



Unidade 3 - Webaula 10 - Camada de Aplicação - Suplementar

Olá! Tudo bem?

Seja bem-vindo(a) à Webaula 10 de **Redes de Computadores**.

INTRODUÇÃO

Introdução à Webaula 10

TÓPICO 1

World Wide Web

Atividade de Passagem

TÓPICO 2

Redes Multimídia

Atividade de Passagem

RESUMO

Resumo da Webaula 10

Conclusão da Unidade

REFERÊNCIAS

Referências

Créditos

Introdução à Webaula 10

Olá!

Nesta aula abordarei alguns tipos de aplicações de rede para internet consideradas as mais importantes, atuais e significativas: navegação na internet e redes multimídia.

O objetivo desta aula é que você compreenda como funcionam os mecanismos de transferência de dados na internet, tanto na transferência de conteúdo web quanto de conteúdo multimídia.

CONTINUE

World Wide Web

A *World Wide Web* (**WWW**, popularmente conhecido como **internet** ou simplesmente **web**) teve início em 1989 como resultado do trabalho do cientista Tim Berners-Lee do CERN, um centro europeu para pesquisa nuclear, que visava permitir que grupos de cientistas de diversos países pudessem compartilhar seus trabalhos através da divulgação de relatórios, plantas, desenhos, fotos e vários outros tipos de documentos.

Do ponto de vista dos usuários, a internet é uma vasta coleção mundial de documentos, chamados **páginas**, distribuídos em **sites** da *World Wide Web*. Cada página pode conter **links** para outras páginas em qualquer lugar do mundo. As páginas da internet são visualizadas com o auxílio de um programa denominado **navegador internet**, que busca a página solicitada, interpreta seu texto e seus comandos de formatação e a exibe formatada.

Mas você imagina a quanto tempo que se navega na internet como estamos acostumados atualmente? Pois o primeiro navegador gráfico para a internet surgiu somente no início de 1993 na Universidade de Illinois. Foi desenvolvido por Marc Andreessen e se chamava Mosaic. A evolução deste navegador internet levou ao desenvolvimento do Netscape Navigator. Em 1995, a Microsoft entrou no mercado de navegadores internet com o Internet Explorer. A partir de 1998, com a disponibilização do código fonte do Netscape Navigator, criou-se uma organização chamada Mozilla que desenvolveu o Mozilla Browser utilizando um interpretador próprio de conteúdo chamado Gecko. A evolução deste navegador pela própria comunidade internet levou ao lançamento em 2004 do Mozilla Firefox. Em 2000 a empresa norueguesa Telenor lançou o Opera, primeiro navegador disponível também para aparelhos celulares (KUROSE; ROSS, 2013).

Em 2000 criou-se um novo interpretador de conteúdo internet não proprietário chamado WebKit, o qual serviu como base para que em 2003 a Apple lançasse uma versão própria de um navegador internet chamado Safari. Este mesmo interpretador serviu como núcleo do navegador internet

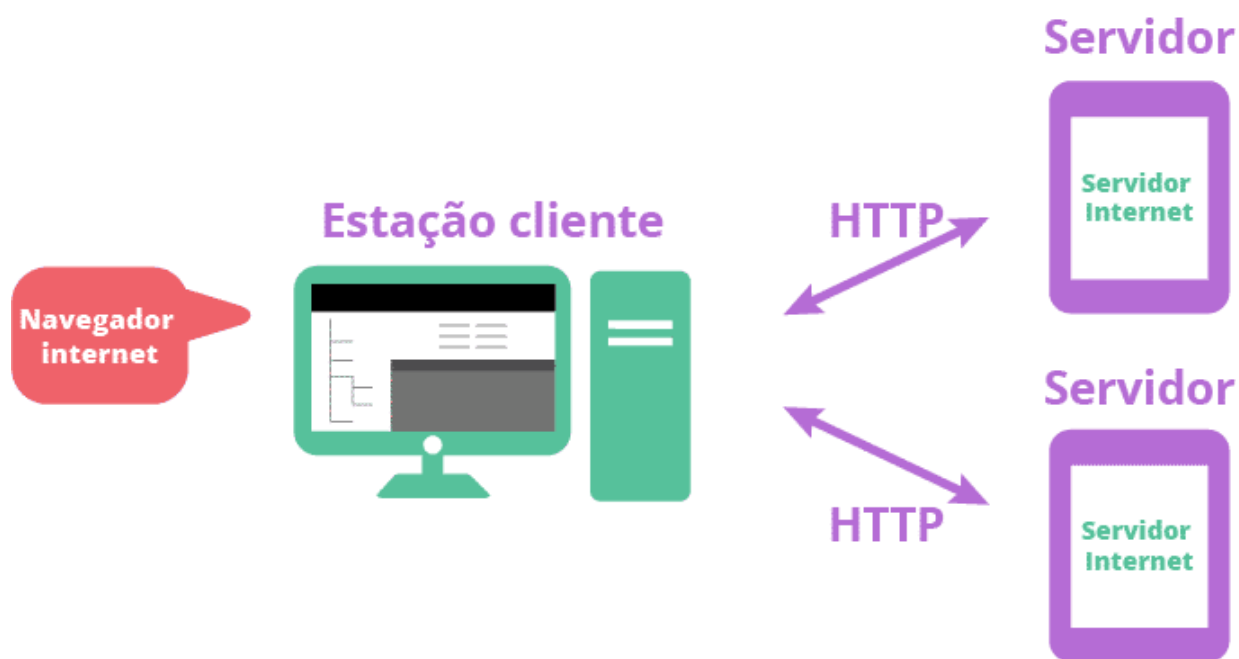
Chrome lançado pelo Google em 2008, e que a partir de 2013 teve seu interpretador de conteúdo substituído pelo Blink da própria Google.

Em 1994 fundou-se a World Wide Web Consortium (**W3C** – www.w3c.org), uma organização voltada para o desenvolvimento e padronização das tecnologias utilizadas nas páginas da internet e para o incentivo à interoperabilidade entre os seus servidores (MENDES, 2010).

O protocolo de aplicação utilizado para a transferência de páginas internet entre sistemas computacionais é o **HTTP** (*Hypertext Transfer Protocol*), definido na RFC 2616.

A aplicação HTTP é implementada através de dois programas: o **navegador internet** (cliente) e **servidor internet** (servidor), conforme representado na Figura 1. Esses programas conversam entre si através da troca de mensagens HTTP, passando requisições e respostas com páginas internet.

