

Capítulo 1:

Conceitos iniciais

Sistemas Finais: Dispositivos onde a aplicação é executada

Enlaces de comunicação:

Comutadores (switches):

Taxa de transmissão: Quantidade de bit por unidade de tempo que um pacote é enviado da origem ao destino.

Rota: Sequência de enlaces de comunicação e comutadores de pacotes que um pacote percorre desde o sistema final transmissor até o sistema final receptor.

ISP's (Internet Service Provider): Uma rede de comutadores de pacotes e enlaces de comunicação. Todas as ISP's precisam estar conectadas direta ou indiretamente para que haja conexão de todos os sistemas finais conectados à diferentes ISP's.

API (Application Program Interface): Especificação de como o programa em um sistema final solicita à estrutura da internet que envie dados a um programa de destino específico, executado em outro sistema final. Conjunto de regras que o software emissor deve cumprir para que a internet seja capaz de enviar os dados ao programa destino.

Protocolo: Definição de um formato e ordem de mensagens trocadas entre duas ou mais entidades comunicantes, bem como as ações realizadas na transmissão e/ou recebimento de uma mensagem ou outro evento.

Redes de acesso

Redes de acesso: Rede física que conecta um sistema final ao primeiro roteador (conhecido como roteador de borda) de um caminho partindo de um sistema final a outro qualquer.

Nome	Tipo de acesso	Fornecida por	Estrutura	Taxa de transmissão	Característica notável
DSL (Linha digital de Assinante)	Doméstico	Operadora telefônica	Utiliza linha telefônica existente. Se conectam às CT, onde o sinal analógico é convertido para digital	até 24 MBits/s downstream até 2,5 MBits/s upstream	Distâncias curtas entre a casa e a CT
Internet à Cabo	Doméstico	Operadora de televisão	Utiliza infraestrutura de TV existente. Se conectam por cabo coaxial à nós de fibra ótica, ligados por fibra ótica à terminais de distribuição. Uso de fibra e coaxial (HFC)	até 42,8 MBits/s downstream até 30,7 Mbits/s upstream	Utiliza um meio de transmissão compartilhado . A taxa de envio e recebimento de pacotes de cada cliente é afetada pela atividade dos cliente

FTTH (Fiber to the Home)	Doméstico		Fibra ótica diretamente até a residência. Fibra direta (diretamente da CT até a casa) ou compartilhada (uma fibra para várias residências).	pode fornecer taxas de Gbits/s	Passive Optic Network (PON) e Active Optic Network (AON)
Ethernet	Empresa (e doméstico)		Par de fios de cobre trançado para conexão a um comutador ethernet. Comutador ethernet, ou uma rede de comutadores, é conectado à internet maior.	usuários com acesso normalmente de 100Mbps/s servidores com acesso de 1 Gbit/s até 10 Gbit/s	Principal rede usadas em Local Areas Networks (LAN's) em redes universitárias, corporativas e domésticas
Wi-fi	Empresa (e doméstico)		Usuários transmitem/recebem pacotes para/de um ponto de acesso remoto, que por sua vez se conecta à internet com fio		O usuário deve estar no espaço próximo de alguns metros do ponto de acesso.
3G	Sem fio em longa distância		Utilizam a infraestrutura sem fios usada para telefonia celular para enviar/receber pacotes por uma estação-base controlada pela operadora da rede celular.		O usuário pode estar a quilômetros de distância da estação-base

Meios físicos

Meios Físicos: Espaço material onde ocorre a propagação de bits.

Meios guiados: Meios físicos em que as ondas são dirigidas ao longo de um meio sólido como fibra ótica, par de fios trançado ou cabo coaxial.

Meios não guiados: Meios físicos em que as ondas se propagam na atmosfera e no espaço como o uso de uma LAN sem fio, 3g, ou canal digital de satélite.

Nome	Característica 1	Característica 2	Taxa de transmissão	Usabilidade
Par de fios de cobre trançado	Reduz interferência elétrica de pares semelhantes que estejam próximos	Um único enlace de comunicação	taxas de 10 Mbits/s a 10 Gbits/s. Dependem da bitola do fio e distância entre transmissor e receptor	Muito usado em acesso residencial
Cabo coaxial	Isolante e com blindagens especiais, alcança altas taxas de transmissão	Meio compartilhado guiado, onde vários sistemas finais podem ser conectados diretamente ao cabo	taxas de dezenas de Mbits/s	Usado em acesso residencial
Fibra ótica	Meio delgado e flexível que conduz pulsos de luz, cada um representando um bit	Suporta altíssimas taxas de transmissão. Imunes à interferência eletromagnética e baixíssima atenuação de sinal para longas distâncias	taxas de centenas de Gbits/s	Meio preferido para transmissão guiada de grande alcance, como cabos submarinos
Canais de rádio terrestres	Carregam sinais dentro do espectro eletromagnético	Não requerem cabos físicos, atravessam parede e transmitem sinal a grandes distâncias	depende do porte	desde redes locais a grandes distâncias
Canais de rádio por satélite	Um satélite de comunicação liga dois ou mais transmissores-receptores de	Uso de satélites geoestacionários ou satélites de órbita baixa (LEO)	centenas de Mbits/s	

	micro-ondas na terra. Recebe transmissões em uma frequência e repete o sinal em outra.			
--	--	--	--	--

Núcleo da rede

Pacotes: Fragmento de uma mensagem a ser enviada de um sistema final remetente a um sistema final destinatário.

Tempo para transmissão de um pacote: Se um pacote de L bits é enviado por um enlace a uma taxa de transmissão de R bits por segundo, então o pacote é transmitido em L/R segundos.

