communications operations such as sending mail or attempting XMLRPC operations failing, although firewall permissions are OK, etc.

You can confirm that SELinux is causing the problem by turning SELinux off temporarily (run the command `setenforce 0` as root) and try the operation. If it succeeds, likely SELinux is the culprit. (I'm assuming that this is a development setup, not a production machine - SELinux is designed to protect your system, so turning it off on a production machine is not to be done lightly). You can get more information about exactly why SELinux is shutting you down by looking in the log files that it generates, for example, in `/var/log/audit/audit.log` on FC4. Look for 'avc' (access vector cache) messages.

Once you have tracked down exactly what aspect of SELinux policy is causing your operation to fail, you can modify the SELinux configuration to fix the problem. This may be as easy as turning on a boolean configuration setting in a configuration file, or as complicated as writing a new snippet of SELinux policy.

Using the avc messages, the supplied SELinux administration tools and a little bit of help from Google, the SELinux FAQs/tutorials on the web, and folks on the various the SELinux mailing lists, you should be able to find your way around configuring additional policy to get Drupal to do what you need.

I highly encourage you to bite the bullet and run with SELinux enabled, though it does involve a rather steep learning curve.

## Fedora Core 4 Users

Some people may be having SELinux problems on Fedora due to the installation instructions for Drupal. These suggest the use of `mv` to move the Drupal source files into the web root. `mv` by default preserves the context associated with the file so that, if the Drupal source archive was unpacked in a user's home directory, they will have the context `user_home_t`. The default SELinux settings on FC4 restrict `httpd` from reading files in users home directories (ie with a context of

user\_home\_t).

To check if this is your problem, instead of turning off SELinux as suggested above, you might try narrowing down the problem first by seeing if you can access the Drupal installation directory via your web browser after turning off the restriction on user's home directories. You can do this by running `setsebool httpd_enable_homedirs true`. If you can access Drupal after running this, then you need to reset the contexts on the moved files. You should undo the change you just made by running `setsebool httpd_enable_homedirs false`. (setsebool is in `/usr/sbin` if you get a no such file or directory error when trying to use it - `/usr/sbin` may not be in your path.)

If you use `mv` and are getting 403 Forbidden errors when following the installation instructions, check the SELinux context for your Drupal website. Eg. if your files are in `/var/www/html/myDrupal`, you can check the contexts using `ls -laZ /var/www/html/myDrupal`. If these files have the `user_home_t` context, then you can fix this by running `chcon -R -t httpd_sys_content_t /var/www/html/myDrupal`. Change `/var/www/html/myDrupal` to match where your installation is located.

A global solution might be to suggest using `cp` instead of `mv`, because `cp` creates a new file in the web root, the new files inherit the context of the directory - which will be the desired `httpd_sys_content_t`.

## Fedora 7 Users

For some reason there's a rogue boolean for `httpd` in the Other section called "`httpd_can_sendmail`". This should surely be listed under HTTPD Service. Simply turn this on and you are all good!

# Splitting large sql batch files when migrating to a new host

If you have to migrate a large sql batch file from your old server, chances are that the script in `phpmyadmin` will time out and only part of your database will be migrated. The script below will let you split a sql batch file in small chunks. It is intended to be run on your local web server (or CL).

```
<?php
print "hello<br>";

$splitEvery = 840000;
$base = "/home/ren/dev/ms/";
$ext = '.sql';
$file_name = $base . 'db' . $ext;

// (the first) output file
$outCnt = 1;
$out = fopen($base . 'db_' . $outCnt . '.sql', "w");

// read input file
$fp = fopen($file_name, "r");
while($line = fgets($fp)) {

    $strLen += strlen($line);
    if ($strLen > $splitEvery && preg_match("/^(^[\r\n]*|[\r\n]+)[\s\t]*[\r\n]+/", $line) ) {
        print 'empty line after ' . $strLen . '<br>';
        $strLen = 0;

        //new output file
        fclose($out);
        $outCnt++;
        $out = fopen($base . 'db_' . $outCnt . '.sql', "w");
    }
}
```

```
//write to current out file
fwrite($out, "$line");
}
fclose($fp);
print "done"; ?>
```

## Typical webhosting setups

The typical webhost sets up their servers so that each website gets a user account to manage it (eg your FTP login). Each user account has a home directory where they can create files. These website files will generally be owned by the user account that uploaded them.

The default permissions on these files usually won't allow other users to write to them for obvious reasons.

On most servers by default, the webserver runs under a different user account. eg on a Debian Linux server, Apache usually runs as the www-data user. Some webhosts though might have installed Apache modules that switch the webserver to run as the owner of the files it is serving up.

Note: To find out what user the webserver is running as, you can use the `phpinfo()` function. There are various ways of doing this - you can upload your own php file to do so, or that functionality might already be built into tools you already have available like `phpmyadmin`.

Most documents or people answering forums tend to assume that the webserver is running as a different user account. The explanations get a bit too fiddly and convoluted otherwise. But if your server does run the webserver as your own account, keep that in mind when reading other docs or answers - they might be aimed at the case where it runs as a different user.

How the web server is set up determines which permissions get used to control access to your files. If the webserver runs under your user account, then Unix uses the file owners permissions to provide access to your files. If the webserver runs as a different user, then Unix uses either the 'group' or 'other' permissions to determine access depending on whether or not the webserver account is in the files group or not.

## Using PHP to change file permissions on the webserver

A module, [Take Control](#) is now available for Drupal 6.x which you can use to change permissions of your files/folders on the server.

**Warning: You cannot undo the file permissions that are changed by the script below. Proceed with extreme caution.**

Symptom:

- "Permission Denied" When trying to work with your site over FTP.
- You cannot modify directories inside the /files folder that were created by Drupal.

Solution:

- Tell Apache to give you back control of your files.

One side effect of having files created by Drupal (eg the image module), is that your user account might not have ownership of them any more. And you might not be able to delete the or move them around.

There is a workaround though. You can create a small PHP script containing the commands you want to carry out and upload it to the server. Once uploaded, you can run it from your web browser by entering the URL for it. The script will run as the user account the webserver runs as. Be sure to remove the script after you have used it though.

simple case:

Use a drupal PHP-format page (eg just a php code snippet) to run

```
<?php
`chmod -R a+w sites/default/files`;
?>
```

and you (and the rest of the neighborhood!) should have write access so you can write/delete anything in your files and images directory via shell/FTP again. Modify the command as needed.

If that doesn't work, and particularly if you are trying to uninstall Drupal and want to be able to erase all files, save the following snippet as a php file (e.g. **fix.php**). It will try to make all directories and files writable in a recursive fashion. If you put this in the root folder of your Drupal installation and run it by going to <http://www.example.com/fix.php> it should operate on all possible files within your Drupal site. If you put it in your files directory, it will operate there, e.g: <http://www.example.com/sites/default/files/fix.php>

**Important:** this code should only be used if you remember to delete it immediately after use. As above, its use may put your site into an insecure state.

```
<?php
file_fix_directory(dirname(__FILE__));

function file_fix_directory($dir, $nomask = array('.', '..', 'CVS')) {
    if (is_dir($dir)) {
        // Try to make each directory world writable.
        if (@chmod($dir, 0777)) {
            echo "<p>Made writable: " . $dir . "</p>";
        }
    }
    if (is_dir($dir) && $handle = opendir($dir)) {
        while (false !== ($file = readdir($handle))) {
            if (!in_array($file, $nomask) && $file[0] != '.') {
                if (is_dir("$dir/$file")) {
                    // Recurse into subdirectories
                    file_fix_directory("$dir/$file", $nomask);
                }
                else {
                    $filename = "$dir/$file";
                    // Try to make each file world writable.
                    if (@chmod($filename, 0666)) {
                        echo "<p>Made writable: " . $filename . "</p>";
                    }
                }
            }
        }
        closedir($handle);
    }
}
?>
```

Documentation for PHP filesystem functions:

<http://php.net/manual/ref.filesystem.php>

And documentation for the PHP system() function:

<http://php.net/manual/function.system.php>

With these techniques you should be able to get the webserver to do anything the Unix shell can do. Some hosts however limit your ability to use shell-level commands like this, in which case you must approach them to ask for the appropriate fix.

Take care doing this though. These are the kind of tasks where it pays to practice them on your test instance first - you do have a test instance right?

## What do all those Unix commands mean?

You will notice references to Unix commands in some documentation or forum posts and might be a little unsure what they do. The reason a lot of the docs or forum posts just list the underlying commands is because they are a much more universal interface and the process can be expressed quickly without writing long winded paragraphs describing things.

You may wonder why you would want to know what these Unix commands do, but once you have a better idea of what they do you can then translate them to the functionality provided by your control panel or FTP client etc.

### Some Unix command you'll see mentioned:

chmod

changes the permissions on files and directories. The '-R' (for recursive) option switch changes the permissions on subdirectories and files as well.

mv

moves or renames files and directories. In general to rename a file or a directory you 'move' it to its new name.

rm

removes (ie deletes) files and/or directories. To remove a directory you need to use the r (for recursive) switch ie 'rm -r'.

cp

copies files and directories. The '-p' (for preserve) option switch also copies permissions and ownerships etc. The '-r' (for recursive) option switch also copies subdirectories and files as well. The '-a' (for archive) option switch includes both -r and -p options.

mkdir

creates a directory.

ln

creates a filesystem link. The main kind you will see mentioned is a 'symbolic link' (using the '-s' option switch) which behaves a bit like a shortcut in windows. You can create a link to a file or directory located somewhere else and it will behave just like a copy of that file or directory. But because they are linked rather than just copied, changes to one are reflected in the other.

wget

a command to download web pages or files off the net and save them to disk.

tar

a zipping and unzipping utility. eg the Drupal download is what's called a tarball. tar is used to 'unzip' the tarball into a subdirectory.

mysql and mysqladmin

command line utilities for connecting to and managing a MySQL database. Just about anything they can do can also be done by phpmyadmin.

### Special characters:

- / the Unix directory separator (just like with URLs) and also represents the root directory. When a path starts with / it is an absolute path ie it starts with the root directory. A path ending in / is an optional way of explicitly referring to a directory rather than a file.
- .. refers to the parent directory (just like in Windows). A path starting with this is relative to the current directory. Can be chained together eg ../../ refers to the parent of the parent directory.
- .
- ~ like '..' but refers to the current directory.
- ~ it refers the user's current home directory (e.g., /home/username)
- \* wildcard that matches any number of characters (like in Windows)
- ? wildcard that matches just one character (like in Windows)

## What permissions does Drupal need?

Now that you know which user the webserver runs as, you'll need to make sure your file and directory permissions are set properly. If you set them too tight Drupal won't run properly or even at all. Too loose and you run a higher risk of security breaches.

Most of the time, any files you upload should end up with the correct permissions to run a basic Drupal site. Your webhost will have set the default permissions so they will be able to be read by the webserver.

Where things get trickier is with the infamous 'files' directory. If you install or enable modules that upload files or images, they get stored under this directory. To do this, the webserver will need write access to this directory. What this also means is that any files you upload will be owned by the webserver user account and may not be able to be moved or deleted any more by your FTP client or control panel as you might not have enough permissions. In most cases don't worry about this too much, but if you really have to delete some of these files manually there are ways around the problem by uploading your own PHP scripts for the webserver to run and change the permissions.

Basic summary of file permissions for a Drupal installation:

All the Drupal files (eg .php, .module, .css, .theme and images etc) will need to be able to be read by the webserver account. The 'files' is generally the only directory will need to be writable by the webserver account.

If you get error messages complaining about missing files, or not being able to open/read a file etc and you know that the file really is there - chances are that the webserver doesn't have read permissions for it. Recheck the permissions on subdirectories etc.

Ideally your settings.php file won't be world readable as it contains your database connection string (with password). But sometimes you can't avoid it if making it world readable is the only way your webserver can read it.

Along similar lines, ideally for security reasons you won't have to make anything world writable. But on a lot of webhosts it is hard to avoid having to make the 'files' directory world writable. What ever you do don't go making anything other than the 'files' directory world writable. That makes it easy for other users to overwrite your Drupal files.

If it is possible through your admin interface it can be useful to assign the group ownership of the 'files' directory to the group the webserver runs as, and allow group write access. This improves

security a bit by not requiring the directory to be world writable.

Quite often you will hear people talk about setting permissions to 777 which is no restrictions at all. While that is a good way to isolate any permissions issues when troubleshooting, you should try to tighten the permissions back again afterwards if possible. Preferably you wouldn't need to use 777 permissions anywhere.

Another thing to keep in mind is that a permission setting only tells part of the story about access - when troubleshooting it is also important to know who the owner and group are, and well as which user the webserver runs as.

## File Permissions in a Nutshell

```
/default on 755
/default/files including all subfolders and files on 744 (or 755)
/default/themes including all subfolders and files on 755
/default/modules including all subfolders and files on 755
/default/settings.php and /default/default.settings.php on 444
```

## Why is this uploading stuff so difficult?

The reason it is more complicated when you need to upload files and/or images is because you are now operating the web server in a different way.

With standard HTML only Drupal configuration, Drupal is not creating any files on the web servers filesystem. When in that mode it runs like a standard web site - all the web server needs to do is read the files, it doesn't need to write anything. All the content you submit to the site is going into the database not the filesystem.

But to be able to upload files or images through the Drupal interface, now Drupal needs to be able to write files to the filesystem on the server.

This gets tricky because most of the time you don't want the webserver to be able to write files into your site directory. Just think what would happen if someone exploited a vulnerability in someone else's website on that server - they could now overwrite any of your website files very easily. Just think what you could get up to just by uploading PHP files if other peoples sites had wide open write access - no don't try this.

So generally web servers aren't allowed to write files in peoples websites. If you do want to Drupal to be able to write files into a subdirectory - you will have to manually weaken the filesystem security to make it work.

This isn't a Drupal issue - any code (eg PHP, Perl, Python etc etc) run by the web server will need those permissions weakened if they are to be able to write files to part of your web directory.

You may also run into upload filesize problems that require changing settings like: `memory_limit`, `post_max_size`, `upload_max_filesize` etc. These settings are there to limit what any one website can do to overload the server. If these are too high, it allows one site to completely tie up the server causing trouble for all the other sites.

Its all just the nature of sharing a server with lots of other people. There will be security and resource limits put in place to stop a badly behaved user causing trouble for the others.

Your webhost is trying to strike a balance between ease of use and security. Different webhosts will

draw the line in different places - that is why it is impossible to write detailed docs that cover all situations.

## Account, Permission and Login Issues

The following pages provide information about problems you might encounter with setting up accounts, configuring permissions and logging in to your site.

See Also:

- [How to log in and repair your site if clean URLs stop working](#) (all links break or redirect you to the front page)

## Anonymous users and \$\_SESSION.

While testing our main web site I noticed that anonymous users were not able to store and save values in \$\_SESSION. Running down the path of what could be going wrong I called functions such as `session_id()`, and `debug_print_backtrace()` to see what might be causing this anomaly. I could not find anything relevant and then decided to do a few searches in the forums. I came across a few posts that seemed like they might be a solution but one finally caught my eye:

<http://drupal.org/node/192165#comment-630167>

Now this post mentioned that the user whose uid was zero is missing from the database and that restoring it was the answer to the issue. Opening a terminal and logging into our database I executed `SELECT * FROM users WHERE uid=0;`, the results returned zero rows for the query. Thinking back to when Drupal was first installed, I found a user with a uid of zero and though that this user with uid set to zero, no user name and password was a bug from installation and deleted the row.

If \$\_SESSION is not being read from and written to as an anonymous user from page to page, take a look at the users table in the database. Chances are that you can save yourself a lot of debugging time by making sure that a user of uid set to zero exists. If there isn't one in the database there is a simple fix for this issue, execute the following SQL command: `INSERT INTO users (uid) VALUES (0);`

Clear your cache and then go back to the pages in question and see if it makes any differences in what values are stored in \$\_SESSION as an anonymous user.

## Another tip for User 0

This was more my problem, since I had a user 0:

<http://drupal.org/node/205933>

## Deleted first user account?

If, by accident, you happen to delete the original "Administrator" account/user, you will need to recreate it.

```
INSERT INTO users (uid, name, pass) VALUES ('1', 'yourname', md5('yourpassword'));
```

you can also try and work with the methods below:

UID1 is the 'super admin' account.

- create a new account with user name and password of your choice : this account will initially be a regular user account, to be turned into an "Administrator" account following the next step.
- go to database table "users" and change the UID of this new account to "1" - You may now login to this account as the super admin account.

if you use phpmyadmin for MySQL database administration, this should only take a few seconds to fix.

## user\_register not enabled

The above only works if the "user\_register" variable is *\*not\** set to `s:1:"0";`.

```
UPDATE variable SET value = 's:1:"1";' WHERE name = 'user_register';
DELETE FROM cache WHERE cid = 'variables';
```

This will change your site so that you can follow the above instructions.

## Prevent account cancellation for uid 1

For Drupal 5 and 6, you may want to check out the [Protect Critical Users](#) module.

For Drupal 7, this issue will be probably solved by then. See: [#46149: Prevent account cancellation for uid 1](#)

# Failure to Login as Admin after initial installation

If the configuration in an Apache 2 installation is not correct with respect to the naming of the sites-enabled and sites-available entries in the /etc/apache2 configuration subdirectories, the sessions become messed up and this prevents login without any errors for diagnosis.

Thus, if you are installing fresh and encountering a situation where you cannot login with your admin user, check the Apache configuration to ensure that all naming is aligned as so:

```
>ls /etc/apache2/sites-available
drupal
>ls -l /etc/apache2/sites-enabled
000-drupal -> /etc/apache2/sites-available/drupal
and NOT
drupal -> /etc/apache2/sites-available/drupal
OR
000-default -> /etc/apache2/sites-available/drupal
```

Also ensure that your apache configuration file looks something like:

```
NameVirtualHost *:80
<VirtualHost *:80>
    ServerAdmin webmaster@<your domain>

    DocumentRoot /var/www/drupal
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /var/www/drupal>
```

```

        Options FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>

    ErrorLog /var/log/apache2/drupal_error.log

    # Possible values include: debug, info, notice, warn, error, crit,
    # alert, emerg.
    LogLevel warn

    CustomLog /var/log/apache2/drupal_access.log combined
    ServerSignature On
</VirtualHost>

```

and your `/etc/apache2/ports.conf` looks as follows (at least this entry):

```
Listen 80
```

## Forgotten your Drupal account password

When the Drupal account password for user 1 (the administrator) is lost and the email notification doesn't work, it is possible to set the password via a database query.

Execute the following query on the Drupal database:

```
UPDATE `users` SET pass = MD5('newpwd') WHERE uid=1;
```

Of course, change 'newpwd' to the desired password.

To execute this query it will be necessary to login to the database. This is typically done through the command line or through a GUI interface such as phpMyAdmin.

## How to log in once you have turned your site off-line for maintenance

Once you have turned your site off-line using *admin » settings » site maintenance*(`admin/settings`), you can log back in by visiting:

```
http://example.com/?q=user
```

**Note:** Use the literal word **user**, not your username or user id. Do, however, replace *example.com* with the proper URL parts pointing to your website.

You need to enter the username and password of an account with permission to administer site configuration. User 1 has this ability as a safety net, should you not have this turned on in other administrator account(s).

If you attempted to log in with a username that does not have the "administer site configuration" permission you will be stuck in that user account until you clear the browser's cookies. Simply instruct your browser to delete cookies for the site and visit the referenced URL above again. You can also use another browser to access the URL.

To return the site to online mode visit the following page:

Drupal 4: *administer » settings* (`admin/settings`)

Drupal 5&6: *Administer* » *Site configuration* » *Site maintenance* (admin/settings/site-maintenance).

On the page, set Site status to Online.

If you are truly desperate you can try the following database queries to restore access:

```
UPDATE variable SET value = 's:1:"0";' WHERE name= 'site_offline';
DELETE FROM cache WHERE cid = 'variables';
```

## Log in after disabling the User login block

If you disabled the User login block and need to log in, you can visit the login page on:

`http://example.com/?q=user`

The alternative `http://example.com/user` will only work when clean URLs are enabled.

**Note:** Use the literal word **user**, not your username or user id.

## Login after Blocking user 1 by an access rule

Say you accidentally 'block' your admin account (user 1) by an access rule (for example `"%admin%"`) and cannot log in. You can undo this by accessing the table `access` and deleting the row where the `mask` field contains the access rule (in our example `"%admin%"`.)

Using phpMyAdmin this is done by:

1. Clicking on the name of the drupal database in the left menu (e.g. `"_Drupal"`)
2. Beside the table name (`access`) and under the *Action* column click on the *Browse* icon.
3. Find the row in the table with the `mask` field (e.g. `"%admin%"`) and click on the *delete* icon beside it.
4. Press *OK* on the confirmation message.
5. Now login normally from the frontpage...

## Login doesn't "stick" after upgrade to PHP 5.2+

This is a specific problem that affects older versions of Drupal on a server running PHP 5.2 or later. Often it arises when PHP is upgraded (sometimes without your knowing!).

Symptom: You can appear to log in OK, but on the very next page view the system seems to forget that you have just logged in and you revert to being an "anonymous" user.

If you are running Drupal 5.x or later then this specific problem won't affect you, so if you still have trouble staying logged in then check [here](#).

This problem occurs due to the way PHP 5.2+ handles objects in session handlers. It affects all Drupal versions up to Drupal 4.6.10, Drupal 4.7.4 and Drupal 5-beta 1 inclusive.

**The best method of fixing this is to upgrade to the latest official release of a supported version of Drupal. Currently (Dec 2008) this means the [latest release of either the 5.x or](#)**

## [the 6.x series.](#)

For those wishing not to do a full upgrade yet but who need an "instant fix", you can add the following line at the bottom of your settings.php file:

```
// Temporary fix to login/sessions problem.  
// Remove this line when upgrading to 4.6.11, 4.7.11 or 5.x or later.  
register_shutdown_function('session_write_close');
```

You should regard this as a temporary fix only. If you subsequently upgrade your site to a newer version of Drupal you should remove this line from settings.php to prevent potential incompatibility with future versions of Drupal.

The problem is fixed in the 4.6.11 and 4.7.5 releases of Drupal. So another option for 4.6.x and 4.7.x users is to upgrade to the final release of the 4.6.x/4.7.x series - 4.6.11 or 4.7.11, which are available [here](#).

Note that while Drupal 5.0 doesn't exhibit this particular problem with PHP 5.2 it is still not fully compatible with this version of PHP; Drupal 5.1 *is* fully compatible (but of course doesn't contain the security fixes in the latest 5.x release).

Here is the [original issue](#) in which this problem was identified and fixed.

## Login doesn't work or must be done twice

There are at least two possible reasons why you can experience a symptom of "Visit the site, login, and while the username and password are correct the user does not appear to be logged in."

### Cookies

Make sure cookies are enabled in your browser.

### Cache Problems

It seems that sometimes the cache doesn't get updated when logging into a site so even after logging in the user is still shown the cached version of the site: not logged in. The source of this problem (browser, webserver, Drupal settings) is not completely understood.

One solution seems to be to disable the Drupal cache of the site, though that has the undesired side effect of increasing the load on your site.

Another workaround is for the users to hit the browser refresh button which seems to work fairly well, but has the drawback of not being easily discoverable and is a hassle for every user of the site. On a site where only one or two users login, this is not a problem. On a community site it is a bigger problem.

For the next version of Drupal - the problem is fixed as a result of [this issue](#).

