FTP Security Extensions

RFC- 2228

The Project Presentation

Written by:

USAMA YASSA

March 14,2001

OBJECTIVE

The standard track document, RFC-2228, defines FTP Security Extensions to the File Transfer Protocol (FTP). In the current FTP it is necessary to log into the remote host; the user must have a user name and password to access files and directories. This is primary method of handling the security. But there is no encryption or verification available to authenticate Username, Passwords, Commands, Replies or Transferred Data. Usernames and passwords passed in clear text to authenticate clients to servers

For services such as "anonymous" FTP archives, this represents a security risk whereby passwords can be stolen. A need exists to securely transfer files.

The FTP Extensions offer several new commands, but all are optional. None of the new commands has to be implemented, although there are some dependencies between them.

The FTP extensions are fully compatible with the previous FTP standard STD 9.

FTP Security Overview

In the context of FTP security, authentication is the establishment of a client's identity and/or a server's identity in a secure way, usually using cryptographic techniques.

Authorization is the process of validating a user for login. The basic authorization process involves the USER, PASS, and ACCT commands. With the FTP security extensions, authentication established using a security mechanism may also be used to make the authorization decision.

FTP Security Scenario

1) AUTH command

An FTP security interaction begins with a client telling the server what security mechanism it wants to use with the AUTH command.

Reject this mechanism, In the case of a Server which does not implement
The security Extensions, reject the command completely.
If none is needed, this will usually mean that the mechanism is one where
The password is to be interpreted differently, such as with token or one -time
Password system.
If the server accepts this mechanism, the server's reply will
Indicate if the client must respond with additional data for the security
Mechanism to Interpret If the server requires additional security
Information, then the client and server will enter into a security data
Exchange

2) ADAT command

The client will send an ADAT command containing the first block of security data. The server's reply will indicate if the data exchange is complete. If more data is needed, the client will send another ADAT command containing the next block of data, and await the server's reply. This exchange can continue as many times as necessary. Once this exchange completes, the client and server have established a security association.
The ADAT command must be preceded by a successful AUTH command

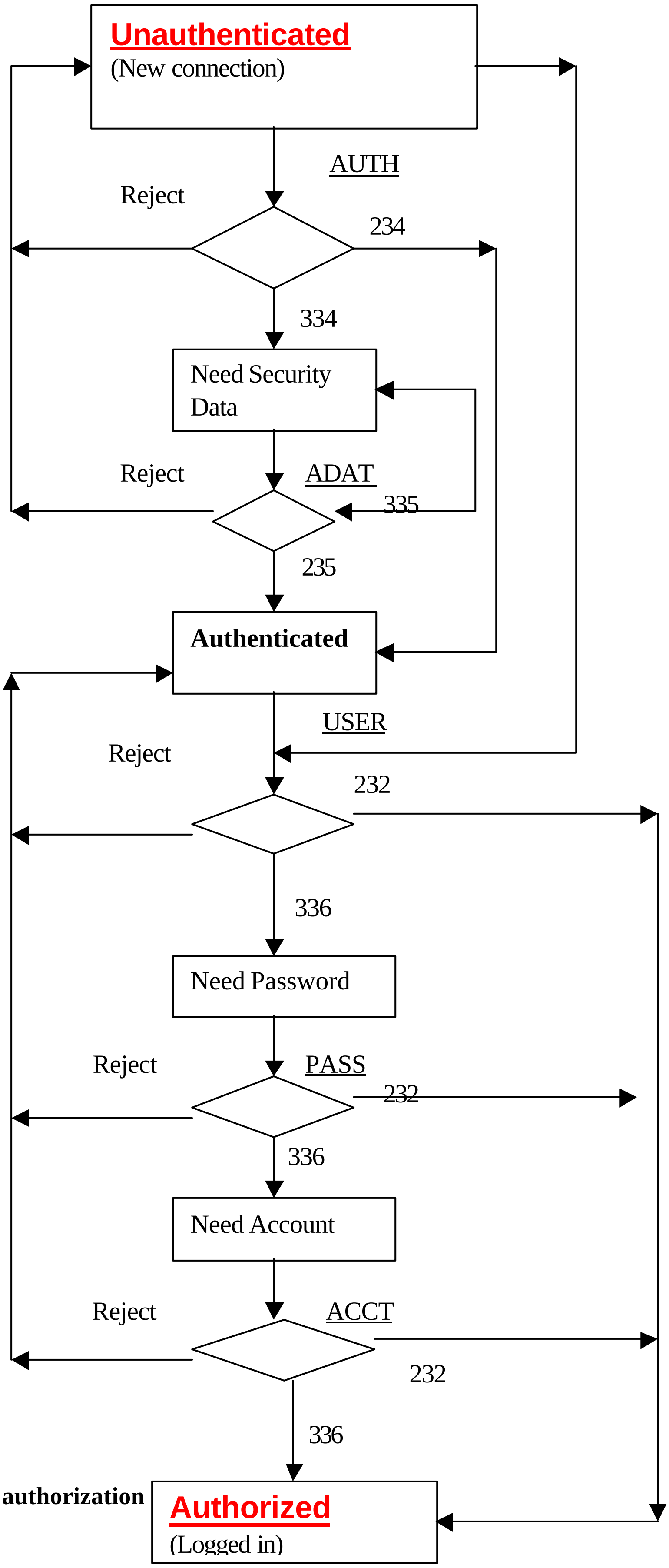
> The security data exchange may, among other things, establish the identity of
> The client in a secure way to the server, This identity may be used as one
> Input to the login authorization process. In response to the FTP login
> Commands (**AUTH, PASS, ACCT**),

3) USER command

A username specified by the USER command is always required to specify the
Identity to be used on the server
If the server is willing to allow the user named by the USER command to log in, it should respond with reply code 232.
If the security mechanism requires a password, it should respond to the USER command with reply code 336.

Individual reply codes

234 Security data exchange complete.
334 This reply indicates that the requested security
Mechanism, is ok,
335
This reply indicates that the security data is; acceptable, and more
Is required to complete the; security data exchange
235 This reply indicates that the security data exchange; completed
Successfully.
232 User logged in, authorized by security data exchange.
336 Username okay, need password. Challenge is "...."



Authentication and authorization Flow Chart