ZScripts User Manual (WIP manual)

July 3, 2013

1 main.json

main.json is a file you will use to configure the server's initial modloading. It will look something like:

```
{"modules": ["default"]}
```

This means to load the module "default". To load other modules, simply add them into the array, with commas to separate them, e.g.:

```
{"modules": ["default", "my_custom_module"]}
```

Keep in mind if you make your own module or modify an existing one that you need to distribute it to users of your server.

2 Command parser style

This might trip up some new users. If you're used to using bash, you should be familiar with how to use commands. Otherwise it may be somewhat confusing at first, but is actually much easier once you get used to it. Each command has an indefinite number of arguments. For example, "/ban noob player1 player2" will ban 3 players at once. This might become annoying when players have spaces or unusual characters in their name. In this case the solution is simple, surround their name with quotes ("): "/ban "player with spaces"".

This may seem tedious but when you understand the other options you will find it easier. Lets continue, most commands have optional arguments, for example the modprobe command does something very different when it's used with the "–unload" flag versus no flag at all. Flags start with "–flagname" and may have an optional "=value" or "="quoted value"".

Here is an examples of this:

```
/ban "very annoying spammer" spammer2 spammer3 -reason="spamming" -time="4 hours"
```

You can form very useful commands this way. It's much easier to understand than the spammer:4:3:true:spamming:whgat:~?! syntax some other scripts have adopted, once you get used to it.

3 /commands

Use "/commands" or "/commands –all" to get a list of commands. It has fairly complete information and always lists *every* command that's relevant, but is rather brief.

4 /modprobe

/modprobe will help you manage LSMs (loadable script modules.) By default, it will list loaded modules. If you include modules as arguments, it will list information about that module.

/modprobe has a few options as well:

–load Loads modules

–reload Loads modules, if they were already loaded then unloads them first.

–unload Removes modules

5 /ioctrl

/ioctrl provides you with a critical command to manage the script databases in a simple and straightforward manner. By default it lists the databases open. You also have a few options to apply to open databases:

–all Changes the selected databases to include all databases, not just the ones you listed manually.

–commit Commits all database changes to file.

–sync Synchronizes all databases to file. Basically the same as commit except this writes the *whole* database, not just changes. Will merge all commits into the database.

–backup Makes a copy of the entire database and writes it to a file.

Note that databases automatically commit data every once in a while
and synchronize at open and close, by default.

6   /eval

Used by nearly every script admin. This allows you run arbitrary
code on the server.

7   Security Commands

7.1   /ban

Bans usernames from the server. Options:

–reason= Reason for the ban

–time= Duration for the ban, e.g. "4 hours", or "5 days, 3 hours, 1 minute".

–ip Ban IP (disables profile bans) *unimplemented*

–name Ban name (disables profile bans) *unimplemented*

7.2   /mute

Mutes usernames in all channels. Options:

–reason= Reason for the mute

–time= Duration.

–ip *unimplemented*

–name *unimplemented*