



Agentes de Internet

■ Características

■ Ambiente

- | Internet

■ Sensores e Atuadores

- | Sockets

■ Percepção e Atuação

- | Mensagens de entrada e de saída via sockets

■ Peculiaridade

- sensores e atuadores em um mesmo canal bi-direcional

- necessidade de sincronismo

- | protocolo de comunicação

- Protocolos de Internet

- | HTTP, FTP, NNTP, SMTP, IRC, etc ...



FTP

■ File Transfer Protocol (RFC 959)

- promover o compartilhamento de arquivos (programas ou dados)
- encorajar o uso indireto ou implícito (via programas) de computadores remotos
- dar suporte ao usuário quanto a diferenças entre sistemas de arquivos em diferentes máquinas
- transferir dados de maneira confiável e eficiente

■ Modos de Acesso

- interativo - via terminal
- não interativo - acesso via programas
- Apesar de permitir o acesso interativo via terminal, o FTP foi planejado para ser utilizado por programas



FTP

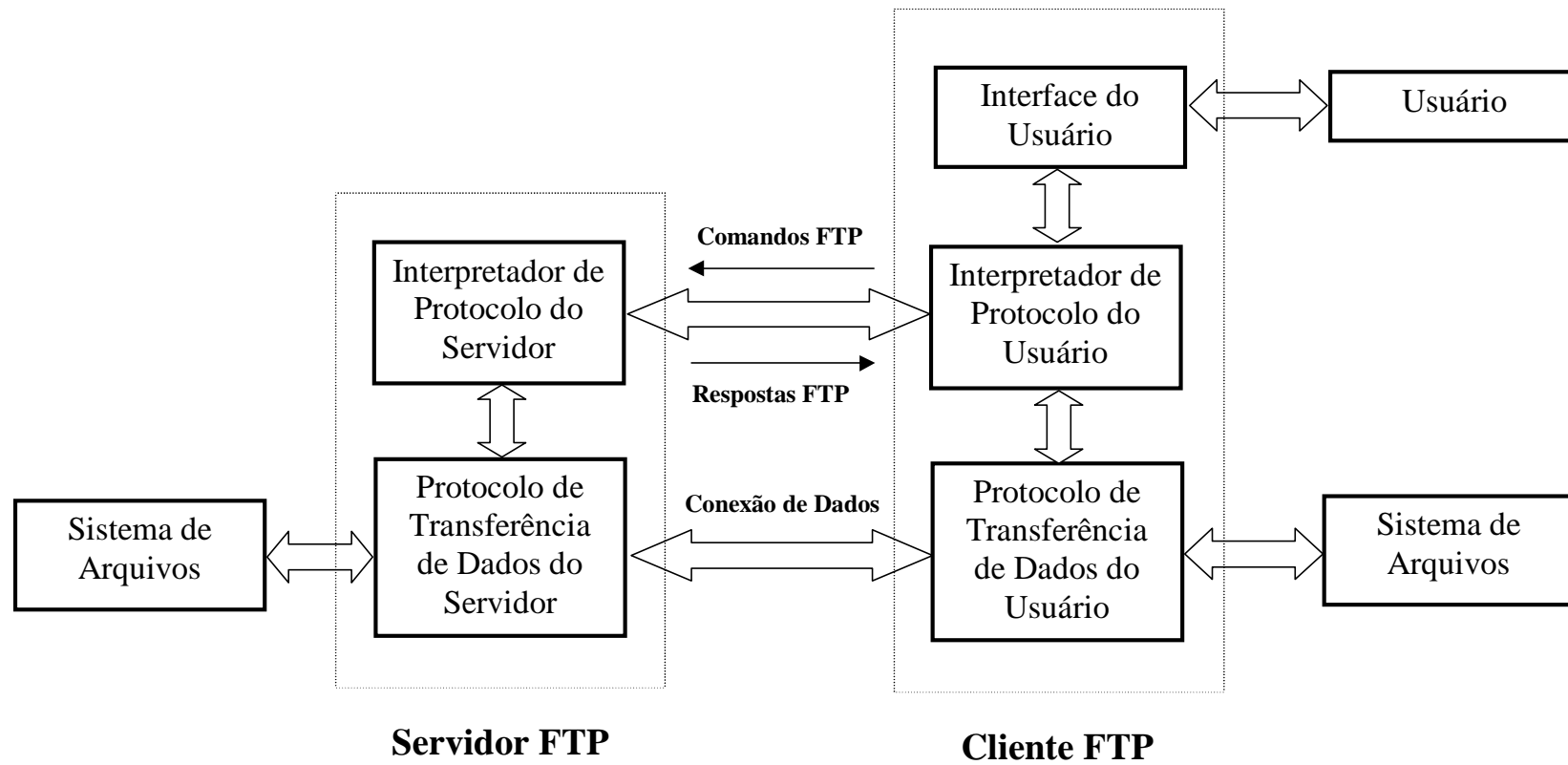
■ Histórico

- Mecanismo para transferência de arquivos foi proposto em 1971 e implementado em hosts do MIT (RFC 114)
- Em seguida, diversos outros RFC's discutiram e aprimoraram o mecanismo (RFCs 141, 172, 264, 265, 281, 294, 354, 385, 414, 430)
- O primeiro documento oficial regulando o FTP foi o RFC 454
- Em 1973, o RFC 542 incorporou uma série de modificações, tornando-se a especificação oficial
- Em seguida vieram as RFCs 607, 614, 624, 686 e 691
- Com a definição do TCP como protocolo de comunicação, veio a RFC 765
- Por fim, em 1985, a RFC 959 veio introduzir uma série de novos comandos



FTP

Modelo FTP





Comandos FTP

■ Norma

- <http://www.w3.org/Protocols/rfc959/>

■ Portas Default: (ftp - 21, ftp-data - 20)

■ Controle de Acesso

- User Name (USER)
- Password (PASS)
- Change Working Directory (CWD)
- Logout (QUIT)

■ Parâmetros de Transferência

- Porta de Dados (PORT)
- Tipo de Representação (TYPE)
 - A - ASCII, E - EBCDIC, I - IMAGE, N - Non-print, T - Telnet, C - Carriage Control



Comandos FTP

■ Comandos de Serviço

- Retrieve (RETR)
- Store (STOR)
- Append (APPE)
- Remove Directory (RMD)
- Make Directory (MKD)
- Print Working Directory (PWD)
- List (LIST)

■ Implementação Mínima

- TYPE - ASCII Non-print, MODE - Stream,
- STRUCTURE - File, Record