## Logging in at www on a site with no www in the baseurl

This problem is more easily understood and fixed. If a site is set with <http://example.com> as the base URL and the user types [www.example.com](http://www.example.com) into his URL then he will still be presented the main page. When the user uses the login block to login then Drupal will send them a cookie with

[www.example.com](http://www.example.com) as the domain and will forward the user to <http://example.com> where the cookie will not be valid and the user will not be logged in. One solution to this problem is to configure your server so that the URL and the Drupal settings match perfectly. If you are using apache and .htaccess is enabled you can achieve this with an .htaccess rewrite directive inside of the "Various rewrite rules" section:

```
RewriteCond %{HTTP_HOST} ^www\.example\.com$ [NC]
RewriteRule ^(.*)$ http://example.com/$1 [R=301,L]
```

While the example has always been to remove the www, the problem and this solution can be reversed for adding www.

This second issue has been fixed for the "next" version of Drupal in [this issue](#)

## Unblocking an account using SQL

Say you accidentally 'block' your admin account and cannot log in. From within a MySQL client you can run this command to unblock it

```
update users set status = 1 where uid = 1;
```

## Locked Out

This may be really basic, but I can imagine it happening to other people, so... Since the first person to login to a fresh drupal installation has full administrative privileges, the first time I installed drupal, I felt some urgency in locking down the site. After installation, I added a couple regular user accounts with less god-like powers. Under **administer->access controls**, I added rules to allow these specific users, and then added a rule to deny all users (except these specifically allowed users). I had assumed that the special account with uid=1 would be exempt. Wrong. As soon as I clicked the button, I found myself locked out from my own site. I then wished I had an undo button.

The equivalent of the undo button is to kill that last rule from the access table in the database. Using the mysql monitor, you can look at the access table:

```
mysql> select * from access;
```

yields something like:

aid	mask	type	status
1	bobby	user	1
2	peter	user	1
3	greg	user	1
4	alice	user	1
5	%	user	0

To kill that last rule, you can:

```
delete from access where aid='5';
```

The right way to have done this would have been to add the uid=1 superadmin account to the list of authorized users before setting the rule to deny all users. Live and learn, eh?

## Blank pages and missing content

The following pages provide information that may help you if you find blank pages in your site or if content and page elements are missing.

See also [Plain unstyled output](#) if you see *some* content but no navigation.

## My modules page is blank, says "page not found" or returns an error message

If you attempt to load admin/build/modules and get a blank page, a message that says "The requested page could not be found" or a server error, this is most likely a memory issue.

When viewing this page, modules are (since Drupal 5.x) only loaded if they are enabled (previously, even disabled modules were loaded). Even so, a fair amount of ancillary processing takes place on this page and can cause PHP to run out of available memory. Note that all modules' .info files *are* loaded whether or not the modules are enabled.

There are two fixes:

1. Increase PHP's memory limit, e.g. to set it to 16MB try one of these:
  - `memory_limit = 16M` to your system's main php.ini file (recommended, if you have access. In case you don't, you can try to upload your own php.ini file in the root folder of your Drupal installation, but bear in mind that this will only have an effect if PHP is running as CGI)
  - `ini_set('memory_limit', '16M');` in your sites/default/settings.php file (this doesn't work on all servers)
  - `php_value memory_limit 16M` in your .htaccess file in the Drupal root (this only works if PHP is running as an Apache module)

Depending on the amount of modules you have enabled and their 'impact' on the site you may need to increase the `memory_limit` even more, but don't exceed your server memory limit as it could cause your site more problems. Experiment with what memory value works for your needs. **Some people find they need to set the memory to 24M or 32MB or higher** (e.g. 96MB is recommended for a site with built-in image processing using ImageAPI GD).

2. You can reduce the memory required by your site by disabling modules by directly editing the {system} table in the database and setting the status column to 0. It's best just to disable contributed modules, and preferably those that are not dependents of others. Take a database backup first, in case you mess things up.

*Prior to Drupal 5.x a quick fix was to remove disabled modules' folders from your server; however, this now has negligible impact since it only prevents their .info files from being loaded.*

- *Note: If you had a module enabled at one time, before removing it you may wish to first click the "Uninstall" tab on the admin/build/modules page, as this will allow you to remove disabled modules' database tables. Once you delete the files from the server, you won't be able to use the Uninstall tab to remove that module's information from the database.*

Note that some hosts allow a php.ini file in the root of your site.

All fatal errors can result in a blank modules page. If you want to be sure if the memory limit is causing this problem, you should check your web server error logs. Hunt for a line that looks like:

```
Fatal error: Allowed memory size of 8388608 bytes exhausted (tried to allocate 418591 bytes)
in /path/to/drupal/includes/database.mysql.inc on line 29
```

That indicates that Drupal needed more memory than PHP was allowed to give it.

Always keep in mind that generally, "Less is More." The less memory your installation consumes, the faster it is, and the more people can visit your site at one time.

You may have to restart Apache for the configuration changes to take effect, especially if you edit php.ini.

If you get an "internal server error" message when you try to access the admin modules page, read the [white screen of death WSOD page](#) because it may be able to solve your problem.

## How do I get the User Login block back

If you have disabled the User Login block, you can always log in to your site by visiting `example.com/?q=user` (or `example.com/user` if you have clean URLs enabled).

Once you have logged in to an account with sufficient permissions, you can go to the block administration page to turn the User Login block back on:

Drupal 4: Administer > Blocks  
`example.com/?q=admin/block` or `example.com/admin/block` with clean URLs

Drupal 5/6: Administer > Site Building > Blocks  
`example.com/?q=admin/build/block` or `example.com/admin/build/block`

Drupal 7: Administer > Structure > Blocks  
`example.com/?q=admin/structure/block` or `example.com/admin/structure/block`

## Admin pages hang or are very slow

Since Drupal 6, [Update Status module](#) has been included in the core download.

This module makes your Drupal website connect to drupal.org so that it can check for newer versions of Drupal and of core and contributed themes. However, some hosts do not allow the server to make outgoing connections, and this can cause the admin pages to "hang". This can also be a problem if you are running Drupal on your own computer, for example, behind a corporate firewall/proxy. There are also reports of this problem when using Vista/WAMP.

The Update Status module is enabled by default in Drupal 6 (if you leave the checkbox "Update notifications: Check for updates automatically - With this option enabled, Drupal will notify you when new releases are available..." checked).

To turn it off, visit the modules admin page (Admin -> Site building -> Modules) at path `/admin/build/modules`. If you can't reach that page then you will need to edit the database manually: see <http://drupal.org/node/157632>.

There are open issues to improve the behaviour of Drupal in this situation [need references]...

## All My Pages are Blank!

The following advice may be helpful for users of the Web.Developer Server Suite and others testing

their sites through Windows and on their own server:

If you find yourself facing the "White Screen of Death" (a Drupal page that loads with nothing, just a white screen), you can visit a page that solves most folks' problems here: <http://drupal.org/node/158043>. But if you find that EVERY NODE YOU HAVE is blank, chances are Apache simply isn't running.

Chances are you're testing your site through a service like Web.Developer. If that's the case, confirm this by opening your Web.Developer Controller (WDController.exe, likely in C:\www) and trying to run Apache. If you get Error[1], make sure your port is open. Perhaps you're running Skype in the background, in which case you shut it down totally, restart Apache in WDController, and now since the Port has been freed up, you can use Web.Developer (you can also now happily restart Skype as well).

## How do I get the Navigation block back

A common question in the forums, is you disable a block without realizing the implications of what it does. For instance the main menu/navigation block.

Visit <http://www.example.com/?q=admin/build/block> to change block settings.

Visit <http://www.example.com/?q=admin/build/menu> to change menu items.

## Plain unstyled HTML output (Theme unavailable)

If your site appears as plain HTML with missing styles and/or images:

Try a view-source and see what the path to the theme CSS in the header is. Does it look correct? Sometimes this can get out of sync when migrating between sites that are almost-but-not-quite identical.

Some things to check:

1. It could be **you are trying to use a theme that is not active**. This can happen when you have set the default theme in settings.php but that theme has not yet been added or configured. It can also occur when did not switch back to the default theme before upgrade. To fix that issue, go to admin > build > themes and select the default theme (Garland in 5.x and Bluemarine in 4.7 and 4.6).
2. Your `$base_url` setting (in your `settings.php` file) may be misconfigured. The `$base_url` setting is what gets prepended to all paths for extra resources like stylesheets, images, javascript includes and form handlers. If the browser can't find those resources on that path, then they will be missing from the rendered HTML you see in your browser.

### example:

When developing and testing on a local Drupal version you might use `localhost` in your `$base_url`. This works fine when using the browser on the development computer, but when you access Drupal from another computer all these extra resources are missing. What is happening is that browser on the other computer is trying to access these extra resources from `localhost` and failing to find them there. The solution is to change `$base_url` to use a hostname that points to Drupal from both computers (ie not `localhost`).

3. This can also happen if the expected theme is *deleted*, *renamed* or *moved* by accident. Occasionally if you are using multisites and have placed the theme into the `default` or `sitename` folder instead of `all/themes` the theme will be invisible after a migration. In this case you can usually visit your themes configuration page to fix it. If the navigation block is unavailable, enter the URL `?q=admin/build/themes` directly. ([You may have to log in the hard way first](#)). Often, **just viewing and saving that themes admin page** is enough to self-repair, as the system can now locate the real theme path by name. If not, you can at least choose a default theme to use and troubleshoot from there - find where your theme has gone.
4. Check your theme folder is readable! Sometimes a bad copy or restore from backup can put the files there, but not readable to the web process.
5. When using css aggregation, your system needs to be able to write to the `sites/{sitename}/files/` folder (as always). Ensure that your files folder is still correctly writable. Try turning off css aggregation to test.

## The White Screen of Death (Completely Blank Page)

Occasionally a site user or developer will navigate to a page and suddenly the page content disappears, and is completely blank. No content. No errors. Nothing. This often, but not always, happens after updating a module, theme, or Drupal core. This is what is referred to by most members of the Drupal community as the White Screen of Death or WSOD. There are several reasons why this might occur, and therefore several possible solutions to the issue.

(Note: The suggestions on this page might solve the problem even when you do not get the WSOD as it relates to an Internal Server Error.)

### "Invisible" Errors

If error reporting is turned off, you could be getting a fatal error but not seeing it. On a production site, it is common to have error reporting turned off. If that is the case and PHP has hit an unrecoverable error, neither an error nor content will be displayed, therefore you end up with a completely blank page.

What you can do about this is either turn on PHP error reporting so it displays a message on the page itself, or check your log files (from the server) to look for the error. How to do both of these are explained below.

### Enable Error Reporting

Although it may be turned off on commercial hosts and production sites (for good reason, so that users do not see the errors), these errors are one of your best tools for troubleshooting. To enable error reporting, temporarily edit your `index.php` file (normally located in your root directory) directly after the first opening PHP tag (do *not* edit the actual file info!) to add the following:

```
<?php
error_reporting(E_ALL);
ini_set('display_errors', TRUE);
ini_set('display_startup_errors', TRUE);

// $Id: index.php,v 1.94 2007/12/26...
```

You will now be able to see any errors that are occurring directly on the screen. Memory problems

may still not be displayed, but it's the first step in a process of elimination.

If you are using a multi-site setup and only want errors to appear for one site, then check the name of the host first as in:

```
<?php
if ( $_SERVER['HTTP_HOST'] === 'some.domain.name.here' ) {
    error_reporting(E_ALL);
    ini_set('display_errors', TRUE);
    ini_set('display_startup_errors', TRUE);
}
?>
```

If the problem occurs while running update.php open update.php in a text editor and uncomment the following line:

```
ini_set('display_errors', FALSE);
```

## Log Files

Your log files can be accessed a few different places. This will vary depending on your host, but it's good to know what and where they are.

To access the files directly on the server, on some unix shells (you may need to alter this to suit your environment), you can type the following command:

```
tail /var/log/apache2/error.log
```

To check that you are looking at the right file, you may wish to type the following commands to find where the log files are.

```
grep 'ErrorLog' /etc/apache2/*
grep 'ErrorLog' /etc/apache2/*/*
```

Otherwise, if you are still able to access your admin pages through your site, which you often can during a WSOD, check the watchdog log for errors. For example you may see the 'headers already sent' error, which relates to the whitespace error (explained in the next section).

The path to your watchdog log, should you lose your admin menu is:

- (Drupal 4.7) <http://www.example.com/admin/logs/watchdog>
- (Drupal 5) <http://www.example.com/admin/logs/watchdog>
- (Drupal 6) <http://www.example.com/admin/reports/dblog>

Your results will vary in different hosting environments, but this is a good starting point.

## Whitespace at the End of a PHP File

The most common code error that causes a WSOD is having additional whitespace at the end of a PHP file. To avoid this issue, it is a [Drupal coding standard](#) to not include the closing ?> on a PHP file.

You may also have the 'Include Unicode Signature (BOM)' option turned on on your editor, which should be turned off.

## PHP4 Syntax Errors and Incompatibility

Some versions of PHP4 gag on some function declaration syntax. Here are examples of syntax that fails:

```
function media_mover_api_media_mover($op, $action = null, $configuration = null, &$file =
array(), $running_config = null ) { ...
```

```
function media_mover_api_nodeapi(&$node, $op, $a3 = NULL, $a4 = NULL) { ...
```

This appears to be a bug in PHP4 parser that makes it not be able to handle either too many "=" clauses in a function declaration or handle "&\$..." So in at least some cases, this is a de facto PHP4 incompatibility.

Solution: Upgrade to PHP5.

## PHP Versions

If you were previously running Drupal on a server with multiple versions of PHP you may have had special code in your .htaccess file telling Drupal which version to use. For instance, you may have added `AddType x-mapp-php5 .php` to your .htaccess file if your hosting provider required it to ensure PHP5 was used rather than PHP4.

If that is the case, remove this line in your .htaccess file.

## Invisible Errors for Developers

If you are developing a module, you may first want to test loading your file(s) to make sure that there aren't any obvious PHP syntax errors. This happens because the `include_once()` or `require_once()` function calls simply do not always report the errors *[if someone knows why and has a fix for that one, please add the info here!]*. The following is the command you want to run on your system (it requires PHP console, or CLI, to be installed):

```
<?php
php <filename>
?>
```

This command will ask PHP to parse your file. If you forgot a semi-colon ";", or a closing bracket "}" it will give you the error immediately. Fix it, and try again on your server.

Better yet, you may want to write an automated test to check for such errors. More info about Simpletest [here](#).

## Implement Hook Twice

You can also get a blank screen if you have by mistake implemented the same hook more than once. For example, accidentally implementing `hook_help` twice.

## Output Buffering

Some modules need output buffering turned on.

To do this, try adding these lines to your .htaccess file (normally located in your root directory):

```
php_value output_buffering On
php_value output_handler mb_output_handler
```

## Zend Compatibility Mode

If you get the WSOD while setting up a new server, you may have a problem with zend compatibility being on. If you check the error reporting you may see an error with "Trying to clone an uncloneable

object of class mysqli." This is caused by the zend compatibility mode being On in the php.ini file.

To fix this, set zend compatibility to off by editing the applicable line:

```
; Enable compatibility mode with Zend Engine 1 (PHP 4.x)
zend.zel_compatibility_mode = Off
```

[More detail about zend compatibility can be found here.](#)

## Clearing the Cache Table

Depending on the problem, clearing the cache table (via phpmyadmin for example) can resolve a WSOD.

## Ionic Rewriter: WIMP

If you are using Drupal on a WIMP stack and getting the WSOD on the <http://www.example.com/install.php> page it may be that you cannot have the Ionic Rewriter ISAPI module installed in IIS during the Drupal install.

The solution in this case is to add it in after install is completed.

## PHP Memory Limits

Another common reason for the WSOD is issues with memory limit. Traditionally, this has most often been a problem showing up (or rather, not showing up) in the modules admin screen, by it giving a WSOD. This issue has pretty much been solved for that page; however, there are still instances that will occur in other modules (usually showing up during admin actions like bulk updates) where PHP memory can be exhausted.

[Try the solutions here](#) first if this issue came up when you tried to go to the module page.

You may also want to try running the update.php script. If you do not know how to run update.php or your user does not have the permissions to run it (and you do not have the user1 login), [there is more info about update.php here](#).

Next, you'll want to confirm that the change has had an effect with a [phpinfo\(\) page](#). If you are hosting the site and it didn't work, check that you were modifying the correct php.ini file (it's named in the phpinfo). If your site is hosted by someone else and you failed to increase the memory limit, then your host has probably locked it down (for good reason) and you'll have to negotiate with them. There may be a few work-arounds to try, like [creating a custom php.ini](#), but it will vary from host to host.

## Module and Theme Related Errors

If the error is not originating from a memory limit, or any of the above errors, the error may be coming from bad code in a module or your site's theme.

## Non-recommended Module Versions

If you are working with a module that is not in a recommended release version, you may have success by upgrading it to a recommended version or disabling/removing the module. To disable the module, simply go to the module admin page (*Administer > Site Building > Modules*) and uncheck the checkbox next to the module, then click "Save Configuration."

You can tell whether a module is recommended by looking on the module page, the version will have a green background with a green checkmark at the right, and say eg. "Recommended for 6.x." Particularly if the module is a development version, eg. "6.x-1.x-dev" it may not be recommended, and will then have a red background with an "x" at the right.

## Name Clashes

Another possible cause for a blank page is a name clash, i.e. a module and a theme are using the same name. For example, if module "foo" implements `hook_block()` with `foo_block()` and there is also a theme "foo", then the theme engine will invoke `foo_block()` as the theme function to render a block. While `foo_block()` might not trigger a WSOD, `foo_page()` will.

No error messages are produced, because this is a wanted behavior of Drupal's theme system.

The solution in most cases is that if either the module or the theme (or both) are custom (created by yourself), rename it.

## Character Encoding on template.php

If the error is as follows, particularly if the output started at line 1, it may be that the character encoding on your `template.php` file is not set correctly.

```
Cannot modify header information - headers already sent by (output started at
.../sites/all/themes/THEME_NAME/template.php:1) in .../includes/common.inc on line 314.
```

This is most likely if you are using an editor such as Dreamweaver, in which case you should either set the encoding to `utf8_unicode_ci` or use a plain text editor to edit the template code.

## Disabling Modules

### Via the Module Administration Page in the UI

Disabling all the modules, then enabling them one by one can help narrow down a culprit module. To disable the module, simply go to the module admin page (*Administer > Site Building > Modules*) and uncheck the checkbox next to the module, then click "Save Configuration."

### Via the Database

If your WSOD is caused by a specific module (e.g. you enabled a module, then got the white screen) and you cannot access the module admin page, it's usually effective to disable the module in the system table of the Drupal database by setting its status to 0 and then clear the cache table.

## WSODs Due to Specific Modules

If you are using any of the following modules, you may want to look at these specific issues that have the potential to cause the WSOD.

### Node Access

If you have just enabled or disabled [node access](#) module and get the WSOD when attempting to update, you may need to rebuild node permissions. You can do this one of two ways.

First, you can rebuild permissions via the "Rebuild Permissions" button under "Node Access Status"

on the Post Settings page at <http://www.example.com/admin/content/node-settings>.

Alternately, you can use a script to update the db node permissions table:

```
<?php
require_once './includes/bootstrap.inc';
drupal_bootstrap(DRUPAL_BOOTSTRAP_FULL);
user_authenticate('admin', 'admin');
$actual=db_result(db_query("SELECT nid FROM {node} WHERE nid > %d ORDER BY
nid", $_GET['node']));
if ($actual>0) {
    $sencer=node_load($actual);

    node_access_acquire_grants($sencer);
}
?><html>
<head>
<script type="text/javascript">
<!--
function delayer(){
    window.location = "rebuild_permissions.php?node=<?=$actual?>"
}
//-->
</script>
</head>
<body <?=( $actual>0 ? " onLoad=\"setTimeout('delayer()', 500)\" : "" ) ?>>
<?php
if ($actual>0) echo "doing... ".$actual;
else echo "Done";
?>
</body>
</html>
```

## Gallery 2 (Embedded)

If you are using [embedded Gallery 2](#), you have to edit gallerydirectory/config.php to point to the new database server. If you do not do this, you may get the WSOD without any errors (after enabling errors as above).

## PEAR Wiki Filter

If you have moved your module directory, the PEAR Wiki filter will not be able to find the PEAR modules (assuming you installed them underneath the module directory.)

Check your input format in the admin pages and make sure that the "Path of PEAR packages" is correct.

## WSOD on Multisite Installs

If this happens during installation or when starting a multi-site configuration *where you are sharing databases or database tables*, the reason may also be in the settings.php file in the sites/your-site/ directory. This happens if you have already edited the \$db\_url variable in settings.php. The installer will no longer be automatically invoked, because Drupal assumes installation has already been completed.

If you get the following error messages for example (Drupal 6), the above may be the reason:

```
PHP Warning: Table 'testing.access' doesn't exist
query: SELECT 1 FROM access WHERE type = 'host' AND LOWER('127.0.0.1') LIKE LOWER(mask) AND
status = 0 LIMIT 0, 1 in C:\My_webdev\htdocs\drupal\includes\database.mysql.inc on line
128
PHP Warning: Table 'testing.users' doesn't exist
query: SELECT u.*, s.* FROM users u INNER JOIN sessions s ON u.uid = s.uid WHERE s.sid =
'da5qecc9rsf1dinhb4ko1e5dm0' in C:\My_webdev\htdocs\drupal\includes\database.mysql.inc
on line 128
```

```

PHP Warning: Table 'testing.cache' doesn't exist
query: SELECT data, created, headers, expire, serialized FROM cache WHERE cid = 'variables'
in C:\\My_webdev\\htdocs\\drupal\\includes\\database.mysql.inc on line 128
PHP Warning: Table 'testing.variable' doesn't exist
query: SELECT * FROM variable in C:\\My_webdev\\htdocs\\drupal\\includes\\database.mysql.inc
on line 128
PHP Notice: Undefined variable: variables in
C:\\My_webdev\\htdocs\\drupal\\includes\\bootstrap.inc on line 427
PHP Warning: Table 'testing.cache' doesn't exist
query: UPDATE cache SET data = '', created = 1209809924, expire = 0, headers = '',
serialized = 0 WHERE cid = 'variables' in
C:\\My_webdev\\htdocs\\drupal\\includes\\database.mysql.inc on line 128
PHP Notice: Undefined variable: variables in
C:\\My_webdev\\htdocs\\drupal\\includes\\bootstrap.inc on line 434
PHP Warning: Table 'testing.system' doesn't exist
query: SELECT name, filename, throttle FROM system WHERE type = 'module' AND status = 1 AND
bootstrap = 1 ORDER BY weight ASC, filename ASC in
C:\\My_webdev\\htdocs\\drupal\\includes\\database.mysql.inc on line 128
PHP Warning: Table 'testing.url_alias' doesn't exist
query: SELECT COUNT(pid) FROM url_alias in
C:\\My_webdev\\htdocs\\drupal\\includes\\database.mysql.inc on line 128

```

## WSOD on Cloned or Duplicate Sites

There are a few possible culprits for a WSOD on cloned sites.

### Include Paths

If you have created a new Drupal site by copying the files from an install and then importing the db over, and then edited the settings.php file with your new db info, you may still end up with the WSOD.

The [pathauto module](#) has caused problems in this scenario, where it throws fatal errors because of missing includes. In this case, adding this line to the .htaccess file solves the problem instantaneously (the right base path for your own server may be obtained through info.php or by logging into your control panel):

```
php_value include_path './var/www/vhosts/example.com/httpdocs'
```

### Missing Database Tables

If you are moving a large site and using phpmyadmin to import the DB, it may have either hanged and timed out or caused some tables not be loaded into the db. to begin, be sure to double check your db to see if all the tables are there.

If moving a large site with a huge db then better import the db via a console (command line), as phpmyadmin is not always able to handle large db files.

