

Contexto Geral

O servidor mantém uma pasta chamada **arquivos** com os arquivos disponíveis para download. O cliente conecta via TCP e pode executar comandos para listar arquivos, baixar arquivos, obter hash MD5 parcial, retomar downloads e baixar múltiplos arquivos por máscara. A comunicação é textual, com comandos simples e respostas claras. A transferência de arquivos é feita em blocos de até 1024 bytes.

1. Estabelecimento da conexão

- O cliente abre um socket TCP para o **HOST:PORT** (no caso, **localhost:5000**).
 - O servidor aceita conexões e cria uma thread para atender cada cliente simultaneamente.
-

2. Comandos e Respostas

Comando DIR — Listar arquivos do servidor

Cliente envia:

DIR\n

- (5 bytes: 3 caracteres + newline)

Servidor responde com uma string UTF-8 contendo:

nome_arquivo1 - tamanho_bytes\n

nome_arquivo2 - tamanho_bytes\n

...

- O tamanho da resposta é variável, dependendo da quantidade e tamanho dos arquivos.
 - O cliente lê até 4096 bytes da resposta de uma vez.
-

Comando DOW <nome_arquivo> — Download completo de arquivo

Cliente envia:

DOW nome_arquivo\n

- Servidor responde:

Se o arquivo não existe:

ERRO: Arquivo nao encontrado.

Se existe:

OK <tamanho_do_arquivo>

- Exemplo: **OK 12345**
- Cliente lê essa resposta até 1024 bytes.
- Se resposta OK, inicia a transferência dos dados:
 - Servidor envia os dados do arquivo em blocos de até 1024 bytes.
 - Cliente recebe e salva localmente até completar o tamanho informado.

Comando MD5 <nome_arquivo> — Obter hash MD5 parcial

Cliente envia:

MD5 nome_arquivo pos\n

- Onde o **pos** é o número de bytes para cálculo do hash.
- Servidor lê até **pos** bytes do arquivo, calcula o hash MD5 e envia o resultado (32 caracteres hexadecimais).
- Cliente recebe até 1024 bytes da resposta.

Comando DRA <nome_arquivo> <tamanho_local> — Retomar download

Cliente envia:

DRA nome_arquivo tamanho_local\n

- Onde o `tamanho_local` é a quantidade de bytes já baixados localmente.
 - Servidor:
 - Se arquivo existe, posiciona o ponteiro no byte `tamanho_local`.
 - Envia `OK` (2 bytes).
 - Envia os dados restantes do arquivo em blocos de até 1024 bytes.
 - Cliente recebe a confirmação e continua a baixar os dados restantes, acrescentando ao arquivo local.
-

Comando DMA <máscara> — Download múltiplo por máscara

Cliente envia:

DMA *.exe\n

- Servidor procura arquivos que batem com a máscara.
 - Envia o número de arquivos encontrados (exemplo: `3`).
 - Para cada arquivo encontrado, há uma sequência de confirmações e envio:
 1. Cliente envia `OK` (2 bytes) para receber o nome.
 2. Servidor envia o nome do arquivo.
 3. Cliente envia `OK` para receber o tamanho.
 4. Servidor envia o tamanho do arquivo.
 5. Cliente envia `OK` para receber os dados.
 6. Servidor envia o arquivo em blocos de 1024 bytes.
 - Cliente salva os arquivos localmente.
-

3. Formato dos Pacotes

Momento	Enviado por	Conteúdo	Formato / Tamanho
Comando	Cliente	Texto + \n	Variável, termina com 1 byte newline
Resposta DIR	Servidor	Lista de arquivos	Texto UTF-8, tamanho variável
Resposta DOW (OK)	Servidor	OK <tamanho>	Texto, ex: OK 12345
Arquivo (DOW, DRA, DMA)	Servidor	Dados binários	Blocos de até 1024 bytes
Resposta MD5	Servidor	Hash MD5 (32 caracteres)	Texto ASCII, 32 bytes
Resposta erro	Servidor	Mensagem de erro	Texto ASCII
Confirmação OK no DMA	Cliente	OK	2 bytes

4. Fluxo típico de download (DOW)

1. Cliente envia DOW arquivo\n.
 2. Servidor responde OK <tamanho>.
 3. Cliente recebe o arquivo em blocos de 1024 bytes até o total informado.
 4. Cliente salva o arquivo e finaliza o download.
-

5. Considerações Técnicas

- O tamanho máximo de blocos para transferência é 1024 bytes.
- Mensagens de controle são texto ASCII/UTF-8, geralmente terminadas com newline, exceto no DMA, que usa confirmações simples sem newline.
- Cliente confia no tamanho informado para saber quando terminar de receber dados.
- Mensagens de erro começam com ERRO e indicam problemas.