

An Admins Manual

Noquarter and NOQ



Open Edition

by luborg

©April 13, 2011

Contents

I	Noquarter	5
1	Builtin Commands	5
1.1	readconfig	6
1.2	time	6
1.3	setlevel	6
1.4	kick	6
1.5	ban	6
1.6	banguid	6
1.7	banip	6
1.8	unban	7
1.9	put	7
1.10	pause	7
1.11	unpause	7
1.12	listplayers	7
1.13	listteams	7
1.14	mute	7
1.15	unmute	8
1.16	showbans	8
1.17	help	8
1.18	admintest	8
1.19	cancelvote	8
1.20	passvote	8
1.21	spec999	9
1.22	shuffle	9
1.23	rename	9
1.24	gib	9
1.25	slap	9
1.26	burn	9
1.27	warn	9
1.28	news	10
1.29	lock	10
1.30	unlock	10
1.31	nade	10
1.32	pip	10
1.33	pop	10
1.34	restart	10

1.35	reset	11
1.36	fling	11
1.37	throw	11
1.38	launch	11
1.39	disorient	11
1.40	orient	11
1.41	resetxp	11
1.42	resetmyxp	12
1.43	nextmap	12
1.44	swap	12
1.45	revive	12
1.46	rocket	12
1.47	disguise	12
1.48	poison	12
1.49	ammopack	13
1.50	medpack	13
1.51	pants	13
1.52	give	13
1.53	dw	13
1.54	finger	13
1.55	uptime	13
1.56	glow	14
1.57	freeze	14
1.58	unfreeze	14

II NOQ 15

2 Featurelist 15

2.1	Database Connectivity	15
2.2	Bans/Mutes	15
2.3	XPSave	15
2.4	Custom Commands	16
2.5	Greetings	18
2.6	Pussyfactor	18
2.7	The Evener	18
2.8	Selfkill restriction	19
2.9	Poll restriction	19
2.10	Offlinemessages and register command	19
2.11	Clantag/Name Protection	20

2.12 New Servercommands	21
2.12.1 csay	21
2.12.2 plock	21
2.12.3 noq_irc	21
2.13 IRC-relay	21
3 Commands	22

Part I

Noquarter

1 Builtin Commands

1.1 readconfig

Flag: G
Usage: !readconfig
Description: reloads the shrubbot config file and refreshes user flags

1.2 time

Flag: C
Usage: !time [name|slot#] (reason)
Description: displays the local time

1.3 setlevel

Flag: s
Usage: !setlevel [name|slot#] [level]
Description: sets the admin level of a player

1.4 kick

Flag: k
Usage: !kick (command)
Description: kick a player with an optional reason

1.5 ban

Flag: b
Usage: !ban [name|slot#] (time) (reason)
Description: ban a player by NAME/slot with an optional expiration time (seconds) and reason

1.6 banguid

Flag: b
Usage: !banguid [GUID] (time) (reason)
Description: ban a (perhaps not ingame) player by GUID with an optional expiration time (seconds) and reason

1.7 banip

Flag: b
Usage: !banip [IP] (time) (reason)
Description: ban a (perhaps not ingame) player by IP with an optional expiration time (seconds) and reason

1.8 unban

Flag: b

Usage: !unban [ban slot#]

Description: unbans a player specified ban number as seen in !showbans

1.9 put

Flag: p

Usage: !put [name|slot#] [r|b|s]

Description: move a player to a specified team

1.10 pause

Flag: Z

Usage: !pause

Description: pauses the game for all players

Comment: will resume game after 180 seconds

1.11 unpause

Flag: Z

Usage: !unpause

Description: unpauses the game if it had been paused by !pause

1.12 listplayers

Flag: i

Usage: !listplayers

Description: display a list of connected players, their slot numbers as well as their admin levels

Comment: get the Punkbuster slotnumber by adding 1 to the normal slotnumber

1.13 listteams

Flag: I

Usage: !listteams

Description: displays info about the teams

1.14 mute

Flag: m

Usage: !mute [name|slot#]

Description: mutes a player
Comment If no time is entered, the player is muted for g_defaultMute time.