Figura 1 - Aplicação HTTP



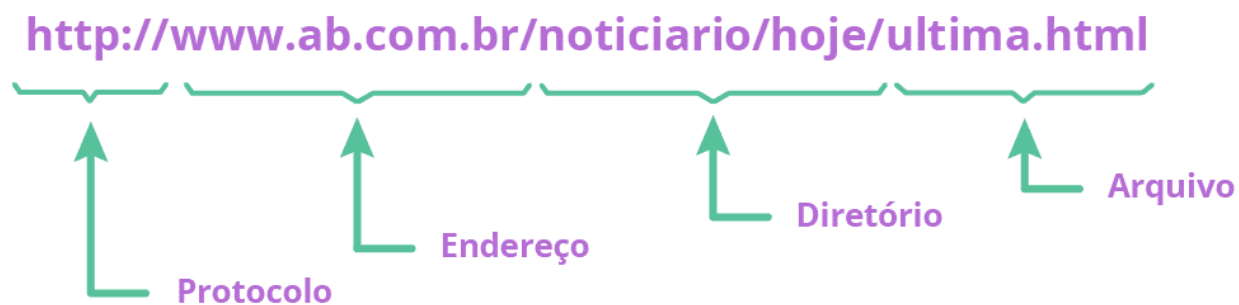
Fonte: Elaboração própria (2022).
Arte/Diagramação DME/FURB (2023).

Os arquivos de uma página internet são identificados desde o navegador internet através dos seus endereços completos dados por uma **URL** (*Uniform Resource Locator*).

Uma URL completa tem basicamente quatro componentes (Figura 2):

- protocolo a ser utilizado para obter o arquivo;
- endereço do equipamento servidor (que normalmente é dado na forma de nome de domínio e que deve ser traduzido por um DNS) que contém o arquivo a ser visualizado (endereço de rede e opcionalmente endereço de porta separado por ':');
- diretório dentro do servidor onde o arquivo está armazenado
- arquivo com o conteúdo requisitado.

Figura 2 - Constituição de Uma URL



Fonte: Elaboração própria (2022).
Arte/Diagramação DME/FURB (2023).

As páginas da internet são constituídas de objetos, os quais são, em sua forma mais simples, arquivos estáticos endereçados por **URLs**.

A maioria das páginas da internet é constituída por um arquivo **HTML** (*Hypertext Markup Language*) base e várias referências a objetos. Por exemplo, uma página internet composta por texto HTML, 1 *applet* Java e 3 figuras JPEG (*Joint Photographic Experts Group*), possui ao todo 5 objetos.

Os arquivos HTML são basicamente arquivos de texto ASCII. Além do texto simples, um arquivo HTML contém muitos marcadores os quais podem determinar, por exemplo, o estilo da fonte, o tipo do parágrafo, uma referência para outro arquivo HTML ou para uma imagem, diferentes níveis de cabeçalhos, a definição de uma tabela etc.

A maioria dos marcadores ocorre em pares de abertura e fechamento: normalmente o marcador de fechamento é igual ao de abertura, mas é precedido por uma barra (/), indicando onde o estilo ou conteúdo começa e onde termina.

Para os marcadores que não têm nenhum correspondente marcador de fechamento, uma barra (/) deve preceder o fechamento do marcador. Alguns marcadores têm parâmetros específicos dados na forma *nome="valor"* (KUROSE; ROSS, 2013).

Um documento HTML começa com o marcador `<html>` e é dividido em duas partes marcadas como `<head>` (cabeçalho da página) e `<body>` (corpo da página). Cada um desses três marcadores exige o terminador correspondente.

Um item importante do cabeçalho é o título, identificado pelo marcador `<title>`, que não aparece na página internet, mas que é utilizado para identificar a janela dos navegadores internet.

Alguns marcadores comuns utilizados no corpo de uma página internet com parâmetros básicos estão na Tabela 1.

Tabela 1 - Marcadores Comuns Utilizados em uma Página Internet

Marcador	Descrição
<code><hn> ... </hn></code>	Delimita um cabeçalho de nível n (de 1 a 6)
<code> ... </code>	Define um texto em negrito
<code><i> ... </i></code>	Define um texto em itálico
<code><center> ... </center></code>	Centraliza horizontalmente um objeto na página
<code><p> ... </p></code>	Define um parágrafo
<code>
</code>	Quebra uma linha
<code><hr/></code>	Quebra uma linha com uma linha horizontal
<code></code>	Exibe uma imagem referenciada por uma URL
<code> ... </code>	Associa um texto/objeto a um <i>hiperlink</i> referenciado por uma URL
<code> </code>	Delimita uma lista ordenada
<code> </code>	Delimita uma lista numerada
<code> ... </code>	Define um texto/objeto como um item de uma lista ordenada ou numerada

Marcador	Descrição
<code><table border="n"></code> <code></table></code>	Delimita uma tabela com borda de largura n
<code><col align="n"/></code>	Define as colunas de uma tabela e o alinhamento n a ser utilizado (left, center, right)
<code><tr> ... </tr></code>	Delimita a lista de células que compõe uma linha de uma tabela
<code><th> ... </th></code>	Delimita um texto/objeto de uma célula de cabeçalho de uma tabela
<code><td> ... </td></code>	Delimita um texto/objeto de uma célula de uma tabela

Fonte: elaborado pelo autor (2022).

Pratique

No seu navegador internet, abra uma página de um site e, utilizando o botão direito do mouse sobre ela, peça para exibir o seu código fonte.

Para permitir que houvesse uma comunicação também entre o usuário da internet e os provedores de informações, criou-se a funcionalidade de formulários que podem conter caixas de texto ou botões que permitem fornecer dados e retorná-los para o proprietário da página.

Os formulários são delimitados pelo marcador `<form>`. Todos os demais marcadores comuns são permitidos no corpo de um formulário.

Os campos onde o usuário fornece informações são identificados com o marcador `<input>` que possui uma variedade de parâmetros opcionais para determinar o tamanho, a natureza e o uso da caixa de entrada exibida. Os principais parâmetros do marcador `<input>` estão na Tabela 2.

Tabela 2 - Parâmetros do Marcador `<input>`

Parâmetro	Descrição
<code>name="..."</code>	Define o nome da variável que deve ser associado ao valor fornecido pelo usuário
<code>size="n"</code>	Define o tamanho do campo
<code>type="text"</code>	Define que o campo é de texto de uma linha
<code>type="password"</code>	Define que o campo é de texto de uma linha onde os caracteres digitados não devem ser exibidos
<code>type="textarea"</code>	Define que o campo é de texto de mais de uma linha
<code>type="radio"</code>	Define um conjunto de opções com o mesmo nome de variável onde apenas uma opção pode ser selecionada
<code>type="checkbox"</code>	Define uma opção que pode ou não ser selecionada
<code>type="submit"</code>	Faz com que as informações do usuário sejam enviadas de volta ao servidor que forneceu o formulário
<code>value="..."</code>	Define um valor padrão que deve ser retornado por aquele campo

Fonte: elaborado pelo autor (2022).

