

```

-----+-----
|                                     |
|               TomeNET Administrator Guide               |
|-----+-----|
| Latest update: 22. May 2026 - written by C. Blue (c_blue@gmx.net) |
| for TomeNET version v4.9.4 - official websites are:           |
| https://www.tomenet.eu/ (official main site, formerly www.tomenet.net) |
| https://muuttuja.org/tomenet/ (Mikael's TomeNET site)         |
|
| This guide is meant for TomeNET server administrators, not for players.
| The guide for players is just called "TomeNET Guide" (without 'Admin').
|
| If your text editor cannot display the guide properly (needs fixed-width
| font like for example Courier), simply open it in any web browser instead.
+---+
| Welcome to this guide!
| Although I'm trying, I give no guarantee that this guide is always correct.
| Don't blame me if something differs or is missing; it shouldn't though.
|
| If you have any suggestions about the guide or the game, please use the
| /rfe command in the game or write to the official forum on www.tomenet.eu.
|
|
\

```

Contents

```

-----
(0) Administrative slash commands - 1
    (0.1) Shutting down the server gracefully - 1
    (0.2) Live LUA changes
(1) Accounts - 2
    (1.1) The account editor 'accredit' - 2
    (1.2) New accounts and validation - 2
    (1.3) The administrator account - 2
    (1.3a) Dungeon Master - 2
    (1.3b) Dungeon Wizards
    (1.4) Commands to change powers of individual accounts - 3
(2) Admininstrator menu
(3) Server command-line options - 4
(4) Server configuration files - 4
(5) Compilation - 5
    (5.1) Server compilation - 5
    (5.2) accredit compilation - 5

```

(0) Slash commands

```

=====
Issuing slash commands is done in-game, by pressing ':' key to open the chat
prompt and entering the command there, starting with the slash '/' symbol.
This guide only mentions slash commands that are specifically for use by
administrators or players with elevated privileges and thereby not accessible
to normal player accounts.

```

Account-level related commands (validation, privileging) are listed in (1.4).

(0.1) Shutting down the server gracefully

```

-----
Log in with a Dungeon Master or Dungeon Wizard and issue the slash command
/shutdown 0

```

This will initiate an orderly shutdown with proper cleanup and therefore is much preferable over things such as spamming CTRL+C into the server's terminal or to kill it otherwise.

To make the server wait for all players to leave the dungeons before executing the shutdown, use this command:

```

/shutempty

```

(0.2) Live LUA changes

To reinitialize the LUA framework immediately without the need for a server restart, use this command:

```
/initlua
```

Use this if you changed any LUA files and want to put the changes live immediately, mainly for the file 'lib/scpt/custom.lua' (compare (4)).

(1) Accounts

=====

Everyone who interacts with the server within the game world uses an 'account', defined by login name and login password.

(1.1) The account editor 'accedit'

Accounts can be moderated and managed with the account editor 'accedit', which resides in 'src/account' folder. The compiled 'accedit' binary must be placed into the same folder where file 'tomenet.acc' resides, usually 'lib/save'. This tool can be used to validate, invalidate or change privileges of players, and much more. However, it is usually simpler and more comfortable to use administrative slash commands instead, issued in-game on an administrator character, ie a DM or DW (see below), see (1.4).

For compilation of accedit see (5.2) much further down.

(1.2) New accounts and validation

Whenever someone logs in to the TomeNET game server for the first time ie with an account name that is not yet known to the server, the server automatically creates a new account for this player.

These accounts start out as 'invalid', which means they are restricted in what they can do and how far they can level up, until they are 'validated' by a server administrator character (by using '/val' command, see (1.4)) or via the 'accedit' program.

Only the first account ever created on a server is not automatically an invalid player account, instead it will become...

(1.3) The administrator account

The very first account created on a server will automatically become the main administrative account: The "dungeon master" (DM) account.

(1.3a) Dungeon Master

The character of the same name as an administrative account will be the Dungeon Master (DM) character, which means it'll automatically be a ghost while still retaining the powers of carrying items and doing basically everything a living character could.

At the same time it can move through any kind of walls except for level boundaries and can also do various things that normal player characters cannot do, both in terms of gameplay interactions and also in usage of specific slash commands that are not available to normal players.

Last but not least the DM character is always invisible to everyone (except for other DM characters) and does not appear on the players list ('@' key), unless this is changed explicitly in the tomenet.cfg file.

(1.3a) Dungeon Wizards

All other characters on the DM account will become Dungeon Wizards (DW), which are not ghosts but still have full administrative capabilities and therefore can use all administrative slash commands same as the Dungeon Master can. The DWs are fully visible to everyone, just as any other character.

(1.4) Commands to change powers of individual accounts

Instead of modifying account status using the 'accredit' tool, it's usually much easier and more comfortable to log in with a Dungeon Master or Dungeon Wizard and use administrative slash commands to moderate player accounts and change their privileges:

Privileged commands

These commands can be used by DMs, DWs but also by normal players whose accounts were set to 'privileged mode' by an administrator.

/val Validate a newly created player account, so it can be used for normal gameplay without 'trial mode' restrictions, such as:
 - no levelling beyond 25.
 - no private chatting.
 - no dropping items.

