

The Complete Idiot's Guide to AC Servers (1.0.4)

by stef

August 4, 2009

rev 3

Contents

1	Introduction	2
1.1	What's new since 1.0.2?	2
1.2	What's with the server wizard?	2
1.3	General server setup procedure	3
1.4	Bandwidth and traffic requirements	3
2	Map Management	4
2.1	New in 1.0.4: read-only maps	4
2.2	map whitelist	4
3	Needed Files	4
3.1	config files: general structure	5
3.2	maprot.cfg	5
3.3	serverpwd.cfg	6
3.4	serverblacklist.cfg	6
3.5	nicknameblacklist.cfg	7
3.5.1	Examples	7
3.5.2	Advanced usage	8
3.6	servercmdline.txt	9
3.7	serverinfo_en.txt	9
4	Commandline parameters	10
4.1	You definitely want to use these switches	10
4.2	You will hardly want to use those switches	11
4.3	You don't want to use those switches	12
5	Hints	13
6	Error messages	13
7	Examples	14
7.1	Server-Install on a fresh Debian 5.0	14
7.2	Server logging (Linux)	14
7.2.1	syslog (default)	14
7.2.2	log to file (optional)	15
7.3	LAN server	15
7.4	rogue server	15
7.5	passworded server with multiple passwords	16
7.6	Port forwarding	16
7.7	convenient connect scripts	16
7.7.1	automatic nickname password	16
7.7.2	automatic nickname password in the serverbrowser	16
7.7.3	ask for password in the serverbrowser	17

1 Introduction

Let's start with an apology: Running an AC server is no easy business. Mostly, because some idiot keeps changing the way stuff works...

Rule No 1: Documentation is your friend! Read it!

Rule No 2: Keep your server up to date. Use version 1.0.4 - newer versions usually have fewer bugs. Also keep your custom maps up to date. Otherwise mappers will hate you ;)

Rule No 3: Be easy to contact. If something is not working right on your server, people should be able to contact you.

Configure your server carefully. Read the documentation. Test it. Ask others to test it. When your server is up and running - check the logfiles, ask others, if you don't understand something in the log file.

If you have read this and still have some questions: ask in `#assaultcube` on `irc.quakenet.org`.

1.1 What's new since 1.0.2?

If you update a server from 1.0.2 or 1.0.3 to 1.0.4, be sure to check the new features

- new in 1.0.3:
 - nickname blacklist and nickname whitelist (see section 3.5)
 - lots of additional info in the logfiles
 - check the changelog for fixed bugs - maybe you can get rid of a few workarounds
 - commandline parameters can be read from a file
- new in 1.0.4 (final)
 - provide extended serverinfo (can be displayed in the serverbrowser without connecting to the server)
 - more logging options:
 - * file log for linux
 - * timestamps for console and file logs
 - * configurable logging levels for syslog and file logging
 - * support syslog on windows and Mac
 - * Fix: fix logging on windows servers (1.0.3 bug)
 - fixed a few bugs
 - config/servercmdline.txt is read by default by server.sh and server.bat
 - config files are now checked every minute, if the size has changed
 - 'sendmap' will be rejected if the map is r/o
 - Fix: fix voting during map changes
 - Fix: fix kick & forcteam votes
 - Fix: fix vanishing flag on Mac servers

1.2 What's with the server wizard?

That's simple: don't use it. The wizard lets you fill in a few options only - most probably not all options you need. Use servercmdline.txt instead.

If you really, really, really need to use the wizard: not filling in anything will give you a working server. Every non-blank field raises the possibility, that the server won't work. Fill in stuff, that you are really sure about only.

1.3 General server setup procedure

1. Install all needed files (see section 3)
2. Edit the config files to fit your needs:
 - (a) Write admin passwords to config/serverpwd.cfg
 - (b) Write a nice maprotation to config/maprot.cfg
 - (c) Write unwanted IPs to config/serverblacklist.cfg
 - (d) Write unwanted nicknames to config/nicknameblacklist.cfg (see section 3.5)
 - (e) Add commandline parameters to config/servercmdline.txt (see section 4)
3. Make sure, your server is accessible by the intended audience (forward the correct ports!).
4. Check, if you have set the number of player slots low enough for your uplink bandwidth.
5. Start the server: execute server.bat (Windows) or server.sh (Linux)

1.4 Bandwidth and traffic requirements

Assaultcube does not need a lot of bandwidth, if you compare it to other multiplayer games - but it is absolutely required, that you make sure, that you have enough bandwidth to run a server of your desired size.

Any player will send up to 1000 bytes data plus 700 bytes protocol overhead to the server. The server will send the data of every player to all other players, so every additional player will increase the amount of data and the number of recipients. This means, that the server uplink bandwidth increases with the square of the player number. A doubled number of players will quadruple your uplink bandwidth needs.

