FTP Daemon (legacy, no longer maintained)

Marc Huber

COLLABORATORS

	TITLE : FTP Daemon (legacy, no longer maintained)	

ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Marc Huber	November 17, 2026	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1 Introduction **1**

1.1 Download . . . 1

2 Supported commands **1**

3 Operation **2**

3.1 Command line syntax . . . 2

3.2 Signals . . . 2

3.3 Event mechanism selection . . . 3

4 Configuration directives **3**

4.1 Global Configuration . . . 3

4.1.1 Access Control Lists . . . 4

4.2 ACL-based Configuration . . . 6

4.3 Path-rewriting using PCRE . . . 10

4.4 TLS support . . . 10

4.5 MAVIS Configuration . . . 11

5 Wildcard patterns **11**

6 Magic cookie substitution **11**

7 Sample configuration **12**

8 Railroad Diagrams **13**

9 Bugs **17**

10 References **17**

11 Copyrights and Acknowledgements **18**

1 Introduction

This FTP daemon was written from scratch. The list of supported features includes:

- Small memory footprint
- Event-driven, pre-forking
- Not called by inetd
- Supports traffic shaping
- Highly configurable using access control lists for commands and configuration variables
- Utilizes the MAVIS modular authentication system
- A couple of **wu-ftpd**-like features (banners, checksum calculation, ...) are available
- DNS resolving is done if the daemon is compiled with *c-ares* support
- Asynchronous RFC1413 ident lookups
- Large File support.
- 64bit clean

1.1 Download

You can download the source code from the GitHub repository at https://github.com/tacplus-ng/event-driven-servers/. On-line documentation is available via https://tacplus-ng.github.io/event-driven-servers/doc/index.html/, too.

2 Supported commands

The daemon support several standards and drafts:

- Standard RFC959 FTP commands supported are:

```
ABOR, APPE, CWD,  CDUP, DELE, HELP, LIST, NLST, MDTM, MKD,  NOOP, PASS,
PASV, PORT, PWD,  QUIT, REIN, REST, RETR, RMD,  RNFR, RNTO, SITE, SIZE,
STAT, STOR, STOU, SYST, TYPE, USER, XCUP, XCWD, XMKD, XPWD, XRMD
```

- IPv6 support is available. Both the RFC1639 (aka. FOOBAR) extensions (`LPRT`, `LPSV`) and the more recent ones defined in RFC2428 (`EPRT`, `EPSV`) are supported.
- The feature negotiation commands `FEAT` and `OPTS` introduced in RFC2389 are supported.
- The command `LANG` (RFC2640) allows negotiation of a language for greetings and error messages. Currently supported languages include English and German.
- RFC4217 (Securing FTP with TLS) is supported, If the daemon was compiled with TLS support. `AUTH TLS` et al. may then be used to switch to a secure channel; certificate authentication is supported. This may or may not be legal in your country
- `MDTM` and `SIZE` aren't specified in RFC959, but may become part of a revised FTP specification.
- `MLST` and `MLSD` are supported, but the specification is still in draft status.
- The proposed fact modification commands `MFMT` and `MFF` are supported.
- Virtual host support is available using the `HOST` command (requires explicit support via **MAVIS** backends).

- The experimental commands `ESTA` and `ESTP` are available.
- `MODE Z` enables deflate transmission mode. Alternatively, just add `.gz` to a file name for on-the-fly compression.

Various `SITE` commands are available:

- `SITE CHMOD` changes permission modes.
- `SITE GROUP` may be used to switch to another group id.
- `SITE GROUPS` displays the available group ids in **wu-ftpd** style.
- `SITE ID` displays both user id and the available group ids.
- `SITE IDLE` displays or changes the idle timeout.
- `SITE UMASK` displays or changes the current **umask**.
- `SITE CHECKMETHOD` selects a checksum method (either CRC or MD5), as does `OPTS HASH`.
- `SITE CHECKSUM` calculates and displays checksum values, as does `HASH`. The `RANG` command for specifying byte ranges is supported. **wu-ftpd**-like file conversions for `.md5` and `.crc` are implemented.
- `SITE HTPWD` may be useful for maintaining .htpasswd compliant password files.
- `SITE HELP` or `SITE HELP` *COMMAND* display information about available commands and command syntax.

3 Operation

This section gives a brief and basic overview on how to run **ftpd**.

In earlier versions, **ftpd** wasn't a standalone program but had to be invoked by **spawnd**. This has changed, as **spawnd** is now part of the **ftpd** binary. However, using a dedicated **spawnd** process is still possible and, more importantly, the **spawnd** configuration options and documentation remain valid.

ftpd may use auxilliary **MAVIS** backend modules for authentication and authorization.

3.1 Command line syntax

The only mandatory argument is the path to the configuration file:

```
ftpd [ -P ] [ -d level ]  [ -i child_id ] configuration-file [ id ]
```

If the program was compiled with CURL support, *configuration-file* may be an URL.

Keep the `-P` option in mind - it is imperative that the configuration file supplied is syntactically correct, as the daemon won't start if there are any parsing errors at start-up.

The `-d` switch enables debugging. You most likely don't want to use this. Read the source if you need to.