A linguagem HTML, com ou sem formulários, mistura o conteúdo com a formatação. A fim de permitir que as páginas internet sejam estruturadas para facilitar o processamento automatizado do conteúdo da internet, a W3C desenvolveu duas novas linguagens: a primeira, chamada **XML** (*Extensible Markup Language*), descreve o conteúdo (dados) de uma página internet de uma forma estruturada, e a segunda, denominada **XSL** (*Extensible Style Language*), descreve a formatação de modo independente do conteúdo.

O XML pode ser usado para outros fins além de descrever o conteúdo de páginas internet. Um uso crescente do XML é como uma linguagem para comunicação entre programas aplicativos. Em particular, o **SOAP** (*Simple Object Access Protocol*) é um modo de executar funções entre aplicações de forma independente da linguagem e do sistema. O cliente elabora a solicitação como uma mensagem XML e a envia ao servidor, usando o protocolo SOAP. O servidor devolve uma resposta como uma mensagem XML formatada. Desse modo, aplicações em plataformas heterogêneas podem se comunicar

2

Páginas Internet Dinâmicas Geradas pelo Cliente

Uma página da internet é basicamente um arquivo HTML interpretado pelo navegador internet (programa cliente) com objetivo de disponibilizar informações e eventualmente permitir a interação com o usuário.

Outras tecnologias podem ser misturadas ao HTML para a construção de interfaces ainda mais amigáveis, com um visual mais adequado para situações específicas e capazes de proporcionar mais recursos que o HTML isoladamente não é capaz de suprir.

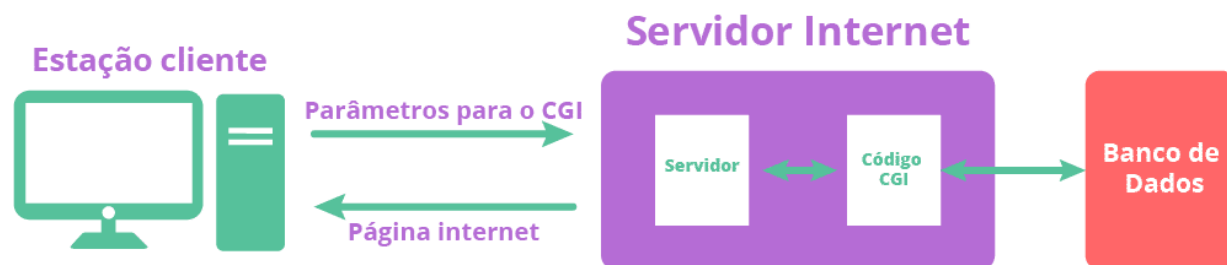
Para este propósito, foi incluída no HTML a possibilidade de incorporar códigos nas páginas internet que são executados na máquina cliente. Esses códigos são identificados nas páginas através do marcador `<script>`.

Uma linguagem importante para escrever estes códigos é chamada JavaScript. O JavaScript é uma linguagem de programação completa, com variáveis, cadeia de caracteres, listas, objetos, funções e todas as estruturas de controle habituais. O JavaScript também tem muitos recursos específicos para as páginas internet, inclusive a habilidade para gerenciar janelas e lidar com formulários. A sua sintaxe é baseada na linguagem de programação Java (YNEMINE, 2005).

Há um tipo mais específico de página internet onde o seu conteúdo depende de um processamento junto à aplicação servidora, normalmente havendo interação com recursos (um banco de dados, por exemplo) acessíveis somente do servidor. Para permitir este tipo de processamento e o envio da resposta para o programa cliente, desenvolveram-se algumas tecnologias que possibilitam esta forma de interação entre os programas cliente e servidor (KUROSE; ROSS, 2013).

A forma com que todas essas tecnologias trabalham é similar: o programa servidor recebe uma solicitação de processamento de um cliente, processa essa solicitação e responde quase sempre na forma de uma página HTML. Este mecanismo é chamado de **CGI** (*Common Gateway Interface*): um padrão para interação de aplicações cliente com servidores que pode ser escrito em diversas linguagens de programação, tanto interpretadas quanto compiladas, e que residem e são executadas na máquina servidora. Este mecanismo está representado na Figura 3.

Figura 3 - Mecanismo CGI



Fonte: Elaboração própria (2022).
Arte/Diagramação DME/FURB (2023).

Um tipo especial de código cujo funcionamento é similar ao do CGI chama-se **Servlet Java**. Um Servlet Java é um programa executado pelo servidor, escrito na linguagem Java, que apenas exporta a sua interface para o navegador internet. Desta forma, a interação de um Servlet Java com o usuário ocorre no navegador internet enquanto o seu processamento ocorre no servidor internet. Isto implica que um Servlet Java deixa de funcionar assim que a conexão do cliente com o servidor for desfeita.

Outra forma de gerar conteúdo dinâmico no lado servidor é incorporando códigos em páginas HTML e fazendo com que eles sejam interpretados e executados pelo próprio servidor para gerar a página internet (ROCHA, 2003).

Uma linguagem importante e não proprietária para escrever estes códigos é chamada **PHP** (*Hypertext Preprocessor*). O PHP é especialmente indicado para manipulação de formulários e é mais simples do que usar um código CGI. A sua sintaxe é baseada nas linguagens de programação C, Java e Pearl.

Outras duas linguagens utilizadas para gerar conteúdo dinâmico no lado servidor chamam-se **JSP** (*Java Server Pages*) e **ASP** (*Active Server Pages*). Ambas funcionam de forma semelhante ao PHP, porém o JSP utiliza a linguagem de programação Java na parte dinâmica da página HTML enquanto o ASP utiliza a linguagem de programação Visual Basic Script proprietária da Microsoft (SILVA, 2003).

A escolha entre as linguagens PHP, JSP e ASP normalmente está mais relacionada à estratégia do que à tecnologia, pois as três linguagens são equivalentes.

3


Comunicação entre Programa Cliente e Servidor

Quando um programa cliente, denominado navegador internet, pede uma página internet, ele envia ao servidor internet mensagens **HTTP** (*Hypertext Transfer Protocol*) de requisição para cada um dos objetos constituintes da página. O programa servidor recebe estas solicitações e responde com mensagens HTTP contendo os objetos. O HTTP está definido na RFC 2616.

Há dois tipos de conexões HTTP: **persistentes** e **não persistentes**. Quando o HTTP usa conexões não persistentes (é o caso da versão 1.0), para cada objeto constituinte da página, uma nova conexão é criada e encerrada ao concluir a transmissão. Quando o HTTP usa conexões persistentes (é o caso da versão 1.1), é criada uma conexão para cada grupo de objetos constituintes da página e presentes em um mesmo equipamento servidor.

No caso do HTTP versão 1.1, o navegador pede um objeto da página após o outro através da conexão, o que costuma tornar o processo lento para páginas com muitos objetos. Para aumentar o desempenho do HTTP criou-se a versão 2 que, além de possuir um cabeçalho binário comprimido, recebe pela conexão vários objetos simultaneamente.

