

Cybersecurity Project

F. Barbarulo, G. B. Rolandi, Bruk T. Gurmesa

6 aprile 2019

Indice

1	Protocol	2
1.1	Request commands	2
1.2	Reply status	2
1.3	Request commands list	3
1.3.1	ALLO	3
1.3.2	DELE	3
1.3.3	LIST	3
1.3.4	QUIT	3
1.3.5	RETR	4
1.3.6	STOR	4
1.4	Generic return values	4

1 Protocol

Client must create a secure communication channel to connect to the server.

A parola is defined as a pure ASCII sequence which can match the following regular expression `[A-Za-z0-9\._]*`. Different parolas can be separated by a whitespace.

1.1 Request commands

A request command is sent by the client to the server.

Generic structure of a request command is:

```
COMM filename\n
tag: value\n
\n
body
```

Each line is terminated by a LF char, and each command is terminated by a LF char. It is important to send the line feed char because server will not parse the command until two trailing LFs are received.

Note: a LF char is represented by a byte whose value is 0x0A.

A command is composed by the following parts:

- COMM is a 4 char parola and represents the command to issue;
- filename is the name of a file on which issue the command. Filenames are composed by one parola. Depending on the specific command, a filename is mandatory or forbidden;
- tag: value is a pair of two parolas, composed by a parola immediately followed by a colon, a whitespace and another parola. First parola represents the tag for an additional parameter of current command, while second one represents the actual value for the parameter; depending on the specific command, one or more parameters may be mandatory or forbidden;
- body is a stream of bytes whose length must be specified in a command parameter; depending on the specific command, a body is mandatory or forbidden;

1.2 Reply status

A reply status is sent by the server to the client.

Generic structure of a reply status is:

```
DDD\n
tag: value\n
body
```

A reply status is composed by the following parts:

- DDD a 3 ASCII digits number representing the state of the previous request command;

- tag: value is a pair to represent a parameter, with the same format of the request command parameter; depending on the specific command, one or more parameters may be mandatory or forbidden;
- body is a stream of bytes whose length must be specified in a reply status parameter; depending on the specific command the reply refers to, a body is mandatory or forbidden;

1.3 Request commands list

1.3.1 ALLO

ALLO filename

Request server to allocate a file called *filename*. This command is useful to atomically:

- test if a file exists on the remote server
- create that file

and must be used before issuing a STOR command.

Reply status:

- 200: ok
- 201: ok, file already exists
- 452: bad file

1.3.2 DELE

DELE filename

Reply status:

- 200: ok
- 452: bad file

1.3.3 LIST

LIST

Request the server to send a list of available files

Reply status:

200\n

Size: listsize\n

\n

body

1.3.4 QUIT

QUIT

Close connection with the server.

Reply status:

- 200: ok

1.3.5 RETR

RETR filename

Retrieve a file from the server.

Reply status:

- success:

```
200\n
Size: filesize\n
\n
body
```

- failure:
452: bad file

1.3.6 STOR

STOR filename

Size: filesize

body

Store a file on the remote server. This command can be issued only after an ALLO command.

- Size: filesize is a parameter pair. filesize is the decimal ASCII representation of the file size in bytes;
- body is the actual content of the file;

Reply status:

- 200: ok

1.4 Generic return values

- 500: syntax error