Transmissão armazena-e-reenvia: O comutador de pacotes deve receber o pacote inteiro antes de transmitir o primeiro bit para o enlace de saída.

Buffer de saída: Estrutura de um comutador de pacotes que armazena pacotes a serem enviados pelo roteador para um enlace de saída.

Atraso de fila: Quando comutador de pacotes está ocupado transmitindo um pacote mas um novo pacote que chega, ele é alocado em uma fila no buffer de saída. O tempo de espera no buffer é o atraso de fila.

Perda de pacote: Como o espaço no buffer é finito, um pacote que está chegando pode encontrá-lo lotado e ser descartado pelo comutador. Do ponto de vista de um sistema final, a perda de pacote é vista como um pacote que foi transmitido para o núcleo da rede mas não chegou ao destino.

Tabela de encaminhamento: Tabela contida nos roteadores que mapeia endereços de destino (aka IP) para um dado enlace de saída do roteador.

Comutação de circuitos: Os recursos necessários ao longo de um caminho (buffers, taxa de transmissão de enlaces) para oferecer a comunicação entre sistemas finais são reservados pelo período da sessão entre os hosts. É utilizado para esse fim a multiplexação por divisão de frequência (FDM) e a multiplexação por divisão de tempo (TDM).

Comutação de pacotes: Os recursos necessários para a comunicação entre dois sistemas finais não são reservados, as mensagens utilizam os recursos por demanda e podem ter que esperar para conseguir acesso a um enlace de comunicação. É utilizado o princípio da multiplexação estatística.

FDM (Multiplexação por divisão de frequência): O espectro de frequência de um enlace é compartilhado e as conexões são estabelecidas através desse enlace. O enlace reserva uma banda de frequência para cada conexão.

TDM (Multiplexação por divisão de tempo): O tempo de uso dos recursos de um enlace é dividido em quadros de duração fixa, onde cada quadro é dividido em um número fixo de compartimentos (slots). Cada slot é reservado para uso exclusivo de uma conexão e um dos compartimentos de tempo fica disponível para transmitir dados dela.

ISP global: Rede de roteadores e enlaces de comunicação que se espalham pelo planeta e se conectam a ISP's de acesso (abaixo na hierarquia).

ISP de nível 1: ISP onde ISP's regionais se conectam.

ISP regional: ISP onde diversas ISP's de acesso de uma região se conectam.

ISP de acesso: ISP de nível inferior, de menor escala na hierarquia.

PoP's (Pontos de presença): Um grupo de um ou mais roteadores na rede do provedor onde ISPs clientes podem se conectar a um ISP provedor. Presentes em todos os níveis de hierarquia de ISP, exceto na de acesso.

Multi-home: Conexão de um ISP (exceto de acesso) a dois ou mais ISP's provedores quaisquer.

Emparelhamento: Conexão direta entre ISP's de uma mesma hierarquia de modo que o tráfego entre elas passe por uma conexão direta, evitando intermediários.

IXP (Internet exchange point): Ponto de encontro onde vários ISP's podem se emparelhar.

Redes de provedor de conteúdo: Rede própria de um provedor de conteúdo, se conectando diretamente a ISP's de níveis 1, IXP, regional e de acesso, reduzindo seus pagamentos à ISP's da camada mais alta e tendo maior controle sobre como seus serviços são entregues aos usuários finais.

Atraso, perda e vazão em redes de comutação de pacotes

Atraso de processamento nodal: Tempo exigido para examinar o cabeçalho do pacote e determinar para onde direcioná-lo e verificar os erros em bits existentes no pacote. Da ordem de microssegundos.

Atraso de fila: Tempo de espera de um pacote para ser transmitido em um enlace. Depende da quantidade de pacotes que chegaram antes e estão aguardando na fila. Da ordem de micro a milissegundos.

Atraso de transmissão: Tempo necessário para transmitir todos os bits do pacote para o enlace. Dada pela relação entre tamanho L de bits de um pacote e a taxa de transmissão do roteador bits/s, L/R segundos. Da ordem de micro a milissegundos.

Atraso de propagação: Tempo necessário para um bit propagar pelo enlace de um roteador A a um roteador B. Em função da velocidade de propagação do enlace, na faixa de $2 \cdot 10^8$ m/s a $3 \cdot 10^8$ m/s, que depende do meio físico (fibra ótica, par de fios de cobre trançado, etc). Dada pela relação entre a distância total do enlace d e a velocidade de propagação do enlace s , d/s .

Atraso nodal total: Resultado da soma de todos os atrasos anteriores. Referente à análise em um roteador.

Intensidade de tráfego: Razão entre a taxa média a que pacotes chegam à uma fila, quantidade de bits L de um pacote e taxa de transmissão R do roteador, dada por La/R . Se $La/R > 1$ a velocidade média que os bits chegam à fila excede a que eles são transmitidos para fora da fila. Nesse caso, a fila tende a aumentar sem limite, cujo atraso de fila tenderá ao infinito.

Atraso fim a fim: Resultado pelo atraso de processamento, propagação e transmissão de $N - 1$ roteadores entre a máquina de origem e a de destino. Dado por $d = N(d_{proc} + d_{trans} + d_{prop})$.

Vazão instantânea: Dada pela taxa (em bits/s) em que um hospedeiro está recebendo um arquivo.

Vazão média: Dada pela taxa média (em bits/s) em que um hospedeiro recebe um arquivo com F bits em um intervalo T de tempo, cujo valor é F/T bits/s.