Uma **mensagem HTTP de requisição** é composta por uma linha inicial (denominada linha de requisição), várias linhas adicionais contendo opções específicas para a requisição (denominadas linhas de cabeçalho) e uma linha final em branco. Seja a requisição HTTP a seguir:

 GET /algumdiretorio/subdir/pagina.html HTTP/1.1

Host: www.universidade.com.br

Neste exemplo, a linha de requisição indica que se deseja obter o arquivo pagina.html localizado no diretório *algumdiretorio/subdir* utilizando a versão 1.1 do protocolo HTTP. A linha adicional indica que o servidor requisitado é o *www.universidade.com.br*. Essa informação adicional é obrigatória, pois alguns endereços de servidores podem servir a vários nomes de domínio e o servidor precisa saber a qual deles a requisição está sendo feita.

O HTTP especifica sete tipos de mensagens de requisição a um servidor internet, os quais estão apresentados na Tabela 3.

Tabela 3 - Tipos de Mensagens de Requisição a um Servidor Internet

Comando	Significado
GET	Solicita um arquivo
HEAD	Solicita apenas o cabeçalho da mensagem de resposta sem o arquivo
PUT	Envia um arquivo
POST	Anexa dados a um arquivo no servidor
DELETE	Exclui um arquivo no servidor
TRACE	Envia de volta a requisição com propósito de depuração
OPTIONS	Permite consultar o servidor sobre suas propriedades ou sobre as de um arquivo

Fonte: Elaborado pelo autor (2022).

Uma mensagem HTTP de resposta é composta por uma linha inicial (denominada linha de status), várias linhas adicionais contendo opções específicas para a resposta (denominadas linhas de cabeçalho), uma linha em branco e o conteúdo do objeto propriamente dito. Veja a resposta HTTP a seguir:

```
HTTP/1.1 200 OK
Date: Thu, 18 Oct 2016 12:00:00 GMT
```

```
Server: Apache/1.3.20 (Unix)
Last-Modified: Mon, 15 Oct 2016 08:00:00 GMT
Content-Length: 1508
Content-Type: text/html
```

(...)

Neste exemplo, a linha de status indica que o servidor está utilizando a versão 1.1 do protocolo HTTP e que a requisição está sendo respondida com sucesso. As linhas de cabeçalho contêm a data em que a requisição está sendo respondida, o tipo de servidor que está respondendo, a data da última atualização do arquivo, o comprimento do dado (objeto) que está sendo enviado e o seu tipo para que possa ser devidamente tratado.

Com o objetivo de tornar a navegação internet mais segura, reduzindo o risco de roubo de dados, criou-se o **HTTPS** (*Hypertext Transfer Protocol Secure*), definido na RFC 2818, para garantir a origem das informações através da inclusão do suporte ao uso de certificados digitais para implementar a autenticação do servidor de origem e o sigilo dos dados transferidos pela rede internet.

Com o propósito de melhorar o desempenho da internet como um todo, criou-se uma técnica chamada armazenamento de respostas em *cache* que visa guardar o resultado das requisições para uso subsequente. Esse procedimento é executado em um processo chamado *proxy*.

Para usar o armazenamento em *cache*, um navegador internet pode ser configurado para fazer todas as solicitações de páginas internet a um *proxy* em vez de solicitá-las ao servidor real das páginas. Se o *proxy* tiver a página, ele a retornará imediatamente; caso contrário, ele buscará a página no servidor, guardando uma cópia no cache para uso futuro e encaminhando-a para o cliente internet que a solicitou.

Como vimos, um navegador internet conecta-se a um servidor internet para obter o conteúdo de páginas internet através do protocolo HTTP cliente/servidor. Acontece que atualmente esse paradigma cliente/servidor não tem atendido satisfatoriamente aos conteúdos cada vez mais interativos entre o usuário e os servidores internet, e por isso criou-se uma solução *peer-to-peer* de troca de informações entre os navegadores internet e os servidores internet: o **WebSocket**.

O *cookie* (RFC 2109) é um mecanismo que alguns servidores internet utilizam para acompanhar os acessos sucessivos de um determinado usuário, implementando assim um conceito de sessão de usuário.

Quando um navegador internet acessa um servidor que usa cookies, este gera um número de identificação o qual é retornado e passa a ser armazenado localmente no equipamento cliente. Quando o equipamento cliente voltar a acessar este servidor, ele envia o número de identificação de forma que o servidor possa identificar quem está acessando e quais características e/ou informações são relevantes a este cliente.

É importante ressaltar que, quando bem utilizado, não há qualquer identificação do usuário armazenada no *cookie*, mas apenas uma identificação que permite que o servidor consulte a sua base de dados para obter as informações de todos os acessos que este usuário fez nos acessos anteriores desde um mesmo equipamento cliente.

Por outro lado, um uso controvertido dos *cookies* tem a finalidade de reunir secretamente informações sobre a navegação dos usuários, permitindo assim que se elabore um perfil completo dos seus hábitos na internet.

CONTINUE

Atividade de Passagem

(ENADE) Um navegador Web executa em um hospedeiro A, em uma rede de uma organização, e acessa uma página localizada de um servidor Web em um hospedeiro B, situado em outra rede na Internet. A rede em que A se situa conta com um servidor DNS local junto ao gateway. Um profissional deseja fazer uma lista com a sequência de protocolos empregados e comparar com o resultado apresentado por uma ferramenta de monitoramento executada no hospedeiro A. A lista assume que: (i) todas as tabelas com informações temporárias e caches estão vazias; (ii) o hospedeiro cliente está configurado com o endereço IP do servidor DNS local. Qual das sequências a seguir representa a ordem em que mensagens, segmentos e pacotes serão observados em um meio físico ao serem enviados pelo hospedeiro A?

- ☐ ARP, DNS/UDP/IP, TCP/IP e HTTP/TCP/IP
- ☐ ARP, DNS/UDP/IP, HTTP/TCP/IP e TCP/IP
- ☐ DNS/UDP/IP, ARP, HTTP/TCP/IP e TCP/IP

☐

DNS/UDP/IP, ARP, TCP/IP e HTTP/TCP/IP

☐

HTTP/TCP/IP, TCP/IP, DNS/UDP/IP e ARP

SUBMIT

(ENADE) Os protocolos da camada de aplicação que utilizam, na camada de transporte, o protocolo TCP, para o estabelecimento de conexões, fazem uso de portas específicas por padrão. Além disso, o funcionamento destes protocolos pode variar no estabelecimento e manutenção dessas conexões, bem como na troca de dados entre cliente e servidor. Considerando o funcionamento desses protocolos, analise as afirmações que se seguem.

I. O protocolo HTTP utiliza, por padrão, para conexão do cliente ao servidor, a porta 80/TCP. O estabelecimento de conexões HTTP tem início com a solicitação por parte do cliente (browser) ao servidor web. Após o estabelecimento da conexão, o socket permanece ativo até que o cliente finalize a conexão enviando um segmento TCP ao servidor com a flag FIN ativada.