## Relic index.html Files, and Server Setup

Sometimes a server will be setup with a blank index.html file in the root directory, and/or Apache may not be configured to search for index.php files with the DirectoryIndex directive. It can look like a WSOD, but really is just an empty page.

To check if this is the case, navigate to <http://www.example.com/index.php> or look for an index.html file in your root directory. If there is an index.html file, make a copy of it outside your root folder, and then delete it and see if it resolves the issue.

## Infinite Loops in your Code

If your site gets a WSOD after a CPanel system update, with notices like this in the error log:

```
[notice] child pid ##### exit signal Illegal instruction (4)
```

You may have an infinite loop. Apache has become smart enough to detect this and terminate the script without waiting for a timeout.

You can try disabling modules in the system table until you find the offending module, or put a call to watchdog at the start of every function call in the module, so you can trace the call chain by looking at the watchdog table. If you find a repeating call pattern, look through the code for a loop cycling, eg. node\_load through hook\_nodeapi and/or other functions, and repeating node\_load again.

## PHP Timeout

The default PHP timeout defined in php.ini is 30 seconds. This is too short for some activities like listing/enabling modules.

[Here are details on how to increase the maximum execution time.](#)

## Inappropriate naming of functions

Defining a custom function in your module code which acts as a Drupal hook function may result in WSOD. Check that the functions, that you define in your module do not use \_menu, \_auth or other extensions if they are not meant to implement that specific hook.

The following code generetase WSOD and Apache segfault in Drupal 5.

```
function mymodule_menu() {
  array(
    'path' => 'auth'
    'callback' => 'mymodule_auth'
  );
}

function mymodule_auth() {
  drupal_execute('user_login', array('name' => $_POST['name'], 'pass' => $_POST['pass']));
}
```

## Any other solutions available?

If above information was not helpful enough, you may follow the next article which is dealing with WSOD diagnostic solutions:

<http://drupal.org/node/482956> - Silent WSODs (White Screen of Death) - fixing step by step

## What is the direct link to my main admin page

If you haven't enabled the clean URLs option for your site, adding a /?q=admin after your website address will direct you to your admin area. For example: example.com/?q=admin or example.com/drupalsubdir/?q=admin

If you have clean URLs enabled, you can just add /admin after your site address. For example: example.com/admin or example.com/drupalsubdir/admin

# Where is the taxonomy choice when adding content?

Taxonomy, or 'Categories' in 5.x and on, requires a vocabulary to the content type.

Just create a taxonomy (category) and fill it with a perfect array of terms and make sure your content type is assigned the new taxonomy (category) type.

Go to (*administer > categories > edit vocabulary*) and look under the "Types" heading. There you can assign your taxonomy to any content types you want. Save your configuration here, and try to submit a post. Your taxonomy should now be available in the submission post.

## Test PHP code before putting it in blocks

Drupal provides a great deal of power and flexibility when using [PHP code in blocks](#). Unfortunately, a stray character or a missing semicolon may break code. Drupal then attempts to evaluate this broken code on any requested page, the PHP interpreter chokes on it and aborts, and a whole site can become inaccessible.

Fortunately, there is a very simple solution. Instead of writing and testing code inside the **Administer >> Site building >> Blocks** page, go to **Create content** and create a new story or page node. Use PHP input format, write the code, and the Preview to debug your code. Once the code is working, copy and paste the code into the block.

Alternate methods include:

- Create a block and set use the visibility section to restrict it to a single page until its functionality is confirmed

As of Drupal 6 the PHP input filter is a separate module. It is not available unless the optional **PHP Filter** module is enabled. It is considered bad coding style to have php code stored in the database for maintain and security reasons so after testing with the php filter the php code should go into a template file. (For example block-block-1.tpl.php)

*Note: see <http://drupal.org/node/166556> for suggestions on recovering from an inaccessible site caused by a block with bad PHP code in it.*

## Error Messages

The following pages provide information about error messages that you might encounter. You can find error messages listed in your log files. You can find recent log entries in Administer/Reports/Recent log entries.

## Client does not support authentication protocol requested by server...

You are installing a new install of Drupal and during setup receive the error

```
Client does not support authentication protocol requested by server; consider upgrading
```

MySQL client.

You are probably using MySQL 4.1 or higher and need to [reset the password to pre 4.1 style](#).

There's a quick way to fix this if you use recent versions of PhpMyAdmin and have the rights to edit MYSQL user information:

1. Select the 'privileges' option on the main PhpMyAdmin page
2. Edit the user you're using to connect to the Drupal database
3. Scroll down to where it says 'Change password'
4. Enter and re-type the password for that user
5. **IMPORTANT** Where it says 'Password Hashing', choose the option that says 'MySQL 4.0 compatible'.
6. Save your selection

You should now be able to connect to your Drupal database using that user and password with no more errors.

## "500 Internal Server Error" error

When using Drupal your browser returns the following error

### **Internal Server Error**

**The server encountered an internal error or misconfiguration and was unable to complete your request.**

**Please contact the server administrator, [webmaster@\\*\\*\\*\\*\\*.com](mailto:webmaster@*****.com) and inform them of the time the error occurred, and anything you might have done that may have caused the error.**

**More information about this error may be available in the server error log.**

**Additionally, a 500 Internal Server Error error was encountered while trying to use an ErrorDocument to handle the request.**

An "Internal Server Error" is an error on the web server that you're trying to access. That server is misconfigured in some way that is preventing it from responding properly to what you're asking for.

An "Internal Server Error" often occurs on:

- Misconfigured Drupal core file or folder read and write permissions.
- Misconfigured web server file or folder read and write permissions.
- Corrupted file(s).
- After enabling a module.

To fix this click on one of the bellow possible causes.

## Problems related to WSOD (White Screen of Death)

Try the suggestions at the WSOD (White Screen of Death) page. Some of those problems will produce an "internal server error" message rather than a blank screen. (This may depend on your configuration).

<http://drupal.org/node/158043>

This is true especially if you get this message after attempting to enable a module, and find that the site seems to work okay, but the admin modules page no longer works.

## Misconfigured folder or file permissions

Check your Drupal file or folder permissions. Permissions means read and write accesses. They are maybe misconfigured. This is one of the most likely cause of this error.

For Drupal 6 read more at <http://drupal.org/getting-started/6/install/set-permissions>

Read the `INSTALL.txt` file that comes with Drupal. There is a section in there about permissions.

## Corrupted Drupal file(s)

If none of the previous solutions worked you could try the following. **But this should be try as a last resort.**

### Steps

1. Backup your database and files so you can revert if the following is not working. Read more about backups at <http://drupal.org/node/22281>
2. Delete all Drupal core files. But do not delete other appropriate files though. For example do NOT delete the `settings.php` file located under `sites/default/settings.php`. And do NOT delete the `all` folder and its sub-folders located under `/sites/all`
3. Download a fresh `drupal-X.XX.tar.gz` at <http://drupal.org/node/3060/release> Make sure you download the exact same version that was installed before the error. For example if you were using Drupal version 6.12 then you should download Drupal version 6.12 tarball. Not a more recent version. Otherwise it might not work. To know which version of Drupal 6 was installed, open the `CHANGELOG.txt` file. This file is located in your Drupal root folder. The first section of this file indicated your Drupal version.
4. Install the fresh Drupal core. To do so read the `README.txt` file located inside the Drupal tarball.
5. Go to <http://YOUR-DOMAIN-NAME-HERE.com/update.php>
6. Follow instructions on screen. Start by clicking on `Continue` button.
7. If not working revert to your backup files. The files from above step 1.

## Something wrong with server side (PHP server, CGI server, other servers or control panel)

If you do NOT have full editing access to the web server. Such as share hosting you must contact your web hosting provider to open a ticket asking them to investigate. To speed up the investigation provide them with a screen shot or a copy of the error. Plus information about the last things you did just before noticing the error. There's maybe something wrong with the PHP server, CGI server or other servers. If this is the case there's absolutely nothing you can do to resolve this problem because you don't have access to those servers. The web hosting provider would be responsible for investigating and fixing.

If you do have full editing access to the web server. In other words 'root access'. Such as dedicated

or VPS hosting the most likely cause of this issue is that one or multiple server files got somehow corrupted or misconfigured.

### This often occurs on

- **An error in a cgi script** that caused it to fail or output an error message before it started producing valid HTML. It happens to me all the time if I have a syntax error in one of my Perl cgi scripts.
- **A permissions issue attempting to access a cgi script.** Depending on how your web server is configured, quite often it's not enough for the script to have "execute" permission, but it must also be owned by the correct user, and belong to the correct group. Correct settings varies widely from server to server (and even site to site on the same server). One way of fixing this is looking at the attributes of a script that's working, and copy that.

Read more about web server & webhosting issues <http://drupal.org/server-permissions>

The format of "500 Internal Server Error" is common for Apache web servers. That means you can also look for common error logs. Once again, the location of these logs can vary a great deal based on the specific web server configuration.

- `access_log` is the log of successful accesses. On a shared or virtual hosting server there may be many of these, one per web site, often with site-specific names, or in site-specific locations on the server.
- `error_log` is the log of errors. On some hosts, there is an error log per site. On others, though, even though each site may have its own `access_log`, there may only be a single `error_log` for the system.
- `suexec_log` is the log most people forget about. This log deals with the permissions used to execute cgi and other scripts. When a cgi fails to execute because its ownership is wrong, the generic "Internal Server Error" shows in the error log, but a more specific error detailing the permissions involved shows in this log.

## "An image thumbnail was not able to be created." error

When trying to upload a picture to Drupal 6.x the following error is return

**An image thumbnail was not able to be created.**

The most likely cause of this error is the security settings on the `files` folder as well as its sub-folders. Drupal is trying to write a file on the server but is denied write access.

### Here are the steps to fix this:

Set the security permissions on the following folders to allow WRITE. In other words set the CHMOD to 777 on the following folders. To do so you could use a FTP tool.

```
/sites/default/files/
/files/
/sites/your-domain-name-here.com/files/
```

Note 1: If you can't find some of those folders that's normal. It depends on your Drupal setup. But you should at least have one.

Note 2: You must allow WRITE to all sub-folders inside the `files` folder.

Note 3: For security reason some sub-folders inside the `files` folder should NOT have write access.

Which folder? Usually each module comes with a README.txt file. Read each README file the answer is there. For example the module 'backup\_migrate' as some folders that should not have write permission.

That's it. You have successfully set the security permission to WRITE on the appropriate folders.

Test it by uploading a picture to your site.

## "Got a packet bigger than 'max\_allowed\_packet' " error

When using Drupal your browser returns the following error

### Got a packet bigger than 'max\_allowed\_packet'

The most likely cause of this error is your MySQL server. If you're new to Drupal think of MySQL server as a database.

**To fix this, select the option below that better matches your needs:** The beginners option or the advanced option.

## Beginners option | "Got a packet bigger than 'max\_allowed\_packet' " error

**The following is the beginners option.** This should work on most share hosting.

When using Drupal your browser returns the following error

Got a packet bigger than 'max\_allowed\_packet'

The most likely cause of this error is your MySQL server. If you're new to Drupal think of MySQL server as a database.

### Steps to fix this

Install the 'drupal\_tweaks' module [http://drupal.org/project/drupal\\_tweaks](http://drupal.org/project/drupal_tweaks)

Using this module increase the 'max\_allowed\_packet' memory limit. To do that go to [http://your-domain-name-here.com/admin/drupal\\_tweaks](http://your-domain-name-here.com/admin/drupal_tweaks)

Follow instructions on this page. If needed read more in the module README.txt file.

Note: If it doesn't work redo above steps with more memory. For example: 2M, 4M, 8M, ... 128M and so on. The 'M' stands for MEGABYTE make sure you have enough memory available before increasing to much. If unsure ask your hosting provider.

## Advanced option | "Got a packet bigger than 'max\_allowed\_packet' " error

**The following is the advanced option.** You need full root access to your server. This will not work on most share hosting.

When using Drupal your browser returns the following error

```
Got a packet bigger than 'max_allowed_packet'
```

The most likely cause of this error is your MySQL server. If your new to Drupal think of MySQL server as a database.

**Here are the steps to fix this on Ubuntu 8.04.x LTS desktop edition:**

On your Ubuntu desktop click on PLACES menu.

Select COMPUTER.

Open the FILESYSTEM folder.

Open the ETC folder.

Open the MYSQL folder.

Using TEXT EDITOR edit MY.CNF file.

Note: This is optional but you should backup your MY.CNF file before editing it. So if it doesn't work you can revert your change.

Inside the MY.CNF file go under the section `[mysqld]`

Search down this section for the following line

```
max_allowed_packet = 16M
```

Change it for

```
max_allowed_packet = 32M
```

Note: The MY.CNF file contain 2 or more `max_allowed_packet = 16M` line. Make sure you're editing the one under the section `[mysqld]`. Not the one under the section `[mysqldump]`.

Save the MY.CNF file.

Close the MY.CNF file.

Navigate to the following Ubuntu menu APPLICATIONS > ACCESSORIES > TERMINAL.

Type in the following command in TERMINAL to stop your MySQL server.

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

Type in the following command in TERMINAL to restart your MySQL server.

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

If successful TERMINAL will return a message with a `[ OK ]` on the right side.

That's it. You have successfully increased your `max_allowed_packet`. Retry using your Drupal.

Note for beginners: If it doesn't work redo above steps with more than 32M. Such as 64M, 128M and

so on. The 'M' stands for MEGABYTE make sure you have enough memory available before increasing to much.

## "Got error 28 from storage engine query" error

When trying to log-in Drupal the following error is return

```
user warning: Got error 28 from storage engine query: SELECT DISTINCT b.*
FROM blocks b LEFT JOIN blocks_roles r ON b.module = r.module AND b.delta
= r.delta WHERE b.theme = 'garland' AND b.status = 1 AND (r.rid IN (2,3) OR
r.rid IS NULL) ORDER BY b.region, b.weight, b.module in /home/your-
username-here/public_html/modules/block/block.module on line 433.
```

Error 28 occurs when there is not enough disk space left for MySQL install to run properly. Then you start receiving error 28 messages.

"Error 28" often occurs on

- Filesystem full. Hard drive full.
- Host oversell their shared hosting.
- Host misconfigured their shared MySQL install.

**To fix this you must increase the space of your filesystem or make room by deleting files. In other words your hard drive is full and MySQL needs more space. If you don't have access you must contact your host provider to report the issue. To speed up their investigation provide them with a copy or a screenshot of the error message.**

## "HTTP request status Fails" error

Your `status` report page located at `admin/reports/status` returns the following error

### HTTP request status Fails

**Your system or network configuration does not allow Drupal to access web pages, resulting in reduced functionality. This could be due to your webserver configuration or PHP settings, and should be resolved in order to download information about available updates, fetch aggregator feeds, sign in via OpenID, or use other network-dependent services.**

Error screenshot. Under section `HTTP request status`



## The most likely cause of this error is

- You did nothing wrong. This is a well known issue in Drupal since around version 6.9. The community knows about it and is working on it. If you want to help fix this go to <http://drupal.org/node/245990>
- A files format (character set) has been changed. For example when editing then saving your file your text editor changed the file format from UTF-8 Unicode to Latin1. Some text editor will change the file format without asking. Sometime inserting line brake or changing some characters.
- A file got corrupted during transfer. For example while uploading a file via FTP.
- A contributed module.

## To fix this here are a few things you could try

- Deactivate all contributed modules. Test if error is still there. If error is no longer there isolate the error by reactivating only one contributed module at a time. Test each contributed module if error is still there. When error shows you will know which contributed module causes the error. Replace appropriate contributed module file(s) with fresh file(s).
- Install the module 'http\_request\_fail\_reset' [http://drupal.org/project/http\\_request\\_fail\\_reset](http://drupal.org/project/http_request_fail_reset)
- Add a line of code at the bottom of your settings.php file <http://drupal.org/node/245990#comment-1312270>
- Add the address of your site to the list of hostname aliases for 127.0.0.1 in /etc/hosts. [Read more](#)

## For coders

- Explanation of the error: <http://drupal.org/node/245990#comment-909263>
- Error source: <http://drupal.org/node/245990>

# "LOCK TABLES sequences WRITE" error

This is in reference to the following kind of error:

```
user error: Access denied for user: 'user@host' to database 'database'
query: LOCK TABLES sequences WRITE in [path]/database.mysql.inc on line xx
```

When you installed Drupal, you created a database and a **database user**. This error is caused by that **database user** not having a certain privilege over your database.

When you come across this problem, first, double check that you have granted all privileges to your user. You can try to grant the LOCK tables privilege by executing the following query:

```
GRANT LOCK TABLES ON databasename.* TO username@localhost IDENTIFIED BY 'password';
```

If the GRANT command does not seem to have any effect on MySQL, you may need to follow it by "FLUSH PRIVILEGES." If you still get this problem, and you are using shared hosting, it is very likely that your host does not usually let users have the kind of permission required. For reference, you may want to read these [forum posts](#). You need to contact your host (e-mail seems better than live support for this) and ask (if you are using mySQL 4) for **global LOCK TABLES privilege** for your user in that database.

Although there are fixes in the forums, these are not recommended. If your host denies your request, they may not be an appropriate place for your Drupal site.

# "Method POST is not allowed for the URL /index.htm" error (Error 405)

Your Drupal directory contains both an index.html and index.php file. Remove the index.html file or configure your web server to look for index.php first before index.html. The same goes for basically any html file in your Drupal directory. If you have for example "node.html" in your root, you get the error message for every form that is submitted to node/\*.

## "Page Not Found" Errors on every page except homepage.

### Narrow the problem

The most likely cause is misconfiguration of mod\_rewrite or clean URLs on the new server. Confirm this by short-circuiting the clean URL system and requesting pages directly, like this:

<http://www.example.com/index.php?q=user>

If the login page is shown, then you can assume that Drupal's page serving mechanism is working properly, but mod\_rewrite isn't working. To determine if mod\_rewrite is enabled, create a file called phpinfo.php in the root of your site. This file should be accessible from a web browser and contain this code:

```
<?php
phpinfo();
?>
```

### Ensure that mod\_rewrite is enabled

Navigate to phpinfo.php in a browser like this: <http://www.example.com/phpinfo.php>

On that page, look for a section called "Loaded Modules" and check that mod\_rewrite is included. If not, mod\_rewrite is not being loaded by apache and will need to be enabled. The process will be different depending on the server platform and apache build. Keep the phpinfo.php file for now - it will be helpful later.

### Ensure that Drupal's .htaccess file is being used

Once it's been confirmed that mod\_rewrite is enabled, double check that drupal's .htaccess file is in the site's root, and that it is working. Confirm that .htaccess is enabled by temporarily replacing Drupal's .htaccess file. Make a backup of the existing .htaccess and create one like this...

```
DirectoryIndex phpinfo.php
```

Now try the site URL without specifying a page or file in a web browser: <http://www.example.com/>. If you see PHP's phpinfo dump, the .htaccess file worked and you can restore the old one. If not, the server may not be allowing .htaccess overrides for your site.

## Double check your rewrite\_base setting

If phpinfo was displayed, something may be misconfigured in the existing Drupal .htaccess file. Check the documentation for "RewriteBase" and "RewriteEngine" settings.

[http://httpd.apache.org/docs/2.0/mod/mod\\_rewrite.html](http://httpd.apache.org/docs/2.0/mod/mod_rewrite.html)

## Enable necessary .htaccess overrides

If phpinfo was not displayed, locate your apache configuration file and enable the necessary .htaccess overrides. This location will vary by server platform, so check the appropriate documentation. In this example (Ubuntu server), the default apache vhost configuration was located at:

```
/etc/apache2/sites-enabled/000-default
```

To configure mod\_rewrite from .htaccess, the apache config needs to allow at minimum "FileInfo" overrides. In order to make the simple DirectoryIndex test work, apache needs to allow "Index" overrides. Locate the section in the apache configuration file or included vhost configuration file that looks like this:

```
<Directory /filesystem/path/to/your/sites/root>
</Directory>
```

Within that block, either add or edit the apache AllowOverride directive. This can create security risks, so see the official reference here:

<http://httpd.apache.org/docs/1.3/mod/core.html#allowoverride>

To allow all overrides, it will look like this:

```
<Directory /filesystem/path/to/your/sites/root>
  AllowOverride All
</Directory>
```

To only allow selected overrides, specify them individually like this:

```
<Directory /filesystem/path/to/your/sites/root>
  AllowOverride Indexes Options FileInfo
</Directory>
```

Now reload the apache configuration. This is also different by platform, so consult the appropriate documentation. On Ubuntu server the command is:

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

Navigate a browser to <http://www.example.com/> and PHP's phpinfo should be displayed. If so, replace it with Drupal's .htaccess file and your site should now be working as expected.

If clean URLs are still not working, go back to the section titled [Double check your rewrite\\_base setting](#).

# "Page Not Found" error on ALL pages of Drupal 6 website

If you're experiencing the "Page Not Found" error on every single page of your Drupal 6 site, it is possible that your menu router data got corrupted or deleted. This is most likely to happen when

there is an error that occurs during the process of enabling a module. See this bug report: <http://drupal.org/node/238760> The problem should be less likely to occur after the Drupal 6.3 update is out.

To fix this, your first step should be to try to figure out what might have caused it and to undo those last few modifications that you made to the site. Typically, this would involve reverting to old versions of modules, or even disabling all non-core modules. Since the site is not accessible, you may need to disable the modules by SQL. Set the status field of the suspect contributed module(s) to 0 in the {system} table. There are plenty of threads in the forum that discuss how to do this, if you are not familiar.

Once you've done that, you just need to rebuild your menu system to give you access again. The easiest way to do this is to run update.php. Even if there are no updates, running through the process will also rebuild the menu router.

If for some reason running update.php doesn't work, you may need to temporarily alter some code as outlined below. This should be a last resort, and is not normally needed.

To rebuild your menu, find the **menu\_execute\_active\_handler()** function inside **/includes/menu.inc** (approx at line 334) and force it to call menu\_rebuild().

I.e. temporarily change this:

```
if (variable_get('menu_rebuild_needed', FALSE)) {
  menu_rebuild();
}
```

to this:

```
//if (variable_get('menu_rebuild_needed', FALSE)) {
  menu_rebuild();
//}
```

Now if you try to access your site again, it will rebuild the menus and you should be back to normal.

**IMPORTANT:** Don't forget to undo your changes to menu.inc, otherwise your site will suffer a huge performance hit!

## "Page Not Found" error when trying to access a subdirectory

When you create a subdirectory inside the main Drupal directory that requires authentication (with a .htaccess file), you may receive a "Page Not Found" Drupal page instead of the page you expect to receive from the subdirectory.

This is caused by the way Drupal handles unauthorised access to items. There is an easy fix to this problem. The following line should be added to the end of the .htaccess file in the subdirectory you are trying to access (NOT THE DRUPAL .HTACCESS FILE).

```
ErrorDocument 401 "Unauthorized"
```

Doing this rewrites the default 401 error so that it is not intercepted by Drupal.

## 406 Error when XMLRPC is used

Some hosting providers (at least one!) may disable xmlrpc.php scripts to plug security holes in older versions of the XMLRPC libraries. If you've installed Drupal and enabled the BlogAPI module or the Drupal login module on such a provider, you'll get "406 errors" when you try to use the features.

One possible solution (aside from convincing your provider to turn the relatively ineffective security measure off) is to rename the xmlrpc.php file in drupal's main directory. While this is bad and will make upgrading a bit of a hassle, it can get you up and running.

You'll have to alter blogapi.module to point to the newly renamed file, as well as any other modules that point to xmlrpc.php. In the case of blogapi.module, the reference can be found on line 560:

```
560: $xmlrpc = $base_url .'/' . 'xmlrpc.php';
```

## Compilation failed: this version of PCRE is not compiled with PCRE\_UTF8 support

This error occurs on release 4.6.1 onwards.