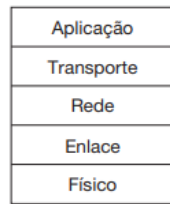
Vazão servidor-cliente: Taxa em que um cliente recebe dados (em bits) de um servidor. Depende dos valores das taxas de transmissões R dos enlaces entre os dois. Se há apenas dois enlaces com taxas R_c e R_s , enlace do cliente e do servidor, a vazão é dada por $\min\{R_c, R_s\}$.

Enlace de gargalo: Referente ao processo de análise da menor taxa de transmissão para a vazão servidor-cliente, como visto acima.

Camadas de protocolo e seus modelos de serviço

Camadas de protocolo: Estrutura de design de redes que organiza diferentes protocolos em estruturas modulares em camadas que executam processos e oferecem serviços às camadas superiores da estrutura. Cada protocolo pertence a uma camada. Uma camada de protocolo pode ser executada em software, hardware ou uma combinação dos dois.

Pilha de protocolos: Organização de vários protocolos de várias camadas. Formada por camadas física, de enlace, de rede, de transporte e de aplicação.



a. Pilha de protocolos da Internet de cinco camadas

Camada de aplicação: Onde residem as aplicações de rede e seus protocolos. Inclui os protocolos *HTTP* e *FTP*. Pacote denominado *mensagem*.

Camada de transporte: Carrega mensagens da camada de aplicação entre os lados cliente e servidor de uma aplicação. Inclui os protocolos *TCP* (orientado a conexão, entrega garantida de mensagens, controle de fluxo, controle de congestionamento) e *UDP* (não orientado a conexão, econômico e sem controle de fluxo e congestionamento). Pacote denominado *segmento*.

Camada de rede: Responsável pela movimentação, de um host a outro, de pacotes da camada de rede. O protocolo da camada de transporte em um host de origem passa um *segmento* da camada de transporte e um endereço de destino à camada de rede. A camada de rede então provê o serviço de entrega do *segmento* à camada de transporte no host destino. Inclui o protocolo *IP*. Pacote denominado *datagrama*.

Camada de enlace: Roteia um *datagrama* por meio de uma série de roteadores entre a origem e destino. Em cada nó, a camada de rede passa o *datagrama* para a de enlace, que o entrega, ao longo da rota, ao nó seguinte, no qual o *datagrama* é passado na camada de enlace para a de rede. Inclui os protocolos *Ethernet*, *Wi-Fi* e *DOCSIS*. Pacote denominado *quadro*.

Camada física: Movimenta os bits individualmente que estão dentro de um *quadro* de um nó ao nó seguinte. Inclui os protocolos *par de fibra trançado*, *cabo coaxial*, *fibra ótica*, etc. Em cada caso, o bit é transmitido pelo enlace de forma diferente.

Encapsulamento: Processo de adição de informações (cabeçalho) de um pacote de uma camada em uma camada inferior, que será usada para a execução do serviço prestado pela camada. Uma *mensagem* quando transmitida da camada de aplicação para transporte recebe um header e se transforma em um *segmento*. Esse processo ocorre em todas as camadas, atravessando a rede de roteadores onde em cada nó de uma respectiva camada o pacote será desencapsulado e o header referente à camada que o recebe será lido.

Comutadores da camada de enlace são utilizados em **redes de acesso**, enquanto os **roteadores** são utilizados principalmente no **núcleo da rede**.

Capítulo 2:

Arquiteturas de aplicações de rede

Arquitetura de rede: É fixa e provê um conjunto específico de serviços.

Arquitetura da aplicação: É projetada pelo programador e determina como a aplicação é organizada nos vários sistemas finais. Podem ser do tipo *cliente-servidor* ou *P2P*.

Arquitetura cliente-servidor: Há um hospedeiro sempre em funcionamento chamado *servidor* que atende requisições de outros sistemas finais chamados *clientes*. Exemplos clássicos são aplicações Web, FTP, Telnet e e-mail.

Datacenter: Estrutura que comporta um grande número de hosts. Usado com frequência para criar um servidor virtual poderoso e contornar problemas de conexão de clientes com um único host.

Arquitetura P2P: Comunicação direta entre duplas de hosts conectados alternadamente, chamados *pares*. Eles não são de propriedade dos provedores de serviço, mas são controlados pelos usuários de computadores em redes domésticas, universitárias, etc. Aplicações comuns dessa estrutura são compartilhamento de arquivos (como BitTorrent), aceleração de download assistida por par (Xunlei) telefonia de internet (como Skype), etc.

A arquitetura é *autoescalável*. Embora cada par gere uma carga de trabalho solicitando arquivos, também cede capacidade de serviço ao sistema distribuindo arquivo aos pares. Em geral, não requerem infraestrutura e largura de banda de servidor significativas, embora precisem lidar com três desafios:

1. **ISP amigável:** A maioria dos ISPs residenciais, por design, fornecem mais largura de banda para download do que upload. A arquitetura P2P exige largura de banda de upload, colocando pressão sobre as ISPs.
2. **Segurança:** Pela natureza altamente distribuída e exposta, a proteção em aplicações P2P é um desafio.
3. **Incentivo:** O sucesso de aplicações P2P exige a participação dos seus usuários na rede para oferecer largura de banda, armazenamento e recurso da computação às aplicações, sendo necessário um projeto de incentivo aos usuários para permanência na rede e concessão desses recursos.

Comunicação entre processos

Processo: Um programa que roda em um sistema final. Quando rodam em um mesmo sistema final, comunicam-se pelo sistema operacional do host. Quando rodam em sistemas finais diferentes, se comunicam trocando mensagens por meio da rede de computadores. Para cada par de processos comunicantes, um é denominado *cliente* e outro de *servidor*.