The `-i` option is only honoured if the build-in **spawnd** functionality is used. In that case, it selects the configuration ID for **ftpd**, while the optional last argument *id* sets the ID of the **spawnd** configuration section.

3.2 Signals

Both the master (that's the process running the **spawnd** code) and the child processes (running the **ftpd** code) intercept the `SIGHUP` signal:

- The master process will restart upon reception of `SIGHUP`, re-reading the configuration file. The child processes will recognize that the master process is no longer available. It will continue to serve the existing connections and terminate when idle.
- If `SIGHUP` is sent to a child process it will stop accepting new connections from its master process. It will continue to serve the existing connections and terminate when idle.

3.3 Event mechanism selection

Several level-triggered event mechanisms are supported. By default, the one best suited for your operating system will be used. However, you may use the environment variable `IO_POLL_MECHANISM` to select a specific one.

The following event mechanisms are supported (in order of preference):

- port (Sun Solaris 10 and higher only, `IO_POLL_MECHANISM=32`)
- kqueue (*BSD and Darwin only, `IO_POLL_MECHANISM=1`)
- /dev/poll (Sun Solaris only, `IO_POLL_MECHANISM=2`)
- epoll (Linux only, `IO_POLL_MECHANISM=4`)
- poll (`IO_POLL_MECHANISM=8`)
- select (`IO_POLL_MECHANISM=16`)

Environment variables can be set in the configuration file at top-level:

```
setenv IO_POLL_MECHANISM = 4
```

4 Configuration directives

Several configuration options are very similar in syntax. For that reason, I'll use a couple of shortcuts below:

- **Boolean**: `yes`/`permit` or `no`/`deny`
- **Path**: A valid file file path on your system.
- **Number**: A positive integer number.
- **Directory**: A valid directory path on your system.
- **CIDR**: A single IP address or network the latter in Classless Inter-Domain Routing notation (*Address/MaskLength*).

4.1 Global Configuration

The following table summarizes configuration options with plain

Variable = Argument

syntax:

Variable	**Description**	
`mimetypes`	This specifies the path to a `mime.types` file. Mime-types are used for the *media-type* fact in `MLST`/`MLSD` replies.	
	Type of Argument	*Path*
	Default Value	*none*
	Example:	
	`mimetypes = /etc/mime.types`	
`buffer size`	Permits tuning of buffer allocation size.	
	Type of Argument	*Integer*
	Default Value	`32k`

Variable	Description	
`buffer mmap-size`	Permits tuning of buffer allocation size. Setting *mmap-size* to 0 will cause whole files to be memory-mapped. However, if you do so on a 32bit system, it may run out of address space.	
	Type of Argument	*Integer*
	Default Value	`256k` (on 64bit systems: unlimited)
`hide-version`	This options controls whether the daemon will omit its version number in the `HELP` response.	
	Type of Argument	*Boolean*
	Default Value	`no`
`retire`	If set, the daemon will terminate after processing *count* sessions, what may be useful to remedy the effects of memory leaks.	
	Type of Argument	*Integer*
	Default Value	*unset*
`log-format command`	Sets format for logging to syslog.	
	Type of Argument	*String*
	Default Value	`"CMD|%i|%r|%I|%t|%u|%C|%c"`
`log-format event`	Sets format for logging to syslog.	
	Type of Argument	*String*
	Default Value	`"EVE|%i|%r|%I|%u|%t|%d"`
`log-format transfer`	Sets format for logging to syslog.	
	Type of Argument	*String*
	Default Value	`"XFR|%i|%r|%I|%t|%u|%d|%m|%b/%s|%`
`log-format delimiter`	All occurrences of the *delimiter* character will be replaced by the *substitute* character before logging.	
	Type of Argument	*Character*
	Default Value	`"|"`
`log-format substitute`	All occurrences of the *delimiter* character will be replaced by the *substitute* character before logging.	
	Type of Argument	*Character*
	Default Value	`"_"`
`nlst`	This directive may be used to limit output of the `NLST` command to regular files. It is provided for **wu-ftpd** compatibility.	
	Argument	`files-only`
	Default Value	*unset*
`use-mmap`	On systems supporting memory-mapped I/O, the daemon may use `mmap`(2) for read-only file access. Preliminary tests indicated that `mmap`(2)/`write`(2) improves binary file transfer performance by about 12% compared to `read`(2)/`write`(2). ASCII transfers and checksum calculations show better performance, too. The daemon will automatically fall back to standard I/O if the `mmap`(2) syscall fails.	
	Argument	*Boolean*
	Default Value	`yes`
`use-sendfile`	On systems supporting `sendfile`(2), the daemon may use that syscall for binary file transfers. Preliminary tests indicated that `sendfile`(2) improves performance by about 18% compared to `read`(2)/`write`(2), and by about 5% compared to `mmap`(2)/`write`(2). The daemon will automatically fall back to memory mapped or standard I/O if the `sendfile`(2) syscall fails.	
	Argument	*Boolean*
	Default Value	`yes`

4.1.1 Access Control Lists

Various configuration directives may depend on ACLs. ACL syntax is

`acl`*ACLName*`= { ... }`

To be more precisely, the above doesn't specify a complete ACL, but adds a ACL rule to *ACLName*. As such, an `acl` declaration may be used multiple times, and the ACL rule will just be added to the end of the current rule list. Likewise, ACL rules are evaluated sequentially, in the order of definition.