1.15 unmute

Flag: m
Usage: !unmute [name|slot#]
Description: unmutes a muted player

1.16 showbans

Flag: B
Usage: !showbans
 (start at ban#) ((banner) (banner's name)) ((find) (banned player)) ((reason) (reason for ban))
Description: display a (partial) list of active bans

1.17 help

Flag: h
Usage: !help (command)
Description: display commands available to you or help on a specific command

1.18 admintest

Flag: a
Usage: !admintest
Description: display your current admin level

1.19 cancelvote

Flag: c
Usage: !cancelvote
Description: cancels the vote currently taking place

1.20 passvote

Flag: V
Usage: !passvote
Description: passes the vote currently taking place

1.21 spec999

Flag: P
Usage: !spec999
Description: moves 999 pingers to the spectator team
Comment also moves all players with ping higher as 500 to spec

1.22 shuffle

Flag: S
Usage: !shuffle
Description: shuffle the teams to try and even them

1.23 rename

Flag: N
Usage: !rename [name|slot#] [new name]
Description: renames a player

1.24 gib

Flag: g
Usage: !gib (name|slot#) (reason)
Description: instantly gibs a player

1.25 slap

Flag: A
Usage: !slap [name|slot#] (damage) (reason)
Description: give a player a specified amount of damage for a specified reason
Comment will never gib a player or kill him

1.26 burn

Flag: U
Usage: !burn [name|slot#] (reason)
Description: burns a player taking some of his health

1.27 warn

Flag: R
Usage: !warn [name|slot#] [reason]
Description: warns a player by displaying the reason

1.28 news

Flag: W
Usage: !news (mapname)
Description: play the map's news reel or another map's news reel if specified

1.29 lock

Flag: K
Usage: !lock [r|b|s|all]
Description: lock one or all of the teams from players joining

1.30 unlock

Flag: K
Usage: !unlock [r|b|s|all]
Description: unlock one or all locked teams

1.31 nade

Flag: x
Usage: !nade [name|slot#]
Description: makes a player drop a bunch of grenades or all players

1.32 pip

Flag: z
Usage: !pip [name|slot#]
Description: show sparks around a player or all players

1.33 pop

Flag: z
Usage: !pop [name|slot#]
Description: pops the helmets of a player or all players

1.34 restart

Flag: r
Usage: !restart
Description: restarts the current map

1.35 reset

Flag: r
 Usage: !reset
 Description: resets the current match

1.36 fling

Flag: L
 Usage: !fling [name|slot#]
 Description: flings a player in a random direction

1.37 throw

Flag: L
 Usage: !throw [name|slot#]
 Description: throws a player forward

1.38 launch

Flag: L
 Usage: !launch [name|slot#]
 Description: launch a player vertically

1.39 disorient

Flag: d
 Usage: !disorient [name|slot#] (reason)
 Description: flips a player's view

1.40 orient

Flag: d
 Usage: !orient [name|slot#]
 Description: unflips a disoriented player's view

1.41 resetxp

Flag: X
 Usage: !resetxp [name|slot#] (reason)
 Description: reset a player's XP

1.42 resetmyxp

Flag: M
 Usage: !resetmyxp
 Description: resets your own XP

1.43 nextmap

Flag: n
 Usage: !nextmap
 Description: loads the next map

1.44 swap

Flag: w
 Usage: !swap
 Description: swap teams

1.45 revive

Flag: v
 Usage: !revive [name|slot#] (reason)
 Description: revives a dead player
 Comment: doesn't work on gibbed players

1.46 rocket

Flag: j
 Usage: !rocket
 Description: have a rocket shoot from the player who uses this command

1.47 disguise

Flag: T
 Usage: !disguise [name|slot#] (class number)
 Description: gives a player a disguise of the specified class

1.48 poison

Flag: U
 Usage: !poison (name|slot#) (reason)
 Description: poisons a player

1.49 ammopack

Flag: J

Usage: !ammopack [name|slot#]

Description: an ammo pack will spawn out of the player if they have enough chargebar

1.50 medpack

Flag: J

Usage: !medpack [name|slot#]

Description: a health pack will spawn out of the player if they have enough chargebar

1.51 pants

Flag: t

Usage: !pants (name|slot#)

Description: removes a player's pants!

