

Core - Bug # 15292

Status:	Closed	Priority:	Should have
Author:	old_pss7	Category:	Install Tool
Created:	2005-12-17	Assigned To:	Sebastian Kurfuerst
Updated:	2006-08-12	Due date:	
TYPO3 Version:	3.8.1		
PHP Version:	4		
Complexity:			
Is Regression:			
Sprint Focus:			
Subject:	login to the BE is not possible on default debian 3.1 systems		
Description	<p>apache2 Server version: Apache/2.0.55 PHP 5.1.1-1 -su-php mysql 4.1.15</p> <p>typo3 BE login always rejects.</p> <p>table syslog: Login-attempt from %s (%s), username '%s' not found!!</p> <p>this issue is only since 3.6 (issue imported from #M2055)</p>		
Related issues:	related to Core - Bug # 15295: Unable to login into BE when using MySQL 5+ Closed 2005-12-19		

History

#1 - 2005-12-17 20:59 - Sebastian Kurfuerst

Please create new BE admin users in the install tool!

Greetings, Sebastian

#2 - 2005-12-17 20:59 - Sebastian Kurfuerst

btw I think that's no bug...

Greetings, Sebastian

#3 - 2005-12-17 21:19 - old_pss7

adding an extra admin user via install script does not help, the same error shows up

br,

Patrick

#4 - 2005-12-18 15:02 - Thomas Peterson

I have tested and on my server it works.

can you post the php.ini ?

#5 - 2005-12-18 23:57 - old_pss7

[PHP]

```
.....
; WARNING ;
.....
; This is the default settings file for new PHP installations.
; By default, PHP installs itself with a configuration suitable for
; development purposes, and NOT for production purposes.
; For several security-oriented considerations that should be taken
; before going online with your site, please consult php.ini-recommended
; and http://php.net/manual/en/security.php.

.....
; About php.ini ;
.....
; This file controls many aspects of PHP's behavior. In order for PHP to
; read it, it must be named 'php.ini'. PHP looks for it in the current
; working directory, in the path designated by the environment variable
; PHPRC, and in the path that was defined in compile time (in that order).
; Under Windows, the compile-time path is the Windows directory. The
; path in which the php.ini file is looked for can be overridden using
; the -c argument in command line mode.
;
; The syntax of the file is extremely simple. Whitespace and Lines
; beginning with a semicolon are silently ignored (as you probably guessed).
; Section headers (e.g. [Foo]) are also silently ignored, even though
; they might mean something in the future.
;
; Directives are specified using the following syntax:
; directive = value
; Directive names are case sensitive - foo=bar is different from FOO=bar.
;
; The value can be a string, a number, a PHP constant (e.g. E_ALL or M_PI), one
; of the INI constants (On, Off, True, False, Yes, No and None) or an expression
; (e.g. E_ALL & ~E_NOTICE), or a quoted string ("foo").
;
; Expressions in the INI file are limited to bitwise operators and parentheses:
; | bitwise OR
; & bitwise AND
; ~ bitwise NOT
; ! boolean NOT
;
; Boolean flags can be turned on using the values 1, On, True or Yes.
; They can be turned off using the values 0, Off, False or No.
;
; An empty string can be denoted by simply not writing anything after the equal
```

```

; sign, or by using the None keyword:
;
; foo =     ; sets foo to an empty string
; foo = none ; sets foo to an empty string
; foo = "none" ; sets foo to the string 'none'
;
; If you use constants in your value, and these constants belong to a
; dynamically loaded extension (either a PHP extension or a Zend extension),
; you may only use these constants after the line that loads the extension.
;
;
.....
; About this file ;
.....
; All the values in the php.ini-dist file correspond to the builtin
; defaults (that is, if no php.ini is used, or if you delete these lines,
; the builtin defaults will be identical).

.....
; Language Options ;
.....

; Enable the PHP scripting language engine under Apache.
engine = On

; Enable compatibility mode with Zend Engine 1 (PHP 4.x)
zend.ze1_compatibility_mode = Off

; Allow the ; NOTE: Using short tags should be avoided when developing applications or
; libraries that are meant for redistribution, or deployment on PHP
; servers which are not under your control, because short tags may not
; be supported on the target server. For portable, redistributable code,
; be sure not to use short tags.
short_open_tag = On

; Allow ASP-style <% %> tags.
asp_tags = Off

; The number of significant digits displayed in floating point numbers.
precision = 12

; Enforce year 2000 compliance (will cause problems with non-compliant browsers)
y2k_compliance = On

; Output buffering allows you to send header lines (including cookies) even
; after you send body content, at the price of slowing PHP's output layer a
; bit. You can enable output buffering during runtime by calling the output
; buffering functions. You can also enable output buffering for all files by
; setting this directive to On. If you wish to limit the size of the buffer
; to a certain size - you can use a maximum number of bytes instead of 'On', as
; a value for this directive (e.g., output_buffering=4096).
output_buffering = Off