Processo cliente: Processo que inicia a comunicação com outro processo.

Processo servidor: Processo que espera ser contatado para iniciar a sessão.

Socket: Interface de software por onde um processo envia e recebe mensagens de outro processo. Interface entre a camada de aplicação e a de transporte dentro de um hospedeiro. A API entre a aplicação e a rede, visto que é a interface de programação pela qual as aplicações de rede são criadas.

Endereço IP: Identificação de um hospedeiro.

Número de porta: Identificação do processo receptor executando no endereço de destino. Essa informação é importante porque uma mensagem que chega ao destino precisa saber a qual aplicação a mensagem deve ser entregue, uma vez que o host pode estar executando várias aplicações.

Serviços de transporte disponíveis para aplicações

Um socket é a interface entre o processo da aplicação e o protocolo da camada de transporte. A aplicação do lado remetente envia mensagens por meio do socket. Do outro lado, o protocolo da camada de transporte tem a responsabilidade de levar as mensagens pela rede até o socket do processo destinatário. Os protocolos da camada de transporte podem oferecer às aplicações diferentes serviços, são eles *transferência confiável de dados, vazão, temporização e segurança*.

Transferência confiável de dados: Garantia que os dados enviados por uma extremidade da aplicação sejam transmitidos pela rede de forma correta e completamente para outra ponta.

Aplicações tolerantes a perda: Não é exigida pela aplicação a entrega íntegra dos dados enviados, como as de multimídia de áudio/vídeo em tempo real ou áudio/vídeo armazenado.

Vazão: No contexto de sessão da comunicação entre dois processos ao longo de um caminho da rede, é a taxa pela qual o processo pode enviar bits ao processo destinatário.

Aplicações sensíveis à largura de banda: Possuem necessidade de vazão, como algumas aplicações multimídia.

Aplicações elásticas: Podem usar qualquer quantidade mínima ou máxima de largura de banda que esteja disponível, como correio eletrônico, transferência de arquivos e transferências web.

Temporização: Garantia de que dados cheguem em um certo intervalo de tempo, como o fato de que cada bit que o remetente insere no socket chega ao socket destinatário em menos de 100ms depois.

Atrativo para serviço de telefonia por internet, teleconferência e jogos multiplayer.

Segurança: Garantia de segurança de dados, como um protocolo de transporte ser capaz de realizar a codificação de todos os dados transmitidos pelo processo remetente e, no hospedeiro destinatário, o protocolo da camada de transporte pode codificar os dados antes de enviá-los ao destinatário.

Aplicações bancárias podem exigir esse serviço.

Serviços de transporte providos pela internet

A camada de transporte disponibiliza dois protocolos: *TCP* e *UDP*.

FIGURA 2.4 REQUISITOS DE APLICAÇÕES DE REDE SELECIONADAS

Aplicação	Perda de dados	Vazão	Sensibilidade ao tempo
Transferência / download de arquivo	Sem perda	Elástica	Não
E-mail	Sem perda	Elástica	Não
Documentos Web	Sem perda	Elástica (alguns kbits/s)	Não
Telefonia via Internet/ videoconferência	Tolerante à perda	Áudio: alguns kbits/s – 1Mbit/s Vídeo: 10 kbits/s – 5 Mbits/s	Sim: décimos de segundo
Áudio/vídeo armazenado	Tolerante à perda	Igual acima	Sim: alguns segundos
Jogos interativos	Tolerante à perda	Poucos kbits/s – 10 kbits/s	Sim: décimos de segundo
Mensagem instantânea	Sem perda	Elástico	Sim e não

Extrema importância

Serviços do TCP:

1. Serviço orientado à conexão: O TCP faz o cliente e servidor trocarem informações de controle de camada de transporte antes de começar a comunicação para envio/recebimento de mensagens. Procedimento conhecido como *apresentação (handshake)*. Após a apresentação, é estabelecida a **conexão TCP** entre processo servidor e cliente onde ambos podem simultaneamente enviar mensagens um para o outro.
2. Serviço confiável de transporte: Os dados enviados pela conexão TCP serão enviados corretamente e completamente.
3. Possui controle de congestionamento, que pode limitar a capacidade de transmissão de um processo quando a rede está congestionada entre o remetente e o destinatário.

Serviços do UDP: Serviço não orientado a conexão, não há handshake. Não provê serviço confiável de dados nem inclui um mecanismo de controle de congestionamento, o que permite o processo originador bombardear dados para dentro de uma camada abaixo à taxa que quiser.

FIGURA 2.5 APLICAÇÕES POPULARES DA INTERNET, SEUS PROTOCOLOS DE CAMADA DE APLICAÇÃO E SEUS PROTOCOLOS DE TRANSPORTE SUBJACENTES

Aplicação	Protocolo de camada de aplicação	Protocolo de transporte subjacente
Correio eletrônico	SMTP [RFC 5321]	TCP
Acesso a terminal remoto	Telnet [RFC 854]	TCP
Web	HTTP [RFC 2616]	TCP
Transferência de arquivos	FTP [RFC 959]	TCP
Multimídia em fluxo contínuo	HTTP (por exemplo, YouTube)	TCP
Telefonia por Internet	SIP [RFC 3261], RTP [RFC 3550] ou proprietária (por exemplo, Skype)	UDP ou TCP

Protocolos de camada de aplicação: Define como processos de uma aplicação, que funcionam em sistemas finais diferentes, passam mensagens entre si. Define os tipos de mensagens trocadas, as sintaxes dos vários tipos de mensagens, semântica dos campos e regras para determinar quando e como um processo envia mensagens e as responde.