1.52 give

Flag: e

Usage: !give [name|slot#] item/thing

Description: gives a player something...

Comment if you specify a negativ ammount, you can also take (ammo/health)

1.53 dw

Flag: D

Usage: !dw (name|slot#)

Description: drops a player's primary and secondary weapons!

Comment best for cheaters

1.54 finger

Flag: f

Usage: !finger [name|slot#]

Description: gives specific information about a player

1.55 uptime

Flag: u

Usage: !uptime

Description: displays server uptime

1.56 glow

Flag: o
Usage: !glow (name|slot#)
Description: makes player(s) glow

1.57 freeze

Flag: E
Usage: !freeze (name|slot#)
Description: freezes player(s) move
Comment also he will not take damage while frozen

1.58 unfreeze

Flag: E
Usage: !unfreeze (name|slot#)
Description: makes player(s) moving again

Part II

NOQ

2 Featurelist

NOQ Features include, but are not limited to:

2.1 Database Connectivity

The NOQ has the possibility to use a DMBS as a backend for most of its advanced commands. This is strongly encouraged, as most features like bans, mutes and XPSave rely on this.

The configpart for the DBMS:

<pre>["dbms"]="Type", ["dbname"]="dbname", ["dbuser"]="dbuser", ["dbpassword"]="dbpass", ["dbhostname"]="dbhost", ["dbport"]="3306", ["useDB"]="1",</pre>	<p>Possible: MySQL and SQLite</p> <p>Databasename, if Sqlite then name of DBFile</p> <p>Only needed for MySQL</p> <p>Only needed for MySQL</p> <p>Only needed for MySQL</p> <p>Only needed for MySQL</p> <p>0 or 1 , 0 to disable all db interaction</p>
---	--

2.2 Bans/Mutes

If you have a DBMS activated, Mutes and Bans are automaticly active. Banning and muting is only possible trough a custom command or direct DB-Interaction till now.

2.3 XPSave

Also active as soon as a DBMS is accessible. The config has some options:

<pre>["recordbots"] = "0", ["xprestore"] = "1",</pre>	<p>This will toggle if bots are recorded into the DB. Values:(0/1)</p> <p>This will enable XPRestore out of database, wich works best on multiple servers</p>
---	---

2.4 Custom Commands

The NOQ's most useful feature are its custom commands, which are oriented at the ETAdminMod syntax, but are more powerful. They are defined in `noq_commands.cfg`

The basic Syntax is:

```
[level] - commandname = [command]
```

mind the whitespace before the level! Comments can be done via *#comment*

All players with equal or higher level will be able to use the command.

If you want ingame help for a command, you have to add a helptext (and optional the syntax):

```
[level] - commandname = [command]
help      = this is the helptext
syntax    = here goes the syntax
[level] - nextcommandname = [nextcommand]
```

Config options for Commands are:

<pre>["usecommands"] = "0", ["commandprefix"] = "!",</pre>	General switch to disable/enable the commands. Values:(0/1) the prefix used to trigger the command.
--	--

Like in ETAdminmod, you can use several placeholders, which will be replaced by their value:

Serverconsole Commands

This is the basic type of command, which is just a rework of Noquarters own custom commands. Both are directly executed at the serverconsole. Examples:

```
2 - swap                      = swap_teams
0 - beer                      = qsay A nice sexy waitress brings ^7<
    COLOR_PLAYER>^7 a nice cup of beer!
```

Shell Commands

Commands starting with `SHL` are shellcommands, and are executed in an OS-Shell. Especially handy on Linux. The output of the command is sent to the ingame chat.

```
2 - showserverinfo           = $SHL$ /usr/local/bin/show.pl <
    COLOR_PLAYER>
```

Lua Commands

Commands starting with `LUA` will be parsed and executed as Lua-code in the LuaVM, thus able to access all of NOQ's and ET's informations and data structures.