```

; You can redirect all of the output of your scripts to a function. For
; example, if you set output_handler to "mb_output_handler", character
; encoding will be transparently converted to the specified encoding.
; Setting any output handler automatically turns on output buffering.
; Note: People who wrote portable scripts should not depend on this ini
; directive. Instead, explicitly set the output handler using ob_start().
; Using this ini directive may cause problems unless you know what script
; is doing.
; Note: You cannot use both "mb_output_handler" with "ob_iconv_handler"
; and you cannot use both "ob_gzhandler" and "zlib.output_compression".
; Note: output_handler must be empty if this is set 'On' !!!!
; Instead you must use zlib.output_handler.
;output_handler =

; Transparent output compression using the zlib library
; Valid values for this option are 'off', 'on', or a specific buffer size
; to be used for compression (default is 4KB)
; Note: Resulting chunk size may vary due to nature of compression. PHP
; outputs chunks that are few hundreds bytes each as a result of
; compression. If you prefer a larger chunk size for better
; performance, enable output_buffering in addition.
; Note: You need to use zlib.output_handler instead of the standard
; output_handler, or otherwise the output will be corrupted.
zlib.output_compression = Off

; You cannot specify additional output handlers if zlib.output_compression
; is activated here. This setting does the same as output_handler but in
; a different order.
;zlib.output_handler =

; Implicit flush tells PHP to tell the output layer to flush itself
; automatically after every output block. This is equivalent to calling the
; PHP function flush() after each and every call to print() or echo() and each
; and every HTML block. Turning this option on has serious performance
; implications and is generally recommended for debugging purposes only.
implicit_flush = Off

; The unserialize callback function will be called (with the undefined class'
; name as parameter), if the unserializer finds an undefined class
; which should be instantiated.
; A warning appears if the specified function is not defined, or if the
; function doesn't include/implement the missing class.
; So only set this entry, if you really want to implement such a
; callback-function.
unserialize_callback_func=

; When floats & doubles are serialized store serialize_precision significant
; digits after the floating point. The default value ensures that when floats
; are decoded with unserialize, the data will remain the same.
serialize_precision = 100

; Whether to enable the ability to force arguments to be passed by reference
; at function call time. This method is deprecated and is likely to be

; unsupported in future versions of PHP/Zend. The encouraged method of
; specifying which arguments should be passed by reference is in the function
; declaration. You're encouraged to try and turn this option Off and make
; sure your scripts work properly with it in order to ensure they will work
; with future versions of the language (you will receive a warning each time
; you use this feature, and the argument will be passed by value instead of by
; reference).

```
allow_call_time_pass_reference = On
```

```
;  
; Safe Mode
```

```
safe_mode = Off
```

; By default, Safe Mode does a UID compare check when
; opening files. If you want to relax this to a GID compare,
; then turn on safe_mode_gid.

```
safe_mode_gid = Off
```

; When safe_mode is on, UID/GID checks are bypassed when
; including files from this directory and its subdirectories.
; (directory must also be in include_path or full path must
; be used when including)

```
safe_mode_include_dir =
```

; When safe_mode is on, only executables located in the safe_mode_exec_dir
; will be allowed to be executed via the exec family of functions.

```
safe_mode_exec_dir =
```

; Setting certain environment variables may be a potential security breach.
; This directive contains a comma-delimited list of prefixes. In Safe Mode,
; the user may only alter environment variables whose names begin with the
; prefixes supplied here. By default, users will only be able to set
; environment variables that begin with PHP_ (e.g. PHP_FOO=BAR).

```
;  
; Note: If this directive is empty, PHP will let the user modify ANY  
; environment variable!
```

```
safe_mode_allowed_env_vars = PHP_
```

; This directive contains a comma-delimited list of environment variables that
; the end user won't be able to change using putenv(). These variables will be
; protected even if safe_mode_allowed_env_vars is set to allow to change them.

```
safe_mode_protected_env_vars = LD_LIBRARY_PATH
```

; open_basedir, if set, limits all file operations to the defined directory
; and below. This directive makes most sense if used in a per-directory
; or per-virtualhost web server configuration file. This directive is
; **NOT** affected by whether Safe Mode is turned On or Off.

```
;open_basedir =
```

; This directive allows you to disable certain functions for security reasons.
; It receives a comma-delimited list of function names. This directive is
; **NOT** affected by whether Safe Mode is turned On or Off.

disable_functions =

; This directive allows you to disable certain classes for security reasons.

; It receives a comma-delimited list of class names. This directive is

; **NOT** affected by whether Safe Mode is turned On or Off.

disable_classes =

; Colors for Syntax Highlighting mode. Anything that's acceptable in

; would work.

;highlight.string = #DD0000

;highlight.comment = #FF9900

;highlight.keyword = #007700

;highlight.bg = #FFFFFF

;highlight.default = #0000BB

;highlight.html = #000000

;

; Misc

;

; Decides whether PHP may expose the fact that it is installed on the server

; (e.g. by adding its signature to the Web server header). It is no security

; threat in any way, but it makes it possible to determine whether you use PHP

; on your server or not.

expose_php = On

.....

; Resource Limits ;

.....

max_execution_time = 30 ; Maximum execution time of each script, in seconds

max_input_time = 60 ; Maximum amount of time each script may spend parsing request data

memory_limit = 40M ; Maximum amount of memory a script may consume (8MB)

.....