- For n players, the average incoming (= downlink) bandwidth will be $n * 1700$ Bytes/second.
- For n players, the average outgoing (= uplink) bandwidth will be $n * ((n - 1) * 1000 + 700)$ Bytes/second.

used slots	uplink bandwidth	downlink bandwidth	monthly traffic
4	118 kBit/sec	54 kBit/sec	52 GB
5	188 kBit/sec	68 kBit/sec	77 GB
6	274 kBit/sec	82 kBit/sec	107 GB
7	375 kBit/sec	95 kBit/sec	142 GB
8	493 kBit/sec	109 kBit/sec	182 GB
9	626 kBit/sec	122 kBit/sec	226 GB
10	776 kBit/sec	136 kBit/sec	275 GB
11	942 kBit/sec	150 kBit/sec	329 GB
12	1123 kBit/sec	163 kBit/sec	388 GB
13	1321 kBit/sec	177 kBit/sec	452 GB
14	1534 kBit/sec	190 kBit/sec	520 GB
15	1764 kBit/sec	204 kBit/sec	594 GB
16	2010 kBit/sec	218 kBit/sec	672 GB
17	2271 kBit/sec	231 kBit/sec	755 GB
18	2549 kBit/sec	245 kBit/sec	843 GB
19	2842 kBit/sec	258 kBit/sec	936 GB
20	3152 kBit/sec	272 kBit/sec	1033 GB

If you host a server at home, you have to pay attention to a few more things. Don't mix up the uplink and downlink speed of your internet connection. The common ADSL line has much more downlink than uplink capability - your AC server needs much more uplink than downlink bandwidth instead. For example, if you have 3MBit-ADSL with 384kBit uplink speed, your server will be limited by the uplink speed: 7 player slots is maximum.

If your AC server is not the only one using that internet connection, you have to be careful. Do not host a server, if you use the connection for *anything* else without bandwidth limit (downloading, file sharing, other servers). Usually, it is OK, if you play AC on another server while you are hosting an own server, since the bandwidth usage of the AC client is limited. However, you should not use any VoIP services (teamspeak, mumble, etc.) while hosting your own server.

Do not host an AC server at home, if you are sharing the internet connection with others.

2 Map Management

A few things have changed, since the old 0.93 days:

- The server needs the map file like the client does (but only the *cgz*-file). The server will skip lines in *maprotation*, if the map file cannot be found.
- Voting can be restricted to maps, that are stored on the server.
- The server can distribute *cgz*- and *cfg*-files (but no skyboxes and textures).

It is very important, that the *packages/maps/official* and *packages/maps/servermaps* directories contain the right files. Be sure to check it! Section 3 will help you.

Then you have to choose a few options for your server:

- Do you want your users to be able to use own custom maps?
- Do you want your admins to be able to use own custom maps?

If you want to allow new custom maps to be used on your server, you should really create a *packages/maps/servermaps/incoming* directory. With that, only one single "sendmap" is needed for a map. If you want to restrict the addition of new custom maps to admins, you need to add *-PX* to your commandline.

Note: An admin can *always* vote for any map/mode combination he wants.

2.1 New in 1.0.4: read-only maps

A server of version 1.0.4 (or above) considers all maps in *packages/maps/servermaps* as "read only". The server will not accept the map upload via 'sendmap', if the map is already in *packages/maps/servermaps*. Only new maps and maps from *packages/maps/servermaps/incoming* will be accepted. This will prevent, that anyone can upload a modified version of a map. Please check your *incoming* directory regularly. Move welcome maps to *servermaps* to protect them. Delete unwanted/damaged versions.

2.2 map whitelist

If you use the *-PX* commandline parameter, (non-admin-) players can only vote for maps which are already stored on your server (in the directories *packages/official*, *packages/servermaps* and *packages/servermaps/incoming*). So, the combined map files in those three directories are forming the 'whitelist'. If the incoming directory exists, admins can vote for new maps and upload them via *sendmap* - so admins can add maps to your whitelist.

A map blacklist is not implemented. It would be too easy, to bypass it by renaming the blacklisted map.

3 Needed Files

The servers of version 1.0.4 need a lot of files at the right place to work fine. Some of the file-paths can be adjusted by commandline - but not all.

All paths are relative paths to the AC base directory (for example, *C:\Program files\Assaultcube*).

server.sh server.bat shell script/batch file to start the server

bin_unix/ bin_win32/ (contains the server binary)

config/ (contains the configuration files)

maprot.cfg the map rotation file

serverpwd.cfg contains admin and deban passwords

serverblacklist.cfg IPs and IP ranges that are not welcome on your server

nicknameblacklist.cfg contains black- and whitelisted nicknames

servercmdline.txt optional but very convenient way of passing commandline parameters

serverinfo.en.txt provide much more information about your server than what would fit into the MOTD

packages/maps/official (all 1.0 release maps)

This directory needs to contain all 1.0 release maps including their cfg-files. All these files are needed, but put no other maps in this directory! Be sure, not to use maps from 1.0beta packages! The maps are also included in the 1.0.4 update.