/inval Invalidate a player account, reapplying 'trial mode' restrictions. This is for example done on the official servers where a strict one-account-per-player rule is enforced, when an admin notices that a player uses more than one account. Usually the admin will /inval all accounts aside from the first one, or all accounts other than the one with most/highest characters.

/linv List all newly created invalid accounts (so you can decide to validate them one by one using the /val command then.)

Administrative commands

These commands can only be used by administrators (DMs and DWs).

/privilege Set a player account to 'privileged' mode, unlocking the 3 commands listed above. Sometimes, some quests in 'testing' stage might be accessible to privileged players for early testing.

/vprivilege Set a player account to 'very privileged' mode. Has no further benefits over privileged mode, but may sometimes allow early testing of new features or commands if available.

/unprivilege Set a player's account back to non-privileged aka normal.

/makeadmin Turns a player account into an administrative account (see (1.3) for administrative accounts). All its characters will turn into DM (if same name as account) and DWs. Use with care as this action cannot be undone; instead, you will have to invoke the 'accredit' tool and 'D'eleate the account there, to get rid of it again. Alternatively to 'accredit' you can also use the admin menu in-game, invoked with ']' key ->Player Commands ->Delete Player which asks for an account name and will delete all characters on that account and the account itself (compare (2)).

When logged in on an account you can check its status by pressing = A.

(2) Administrator menu

=====

When logged on as DM or DW you can press the ']' key in-game to invoke the administrator menu:

- (1) Level Commands
 - (3) Add dungeon --- Use this to place an IDDC dungeon on a newly created server; make sure its coordinates match the ones the server compiled with in 'src/common/defines-local.h'.
- (2) Building Commands
- (3) Summoning Commands
- (4) Generation Commands
- (5) Player Commands
 - (6) Delete player --- fully deletes an account and all its chars.
- (6) System Commands

(3) Server command-line options

=====

The server binary (usually 'tomenet.server') can be invoked with command-line options.

Some of these are only available the first time a new TomeNET server is started as they will affect the random world generation - a process that is only done the very first time a server starts.

If you want to change the server's world again with those particular options labeled 'On server creation', you must delete the file 'lib/save/server' which contains the world. This will reset the server's world to a fresh state.

All command-line options available are:

- h Just display all available command-line options and exit.
- r Reset the server (implies -w and -f)
- w Reset the server partially: New wilderness
- f Reset the server partially: New flavours
- a (On server creation!) Ensure that all terrain types are created
- b (On server creation!) Don't allow watery wilderness around Bree
- l (On server creation!) Place low-level dungeons not far from Bree
- H Reinitialize houses
- z Don't catch signals
- u<path> Look for user files in the directory <path> (UNUSED)
- s<path> Look for save files in the directory <path>
- t<path> Look for text files in the directory <path>
- m<file> Specify configuration <file>

(4) Server configuration files

=====

Various files influence the server configuration and the world.

- lib/config/tomenet.cfg
This is the main configuration file, which the server reads each time it starts up.
- src/common/defines*.h
Header files starting on 'defines...' that contain a lot of server settings, most notably the world map location of the 'Ironman Deep Dive Challenge' dungeon (defines-local.h).
If anything in the header files is changed, a full, clean recompilation of the server is required (see (5.1)).
- lib/scpt/custom.lua
LUA code in here can manage things such as starting scheduled events or season changes.

Any changes to LUA files can be put live instantly if a DM/DW issues the '/initlua' command, which reinitializes the LUA framework without the need for a server restart.

(5) Compilation

=====

Anything described here is for Linux.

Unlike the TomeNET client the server side has way less package requirements. See TomeNET-Guide.txt (2.1b) for basic requirements, only some of that will be needed for the server, eg having a compiler installed - default is 'clang' but you can edit 'src/makefile' to change 'CC = clang' to 'CC = gcc' if you prefer.

(5.1) Server compilation

To compile the server, assuming you are in the 'tomenet' folder, switch to folder 'src':

```
cd src
```

and issue clean-compilation commands:

```
make clean
```

```
make tomenet.server
```

Now move the binary up one level so it's no longer in 'src' but in the main tomenet folder:

```
mv tomenet.server ..
```

And then switch back up one folder to the binary:

```
cd ..
```

and run it with:

```
./tomenet.server
```

Use the same instructions to recompile the server if you modified any .h-files.

(5.2) accedit compilation

To compile the 'accredit' account editor, assuming you are in the 'tomenet' folder, switch to folder 'src':

```
cd src
```

and type:

```
make accedit
```

Now move the resulting 'accredit' binary to the server's savegame folder:

```
mv accedit ../lib/save
```

and return to the tomenet folder:

```
cd ..
```

Then you can run 'accredit' by going into folder 'lib/save':

```
cd lib/save
```

and then invoking the binary:

```
./accredit
```

The reason we moved accedit to the save folder is that it must find the file 'tomenet.acc' in its folder, and tomenet.acc is located in 'lib/save'. Alternatively you can just sym-link the tomenet.acc file to any location you prefer and put accedit there accordingly, for example the root 'tomenet' folder.

=====