Inside the curly brackets, recognized matching criteria are:

- `src` = [`not`] *CIDR*
  (matches source address of client)
- `dst` = [`not`] *CIDR*
  (matches local destination address)
- `authenticated` = [`not`] (`yes` | `no` | `real` | `anon`)
  (matches if the user has authenticated as a `real` or `anon`ymous user; `yes` matches both)
- `protected` = *Boolean*
  (matches according to the TLS protection status)
- `time` = [`not`] *TimeSpecName*
  Matches depending on current time.
  `timespec` objects may be used for time based profile assignments. Both `cron` and `Taylor-UUCP` syntax are supported, see you local `crontab(5)` and/or UUCP man pages for details. Syntax:
  `timespec` =*timespec_name* `{` `"`*entry*`"` [...] `}`
  Example:

```
# Working hours are from Mo-Fr from 9 to 16:59, and
# on Saturdays from 9 to 12:59:
timespec = workinghours {
    "* 9-16 * * 1-5"   # or: "* 9-16 * * Mon-Fri"
    "* 9-12 * * 6"     # or: "* 9-12 * * Sat"
}

timespec = sunday { "* * * * 0" }

timespec = example {
    Wk2305-0855,Sa,Su2305-1655
    Wk0905-2255,Su1705-2255
    Any
}
```

- `user` = [`not`] [`regex`] [`caseless`] *User*
  (matches current user name verbatim or as POSIX regular expression)
- `arg` = [`not`] [`regex`] [`caseless`] *Arg*
  (matches command argument verbatim or as POSIX regular expression)
- `path` = [`not`] [`regex`] [`caseless`] *Path*
  (matches path verbatim or as POSIX regular expression)
- `host` = [`not`] [`regex`] [`caseless`] *Host*
  (matches virtual host name verbatim or as POSIX regular expression)

For `src` and `dst` multiple definitions may be given within the same rule.

Example:

```
acl rfc1918 = {
    src = 127.0.0.1
    src = 10.0.0.0/8
    src = 172.16.0.0/12
    src = 192.168.0.0/16
}

acl ipv6_any = {
    src = ::0
}

acl notsunday = {
    time = workinghours
}

acl test001 = {
    arg regex = ^.cshrc$
    authenticated = real
}

acl test002 = {
    user = root
    authenticated = real
}
```

These are predefined:

```
acl = secure { protected = yes }
acl = any { }
acl = connect { }
acl = real { authenticated = real }
acl = anon { authenticated = anon }
acl = login { authenticated = yes }
```

4.2 ACL-based Configuration

The following table summarizes configuration options with

Variable [`acl` [`not`] *AclName*] = *Argument*

syntax. Example:

```
access acl not someacl = permit
access acl otheracl = permit
access = deny
```

Variable	**Description**	
`access`	Grants initial connection setup based on ACLs.	
	Type of Argument	*Boolean*
	Default Value	`permit`
`address-mismatch`	Permit or deny address mismatches between data and control channel, only necessary for server-to-server transfers.	
	Type of Argument	*Boolean*
	Default Value	`deny`
`ascii-size-limit`	Sets an upper file size limit for size calculations in ASCII transfer mode.	
	Type of Argument	*Number*
	Default Value	*unset*

Variable	Description	
`authentication-failures max`	Sets an upper limit for authentication failures. Stop verifiying authentication after limit is exceeded, just reject.	
	Type of Argument	*Number*
	Default Value	`5`
`authentication-failures bye`	Terminate connection after the specified number of authentication failures.	
	Type of Argument	*Number*
	Default Value	`10`
	Example:	
	`authentication-failures bye = 5`	
`auto-conversion checksum`	Allow or deny on-the-fly calculation of checksum (*.md5, *.crc) files.	
	Type of Argument	*Boolean*
	Default Value	`deny`
`auto-conversion ( gzip | deflate)`	Allow or deny on-the-fly compression to gzip (deflate) format by appending `.gz` to the filename.	
	Type of Argument	*Boolean*
	Default Value	`deny`
	Example:	
	`acl may-compress = { path = regex "\.(txt|doc)$" }` `auto-conversion gzip acl may-compress = permit`	
`banner`	Specifies a file to be displayed before the initial greeting message. Magic cookie substitution applies.	
	Type of Argument	*Path*
	Default Value	*unset*
`banner-action`	Terminates the session after displaying a banner.	
	Argument	`logout`
	Default Value	*unset*
`binary-only`	Rejects non-binary file transfers. Will also be evaluated for `SIZE` calculations in ASCII mode.	
	Type of Argument	*Boolean*
	Default Value	`deny`
	Example:	
	`acl binary = { path = regex "\.(gif|jpg|mp3)$" }` `binary-only acl binary = permit`	
`check-uid`	If enabled, only files belonging to the actual user are accessible.	
	Type of Argument	*Boolean*
	Default Value	`no`
`check-gid`	If enabled, only files belonging to the actual user's group are accessible.	
	Type of Argument	*Boolean*
	Default Value	`no`
`check-perm`	If enabled, only publicly accessible files are permitted.	
	Type of Argument	*Boolean*
	Default Value	`no`
`chmod-mask ( file | directory )`	Bits set in *mask* can not be removed using the `SITE UMASK` or `SITE CHMOD` commands.	
	Type of Argument	*Octal*
	Default Value	*unset*
	Example:	
	`chmod-mask file = 0600`	