ac_arctic.cgz ac_complex.cgz ac_depot.cgz ac_desert2.cgz ac_desert3.cgz ac_desert.cgz ac_elevation.cgz ac_keller.cgz ac_mines.cgz ac_power.cgz ac_scaffold.cgz ac_shine.cgz ac_snow.cgz ac_sunset.cgz ac_toxic.cgz ac_urban.cgz ac_arctic.cfg ac_complex.cfg ac_depot.cfg ac_desert2.cfg ac_desert3.cfg ac_desert.cfg ac_elevation.cfg ac_keller.cfg ac_mines.cfg ac_power.cfg ac_scaffold.cfg ac_shine.cfg ac_snow.cfg ac_sunset.cfg ac_toxic.cfg ac_urban.cfg

packages/maps/servermaps (custom maps)

Put all custom maps (.cgz and .cfg files), that you want to use on your server, into this directory. Have at least the 1.0.4 bonus maps in this directory.

ac_depot_classic.cfg ac_depot_classic.cgz ac_douze.cfg ac_douze.cgz ac_gothic.cfg ac_gothic.cgz ac_aqueous.cfg ac_aqueous.cgz ac_wasteland.cfg ac_wasteland.cgz

packages/maps/servermaps/incoming (user provided custom maps, *optional* directory)

If you create this directory, every "sendmap" will write the received map into this directory (the server needs to have write access rights here!). If you want to allow the use of all maps on your server, you should provide this directory - so every map has to be sent only once. Maybe you should check the contents of this directory from time to time - and move good maps to packages/servermaps - and delete unwanted ones.

3.1 config files: general structure

- All config files are plain ASCII text files. Be careful, if you edit them with a full-blown office suite.
- You can't use cubescrpt in the server config files.
- A double '/' will start a comment. Comments end at the end of the line.
- TAB characters will be converted to space characters.
- Don't mix End-Of-Line character styles. Use either LF, LF+CR or CR. The rule is: if the file contains any LF characters, then all CR characters will be ignored. This should avoid any problems on Windows and Linux systems, but Mac users should be careful.
- The server checks the maprot.cfg, serverpwd.cfg, serverblacklist.cfg and nicknameblacklist.cfg every minute if their filesize has changed. If the filesize has changed, the file will be re-read. So, if you make a small change to a config file of a running server, make sure to change the filesize, so the file will be re-read.

3.2 maprot.cfg

The map rotation file uses one line per map/mode combination. One line looks like:

map:mode:time:allowVote:minplayer:maxplayer:skiplines

The server uses one line at a time, starting at the top. After the last line, the first line will be used again. A line will be skipped, if minplayer/maxplayer requirements are not met. Minplayer and maxplayer numbers are only checked when a new game is picked from the map rotation. If the player number changes during gameplay, no action will be taken.

map map filename without extension
 do only use standard maps and maps from your packages/maps/servermaps folder;
 for the flag modes (5, 14 & 15), be sure, that the map contains flag bases!

mode game mode number, one of:
 0 Team Deathmatch
 2 Deathmatch
 3 Survivor
 4 Team Survivor
 5 Capture the Flag

6 Pistol Frenzy
 9 Last Swiss Standing
 10 One Shot One Kill
 11 Team One Shot One Kill
 13 Hunt the Flag
 14 Team Keep the Flag
 15 Keep the Flag

time game time in minutes
 usually 15 minutes for flag modes, 10 minutes for all other modes

vote (1) allow players to vote for other maps
 (0) only admin can change the map

minplayer minimum number of players to start the game (optional)
 if set, the server will skip the map, if fewer than required players are connected

maxplayer minimum number of players to start the game (optional)
 if set, the server will skip the map, if more than allowed players are connected

skiplines (optional)
 lines to skip in maprot.cfg, after this map got actually played. This can be used to
 select one of several lines, instead of playing all lines.

Examples:

```

// dm, 8 minutes, 0..3 players
ac_scaffold : 2: 8: 1: 0: 3
// ctf, 12 minutes, any player number
ac_power : 5:12: 1
// ac_desert, 10 minutes, either dm, 0..4 players or tosok, 5.. players
ac_desert : 2:10: 1: 0: 4: 1 // skip 1 line, if played
ac_desert :11:10: 1: 5: 0: 0
  
```

See file config/maprot.cfg for more example lines.

3.3 serverpwd.cfg

Contains a list of admin and deban passwords. One password per line. One line looks like:

password denyadmin

password cannot contain '/' or whitespace characters; should not have more than 100 characters

denyadmin (optional)

if '1', the password can only be used to connect to the server in case of ban, not to claim admin

Examples:

```

foo // admin password 'foo'
bar 0 // admin password 'bar'
foobar 1 // deban password 'foobar', can not be used with 'claimadmin'
  
```

3.4 serverblacklist.cfg

List any IPs and IP ranges that are not welcome on your server in this file. Three possible line formats are accepted:

1.2.3.4 ban a single IP

1.2.3.4/24 ban an IP block (1.2.3.0 - 1.2.3.255)

1.2.3.4 - 2.3.4.5 ban an IP range

The server will automatically join overlapping ranges and remove double entries.

Note: Two IPs are hardcoded blacklisted - so the blacklist will at least contain two entries.