*Note: This issue is fixed by default on all VPS3 and new VPS2 servers, resolve-able on VPS1 and old VPS2 (with some updating).*

This fix works for Verio VPS2 Servers which are installed with PCRE 6.4 by default (at this time). I see no reason why it should not work for all unix servers (perhaps with directory changes etc).

The fix is to uninstall pcre and install the correct version as follows:

```
login as root
Type:
# pkg_info
which will show you the installed packages.
type:
# pkg_delete pcre-6.4 (assuming you see pcre-6.4 as the installed version, change as
required)
change to the directory where your ports collection is.
on FreeBSD it is:
/ports/devel/pcre-utf8
so type into your shell prompt:
# cd /ports/devel/pcre-utf8
# make
# make install
# make clean
```

then restart the apache server...

something like this (this is a VPS2 specific script, yours may be different):

```
# restart_apache
```

## Configuring PHP to resolve the "Error sending email" message

This article is based on experience with a Linux system running an Apache web server, but the problems and solutions apply to many other Linux and non-Linux environments.

This won't solve all problems that cause the "Error sending email" message, but this describes some solutions to a couple of common causes.

**IMPORTANT NOTE:** Some web servers, including Apache, require you to reload the server for changes to php.ini to have any effect.

Internet email has two "from" addresses: the header "From" address and the "envelope-from" address. The default PHP and sendmail setups use the current username for both, but many configurations don't provide an email address for each username and system account, in which case both "from" addresses may need correcting to help resolve the problems that result in the "Error sending email" message.

## Bogus "From" header address

By default, the [PHP mail\(\) function](#) uses the current username in the message header "From" address. The solution is to change the following line in php.ini:

```
sendmail_from = "webmaster@MYDOMAIN.TLD"
```

## Bogus "envelope-from" address

Just as with the "From" address, by default **sendmail** makes the same assumption that PHP does, that the current user has an email address. PHP calls sendmail with default arguments and, by default, sendmail uses the current user name for the "envelope-from" address. This problem can also be resolved with a change to php.ini:

```
sendmail_path = "/usr/lib/sendmail -t -f webmaster@MYDOMAIN.TLD"
```

# Duplicate entry error during node add

Occurred running Drupal 6.4 using a database migrated from Drupal 5

The error:

```
# user warning: Duplicate entry '0' for key 2 query: INSERT INTO node (vid, type, language, title, uid, status, created, changed, comment, promote, moderate, sticky, tnid, translate) VALUES (0, 'page', 'es', 'eee', 1, 1, 1222437644, 1222437644, 2, 1, 0, 0, 0, 0) in /var/www/webapps/drupal6/includes/common.inc on line 3318.
# user warning: Duplicate entry '0' for key 1 query: INSERT INTO node_comment_statistics (nid, last_comment_timestamp, last_comment_uid, comment_count) VALUES (0, 1222437644, NULL, 1, 0) in /var/www/webapps/drupal6/modules/comment/comment.module on line 607.
```

Apparently is produced by a problem during database upgrade, because auto\_increment is not set for node\_revisions table.

Check if 'vid' is PRIMARY KEY on 'node\_revisions' and also check that is set to 'auto\_increment'

### If not try this to fix it:

Obtain last node ID

```
SELECT max( nid ) FROM `node`
```

And sum 1 to the number obtained from previous query and replace YOUR\_NUMBER with it and run:

```
ALTER TABLE `node_revisions` ADD PRIMARY KEY ( `vid` )
ALTER TABLE `node_revisions` CHANGE `vid` `vid` INT(10) UNSIGNED NOT NULL AUTO_INCREMENT
ALTER TABLE `node` AUTO_INCREMENT=YOUR_NUMBER
ALTER TABLE `node_revisions` AUTO_INCREMENT=YOUR_NUMBER
```

## This Works

I have also had this issue running 6.10 (never upgraded from drupal 5), and can confirm running

```
SELECT max( nid ) FROM `node`
ALTER TABLE `node` AUTO_INCREMENT=YOUR_NUMBER
ALTER TABLE `node_revisions` AUTO_INCREMENT=YOUR_NUMBER
```

fixed the issue.

# Error 1364 upon importing database.mysql with MySQL 5.0+

## Symptom

When you use the file database/database.mysql to create a Drupal database, MySQL reports the following error:

```
error 1364 (hy000) line 803: field 'page' doesn't have a default value.
```

This error is caused by the following lines in database.mysql:

```
REPLACE blocks SET module = 'user', delta = '0', status = '1';
REPLACE blocks SET module = 'user', delta = '1', status = '1';
```

You may have additional problems when working with Drupal. Sidebars will be absent and posting content results in an error:

```
user error: Field 'revisions' doesn't have a default value [snip]
```

## Cause

MySQL 5.0 and higher have a strict mode that is currently incompatible with a number of queries in Drupal. The Windows Installer from [MySQL.com](http://MySQL.com) enables this strict mode by default.

## Workaround

There are several workarounds

1. Replace in my.ini the current sql-mode line with `sql-mode="MYSQL40"`
2. Start MySQL with the option `--sql-mode="MYSQL40"`
3. Execute the query `SET GLOBAL sql_mode='MYSQL40'`

MySQL needs to be restarted before changes in my.ini have an effect. My.ini can be found in the MySQL installation directory or the Windows directory, depending on your configuration.

Note: After switching MySQL mode you have to recreate your Drupal database or your site will not function properly.

## Another option

If you don't want (or can't) switch the mode, you may do the following :

Replace the offending lines in the database script :

```
REPLACE blocks SET module = 'user', delta = '0', status = '1', pages = '', types = '';
REPLACE blocks SET module = 'user', delta = '1', status = '1', pages = '', types = '';
```

Change the table node so that the column revision gets the default value "".

## Another option

Instead of changing my MySQL install, I change your database.mysql file. Navigate to the folder

```
webroot\documents\drupal_download\database
```

and open the database.mysql file.

Add two new default entries in table block. Originally it is:

```
CREATE TABLE blocks (
  module varchar(64) DEFAULT '' NOT NULL,
  delta varchar(32) NOT NULL default '0',
  status tinyint(2) DEFAULT '0' NOT NULL,
  weight tinyint(1) DEFAULT '0' NOT NULL,
  region tinyint(1) DEFAULT '0' NOT NULL,
  custom tinyint(2) DEFAULT '0' NOT NULL,
  throttle tinyint(1) DEFAULT '0' NOT NULL,
  visibility tinyint(1) DEFAULT '0' NOT NULL,
  pages text NOT NULL,
  types text NOT NULL
) TYPE=MyISAM;
```

Add DEFAULT "" (those are single quotations, and no space in between) after the text in both pages and types. You can use the one from module as a guide.

Now you can use either your command line or web based admin to create a new database and import the database.mysql file.

## Error on installation step 3: Warning: Table '[database].access' doesn't exist [...]

To run the install script point your browser to the base url of your website (i.e. <http://www.example.com>). You will be presented with the "Database Configuration" page.

On Step 3 of [INSTALL.txt](#) - RUN THE INSTALL SCRIPT you do not arrive at the Database Configuration page, but are greeted with the following warning(s) instead:

```
Warning: Table '[database].access' doesn't exist query: SELECT CASE WHEN status=1 THEN 0
ELSE 1 END FROM access WHERE type = 'host' AND LOWER('[ip-address]') LIKE LOWER(mask) ORDER
BY status DESC LIMIT 0, 1 in [path]/includes/database.mysql.inc on line 172
```

Followed by one or more warnings concerning headers that have already been sent:

```
Warning: session_start() [function.session-start]: Cannot send session cookie - headers
already sent by (output started at [path]/includes/database.mysql.inc:172) in
[path]/includes/bootstrap.inc on line 811
```

This happens if you have already edited the \$db\_url variable in settings.php. The installer will no longer be automatically invoked, because Drupal assumes installation has already been completed.

To remedy, run the installer manually by visiting <http://example.com/install.php>.

# Fatal error: Allowed memory size of X bytes exhausted (tried to allocate Y bytes)...

That indicates that Drupal needed more memory than PHP was allowed to give it.

Increase PHP's memory limit, either by adding:

- `memory_limit = 16M` to your `php.ini` file (recommended, if you have access)
- `ini_set('memory_limit', '16M');` in your `sites/default/settings.php` file
- `php_value memory_limit 16M` in your `.htaccess` file in the Drupal root

Slightly more [instructions on increasing PHP memory on your server](#) in the installation guide.

Some hosts allow a `PHP.ini` in the root of your site. Other hosts will not allow any modification of your capacity at all.

Depending on the amount of modules you have enabled and their 'impact' on the site you may need to increase the `memory_limit` even more (sometimes to 32 MB or more). Image processing often takes a lot of memory, as can working with any large files. Experiment with what memory value works for your needs.

Clearly, if your error was `memory size of 16777216 bytes exhausted (16M)` in the first place, then you are going to have to be bumping the limit up even higher than that. Do the binary thing and double it to 32M.

You may need to restart your server before the `php.ini` settings take effect.

Note: *Do not* just set an arbitrarily high number just to avoid this potential problem - it may limit your ability to have multiple simultaneous connections run efficiently, and simultaneous connections are important on web servers.

Note: There is no way to know how many or what combination of modules will put any one account over their memory limit. Every module uses a different amount of memory. Core requires memory by itself and requirements for core alone should be understood. see: <http://drupal.org/requirements>

## Workaround for images

If this memory exhausted error occurs during image uploads, there is a workaround: decrease the image dimensions. I got this errors for 2448x1836 pixels, and after decreased the dimensions to 1024x768 pixels, it works fine at 32M PHP memory.

## Fatal error: Call to undefined function

This is **almost always** a symptom of version problems with a contrib module ([or theme](#))

This is most often seen in form functions -undefined function: `form_*()` but could result from any missed upgrade step listed in the [module developers upgrade notes](#) which lists the deprecated functions, but most likely means *you have installed the wrong version of the module* (or that a compatible version doesn't exist yet).

You may find that turning on `legacy.module` will support a few of the older (deprecated) functions

that the troublesome module is trying to call.

## Fatal error: Call to undefined function: form\_\*() on Drupal 4.7

After installing a module on Drupal 4.7 you get the error:

```
Fatal error: Call to undefined function: form_*() in [module_path]
```

This error usually occurs on the modules settings page, but also on other module pages containing forms.

Most likely cause: the module is specific for Drupal **version 4.6**.

The way forms are created changed completely in version 4.7. This new forms API means that all former form\_\* functions such as form\_checkbox, form\_select and form\_submit, are no longer available.

Solution: You need to download a 4.7 specific version of the module in question. If the module isn't available for 4.7 you might want to try the version marked as "CVS". If that one does give the same error you should try to update it yourself. For easy forms you can use formupdater.module. An online version of this is available here:

<http://lullabot.com/formupdater>

If you succeed in converting the module you should send it to the author so that he can upload it for others to download.

You may also get this error if you use php snippets in pages or blocks that display forms. Disable such snippets before upgrading.

## Fatal error: Cannot redeclare blah\_function() in ../modules/blah.module

When using Drupal your browser returns the following error

```
Fatal error: Cannot redeclare blah_function() (previously declared in /modules/blah.module:blah) in ...blah... on line blah
```

Often occurs on:

- admin/modules
- admin/themes
- after modifying module or theme files
- after duplicating module folder
- after upgrading the module and forgot to remove old files

**To fix this, select one of the possible causes listed below.**

# Possible Cause: A function got declared inside of another function

## Possible Cause:

A function got declared inside of another function (i.e. “}” in the wrong place)

```
function listing_form_alter($form_id, &$form){
  if($form_id == 'listing_node_form'){
    //do stuff
  }

function listing_form(){
  //do stuff
} //end listing_form

} //end listing_form_alter
```

## Fix:

Move the function.

```
function listing_form_alter($form_id, &$form){
  if($form_id == 'listing_node_form'){
    //do stuff
  }
} //end listing_form_alter

function listing_form(){
  //do stuff
} //end listing_form
```

# Possible Cause: Duplicate module folder

## Possible Cause:

There is one or multiple copies of the same module folder.

## Fix:

For example in my case the error returns Fatal error: Cannot redeclare system\_requirements() (previously declared in /modules/system/system.install:12) in /modules/modules/system/system.install on line 299

1. Search for duplicate(s) module folder that's triggering the error. In my case I search for a duplicate `system` folder.
2. I found two folders with the same name `system`. The first one is located under `modules/system`. And the second one is located under `modules/modules/system`
3. To fix this delete the duplicate folder(s). In my case I delete the `system` folder located under `modules/modules/system`

If above fix isn't working try the same above steps but search for duplicate(s) inside the other `modules` folder located under `sites/all/modules`

# Possible Cause: Duplicated PHP in block or node body

## Possible Cause: Duplicated PHP in block or node body

In this case, PHP code may have been inserted into the body of a block, or is contained in a node's

"body" field. This code declares the function in question more than once, producing the conflict. For example,

```
Fatal error: Cannot redeclare getnodecount() (previously declared in
/var/www/sites/site.org/htdocs/includes/common.inc(1685) : eval()'d code:3) in
/var/www/sites/site.org/htdocs/includes/common.inc(1685) : eval()'d code on line 9
```

*Fix:*

You can search your node\_revisions table for the node IDs containing the offending code with something like:

```
SELECT nid
FROM node_revisions
WHERE body LIKE '%getnodecount%';
```

Then examine those nodes through Drupal's administrative interface, ensuring that each function is declared in only one place.

## Possible Cause: Module and theme use same name

*Possible cause:*

An installed module conflicts with an installed theme, (i.e. they both have the same name).

*Fix:*

Rename one of them.

## Possible Cause: Two functions with the same name

*Possible Cause:*

Because, EVERY file in `themes/**/*module` and `themes/**/*theme` is evaluated once, if a module is created by copying an existing module, and the module developer did not rename every function inside the module directory structure, you could have two functions (albeit in separate locations) with the same name.

The first function will be evaluated, and the second will throw an error.

*Fix:*

1. Do a global search for the function name that's triggering the error.
2. Rename or remove the duplicate function

Note: All files of the form `*.module` will be evaluated. Utilize an alternate extension like `*.module.bak` to prevent drupal from scanning the old file.

## Possible Cause: Two modules bundle the same third party library

*Possible Cause:*

Duplicate includes of third party libraries (i.e. when two modules bundle the same third party library)

could also trigger this.

Fix:

Do the find-in-files to identify the problem, and inquire with the respective module projects.

## Fatal error: Unsupported operand types in common.inc

This error happens because the parameters to `url()` changed in Drupal 6. The most common problem is an enabled module hasn't fully converted it's code from Drupal 5 and is causing the error. To find which module is the problem, change the following code to the function `url()` in `./includes/common.inc` (down around line 1300 or so):

```
<?php
function url($path = NULL, $options = array()) {
  // Merge in defaults.
  $options += array(
    'fragment' => '',
    'query' => '',
    'absolute' => FALSE,
    'alias' => FALSE,
    'prefix' => ''
  );
?>
```

To the following:

```
<?php
function url($path = NULL, $options = array()) {
  if (!is_array($options)) {
    echo "<pre>";
    $backtrace = debug_backtrace();
    var_export($backtrace[0]);
    die();
  }
  // Merge in defaults.
  $options += array(
    'fragment' => '',
    'query' => '',
    'absolute' => FALSE,
    'alias' => FALSE,
    'prefix' => ''
  );
?>
```

Now instead of the fatal error, you will get a result similar to:

```
array (
  'file' => '/www/drupal-6/sites/all/modules/admin_links/admin_links.module',
  'line' => 65,
  'function' => 'url',
  'args' =>
  array (
    0 => 'node/62/edit',
    1 => NULL,
    2 => NULL,
    3 => true,
  ),
)
```

The 'file' line should give you a clue as to which module is calling `url()` incorrectly. In this case, it would be the 'admin\_links' module. Then search the module's issue queue to see if anyone else has reported the same problem. If not, file a new bug report so the maintainer can fix the problem. Be sure to include the data from the debugging report in the bug report that you file.

# MySQL : " Warning: MySQL server has gone away " - Tune MySql to resolve this problem

by [abramo](#)

" Warning: MySQL server has gone away "

This dreaded MySql error and subsequent Warning is due, except in rare cases, to a lack of resources available to MySql, such as required for the operation of your Drupal installation. Allowing the necessary resources to MySql resolves this issue most of the time. Also, it is easy to resolve, if you know how to proceed.

Here is a step by step guide, equally valid for your Linux server as well as any local Windows MySql installation you may be using as a trial installation along with your local Drupal installation.

MySql comes with a default configuration of the resources it is going to use, specified in the "my.cnf" file (Linux) or "my.ini" file (Windows) during installation of MySql. This file is located by default at "C:\Program Files\MySQL\MySQL Server X.Y\my.ini" in Windows - or in Linux "/etc/my.cnf" (to set global options) or "/usr/local/var/mysql-data-dir/my.cnf" (to set server-specific options).

Resources allowed by the default configuration are normally insufficient to run a resource-intensive application (but on the safe side just in case the server is not powerful enough). You must modify the following resource specifications if they are available in your original configuration file, or add them to the configuration file if they are not already specified (because some are not present by default) :

(Important: remember to keep backup files \*before\* you do anything !!)

GENERAL SPECIFICATIONS:

[mysqld]

```
port = 3306
socket = /tmp/mysql.sock
skip-locking
key_buffer = 384M
max_allowed_packet = 64M
table_cache = 4096
sort_buffer_size = 2M
read_buffer_size = 2M
read_rnd_buffer_size = 64M
myisam_sort_buffer_size = 64M
thread_cache_size = 8
query_cache_size = 32M
```

INNODB SPECIFIC:

```
innodb_buffer_pool_size = 384M
innodb_additional_mem_pool_size = 20M
innodb_log_file_size = 10M
innodb_log_buffer_size = 64M
innodb_flush_log_at_trx_commit = 1
innodb_lock_wait_timeout = 180
```

Note: It is assumed here that you are using the InnoDB database tables, as Drupal is a resource

intensive application. If you are not using the InnoDB database tables try to change this, in view of the fact that you are getting the "Warning: MySQL server has gone away" - apparently meaning that your setup is resource intensive. [Convert MyISAM Tables to InnoDB](#).

Where do these specifications come from?

MySQL provides sample configuration files within its package (called in Windows "my-huge.ini" , "my-innodb-heavy-4G.ini" , "my-large.ini" , etc - for Linux "\*.cnf"). They are normally to be found in the default MySQL directory. Browse them and determine by yourself what suits you (and your system) best. The above recommendations, provided here as a starter guideline, will do the trick in most Drupal cases and average modern machine specifications. Remember, anything you modify or add must reside within your "my.cnf" / "my.ini" file for the system to find and use. Also, remember to keep backup files *\*before\** you do anything.

## Mysterious "2" or "3" errors

Sometimes a misconfigured Drupal site will display a mysterious "2" or "3" error in the general content area. This happens if the "Error reporting" settings have been configured to display a custom page instead of the default 404 (not found) or 403 (access denied) message and the custom page doesn't actually exist.

This error message has been [updated in Drupal 5.x](#) and later versions so that the "2" and "3" are replaced by more helpful error messages.

For cases when the custom page does exist but the "access content" permission is disabled, please see <http://drupal.org/node/198975>

## Mysterious 403, 404, 406, 500 or "Page not found" errors depending on submitted content

When submitting certain words in content, you receive a 403, 404, 406, 500 or "Page not found" error message.

This behaviour is most likely caused by the Apache module [mod\\_security](#). Depending on its filter settings, phrases that trip the module include

lynx, perl, mother, select from, table, cc:, and many more.

Ask your host to tone down the mod\_security settings.

Alternatively you can try to add the following to .htaccess:

```
# Turn off mod_security filtering.
<IfModule mod_security.c>
  SecFilterEngine Off
</IfModule>
```

This will turn off filtering by mod\_security.

Not all hosts will allow this, so you may need to contact your hosting provider.

## Notice: Undefined variable

This is a warning directly from PHP, and indicates a *very verbose level of logging* is enabled on your server.

See [PHP error reporting](#)

They don't mean that anything is wrong with your install, just that the code is probably taking some short-cuts.

It is unlikely you will encounter (or use) this setting on a production server, but your PHP configuration may have come with it turned on. **It is safe to turn these notices off** and you can do so as documented in the PHP section of the [drupal requirements](#) document by making a change in your php.ini.

```
error_reporting = E_ALL & ~E_NOTICE
```

It is a *goal* of Drupal 6 to eliminate even this level of 'notice'-level warnings in core code, although it's likely that in-development modules may still throw them now and then.

In Drupal 5, the answer is - yeah, just turn off that logging.

In Drupal 6, the answer is - if you really really care - raise an issue against the module that owns the name named in the message. You may be told to just try solution #1 :-)

## PHP Notice: unserialize() ... : Error at offset 6 of 10 bytes in ... \includes\bootstrap.inc on line 428

### Improperly Serialized Variables in Database

When loading a page the following error occurs when loading an improperly serialized array from the variables table. The same technique can be used for bad serialized data in other tables.

### Error

```
PHP Notice: unserialize() [

```

Depending on how caching is set on your site, you may only see this error sometimes. Because once the variables are cached they may not be loaded every time.

### Some Causes of Serialization Issues

- Changing serialized content via sql replace command such as when migrating a site. You can't do this unless you follow up and put the length of the new value in by hand.
- Serializing [resources](#)
- Serializing objects and unserializing them without the class code loaded
- Serialized string is truncated because it is too long for the field it is stored in.

- Encoding/Decoding PHP/mysql issues

## Debugging Technique

Around 480 of bootstrap.inc in drupal 6 change the code to as follows. (Set aside the old code to replace afterward) This will show you the variables that are throwing errors. The @ symbol before unserialize suppresses the error so all the variables can be checked before code execution stops.

```
// if ($cached = cache_get('variables', 'cache')) {
//   $variables = $cached->data;
// }
// else {
  $result = db_query('SELECT * FROM {variable}');
  while ($variable = db_fetch_object($result)) {

    $variables[$variable->name] = @unserialize($variable->value);

    if ($variables[$variable->name] === FALSE) {
      print "<hr/>Unserialize Error for variable:". $variable->name . '=' .
$variables[$variable->name] . "<br/>". $variable->value;
    }
  }
  die;
  cache_set('variables', $variables);
// }
```

You may only get the error when variables are first loaded. After that they may be cached. So to reproduce the error, flush the cache via sql:

```
truncate table cache_block;
truncate table cache;
truncate table cache_content;
truncate table cache_filter;
truncate table cache_form;
truncate table cache_menu;
truncate table cache_views;
```

## Resolution

If you are getting errors with all or many of your variables, this technique will not work as the cause is likely not isolated bad record(s) in your variables table.

Once you find the bad variable(s), you will need to fix it in the database. (You may also override the variable by setting its value in the \$conf array as described at the bottom of the settings.php file but you will not be able to change the variable value via the web interface until you remove the override in settings.php)

Below are what serialized arrays should look like. (look in the variables table):

```
name: node_options_bio
value: a:1:{i:0;s:6:"status";}

name: googleanalytics_track_6
value: b:0;
```

The i represent integers, s strings, and the count is the length within the quotes.

You may be able to fix the value by hand editing it the database table. If you can't, you may just want to delete it, saving the old value somewhere, and let Drupal set it to its default. This may cause problems for some variables.

## Page not found error upon returning from

# the OpenID Provider.

When the user returns from the OpenID provider, a 404 Page not found error is shown.

This behaviour is most likely caused by the Apache module [mod\\_security](#). This module disallows access to URLs with http://, https:// or ftp:// in the query string.

A quick test is to access your site via the URLs

```
http://example.com/?q=node&test=abcd
http://example.com/?q=node&test=http://
```

If the first URL works, but the second displays a "Page not found" error, mod\_security is the likely cause.