; Error handling and logging ;

.....

; error_reporting is a bit-field. Or each number up to get desired error

; reporting level

; E_ALL - All errors and warnings (doesn't include E_STRICT)

; E_ERROR - fatal run-time errors

; E_WARNING - run-time warnings (non-fatal errors)

; E_PARSE - compile-time parse errors

; E_NOTICE - run-time notices (these are warnings which often result

; from a bug in your code, but it's possible that it was

; intentional (e.g., using an uninitialized variable and

; relying on the fact it's automatically initialized to an

; empty string)

; E_STRICT - run-time notices, enable to have PHP suggest changes

; to your code which will ensure the best interoperability

; and forward compatibility of your code

; E_CORE_ERROR - fatal errors that occur during PHP's initial startup

; E_CORE_WARNING - warnings (non-fatal errors) that occur during PHP's

```

;         initial startup
; E_COMPILE_ERROR - fatal compile-time errors
; E_COMPILE_WARNING - compile-time warnings (non-fatal errors)
; E_USER_ERROR - user-generated error message
; E_USER_WARNING - user-generated warning message
; E_USER_NOTICE - user-generated notice message
;
;
; Examples:
;
; - Show all errors, except for notices and coding standards warnings
;
;error_reporting = E_ALL & ~E_NOTICE & ~E_STRICT
;
; - Show all errors, except for notices
;
;error_reporting = E_ALL & ~E_NOTICE
;
; - Show only errors
;
;error_reporting = E_COMPILE_ERROR|E_ERROR|E_CORE_ERROR
;
; - Show all errors except for notices and coding standards warnings
;
error_reporting = E_ERROR | E_USER_ERROR

; Print out errors (as a part of the output). For production web sites,
; you're strongly encouraged to turn this feature off, and use error logging
; instead (see below). Keeping display_errors enabled on a production web site
; may reveal security information to end users, such as file paths on your Web
; server, your database schema or other information.
display_errors = On

; Even when display_errors is on, errors that occur during PHP's startup
; sequence are not displayed. It's strongly recommended to keep
; display_startup_errors off, except for when debugging.
display_startup_errors = Off

; Log errors into a log file (server-specific log, stderr, or error_log (below))
; As stated above, you're strongly advised to use error logging in place of
; error displaying on production web sites.
log_errors = Off

; Set maximum length of log_errors. In error_log information about the source is
; added. The default is 1024 and 0 allows to not apply any maximum length at all.
log_errors_max_len = 1024

; Do not log repeated messages. Repeated errors must occur in same file on same
; line until ignore_repeated_source is set true.
ignore_repeated_errors = Off

; Ignore source of message when ignoring repeated messages. When this setting
; is On you will not log errors with repeated messages from different files or
; sourcelines.

```

ignore_repeated_source = Off

; If this parameter is set to Off, then memory leaks will not be shown (on
; stdout or in the log). This has only effect in a debug compile, and if
; error reporting includes E_WARNING in the allowed list
report_memleaks = On

; Store the last error/warning message in \$php_errormsg (boolean).
track_errors = Off

; Disable the inclusion of HTML tags in error messages.
; Note: Never use this feature for production boxes.
;html_errors = Off

; If html_errors is set On PHP produces clickable error messages that direct
; to a page describing the error or function causing the error in detail.
; You can download a copy of the PHP manual from <http://www.php.net/docs.php>
; and change docref_root to the base URL of your local copy including the
; leading '/'. You must also specify the file extension being used including
; the dot.
; Note: Never use this feature for production boxes.
;docref_root = "/phpmanual/"
;docref_ext = .html

; String to output before an error message.
;error_prepend_string = ""

; String to output after an error message.
;error_append_string = ""

; Log errors to specified file.
;error_log = filename

; Log errors to syslog (Event Log on NT, not valid in Windows 95).
;error_log = syslog

.....
; Data Handling ;
.....

; Note - track_vars is ALWAYS enabled as of PHP 4.0.3

; The separator used in PHP generated URLs to separate arguments.
; Default is "&".
;arg_separator.output = "&"

; List of separator(s) used by PHP to parse input URLs into variables.
; Default is "&".
; NOTE: Every character in this directive is considered as separator!
;arg_separator.input = ";&"

; This directive describes the order in which PHP registers GET, POST, Cookie,
; Environment and Built-in variables (G, P, C, E & S respectively, often

; referred to as EGPCS or GPC). Registration is done from left to right, newer
; values override older values.

```
variables_order = "EGPCS"
```

; Whether or not to register the EGPCS variables as global variables. You may
; want to turn this off if you don't want to clutter your scripts' global scope
; with user data. This makes most sense when coupled with track_vars - in which
; case you can access all of the GPC variables through the \$HTTP_*_VARS[],
; variables.

```
;
```

; You should do your best to write your scripts so that they do not require
; register_globals to be on; Using form variables as globals can easily lead
; to possible security problems, if the code is not very well thought of.

```
register_globals = On
```