Variable	Description	
`deflate-level ( min|max| default )`	These parameters set and/or limit the deflate compression level for both `transmission-mode = z` and `auto-conversion gzip`. Valid levels are from `0` to `9`.	
	Type of Argument	*Number*
	Default Value	*unset*
	Example:	
	`deflate-level default = 7`	
`dotfiles`	Permit or deny access to files starting with a dot.	
	Type of Argument	*Boolean*
	Default Value	`deny`
`fake-group`	Sets the group name to display in directory listings if resolving the GID is not possible or deactivated with the *resolve-ids* clause.	
	Type of Argument	*String*
	Default Value	`ftp`
`fake-owner`	Sets the user name to display in directory listings if resolving the UID is not possible or deactivated with the *resolve-ids* clause.	
	Type of Argument	*String*
	Default Value	`ftp`
`goodbye`	Specifies the absolute path to some file to be displayed at logout time. Magic cookie substitution applies.	
	Type of Argument	*Path*
	Default Value	*unset*
`greeting`	Specifies the initial greeting message in 220 response. Magic cookie substitution applies.	
	Type of Argument	*String*
	Default Value	`"Welcome, pilgrim."`
	Example:	
	`greeting = "%L FTP server (Version %V)"`	
`hostname`	Sets the the virtual hostname for the current session.	
	Type of Argument	*String*
	Default Value	`"misconfigured.host"`
`ident`	If enabled, **ftpd** will attempt to query the remote RFC1413 daemon (if any) for the remote user name, which is informal only and may be used in banners using the `%u` modifier. The ident query is performed asynchronously and doesn't defer the login process.	
	Type of Argument	*Boolean*
	Default Value	`no`
`maintainer`	Sets the site maintainers email address.	
	Type of Argument	*String*
	Default Value	*unset*
`log`	Enables logging for the specified *LogType*s (`command`, `transfer`, `event`, `ident`)	
	Type of Argument	*LogType*
	Default Value	*unset*
	Example:	
	`log acl someacl = ident command transfer`	
`passive address`	Specify the IP address used in `PASV` replies. Might be useful for NAT.	
	Type of Argument	*IPAddress*
	Default Value	*unset*
`passive port ( min|max )`	Specify the port range for `PASV` replies.	
	Type of Argument	*Number*

Variable	Description	
	Default Value	*unset*
`readme`	Specifies the file to be displayed upon entering a directory. That file needs to be world-readable, or it may or may not be displayed. If *File*contains '%s', the daemon will substitute that character sequence with and '-' plus the current language abbreviation, e.g. '-en' or '-de'. If that fails, '%s' will be substituted with an empty string. More than one occurence of '%s' in *file* will most likely result in a segmentation fault. Magic cookie substitution applies.	
	Type of Argument	*File*
	Default Value	*unset*
`readme-once`	Display the `readme` file only once.	
	Type of Argument	*Boolean*
	Default Value	*unset*
`readme-notify`	Notify that the `readme` file exists, but don't display it.	
	Type of Argument	*Boolean*
	Default Value	*unset*
`resolve-ids`	If set to `deny` hides real file ownerships.	
	Type of Argument	*Boolean*
	Default Value	`deny`
`shape-bandwidth`	Establish a session-based upper limit for outgoing bandwidth. The argument is the absolute bandwidth available for the session.	
	Type of Argument	*Number*
	Default Value	*unset*
`symlinks`	Specify which symbolic links to trust. This option is quite critical for system security and defaults to `none`. Recognized keywords: • `all` - accept all symbolic links • `none` - ignore all symbolic links • `root` - accept symbolic links owned by root • `same` - accept symbolic links owned by owner of target • `real` - accept symbolic links for non-anonymous users	
	Type of Argument	*SymlinkType*
	Default Value	*unset*
	Example:	
	`symlinks = root same real`	
`accept timeout`	Sets the timeout for establishing incoming data connections.	
	Type of Argument	*Seconds*
	Default Value	`30`
`connect timeout`	Sets the timeout for establishing outgoing data connections.	
	Type of Argument	*Seconds*
	Default Value	`30`
`idle timeout ( default | min | max )`	This option sets the default, minimum and maximum session timeouts, the latter two for `SITE IDLE`.	
	Type of Argument	*Seconds*
	Default Value	`600`
`transmission-mode z`	Enables/disables the `Z` transmission mode. When enabled, *deflate* data transfer compression may be used. This option is only available if the software was compiled with zlib support.	
	Type of Argument	*Boolean*
	Default Value	`deny`
`umask`	Specifies the default umask. Both MAVIS derived umasks and umasks set with the `SITE UMASK` command have higher priority. Defaults to 022	