II. O protocolo FTP utiliza, por padrão, para conexão do cliente ao servidor, a porta 21/TCP. Após o estabelecimento de conexões FTP, além da porta 21/TCP, utilizada para o controle da conexão, as portas TCP utilizadas para a

troca de dados entre cliente e servidor podem variar de acordo com o modo configurado no servidor (ativo ou passivo).

III. O protocolo SMTP utiliza, por padrão, para conexão do cliente ao servidor, a porta 25/TCP. O uso deste protocolo é parte do serviço de correio eletrônico, uma vez que é responsável pelo envio de e-mails. Para o acesso às caixas de mensagens e recebimento desses e-mails, utilizam-se os protocolos POP ou SSH, que usam, por padrão, respectivamente, as portas 110/TCP e 22/TCP.

IV. O protocolo DNS utiliza, por padrão, para conexão do cliente ao servidor, a porta 53/TCP. Através desta porta, o cliente, após estabelecida a conexão, pode fazer consultas a hosts definidos nos mapas de zona do servidor autoritativo. A consulta a nomes atribuídos aos hosts tem como respostas os endereços IP a eles atribuídos enquanto a consulta aos endereços IP (quando configurado o DNS reverso) resultam nos respectivos nomes.

É correto apenas o que se afirma em:

-
- ☐ I.
 - ☐ II.
 - ☐ I e III.
 - ☐ II e IV.
 - ☐ III e IV.

SUBMIT

(ENADE) Um protocolo define o formato e a ordem das mensagens trocadas entre duas ou mais entidades comunicantes, bem como as ações utilizadas na transmissão e/ou recebimento de uma mensagem ou evento. A internet utiliza a arquitetura de protocolos TCP/IP para propiciar a comunicação entre os computadores a ela conectados. O TCP/IP está organizado em cinco camadas: Aplicação, Transporte, Rede, Enlace e Físico. Cada camada possui um conjunto de protocolos que possibilitam a implementação de diversos serviços como sites de comércio eletrônico, internet banking, páginas web dentre outros. A oferta destes serviços é feita por computadores denominados servidores e acessados por computadores clientes. Com base nas informações apresentadas, considere o cenário de uma empresa que possui um site de comércio eletrônico para a venda de produtos. Quando um cliente acessar o site, utilizando um navegador web, a comunicação entre o navegador e o site utilizará protocolos das camadas de aplicação e transporte da arquitetura TCP/IP, dentre os quais se incluem:



HTTP, SSH e UDP.

- ☐ HTTP, SSL/TLS e TCP.
- ☐ HTTP, SSL/TLS e UDP.
- ☐ WWW, SSL/TLS e IP.
- ☐ WWW, SSH e TCP.

SUBMIT

CONTINUE

Redes Multimídia

Observe no seu dia a dia o quanto você tem consumido de conteúdo multimídia da internet. Pois atualmente tem havido um crescimento no desenvolvimento e implementação de aplicações que transmitem e recebem conteúdo de áudio e vídeo pela internet, tais como vídeo de entretenimento, telefonia IP, rádio pela internet, páginas internet com conteúdo multimídia, teleconferência, jogos interativos, mundos virtuais, educação à distância, entre outros.

Ao contrário das aplicações tradicionais da internet (correio eletrônico, transferência de arquivos etc.), aplicações multimídia são altamente sensíveis ao atraso fim-a-fim e à variação deste atraso (*jitter*), mas, por outro lado, toleram perdas ocasionais de dados.

A solução para a transmissão de multimídia pela internet reside em classificar todo o seu tráfego, definir prioridades para diferentes aplicações e reservar banda de transmissão (KUROSE; ROSS, 2013).

O IETF desenvolveu um modelo chamado *Integrated Services* (**IntServ**) que inclui a transmissão em tempo real à arquitetura internet. O serviço de tempo real habilita as redes IP a prover qualidade de serviço (QoS - *Quality of Service*) às aplicações multimídia. A principal característica do IntServ é a capacidade de reservar recursos através da rede implementando um mecanismo de controle de admissão de aplicações multimídia e de reserva de recursos ao longo da rede.

Esta reserva de recursos pode ser feita através de um protocolo de sinalização específico denominado **RSVP** (*Resource Reservation Protocol*), proposto pelo IETF, que deve estar implementado em cada um dos routers da rede.

Em virtude das dificuldades encontradas em se implementar a arquitetura IntServ/RSVP, outro mecanismo para prover QoS foi proposto pelo IETF: é o *Differentiated Services* (**DiffServ**). O objetivo do

DiffServ é oferecer um serviço escalável discriminatório sem a necessidade de se implementar um estado de reserva de recursos por fluxo de dados e uma sinalização a cada router da rede.

O DiffServ faz uso do campo **ToS** (*Type of Service*) do cabeçalho IP, o qual permite a escolha de alguns serviços como minimizar custos e/ou maximizar fluxo, formando o início de um QoS para certos tipos de dados. Cada pacote recebe um encaminhamento particular com base na sua marcação no campo ToS.

O pacote que receber determinada marcação sempre terá o serviço provido por essa marcação. Visto que não há mais a existência de um estado de reserva de recursos por sinalização RSVP, cada *router* na rede só conhece o valor da marcação do pacote a ele encaminhado. Desta forma, cada aplicação de rede pode criar diferentes tipos de serviços de transmissão de dados apenas marcando o campo ToS de forma apropriada.

1 Aplicações Multimídia

As aplicações multimídia podem ser divididas em três classes: fluxo contínuo de áudio e vídeo armazenado, fluxo contínuo de áudio e vídeo em tempo real e fluxo contínuo de áudio e vídeo interativo em tempo real.

Na classe das aplicações de **fluxo contínuo de áudio e vídeo armazenado**, as aplicações cliente pedem por arquivos de áudio ou vídeo que estão armazenados em aplicações servidoras. Este conteúdo multimídia foi previamente armazenado no servidor e pode ser apresentado instantes após estar sendo recebido pela aplicação cliente. Como resultado, o usuário desta classe de aplicações pode pausar, retroceder, avançar ou navegar dentro do conteúdo multimídia.

O tempo decorrido entre a solicitação de uma dessas ações pelo cliente e a execução da ação deve ser da ordem de até 5 segundos para que a resposta seja considerada aceitável (KUROSE; ROSS, 2013).

A classe das aplicações de **fluxo contínuo de áudio e vídeo em tempo real** é equivalente à transmissão de rádio e televisão, porém utilizando a internet como meio de difusão. Estas aplicações

permitem que um usuário receba o conteúdo da programação de rádio ou de televisão ao vivo de qualquer lugar do mundo. Como este fluxo contínuo não é armazenado, um cliente não pode pausar ou avançar o programa que está recebendo.

