FTP – RFC 959

Einleitung:

In diesem Dokument wird Bezug auf den RFC 959 Standard genommen. Es soll einen Überblick über die notwendigen FTP Befehle geben, welche zur erfolgreichen Datenübertragen erforderlich sind.

Das Protokoll:

Durch das ICP (Initial Connection Protocol) wird ein Kommunikationskanal zwischen dem Client (Protocol Interpreter) und dem Server über einen Standard Socket hergestellt. Der Client ist für das Senden der FTP-Kommandos und das Interpretieren der zurückgeschickten Kommandos verantwortlich. Der Server muss ebenfalls Kommandos interpretieren, Antwort Kommandos senden und seinem DTP (Data Transfer Process) die Anweisung zu einem Datenverbindungsaufbau geben. Die FTP-Kommandos werden als TELNET Strings zwischen User und Server geschickt. Die Kommando Codes sind Buchstaben die, wenn Parameter folgen, durch ein <SP> (Space) getrennt werden. Das Ende eines Kommando Code Strings wird durch ein TELNET-EOL (End of Line) definiert. In der C# Programmierung wird dem Client nach jedem Befehl String ein „\r\n“ mitgeschickt.

FTP-Kommandos:

* **USER:** Dieses Kommando spezifiziert den Benutzernamen. Es ist das erste Kommando das nach Aufbau einer TELNET Verbindung gesendet wird. Die User Authentifizierung wird benötigt um Zugriff auf das Server File System zu bekommen.
* **PASS:** Dieses Kommando repräsentiert das Passwort des Users. Es muss nach dem USER Kommando gesendet werden.
* **CWD:** CWD (Change Working Directory) wird dazu verwendet um den Pfad für den Empfang oder Senden einer Datei zu ändern. Nach diesem Kommando wird der neue Pfadname angegeben.
* **QUIT:** Dieses Kommando beendet das USER Kommando und falls keine Datenverbindung läuft, wird die Verbindung zum Server geschlossen. Werden noch Daten übertragen, wartet der Server noch auf eine Bestätigung, dass die Daten übertragen wurden und erst dann wird die Verbindung geschlossen.
* **PORT:** Mit dem PORT Kommando kann ein Port für die Datenverbindung definiert werden. Im Port Kommando wird die IP- Adresse inklusive gewünschten Port angegeben, worüber die Daten gesendet werden. Dabei werden die IP-Adresse und Port in 8 Bit großen Feldern unterteilt, als Dezimalzahlen dargestellt und mit Beistrichen getrennt. Es wird nun mit dem PORT Befehl als Präfix an den Server als String gesendet. Bsp.: 192.0.0.1:4422 = PORT 19,20,01,44,22. Dieser Befehl wird in der Implementierung nicht verwendet, da die Datenverbindung über den PASV Befehl geregelt wird.
* **PASV:** Mit diesem Befehl wird der Server in einen Passiv Modus (Listen) gesetzt. Es werden als Parameter die IP-Adresse und Port mitgeschickt. Auf diesem Port wird nun auf eine Antwort des Clients gewartet, um dann eine Datenverbindung aufzubauen.
* **TYPE:** Der Type Befehl gibt das Repräsentationsformat für den Datentransfer vor.
* **PWD:** PWD (Print Working Directory) gibt den momentanen Pfad am Server zurück.
* **RETR:** Das Retrieve Kommando fordert den Server auf eine Kopie eines Files auf den Client über die Datenverbindung zu schicken.
* **STOR:** Über diesen Befehl können Files auf den Server abgelegt werden. Der Server nimmt die Daten entgegen und speichert das File. Ist ein File mit gleichem Pfadnamen vorhanden wird eine Rückmeldung ausgegeben, andernfalls wird es neu angelegt. Bei der Implementierung ist aufgefallen, dass die Client Programme validieren ob das File bereits vorhanden ist oder nicht.
* **MKD**: Mit dem MKD Befehl kann ein Ordner auf dem Server erstellt werden.
* **LIST:** Dieser Befehl sendet eine Liste von Files in einem Ordner bzw. Information bezüglich einem File an den Client. Wird ein Pfadname angegeben, wird eine Liste von Files in diesem Pfad zurückgegeben. Wird ein Pfad eines Files angegeben, werden Informationen über das File zurückgegeben. Wird kein Pfad mit gesendet (Null Argument), werden Informationen über das momentane, bzw. Standard Directory geschickt.
* **FEAT:** Dieser Befehl sendet eine Liste der Befehle die über den RFC 959 Standard hinaus unterstützt werden.
* **DELE:** Dieser Befehl löscht ein File vom Server.
* **RMD:** Mit diesem Befehl wird ein Ordner am Server gelöscht.
* **SYST:** Der System Befehl gibt dem Client die Art des Server Systems zurück.

Der Datentransfer:

Die Kommunikation zwischen Client und Server wird über eine TELNET Verbindung abgehandelt. Diese TELNET Verbindung wird zu Beginn vom PI (Protokoll Interpreter) des Clients initialisiert. Die FTP-Kommandos werden mit Hilfe des Client-PI erstellt und an den Server gesendet. Wenn der Server die Kommandos erhalten hat, wird wieder über die TELNET Verbindung eine Antwort zu den Kommandos zurückgeschickt und der Server wartet auf die Befehle. Bei den meisten FTP Servern ist eine Übertragung der Zugangsdaten des Benutzers Voraussetzung. Deshalb wird dem Server das „username“ Kommando mit dem Benutzernamen geschickt, welcher dann die Eingabe bestätigt (Code 330, wenn korrekt). Als nächstes wird das Passwort („password“ Kommando) geschickt und vom Server bestätigt. (Code 230, wenn korrekt). War die Anmeldung auf den Server erfolgreich, wird nun die Datenübertragung vorbereitet. Je nachdem ob ich ein File vom oder auf den Server übertragen will gibt es verschiedene Vorgehensweisen.

File vom Server:

Der Client muss über das „retrieve“ Kommando den Typ-Encoding, lokalen Pfadnamen und originalen Pfadnamen angeben, um ein File transferieren zu können. Werden die Kommandos bestätigt (Code 250), wird die Datenverbindung zum Transfer der Daten aufgebaut. Wurden alle Daten für den Transfer übertragen, sendet der Server die Response „complete“.

File zum Server:

Bei einer File Übertragung zum Server muss vorher der Typ angegeben werden („type“ Kommando). Falls die Übertragung erfolgreich war, wird als Bestätigung der Code 200 zurückgeschickt. Durch das „store“ Kommando wird dann das File am Filesystem des Servers gespeichert. „store“ braucht dafür einen Typen Parameter, einen lokalen Pfadnamen und einen Pfadname des originalen Files.

Fehlerszenarien:

Es kann passieren, dass der Server einen FTP Befehl nicht implementiert. Ist dies der Fall, wird eine Standardantwort zurückgesendet, die dem Client mitteilt, dass der Befehl nicht interpretiert werden konnte. Es wird als Antwort „502 Command not implemented“ zurückgeschickt.

Wenn der Server einen gültigen Benutzer benötigt (Username, Passwort), wird der Zugang zum Server verweigert.