Descrição geral do HTTP

HTTP (HyperText Transfer Protocol): Protocolo da camada de aplicação da Web, definido no RFC 2616. Executado em um cliente e servidor. Os sistemas finais diferentes conversam entre si por meio da troca de mensagens HTTP. O HTTP define a estrutura dessas mensagens e como os hosts as trocam.

Página Web: Documento Web constituída por objetos.

Objeto: Um arquivo que pode ser acessado com um único endereço URL.

Servidores Web: Abrigam objetos Web, cada um endereçado por um URL.

Quando um usuário requisita uma página Web, o navegador envia ao servidor uma requisição HTTP para os objetos da página. O servidor recebe as requisições e responde com mensagens de resposta HTTP contendo os objetos. O servidor, ao enviar os arquivos solicitados, não armazena qualquer informação de estado sobre o cliente, em função disso, o HTTP é dito um **protocolo sem estado**.

Conexões persistentes e não persistentes

Conexão persistente: Uso de uma mesma conexão para cada par de requisição/resposta em uma interação cliente-servidor. HTTP é, por padrão, persistente.

Conexão não persistente: Uso de uma conexão distinta para cada par de requisição/resposta em uma interação cliente-servidor.

HTTP com conexões não persistentes: Suponhamos que uma página consista em um arquivo-base HTML e em dez imagens JPEG e que todos esses arquivos estão no servidor. Suponha o URL para o arquivo <http://www.someschool.edu/someDepartment/home.index>.

1. O processo cliente HTTP inicia uma conexão TCP com o servidor www.someSchool.edu na porta 80, default para HTTP. Em uma conexão TCP, haverá um socket no cliente e servidor
2. O cliente HTTP envia uma mensagem de requisição HTTP ao servidor por meio de seu socket. A mensagem inclui o nome de caminho /someDepartment/home.index
3. O processo servidor HTTP recebe a mensagem de requisição por meio de seu socket, extrai o objeto requisitado de seu armazenamento (RAM), o encapsula em uma mensagem de resposta HTTP e envia ao cliente pelo socket.
4. O processo servidor ordena que o TCP encerre a conexão TCP.

5. O cliente HTTP recebe a mensagem de resposta e a conexão TCP é encerrada. A mensagem indica que o objeto encapsulado é um arquivo HTML. O cliente extrai o arquivo da mensagem e resposta, analisa o arquivo HTML e encontra referências aos dez objetos JPEG.
6. Para cada objeto JPEG, as etapas 1-4 são repetidas.

HTTP com conexões persistente: Segue o esquema anterior, mas o servidor deixa a conexão aberta após enviar a resposta na etapa 3. Requisições e respostas subsequentes entre os mesmos cliente e servidor podem ser enviadas por meio da mesma conexão.

RTT (Round-trip time): Tempo que leva para um pequeno pacote viajar do cliente ao servidor de volta ao cliente.

Mensagem de requisição HTTP

```
GET /somedir/page.html HTTP/1.1
Host: www.someschool.edu
Connection: close
User-agent: Mozilla/5.0
Accept-language: fr
```

Linha de requisição: Primeira linha contendo três campos: método, URL e versão do HTTP

Métodos HTTP: GET (solicitar página), POST (solicita página em função de preenchimento do formulário), HEAD (solicita informações de uma página, mas não o objeto), PUT (carregar um objeto a um diretório no servidor), DELETE (remover um objeto do servidor).

Linhas de cabeçalho: Demais linhas da mensagem.

Host: Especifica onde o objeto reside.

Connection: Informa ao servidor se quer usar conexões persistentes (open) ou não (close).

User-agent: Especifica o agente de usuário, o navegador que está fazendo a requisição do servidor.

Accept-language: Especifica que o usuário prefere receber uma versão em determinada língua do objeto solicitado.

Mensagem de resposta HTTP

```
HTTP/1.1 200 OK
Connection: close
Date: Tue, 09 Aug 2011 15:44:04 GMT
Server: Apache/2.2.3 (CentOS)
Last-Modified: Tue, 09 Aug 2011 15:11:03 GMT
Content-Length: 6821
Content-Type: text/html
```

(dados dados dados dados dados ...)

Linha de estado: Primeira linha de resposta. Possui três campos: versão do protocolo, código de estado e mensagem do estado correspondente.

- **200 OK:** requisição bem-sucedida e a informação é entregue com a resposta.
- **301 Moved Permanently:** objeto requisitado foi removido em definitivo; novo URL é especificado no cabeçalho Location: da mensagem de resposta. O software do cliente recuperará automaticamente o novo URL.
- **400 Bad Request:** código genérico de erro que indica que a requisição não pôde ser entendida pelo servidor.
- **404 Not Found:** o documento requisitado não existe no servidor.
- **505 HTTP Version Not Supported:** a versão do protocolo HTTP requisitada não é suportada pelo servidor.

Linhas de cabeçalho: Demais linhas da resposta.

Connection: Informa ao cliente se fechará (close) ou não (open) a conexão TCP após o envio da mensagem.

Date: Informa a hora e a data em que a resposta HTTP foi criada e enviada pelo servidor.

Server: Informa o servidor em que a mensagem foi criada.

Last-Modified: Informa a data e hora em que o objeto foi criado ou sofreu a última modificação.