3.5 nicknameblacklist.cfg

This file configures the nickname blacklist and the nickname whitelist. The following commands are accepted:

```
accept nickname [IP range ...] [password ...] add nickname to the whitelist
    IP ranges and passwords are optional;
    if IP ranges are specified, only connects from that ranges are allowed; if one or more
    passwords are specified, the connect password has to match one of them

block nicknamefragment1 [nicknamefragment2 ...] block nicknames that contain all specified frag-
ments (up to 5)

blocki nicknamefragment1 [nicknamefragment2 ...] block nicknames that contain all specified frag-
ments (up to 5), ignore cases
```

- accept, block and blocki can be shortened to a, b and bi
- 'accept' has higher priority than 'block'
- accept matches whole nicknames; block matches parts of nicknames
- accept and block commands are case sensitive; the blocki command ignores cases
- the order of lines/commands in the file is irrelevant; no sorting or grouping is required
- 'accept' commands can be split up into several commands by using the same nickname

Whenever a player connects (or changes his nickname), the nickname is compared to the whitelist and the blacklist. First, the nickname (as a whole) is searched in the whitelist. If the nickname is in the whitelist, the player's IP and password are checked (if required). If an IP range or password requirement is not met, the player gets kicked. If the nickname is not in the whitelist, it is checked, if any blacklist entries match. If a blacklist entry matches, the player gets kicked.

To prevent whitelist entries to be misused by other players, each entry can be secured by one or more IP ranges and one or more passwords. IP ranges are very convenient to use. For players with static IP, this is a very secure method to prevent others from using the nickname. If the IP is dynamic, it depends on how big the used IP range is. A single /16 range should be secure enough for most cases. If the range is bigger (yep: brazil), the IP range will not block others sufficiently. In those cases, passwords can be used.

To connect to a server, where your nickname requires a password, use this command to connect:

```
/connect SERVER-IP PORT password
```

If the server is using the default ports, you can write '0' instead. Because the password method is very inconvenient for the player, it should only be used if really necessary. If the player should also be able to claim admin while connecting (by connectadmin), the passwords for nickname and admin have to be identical.

Note: It is not possible, to change the nickname to a password protected one, while you are connected to the server. There is no way to specify the password without reconnecting. If the server is password protected (commandline parameter '-p'), connecting with a password protected nickname is only possible, if the nickname password is identical to either the server password or an admin password.

3.5.1 Examples

Simple whitelisted nickname, no IP restrictions or passwords, no blacklist entries can block this name:

```
accept lorem
```

accept 'ipsum' only, if he connects with an IP in the range 192.168.0.0-192.168.255.255

```
accept ipsum 192.168.0.0/16
```

add another allowed IP range for 'ipsum':

```
accept ipsum 10.0.0.0/8
```

add two possible passwords for 'ipsum':

```
accept ipsum foo bar
```

combine the three ipsum lines in one line:

```
accept ipsum foo bar 192.168.0.0/16 10.0.0.0/8
```

block all nicknames that contain the substring 'dolor', for example 'dolores' or 'odolor'; do not block 'DOLOR':

```
block dolor
```

block all nicknames that contain both substrings, for example 'sitamet' or 'dametosit':

```
block sit amet
```

block all nicknames that contain the substring 'consectetur', ignore case; block for example 'aconsectetur', 'oConSecTeTuR':

```
blocki consectetur
```

block any combination of '1', '2', '3' but allow '123' (block for example '1231', '321', '2x1x3x2x1' etc.)

```
block 1 2 3
accept 123
```

3.5.2 Advanced usage

Ban a whole nickname, not a nickname part

To ban a whole nickname only (and not all the nicknames, that just contain that name), use the whitelist and set an impossible IP address (or password). For example, to ban the player 'A' but not all other players containing an 'A' (like 'block A' would do), write

```
accept A 0.0.0.1 // block player 'A'
```

Clan tag protection

To allow only clan members to use the clan tag, you have to use black- and whitelist entries. Block the usage of the clan tag but allow clan members. Since the whitelist has higher priority than the blacklist, this will work fine.

Clan whitelist example: a clan named 'DEV' with two members, 'foo' and 'bar'

```
accept DEV|foo 1.2.3.4 // foo has a static IP; no password required
```

bar has a dynamic IP from that range; in addition, the connect password has to match 'pwdpwd':

```
accept DEV|bar 192.168.0.0/16 pwdpwd
```

don't allow anyone else to wear the clan tag:

```
block DEV|
```

or similar clan tags:

```
blocki DEV|
blocki {DEV}
blocki *DEV*
```

Hint: Don't be careless with block statements. For example: 'blocki DEV' would also block 'devon' and 'Devon'. Check the logfile regularly for rejected nicknames.

3.6 servercmdline.txt

By using the '-C' commandline parameter, you can read commandline parameters from a file.

- one commandline parameter per line only
- you have to use the '-' characters for the switches
- you can separate the switch and it's argument by whitespace
- the file is only read once at server startup
- changing the file's content will not affect a running server
- no file is read by default

If you start a windows server by executing server.bat, or a linux server by executing server.sh, the server will read commandline parameters from config/servercmdline.txt (for example: server.bat contains 'ac_server.exe -Cconfig/servercmdline.txt %1 %2 %3 %4 %5').