Variable	Description	
	Type of Argument	*Octal*
	Default Value	`022`
`welcome`	Specifies a file to be displayed just after login. Magic cookie substitution applies.	
	Type of Argument	*Path*
	Default Value	*unset*
`welcome-action`	Terminates the session after displaying the welcome message.	
	Argument	`logout`
	Default Value	*unset*

FTP commands may depend on ACLs, too. Syntax for that is:

`command =` [`site`] *Command* { (`acl` [`not`] *ACLName* = [`log`] (`permit` | `deny`))* }

Example:

```
command = site chmod { acl connect = log permit }
command = pass { acl not real = log permit }
```

4.3 Path-rewriting using PCRE

If compiled with PCRE (Perl Compatible Regular Expressions) support,

`rewrite` *perl-regex replacement [flags]*

may be used to implement Perl-like file path rewriting rules. Valid flags are `L` (last), `N` (next) and `R` (reject). `$n` (or `${n}` for `n` > 9) in *replacement* will be substituted by the corresponding match in *perl-regex*. This option is available only if PCRE support is compiled in. Example:

```
rewrite ^/ftp/mirror-(.*)$    /ftp/mirror/$1
rewrite ^/tmp/test/(..)$      /tmp/test
rewrite ^/tmp/test/../.*$     $0                      L
rewrite ^/tmp/test/(..)(.*)   /tmp/gaga/${1}/${1}$2   L
rewrite ^/tmp/test123         $0                      R
```

4.4 TLS support

If compiled with TLS support, various TLS related parameters may be specified. Most of the options should obvious enough:

- `tls certfile` = *CertFile*
- `tls keyfile` = *KeyFile*
- `tls passphrase` = *PassPhrase*
- `tls auth` = *Boolean*
- `tls required` = *Boolean*
- `tls cafile` = *CAFile*
- `tls capath` = *CAPath*
- `tls depth` = *Depth*
- `tls ciphers` = *Ciphers*
- `tls old-draft` = *Boolean*

The `auth` keyword enables client certificate based authentication. This requires some further configuration within the `auth` MAVIS module. Certificate based authentication will require at least OpenSSL version 0.9.7.

If `old-draft` is specified, the daemon responds with a `234` instead of a `334` message after successfully negotiating TLS. This enables use of clients conforming to older versions of `draft-murray-auth-ftp-ssl`. It is recommended not to use that option, but to fix the client.

`keyfile` may be omitted, it defaults to *CertFile*.

All this is unset by default.

4.5 MAVIS Configuration

Directives to configure the MAVIS backends are:

- `mavis module =`*module* `{ ... }`
  Load MAVIS module *module*. See the MAVIS documentation for confi guration guidance.
- `mavis path =` *path*
  Add *path* to the search-path for MAVIS modules.

5 Wildcard patterns

Limited file name globbing for the `LIST` and `NLST` commands is implemented for files in the current working directory.

Recognized glob patterns are:

- `*` matches any string, including the empty string
- `?` matches any single character
- `[...]` matches exactly one single character between the brackets. If the first character inside the brackets is a `!`, the expression matches the complement. If it is a `]` it matches the literal `]`. Two characters separated by − denote a range.

For the `CWD` command only, a tilde (`~`) character at the beginning of the argument expands to the users home directory.

6 Magic cookie substitution

The magic cookies used are partially compatible to those utilized by **wu-ftpd**. Text and files specified using the configuration directives **banner**, **goodbye**, **greeting**, **readme** and **welcome** are subject to cookie substitution.

Available conversions are:

- `%A` - number of transfers
- `%B` - build time
- `%C` - current working directory as displayed to user
- `%D` - time for last transfer
- `%E` - maintainer
- `%F` - number of files transfered
- `%H` - virtual host if set, local hostname else
- `%I` - identity - user name for real users, email or empty else

- `%L` - local hostname
- `%P` - email for anonymous users, empty string else
- `%R` - remote host name, `[%r]` if unavailable
- `%T` - local time
- `%U` - user name
- `%V` - version number
- `%a` - total number of bytes transfered
- `%b` - bytes transferred during last transfer
- `%c` - command or file name
- `%d` - direction of transfer (**I**n, **O**ut, **i**n failed, **o**ut failed, **X**: aborted)
- `%e` - event (`login`, `logout` or `reject`)
- `%f` - number of bytes for file transfers
- `%i` - unique session id
- `%l` - local ip address
- `%m` - transfer mode (**a**scii or **b**inary)
- `%r` - remote ip address
- `%s` - file size of last transferred file
- `%t` - type of user (**r**eal, **a**nonymous or **u**nknown)
- `%u` - user name from RFC1413 lookup
- `%%` - literal percent sign

7 Sample configuration

This is from the `ftpd/sample` directory:

```
#!../obj.darwin-9.6.0-i386/ftpd
id = spawnd {
  listen = { port = 2121 }
  spawn = {
    instances min = 1
  }
  background = no
}

id = ftpd {
  debug = NET CMD
  mavis path = ../../mavis/obj.darwin-9.6.0-i386

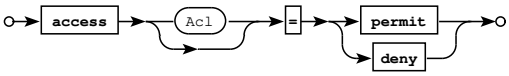
        mavis module = anonftp {
                userid = 100
                groupid = mail
                home = /
                root = /tmp/
                upload = /tmp/incoming/
        }
```

```
  symlinks = all
  check-uid = no
  check-gid = no
  check-perm = no
}
```