Content-Length: Informa o número de bytes do objeto que está sendo enviado

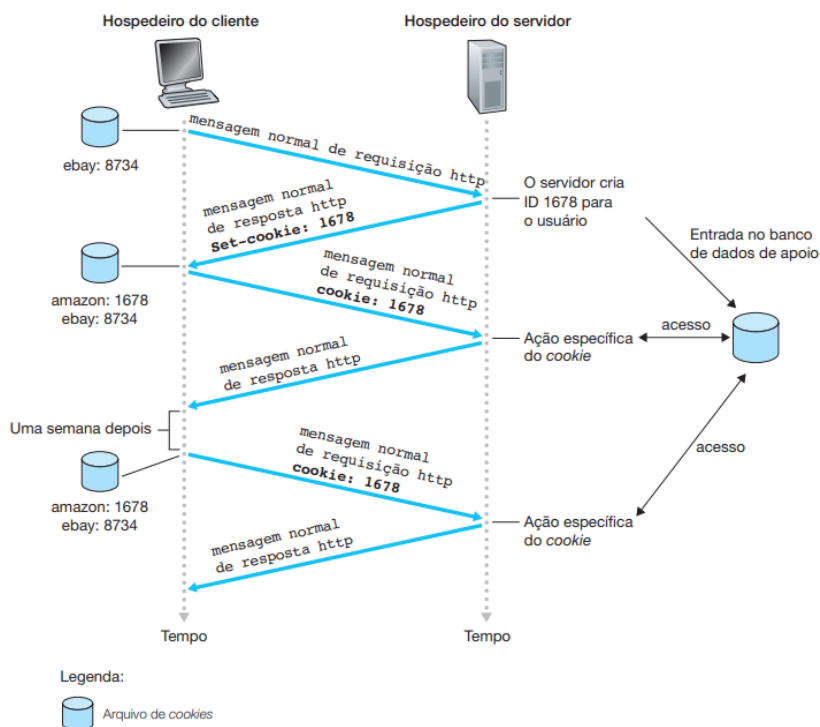
Content-Type: Informa o tipo do objeto enviado

Corpo da entidade: Contém o objeto solicitado.

Interação usuário-servidor: cookies

Cookies: Tecnologia para simular estado em processos HTTP. Possui quatro componentes:

1. Uma linha de cabeçalho de cookie na mensagem de resposta HTTP
2. Uma linha de cabeçalho de cookie na mensagem de requisição HTTP
3. Um arquivo de cookie mantido no sistema final do usuário e gerenciado pelo navegador do usuário
4. Um banco de dados de apoio no site



Caches Web

Caches Web (Servidor proxy): Uma entidade da rede que atende requisições HTTP em nome do servidor Web de origem. O cache web tem seu próprio disco de armazenamento e mantém, dentro dele, cópias de objetos recentemente requisitados.

Suponhamos que um navegador esteja requisitando o objeto <https://someSchool.edu/campus.gif>. O que acontece é:

1. O navegador estabelece uma conexão TCP com o cache Web e envia a ele uma requisição HTTP para o objeto.
2. O cache Web verifica se tem uma cópia armazenada localmente. Se tiver, envia o objeto ao navegador do cliente dentro de uma mensagem de resposta HTTP
3. Se não tiver, o cache web abre uma conexão TCP com o servidor de origem, isso é, com o www.someSchool.edu. Então, envia uma requisição HTTP do objeto para a conexão TCP. Ao recebê-la, o servidor de origem envia o objeto solicitado ao cache Web dentro de uma resposta HTTP.
4. Quando recebe o objeto, o cache web guarda uma cópia em seu armazenamento local e envia outra, dentro de uma resposta HTTP, ao navegador cliente.

GET condicional: Uma mensagem de requisição HTTP em que é usado o método GET e possui uma linha de cabeçalho *If-Modified-Since*. Mecanismo que permite que um cache verifique se seus objetos estão atualizados.

FTP

Em uma sessão FTP típica, o usuário quer transferir arquivos de ou para um host remoto. Para acessar a conta remota, o usuário deve fornecer uma identificação e uma senha. Dadas as informações de autorização, pode transferir dados de um sistema remoto ou vice-versa.

O usuário interage com o FTP por meio de um agente de usuário FTP. Primeiro ele fornece o nome do hospedeiro remoto, o que faz com que o processo cliente FTP do host local estabeleça uma conexão TCP com o processo servidor FTP do host remoto. O usuário então fornece sua identificação e senha, que são enviadas pela conexão TCP como parte dos comandos FTP. Assim que autorizado pelo servidor, o usuário copia um ou mais arquivos armazenados no sistema de arquivo local para o sistema de arquivo remoto (ou vice-versa). FTP é constituído por uma *conexão de controle* e uma *conexão de dados*. Como usa duas conexões separadas, o FTP envia informações de controle **fora da banda**. O FTP mantém informações de estado sobre o usuário. Cada operação sobre um arquivo implica na abertura e encerramento de uma conexão de dados.

Conexão de controle: Usada para enviar informações de controle entre os dois hospedeiros, como identificação de usuário, senha, comandos para troca de diretório, comandos de enviar (pu) e receber (get) arquivos.

Conexão de dados: Usada efetivamente para transporte de arquivos.

Comandos FTP:

- **USER username:** usado para enviar identificação do usuário ao servidor.
- **PASS password:** usado para enviar a senha do usuário ao servidor.

- **LIST:** usado para pedir ao servidor que envie uma lista com todos os arquivos existentes no atual diretório remoto. A lista de arquivos é enviada por meio de uma conexão de dados (nova e não persistente), e não pela conexão TCP de controle.
- **RETR filename:** usado para extrair (isto é, obter) um arquivo do diretório atual do hospedeiro remoto. Ativa o hospedeiro remoto para que abra uma conexão de dados e envia o arquivo requisitado por essa conexão.
- **STOR filename:** usado para armazenar (isto é, inserir) um arquivo no diretório atual do hospedeiro remoto.