For the mod\_security core rules set 2.0.2, it is likely one of the anti-remote file inclusion (RFI) rules in base\_rules/modsecurity\_crs\_40\_generic\_attacks.conf. Rules 1234234 and 340153 are sometimes to blame.

Ask your host to disable the offending rule.

## The GD library for PHP is missing or outdated

Your Drupal 6 *Status report* page returns the following error

**GD Image Rotation Low Quality / Poor Performance**  
**The installed version of PHP GD does not support ...**

This error occurs because PHP is missing the GD graphics library. Not all operating systems (platforms) install it as part of PHP.

Click on below appropriate operating system.

### PHP GD Library on Ubuntu

#### Get GD fully working version for Ubuntu

Click on below appropriate Ubuntu Server version.

-----

### PHP GD library on Ubuntu Server 8.04.x LTS

#### Get GD fully working version

Your Drupal 6 *Status report* page returns the following error

## GD Image Rotation Low Quality / Poor Performance The installed version of PHP GD does not support ...

Here is the steps for **Ubuntu Server 8.04.x LTS**, PHP5, Drupal 5.x or 6.x. It has also been found to work with Ubuntu Server 9.04. It will install a GD pre-compiled fully working version. It is a complete bundled (forked) GD libraries.

### STEPS:

If not already done type in the following command in your TERMINAL to remove your current GD package and its configurations.

```
sudo apt-get --purge remove php5-gd
```

Add 2 lines at the end of your file /etc/apt/sources.list

```
deb http://packages.dotdeb.org stable all
deb-src http://packages.dotdeb.org stable all
```

Type in the following command in your TERMINAL to update your current apt-get list

```
apt-get update
```

Type in the following command in your TERMINAL to install the working GD package.

```
apt-get install php5-gd
```

Wait. When asked 'Do you want to continue' press Y key. Then press RETURN key.

When asked 'Install these packages without verification' press Y key. Then press RETURN key.

This is optional but you might want to make a backup of your php.ini file. Because it might be modify during the following steps. The php.ini file is located under */etc/php5/apache2/php.ini*

Important note: If you have customized your php.ini file all your edits will be erase during the following steps. You will have to edit your php.ini file later. If you have not customized your php.ini file ignore this note and just go to next step.

When asked 'What would you like to do about php.ini?' select the 'MAINTAINER'S VERSION'. Press Y key. Then press ENTER key.

Wait. If successful TERMINAL will return 'ldconfig deferred processing now taking place'.

Type in the following command in your TERMINAL to restart your Apache.

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

Go to <http://your-domain-name-here.com/admin/reports/status> to test it. GD line should have a green checkmark.

If it doesn't work restart your whole Ubuntu server. Wait a few minutes. Then test again.

Enjoy

## Ubuntu Server 9.10.x

For information on compiling PHP, see one of the following links. Be warned it takes about 40 minutes to compile the necessary package.

- <http://www.phpmag.ru/2009/09/12/ubuntu-9-04-php-5-gd-2/>
- <http://cumu.li/2008/5/13/recompiling-php5-with-bundled-support-for-gd-on...> (Note that comments #9 and #30 can be particularly helpful, as PHP 5.2.\* has newer versions since the date of the original post.)

## Ubuntu Server 9.04.x

### Get GD fully working version

The steps on the following page has been found to work with Ubuntu Server 9.04  
<http://drupal.org/node/540838>

## Windows XP

Issue: The GD library for PHP is missing or outdated

Solution: Go to the Windows XP control panel where you can add or change installed programs. Find the line for PHP and select change. A window pops up: find the GD2 module that is not installed by default and select it from the change application.  
 Restart the Apache Server.

This page is derived from a forum thread (<http://drupal.org/node/134331>) which may contain more information.

## The selected file /tmp/file\*\*\* could not be uploaded, because the destination \*\*\*/\*\*\*.\*\*\* is not properly configured.

When using Drupal your browser returns the following error

**The selected file /tmp/file\*\*\* could not be uploaded, because the destination \*\*\*/\*\*\*.\*\*\* is not properly configured.**

This means that the file system path where uploaded files are store during previews is misconfigured. In simple terms Drupal try to upload a temporary file to your server but the process is not working.

An "Internal Server Error" often occurs on:

- Migrating a website to a different server.
- Misconfigured `Temporary directory` setting under `admin/settings/file-system`.

### To fix this

Navigate to `admin/settings/file-system`

Under the section `Temporary directory`: configure the path appropriately. Each server is different.

`tmp`

You must type in the appropriate path to your `tmp` folder on your server. If unsure use a FTP tool. Connect to your server. And search for the folder `tmp`. If nothing works type in `/tmp`. This is the most common path.

If above isn't working then using an FTP tool ensure the CHMOD security setting on your `tmp` folder is appropriate. CHMOD should be `750`.

## Unable to send e-mail.

This is common on shared hosting.

Some reasons this error could crop up are:

1. `php_mail()` is disabled.
2. The user that `php_mail()` is run as (and apache is run as - usually ``nobody'`) does not have permission to send mail on the server.
3. There is no mail server program running on the server.

There may be other reasons that I'm not aware of yet.

Generally, a quick fix (and the best one for shared hosting environments) is to bypass the `php_mail()` entirely using the `smtp.module` (<http://drupal.org/project/smtplib>). You can then configure your site to use an external email account, for example, Gmail.

There may be other ways, depending on the reason, to rectify this by editing or adding `php.ini` settings.

## User error: Duplicate entry

This page is for errors similar to the one below:

```
user error: Duplicate entry '24' for key 1
query: INSERT INTO node (status, moderate, promote, sticky, title, body, format, uid,
created, type, teaser, changed, nid) VALUES('1', '0', '1', '0', 'test', 'this is a test',
'1', '1', '1149692821', 'story', 'this is a test', '1149692832', '24') in
.../includes/database.mysql.inc on line 66.
```

```
warning: Cannot modify header information - headers already sent by (output started at
.../includes/common.inc:384) in .../includes/common.inc on line 192.
```

**The likely problem (short form):** At some point, the sequence table in your Drupal database was not updated.

### The solution

1. Access your database. Check the affected table. In this case, the error tells us that it is the `node` table (query: `INSERT INTO node`). This also happens with other tables, modify these instructions accordingly.
2. This is the node table, so the problem is the `node id (nid)`. Each node has a unique id. Look at your table, and find the highest node id (nid). If you have many nodes, it may help to sort your table by `nid` to find the highest one.
3. Go to your `sequences` table. Change the node id in the sequences table to **a number higher than** the id you found in step 2.

**The problem (long form):**

The 'key' in a table is a column where every entry has to be unique. For example, in the users table, every UID (user id number) must be unique. You don't want two users with the same id!

Drupal keeps track of what is the highest ID value separately, in the sequences table. When a new user is created, Drupal checks the sequences table to find out what the next UID should be. Presumably this is more efficient than checking the user table itself. Drupal then creates the new user, updates the user table with the new user information, and *updates the sequences table with the new UID*. The same thing happens for the node table, comment table, really, a whole bunch of tables.

Sometimes, for some reason, Drupal does not update the sequence table. The number there becomes out of date. So let's say that in the node table the highest NID is 300, but somewhere, something went wrong, and in the sequences table, the NID is only 297. The next time Drupal makes a node, it will check the sequences table, and try to make a node with a NID of 298...whoops! That already exists.

### The solution (long form):

The error itself can tell us a lot. Let's divide it into 3 parts:

The error:

```
user error: Duplicate entry '24' for key 1
```

Duplicate... meaning the '24' is the same as something else in the table, something where no two things can be the same. Which table? Which column?

The query:

```
query: INSERT INTO node (status, moderate, promote, sticky, title, body, format, uid,
created, type, teaser, changed, nid)
```

INSERT INTO [tablename] tells us which table produced the error. What follows is a list of column names.

The values:

```
VALUES('1', '0', '1', '0', 'test', 'this is a test', '1', '1', '1149692821', 'story', 'this
is a test', '1149692832', '24') in ../includes/database.mysql.inc on line 66.
```

The values we tried to insert into columns of the NODE table... We already know '24' was the problem, we see here it is the LAST value, corresponding to the LAST column, the *nid*, or node id.

So the problem is, we tried to insert **24** into the **nid** column of the **node** table, when that value already existed (**duplicate entry!**).

Why? Because the sequence table was not updated, and told Drupal the wrong *nid*. The solution is to enter a value for *nid* into the sequence table that is higher than ANY *nid* in the node table.

To find the highest *nid* in the node table, access your database, however you do it. Find the node table, and sort it by *nid*. Write down the highest value. Let's say for this example that the highest number is 68.

Go to the sequences table, and find the row that has *nid* in it. Replace the number there with ANY number higher than the highest value in the node table. For this example, the highest value was 68. 69 would work, so would 70, 90, and 201. ANY NUMBER higher than 68 would work in this example.

**Additional point:** 99% of the time, the solution is in the sequence table, but this error will come up

ANY time you try to insert duplicate data into a column where every value has to be unique.

# User warning: Illegal mix of collations

## Situation:

You installed Drupal core using a third-party installer. Then, you installed or enabled more modules to add more features to your site.

You encounter illegal mix of collation errors involving latin1\_swedish\_ci and utf8\_general\_ci collations.

For example, whenever you perform a search using a Chinese keyword, you encounter an illegal mix of collation error.

You realised that about 40 tables in the Drupal database have been created with latin1\_swedish\_ci collation.

And, Drupal created the rest of the tables with utf8\_general\_ci collation.

## Solution:

### Manual Method (30 mins to 1 hr)

1. Put your site to offline mode.
2. Backup all your databases (Important).
3. Select phpMyAdmin.
4. Select the correct Drupal database.
5. Modify table collation:
  - Locate table with latin1\_swedish\_ci collation
  - Select Structure (Index finger pointing icon)
  - Select Operations tab
  - Change collation to utf8\_general\_ci
  - Go

6. Modify field collation:

\* Some fields have collations, some do not have collation. Do *not* modify those fields without collation.

- (Select Structure tab)
  - Locate field with latin1\_swedish\_ci collation
  - Select Change (Pencil icon)
  - Change collation to utf8\_general\_ci
  - Save
  - Repeat step 6 to modify other fields with latin1\_swedish\_ci collation to utf8\_general\_ci collation.
7. Repeat steps 4 to 6 to modify other tables with latin1\_swedish\_ci collation to utf8\_general\_ci collation.

8. Put your site to online mode.
9. Make sure that your site is functioning normally.

## Easier Method (2 mins to 10 mins)

After making the mistake of installing Fantastico, there is an easier way to correct the collation of all tables:

Create a new php file on the server with the following and then run it:

```
<?php
$db = mysql_connect('localhost','myuser_mydbuser','mypassword');
if(!$db) echo "Cannot connect to the database - incorrect details";
mysql_select_db('myuser_mydbname'); $result=mysql_query('show tables');
while($tables = mysql_fetch_array($result)) {
foreach ($tables as $key => $value) {
mysql_query("ALTER TABLE $value COLLATE utf8_general_ci");
}}
echo "The collation of your database has been successfully changed!";
?>
```

Make sure to substitute in the above script:

- myuser\_mydbname with your database name;
- myuser\_mydbuser with your mysql username;
- mypassword with your password for the mysql user;

## Warning: "Headers already sent" or "Cannot modify header information"

This error also presents as "Cannot modify header information" depending on PHP version.

In short, it means that *somewhere* in the code, something was printed to the browser *before* Drupal had finished preparing the page. This is most often caused by custom or modified code contributed from sources outside Drupal, so inspect your uniquely added code (including themes) first..

If you get a "Headers already sent" error, there are three likely causes.

If this error is not the first error message on the page, then it is most likely a '**avalanche effect**' of previous errors and you may *ignore it*. Instead, focus on fixing the errors before it. When you fix the first error message(s), the "Headers already sent" error(s) will most likely disappear.

Text editors sometimes insert a UTF-8 byte order mark at the top of a file. In this case, the error message will usually say that "output started" at line 1 of some file. To fix this, configure your text editor to save the file without a byte order mark.

However, if you get an error "Headers already sent" as the **first** error and it tells you the error is near the end of a file (check which file "**output started at**" points to), that probably means that there are extra spaces or lines after a closing `?>` php tag. In [Drupal coding standards](#), it is strongly recommended (for this very reason) that **PHP files should not have any closing `?>` tags** . Delete them, and everything should work fine.

Extra whitespace being added probably is caused by a bad unpacking program and / or a non-compliant editor (Windows Notepad or Wordpad, Mac TextEdit) adding it.

Problems with "headers already sent" can also be caused by having a blank line at the end of \*.inc files. Drupal or more likely PHP seem to have problems with extra spaces here and there.

Check all \*.inc files to make sure there are no closing php ?> tags in any of them. Closing php ?> tags are not needed in your \*.inc files. Also, check all \*.php files to make sure there are no blank lines at the beginning or at the end of the file.

If the error message indicates that this is caused by a module, disable modules one by one to find out which one is causing the problem.

This can also be caused by UTF-8. If a website is coded in ASCII and php files are being saved as UTF-8, it can cause this message. If the website and DB are both UTF-8, it should be ok to save php files as UTF-8.

Additionally, this error message is related to the "output\_buffering" variable in php.ini. If output\_buffering is set to some cache, the server will send headers with delay (or modify them shortly after they are sent), and this error will not be tripped. But, if output\_buffering is set to 0 or not at all, then headers can be sent at only one moment and, if there's bad code, it will trip this error message. To sum up, turning on the "output\_buffering" variable in php.ini fixes this problem.

## Warning: Invalid argument supplied for foreach()

OS X users may notice warnings such as these suddenly appearing in their (remote) Drupal install:

```
warning: Invalid argument supplied for foreach() in /path/to/drupal/includes/theme.inc on
line 435.
warning: Invalid argument supplied for foreach() in /path/to/drupal/includes/theme.inc on
line 442.
```

This problem is caused by using Textmate to directly modify Drupal files on a mounted network drive. Textmate tries to store some information about edited files (such as the caret position) in hidden files, but in some environments the names of these files will be mangled. For example, '.\_myfile.php' becomes ':2e\_myfile.php'. Besides the aforementioned warnings, the presence of such files will also cause other unexpected behaviour in Drupal.

### Solution

Run the following command in a new terminal window to prevent Textmate from creating hidden files on network drives:

```
defaults write com.macromates.textmate OakDocumentDisableFSMetaData 1
```

Of course, you'll still have to manually track down and delete any existing files. You'll need a shell connection for this, as these files are invisible in Finder.

## Warning: MySQL server has gone away

This dreaded MySQL error and subsequent Warning is due, except in rare cases, to a lack of resources available to MySQL required for the operation of your Drupal installation. Allowing the necessary resources to MySQL resolves this issue most of the time. See <http://drupal.org/node/259580> and <http://dev.mysql.com/doc/refman/5.0/en/gone-away.html> about the causes and solutions for this error message.

## Warning: Permission denied in includes/file.inc

This describes an error which may occur with Drupal 4.5.2. The details for other versions may differ.

<http://www.example.com/?q=admin/settings> might yield error messages like this:

```
warning: mkdir(files): Permission denied in /data/www/public/includes/file.inc on line 77.
warning: mkdir(files/tmp): No such file or directory in
/usr/data/www/public/includes/file.inc on line 77.
```

This means that Drupal needs write access to create (and later access) the files and files/tmp directories. One way to solve this is to give the webserver write access in the directory. Another common solution seems to be granting everybody write access to the "files" directory. Both solutions have the drawback that somebody else is able to write files into that directory. [if you know a better solution, please mention it here]

### Solution 1 (recommended)

In the drupal root directory:

```
mkdir -p files/tmp
chown -R www files
```

Note: you may need to substitute www with the user id of your webserver process.

### Solution 2

In the drupal root directory:

```
mkdir -p files/tmp
chmod -R 777 files
```

## Warning: fopen(./sites/default/settings.php) [function.fopen]

When installing Drupal your browser returns the following error

```
Warning: fopen(./sites/default/settings.php) [function.fopen]: failed to open stream:
Permission denied in ...domainname.com\includes\install.inc on line 236
```

```
Warning: Cannot modify header information - headers already sent by (output started at
...domainname.com\includes\install.inc:236) in ...domainname.com\includes\install.inc on
line 618
```

```
Warning: Cannot modify header information - headers already sent by (output started at
...domainname.com\includes\install.inc:236) in ...domainname.com\includes\install.inc on
line 619
```

To fix this click on one of the bellow possible causes.

## Misconfigured Drupal file or folder read and write permissions. On a multi-sites

# installation.

The following assume that you have a Drupal **multi-sites** installation.

## Steps

1. Using a FTP tool set the following file CHMOD to 777  
`.../sites/default/default.settings.php`
2. Set the following file CHMOD to 777  
`.../sites/default/settings.php`
3. Then set the following folder CHMOD to 777  
`.../sites/default`
4. **Important: For security reason when the installation is done** set the following CHMOD.  
 Set the following file CHMOD to 700.  
`.../sites/default/default.settings.php`  
 Then set the following file CHMOD to 744.  
`.../sites/default/settings.php`  
 Finally set the following folder CHMOD to 755.  
`.../sites/default`

## You have chosen to open ... which is a PHTML file from http://localhost

After completing an installation using <http://localhost/mydrupalsite> you are unable to see the site. Instead, you are presented with a dialog box. If you are using Firefox, the box is called "Opening" which reads "You have chosen to open [blank space] which is a: PHTML file from <http://localhost>. What should Firefox do with this file?" etc.

Try opening your site using the local IP address <http://127.0.0.1/mydrupalsite>.

Sometimes you may have fixed the problem but you still need to clear your browser cache (or use another browser) before the site displays properly.

## HTTP request status problems and solutions

I feel it is necessary to start yet another page on the subject.

As I am fairly new to Drupal and are trying to find my way through this problem of **HTTP request status** for a second time I end up more confused and depressed as I keep reading on. And as I try out more & more of the suggested solutions I get further away from a working one for me.

So, the first idea is to get going with a list of all those discussions and nodes available on the subject. And please help by adding all the nodes not shown here yet - if you post a comment I will gladly add them.

On starting this node it seems the problem needs to be approached with the following steps:

- Listing of the problem - Node List of discussions & information below
- Classification of the problem - Installation: hosted, local server (development), RSS related, etc.
- ...
- Identification of the solution related to Installation (above)

I will try to add a summary of the gist of the thread - at least from the starting point. There is no specific order in the list yet.

Here we go:

## List of Discussions & Information

<http://drupal.org/node/245990#comment-909263> -->> "...here is an explanation of this bug ..."  
<http://drupal.org/node/222454> -->> Drupal 6 - DNS problems ....

<http://drupal.org/node/245990> -->> Drupal 7.x-dev - Component: base system - Category: bug report  
<http://drupal.org/node/245853> -->> D 6.2 - moving to a new server ..  
<http://drupal.org/node/227791> -->> zilla note for kindred souls - web host problems ...  
<http://drupal.org/node/7881> -->> Drupal 7 and then back to 4.5 and back up-to-date - "... RSS feeds don't work if there is proxy server running ..."  
<http://drupal.org/node/50912> -->> Drupal 6.2. - HTTP request overrides headers and data on redirect  
<http://collectiveidentity.net/node/13> -->> "Collective Identity in depth discussion and walk thru of core hack"  
[http://drupal.org/project/http\\_request\\_fail\\_reset](http://drupal.org/project/http_request_fail_reset) -->> Drupal 6 module - Http Request Fail Reset  
[http://wiki.dreamhost.com/index.php/Allow\\_url\\_fopen](http://wiki.dreamhost.com/index.php/Allow_url_fopen) -->> Dreamhost wiki - "allow\_url\_fopen"  
<http://kb.mediatemple.net/article.php?id=793> -->> Article from Media Temple (MT) - Why is allow\_url\_fopen disabled on the (gs) Grid-Service?  
<http://drupal.org/node/346929> -->> Drupal 6.8. - Http request status fails after upgrade to 6.8  
<http://drupal.org/node/203998> -->> Drupal 6 - Upgrading from Beta 6 to RC1 ...  
<http://drupal.org/node/258760> -->> Drupal 6.2 - RSS feed ..  
<http://drupal.org/node/258735> -->> Drupal 7.x - dev - News aggregator should handle more HTTP statuses - RSS feed ..  
<http://drupal.org/node/236664> -->> "Unable to fetch any information about available new releases and updates" (HTTP Server Request Fails) while "off-line"  
<http://drupal.org/node/245990> -->> HTTP request status fails - 7.x-dev - .. problem appeared after the Feed aggregator had been working for several days ...  
.. -->> ..

## Classification of the problem

.....

Please help – your comments will be added here.

## Solutions

Some general points in order to see if any change you did works:

Backup your Database - MySQL, etc.

Make a note of the changes you are doing and why – what is the expected outcome?!? then

### Clear the Cache

- **Empty Drupal cache** -->> ... /admin/settings/performance or

use the development module, add the block and click it from the block (usually placed in the left sidebar)

Another great tool -imho- is the administration module - [http://drupal.org/project/admin\\_menu](http://drupal.org/project/admin_menu). It is a simple, clean and small menu bar across the top and it can be made to stay there when scrolling. There are other modules available, but for a beginner this might be a good start.

- **Clear your browsers cache** as well (Safari) has the "Develop" menu where you can tick the "Disable Cache" option easily.

- **Run cron manually** and then check if your changes have worked.
- **Sometimes** it might even help to restart your browser - and all the open windows - not just the one you are working in.
- **Working on a localhost** on your desktop (Wamp - such as WampServer, Xampp, A2T, etc.) it might be good practice to "Restart all Services" in order to get your server to start clean and fresh. And be aware that the HTTP problem may be there **one day and disappear the next**. Especially with RSS feeds check if the feed is actually working or is the server of your RSS feed not available or so slow that Drupal is getting bored to wait and is timing out? This aspect of unpredictable behaviour one yes, one day no makes this a very frustrating business to resolve.  
Is the problem only on a particular feed or throughout your installation?

.....

Please help – your comments will be added here.

#### **Note:**

Please when commenting add sufficient detail about

- your set-up (hosted, running on localhost at home using Wamp, etc.),
- when the problem started (suddenly, after installing module XYZ, etc.) and
- how you solved it (installing module ABC, cleaning cache, running cron, snippet ..., patch ..., etc.).

Please do not be shy to tell us newcomers where exactly to put the code, where to find the file, etc. (for example some solutions refer to modifying php.ini - well a newcomer might not find it, because on most third party host servers it is not accessible).

And thanks for the help of all and everyone!!

## Pages Display Incorrectly

The following pages provide information that might help if you find that your site does not display content the way that you expect.

## How do I get rid of the "Welcome to your new Drupal website" on the front page?

If you are seeing this message, it is because your front page is set to 'node' and you have no content promoted to the front page.

To fix this, you either need to (1) [promote something](#) or (2) [change your front page](#), or (3) [add an url alias](#).

1. # Edit a node (or create a new node) and click the arrow next to "Publishing Options" to expand that section. There is a checkbox labeled "Promoted to front page". If you check that and (re)submit the node it will show up on the front page and the welcome message will disappear.
2. # By default, the front page of your site is 'node', which is a list of all nodes that have been promoted to the front page as explained in (1). If you don't like this, you can change your front page to any page you like.
  - In Drupal 4.7, go to [www.example.com/admin/settings](http://www.example.com/admin/settings) and click on "General settings" to

expand that section

- In 5.x and 6.x, go to [www.example.com/admin/settings/site-information](http://www.example.com/admin/settings/site-information)

Towards the bottom, you will see 'Default front page'. If it's still at the default, it will be 'node'. Change this to the page you would like to be your front page. That is, the page that loads when you go to [www.example.com](http://www.example.com). For example, if you want people to go straight to the blogs, you would put 'blog' in that setting, or you could set it to a specific node or view.