See section 4 or docs/commandline.html for a full description of commandline parameters. The file config/servercmdline.txt also contains brief descriptions of many commandline parameters.

Note: commandline parameter files can be nested (up to three levels)

Examples:

```
-o \f3example MOTD\n\f2with 2 lines and \f43 colors
-D 5 // never disable this...
-n \fs\f2foobar\fr server // server description
-n1 * // custom server description prefix
-n2 *@\f2foobar // custom server description suffix
-c 12 // max client number, 1..20, default 6
-T // add timestamps to console and file logs

// import commandline options from file (can be used recursively)
-C config/servercmdline_common.txt
```

3.7 serverinfo_en.txt

Use this file to provide additional information about your server. This text file is sent, whenever a client requests extended server information. Comments are removed, also leading and trailing blanks and empty lines. To send an intentionally left blank line, use a line with a single dot ('.'). Don't use more than 80 characters per line. You can use colors and TABs.

Please use this file to provide contact information, especially for players who are innocently affect by an IP range ban. You can also use this page to advertize your clan or provide contact information for recruiting.

Please provide at least contact information like an email address or an IRC channel.

You can also provide information in different languages. If you want, for example, to provide an alternate file for players with french language, you should create a file named *config/serverinfo_fr.txt*. For a list of all language codes, see http://en.wikipedia.org/wiki/List_of_ISO_639-1_codes. Please only use those ISO 639-1 codes. The players that want to request a language different from the default english, have to set the *LANG*-alias, for example:

```
/LANG = fr (press 't' and type this into the game console)
```

If a player requests a language file that does not exist on a server, the english version will be sent instead.

The server only reads the files once. The english version is read during startup, the other versions will be read when they are requested the first time. If you want to change the serverinfo, you have to restart the server. Sorry.

Game clients will only fetch the info once per server (select the server and press F9). To reread the info (for example, if you've changed the LANG-alias) you have to refetch the server list from the masterserver (press F5 in the serverbrowser).

Example:

```

\f1Welcome to \f3A&D\f2:\f3Flopsy 1.0.4\f1.
Have fun and play nicely :)
Report abuse to #and-clan@quakenet.
.
If you are banned from this server and don't
know why, contact \fs\f0stef\fr on #and-clan@quakenet.
.
If you want an admin password for this server,
contact \fs\f0stef\fr on #and-clan@quakenet.
.
If you want to join A&D, you're out of luck: we're not recruiting.
But you can hang out with us on #and-clan@quakenet.

```

If you want the server to read the extended information from a different path, you can use the *'-If'* command line switch. This will set the serverinfo text filename prefix. The default is *-Iconfig/serverinfo*. The server adds *".en.txt"* to the prefix to complete the filename.

4 Commandline parameters

(see also http://actiongame.svn.sourceforge.net/viewvc/actiongame/branches/v01_00_xx/ac/docs/commandline.html or docs/commandline.html)

Read this, to learn, which switches you want to use - and which you don't ;)

4.1 You definitely want to use these switches

- D Enables permanent demo recording. This switch is obligatory. You need a very good reason *not* to use it. To change the default of 5 demos in RAM, you can add the number of demos to the parameter, e.g. *-D5*

- nS Sets the server description. Keep it short, i.e. *-nBobsInstagibServer*. No need, to mention your mother's maiden name here.

- n1S Sets a server description prefix. *-n2S* Sets a server description suffix.
 To enable the admin to temporarily change the server description, a server description prefix and/or suffix needs to be set. If the admin executes the command */serverdesc DESC*, the server description will be changed to *"prefixDESCsuffix"*. At the end of the game, the default server description (set by *-n*) is restored automatically. Please set at least a suffix, like *-n2@BobsInstagibServer*, so the server description can be set to *"cup_match@BobsInstagibServer"* for a few games.

- fPORT Sets the server base port to PORT. An AC server always uses 2 UDP ports: PORT & PORT+1. The default ports are 28763 and 28764. You need to use different ports, if you want to run several servers on the same machine (with one IP address). For example, use *-f28765* for a second server. Do not use this switch, if you want to run only one server.

- cN Sets the maximum number of players that can play on this server to N (default 6). If more players try to connect they will get a "server full" message. Choose carefully - don't allow more than 20 players.

- oN Sets the server's "message of the day" (MOTD). Include a way to contact you, like an email address or an IRC channel.

- PM Sets the required role to call specific votes. The two possible player roles are 'player' and 'admin'. 'M' is a string of characters to set vote role requirements per vote type. A lower case character enables the vote call for normal players, an upper case character sets the vote call to require an admin. If a character does not appear in the string, the default setting is used.
 The following votes can be adjusted:
 - k kick players
 - b ban players (and remove bans)

m	change mastermode (open/private)
f	force a player to the other team
a	enable/disable autoteam
s	shuffle the teams
r	record a demo of the next game
c	clear all demos on server
d	change server description (needs -n1 or -n2 set)
x	vote for maps, that are not present on the server
p	vote for a gamemode that is not supported by the map (like ctf ac_desert)

For example: -PbmAX enables ban & mastermode to be voted by normal players but only admins can change autoteam or vote for nonexistent maps. The default settings are: "fkBMasRCDxP". You can use several -P switches on your commandline - their content will be concatenated. Don't use one character with both upper and lower case, as this is undefined!