<PARAMETER>	text followed by the command. (Can be used to enter multiple values.)
<CLIENT_ID>	the client id of the calling player.
<PLAYER_CLASS>	class of calling player
<PLAYER_TEAM>	side / team of calling player
<PLAYER>	Name of the calling player (without color codes)
<COLOR_PLAYER>	Name of the calling player (with color codes)
<GUID>	Guid of the calling player
<PLAYER_LAST_VICTIM_ID>	
<PLAYER_LAST_VICTIM_NAME>	
<PLAYER_LAST_VICTIM_CNAME>	
<PLAYER_LAST_VICTIM_WEAPON>	
<PLAYER_LAST_KILLER_ID>	
<PLAYER_LAST_KILLER_NAME>	
<PLAYER_LAST_KILLER_CNAME>	
<PLAYER_LAST_KILLER_WEAPON>	
<PART2_CLASS>	class
<PART2_TEAM>	team
<PART2_CNAME>	colored name
<PART2_ID>	id
<PART2_PPID>	punkbuster slotnumber
<PART2_GUID>	guid
<PART2_LEVEL>	adminlevel
<PART2_NAME>	name without color
<PART2_IP>	IP-adress
<P1>	parameter 1
<P2>	parameter 2
<P3>	parameter 3
<PARAMETER>	all parameters concatenated

```

0 - showmaps = $LUA$ showmaps()
2 - evener = $LUA$ checkBalance(true) # this calls the Evener and
    allows him to take action
0 - cmdlist = $LUA$ listCMDs(<CLIENT_ID>, "<PARAMETER>"); # !cmdlist
    command
0 - ratio = $LUA$ et.trap_SendConsoleCommand(et.EXEC_APPEND, "chat \"
    ^2Kills^7\\^1TKS ^2" .. et.gentity_get(<CLIENT_ID>, "sess.kills")
    .."^7\\^1" .. et.gentity_get(<CLIENT_ID>, "sess.team_kills") .. "^7"
    );
# Yeah, thats a bigger one. It will show Kill/Death ratio

```

You may notice that in most examples just one function is called. Some functions are predefined in the mod and therefore these builtin commands can be easy renamed and unlocked for different levels.

2.5 Greetings

The NOQ can greet players individually, and he can announce publicly that a specific player just joined the game. Public announcements per level are saved in noq_greetings.cfg (see it for an example and customize), while the personal message is in the main config:

<code>["persgamestartmessage"] = "Welcome",</code>	the message wich is displayed for the joining player.
<code>["persgamestartmessagelocation"] = "cpm",</code>	the location where the message will be printed
<small>please note that the playername is added at the end of the string, separated by a comma.</small>	

2.6 Pussyfactor

For every kill, we add a value to the clients number, and to determine the the Pussyfactor, we divide that number trough the number of his kills multiplied with 100. If we add 100 for an MP40/Thompsonkill, if the player does only those kills he will stay at pussyfactor 1. If we add more or less(as 100) to the number, his pf will rise or decline.

Pussyfactor < 1 means he did "cool kills" = poison, goomba, knife Pussyfactor = 1 means he did normal kills Pussyfactor > 1 means he did uncool kills (Panzerfaust, teamkills, arty?)

As we add 100 for every normal kill, the pussyfactor approaches 1 after some time with "normal" kills.

Config:

<code>["pussyfactor"] = "1",</code>	enable/disable the pussyfactor Values:(0/1)
-------------------------------------	---

2.7 The Evener

NOQ's main mechanism to improve gameplay is the Evener. Basically it checks all X seconds if teams are unfair, and then notifies or takes action. Stage one is a warning combined with a request to all players to even the teams. This will happen if the teams uneven, wich means 3 or more players difference. After 3 warnings action will be taken and a random player from bigger team will be moved to the smaller team. If the difference between teams is greater or equal to 5, teams will be shuffled after 1 warning. Config options are:

<code>["evenerCheckallSec"] = "XX",</code>	time in seconds between each evenercheck. 40 - 80 is recommended
--	--

2.8 Selfkill restriction

You can restrict selfkills on your server to a specific number.

<code>["maxSelfKills"] = "X",</code>	X is the number of selfkills you want to allow. -1 to disable
--------------------------------------	---

2.9 Poll restriction

Basic restrictions for votes are also available, but a better system is in development. Till now, you have the option to define a min-distance between votes and restrict the nextmapvote to a specific time. Both restrictions don't work if the player has set the shrubbot flag "7".