O tempo decorrido entre a solicitação do fluxo contínuo até o início da sua reprodução deve ser da ordem de até 10 segundos para que a resposta seja considerada tolerável.

A transmissão de um fluxo contínuo de áudio e vídeo em tempo real para vários receptores pode ser feita de forma muito eficiente através da funcionalidade de multidifusão da internet.

A classe das aplicações de **fluxo contínuo de áudio e vídeo interativo em tempo real** permite que as pessoas usem áudio ou vídeo para se comunicarem em tempo real. Áudio interativo em tempo real é denominado **Telefonia IP**, pois do ponto de vista do usuário, este tipo de aplicação é similar ao sistema de telefonia tradicional. Vídeo interativo em tempo real equivale a sistemas de **Videoconferência** e **Videofonia IP**.

Para uma conversação com interação entre vários usuários, o atraso entre o momento em que um usuário fala ou se move e o momento em que a ação se manifesta nos hospedeiros receptores deve ser menor do que meio segundo (0,5 segundos) para ser considerada aceitável.

Antes que o áudio ou o vídeo possa ser transmitido pela internet, eles precisam ser **digitalizados** e **comprimidos**.

A digitalização é necessária, pois as redes de computadores transmitem bits, portanto os sinais analógicos precisam ser representados por bits. Esta digitalização é feita a partir de amostragem do sinal a uma taxa constante e de posterior quantização em função do número de bits a ser utilizado no processo. Por exemplo, se cada amostra for representada por um byte, a amostra será quantizada em 256 níveis distintos.

A compressão é importante porque áudio e vídeo não comprimidos consomem uma quantidade enorme de memória e de largura de banda. Por exemplo, uma imagem de 1024 pontos por 1024 pontos, cada ponto codificado por 24 bits de cores, consome 3 Mbytes de memória. Se a imagem for comprimida a um fator de 12:1, a memória requerida passará a ser de apenas 256 Kbytes.

No caso específico de áudio, a técnica básica de digitalização é o **PCM** (*Pulse Code Modulation*). A codificação de voz normalmente usa o PCM, com taxa de 8000 amostras por segundo, utilizando quantização de 8 bits por amostra, obtendo uma taxa de 64 kbps. A codificação de áudio utilizada nos CDs usa taxa de 44100 amostras por segundo, com quantização de 16 bits por amostra, obtendo uma taxa de 705,6 kbps para áudio mono e 1,4112 Mbps para áudio estéreo.

Estas taxas de transferência são muito altas para serem utilizadas na internet, por isso são necessários procedimentos de compressão de forma que as taxas sejam reduzidas. Algumas dessas técnicas de compressão para voz são o **GSM** (13 kbps), o **G.729** (8 kbps) e o **G.723.3** (5,3 kbps a 6,4 kbps). A principal técnica de compressão para música estereó é o **MPEG-1** (*Moving Pictures Experts Group*), em cujo padrão se baseia o **MP3** (112 kbps a 128 kbps).

No caso de vídeo, a necessidade de compressão é ainda mais significativa, pois trata de uma sequência de imagens tipicamente capturadas a taxas de 24 ou 30 imagens por segundo. As técnicas de compressão mais difundidas são **MPEG-1** para vídeo CD (1,5 Mbps) e **MPEG-2** para vídeo DVD (3 Mbps a 6 Mbps).

2

Fluxo Contínuo de Áudio e de Vídeo

No fluxo contínuo de áudio e de vídeo, aplicações cliente requisitam áudio ou vídeo de um servidor de fluxo contínuo. Ao responder a esta requisição, o servidor segmenta o fluxo contínuo de áudio ou vídeo e envia esses segmentos para a aplicação cliente encapsulados no protocolo especial para tráfego de fluxo contínuo de mídia denominado **RTP** (*Real-time Transport Protocol*).

Alguns segundos após a aplicação cliente começar a receber o fluxo contínuo, o **tocador de mídia** inicia a sua reprodução. Ao receber esse fluxo contínuo de áudio e/ou de vídeo, ele executa várias funções importantes (KUROSE; ROSS, 2013):

(Clique nas abas para acessar os Conteúdos)

DESCOMPRESSÃO	REMOÇÃO DE JITTER	CORREÇÃO DE ERROS	INTERFACE GRÁFICA
---------------	-------------------	-------------------	-------------------

O fluxo contínuo de áudio e vídeo precisa ser descomprimido em tempo real durante a reprodução da mídia;

DESCOMPRESSÃO	REMOÇÃO DE JITTER	CORREÇÃO DE ERROS	INTERFACE GRÁFICA
---------------	-------------------	-------------------	-------------------

O *jitter* é uma variação do tempo de atraso entre a origem e o destino que acontece durante a transmissão de dados pela rede. Como o áudio ou vídeo precisa ser reproduzido a uma taxa constante, esta variação de atraso precisa ser removida e compensada, o que é feito através do armazenamento local de uma quantidade suficiente de pacotes antes de serem reproduzidos;

DESCOMPRESSÃO	REMOÇÃO DE JITTER	CORREÇÃO DE ERROS	INTERFACE GRÁFICA
---------------	-------------------	-------------------	-------------------

Como pacotes podem ser perdidos na rede, os tocadores de mídia precisam procurar contornar estas perdas através da transmissão de pacotes redundantes ou da interpolação de dados entre os pacotes recebidos, evitando assim que a qualidade da reprodução se torne inaceitável;

DESCOMPRESSÃO	REMOÇÃO DE JITTER	CORREÇÃO DE ERROS	INTERFACE GRÁFICA
---------------	-------------------	-------------------	-------------------

A apresentação do conteúdo do fluxo contínuo é feita através de uma interface gráfica que pode ou não estar incorporada a um ambiente de navegação da internet.

Os tocadores de mídia normalmente disponibilizam uma série de facilidades ao usuário permitindo que este pare, retroceda ou avance a reprodução da mídia. Para que esta interação com a aplicação do usuário seja possível, foi criado um protocolo auxiliar especial denominado **RTSP** (*Real-time Streaming Protocol*), especificado na RFC 2326.

O RTSP é um protocolo que permite ao tocador de mídia controlar a transmissão do fluxo contínuo de áudio e vídeo. As ações de controle implementadas pelo protocolo são pausar, reproduzir, gravar, reposicionar, avançar e retroceder.

Vídeo Complementar

O tráfego contínuo de áudio/vídeo chama-se *streaming*, mas hoje se fala muito sobre conteúdo de mídia *live* e *on demand*. Qual a diferença? Tecnicamente, o que se chama popularmente de *streaming* ou *live* é o fluxo contínuo de áudio/vídeo em tempo real (uma webradio ou uma webcam, por exemplo) e *on demand* é o fluxo contínuo de áudio/vídeo armazenado (o YouTube, por exemplo). Assista ao vídeo [TecMundoExplica: Streaming](#) para saber mais sobre a diferença.