; Whether or not to register the old-style input arrays, HTTP_GET_VARS
; and friends. If you're not using them, it's recommended to turn them off,
; for performance reasons.

```
register_long_arrays = On
```

; This directive tells PHP whether to declare the argv&argc variables (that
; would contain the GET information). If you don't use these variables, you
; should turn it off for increased performance.

```
register_argc_argv = On
```

; When enabled, the SERVER and ENV variables are created when they're first
; used (Just In Time) instead of when the script starts. If these variables
; are not used within a script, having this directive on will result in a
; performance gain. The PHP directives register_globals, register_long_arrays,
; and register_argc_argv must be disabled for this directive to have any affect.

```
auto_globals_jit = On
```

; Maximum size of POST data that PHP will accept.

```
post_max_size = 8M
```

; Magic quotes

```
;
```

; Magic quotes for incoming GET/POST/Cookie data.

```
magic_quotes_gpc = On
```

; Magic quotes for runtime-generated data, e.g. data from SQL, from exec(), etc.

```
magic_quotes_runtime = Off
```

; Use Sybase-style magic quotes (escape ' with " instead of \').

```
magic_quotes_sybase = Off
```

; Automatically add files before or after any PHP document.

```
auto_prepend_file =
```

```
auto_append_file =
```

; As of 4.0b4, PHP always outputs a character encoding by default in
; the Content-type: header. To disable sending of the charset, simply

```

; set it to be empty.
;
; PHP's built-in default is text/html
default_mimetype = "text/html"
;default_charset = "iso-8859-1"

; Always populate the $HTTP_RAW_POST_DATA variable.
;always_populate_raw_post_data = On

.....
; Paths and Directories ;
.....

; UNIX: "/path1:/path2"
;include_path = "./usr/share/php"
;
; Windows: "\path1\;path2"
;include_path = ".;c:\php\includes"

; The root of the PHP pages, used only if nonempty.
; if PHP was not compiled with FORCE_REDIRECT, you SHOULD set doc_root
; if you are running php as a CGI under any web server (other than IIS)
; see documentation for security issues. The alternate is to use the
; cgi.force_redirect configuration below
doc_root =

; The directory under which PHP opens the script using /~username used only
; if nonempty.
user_dir =

; Directory in which the loadable extensions (modules) reside.
; extension_dir = "./"

; Whether or not to enable the dl() function. The dl() function does NOT work
; properly in multithreaded servers, such as IIS or Zeus, and is automatically
; disabled on them.
enable_dl = On

; cgi.force_redirect is necessary to provide security running PHP as a CGI under
; most web servers. Left undefined, PHP turns this on by default. You can
; turn it off here AT YOUR OWN RISK
; You CAN safely turn this off for IIS, in fact, you MUST.
cgi.force_redirect = 0

; if cgi.nph is enabled it will force cgi to always sent Status: 200 with
; every request.
; cgi.nph = 1

; if cgi.force_redirect is turned on, and you are not running under Apache or Netscape
; (iPlanet) web servers, you MAY need to set an environment variable name that PHP
; will look for to know it is OK to continue execution. Setting this variable MAY
; cause security issues, KNOW WHAT YOU ARE DOING FIRST.
; cgi.redirect_status_env = ;

```

```

; FastCGI under IIS (on WINNT based OS) supports the ability to impersonate
; security tokens of the calling client. This allows IIS to define the
; security context that the request runs under. mod_fastcgi under Apache
; does not currently support this feature (03/17/2002)
; Set to 1 if running under IIS. Default is zero.
; fastcgi.impersonate = 1;

; Disable logging through FastCGI connection
; fastcgi.log = 0

; cgi.rfc2616_headers configuration option tells PHP what type of headers to
; use when sending HTTP response code. If it's set 0 PHP sends Status: header that
; is supported by Apache. When this option is set to 1 PHP will send
; RFC2616 compliant header.
; Default is zero.
;cgi.rfc2616_headers = 0

.....
; File Uploads ;
.....

; Whether to allow HTTP file uploads.
file_uploads = On

; Temporary directory for HTTP uploaded files (will use system default if not
; specified).
;upload_tmp_dir =

; Maximum allowed size for uploaded files.
upload_max_filesize = 20M

.....
; Fopen wrappers ;
.....

; Whether to allow the treatment of URLs (like http:// or ftp://) as files.
allow_url_fopen = On

; Define the anonymous ftp password (your email address)
;from="john@doe.com"

; Define the User-Agent string
; user_agent="PHP"

; Default timeout for socket based streams (seconds)
default_socket_timeout = 60

; If your scripts have to deal with files from Macintosh systems,
; or you are running on a Mac and need to deal with files from
; unix or win32 systems, setting this flag will cause PHP to
; automatically detect the EOL character in those files so that
; fgets() and file() will work regardless of the source of the file.
; auto_detect_line_endings = Off