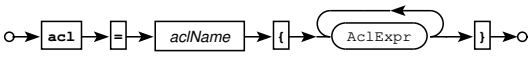
8 Railroad Diagrams



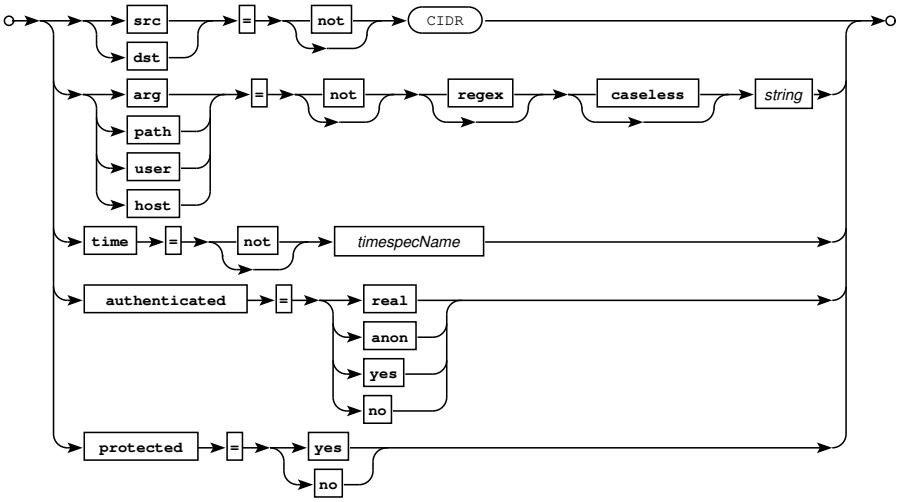
Railroad diagram: AcceptExpr



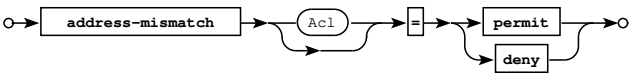
Railroad diagram: AccessExpr



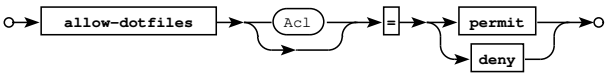
Railroad diagram: AclDecl



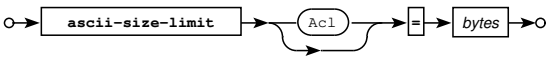
Railroad diagram: AclExpr



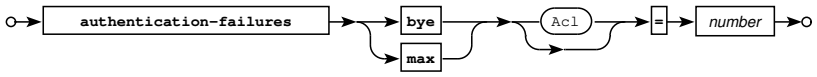
Railroad diagram: AddressMismatchExpr



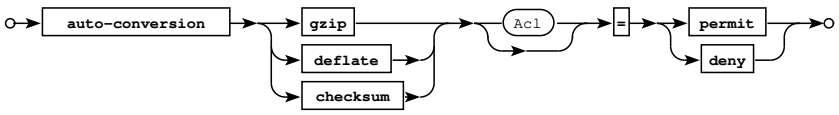
Railroad diagram: AllowDotfilesExpr



Railroad diagram: AsciiSizeExpr



Railroad diagram: AuthFailExpr



Railroad diagram: AutoConvExpr



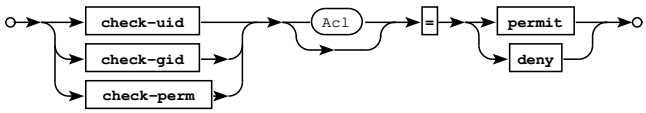
Railroad diagram: BannerActionExpr



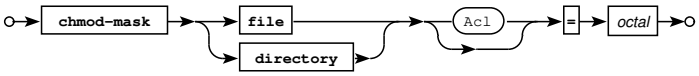
Railroad diagram: BannerExpr



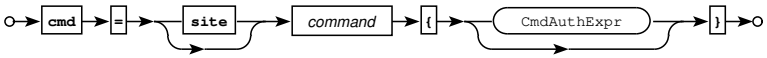
Railroad diagram: BinaryOnlyExpr



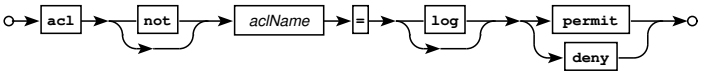
Railroad diagram: CheckExpr



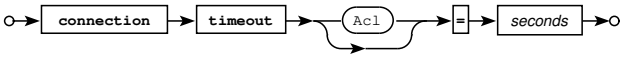
Railroad diagram: ChmodMaskExpr



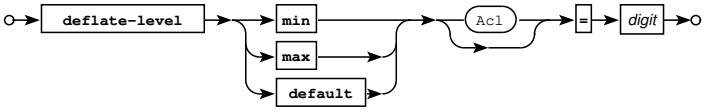
Railroad diagram: CmdAuth



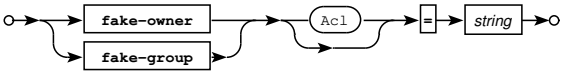
Railroad diagram: CmdAuthExpr



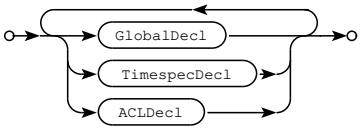
Railroad diagram: ConnectExpr



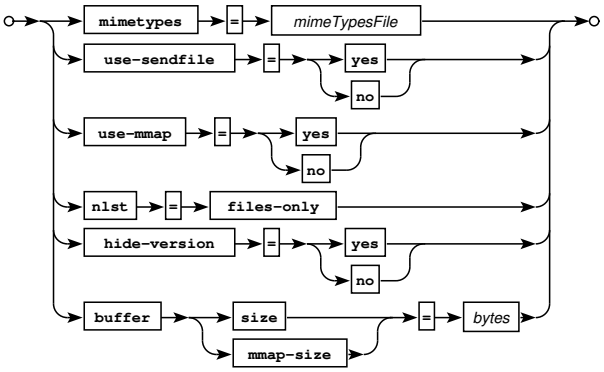
Railroad diagram: DeflateLevelExpr



Railroad diagram: FakeIdExpr



Railroad diagram: FtpdConfig



Railroad diagram: GlobalDecl



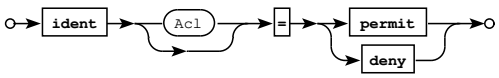
Railroad diagram: GoodbyeExpr



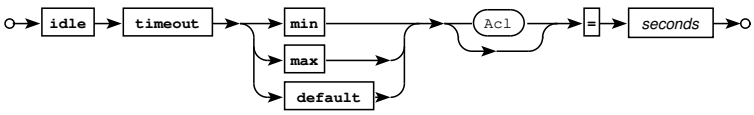
Railroad diagram: GreetingExpr



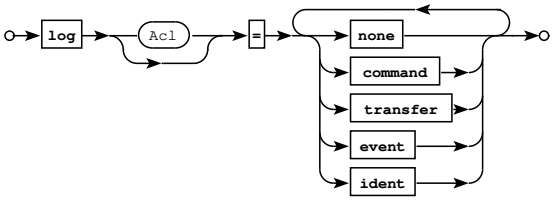
Railroad diagram: HostnameExpr



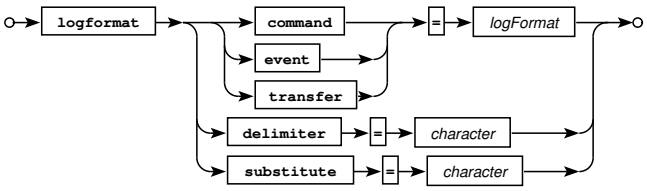
Railroad diagram: IdentExpr



Railroad diagram: IdleExpr



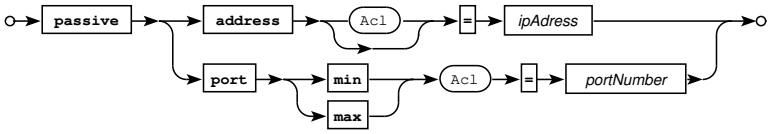
Railroad diagram: LogExpr



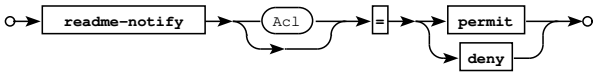
Railroad diagram: LogFormatExpr



Railroad diagram: MaintainerExpr



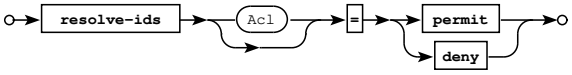
Railroad diagram: PassiveExpr



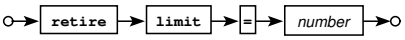
Railroad diagram: ReadmeNotifyExpr



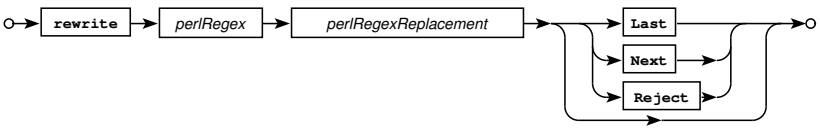
Railroad diagram: ReadmeOnceExpr



Railroad diagram: ResolveIDsExpr



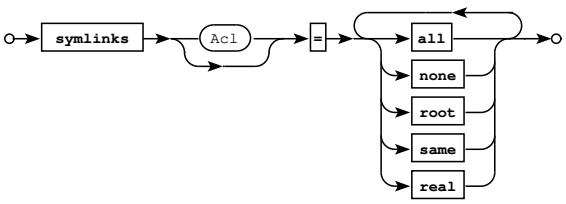
Railroad diagram: RetireExpr