2

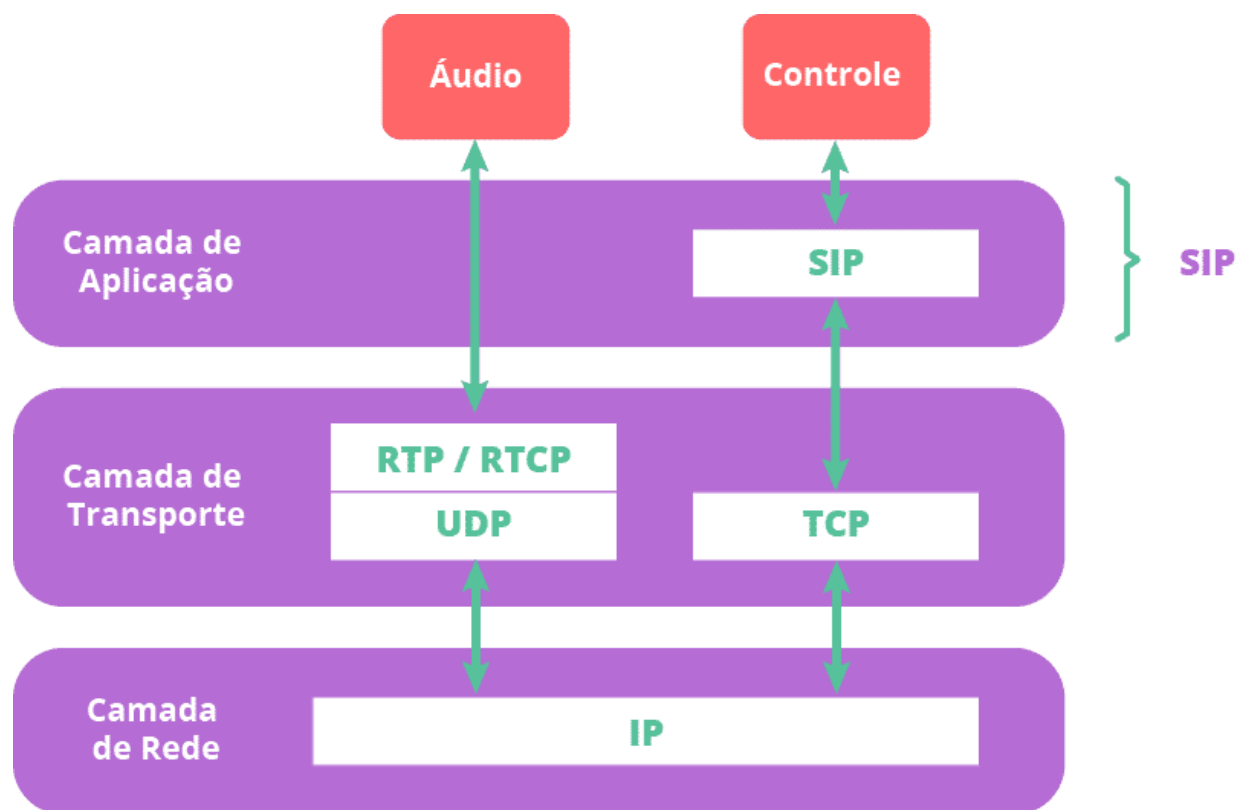
Fluxo Contínuo de Áudio e de Vídeo Interativo

A transmissão de fluxo contínuo de áudio e vídeo interativo em tempo real implica na definição de alguns protocolos de sinalização capazes de gerir a criação e o encerramento de chamadas, como, por exemplo, de telefonia IP, aplicação conhecida como VoIP, ou de videoconferência. Além disso, é preciso criar e manter diretórios de usuários e habilitar a capacidade de compatibilizar diferentes tipos de equipamentos envolvidos nessas chamadas (HERSENT; GUIDE; PETIT, 2002).

Com o propósito de suprir estas funcionalidades, o IETF desenvolveu uma arquitetura (Figura 4) especialmente para telefonia IP e audioconferência, que, através do protocolo **SIP** (*Session Initiation Protocol*), definido na RFC 3261, permite:

- prover mecanismos para estabelecer e encerrar chamadas entre interlocutores por uma rede TCP/IP;
- prover mecanismos para obter o endereço do equipamento onde se encontra um determinado destinatário;
- prover mecanismos para o gerenciamento de chamadas, tais como alterar a codificação, convidar outros participantes e ainda transferir e segurar chamadas.

Figura 4 - Arquitetura do SIP



Fonte: Elaboração própria (2022).
Arte/Diagramação DME/FURB (2023).

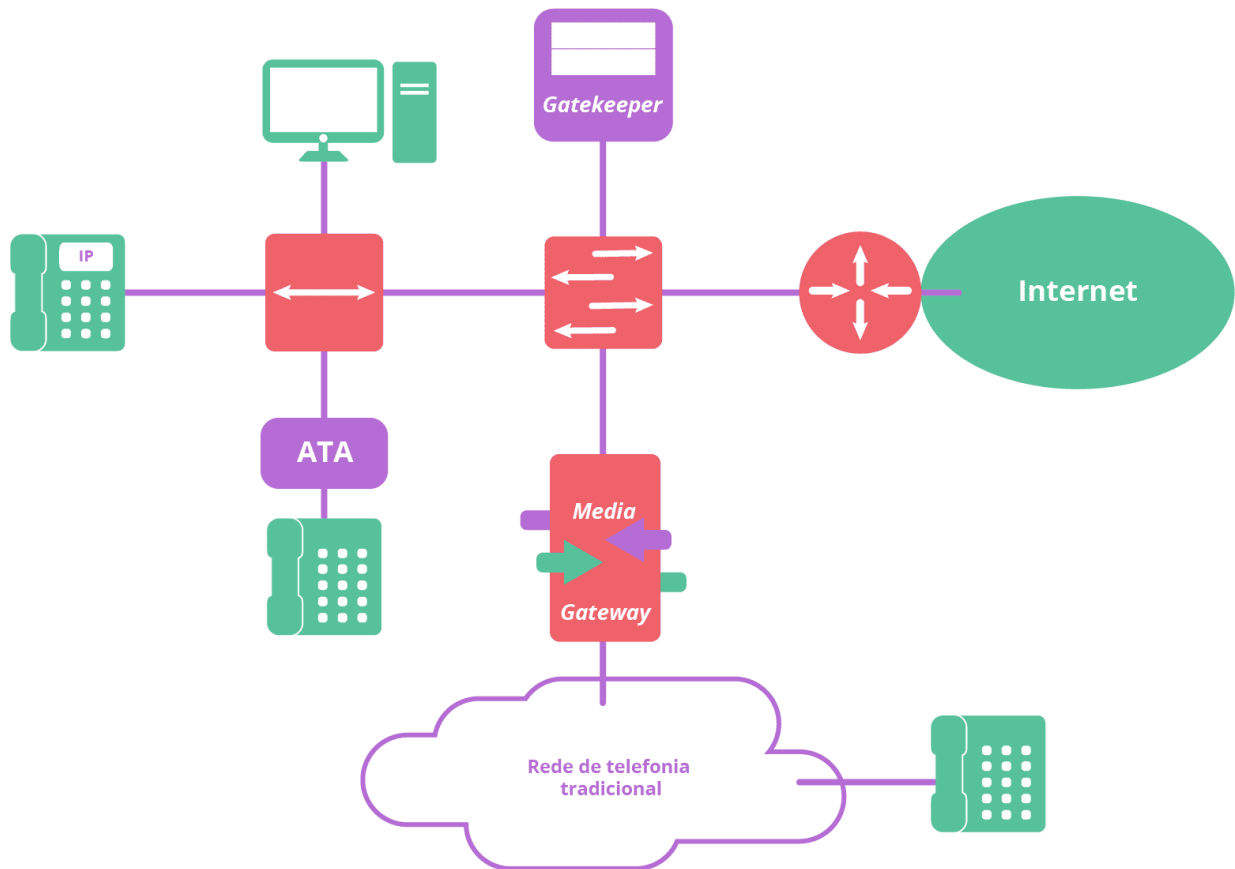
Equipamentos SIP também incluem *Media Gateways*, *Gatekeepers*, **Telefones IP** e **ATAs** (*Analog Telephone Adapter*) (Figura 5).