```

```

.....
; Dynamic Extensions ;
.....
;
; If you wish to have an extension loaded automatically, use the following
; syntax:
;
; extension=modulename.extension
;
; For example, on Windows:
;
; extension=mysql.dll
;
; ... or under UNIX:
;
; extension=mysql.so
;
; Note that it should be the name of the module only; no directory information
; needs to go here. Specify the location of the extension with the
; extension_dir directive above.

; Example lines:

extension=gd.so
extension=mysql.so
.....
; Module Settings ;
.....

[Syslog]
; Whether or not to define the various syslog variables (e.g. $LOG_PID,
; $LOG_CRON, etc.). Turning it off is a good idea performance-wise. In
; runtime, you can define these variables by calling define_syslog_variables().
define_syslog_variables = Off

[mail function]
; For Win32 only.
SMTP = localhost
smtp_port = 25

; For Win32 only.
;sendmail_from = me@example.com

; For Unix only. You may supply arguments as well (default: "sendmail -t -i").
;sendmail_path =

; Force the addition of the specified parameters to be passed as extra parameters
; to the sendmail binary. These parameters will always replace the value of
; the 5th parameter to mail(), even in safe mode.
;mail.force_extra_parameters =

[SQL]
sql.safe_mode = Off

```

[ODBC]

;odbc.default_db = Not yet implemented

;odbc.default_user = Not yet implemented

;odbc.default_pw = Not yet implemented

; Allow or prevent persistent links.

odbc.allow_persistent = On

; Check that a connection is still valid before reuse.

odbc.check_persistent = On

; Maximum number of persistent links. -1 means no limit.

odbc.max_persistent = -1

; Maximum number of links (persistent + non-persistent). -1 means no limit.

odbc.max_links = -1

; Handling of LONG fields. Returns number of bytes to variables. 0 means

; passthru.

odbc.defaultlrl = 4096

; Handling of binary data. 0 means passthru, 1 return as is, 2 convert to char.

; See the documentation on odbc_binmode and odbc_longreadlen for an explanation

; of uodbc.defaultlrl and uodbc.defaultbinmode

odbc.defaultbinmode = 1

[MySQL]

; Allow or prevent persistent links.

mysql.allow_persistent = On

; Maximum number of persistent links. -1 means no limit.

mysql.max_persistent = -1

; Maximum number of links (persistent + non-persistent). -1 means no limit.

mysql.max_links = -1

; Default port number for mysql_connect(). If unset, mysql_connect() will use

; the \$MYSQL_TCP_PORT or the mysql-tcp entry in /etc/services or the

; compile-time value defined MYSQL_PORT (in that order). Win32 will only look

; at MYSQL_PORT.

mysql.default_port = 3306

; Default socket name for local MySQL connects. If empty, uses the built-in

; MySQL defaults.

mysql.default_socket =

; Default host for mysql_connect() (doesn't apply in safe mode).

mysql.default_host =

; Default user for mysql_connect() (doesn't apply in safe mode).

mysql.default_user =

; Default password for mysql_connect() (doesn't apply in safe mode).

; Note that this is generally a **bad** idea to store passwords in this file.
; **Any** user with PHP access can run 'echo get_cfg_var("mysql.default_password")
; and reveal this password! And of course, any users with read access to this
; file will be able to reveal the password as well.
mysql.default_password =

; Maximum time (in secondes) for connect timeout. -1 means no limit
mysql.connect_timeout = 60

; Trace mode. When trace_mode is active (=On), warnings for table/index scans and
; SQL-Errors will be displayed.
mysql.trace_mode = Off

[MySQLI]

; Maximum number of links. -1 means no limit.
mysqli.max_links = -1

; Default port number for mysqli_connect(). If unset, mysqli_connect() will use
; the \$MYSQL_TCP_PORT or the mysql-tcp entry in /etc/services or the
; compile-time value defined MYSQL_PORT (in that order). Win32 will only look
; at MYSQL_PORT.
mysqli.default_port = 3306

; Default socket name for local MySQL connects. If empty, uses the built-in
; MySQL defaults.
mysqli.default_socket =

; Default host for mysqli_connect() (doesn't apply in safe mode).
mysqli.default_host =

; Default user for mysqli_connect() (doesn't apply in safe mode).
mysqli.default_user =

; Default password for mysqli_connect() (doesn't apply in safe mode).
; Note that this is generally a **bad** idea to store passwords in this file.
; **Any** user with PHP access can run 'echo get_cfg_var("mysqli.default_pw")
; and reveal this password! And of course, any users with read access to this
; file will be able to reveal the password as well.
mysqli.default_pw =

; Allow or prevent reconnect
mysqli.reconnect = Off

[mSQL]

; Allow or prevent persistent links.
msql.allow_persistent = On

; Maximum number of persistent links. -1 means no limit.
msql.max_persistent = -1

; Maximum number of links (persistent+non persistent). -1 means no limit.
msql.max_links = -1

[PostgreSQL]

; Allow or prevent persistent links.

pgsql.allow_persistent = On

; Detect broken persistent links always with pg_pconnect().

; Auto reset feature requires a little overheads.

pgsql.auto_reset_persistent = Off

; Maximum number of persistent links. -1 means no limit.

pgsql.max_persistent = -1

; Maximum number of links (persistent+non persistent). -1 means no limit.

pgsql.max_links = -1

; Ignore PostgreSQL backends Notice message or not.