- COMMANDS - USER, QUIT, PORT, TYPE, MODE, STRU, for the default values (TYPE - ASCII Non-print, MODE - Stream, STRU - File) RETR, STOR, NOOP.



EC2 - Exercício Computacional 2

- Implementar o agente de mirror onde agora um dos lados do mirror é um diretório local e o outro um servidor ftp remoto
- Lembrar de manter o agente em estado "sleep" um certo tempo para evitar o consumo excessivo de tempo de CPU e o congestionamento da rede
- A implementação deve ser feita em cima de sockets e não em cima de outros objetos mais especializados existentes no Java



O World Wide Web (WWW) e seus recursos

■ Recursos na Web

■ Múltiplos Protocolos

| http, gopher, wais, ftp, mail, news, telnet, local files

■ URI

■ Universal Resource Identifier

■ Sintaxe

| scheme://user:password@host:port/url-path

■ URL

■ Universal Resource Locator

■ Tipo de URI que expressa um endereço mapeado em um algoritmo de acesso via protocolos de rede



HTTP

- Protocolo http (hyper text transfer protocol)
 - Protocolo de transmissão de dados multimídia
 - Por que ? Já não há o ftp ?
 - Eficiência na transmissão
 - Dados transferidos podem ser quaisquer
 - | texto, hiper-texto, imagens, audio, video, etc
- Protocolo simples de 4 fases
 - Conexão
 - Requisição
 - Resposta
 - Desconexão



HTTP

■ Histórico

■ HTTP 0.9 (1990)

- | protocolo simples para transferência de dados não formatados na Internet

■ HTTP 1.0 (RFC 1945 - Maio 1996)

- | introdução de formatação de dados do tipo MIME
- | meta-informação sobre os dados transferidos
- | modificações na semântica de requisições/respostas
- | não levava em consideração o efeito de proxies hierárquicos, cacheamento, a necessidade de conexões persistentes ou hosts virtuais

■ HTTP 1.1 (RFC 2068 - Janeiro 1997/ RFC 2616 - Junho 1999)



HTTP

■ Requisição

- Cliente envia uma requisição ao servidor
- METHOD URI PROTOCOL_VERSION
- Requisição é seguida por uma mensagem do tipo MIME contendo modificadores de requisição
- Mensagem pode ainda ser seguida por um "corpo de conteúdo "

■ Resposta

- Servidor envia uma linha de status, incluindo a versão do protocolo e um código de erro/sucesso
- Em seguida envia uma mensagem do tipo MIME contendo informações do servidor, meta-informações da entidade e possivelmente o corpo de conteúdo da entidade



