# TYPO3 Security Cookbook



The content of this document is related to TYPO3

- a GNU/GPL CMS/Framework available from www.typo3.com

## "What does it do?"

This document contains a checklist for TYPO3 system administrators. You can use it to ensure you closed common open doors of your system.

## Introduction

Basically being the result of a discussion at T3Board04 in Kitzbühel, this document will of course have grow in the future. If you want to contribute by adding new chapters, please send your text to the author of this document.

## Requirements

The checklist ties up on a working TYPO3 environment. It is not intended to be another installation manual!

Thus, you can just install your server as usual, and use this list for checking each chapter one by one.

## Priorities & Structure

Although it is recommended that you walk through every chapter in this document, there are of course priorities. To make it easy for you, all issues are sorted from the top to the bottom.

## TYPO3

**Secure the Install Tool**

**Priority**: High

Explanation of Backgrounds: The TYPO3 Install Tool is the powerful center of your TYPO3 system. As a basic rule, it should never be accessible from the Web unless you actually need it.

**Measures**:

disable the Install Tool (remove the comment in front of the “die()” line in typo3/install/index.php) OR

move away the typo3/install/ directory or make it inaccessible for the web server OR

limit access to typo3/install to specific hosts / networks / domains (using .htaccess) – depreciated !

you may want to add .htaccess authentication (though also not considered secure)

at LEAST make sure to change the Install Tool password to a non-trivial value

## Change “admin” Password, Rename “admin” User

**Priority**: High

Explanation of Backgrounds: The default admin user and –password is always a first try for hackers.

**Measures**:

change the password of the “admin” user immediately after install

replace the “admin” user by other admins – preferably personalized ones (see “Choose Personal User Names”)

## Do not use “Quickstart“, “Testsite” et al. for Live Systems

**Priority**: High

Explanation of Backgrounds: The “Quickstart” package – like other demo packages - is intended to provide a read-to-run demo system. It contains a lot of code and content that you would have to clean up prior to use the installation for production purposes. It is much better to start with a “clean” system and install (maybe import) only what you really need.

**Measures**:

use the “dummy” package for live sites

at LEAST make sure to remove all FE and BE users

## File System Access Rights

**Priority**: High

Explanation of Backgrounds: Least privileges should be given in the TYPO3 and htdocs directories

**Measures**:

make sure to revoke all WRITE privileges in typo3_src for the web server’s user account

set ownership and umask in htdocs to appropriate values (differs for the various subdirectories!)

paranoid’s setting: Place localconf.php outside of htdocs by changing typo3conf/localconf.php to the following:

```
<?php
    require("<directory outside htdocs>/localconf.php");
?>
```

## Remove unneeded code

**Priority**: High

Explanation of Backgrounds: Depending on your base package (esp. if you use CVS code – depreciated anyway!), it may contain extra code that is not needed for production and therefore should not be accessible for potential offenders.

**Measures**:

Delete the ./misc, ./cvs and the ./dev directory if present, or at least make them inaccessible for the web server’s user account

if you have live server separate from your editors’ production system, remove the BE from the live servers

Only install required Extensions

## Configure TYPO3 Security Options

**Priority**: High

Explanation of Backgrounds: TYPO3 provides numerous configuration options that increase system security. Check them out and use what makes sense in your situation!

Measures / Install Tool (see Install Tool sections for latest options and detailed descriptions):

[strictFormmail] – set to "1"

[encryptionKey] – should be set (e.g. in "Basic Configuration")

[warning_email_addr]

[lockIP]

[lockRootPath]

[fileCreateMask]

[fileDenyPattern] – should at least contain \.php$|\.php.$

[folderCreateMask]

[warning_mode]

[IPmaskList]

[lockBeUserToDBmounts]

[lockSSL]

[enabledBeUserIPLock]

[disable_exec_function]

[usePHPFileFunctions]

[noPHPscriptInclude] – consider this if others have access to your template files

[lockHashKeyWords]

[devIPmask]

Measures / BE GUI

Add lockToDomain in be_users/be_groups records

## Avoid config.baseURL=1

**Priority**: High

Explanation of Backgrounds: In older versions, your cache may be poisoned, resulting in foreign pages being displayed instead of your own.

**Measures**:

use the absolute URL instead OR

make sure the website can only be accessed with the correct URL (i.e. use name based virtual hosts in your web server)

## Consider Using SSL for Backend Access

**Priority**: Medium

Explanation of Backgrounds: Although BE login itself is encrypted, the follow-up BE access is unprotected unless you use SSL. Since that may affect sensitive information, you are advised to use SSL for all BE access.

**Measures**:

configure HTTPS for your server

redirect HTTP access to /typo3 to HTTPS in your web server

use the lockSSL Install Tool option (see “Configure TYPO3 Security Options”)

## FE User Security

**Priority**: Medium

Explanation of Backgrounds: Please take your FE users security concerns seriously, i.e. protect their sensitive data.