; Notice message logging require a little overheads.

pgsql.ignore_notice = 0

; Log PostgreSQL backends Notice message or not.

; Unless pgsql.ignore_notice=0, module cannot log notice message.

pgsql.log_notice = 0

[Sybase]

; Allow or prevent persistent links.

sybase.allow_persistent = On

; Maximum number of persistent links. -1 means no limit.

sybase.max_persistent = -1

; Maximum number of links (persistent + non-persistent). -1 means no limit.

sybase.max_links = -1

;sybase.interface_file = "/usr/sybase/interfaces"

; Minimum error severity to display.

sybase.min_error_severity = 10

; Minimum message severity to display.

sybase.min_message_severity = 10

; Compatibility mode with old versions of PHP 3.0.

; If on, this will cause PHP to automatically assign types to results according

; to their Sybase type, instead of treating them all as strings. This

; compatibility mode will probably not stay around forever, so try applying

; whatever necessary changes to your code, and turn it off.

sybase.compatibility_mode = Off

[Sybase-CT]

; Allow or prevent persistent links.

sybct.allow_persistent = On

; Maximum number of persistent links. -1 means no limit.

sybct.max_persistent = -1

; Maximum number of links (persistent + non-persistent). -1 means no limit.

sybct.max_links = -1

; Minimum server message severity to display.

sybct.min_server_severity = 10

; Minimum client message severity to display.

sybct.min_client_severity = 10

[dbx]

; returned column names can be converted for compatibility reasons

; possible values for dbx.colnames_case are

; "unchanged" (default, if not set)

; "lowercase"

; "uppercase"

; the recommended default is either upper- or lowercase, but

; unchanged is currently set for backwards compatibility

dbx.colnames_case = "unchanged"

[bcmath]

; Number of decimal digits for all bcmath functions.

bcmath.scale = 0

[browscap]

;browscap = extra/browscap.ini

[Informix]

; Default host for ifx_connect() (doesn't apply in safe mode).

ifx.default_host =

; Default user for ifx_connect() (doesn't apply in safe mode).

ifx.default_user =

; Default password for ifx_connect() (doesn't apply in safe mode).

ifx.default_password =

; Allow or prevent persistent links.

ifx.allow_persistent = On

; Maximum number of persistent links. -1 means no limit.

ifx.max_persistent = -1

; Maximum number of links (persistent + non-persistent). -1 means no limit.

ifx.max_links = -1

; If on, select statements return the contents of a text blob instead of its id.

ifx.textasvarchar = 0

; If on, select statements return the contents of a byte blob instead of its id.

ifx.byteasvarchar = 0

; Trailing blanks are stripped from fixed-length char columns. May help the

; life of Informix SE users.

```
ifx.charasvchar = 0
```

```
; If on, the contents of text and byte blobs are dumped to a file instead of  
; keeping them in memory.
```

```
ifx.blobinfile = 0
```

```
; NULL's are returned as empty strings, unless this is set to 1. In that case,  
; NULL's are returned as string 'NULL'.
```

```
ifx.nullformat = 0
```

```
[Session]
```

```
; Handler used to store/retrieve data.
```

```
session.save_handler = files
```

```
; Argument passed to save_handler. In the case of files, this is the path  
; where data files are stored. Note: Windows users have to change this  
; variable in order to use PHP's session functions.
```

```
;
```

```
; As of PHP 4.0.1, you can define the path as:
```

```
;
```

```
; session.save_path = "N;/path"
```

```
;
```

```
; where N is an integer. Instead of storing all the session files in  
; /path, what this will do is use subdirectories N-levels deep, and  
; store the session data in those directories. This is useful if you  
; or your OS have problems with lots of files in one directory, and is  
; a more efficient layout for servers that handle lots of sessions.
```

```
;
```

```
; NOTE 1: PHP will not create this directory structure automatically.
```

```
; You can use the script in the ext/session dir for that purpose.
```

```
; NOTE 2: See the section on garbage collection below if you choose to
```

```
; use subdirectories for session storage
```

```
;
```

```
; The file storage module creates files using mode 600 by default.
```

```
; You can change that by using
```

```
;
```

```
; session.save_path = "N;MODE;/path"
```

```
;
```

```
; where MODE is the octal representation of the mode. Note that this
```

```
; does not overwrite the process's umask.
```

```
;session.save_path = /var/lib/php5
```

```
; Whether to use cookies.
```

```
session.use_cookies = 1
```

```
; This option enables administrators to make their users invulnerable to
```

```
; attacks which involve passing session ids in URLs; defaults to 0.
```

```
; session.use_only_cookies = 1
```

```
; Name of the session (used as cookie name).
```

```
session.name = PHPSESSID
```

```
; Initialize session on request startup.
```

```
session.auto_start = 0
```

```
; Lifetime in seconds of cookie or, if 0, until browser is restarted.
```

```
session.cookie_lifetime = 0
```

```
; The path for which the cookie is valid.
```

```
session.cookie_path = /tmp
```

```
; The domain for which the cookie is valid.
```

```
session.cookie_domain =
```

```
; Handler used to serialize data. php is the standard serializer of PHP.
```

```
session.serialize_handler = php
```

```
; Define the probability that the 'garbage collection' process is started
```

```
; on every session initialization.
```

```
; The probability is calculated by using gc_probability/gc_divisor,
```

```
; e.g. 1/100 means there is a 1% chance that the GC process starts
```

```
; on each request.
```

```
; This is disabled in the Debian packages, due to the strict permissions
```

```
; on /var/lib/php5. Instead of setting this here, see the cronjob at
```

```
; /etc/cron.d/php5, which uses the session.gc_maxlifetime setting below
```

```
;session.gc_probability = 0
```

```
session.gc_divisor = 100
```

```
; After this number of seconds, stored data will be seen as 'garbage' and
```

```
; cleaned up by the garbage collection process.
```

```
session.gc_maxlifetime = 1440
```

```
; NOTE: If you are using the subdirectory option for storing session files
```

```
; (see session.save_path above), then garbage collection does not
```

```
; happen automatically. You will need to do your own garbage
```

```
; collection through a shell script, cron entry, or some other method.
```

```
; For example, the following script would be the equivalent of
```

```
; setting session.gc_maxlifetime to 1440 (1440 seconds = 24 minutes):
```

```
; cd /path/to/sessions; find -cmin +24 | xargs rm
```

```
; PHP 4.2 and less have an undocumented feature/bug that allows you to
```

```
; to initialize a session variable in the global scope, albeit register_globals
```

```
; is disabled. PHP 4.3 and later will warn you, if this feature is used.
```

```
; You can disable the feature and the warning separately. At this time,
```

```
; the warning is only displayed, if bug_compat_42 is enabled.
```

```
session.bug_compat_42 = 1
```

```
session.bug_compat_warn = 1
```

```
; Check HTTP_REFERER to invalidate externally stored URLs containing ids.
```

```
; HTTP_REFERER has to contain this substring for the session to be
```

```
; considered as valid.
```

```
session.referer_check =
```

```
; How many bytes to read from the file.
```