3. <#> If you rely on a custom front page like in (2), and that you don't use nor need to have nodes promoted to the front page, you can still see the welcome message when visiting [www.example.com/node](http://www.example.com/node). Moreover, it will be indexed by search engines. While you can prevent it from being indexed by editing the robots.txt file, or add a redirect in your .htaccess, you can also do it directly in Drupal by adding an alias (provided you have the Path module enabled). To do so, go to [www.example.com/admin/build/path/add](http://www.example.com/admin/build/path/add) and fill out the fields. For the "Existing system path", you can enter the path to your custom front page or any other page, and for "Path alias", enter "node". By typing [www.example.com/node](http://www.example.com/node) you will then land on the page you defined instead of seeing the welcome message.

The welcome message is hard coded into Drupal's core files so it is not possible to have your front page set to 'node', have no alias, have no nodes promoted, and not have this message without changing the source code.

### Anonymous users can see the welcome page

If you are having problems with anonymous users seeing the "Welcome to your new Drupal website" page, then you are likely experiencing problems with a contributed node access module. For troubleshooting, visit, [Why is my "Welcome to your new Drupal website" visible to anonymous users?](#)

### Change the front page through the theme <#>

From [a tip by garcia](#)

A simple way to change what displays on the front page without installing modules:  
In your themes `page.tpl`, change the line

```
<?php print $content; ?>
```

and replace it with

```
<?php if (!drupal_is_front_page()) { ?>
<?php print $content; ?>
<?php } ?>
```

This prevents the "front" page from displaying the any of the content that would normally show there. You can have your frontpage free from the default welcome message of "River of News" but with all the other things.

You can get a similar result by using the [theme system template override filenames](#) to [create an entirely different front page template](#)

### Modules to help <#>

[Front.module](#) can be installed to give you lots more control over the front page.

[Panels.module](#) is also often used to create a front page made up of different configurable sections.

## Why is my "Welcome to your new Drupal website" visible to anonymous users?

When anonymous users see the 'Welcome to Drupal' message on the frontpage (?q=node) *even though you have promoted nodes to the frontpage*, you are experiencing an access problem.

Here's a quick list of things to check:

- Check admin/user/permissions. Do anonymous users have 'access content' permission?
- Do you really have some nodes promoted to the front page? Check admin/content/node/overview. In the filter section, select "status" and "promoted". Are there any nodes in the list?
- Do you have the "Organic groups access control" module enabled? If so, open the "Groups" section of the node's Edit page. If any of the "Audience" items are checked, make sure the "Public" checkbox is also checked. Otherwise, that node won't be visible to anyone outside of that audience (and users with 'administer nodes' permission).
- Have you set the front page path on the settings page (admin/settings/site-information "Default front page") to something other than "node"?
- Did you uninstall an access module\* without disabling it properly?
- Did you try rebuilding permissions? Go to admin/content/node-settings and click "Rebuild permissions".

If the above conditions have been met, you may use the following SQL queries to reset the node\_access table. You can run these from the command line or in phpmyadmin by clicking on the SQL tab of your right hand pane and pasting the queries into the text area, then clicking go.

**Warning:** the following SQL queries will reset **the entire** node access table:

```
TRUNCATE node_access;
```

This query deletes all access rules. The next query grants the 'view' permission to all nodes.

```
INSERT INTO node_access (nid, gid, realm, grant_view, grant_update, grant_delete) VALUES (0, 0, 'all', 1, 0, 0);
```

\* For example: taxonomy access, organic groups, nodeprivacy by role, simple access

## Database Collation (language) Inncorrect, page looks scrambled, or you get next Ä >>

This can be a serious problem, if you have a site that's in use and it was put in with another language by default in sql, which is possible even if Drupal tries it's best during the setup to avoid this problem. This problem can be from a few sources, on Linux, your system language **MUST** be UTF8 done through the "locales" package, which you will need to look up on your own however feel free to add a page with info for fixing that if you have it.

Problems in the database will crop up in odd ways, perhaps only when new content is added, or when a module changes something, and there for strike you after you've gone out of sandbox and lead to HUGE headaches.

This solution is From: <http://stackoverflow.com/questions/105572/a-script-to-change-all-tables-and-fields-to-the-utf-8-bin-collation-in-mysql> and is not my work.

```
<?php
function MysqlError()
{
    if (mysql_errno())
```

```

    {
        echo "<b>Mysql Error: " . mysql_error() . "</b>\n";
    }
}

$username = "root";
$password = "";
$db = "database";
$host = "localhost";

$target_charset = "utf8";
$target_collate = "utf8_general_ci";

echo "<pre>";

$conn = mysql_connect($host, $username, $password);
mysql_select_db($db, $conn);

$stabs = array();
$res = mysql_query("SHOW TABLES");
MysqlError();
while (($row = mysql_fetch_row($res)) != null)
{
    $stabs[] = $row[0];
}

// now, fix tables
foreach ($stabs as $stab)
{
    $res = mysql_query("show index from {$stab}");
    MysqlError();
    $indicies = array();

    while (($row = mysql_fetch_array($res)) != null)
    {
        if ($row[2] != "PRIMARY")
        {
            $indicies[] = array("name" => $row[2], "unique" => !($row[1] ==
"1"), "col" => $row[4]);
            mysql_query("ALTER TABLE {$stab} DROP INDEX {$row[2]}");
            MysqlError();
            echo "Dropped index {$row[2]}. Unique: {$row[1]}\n";
        }
    }

    $res = mysql_query("DESCRIBE {$stab}");
    MysqlError();
    while (($row = mysql_fetch_array($res)) != null)
    {
        $name = $row[0];
        $type = $row[1];
        $set = false;
        if (preg_match("/^varchar\\((\\d+)\\)$/i", $type, $mat))
        {
            $size = $mat[1];
            mysql_query("ALTER TABLE {$stab} MODIFY {$name} VARBINARY({$size})");
            MysqlError();
            mysql_query("ALTER TABLE {$stab} MODIFY {$name} VARCHAR({$size})
CHARACTER SET {$target_charset}");
            MysqlError();
            $set = true;

            echo "Altered field {$name} on {$stab} from type {$type}\n";
        }
        else if (!strcasecmp($type, "CHAR"))
        {
            mysql_query("ALTER TABLE {$stab} MODIFY {$name} BINARY(1)");
            MysqlError();
            mysql_query("ALTER TABLE {$stab} MODIFY {$name} VARCHAR(1) CHARACTER
SET {$target_charset}");
            MysqlError();
            $set = true;

            echo "Altered field {$name} on {$stab} from type {$type}\n";
        }
        else if (!strcasecmp($type, "TINYTEXT"))
        {
            mysql_query("ALTER TABLE {$stab} MODIFY {$name} TINYBLOB");
            MysqlError();

```

```

mysql_query("ALTER TABLE {$stab} MODIFY {$name} TINYTEXT CHARACTER
SET {$target_charset}");
MysqlError();
$set = true;

    echo "Altered field {$name} on {$stab} from type {$type}\n";
}
else if (!strcasecmp($type, "MEDIUMTEXT"))
{
    mysql_query("ALTER TABLE {$stab} MODIFY {$name} MEDIUMBLOB");
    MysqlError();
    mysql_query("ALTER TABLE {$stab} MODIFY {$name} MEDIUMTEXT CHARACTER
SET {$target_charset}");
    MysqlError();
    $set = true;

    echo "Altered field {$name} on {$stab} from type {$type}\n";
}
else if (!strcasecmp($type, "LONGTEXT"))
{
    mysql_query("ALTER TABLE {$stab} MODIFY {$name} LONGBLOB");
    MysqlError();
    mysql_query("ALTER TABLE {$stab} MODIFY {$name} LONGTEXT CHARACTER
SET {$target_charset}");
    MysqlError();
    $set = true;

    echo "Altered field {$name} on {$stab} from type {$type}\n";
}
else if (!strcasecmp($type, "TEXT"))
{
    mysql_query("ALTER TABLE {$stab} MODIFY {$name} BLOB");
    MysqlError();
    mysql_query("ALTER TABLE {$stab} MODIFY {$name} TEXT CHARACTER SET
{$target_charset}");
    MysqlError();
    $set = true;

    echo "Altered field {$name} on {$stab} from type {$type}\n";
}

if ($set)
    mysql_query("ALTER TABLE {$stab} MODIFY {$name} COLLATE
{$target_collate}");

    // re-build indicies..
    foreach ($indicies as $index)
    {
        if ($index["unique"])
        {
            mysql_query("CREATE UNIQUE INDEX {$index["name"]} ON {$stab}
({$index["col"]}");
            MysqlError();
        }
        else
        {
            mysql_query("CREATE INDEX {$index["name"]} ON {$stab}
({$index["col"]}");
            MysqlError();
        }

        echo "Created index {$index["name"]} on {$stab}. Unique:
{$index["unique"]}\n";
    }

    // set default collate
    mysql_query("ALTER TABLE {$stab} DEFAULT CHARACTER SET {$target_charset} COLLATE
{$target_collate}");
}

// set database charset
mysql_query("ALTER DATABASE {$db} DEFAULT CHARACTER SET {$target_charset} COLLATE
{$target_collate}");

mysql_close($conn);
echo "</pre>";

```

save that as utf8.php and put it some where in your sites/[name]/ directory..  
example:

```
/home/drupal/drupal/sites/test.com/utf8.php
```

All other directories are protected by your .htaccess (assuming it's properly configured) and you won't be able to access it. Or change your .htaccess properties to allow it to be run.

make sure it's set to run

```
chmod 770 utf8.php
```

and then point your browser to your site

```
http://test.com/sites/test.com/utf8.php
```

It will then run, fixing, and adjusting the collation tables on all necessary tables. This script on your own server can be run from the shell aswell by adding:

```
#!/usr/lib/php5
```

this may change depending on your OS, this works on Debian.

At the file it will run through the php parser when you run it execute it with:

```
./utf8.php
```

Run through the shell it will avoid the need to put it into the Drupal directory and avoid the security issues that come along with that. This would be the advised course, however not possible with every web host, or server setup, which is why both sets of instructions are provided.

## Forum overview stopped working

Following a problem of the forum overview stopped working (missing vocab after disabling, deleting, re-enabling the module [as described here](#))

When the forum module is first enabled, it creates a vocabulary of its own (called Forum) where it stores half its structure.

*If* for any reason this vocabulary is deleted (like you decide you don't want forums yet, or you're working on an unstable development site) re-enabling the module **will not correctly re-create the needed vocab again**

Thus, you end up with only half a working forum. It seems you can post topics, but the indexing and overview pages are a bit scrambled, and don't list anything useful.

(Just remaking a new 'Forum' vocabulary manually will not work.)

### The Fix

... is to tell the forum module to forget its old invalid vocabulary id and make a new one.

Short answer is this code needs to run:

```
<?php
variable_del('forum_nav_vocabulary');
?>
```

I use the devel.module, which provides a console where I popped this line, and a variable viewer which showed me what happened, but another quick way is:

- Create new node (don't save)
- Set content to php code. Disable wysiwyg if needed
- Paste the above in the body
- Preview
  - the php code will execute, although you won't see anything yet.
  - discard the node, don't need to save it.
- visit admin/forums
  - this triggers the forum code to check if its parameters are correct.
  - they are *not* so it will create the Forum vocab as needed
- continue, create a new forum or container. The pages should work again as they are supposed to.

## From comments

You may need to perform a few more steps for Drupal 5. At least I did when I disabled and then enabled my forum in Drupal 5.2

- 1) Follow step one from above
- 2) Then visit the /admin/content/forum URL as described above
- 3) Then preview the following php code in the same manner described above

```
<?php
$vid = _forum_get_vid();
print 'This is the result from _forum_get_vid() ' . $vid;
$vid = variable_get('forum_nav_vocabulary', '');
print '<br>This is the result from variable_get(forum_nav_vocabulary, ) ' . $vid;
?>
```

You should get a number. Use that number for the next step. For instance if your number was a 1 then use 1 below ...

- 4) On table vocabulary\_node\_types I added the row vid=1 type=forum

This brought back all my forums and I did not have to delete any posts, categories, forums, or database tables.

Another thing that can happen is the forum vocabulary forgets the forum node type is part of it. It's impossible to fix this through the UI because the checkbox is force checked and greyed out. So you can do it in the database:

```
INSERT INTO `YOUR_DATA_BASE_NAME`.`vocabulary_node_types` (`vid`, `type`) VALUES ('1', 'forum');
```

Replace YOUR\_DATA\_BASE\_NAME with the name of your database and '1' with the ID of your forum vocabulary.

Obviously back up your database before you do anything

## Junk {head} {styles} codes as output

Symptom:

When visiting your Drupal site for the first time you see what looks like code output, instead of the expected theme:

```
{head} {styles} {_BLOCK_.header.logo} {_BLOCK_.header.site_name} {secondary_links}
{primary_links}
{_BLOCK_.header.blocks}
```

This chunk of code is followed by the standard welcome message.

Cause:

Changes made after 4.6.x was released in PHP 5.0.5+/5.1.1+ and the template engine XTemplate are incompatible.

The quickest solution is to go to <http://www.example.com/?q=admin/themes> and change the default theme to **Chameleon** or **Marvin**.

Before you can change the theme you'll need to be logged in. Either create the first account or login at <http://www.example.com/?q=user>.

Another solution is to switch to a PHPTemplate based theme. PHPTemplate is another theme engine, which will become the default in Drupal 4.7 anyway. You will need to [download and install PHPTemplate](#) before you can use [themes](#) that depend on it.

Make sure to disable all XTemplate themes to prevent visitors from using them. Drupal 4.6 comes with two XTemplate themes included: Bluemarine and Pushbutton. A [PHPTemplate version of Bluemarine](#) is available. Primary/Secondary links configured in the theme general settings will not carry over to PHPTemplate. You will need to re-configure any Primary/Secondary links in the PHPTemplate configuration of the theme.

## My layout collapses - content appears below left column (IE) or overflows over the right (FF)

This commonly happens in themes that use CSS for flexible column layouts.

First, please [Validate your page](#) and fix all the problems you can. [Use the web developer toolbar](#) to validate local pages.  
More on validation below.

The most common problem is:

### The content is too wide for the space it's been given

#### [What's a poor robot to do?](#)

If you tell a robot to push 1130 pixels into a bag that only holds 1000 - something's gonna break. Overlap or overflow are the only things the page renderer can do. (There is currently no supported 'shrink to fit' option, and 'expand to fit' isn't available in current versions of CSS.)

This is often seen in pages that embed tables, objects (like Flash) or large images that are of fixed-width and end up making the content column (plus sidebars) wider than the available screen space. With percentage-width themes, the available space will depend on browser window size, which you cannot control.

CSS work-arounds abound, but the first solution is to *shrink your content* if at all possible. This is especially easy if you've pasted in a fixed-width table. Just remove the `width` attribute.

Another approach is to apply `#content{overflow:auto;}` on your CSS.

This results in a *scrollbar* in the middle of your page when the content forces it, and is not always very nice-looking, but it's better than the alternatives.

`#content{overflow:hidden;}` also works, is safer on your design, and looks better (maybe) but may hurt accessibility slightly.

You can try `#content{min-width:450px;}` and similar fixes, but you'll need to browser-test. Any suggestions on getting this right are welcome.

Due to [different browser handlings of the CSS box-model](#), the calculated width of an element can be different than you imagined.

If you want your left column to be 200 wide and your content of 600px wide in a box 800 wide, it sounds like there should be no trouble. But then setting a 2px border around the column may (in older IE at least) *add* to the width and overflow your layout. Plus there are many more 'quirks' between padding and margins etc. The easiest quick fix is to NOT go for pixel-perfect width, and always give your layout a few pixels elbow room.

Last resort - if you would rather the page always expand to fit your content - and trigger browser scroll bars at lower resolution - is switch to an old-fashioned table-based layout. If CSS is too hard for you, you may have to give up on [tableless design](#).

OTOH, if you are keen, alternative fixes are always being proposed over at [alistapart.com](#) and CSS3 apparently will make this dilemma solvable.

## Validation Woes

While validation is The Right Thing to do, some warnings/errors are important, some less so. In general:

- document type does not allow element "div" here; (inside a P)
  - may have side effects, but is probably non-fatal. It's because of the way some themes, or the line-break converter place extra (block-level) markup inside paragraphs - which is not technically right. Can ignore.
- end tag for element "..." which is not open.
  - Possibly related to the above. Fix it if possible
- there is no attribute ... or required attribute ... not specified.
  - OK to ignore.
- or end tag for "..." omitted, but OMITTAG NO was specified
  - Likely to be a sign of layout problems. If the end of a DIV or a TABLE (or other block-type tags) is missing this will probably be very bad and needs fixing. Traditionally seen in badly cropped teasers that try to display the top half of a table.
- an attribute value specification must be an attribute value literal unless SHORTTAG YES is specified.
  - Fix it if possible, but probably safe to ignore.
- cannot generate system identifier for general entity ... or reference to entity "..." for which no system identifier could be generated.
  - OK to ignore.

For more CSS troubleshooting, [try the instructions for using the CSS inspector to find the active styles](#) and go from there.

## Some CSS from your base theme makes your css fail in IE

for example in genesis starter theme (a great theme for being used as base theme), there is a css line like this:

```
div.block {position:relative;}
```

then if you use *blueprint CSS framework* or any other layout css frameworks, it may lead in RTL - IE the text place **under** your sidebar. you should always take care of the system and base theme css files to not conflict with your css.

for debugging css in IE one may do like this:

<http://drupal.org/node/607110#comment-2767554>

## Unusable theme - How To reset your theme via the database

### Situation:

Either you have installed or modified a theme that just breaks everything, or you've deleted the current theme without disabling it first.

As you've now learned, you should have switched out of the theme via administration before destroying it, because you are left with nothing or almost nothing as a UI to work with now.

But we can probably recover.

### If you see *some content but no navigation blocks*

We can fix up the themes by going directly to the required pages.

1. You'll have to be logged on, so enter <http://example.com/?q=user> in the browser. Logon as normal. (use your own Drupal path of course)
2. Enter <http://example.com/?q=admin/build/themes> (D5) or <http://example.com/?q=admin/themes> (D4) to see a version of the themes page. Select a valid theme (eg BlueMarine) and things should be in a state where you can continue.

**If you see *no content at all, maybe garbage or a totally blank screen*** or the above doesn't work...

We need to do surgery on the database.

It's easier if you choose a theme name that you know used to work, eg 'garland' or 'bluemarine' for D4.

Either on the commandline, or via an administration interface (eg PHPMyAdmin) enter the following query

```
UPDATE system SET status=1 WHERE name = 'garland';
```

Then either:

```
UPDATE variable SET value='s:7:"garland"' WHERE name = 'theme_default';
TRUNCATE cache;
```

Note that 's:7' refers to the length of the following string. Modify as needed. This is database surgery, tricky stuff.

OR

If you are using per-user themes, and you've just messed it up for yourself as admin, try

```
UPDATE users SET theme='garland' WHERE uid = '1';
```

Be careful, as getting either of those lines wrong can mess things up just as badly.

If you see some of the theme, but no navigation blocks, you may be able to proceed as in the first case, described above.

**That didn't work**

It's hard to guess how badly your system is damaged at this point. Did you delete the phptemplate directory?

Try re-uploading the entire contents of the /themes directory from a clean distribution of the exact same version of Drupal.

**See Also:**

[Disabling Modules via phpMyAdmin](#)

[Download a fresh copy of a missing or corrupted module/image/file](#)

**Linux Folder Permissions**

The White Screen of Death occurred while installing the Zen theme for Drupal 5. Renaming the folder from zen to zen-base, and then nothing worked - white screen. Renaming the folder back to zen allowed got the site working again. The solution came from reading this issue:

<http://drupal.org/node/222934>

The problem was complicated by the fact that after unzipping the zen files into the theme directory (fedora linux), but forgetting to set (chown -R) the owner and group of the directory and files it contains to the right users. The theme still shows up in the admin section, but if enabled and it and the directory and files belong to users without permissions, you also have problems with white screens.

It was trying to fix this, without realizing that the owner and group were wrong, followed by renaming the folder, that compounded the problem.

**Why doesn't color picker appear on theme configuration page?**

The theme doesn't support color picker

Not all themes support the color picker. Check the theme's documentation to find out.

Color module is not enabled

Make sure to enable color.module on *Administer » Site building » Modules* (admin/build/modules).

File download settings

The color picker is incompatible with the Private download method and will only show when the download method is set to **Public - files are available using HTTP directly**.

Visit *Administer » Site configuration » File system* (admin/settings/file-system) to check the download method your site uses.

PHP's image library GD is not enabled.

Check your status report at *Administer » Logs » Status Report* (admin/logs/status) to see if the GD library is working on your server. This is needed to actually create the images. If it is not, you need to install it or get your host to do so for you.

Javascript is disabled.

You still see the Color scheme fieldset with a select box and a number of textboxes, but no longer the color wheel or the preview.

Make sure you have Javascript enabled in your browser.

# Your editing seems to be having no effect - browser cache problems

These two Firefox extensions are useful for clearing up site development problems caused by server files being cached by the browser.

These are great for those frustrating situations where you repeatedly edit a file on the server, but it appears to have no effect, because (you eventually realize) your browser is still using the old cached version that it fetched 20 minutes ago.... in my experience this seems to happen quite often with css, js and xml files

CacheViewer (view and delete individual browser cache files)

<https://addons.mozilla.org/en-US/firefox/addon/2489>

Johnnycache (stop the browser from fetching certain files from the cache)

<https://addons.mozilla.org/en-US/firefox/addon/3817>

Use Cacheviewer to delete individual files from the browser cache (delete them one at a time because the multiple select and delete doesn't work). This is foolproof, because if the file isn't in the cache any more, you can be certain the browser will be forced to get the new version from the server.

Cacheviewer is also useful for a general understanding of WTF is going on, such as finding out which files the client is downloading, when they were last grabbed and precisely which path they're coming from. (although other Firefox addons, like FireBug, can help here as well).

In theory, Johnnycache can prevent cache access for individual files or paths - which would be perfection. In practice, this doesn't work in some cases (maybe I'm giving it the wrong path somehow).

You can use Johnnycache to block caching for the entire site under development (just use yoursitename.com). This always works, but it can slow down loading of complex pages. But I'd suggest doing this when you first install Johnnycache, to get an idea how it works.

Note that Drupal also has its own internal caching system which might cause issues similar to those described on this page - for anonymous users only. It can be turned off while you're developing your site (not recommended when site is accessible to the public). Go to Administer > Site configuration > Performance to turn it off. In Drupal 6+, if you need to keep caching on, there's a button to clear the cache.

The only other caching problem these addons can't protect you from is your ISP's badly-configured proxy invisibly caching the files. One way to detect this situation is to rename the file on the server - thereby bypassing the proxy the next time the file is fetched.

## Related Issue - take care with old files and folders

There's another quite different issue that might have the same effect of making edits seem useless.

To take one example: You copy a theme directory, e.g. to make a quick backup. You give the copy a different name, of course. You leave that copy in the same place (inside the same directory) as the original. You assume that Drupal will continue to use the theme files in the original folder (well, why not? it hasn't moved and the name is still the same). But in fact Drupal may start using the theme files from the wrong folder, *even though you have changed its folder name*.

In some situations, Drupal can search parts of the file system for files it needs. This is a useful feature, but it means that it is a bad idea to leave old copies of files and folders lying around inside your Drupal directory and its sub-directories. This guideline *definitely* applies to theme folders, module folders and template (tpl.php) files, maybe to others as well.

## Silent WSODs (White Screen of Death) - fixing step by step

If you experienced White Screen of Death (WSOD) and your page is completely blank, don't panic.