<code>["polldistance"] = "XXX",</code>	time in seconds between votes, -1 to disable
<code>["nextmapVoteSec"] = "0",</code>	seconds after/before mapstart/end in which nextmap vote is allowed. 0 to disable

2.10 Offlinemessages and register command

Offlinemessages are extended private messages which are provided by NOQ. To use offlinemessages players need to register on the server via the clientconsole:

```
|/register
Syntax for the register Command: /register username password
Username is your desired username (for web & offlinemessages)
Password will be your password for your webaccess

|/register name password
Successfully registered. To reset password just re-register.
```

After registering the player can check if he has messages:

```
|/mail
No new offlinemessages
```

To write a offlinemessage use the new clientcommand "om"(analog to pm)

```
|/om
Check your syntax: '/command receiver message '.

|/om name test
Following message was sent to 'name(lastusedname)'
```

```
'test'
```

"lastusedname" is the name last used by the receiving player(wich also needs to be registered). As we just sent a message to player 'name' (wich are weself) we now should have mesages:

```
]/mail

*** NEW OFFLINEMESSAGES ***
*** MESSAGE 1***
*** From: name MSGID: XXXX ***
*** Message: test ***
```

Even if the receiving player is offline or on another server, he will receive the message. To erase use "rmom":

```
]/rmom XXXX
Erased MessageID XXX
```

Now your inbox should be empty.

If you join a server and hear the PM-messaging sound, you should check your inbox for mail.

2.11 Clantag/Name Protection

The NOQ can protect clantags and names from being used by unpriviledged players. This mechanism is automatically enabled after adding at least one protected string. To do this, add following commands to your commands.cfg:

```
[level] - protect = $LUA$ reserveName("<PARAMETER>"); #(choose level and
name to your good will)
[level] - addclan = $LUA$ addClan(<PART2IDS>, "<P2>");
```

You then will be able to reserve a string or even a lua pattern:

```
!protect etplayer
NOQ: Added 'etplayer' to the protected patterns.
```

All playernames containing the string "etplayer" will now have this part renamed to "X". To allow a player to use this pattern, add this pattern to him:

```
!addclan partofname etplayer
NOQ: Added 'etplayer' to the patterns for Fullname.
```

Fullname is now able to use the pattern 'etplayer' in his name. For the matching colorcodes are stripped from the name, but not the pattern, so don't use them. You can add multiple patterns for

a player to use. If you use a regex/lua pattern, have in mind that the added string for the addClan command has to match the regex/pattern. Oh, and if you use special chars that are no letters or numbers, watch out as they might have some additional meaning as a lua pattern matcher. Examples that can be used for reserved patterns:

```
!protect luborg # will protect the name luborg
!protect .*borg # will protect all names containing borg at the end
!protect \.es\. # will protect the .es. clantag, note the escape
                slashes for the dots
!protect .*      # will protect every name – making your server empty
```

2.12 New Servercommands

The NOQ also provides new Servercommands:

2.12.1 csay

Usage: `csay [slot#] text`

Description:

csay will print a text to the console of a player:

```
csay 0 "this is a test"
```

will print "this is a test" in the clientconsole of the player in slot 0.

2.12.2 plock

Usage: `plock [slot#] team(r/b/s) time`

Description:

"plock" is the player analogon to the !lock command for teams, except it is the other way round: The player is locked to a specific team, and won't be able to leave it. Time is the time in seconds, after wich the player is able to leave the team again.

2.12.3 noq_irc

Usage: `noq_irc text`

Description:

Will send text to the IRC-relay, wich will pipe it as an puplic chat to an IRC-channel.

2.13 IRC-relay

Noq has the option to send small texts or updates to an IRC-Relay, wich then will forward those to an IRC-Channel. The relay consists of an ruby-script wich just holds an IRC-Connection open to prevent spamming with joins/leaves during server restart.

["irchost"] = "", ["ircport"] = "1337",	host were relay is running i.e "irc.example.com" , leave blank to disable port were relay is listening
--	---

3 Commands