HTTP

■ Conexão Cliente Servidor

- Direta

- Intermediada

 - | proxy - re-escreve parte da mensagem, quando necessário

 - | gateway - traduz entre protocolos diferentes, quando necessário

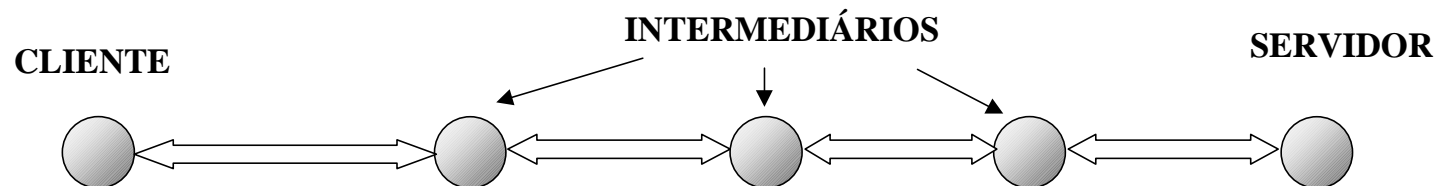
 - | tunnel - não modifica as mensagens

■ Algumas opções de comunicação se aplicam

 - | somente ao vizinho não-túnel mais próximo

 - | somente ao ponto final da cadeia

 - | a todos os pontos da cadeia





HTTP

- Qualquer parte da cadeia que não estiver agindo como túnel
 - pode utilizar algum tipo de cache para gerenciar as requisições
- Efeito do Cache
 - encurtar a cadeia de requisição/resposta
- Nem todas as respostas
 - podem ser cacheadas
- Algumas requisições
 - podem conter modificadores que determinam o comportamento do cache
- Grande variedade de arquiteturas e configurações de cache
 - http 1.1 suporta uma grande gama de configurações possíveis



HTTP

- Usualmente ocorre em conexões TCP/IP
 - Porta 80
- HTTP 1.0
 - utilizava uma conexão por requisição/resposta
- HTTP 1.1
 - permite que mais de um ciclo requisição/resposta ocorra em uma mesma conexão
 - normalmente tem somente um ciclo por conexão
 - conexões persistentes são o default
 - cliente deve assumir que o servidor manterá a conexão



HTTP

■ Methods

- OPTIONS - obtém as opções para a entidade solicitada
- GET - requisita uma determinada entidade
- HEAD - requisita o header de uma determinada entidade
- POST - envia uma entidade local para uma entidade remota
- PUT - envia uma entidade local para um servidor remoto
- DELETE - apaga uma entidade em um servidor remoto
- TRACE - requisita um echo das requisições enviadas
- CONNECT - reservada para uso com proxies que podem dinamicamente se tornar túneis



HTTP

■ Headers

- <field-name>:[<field-value>]

- General Headers

 - | Cache-Control, Connection, Date, Pragma, Trailer, Transfer-Encoding

- Request Headers

 - | Accept, Accept-Charset, Accept-Encoding, Accept-Language, Authorization, Expect, From, Host, If-Match, If-Modified-Since, If-None-Match, If-Range, If-Unmodified-Since, Max-Forwards, Proxy-Authorization, Range, Referer, TE, User-Agent

- Entity Headers

 - | Allow, Content-Encoding, Content-Language, Content-Length, Content-Location, Content-MD5, Content-Range, Content-Type, Expires, Last-Modified



HTTP

■ Headers (cont)

■ Response Header

| Accept-Ranges, Age, ETag, Location, Proxy-Authenticate,
Retry-After, Server, Vary, WWW-Authenticate

■ Sintaxe das Requisições

```
Method SP Request-URI SP HTTP-Version CRLF
*(( general-header | request-header | entity-header ) CRLF)
CRLF
[ message-body ]
```

■ Sintaxe das Respostas

```
HTTP-Version SP Status-Code SP Reason-Phrase CRLF
*(( general-header | response-header | entity-header ) CRLF)
CRLF
[ message-body ]
```



Definição dos Headers

- **Accept**
 - indica os tipos de mídia aceitáveis para a resposta
- **Accept-Charset**
 - indica os conjuntos de caracteres que são aceitáveis para a resposta
- **Accept-Encoding**
 - semelhante a Accept, mas restringindo as codificações permitidas para o conteúdo
- **Accept-Language**
 - semelhante a Accept, mas restringindo as linguagens aceitas para o conteúdo
- **Accept-Ranges**
 - header de resposta, que permite ao servidor indicar os tipos de range aceitos