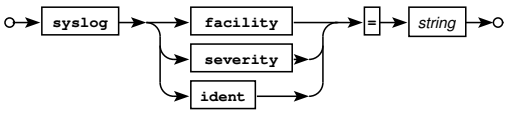
Railroad diagram: RewriteExpr



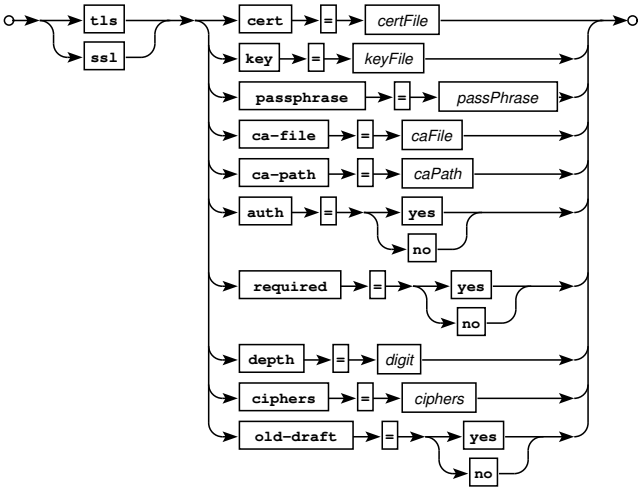
Railroad diagram: ShapeBwExpr



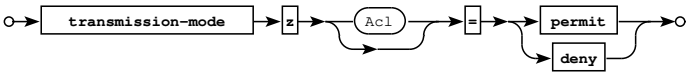
Railroad diagram: SymlinksExpr



Railroad diagram: SyslogExpr



Railroad diagram: TLSExpr



Railroad diagram: TransModeExpr



Railroad diagram: UmaskExpr



Railroad diagram: WelcomeActionExpr

9 Bugs

- The server doesn't perform a `chroot`(2).
- Ftpd has to be started by the super-user unless a non-privileged (and such non-standard) port is used.
- The `LIST` algorithm doesn't permit recursive directory listings, and output differs from POSIX (no `total` line at start of directory listing). However, I don't consider this a serious deficiency, as `LIST` output isn't standardized anyway.
- TLS re-negotiation is currently untested and may or may not work.
- UTF-8 support is likely to be incomplete or plain broken.

10 References

The FTP Daemon hopefully conforms to the following standards and drafts:

- RFC959 - File Transfer Protocol

- RFC1123 - Requirements for Internet hosts - application and support
- RFC1321 - The MD5 Message-Digest Algorithm
- RFC1413 - Identification Protocol
- RFC1639 - FTP Operation Over Big Address Records (FOOBAR)
- RFC2044 - UTF-8, a transformation format of Unicode and ISO 10646
- RFC2228 - FTP Security Extensions
- RFC2389 - Feature negotiation mechanism for the File Transfer Protocol
- RFC2428 - FTP Extensions for IPv6 and NATs
- RFC2577 - FTP Security Considerations
- RFC2640 - Internationalization of the File Transfer Protocol
- RFC4217 - Securing FTP with TLS
- draft-ietf-ftpext-mlst-15.txt - Extensions to FTP
- draft-ftpext-data-connection-assurance-00.txt - FTP Data Connection Assurance
- draft-somers-ftp-mfxx-03.txt - The "MFMT", "MFCT", and "MFF" Command Extensions for FTP
- draft-preston-ftpext-deflate-03.txt - Deflate transmission mode for FTP
- draft-hethmon-mcmurray-ftp-hosts-02.txt - File Transfer Protocol HOST Command
- draft-ietf-ftpext2-hash-01 - File Transfer Protocol HASH Command for Cryptographic Hashes
- draft-bryan-ftp-range-01 - File Transfer Protocol RANG Command for Byte Ranges

11 Copyrights and Acknowledgements

Please see the source for copyright and licensing information of individual files.