Respostas FTP:

- **331 Nome de usuário OK, senha requisitada**
- **125 Conexão de dados já aberta; iniciando transferência**
- **425 Não é possível abrir a conexão de dados**
- **452 Erro ao escrever o arquivo**

SMTP (Simple Mail Transfer Protocol)

SMTP: Principal protocolo de camada de aplicação para correio eletrônico na internet. Usa o serviço confiável de transferência de dados do TCP para transferir mensagens do servidor de correio do remetente para o do destinatário. Não é utilizado servidores intermediários para envio de correspondência. Uso de conexões persistentes.

Para ilustrar essa operação básica do SMTP, vamos percorrer um cenário comum. Suponha que Alice queira enviar a Bob uma simples mensagem ASCII.

1. Alice chama seu agente de usuário para e-mail, fornece o endereço de Bob (por exemplo, bob@some-school.edu), compõe uma mensagem e instrui o agente de usuário a enviá-la.
2. O agente de usuário de Alice envia a mensagem para seu servidor de correio, onde ela é colocada em uma fila de mensagens.
3. O lado cliente do SMTP, que funciona no servidor de correio de Alice, vê a mensagem na fila e abre uma conexão TCP para um servidor SMTP, que funciona no servidor de correio de Bob.
4. Após alguns procedimentos iniciais de apresentação (*handshaking*), o cliente SMTP envia a mensagem de Alice pela conexão TCP.
5. No servidor de correio de Bob, o lado servidor do SMTP recebe a mensagem e a coloca na caixa postal dele.
6. Bob chama seu agente de usuário para ler a mensagem quando for mais conveniente para ele.

Diferenças entre HTTP e SMTP

Ambos persistentes, ambos usam TCP, ambos transferem arquivos, ambos utilizam arquitetura cliente-servidor, mas:

HTTP é um protocolo de recuperação: Carrega informações de um servidor Web através de requisições.

SMTP é um protocolo de envio: O serviço de correio remetente envia o arquivo para o servidor de correio destinatário

SMTP exige o corpo da mensagem no formato ASCII de 7 bits. HTTP não faz restrição quanto aos caracteres da mensagem.

HTTP encapsula os objetos. SMTP apenas coloca todos os objetos da mensagem em uma única mensagem.

Formatos de mensagem de correio

Cabeçalho: Composto pelos campos *from*, *to* e *subject*.

From: alice@crepes.fr

To: bob@hamburger.edu

Subject: Searching for the meaning of life.

POP3

muito cansado para fazer

IMAP

muito cansado para fazer

DNS: o serviço de diretório da internet

Nome de hospedeiro: Identificador de um hospedeiro (hostname), como cnn.com, www.yahoo.com, etc. Fáceis de lembrar.

Endereço IP: Identificação “real” de hosts. Constituído de 4 bytes e sua estrutura hierárquica é rígida. Como, por exemplo, 121.7.106.83.

DNS (Domain name system): Um banco de dados distribuído executado em uma hierarquia de servidores de DNS e um protocolo de camada de aplicação que permite que hospedeiros consultem o banco de dados distribuído. Utiliza *UDP* e a porta 53. Muito utilizado por outras entidades da camada de aplicação para traduzir nomes de hospedeiros fornecedor por usuários para endereços IP.

Suponha como exemplo o que acontece quando um navegador (cliente HTTP) requisita a URL www.someSchoool.edu/index.html. Para que a máquina do usuário possa enviar uma requisição ao endereço www.someSchoool.edu, ela precisa primeiro obter seu endereço IP. Isso é feito da seguinte maneira:

1. A própria máquina do usuário executa o lado cliente da aplicação DNS
2. O navegador extrai o nome de hospedeiro www.someSchoool.edu do URL e passa o nome para o lado cliente da aplicação DNS
3. O cliente DNS envia uma consulta contendo o nome de hospedeiro para um servidor DNS
4. O cliente DNS recebe uma resposta contendo o endereço IP correspondente do nome de hospedeiro fornecido
5. Recebido o endereço do DNS, o navegador pode abrir uma conexão TCP com o processo servidor HTTP na porta 80 naquele endereço IP

Serviços fornecidos pelo DNS:

1. Tradução de nomes de hospedeiro para endereço IP
2. **Apelidos (alias) de hospedeiro.** Nome fictício dado a um hospedeiro ao invés de seu **nome canônico**. O DNS pode fornecer um nome canônico a um alias
3. **Apelidos de servidor de correio.** Segue o princípio de alias para endereços de e-mail
4. **Distribuição de carga.** O DNS pode atribuir a um nome de hospedeiro um endereço IP específico de uma rede de sistemas finais a fim de distribuir a carga entre servidores replicados. No caso, a um sistema final servidor é associado a uma lista de endereços IP e o DNS rotaciona o acesso entre os elementos dessa lista.

Servidores DNS raiz: Conglomerado de servidores replicados. Primeira entidade da rede distribuída de DNS a ser contactada por uma chamada de uma aplicação DNS cliente. Responsável por informar endereços IP dos servidores TLD para o domínio de alto nível do endereço solicitado.

Servidores DNS TLD (top-level domain): Responsáveis por domínios de alto nível como *com*, *org*, *net*, *edu*, *br*.

Servidores DNS autoritativos: Servidor DNS que abriga registros de uma organização mapeando nomes de hospedeiros a endereços IP.

Servidores DNS locais: Servidores DNS associados à ISP's. Quando um hospedeiro se conecta com um ISP, este fornece os endereços IP de um ou mais de seus servidores DNS locais. Quando um hospedeiro faz uma consulta DNS, ela é enviada ao servidor DNS local, que age como *proxy* e a retransmite para a hierarquia do servidor DNS.