**Measures**:

Use SSL for FE logon

Use SSL for FE user self-registration and password change

Use SSL for all sensitive data like forms (not only credit card data…) or personal output

do not store FE user passwords in clear - use an extension like kb_md5fepw, or use secure external password storage like LDAP (preferably via SSL) with MD5

## Restrict Special Content Elements usage

**Priority**: High

Explanation of Backgrounds: Some low-level content elements may let backend users gain system access beyond the level you intended, or may enable them to create security breaches without knowing.

Therefore, the following restrictions are recommended for all users that all not fully aware or capable of understanding the security implications, or are simply not fully trusted.

**Measures**:

- Do not allow Content Element "HTML"

- Do not allow plain HMTL in Text Content Elements

- Do not allow plugins that let the user insert PHP code

## Choose Personal User Names for Backend Access

**Priority**: High

Explanation of Backgrounds: “john.doe” is better then “bigboss” - avoid using shared accounts in general. You should always be able to keep track on who is doing what, and backend users should be aware of that fact.

**Measures**:

give personal user names

inform your BE users about the logging

educate them not to share accounts

## Logging / Auditing

**Priority**: High

Explanation of Backgrounds: Know your log files, and be sure they are configured to audit all information you need.

**Measures**:

The sys_log table is your default BE user log (accessible via Tools->Log)

xxxxxxxxxxxxxxxxx you can enable additional logging using the [logfile_dir] and [logfile_write] Keywords

The [trackBeUser] setting is intended for debug purposes

The [enable_DLOG] (in conjunction with constant TYPO3_DLOG)

## Error Handling

**Priority**: Medium

Explanation of Backgrounds: Even if you try to avoid it – your system may run into one (or more :-) errors one day - so "be prepared". Make sure errors are tracked, and user output is convenient and does not expose any internal information.

**Measures**:

PHP errors should be handled, but normally through PHP means (see below). Thus [displayErrors] should be set to 0.

More a cosmetical thing: TYPO3-internal "Page not Found" errors can be configured using the [pageNotFound_handling] and [pageNotFound_handling_statheader] setting.

## Use Trusted / Reviewed Extensions

**Priority**: Medium

Explanation of Backgrounds: Every extension can potentially expose your entire system, whether by a security bug or even intentionally.

**Measures**:

Use extensions that have undergone the extension review process.

If an extension is not reviewed yet, think about sponsoring its review.

Remember to have your own extensions quality-assured as well.

## Subscribe to TYPO3-Announce, Apply Fixes

**Priority**: High

Explanation of Backgrounds: In case a security issue with TYPO3 or one of its extensions occurs, a "TYPO3 Security Bulletin" will be communicated through the "TYPO3-Announce" Mailing list. A fix or workaround will come along.

**Measures**:

Subscribe to TYPO3-Announce (goto xxxxxxxxxxxxx)

Read the Bulletins, and implement the measures if you are affected.

Make sure to do the same for future installations!

All TYPO3 Security Bulletins can be found on xxxxxxxxxxxxxxx

## Non TYPO3 Settings

**PHP**

These settings should b e done in php.ini

log errors to an error file – needed to reproduce any problems

Display errors  off – do not display any errors through the webserver – dont'push people to possible leaks

use safemode, or at least open basedir to prevent webs from accessing other directories or execute things, they mustn't – again: less is more.

Use a CGI/PHP wrapper (suPHP?) ???

compile your PHP with minimum compile options, or install only needed extentions - what is not included, is not vulnerable.

Register_globals = Off . If this is really required, it could be switched on for single webs in the .htaccess file.

verify and use .htaccess !

**Apache**

In httpd.conf don't load modules, you don't need. Best is not to even install them. Directory listing for example is not needed. This can be done via php script if needed

Only install required modules

disable version info in error pages, tell possible attackers as little as possible

**MySQL**

dissallow network connections to mysql, if needed, tunnel it through a secure connection (stunnel)

don't use the mysql root user, use one user per database

set an own password for mysql root user, don't use the server root password

# General

**problems according to shared hosting**

requirements to the isp

activate su_exec

don't store passwords on servers ! If you need a password.txt file: store it on a sheet of paper, or on a box which is not connected to the web. (i know, this one is nagging, but ...)

subscribe to the security lists of your distribution / Operating System Vendor. (OS, ssh, apache, php, mysql, openssl, …)

if possible, run updates daily through a cron job

try to use secured connections for all protocols (sftp, etc)

restrict acces for users only to needed directories ( i.e. Proftpd: users home = htdocs ; DefaultRoot = ~)

monitor your servers to see, if something unusual happens (i.e. nagios, tripwire, tiger, logsurfer, ...)

harden system (disabel unneeded services, remove compilers, ...)

protect phpMyAdmin with .htaccess

don't do dumps or backups to fileadmin or htdocs, if you use backup extensions, delete the backups after downloading them.

**Topics NOT listed here**

rename "/typo3" --> we discussed this, but decided, not to recommend this.

Backups (should be clear)

sec.-extensions, sso, … (we mentioned checking the resp. sites)

BE roles / permissions

password rules – (this is not “secure internet server for dummies” book )