### 1. Make sure that you have proper PHP Configuration, no PHP syntax errors or memory limit

Read more: <http://drupal.org/node/158043>

You can also check manually your {watchdog} table for recent errors

```
SELECT * FROM watchdog ORDER BY wid DESC LIMIT 20
```

On Windows you can download: MySQL Administrator (usage is very easy)

<http://dev.mysql.com/downloads/gui-tools/5.0.html>

### 2a. You may use DTools module to diagnose common problems for you

Download Link: <http://drupal.org/project/dtools>

and follow the README.txt instruction how to execute it when you've got WSOD on all the pages

It will help you to fix common WSOD problems like:

- menu\_execute\_active\_handler() in index.php returns NULL
- Content of theme('page') rendering is empty
- Duplicated module paths

Sometimes copy the same module to make backup, but if the second copy is inside Drupal, it causing module duplicates

- Fix the wrong theme and module paths in database

### 2b. Easy manual quick-fixing

**- renaming directory name of bad module can temporary ignore that module without database changing and find that one which cause the problem**

Go to contributions module directory (sites\all\modules\ or modules\) and before each refreshment rename couple of modules.

In example:

Rename module dir: sites\all\modules\Views to sites\all\modules\Views\_

and refresh the page if this helped

Rename next module dir: sites\all\modules\Panels to sites\all\modules\Panels\_

and refresh the page if this helped

Note: please ignore core modules like filter, system, user, etc. to prevent internal error dependencies. Those modules doesn't have any critical issues.

### 3. CSS & JS Compression Issues

If you upload your project to a shared host and do not have sites/default/files/css and /js folders with permissions set to 777, you could possibly experience a WSOD when you go to enable CSS and JS aggregation (admin/settings/performance). It's better to create these folders on your dev machine, give them the 777 permissions, and then upload your project to the live site.

However, if you find that you are live and you cannot change the permissions, disable the aggregation and try renaming the css & js folders. Then create new folders with permission 777.

Hopefully when you re-enable your compression, you won't get the WSOD.

### Links to common Drupal WSOD issues

[#495194: Fatal error: Unsupported operand types in form.inc on line 511 cause WSOD](#)

[#352344: Fatal error: Unsupported operand types in common.inc on line 1369](#)

[#496198: module\\_list\(\) called with wrong arguments causing WSOD and breaking theme registry](#)

More comming soon...

---

Above solutions wasn't helpful enough?

Contact me:

kenorb: <http://drupal.org/user/191974>

with the details of the problems, I'll try to help you

## Site Configuration, Module and Installation Issues

The following pages provide information about problems you might encounter with installation, configuring your site or installing modules.

### Solve installation errors in Drupal 6

To install Drupal 6, you need to have several permissions properly set in your file system. This ensures that Drupal can create the required files and that everything will work correctly the first time you visit your newly installed Drupal site.

For more information see [the documentation for granting permissions in Drupal 6](#).

Drupal 6 also checks for the availability of Javascript during the install process. This check sets a cookie to indicate that Javascript is available for the session. If you turn Javascript off during the installation process or reinstall with Javascript turned off, the installation process will fail. These are session cookies so they can be removed by restarting the web browser or by clearing cookies.

See here [If you can't get past the "Set Up Database" screen](#)

# 403 Permission denied error when testing for clean URLs

So you've tried every advice above but still get the 403, "You don't have permission to access", error when testing for clean urls. This is how I solved it after many tedious hours of googling....

Apparently there's an option called Multiview in apache which enables the server to look for files without an extension in the parent directory of the directory you specified, if it can't find the directory. So if you type [www.mysite.com/admin/settings](http://www.mysite.com/admin/settings) and it can't find the directory it looks for a file named settings.(some extension) in [www.mysite.com/admin](http://www.mysite.com/admin), however if you also have directory browsing turned of and it can't find a file it will report it as you trying to find a file in [www.mysite.com/admin](http://www.mysite.com/admin) which you don't have a permission to view or doesn't exist i.e. the 403 error. To solve this you either turn off directory browsing in your .htaccess file by writing:

```
Options +Indexes
```

instead of

```
Options -Indexes
```

However if you don't feel like having directory browsing on, I guess most people do, you just turn of Multiviews like this under the other options in your .htaccess file:

```
Options -MultiViews
```

If that doesn't help, don't forget to check if there are any physical folders with the same name as your path aliases that you can't access.

Some users have also reported it helpful to specify your \$base\_url in settings.php. You'll find settings.php in 'sites/default'.

## Clean URLs not working? Check your .htaccess file!

Here are some of the most common reasons why your clean URLs may not be working (assuming you're using a normal Linux/Apache hosting environment).

### 1. Your site is in a subdirectory, but you didn't set RewriteBase <#>

If your site URL looks like this:

```
http://example.com/~jsmith1/drupal/
```

Then you need to set RewriteBase! Find the section in your .htaccess file that looks like this:

```
# Modify the RewriteBase if you are using Drupal in a subdirectory and
# the rewrite rules are not working properly.
#RewriteBase /drupal
```

And change the last line, like this:

```
RewriteBase /~jsmith1/drupal
```

Make sure you remove the # symbol in front of RewriteBase, otherwise this will not work.

If you move your site later on, this line will need to be changed again. For example, if you later move it to: `http://your-site-here.com/` then update the line to `RewriteBase /` (or just comment it out with a # in front).

## 2. Your .htaccess file didn't get uploaded at all. <#>

Check if the .htaccess file was actually uploaded. It should be in the directory where you uploaded Drupal (for example: `/public_html/drupal/`)

If the .htaccess file is missing, you need to upload it. If you accidentally deleted this file, just download Drupal again, and copy the new .htaccess file.

Make sure the file is only called `.htaccess` and not `htaccess.txt` or anything else. The `.` period `.at` `.the` `.beginning` is required.

This name *usually means that the file will be invisible on folder listings on Unix-based systems* so you might not always see it. If using an FTP client, you may have to configure it to 'show hidden files'. If listing on the commandline, you must `ls -la` to see it. This will be somewhat dependent on your OS.

## 3. Your .htaccess file is corrupt. <#>

The .htaccess file is very fussy, and only works if things are in a certain order. If you've tried to add password protection or to block image hotlinking (especially if you used cPanel or some other software to set that up), you've probably corrupted the file. The simple solution is to delete your corrupted .htaccess file, then upload the standard version that comes with Drupal.

Evidence of a broken htaccess file can usually be seen in the Apache logs. If you have access on the host, the logs may be at `/var/log/apache2/error.log` or some variation like that. If you are on a shared host, see your ISPs documentation or helpdesk.

## 4. Pager and other query strings don't work <#>

Make sure you have not modified the Apache rewrite rule. The default RewriteRule is:

```
RewriteRule ^(.*)$ index.php?q=$1 [L,QSA]
```

What this rule does is rewrite a path like **example.com/node** into **example.com/index.php?q=node**

However, you'd normally then expect that path like **example.com/node?page=1** would be re-written into **example.com/index.php?q=node?page=1** which would fail since both separators are `?`.

However, the rewrite rule has additional flags. The 'L' flag is irrelevant, but 'QSA' makes the difference. What it does is to preserve any existing arguments after the rewrite, thereby changing the second '?' in this scenario into '&'.

# Disabling modules manually in the database

If you need to manually disable a module you can do so in the database. Before you start, do a mysqldump (or export) of your Drupal db to a local file in case things go wrong and you need to roll back.

## phpMyAdmin

To disable a module via phpMyAdmin you must login to phpMyAdmin. Then you must locate the database where Drupal is installed. Once you have done so look for the "system" table. Click and open it. Click to browse the table. Find the record for the module that is causing the problems and select the edit function. Set the status to "0" and save.

## MySQL command line

To disable a module using the MySQL command line, run the following SELECT to look at the state of your data before the change. This will help you to find the full name of the module too.

```
SELECT name,status FROM system WHERE type='module';
```

Then to disable your module, set the status to 0 for the module name that you want to disable.

```
UPDATE system SET status='0' WHERE name='module_name';
```

Check your handiwork using the SELECT statement again.

# E-Mail from Drupal is bouncing or not being sent

If you are not receiving any E-mails from Drupal, or if E-mail sent by Drupal is bouncing, then ensure that the SMTP configuration is set properly in your php.ini.

If you continue to have problems, the use the "user\_mail\_wrapper" option included with Drupal.

You can now hook up your own custom SMTP library to Drupal instead of using the default PHP mail() function. For more people mail() will work just fine, but for others this is a major problem and it does not work properly. If you just want to get started you will have to download a custom wrapper function from the Drupal contrib repository. If you already have a favorite SMTP function you want to use you will have to create your own wrapper function.

Make an include file that defines a user\_mail\_wrapper function: user\_mail\_wrapper(\$mail, \$subject, \$message, \$header); This function should take the parameters and pass them to the SMTP lib. You will probably have to configure the SMTP lib in some way.

Modify your configuration file (conf.php) to include: \$conf["smtp\_library"] = "path/to/wrapper.inc";

Check out <http://cvs.drupal.org/viewcvs/drupal/contributions/tricks/smtp/> for an example.

Originally written by Kjartan on January 9, 2002, with modifications.

Customize smtp.inc from the repository above to ensure that the proper settings for your SMTP server are being used.

# FTP uploads and file permissions using Transmit

Transmit has the capability to change the permissions of a file as it uploads it to your server. Make sure the check-box "Auto-set my permissions" is unchecked. Typically, Drupal (all versions) like to have most of their files at octal 755 or 775. If you accidentally have Transmit set to change things to 644 or something similar, your newly-uploaded Drupal site will not work.

## HOWTO: Download a fresh copy of a missing or corrupted module/image/file

Thanks to revision control, you can download any file from any version of Drupal from 3.0 onwards by using the ViewCVS server.

1. Go to <http://cvs.drupal.org/> . You should see something looking similar to a folder listing.
2. Click on the *drupal* folder.
3. Click the *Select Branch* dropdown next to *Show files using tag:* and select the appropriate *DRUPAL-X-Y-Z* tag in the *Non-branch tags* section. The file index will change to a view of the files from that release.
4. Browse to the appropriate folder. (Ignore any empty folders. Due to the way CVS stores information, you may come across empty folders that didn't actually exist for your release.)
5. Click the dotted number in the Rev. column for the file you need. This will show a preview of the file.
6. Right click the *download* link and choose *Save Target As...*
7. Make sure the file type is set to *All Files*. Browse to the desired location on your disk and press *Save*.

You should now have a fresh copy of the missing or corrupted file.

## Help! I enabled a buggy module and now I can't disable it!

A situation can arise where you've enabled a module through *Administer > Modules* (Drupal 4.7 and below) or *Administer > Site building > Modules* (Drupal 5.x and above), but then because of a fatal error of some kind cannot get back to the modules administration screen to disable it (sometimes you cannot access any pages of your site at all) -- only the error is displayed and nothing else.

This can be resolved by executing the following query in the database (replace `module_name` with the name of the module you are trying to disable):

```
UPDATE `system` SET `status` = '0' WHERE `name` = 'module_name' LIMIT 1;
```

This will disable the module so that you can access your administration panel again. The module name is the filename (without extension) of the `.module` file. For example, `views` is the module name for `/sites/all/modules/views.module`.

Do not delete the module from the file-system before disabling (and optionally uninstalling it) via the instructions above or through the administration interface. Doing so might leave tables and other objects orphaned, and prevents the module from properly cleaning up after itself.

## How do I unset the clean URLs?

It's possible that a Drupal site that was once configured to run with clean URLs later on stops doing so. After that, almost all content is inaccessible, because the paths being generated are not supported by the hosting platform.

This occasionally happens

- after transferring databases between hosts,
- taking a local copy,
- restoring from a backup (and leaving the .htaccess behind),
- deleting the .htaccess,
- or even from your host (or you) making inadvertant security changes.

The *problem* is that you cannot set it back, because you cannot browse to the specific configuration page anymore.

Drupal checks *once* if clean URLs are supported before allowing you to enable them, then assumes that things will continue working.

**The simplest way to resolve this is by entering the 'unclean' system paths directly.**

If you have locked yourself out, visit `http://example.com/?q=user` just like you do to re-enable offline sites.

You can log in there, enough to become admin, although none of the navigation will work yet.

Next, enter `http://example.com/?q=admin/settings/clean-urls` (in DRUPAL 5 and 6) to view the page where you can unset clean URLs.

Your interface will be usable again.

Other options that should get the same result include:

- Run the mysql commands:  

```
UPDATE variable SET value = 's:1:"0";' WHERE name = 'clean_url';
DELETE FROM cache;
```
- Alternatively, you can modify the appropriate settings.php file to include the line  

```
$conf['clean_url'] = 0;
```

at the bottom (or similar code in the site settings array you'll see there).

## How to Resolve module-theme name collision

When developing a site it can happen that a module and theme are being given the same name. This results in very weird behavior: overriding hooks can make your blocks disappear, print \$content returns only 'array', hence disappearing admin interfaces which force us to alter the database to get back on track.

Now we have to find out how to remove the new theme and switch back to an old one. if your site is big, with lots of content, then reinstalling is not an option. I tested this with MySQL, so these hints are for MySQL only.

Make a backup of your database first. We are going to edit the dumped records, and reinsert the database. I used this method because using my favorite editor is way easier and safer than using MySQL shell (anyway we shouldn't edit a live site database).

The theme name is registered in the following tables in your database: *system*, *variables* and *block*. Clear the cache manually, meaning to remove all the **INSERT INTO `cache\_`** lines in order to see

results. Change the 'default\_theme' value in `system`. Note that 's' stands for string length, so be sure you put the string length after the ':'. Also make sure you change the value to an existing theme (available themes are listed in `system`). And remove, or rename your troublesome theme.

The change in `block` is imperative, otherwise the blocks will not appear. Drop your old database from the sql server and insert the edited one. Voila, we have recovered from a theme crash.

## Never give a module and a theme the same name

It leads to undesirable behavior of your webserver and is hard to recover from. Use `clientname.module` and `/themes/clientname_2007/` as themes are more change-sensitive, while client-specific code just layers up.

And remember, don't start a module or theme name with a number, and use underscores (\_) not dashes (-) in names.

## How to stop Drupal taking over my subdirectories

By default, *if a file exists* you can access it directly via an accurate URL, without Drupal even getting involved (this is due to a directive in the .htaccess rewrite)

However, also due to a directive in .htaccess, the default Apache raw directory browsing and 'index.htm' resolving behaviour is turned off - to protect your files from unintended access. Just entering a directory path as a link will not work (Access denied).

See [Configuring .htaccess to ignore specific subfolders](#) if you want to:

- Provide raw directory browse access to areas
- Run static HTML pages alongside (within) a Drupal installation
- Run other web applications (like a stand-alone gallery or forum version)
- Access a host-provided subdirectory or virtual subdirectory (eg /stats or /webmail)

## IP address and referrer blocking

You may find your site being crawled by bad bots, or otherwise find a need to block IP addresses, host names and referrers. This section outlines some common methods to achieve this.

Much of the information presented here is not specific to Drupal, however it provides a quick reference from which you can do more in-depth research if necessary.

### IIS

For those of you using IIS, I have written a guide on how to block referrer spam using ISAPI Rewrite here;

[Blocking referrer spam on IIS with ISAPI Rewrite](#)

-----

# .htaccess sample list plus domain blocking

As described you can block via the use of .htaccess. Here is a sample list to stop trackback spam (or splogs) from occurring in the first place.

Here is a list already made available to use, which blocks several spams.

```
# Single word blocks
RewriteEngine On
RewriteCond %{HTTP_REFERER} poker [OR]
RewriteCond %{HTTP_REFERER} medicine [NC,OR]
RewriteCond %{HTTP_REFERER} pills [NC,OR]
RewriteCond %{HTTP_REFERER} diet [NC,OR]
RewriteCond %{HTTP_REFERER} viagra [NC,OR]
RewriteCond %{HTTP_REFERER} mortgage [NC,OR]
RewriteCond %{HTTP_REFERER} casino [NC,OR]
RewriteCond %{HTTP_REFERER} insurance [NC,OR]
RewriteCond %{HTTP_REFERER} loan [NC,OR]
RewriteCond %{HTTP_REFERER} buy [NC,OR]
RewriteCond %{HTTP_REFERER} xanax [NC,OR]
RewriteCond %{HTTP_REFERER} meridia [NC,OR]
RewriteCond %{HTTP_REFERER} incest [NC,OR]
RewriteCond %{HTTP_REFERER} lesbian [NC,OR]
RewriteCond %{HTTP_REFERER} viagra [NC,OR]
RewriteCond %{HTTP_REFERER} adult [NC,OR]
RewriteCond %{HTTP_REFERER} hentai [NC,OR]
RewriteCond %{HTTP_REFERER} tramadol [NC,OR]
RewriteCond %{HTTP_REFERER} phentermine [NC,OR]
RewriteCond %{HTTP_REFERER} gambling [NC,OR]
RewriteCond %{HTTP_REFERER} texas- [NC,OR]
RewriteCond %{HTTP_REFERER} holdem [NC,OR]
RewriteCond %{HTTP_REFERER} pharmacy [NC,OR]
RewriteCond %{HTTP_REFERER} ultram [NC,OR]
RewriteCond %{HTTP_REFERER} tramadol [NC]
RewriteRule .* - [F,L]
```

Remember to use [ and ] to close the arguments above.

read more on:

[http://www.i-marco.nl/weblog/archive/2005/08/29/saving\\_some\\_valuable\\_ban...](http://www.i-marco.nl/weblog/archive/2005/08/29/saving_some_valuable_ban...)

Or another way is to block domains also in the .htaccess file:

```
SetEnvIfNoCase Referer ".*baddomain.com" BadReferrer
SetEnvIfNoCase Referer ".*anotherbaddomain.com" BadReferrer
```

```
order deny,allow
deny from env=BadReferrer
```

read more on:

<http://www.hojohnlee.com/hacks/2006/01/12/blocking-spam-domain-referrals...>

## Overview of referrer blocking

In some content management systems, you can configure a block that shows what links people clicked to come to your site. This is called a referrer sting. Drupal doesn't display this referrer, which is a good thing.

Many porno and online poker sites have robots that are sending "fake" referrer strings, claiming they entered your site via their own site. When you would have a block displaying these referrers, they would get more hits from people clicking these displayed referrer URL's. So it is a good thing that Drupal doesn't display these referrers, apart from the referrer logs admin page.

Often, the robots sending these referrers do this via a so-called zombienetwork, thousands and thousands of misconfigured or hacked PC's which are "open proxys". You can't really block this referrer spam, the robots are sending real user-agents, are using many URL's and there are so many zombie IP addresses that are changing fast, that you can't block these.

There is some work underway to block bad behaving robots and users from within Drupal. Until that time, the only way to block the referrer spam it by looking at your referrer log and look for often used word in the fake referrer URL's, like "online-poker".

Now you can block these words in your .htaccess file. Say you want to block the referrers "internet-poker" and "viagra" as well as all user-agents that contain "looksmart"

First locate your .htaccess. This file is most likely in your Drupal document root. You can use vi, pico, notepad or another editor. Now go to the end of the file and just before the last line add:

```
# Block referrer spam
RewriteCond %{HTTP_REFERER} (viagra) [NC,OR]
RewriteCond %{HTTP_REFERER} (internet-casino) [NC,OR]
RewriteCond %{HTTP_USER_AGENT} (Looksmart) [NC]
RewriteRule .* - [F]
```

You don't have to restart your webserver, these settings take place immediately. When you look at your logs, you will still see the spamming robots with the fake referrer URL's. But you will see that these clients now get a 404 error, this means that they are not allowed to access that (or any other) page.

If the robot that is sending this referrer spam is a "smart" robot, it will know sending the fake URL didn't work. Now it wont stop all the bad guys, they will probably try to send another URL. Or the will go to another site to spam there. But there is a chance you will make it a better world. Try it.

## Installing Drupal (6.x) Database setup fails to proceed. No error.

### Symptom:

You are on step 3 of install.php, and the system wants your database name and password. You enter it, you are **sure** it's right and that the database exists etc. But on submission the page just refreshes. No Error, no progress. Please enter your database information.

### Possible Cause 1:

You mis-read the instructions for creating settings.php.

**Do not** *rename* **default.settings.php into settings.php**  
Copy it.

The original `sites/default/default.settings.php` must still be present alongside your new one.

## INSTALL.txt

You must create a file named `settings.php`. You may do so by making a **copy** of `default.settings.php` (or create an empty file with this name in the same directory). For example, (from the installation directory) make a copy of the `default.settings.php` file with the command:

```
cp sites/default/default.settings.php sites/default/settings.php
```

This is an acknowledged User Interface problem with this version of Drupal, and [will be fixed](#) (at least with a better warning) in a later release.

## Possible Cause 2

Check if `settings.php` is really writable (especially on Windows)

There are two possible reasons for this problem.

- a) As described, `default.settings.php` was deleted or renamed
- b) `settings.php` is not writable.

`Settings.php` is not writable might happen on Windows if advanced security is activated. Drupal reports that the file is writable but this is not true. If this is the case, read <http://www.ekhoury.com/2007/01/15/permission-denied-php-on-iis/> or if you cant change this yourself, contact your hosting provider.

## Search hints

To assist users who are *still* encountering this issue regularly, Here are some search phrases that might guide you to this page. Maybe if they *search first* they may find themselves here.

- Can't get past database configuration
- Unable to get past database configuration
- Stuck at database configuration
- Won't set up database
- installation stalled entering database information.
- Cannot get past database configuration
- "Set up database" screen, nothing happens

# Issues with rebuild content permissions after OG installation

Hi,

if you can not rebuild your content permissions (rebuild does not finish or finishes with error) after installing OG access control module (part of the OG bundle) do this:

1. run this command on your sql server:  
delete from node\_access;  
insert into node\_access values (0,0,'all',1,0,0);

2. turn off the OG access control module

Well, you'll not able to use this module but at least node permissions are not messed up and your site is not corrupted.

-----

## Update: May 29, 2009 by AlphaGeekBoy

In researching this issue and digging deeper, I found that I had two left-over content items from a content type that I had previously deleted. This was preventing the content permissions rebuild from rebuilding. Once, I removed those two nodes from the node and node\_revisions table, I was able to leave OG Access control turned on and run rebuild permissions with no problem.

## Warning

Content without a title can also cause this. Construct a view to check all your content for missing UID (UID=0), missing title, or missing type. There may be other things that can cause this, but those will definitely do it.

# Modules showing old information

When upgrading a module, the new version may be looking back at old information stored in the variables table. Take a look at the variables table and see if you can find the old information.

You can remove the row from the variables table - it will use the default data if the row is not found.

# MySQL 4.0 -> 4.1 migration

The MySQL version 4.1 introduced some changes to character set handling, which are often giving us trouble if moving a Drupal site from one to the other (i.e. from 4.0 or below, to 4.1 or above). Basically, Drupal always uses utf-8 encoding, but MySQL 4.0 and lower doesn't support it, so utf-8 data are treated as default Latin 1 inside the MySQL 4.0 engine. It doesn't matter there, as MySQL output echoes the same data as sent in still, so Drupal works fine, but it matters a lot in the migration process, where MySQL might attempt to "convert" the data from Latin 1 to utf-8, not knowing that we have utf-8 in there already, and so the site gets broken then.

There are a few typical symptoms, and typical scenarios, (all discussed later in more detail):

- **"Garbled text"**: Any special characters (non-english, advanced punctuation and the like) in your site's contents turned into seemingly random pairs of weird characters after moving the site to 4.1. (Typically, "ÃÃ" are frequently seen, always followed by some other non-english character).
- **"Question marks"**: Your contents is more or less fine, but some special characters (non-english, advanced punctuation and the like) are replaced by question-marks after the move to 4.1.
- **"Illegal mix of collations"**: Your site worked after the move to 4.1 somehow, but later crashed badly (typically after running update.php on the next upgrade, installing a new module, or the like). Missing sidebars, virtually any part of the pages vanished, no way to access even administrative pages - all that is possible. Looking into the logs (manually in the 'watchdog' table in worst case), you'll see error messages "Illegal mix of collations".