- Wf Sets a path (and optional filename prefix) to write all recorded demos to. The example -Wserverdemos/mydemo would save all demos to the directory 'serverdemos' and prefix the filenames with 'mydemo'. The filename will contain the (optional) prefix, the (encoded) time, game mode and map. To set a path without filename prefix, use -Wpath/. A prefix without path can be set by -Wprefix. The server owner has to take precautions that the demo files don't fill up the filesystem - especially, if permanent demo recording is also enabled (switch '-D').
- V Sets the server to verbose mode. This enables additional log messages to be printed, including the server configuration files (after parsing). Be careful: the logfile will contain all content from serverpwd.cfg. Use this switch at least once after making changes to config files like the maprotation - to check, if the file is read properly.
- T Add timestamps to every line of the console and file logs. Use this, if you're logging to file.

4.2 You will hardly want to use those switches

- xM Write your password to the file "config/serverpwd.cfg" instead.
- mM If you want your server to be private and not report to the masterserver, use -mlocalhost. Only very few servers will need this.
- pPASS Sets the server password to PASS. Only clients that use the command "/connect IP PORT PASS" will be able to connect. Only very few servers will need this.
- Xf -rC -Bf -If Use the default file locations, unless you have a really good reason to change them.
- kN -yN Stay with the default values here...
- LFn Sets the logging level for file logs to level n. On linux systems, this switch can be used to enable (additional) file logging. To enable logging to file (level VERBOSE), use for example "-LF1". The following levels can be selected:

0	DEBUG: logs all messages
1	VERBOSE: logs all messages of level VERBOSE and above (same as console log, when using the -V switch)
2	INFO: logs all messages of level INFO and above (same as console without -V switch), default on windows or mac systems
3	WARNING: log only messages of level WARNING and above
4	ERROR: log only messages of level ERROR
5	(or above) do not log to file (default on linux systems)

- LSn Sets the logging level for syslog logging to level n. This switch is ignored on non-linux systems.
 Example: disable syslog (linux only) “-LS5”.
 The following levels can be selected:
- | | |
|---|---|
| 0 | DEBUG: logs all messages |
| 1 | VERBOSE: logs all messages of level VERBOSE and above (same as console log, when using the -V switch) |
| 2 | INFO: logs all messages of level INFO and above (same as console without -V switch), default |
| 3 | WARNING: log only messages of level WARNING and above |
| 4 | ERROR: log only messages of level ERROR |
| 5 | (or above) do not log to syslog |
- Amap Use this, to disable manual voting for a whitelisted map for non-admins. Use separate switches for several map names. You can still use those maps in the maprotation.
 Example: disable voting for ac_desert and ac_snow “-Aac_desert -Aac_snow”.

4.3 You don't want to use those switches

- d You don't need this, because you want to use the server binary.
- uN Don't use this switch.
- iIP Don't use this switch. Use -f.
- NID Don't use this switch (unless you want to have your printout of /var/log/syslog look more interesting).
- SW Use a real operating system :)

5 Hints

- Voting for maps is difficult sometimes. Create a good maprotation to minimize the need for map votes.
- Check out the possibilities of the -P switch. Every server may need different settings: for example, if there are only few admins around on your server, you may want to enable ban voting, to allow the players to get rid of cheaters themselves. If you are running a match server, be sure to allow the changing of the server description and maybe the private mode.
- If you use the -PX switch, your server is using the contents of packages/maps/official, servermaps and incoming as 'map whitelist' for voting. If you create the incoming directory, admins can add maps to the whitelist by voting for them once (and 'sendmap' them). *This is a nice feature - use it.*
- Don't be frugal with your serverpwd.cfg. If you hand out admin passwords to many people, use different passwords. If you hand out 'deban' passwords (and you should do that, if you use the blacklist), then use different passwords for EVERY user. That way you can be sure, you know, who to blame, if a password gets misused.
- Make an effort with your map rotation. Use the default one as an example. Make sure, that the game mode matches the number of players. Try to alternate game modes (even if the player number is constant). Don't run the same map several times in a row. Only use playable map/mode combinations (if the players want osok ac_mines, they can vote for it). Don't put unfinished or unfitting (jump & gema) maps on your rotation. Use at least four different maps in a rotation and more than one gamemode.
- If you use the blacklist, make sure, that blacklisted players have a way of contacting you. Don't hesitate to hand out deban passwords for players in blacklisted IP ranges.
- Check your logfiles regularly. Check for rejected nicknames, blacklisted IPs, attempted cheating, votecall abuse, spamming, etc.