Definição dos Headers

- **Age**
 - header de resposta que determina o tempo de existência da entidade requisitada
- **Allow**
 - header de resposta que indica os métodos aceitos para o URI requisitado
- **Authorization**
 - apresenta credenciais para informações restritas
- **Cache-Control**
 - utilizado para especificar diretivas que devem ser obedecidas pelos mecanismos de cache na cadeia de requisição/resposta
- **Connection**
 - permite a especificação de opções desejadas para uma conexão em particular



Definição dos Headers

■ Content-Encoding

- indica uma modificação do tipo de mídia, ou seja, codificações adicionais foram aplicadas ao corpo da entidade, demandando mecanismos que a decodifiquem de modo que o tipo indicado em Content-Type seja obtido novamente

■ Content-Language

- indica a linguagem a qual o documento (entidade) se destina

■ Content-Length

- indica o tamanho da entidade

■ Content-Location

- usado para determinar a localização de um recurso quando a mesma não está indicada no URI



Definição dos Headers

- **Content-MD5**
 - chave para ser utilizada em mecanismos de verificação de integridade da entidade transferida
- **Content-Range**
 - header de entidade, indicando onde a parte corrente se localiza dentro da entidade global - utilizado em entidades multi-partes
- **Content-Type**
 - indica o tipo de mídia referente à entidade
- **Date**
 - indica a data e a hora em que a resposta foi originada
- **ETag**
 - associa um tag a uma entidade



Definição dos Headers

- **Expect**
 - usado para indicar um tipo determinado de comportamento esperado do servidor
- **Expires**
 - indica a data e hora a partir do qual uma entidade pode ser considerada desatualizada
- **From**
 - utilizado para indicar o autor da requisição na forma de um endereço de e-mail - utilizado para propósito de logging
- **Host**
 - indica o endereço e porta do serviço requisitado



Definição dos Headers

■ If-Match

- usado para tornar um método condicional. Verifica se a entidade requisitada tem o mesmo tag indicado

■ If-Modified-Since

- condiciona a execução do método à data de modificação da entidade solicitada

■ If-None-Match

- condiciona a execução do método ao fato de não haver nenhum match de tags

■ If-Range

- solicita somente a parte que foi modificada, de uma entidade multi-partes



Definição dos Headers

- **If-Unmodified-Since**
 - requisita entidades que não tenham sido modificadas desde a data indicada
- **Last-Modified**
 - indica a data e hora da última modificação registrada na entidade
- **Location**
 - utilizada para indicar a nova localização de uma entidade não mais armazenada na URI indicada
- **Max-Forwards**
 - utilizado para limitar o número de proxies e gateways que podem dar um forward na requisição enviada



Definição dos Headers

■ Pragma

- utilizada para incluir diretivas dependentes de implementação que podem ser aplicadas a apenas alguns dos servidores na cadeia de requisição/resposta

■ Proxy-Authenticate

- indica a autenticação necessária para acessar um determinado proxy

■ Proxy-Authorization

- apresenta as credenciais para a utilização de um proxy

■ Range

- indica o range de uma parte dentro de uma entidade multi-partes



Definição dos Headers

■ Referer

- indica ao servidor o endereço do cliente no qual a requisição foi originada

■ Retry-After

- no caso da entidade não estar disponível por algum motivo, indica quando poderia ser efetuada nova tentativa

■ Server

- indica o nome do software utilizado para fazer a requisição

■ TE

- indica qual extensão de codificações de transferência um cliente está apto a aceitar



Definição dos Headers

■ Trailer

- indica quais os header-fields estão presentes em uma mensagem utilizando codificação de transferência por chunk

■ Transfer-Encoding

- indica quais as codificações foram necessárias para a transmissão confiável da entidade

■ Upgrade

- indica os protocolos de comunicação adicionais disponíveis diante da necessidade de se mudar os protocolos

■ User-Agent

- contém informações sobre o programa originando a requisição



Definição dos Headers

- **Vary**
 - indica quais os headers de requisição que determinam se uma entidade no cache ainda está "fresca"
- **Via**
 - utilizada em gateways e proxies para indicar os protocolos a serem utilizados entre o cliente e o servidor final
- **Warning**
 - utilizado para trazer informações adicionais sobre o status de transformação de uma mensagem por um servidor intermediário
- **WWW-Authenticate**
 - indica o esquema de autenticação necessário para acessar uma determinada entidade protegida por controle de acesso