Consultas recursivas: Quando um cliente utiliza um servidor realizar uma ação seu nome, iterando por outras comunicações (um cliente usar um servidor DNS local para fazer mapeamento de nome de hospedeiro para endereço IP em seu nome)

Consultas iterativas: Quando a consulta é feita sem intermediários, diretamente entre duas entidades (servidor DNS local solicita a um servidor DNS raiz e recebe resposta deste).

Cache DNS: Tecnologia para otimizar as consultas à rede distribuída DNS. Em uma cadeia de consultas, quando um servidor DNS recebe uma resposta DNS ele pode fazer cache da resposta em sua memória local. Se um par nome de host/endereço IP estiver na cache de um servidor DNS, ele pode fornecer a informação caso seja solicitado a ele mesmo sem ter autoridade para esse nome.

RR (registro de recursos): Registros armazenados em servidores DNS para mapeamento de nomes de hospedeiros para endereço IP. Cada mensagem de resposta DNS carrega um ou mais registros de recursos. Um registro é uma tupla (Name, Value, Type, TLL). TLL é o tempo de vida útil do registro, que é usado no cache. *Value* e *Name* dependem de *Type*:

REGISTRO = (NAME, VALUE, TYPE, TLL)

1. **Type = A:** (nome de hospedeiro (canônico), endereço IP, A)
 - a. (relay1.bar.foo.com, 145.37.93.126, A)
2. **Type = NS:** (domínio, servidor DNS autoritativo, NS)
 - a. (foo.com, dns.foo.com, NS)
3. **Type = CNAME:** (alias, nome de hospedeiro (canônico), CNAME)
 - a. (foo.com, relay1.bar.foo.com, CNAME)
4. **Type = MX:** (alias, nome canônico de um servidor de correio, MX)
 - a. (foo.com, mail.bar.foo.com, MX)

Entidade registradora: Organização comercial que verifica se o nome de domínio é exclusivo, registra-o no banco de dados DNS e cobra uma pequena taxa por seus serviços. É encarregada de providenciar a inserção dos registros *Type NS* e *Type A* nos servidores TLD do domínio correspondente para cada um dos servidores de nomes autorizados.

BitTorrent

Torrent: Coleção de todos os pares que participam da distribuição de um determinado arquivo.

Rastreador: Nó de infraestrutura presente em todos os *torrents*. O rastreador mantém um registro de pares que participam do *torrent*. Todo novo par que entra na torrent se registra com o rastreador e periodicamente informa se ainda está lá.

Funcionamento geral: Quando um novo par, Alice, chega à torrent, o rastreador seleciona aleatoriamente um subconjunto de pares do conjunto de pares participantes e envia seus endereços IP ao novo par. Com a lista de pares, Alice tenta estabelecer uma conexão TCP simultaneamente com todos, sendo eles seu pares vizinhos (que rotacionam com outros pares periodicamente). Em um certo momento, cada par terá um subconjunto de blocos de arquivo, com pares diferentes com conjuntos diferentes. Periodicamente, Alice pede a cada um de seus pares vizinhos a lista de blocos que

possuem, podendo emitir solicitação dos blocos que ela não possui e atendendo solicitações de outros pares.

Rarest first: Técnica que consiste em um par determinar e solicitar, dentre os blocos que não possui, os que são mais raros entre seus vizinhos (blocos com menor número de cópias retidas). Dessa forma, os blocos mais raros são redistribuídos mais depressa, buscando equalizar o número de cópias de cada bloco na torrent.

Unchoked: Jargão do BitTorrent designado aos pares cuja taxa de transmissão de bits a um par é alta e ele, reciprocamente, envia com prioridade blocos a esses mesmos pares.

Optimistically unchoked: Jargão do BitTorrent designado a um par escolhido aleatoriamente (e periodicamente) por outro para enviar blocos, com o intuito de analisar a taxa de transmissão e estabelecer (ou não) uma relação de prioridade na rede. Nesse processo, pares com taxas de transmissão compatíveis tendem a se encontrar.

Tit-for-tat: Mecanismo de incentivo da rede BitTorrent, descrito acima.

DHTs (distributed hash tables)

Banco de dados distribuído sobre uma rede P2P tal que cada par só mantém um pequeno conjunto da totalidade (chave, valor). Qualquer par pode consultar o banco de dados com uma chave particular. O banco de dados distribuído, então, localizará os peers que possuem os pares (chave, valor) correspondentes e retornará os pares ao consultante. Qualquer peer pode inserir novos pares (chave, valor) no banco de dados.

Rede sobreposta: Distribuição circular dos pares de uma DHT onde cada par rastreia apenas seu sucessor e predecessor imediatos. Nesse tipo de rede, os pares formam uma rede lógica abstrata que reside acima da de computadores “inferior” que consiste de enlaces físicos, roteadores e hosts.

DHT com atalho: Utiliza a arquitetura de rede sobreposta adicionando uma informação aos pares, que além de saberem seu antecessor e sucessor imediato, identificam um terceiro par adiante do sucessor.

Peer Churn

Abordagem em uma rede sobreposta de DHTs que consiste em cada par rastreie (ou seja, saiba o endereço IP de) seu primeiro e segundo sucessores. É exigido ainda que cada par verifique de tempos em tempos se seus dois sucessores estão vivos. Quando um par sai bruscamente, a rede é capaz de se readaptar, onde os pares que rastreavam o peer que saiu podem se reestruturar, substituindo seus sucessores rastreados.