6 Error messages

Here's a short list of error messages and a few guesses, why it may show up:

server ip not resolved Whatever you're trying to use '-i' for - it's not working. That's your fault: don't use '-i'.

could not create server info socket Failed to open port "base+1" (default 28764). Did you use '-f' to specify a 'cool' but non-working port number? Is the port already in use? For example, by another AC server? Do you need special permission to open a listening port? Is a software firewall blocking you?

could not create server host Failed to open "base+0" port (default 28763). Is already another (AC) server running? Did you use '-f' without knowing why?

could not create LAN server info socket This is a warning only. This server will not show up in a LAN serverbrowser. Usually happens, if you run several servers on a Mac. Connect to this server manually...

could not open socket, could not connect Do you even have an internet connection? Is a firewall blocking you completely?

masterserver reply: Server not registered: Could not ping you. Make sure your server is accessible from the internet and forward your damn ports. This is *always* the cause: you did not forward your ports!

failed to open "serverlog.." for writing Don't use funny characters in your server identity ('-N'). Your OS may not like them in filenames.

7 Examples

7.1 Server-Install on a fresh Debian 5.0

needed packages to build the server binary

(using the `server_install` make target):

`build-essential`

`libz-dev`

building:

- unpack the AC archive
- cd to `source/src`
- make `server_install`
- add/edit commandline parameters in `config/servercmdline.txt`
- edit config files (`serverpwd.cfg`, `serverblacklist.cfg`, `maprot.cfg`, `nicknameblacklist.cfg`)
- test run with additional `-V` parameter, check log if the `cfg` files are correct

```
./server.sh -V
```

- final run

```
./server.sh
```

7.2 Server logging (Linux)

7.2.1 syslog (default)

On Linux machines, AC is using the `syslog` service to write its logs. By default, this can get a little messy, because all AC log data gets written to `/var/log/syslog` and `/var/log/messages`. To prevent writing to those two files you have to make a few changes to your `syslog.conf` (usually located in `/etc`, but may also be named `rsyslog.conf` - check which `syslog` version is installed, and where the file is located on your system).

In `syslog.conf` there will be two lines much like:

```
*.*;auth,authpriv.none          -/var/log/syslog

*.=info;*.=notice;*.=warn;\
    auth,authpriv.none;\
    cron,daemon.none;\
    mail,news.none                -/var/log/messages
```

AC is using `syslog` facility `'local6'` (by default; you can change this with the `'-F'` parameter), so we have to filter `'local6'` from those two channels (notice the two added `'local6'`-parts):

```
*.*;auth,authpriv,local6.none    -/var/log/syslog

*.=info;*.=notice;*.=warn;\
    auth,authpriv.none;\
    cron,daemon.none;\
    mail,news,local6.none         -/var/log/messages
```

To get `syslog` to write all `local6`-messages to a file named `/var/log/aclog`, we have to add the following line:

```
local6.*                          /var/log/aclog
```

Now, you have to get syslog to re-read the config file. An easy but brute-force method would be to restart the machine. You can also send SIGHUP to the syslogd process. Use 'ps aux' to find out the pid of your syslogd process, then 'kill -HUP pid' to send the signal. After this, there should be a new file '/var/log/aclog'. Start an AC server or type 'logger -p local6.info testmessage' to write some log data to the new file.

Note: Many Linux systems activate logfile-rotation for all files specified in syslog.conf. So, after some time, the contents of aclog will be moved to aclog.0 (the former content of aclog.0 is moved to aclog.1 and so on).

Note: If you just add the "local6.* /var/log/aclog"-line to syslog.conf, you will get AC logs in /var/log/syslog, /var/log/messages and /var/log/aclog.

7.2.2 log to file (optional)

You can disable the logging to syslog by adding '-LS5' to your commandline. To enable file logging, add '-LFn' to the commandline, with n being the chosen log level (0:DEBUG, 1:VERBOSE, 2:INFO, 3:WARNING, 4:ERROR, 5:DISABLED). Since file logs don't have timestamps by default, you should also add '-T' to the commandline.

```
// add this to servercmdline.txt to switch from syslog to file logging
-LS 5 // disable syslog
-LF 2 // log level 2 (INFO) and above to file
-T // add timestamps to console and file logs
```

Note: If you disable logging completely, please don't announce your server to the official masterserver. We don't want servers without logging at least level 2 in the list. Logs are very helpful to detect cheaters (and other unwanted persons). Without logs, you can't check, if your server is working properly. For example: please check regularly if one of the blacklists (IP or nickname) is going postal...

7.3 LAN server

If you don't want an AC server to register itself on the masterserver, it is usually sufficient, to disable portforwarding in your router (or use port numbers that are not forwarded). Without portforwarding, the masterserver-registration will fail. Another method is, to provide an invalid masterserver URL on the commandline, like '-mlocalhost'.