session.entropy_length = 0

; Specified here to create the session id.

session.entropy_file =

;session.entropy_length = 16

;session.entropy_file = /dev/urandom

; Set to {nocache,private,public,} to determine HTTP caching aspects

; or leave this empty to avoid sending anti-caching headers.

session.cache_limiter = nocache

; Document expires after n minutes.

session.cache_expire = 180

; trans sid support is disabled by default.

; Use of trans sid may risk your users security.

; Use this option with caution.

; - User may send URL contains active session ID

; to other person via. email/irc/etc.

; - URL that contains active session ID may be stored

; in publically accessible computer.

; - User may access your site with the same session ID

; always using URL stored in browser's history or bookmarks.

session.use_trans_sid = 0

; Select a hash function

; 0: MD5 (128 bits)

; 1: SHA-1 (160 bits)

session.hash_function = 0

; Define how many bits are stored in each character when converting

; the binary hash data to something readable.

;

; 4 bits: 0-9, a-f

; 5 bits: 0-9, a-v

; 6 bits: 0-9, a-z, A-Z, "-", ","

session.hash_bits_per_character = 4

; The URL rewriter will look for URLs in a defined set of HTML tags.

; form/fieldset are special; if you include them here, the rewriter will

; add a hidden <input> field with the info which is otherwise appended

; to URLs. If you want XHTML conformity, remove the form entry.

; Note that all valid entries require a "=", even if no value follows.

url_rewriter.tags = "a:href,area:href,frame=src,input=src,form=,fieldset="

[MSSQL]

; Allow or prevent persistent links.

mssql.allow_persistent = On

; Maximum number of persistent links. -1 means no limit.

mssql.max_persistent = -1

; Maximum number of links (persistent+non persistent). -1 means no limit.

mssql.max_links = -1

; Minimum error severity to display.

mssql.min_error_severity = 10

; Minimum message severity to display.

mssql.min_message_severity = 10

; Compatability mode with old versions of PHP 3.0.

mssql.compatability_mode = Off

; Connect timeout

;mssql.connect_timeout = 5

; Query timeout

;mssql.timeout = 60

; Valid range 0 - 2147483647. Default = 4096.

;mssql.textlimit = 4096

; Valid range 0 - 2147483647. Default = 4096.

;mssql.textsize = 4096

; Limits the number of records in each batch. 0 = all records in one batch.

;mssql.batchsize = 0

; Specify how datetime and datetim4 columns are returned

; On => Returns data converted to SQL server settings

; Off => Returns values as YYYY-MM-DD hh:mm:ss

;mssql.datetimeconvert = On

; Use NT authentication when connecting to the server

mssql.secure_connection = Off

; Specify max number of processes. Default = 25

;mssql.max_procs = 25

[Assertion]

; Assert(expr); active by default.

;assert.active = On

; Issue a PHP warning for each failed assertion.

;assert.warning = On

; Don't bail out by default.

;assert.bail = Off

; User-function to be called if an assertion fails.