**IMPORTANT:** All these symptoms mean, that your database is NOT healthy after the move. Don't try to run your site in such a state: Any new data stored to the database are only making things worse! **You need to fix your database during the migration process, or immediately after, before going live again!** Keep backups! If you did a wrong step, you'll most probably need to restore your database from backup, and try again; it's usually easier and more safe, than fixing broken data afterwards. **If you don't have any backup, create one now!**

Note that basic English characters are the same across all encodings, so your English contents (as well as Drupal's internal data) is not affected, and you might not even notice that there's a problem, if running English site. But still, some things may be broken.

## 1: UPGRADED SERVER TO 4.1, WITHOUT REALLY MOVING DATA

If you (or your hosting provider) just upgraded MySQL to 4.1+, without really moving the data to another machine or another database, you've correct utf-8 data already in your database, but most probably you've wrong encoding information (for details, see **(B)** in the table below). You see the "Garbled text" problem, and your tables are "latin1\_swedish\_ci" in the database.

You have two options:

**A. Alter encoding/collation directly in database.** This must be done in a tricky way (changing columns to binary types and back), to avoid real conversion of stored data. The operation, however simple for the server, is not so easy to do manually, so a specialized helper-script was developed for this purpose.

Just [grab the file mysqlfix.php from the sandbox](#), put it into your Drupal root directory (the one where update.php is), and point your browser there (i.e. something like <http://www.example.com/mysqlfix.php>). Then just follow on-screen instructions. The script will offer you a way to choose tables (both Drupal and/or other tables in your database), and process them (including large databases in batch processing). It's compatible with Drupal versions 4.7.x, 5.x and 6.x, supports prefixed tables, and uses the same access checks as update.php does. Note that if you're not logged in as user 1 yet, it might be safer to edit the access check flag in file, rather than attempting to browse the broken site to log in (any data written to the broken database is a risky business).

Although this script is not really dangerous (theoretically it doesn't hurt to process a table twice), you should always take the common safety precautions while doing this sort of work:

- Before starting, always create a backup copy of your database (even if you're backing up a broken site)
- Give appropriate attention to security: If accessing the script with access check disabled setting, restore the original setting immediately after use.
- Remove any single-use helper scripts (like this one) from your server, if no more needed.
- Use on your own risk.

**B. Export and re-import.** This option is more complicated and resources-expensive for the server, and depends on your permissions on the system, but still it's an alternative solution (especially if you can't run the helper script above for some reason, and/or you're really familiar with command line). It's basically the same process, as migration between two different servers:

First, export your database (from the newly upgraded 4.1 server) in backwards-compatible mode:

```
mysqldump --user=username --password=password --default-character-set=latin1 --
compatible=mysql40 dbname > dump.sql
```

Then, empty your database, by removing all the tables, and change it's default encoding/collation to "utf8\_general\_ci":

```
mysql --user=username --password=password --execute="DROP TABLE table_name,
table_name..... ; ALTER DATABASE db_name DEFAULT CHARACTER SET utf8 COLLATE
utf8_general_ci;"
```

Since specifying all the tables may be annoying, better create a brand new database instead (if you've permissions):

```
mysql --user=username --password=password --
```

```
execute="DROP DATABASE dbname; CREATE DATABASE dbname CHARACTER SET utf8 COLLATE
utf8_general_ci;" Both these ways should result in an empty database with default
encoding/collation "utf8_general_ci".
```

```
Finally, import the file back into the database: mysql --user=username --password=password --
default-character-set=utf8 dbname < dump.sql
```

Now, your database should be perfectly OK, "utf8\_general\_ci" set on all tables, and Drupal working fine. You're done.

## 2. MOVING DATA FROM 4.0 SERVER TO (OTHER) 4.1 SERVER

If you're migrating to a different server (which may as well be localhost for testing, or the like), you need to export the data from the old 4.0 database, and import to 4.1 along with correct utf-8 encoding information, so that the database engine understand the data correctly and doesn't do some unwanted "conversion". This is important; if you imported your file in a wrong way, your data got broken, and you'll probably need to try again from scratch, rather than fixing the broken data (which is always a bit unsafe).

On the old 4.0 server, export your database:

```
mysqldump --user=username --password=password --default-character-set=latin1 dbname >
dump.sql
```

On the new 4.1 server, create the new database with "utf8\_general\_ci":

```
mysql --user=username --password=password --execute="CREATE DATABASE dbname CHARACTER SET
utf8 COLLATE utf8_general_ci;"
```

OR alternatively alter your existing database:

```
mysql --user=username --password=password --execute="ALTER DATABASE db_name DEFAULT
CHARACTER SET utf8 COLLATE utf8_general_ci;"
```

If the 4.1 database already was "utf8\_general\_ci", you may proceed happily to the next step; you needn't to do any of these two options then. If the default was utf8, but some other national collation, and you haven't permissions to alter the database settings, you may proceed too, and alter all the imported tables and columns afterwards manually, to be "utf8\_general\_ci". But if you've the database default encoding other than utf8 (and no permissions to alter), your best chance is probably to try and hack your dump.sql file, adding `/*!40100 DEFAULT CHARACTER SET UTF8 */` at the end of every single `CREATE TABLE` statement, just before the final semicolon (tedious work with an utf-8 compatible text editor). If the database is latin1, you may also import with `--default-character-set=latin1`, and then proceed to the 1.A scenario above, to fix the rest.

Import the file to your new utf8 database:

```
mysql --user=username --password=password --default-character-set=utf8 dbname < dump.sql
```

Now, your database should be perfectly OK, "utf8\_general\_ci" set on all tables, and Drupal working fine. You're done.

## 3. MOVING DATA FROM 4.1 SERVER TO 4.0

This is probably rare - it might only happen, if you're moving the site to a local server for testing, having the local server not upgraded yet. It's rather simple: Export in backwards-compatible mode, then import to 4.0:

From 4.1:

```
mysqldump --user=username --password=password --default-character-set=utf8 --
compatible=mysql40 dbname > dump.sql
```

To 4.0:

```
mysql --user=username --password=password dbname < dump.sql
```

## 4. ENCODING/COLLATION SETTINGS

MySQL keeps track of your data's nature by using an encoding/collation information on various places. While collation (the part with various nation-names in it) is only affecting some operations like sorting and comparing, and so may be changed safely, encoding (utf8 or latin1) is addressing the nature of your data, and so needs to be always correct:

- **Table columns:** The encoding here describes your data directly. This is the most important bit for your database health; any change done to single columns is likely to perform a real conversion of the data, so better don't try to alter encoding here. (The script in scenario 1.A above is a way to change encoding without really changing data.)

- **Tables:** The info on tables is used as a default for any new columns created inside the table. You may change this to utf8 with queries like `ALTER TABLE table_name CHARACTER SET utf8 COLLATE utf8_general_ci;`

- **Database:** The info on database is used as a default for any new tables created inside the database. This is important for imports from 4.0 dumps, as these contain no information about tables encoding/collation. You may change the database defaults to utf8 with the query `ALTER DATABASE db_name DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;`, if you have permissions for that.

- **Connection:** This is the encoding of data coming in/out of MySQL engine, i.e. data processed by Drupal and shown on your screen. It's specified through the `--default-character-set=utf8` switch for each new connection (or with `SET NAMES utf8` query, as seen in Drupal itself, as well as \*some\* MySQL 4.1 dumps). Default is latin1, which we don't want.

If the connection is different from the table column's encoding, MySQL performs a conversion, attempting to present more or less correct data. Being that the case, you may encounter the "Garbled" or "Question marks" problems, either on data read from OR stored to the database, or at the very least a bit worse performance. This is where all the damage happens on migration.

Generally, whichever way you choose to manipulate your data, **you need all these four settings to be "utf8" and "utf8\_general\_ci"**, or at the very least you need to have all table columns as "utf8\_general\_ci".

Having table columns on latin1 means conversion (bad performance, and question-marks problem), and having table columns different from each other means "Illegal mix of collations" (MySQL can't perform certain operations (joins) between different collations).

When Drupal adds a new table to the database (for example during update.php run, or new module installation), "utf8\_general\_ci" is always enforced, so you needn't to worry about the defaults of your database. But however, some other outside-of-Drupal tools/situations - especially imports needed for migrations and backup-restoring, or manually performed queries - may fail to enforce that, resulting in the database defaults being used for new tables. That's why we want to have the database default set to "utf8\_general\_ci" - or else double check all newly created/imported tables, and change them to "utf8\_general\_ci" immediately.

## 5. ALTERNATIVE WAYS TO MANIPULATE YOUR DATABASE

Sometimes it happens, that you can't use the shown command-line examples, because you're on a different system, you've not enough permissions, or you have no access to the command line at all. If your hosting provider don't let you in, you may install some of existing web-based applications, such as for example the popular [PhpMyAdmin](#), or for large databases some other ones, like [MySQL Dumper](#), which is able to handle large data avoiding php timeout problems. You may then execute your operations using these applications - sometimes it's even more comfortable.

**Make sure that such applications are always protected from unauthorized access! If there's no other option (MySQL Dumper), consider at least [basic HTTP authentication](#).**

This page can't describe all the options of all applications in detail - please refer to corresponding manuals for that. As for the examples provided here - for dumping/importing data, you need to use the corresponding functionality provided, while other queries may be usually run directly:

For example, from this command:

```
mysql --user=username --password=password --execute="ALTER DATABASE db_name DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;"
```

You need to take the actual query:

```
ALTER DATABASE db_name DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;
```

And just paste it to the "SQL" input of PhpMyAdmin (or similar).

There's one important bit, though: **The connection encoding.** To emulate the `--default-character-set=utf8` switch, it's necessary to execute `SET NAMES utf8` query on every page load. Some applications do this, some not. PhpMyAdmin lets you choose the connection, while MySQL Dumper (speaking of version 1.21-b6) needs to be hacked a bit to import your MySQL 4.0 dump well: Add a line `$res=MSD_query('SET NAMES utf8');` just above final `return;` of function `MSD_mysql_connect()` in the file `mysqldumper/inc/mysql.php`.

## 6. WHAT ELSE MAY HAPPEN, AND HOW TO FIX IT

Depending on how the migration was done (attempted), you may find yourself in four different states:

		Import done with the switch <code>--default-character-set=utf8</code> or <code>SET NAMES utf8</code> query	Import done without encoding specification, or with <code>latin1</code>
Database default	File from MySQL 4.0 (encoding info not included)	(A)	(B) (Also happens after MySQL upgrade without really moving data)
Database default latin1	File from MySQL 4.1 (with encoding info)	(C) (You want to fix database default collation too )	(D) (You want to fix datab. default collation too)
Database default utf8	No matter which file	(C)	(D)

**(A)** - You have "latin1\_swedish\_ci" on your tables and columns, your contents looks more or less correct, but some special characters are replaced by question-marks. Currently, your data are stored in Latin 1 encoding, which can't handle all the characters, and requires MySQL to perform a conversion on each query.

It's probably best to use your backup now, and start the migration over again.

If you have no backup to revert to, you may try to execute the following query for each of your Drupal tables:

```
ALTER TABLE table_name CONVERT TO CHARACTER SET utf8 COLLATE utf8_general_ci;
```

This will convert your data and tables back to correct utf-8 state (including any data submitted to the site in the broken state, even), but however, the question-marks in your contents will stay. This is caused by latin1 being incapable to store full range of utf-8 characters, so the damage is permanent, and **this way should not be considered safe.**

You still want to correct the database defaults, to avoid running into the "Illegal mix of collations" error later (see paragraph 4.):

```
ALTER DATABASE db_name DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci
```

**(B)** - You have "latin1\_swedish\_ci" on your tables and columns, and your contents looks "Garbled". But still, your data are stored as correct utf-8 and no damage done yet! You have only just wrong encoding information in your database (causing MySQL to break your data on the way out).

The fix is not as simple as executing a query or two. Please read the scenario 1. above for details, or revert to a backup if this is a result of unsuccessful import just done. The fix is safe, however, unless some new data got submitted in garbled state.

It's also a good idea to fix the database defaults (see above).

**(C)** - Congratulations! You have "utf8\_general\_ci" on your tables and columns, correct data inside, and Drupal working well. Unless you need to correct the database defaults, you're done already, and completely safe.

**(D)** - You have "utf8\_general\_ci" on your tables and columns, but your contents is "Garbled". Unfortunately, that's really how your data are stored in this case (after double utf-8 encoding, all two-byte characters turned into four bytes, and so into pairs of weird characters, one per original byte).

It's probably best to use your backup now, and start the migration over again.

If you have no backup to revert to, you may try to convert your data back with the following query for each of your Drupal tables:

```
ALTER TABLE table_name CONVERT TO CHARACTER SET latin1 COLLATE latin1_swedish_ci;
```

If everything went well, this took you back to the (B) state, which is still wrong, but allows you to proceed to an other fix at least. Now you may perform the steps described in scenario 1. above, to fix the remaining problem. But since your data undergone double conversion, **this way should not be considered safe.**

Depending on the circumstances, you might still need to correct the database defaults.

## 7. LINKS

See also the [official MySQL page](#).

More discussion may be also found on the [original issue](#) and various pages linked from there.

# Random core functions come up as undefined

You might in your Drupal travels come across the situation where doing seemingly innocuous things such as logging in or trying to view a page results in errors such as:

```
Fatal error: Call to undefined function taxonomy_get_vocabularies() in /modules/tagadelic/tagadelic.module on line 41
```

or:

```
Fatal error: Call to undefined function: path_set_alias() in /modules/pathauto/pathauto.module on line 435
```

Of course, the very first debugging step should be checking the `system` table and making sure that the `status` column is set to **1** for `path.module`, `taxonomy.module`, etc. If those modules somehow got disabled without your consent, this would cause the problem.

However, if you've eliminated this step and still are running into the issue, the next step is to take a look at the filename of these modules. Here's an excerpt from my system table:

```
mysql> select name, filename, status from system where type = 'module';
+-----+-----+-----+
| name      | filename                                     | status |
+-----+-----+-----+
| taxonomy  | modules/taxonomy/taxonomy.module           | 1      |
| path      | modules/path/path.module                   | 1      |
| pathauto  | modules/pathauto/pathauto.module           | 1      |
| tagadelic | modules/tagadelic/tagadelic.module         | 1      |
+-----+-----+-----+
```

All looks good, right? *Except*, this is a 4.7.x site (which died partially through a 5.x upgrade), and the path to the taxonomy and path modules are in fact `modules/taxonomy.module` and `modules/path.module`, **not** `modules/taxonomy/taxonomy.module` and `modules/path/path.module`. `module_list()` can't find them, and thus it never loads the files, so random fatal errors abound.

To solve the problem, try visiting the modules page (if you can), which will re-build the system tables with the correct filesystem paths. If you can't (for example, if Drupal won't even let you log in), then copy/paste the following into a file called "fix.php" in the root of your Drupal installation (same place as `INSTALL.txt` and `friends`) and run it in the web browser:

```
<?php
// Bootstrap Drupal.
include_once "includes/bootstrap.inc";
drupal_bootstrap(DRUPAL_BOOTSTRAP_FULL);

// Force re-load of the system table.
system_modules();
?>
```

If you get the error "**Fatal error: Call to undefined function system\_modules**" then it means that even your `system.module`'s path is incorrect in the system table. Manually set the filename of `system.module` (normally "`modules/system/system.module`") and run `fix.php` again. You can manually set the filename in `mysql` with:

```
update system set filename="modules/system/system.module" where name="system"
```

Then run `fix.php` again, and it should succeed.

Now, when you reload the page, everything should be back to normal (although you may want to restore from backup just in case).

## Receiving errors when trying to use mysqli extension

PHP 5.0.5 does not support the `mysqli` extension, which is the default extension used by Drupal for `mysql` database access. A work-around for this issue is to modify `$db_url` in `settings.php` and specify `mysql://` rather than `mysqli://`.

You are a victim of this bug if you see the following types of errors:

```
Warning: Couldn't fetch mysqli in /home/thengthi/public_html/includes/database.mysql.inc on
line 329
Warning: Couldn't fetch mysqli in /home/thengthi/public_html/includes/database.mysql.inc on
line 329
Warning: Couldn't fetch mysqli in /home/thengthi/public_html/includes/database.mysql.inc on
line 329
Warning: Couldn't fetch mysqli in /home/thengthi/public_html/includes/database.mysql.inc on
line 329
```

# Relay SMTP mail to external mail server using smtp.class

Note that there is now also a SMTP module that will do this for you: <http://drupal.org/project/smtp>

The following are steps that I used to get Drupal 4.6.x running on Apache with PHP 4.3.10-15 to successfully relay mail to an external mail gateway without using the `php_mail()` function. I have tested this on Postfix and Exchange server mail gateways successfully.

1. Download `class.smtp.inc` from <http://www.phpguru.org/downloads/smtp/smtp.class-1.0.5.tar.gz>.
2. Untar using `tar -xzvf` and copy `class.smtp.inc` to the `/includes` directory.
3. Edit the file and change the following lines to coincide with your external mail gateway:
  - **`$this->host = 'localhost'`**; Change localhost to the IP address of your mail gateway. You may also use the FQDN instead of the IP address if you have DNS set up correctly to something like `yourmailgateway.yourdomain.com`.
  - **`$this->helo = 'localhost'`**; Change localhost to reflect the hostname (FQDN) of your external mail gateway. For example: `yourmailgateway.yourdomain.com`.
  - If your gateway requires SMTP authentication to relay mail, then you will need to modify the following lines to appropriately reflect the authentication credentials. (If you manage your own mail gateway or it resides within local subnets behind your firewall, then you should be able to leave authentication set to `FALSE` and allow relaying based upon IP address/network ID. This will probably only affect you if you use a hosting company for email management.

```

$this->auth = FALSE; (change to TRUE)
$this->user = ''; (put username here(you may need to precede with domain; e.g. domain\username))
$this->pass = ''; (put password here)

```

4. Download `smtp.inc` from <http://cvs.drupal.org/viewcvs/drupal/contributions/tricks/smtp/> and place it in your `/includes` directory. No modification to this file is needed.
5. Add the following line to the `sites/default/settings.php` file before the closing `?>` tag:
  - **`$conf["smtp_library"] = "includes/smtp.inc";`**

6. Finally be sure to add the IP address of the mail gateway to the **[mail function]** section in your `php.ini` file and restart your web server.

Good Luck! I hope this saves you the hours of researching it took me to get it running.

## Setting up Clean URLs

For a fuller step-by-step, you should see [Instructions for setting up Clean Urls](#) in the handbook, which includes distro-specific tips.

To avoid duplication, this troubleshooting page is reduced to a checklist.

Clean URLs must be enabled within the admin area. If the Clean URL setting is grayed out, you need:

- `mod_rewrite` enabled in the Apache conf file
- `AllowOverride` enabled in the Apache conf or `vhosts` file (so that `.htaccess` rules can apply)
- The `.htaccess` filename specified in the Apache conf file

- An `.htaccess` file in the root of the Drupal directory (may be hidden on some filesystems, but should be there)
- Rewrite instructions in `.htaccess` (distributed with Drupal, but sometimes needing tweaking)

Depending on your distribution, some or most of these directives should already be there, probably scattered throughout the file, or even subsidiary 'vhost' conf files. You may have to consult [the Apache docs](#) or find help specific to your distribution, as these files may vary a lot.

## To test your settings, run

```
apache2ctl -t
```

or similar for a syntax check. (XAMMP?)

## To Lastly reload your http server

```
apache2ctl restart
```

(or similar. XAMMP has a shortcut for you)

## Note

It's `apache2ctl` on some Apache installations, on others you should use `apachectl -t`. Also, restarting the server requires administrator privileges, like so:

```
sudo apachectl restart
```

# Taxonomy term IDs deleted during upgrade

Currently (September 13th, 2009) there is a problem if you have more than 39 taxonomy defined when upgrading from Drupal 5.19 to Drupal 6.13. All defined tid's above 39 in the `term_node` table are deleted in the update process.

Here's how to to fix it manually:

<http://drupal.org/node/569534#comment-2018056>

## phpinfo()

On occasion it is useful to see what your php settings are. You can create a php page using the `phpinfo()` function from within your own Drupal install, if you can create php type content. This is probably not a page you would want to leave permanently, but it can be useful for finding specific information.

<http://us3.php.net/phpinfo>

Create a page of php content type and copy and paste the following line into it:

```
<?php
phpinfo();
?>
```

It will probably mess up your site's formatting.

## Consider Devel Module

If you find yourself doing this sort of thing often, consider using the [Devel module](#) during the development phase.

The devel block includes a `phpinfo()` menu item. Plus there is a heap of other useful things that you'll use from time to time.

## Other Useful Commands

Some example commands that tell you what your machine (server) has so that we can know if our server is up to the requirements.

For example, on linux , after you create a `phpinfo` file with a name of `YYY.php`, you then run `wget` on the file like so: `"wget http://yoursite/directory/where/thefile/is/YYY.php"` .

You can follow this with `"grep -i version"` on the results of the `wget` file `ZZZ`. That will show you all the versions you have.

You can also run `"grep -i memory_limit ZZZ"` on that to find the memory setting.

Other helpful commands:

To see mysql version: `mysql --version`

To see machine memory: `top -b -n 1 | grep Mem`

To see disk space: `df -h`

To see apache version: `/usr/sbin/httpd -v`

This is just a quick reference of some commands that help you with a Drupal install or upgrade.

## Translation Issues

When translating a site to another language, several problems can occur. If you think you found a bug in either a translation or in Drupal's locale module, please file bug reports against the project in question.

If you have a more general question you can ask it in the [translations forum](#).

Here we collect some of the more common issues found.

See also [Translation, multilanguage content, and internationalization](#).

## Weird characters or question marks

Symptom: After importing a translation you find all kind of weird characters or question marks on your site.

Solution 1: The translator did not use UTF-8. Drupal is fully UTF-8 aware and expects translations to be supplied in that character set as well.

You can change the charset of a `.po` file using GNU `msgconv`. Or with XEmacs you can use this command:

```
C-x <RET> f utf-8 <RET>
```

Please also file a bug against the translation in question asking them to change the character set to UTF-8.

Solution 2: You do not have the correct font installed to display the language in question, this is an issue wit your browser or operating system.

If you want to check the character set of a .po file, you can use the "file" command.

```
file *.po
```

Example output:

```
ca.po:      UTF-8 Unicode PO (gettext message catalogue) text
de.po:      UTF-8 Unicode PO (gettext message catalogue) text, with very long lines
fr.po:      UTF-8 Unicode PO (gettext message catalogue) text
it.po:      UTF-8 Unicode PO (gettext message catalogue) text
```

## Some strings do not translate

Symptom: After importing a translation, some strings on your site are translated, some are not.

Possible causes (and solutions):

1. The imported translation is [not complete](#). Consider finishing it and contributing your update by filing an issue against the translation.
2. The actual source strings have been modified after the translation file was created. You might need to update the PO file from a fresh POT file (translation template). The latest core POT files are always [available for download](#), but you might need to [generate fresh POT files](#) yourself.
3. There are end-of-line inconsistencies between the actual source strings and the source strings from the PO file. All Drupal files have Unix-style end-of-lines (`\n`), but these sometimes get messed up on Windows (which normally uses `\r\n`). WinZip users beware: Make sure the "TAR file smart CR/LF conversion" option is **not** checked before extracting any Drupal tarball.