Independent from the masterserver, every AC server supports searching LAN servers via IP-broadcast. For that, every server listens on port 28762 for broadcasted pings (by AC clients). So, to set up a LAN game, you just have to make sure, that the server and all clients are in the same IP broadcast domain (a.k.a subnet). For example:

```
192.168.1.1      Internet gateway / router address (irrelevant)
255.255.255.0   subnet mask (subnet range 192.168.1.0-192.168.1.255)
192.168.1.10    server1 IP
192.168.1.11    server2 IP
192.168.1.12    client1 IP (ok)
192.168.1.13    client2 IP (ok)
192.168.1.14    client3 IP (ok)
192.168.5.12    client4 IP outside subnet - can't ping the servers
```

7.4 rogue server

Of course, an AC server can be used without neither masterserver nor LAN server search. Just disable masterserver-registration (commandline parameter '-mlocalhost'). Every client has to use the 'connect' command with the server's IP and port numbers:

```
/connect IP port
```

This method does not depend on subnet ranges. However, the client has to be able to connect to the server ports, so port forwarding may still be needed.

7.5 passworded server with multiple passwords

If a server password is specified (commandline parameter '-p'), only clients using the server password or any admin password can connect to the server. You can't connect to the server with a deban password.

Only one server password can be specified. If you want several different passwords to be able to connect to your server without giving the players the possibility to claim admin, you have to use a different method:

1. don't specify a server password at all
2. ban all players from the server by adding a line '0.0.0.0-255.255.255.255' to the file config/serverblacklist.cfg
3. give admin passwords to the users that you want to be able to connect and claim admin
4. give deban passwords to the users that you want to be able to connect, but *not* to claim admin

7.6 Port forwarding

Sry. I forgot to write this section. And what makes it even worse, is the fact, that there is no information on this topic available on the net.

But:

- If your server can't register at the masterserver, you did not forward port 28764 (or "custom port" + 1).
- If your server registers fine at the masterserver and shows up in the serverbrowser but does not respond to connection attempts, you didn't forward port 28763 (or whatever "custom port" you've chosen).

7.7 convenient connect scripts

A few scripts to improve your autoexec.cfg.

7.7.1 automatic nickname password

This script implements the new command '/nwconnect ip port' which automatically searches a password (matching to the server IP) in a list. Put your server IPs and passwords in that list.

```
alias nicknamepasswordlist [
81.89.107.97 ig_pwd // test password for nickname 'IdiotsGuide' on A&D servers
192.168.1.10 hitfan // example password for a local server
]

alias nwconnect [
  alias _npwdpos [ findlist $nicknamepasswordlist $arg1 ];
  if (!= (_npwdpos) -1)[
    alias _npwd (at $nicknamepasswordlist (+ (_npwdpos) 1) )
  ][
    alias _npwd $arg3
  ];
  connect $arg1 $arg2 $_npwd
]
```

7.7.2 automatic nickname password in the serverbrowser

Use the 'automatic nickname password'-script from above and bind the new nwconnect to the serverbrowser:

```
alias sbconnect [nwconnect $arg1 $arg2 $arg3]
```


7.7.3 ask for password in the serverbrowser

Use this version of sbconnect to get asked for password whenever the server wants one. If you put known passwords (server, admin & deban) in the adminpasswordslist, the script will automatically fill in the password form.

1. If the server needs a password, because it's full, private or passworded (or you're banned from the server), the script will popup a menu and ask for the password. If a password is found in the adminpasswordslist, the form will be filled in already.
2. If the server doesn't need a password (not full, not private, not passworded, not banned or blacklisted), the script will use nwconnect (see above) to find a nickname password for the server. If a nickname password is found, it will be used silently.
3. If no password is needed (1) or found (2), the script will just connect to the server without password. Please note, that the script can't determine, if you'll need a nickname password for your nickname on that server.

```
// convenient connecting to protected servers from the serverbrowser
alias adminpasswordslist [
192.168.1.10 admpass // example password for a local server
]
newmenu "connect with password"
menuitemvar "result $sbc_1" " "
menuitemtextinput "password: " "result $sbc_pwd" "sbc_pwd = $arg1" "" 30
menuitem "--> [connect]" [connect $sbc_ip $sbc_port $sbc_pwd; sbc_pwd = ""]
menuitemvar "result $sbc_2" " "
menuitemvar "result $sbc_3" " "
alias sbc_pwdtypes [" server or admin" " deban or admin" "n admin"]
alias sbconnect [
  alias sbc_pt -1;
  if (= $arg4 1) [sbc_pt = 0; sbc_2 = "the server is password protected"];
  if (!= $arg6 0) [sbc_pt = 1; sbc_2 = "you are banned from this server"];
  if (= $arg5 1) [sbc_pt = 2; sbc_2 = "the server is private"];
  if (= $arg3 1) [sbc_pt = 2; sbc_2 = "the server is full"];
  if (= $sbc_pt -1) [
    nwconnect $arg1 $arg2
  ]
  alias _pwdpos [ findlist $adminpasswordslist $arg1 ];
  if (!= (_pwdpos) -1)[
    sbc_pwd = (at $adminpasswordslist (+ (_pwdpos) 1) )
  ]
  sbc_pwd = ""
];
sbc_1 = (concatword "server: " $arg7 " " $arg1 ":" $arg2);
sbc_3 = (concatword "you need a" (at $sbc_pwdtypes $sbc_pt) " password");
sbc_ip = $arg1; sbc_port = $arg2;
showmenu "connect with password"
]
]
```