;assert.callback = 0

; Eval the expression with current error_reporting(). Set to true if you want

; error_reporting(0) around the eval().

```
;assert.quiet_eval = 0
```

```
[Ingres II]
```

```
; Allow or prevent persistent links.
```

```
ingres.allow_persistent = On
```

```
; Maximum number of persistent links. -1 means no limit.
```

```
ingres.max_persistent = -1
```

```
; Maximum number of links, including persistents. -1 means no limit.
```

```
ingres.max_links = -1
```

```
; Default database (format: [node_id::]dbname[/srv_class]).
```

```
ingres.default_database =
```

```
; Default user.
```

```
ingres.default_user =
```

```
; Default password.
```

```
ingres.default_password =
```

```
[Verisign Payflow Pro]
```

```
; Default Payflow Pro server.
```

```
pfpro.defaulthost = "test-payflow.verisign.com"
```

```
; Default port to connect to.
```

```
pfpro.defaultport = 443
```

```
; Default timeout in seconds.
```

```
pfpro.defaulttimeout = 30
```

```
; Default proxy IP address (if required).
```

```
;pfpro.proxyaddress =
```

```
; Default proxy port.
```

```
;pfpro.proxyport =
```

```
; Default proxy logon.
```

```
;pfpro.proxylogon =
```

```
; Default proxy password.
```

```
;pfpro.proxypassword =
```

```
[com]
```

```
; path to a file containing GUIDs, IIDs or filenames of files with TypeLibs
```

```
;com.typelib_file =
```

```
; allow Distributed-COM calls
```

```
;com.allow_dcom = true
```

```
; autoregister constants of a components typelib on com_load()
```

```
;com.autoregister_typelib = true
```

```
; register constants casesensitive
```

```
;com.autoregister_casesensitive = false
```

```
; show warnings on duplicate constat registrations
```

;com.autoregister_verbose = true

[mbstring]

; language for internal character representation.

;mbstring.language = Japanese

; internal/script encoding.

; Some encoding cannot work as internal encoding.

; (e.g. SJIS, BIG5, ISO-2022-*)

;mbstring.internal_encoding = EUC-JP

; http input encoding.

;mbstring.http_input = auto

; http output encoding. mb_output_handler must be

; registered as output buffer to function

;mbstring.http_output = SJIS

; enable automatic encoding translation according to

; mbstring.internal_encoding setting. Input chars are

; converted to internal encoding by setting this to On.

; Note: Do *not* use automatic encoding translation for

; portable libs/applications.

;mbstring.encoding_translation = Off

; automatic encoding detection order.

; auto means

;mbstring.detect_order = auto

; substitute_character used when character cannot be converted

; one from another

;mbstring.substitute_character = none;

; overload(replace) single byte functions by mbstring functions.

; mail(), ereg(), etc are overloaded by mb_send_mail(), mb_ereg(),

; etc. Possible values are 0,1,2,4 or combination of them.

; For example, 7 for overload everything.

; 0: No overload

; 1: Overload mail() function

; 2: Overload str*() functions

; 4: Overload ereg*() functions

;mbstring.func_overload = 0

[FrontBase]

;fbsql.allow_persistent = On

;fbsql.autocommit = On

;fbsql.default_database =

;fbsql.default_database_password =

;fbsql.default_host =

;fbsql.default_password =

;fbsql.default_user = "_SYSTEM"

;fbsql.generate_warnings = Off

;fbsql.max_connections = 128

```
;fbsql.max_links = 128
;fbsql.max_persistent = -1
;fbsql.max_results = 128
;fbsql.batchSize = 1000
```

[exif]

```
; Exif UNICODE user comments are handled as UCS-2BE/UCS-2LE and JIS as JIS.
; With mbstring support this will automatically be converted into the encoding
; given by corresponding encode setting. When empty mbstring.internal_encoding
; is used. For the decode settings you can distinguish between motorola and
; intel byte order. A decode setting cannot be empty.
;exif.encode_unicode = ISO-8859-15
;exif.decode_unicode_motorola = UCS-2BE
;exif.decode_unicode_intel = UCS-2LE
;exif.encode_jis =
;exif.decode_jis_motorola = JIS
;exif.decode_jis_intel = JIS
```

[Tidy]

```
; The path to a default tidy configuration file to use when using tidy
;tidy.default_config = /usr/local/lib/php/default.tcfg

; Should tidy clean and repair output automatically?
; WARNING: Do not use this option if you are generating non-html content
; such as dynamic images
tidy.clean_output = Off
```

[soap]

```
; Enables or disables WSDL caching feature.
soap.wsdl_cache_enabled=1
; Sets the directory name where SOAP extension will put cache files.
soap.wsdl_cache_dir="/tmp"
; (time to live) Sets the number of second while cached file will be used
; instead of original one.
soap.wsdl_cache_ttl=86400
```

; Local Variables:

; tab-width: 4

; End:

extension=pgsql.so

#6 - 2006-01-03 00:50 - old_pss7

works now with the last t3 beta :-)

#7 - 2006-01-03 21:02 - Sebastian Kurfuerst

Ok, closing the bug.

Greets, Sebastian