- **The following applies if the software was compiled with TLS support:**

  This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/).

  This product includes cryptographic software written by Eric Young (eay@cryptsoft.com").

- **If the software was compiled with PCRE (Perl Compatible Regular Expressions) support, the following applies:**

  Regular expression support is provided by the PCRE library package, which is open source software, written by Philip Hazel, and copyright by the University of Cambridge, England.

  (ftp://ftp.csx.cam.ac.uk/pub/software/programming/pcre).

- **MD5 algorithm:**

  The software uses the RSA Data Security, Inc. MD5 Message-Digest Algorithm.

- **Deflate (gzip) compression support** is implemented using the `zlib` library written by Jean-loup Gailly (jloup@gzip.org) and Mark Adler (madler@alumni.caltech.edu).

- **The original tac_plus code (which this software and considerable parts of the documentation are based on) is distributed under the following license:**

  Copyright (c) 1995-1998 by Cisco systems, Inc.

  Permission to use, copy, modify, and distribute this software for any purpose and without fee is hereby granted, provided that this copyright and permission notice appear on all copies of the software and supporting documentation, the name of Cisco Systems, Inc. not be used in advertising or publicity pertaining to distribution of the program without specific prior permission, and notice be given in supporting documentation that modification, copying and distribution is by permission of Cisco Systems, Inc.

  Cisco Systems, Inc. makes no representations about the suitability of this software for any purpose. THIS SOFTWARE IS PROVIDED ``AS IS'' AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

- **The code written by Marc Huber is distributed under the following license:**

  Copyright (C) 1999-2022 Marc Huber (Marc.Huber@web.de). All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

     This product includes software developed by Marc Huber (Marc.Huber@web.de).

     Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

  THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ITS AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.