Os *media gateways* permitem a comunicação entre terminais SIP e aparelhos telefônicos tradicionais em uma rede de telefonia tradicional.

Os *gatekeepers* são equipamentos que disponibilizam mecanismos para tradução de endereços, autorização de acesso, gerência de caixas postais de voz, criação de grupos de audioconferência, gerência de largura de banda, contabilização dos envolvidos em uma ligação de telefonia IP, entre outras funções.

Os ATAs são adaptadores dedicados que permitem que telefones tradicionais se conectem diretamente em uma rede TCP/IP tornando-os telefones IP.

Figura 5 - Componentes do SIP



Fonte: Elaboração própria (2022)
Arte/Diagramação: DME/FURB (2023).

Quando um usuário pretende telefonar para um destinatário em uma rede TCP/IP utilizando o SIP, ele deve usar um aplicativo de telefonia (chamado *softphone*) ou um equipamento específico (chamado telefone IP). Ao iniciar este software/hardware, este automaticamente se cadastra no *gatekeeper* com a identificação do usuário e o seu respectivo endereço IP.

Para iniciar a chamada, o usuário envia uma mensagem de convite com a identificação do destinatário para o *gatekeeper*. Esta mensagem contém informações como a identificação do usuário, as características da chamada e os serviços que pretende utilizar. O SIP utiliza o padrão da internet (URL)

para identificar um cliente de telefonia IP, cujo formato é igual ao do seu endereço de correio eletrônico.

Durante o processo de estabelecimento de uma chamada, a mensagem SIP de convite contém os parâmetros **SDP** (*Session Description Protocol*), definidos na RFC 2237, com as capacidades do originador da chamada, fornecendo as informações necessárias ao estabelecimento de uma comunicação multimídia, como a identificação do iniciador da sessão, a banda passante disponível e os codificadores utilizados. Os parâmetros SDP da resposta SIP podem ser modificados trazendo as capacidades do destino que responde ao pedido.

Como o *gatekeeper* tem as informações de localização do destinatário se este estiver registrado (apto a receber ligações), ele encaminha a mensagem de convite para o destinatário. Se o destinatário aceitar a ligação, uma resposta de confirmação é enviada ao usuário e então uma sessão de comunicação é criada diretamente entre o usuário e o destinatário para a transmissão do fluxo contínuo de áudio através do protocolo RTP (*Real-time Transport Protocol*).

O SIP é um protocolo baseado em texto, semelhante ao HTTP, e tem apenas seis mensagens de controle para o estabelecimento e encerramento de chamadas, os quais estão apresentados na Tabela 4.

Tabela 4 - Tipos de Mensagens de Controle de Chamadas

Comando	Significado
INVITE	Solicita a inicialização de uma sessão
ACK	Confirma que uma sessão foi inicializada
BYE	Solicita o término de uma sessão

Comando	Significado
OPTIONS	Consulta um equipamento sobre seus recursos
CANCEL	Cancela uma solicitação pendente
REGISTER	Informa a um <i>gatekeeper</i> sobre a localização atual do usuário

Fonte: Elaboração própria (2022).

A sinalização entre terminais SIP e aparelhos telefônicos de uma rede de telefonia tradicional é feita através de um *media gateway* utilizando-se um protocolo especial, definido pelo IETF, denominado **MGCP** (*Media Gateway Control Protocol*), especificado na RFC 2705.

CONTINUE

Atividade de Passagem

(ENADE) Com a convergência das redes cada vez mais presente em nosso dia a dia, diversos canais de televisão vêm disponibilizando sua programação na internet. A grande vantagem dessa abordagem é a possibilidade de personalizar o acesso do espectador, permitindo que os programas sejam vistos sob demanda, ou seja, na hora e no local que seja mais conveniente ao cliente. Para a transmissão desse tipo de tráfego pela internet, as soluções de transporte devem ser desenvolvidas com base em:

- ☐ TCP, orientadas à conexão, pois a transmissão do vídeo deve ocorrer com a menor quantidade de erros possível, a fim de melhorar a experiência do usuário
- ☐ TCP, por meio de um serviço sem estabelecimento de conexão, pois as possíveis retransmissões a que um fluxo de dados orientado à conexão está submetido, sempre que há problemas na transmissão, podem causar atrasos incompatíveis com um serviço multimídia
- ☐ ICMP, devido à leveza desse protocolo e ao fato de ele ser o mais adequado para o transporte de conexões

multimídia, tais como a transmissão de programas de televisão pela internet

- ☐ UDP, orientadas à conexão, pois a transmissão de vídeo deve ocorrer com a menor quantidade de erros possível, a fim de melhorar a experiência do usuário
- ☐ UDP, por meio de um serviço sem estabelecimento de conexão, pois as possíveis retransmissões a que um fluxo de dados orientado à conexão está submetido, sempre que há problemas na transmissão, podem causar atrasos incompatíveis com um serviço multimídia

SUBMIT

CONTINUE

Resumo da Webaula 10

Nessa aula você viu as principais características em relação a mais importante aplicação da internet: a WWW – *World Wide Web*. Foram apresentados o histórico, as principais características e o protocolo de comunicação entre os navegadores e os servidores de conteúdo da internet.

Além disso foi apresentada a mais recente funcionalidade muito importante atualmente: o tráfego de conteúdo multimídia de áudio/vídeo em tempo real, suas características e protocolos, e também apresentei a estrutura de um sistema interativo de mídia sobre IP: a Telefonia IP.

CONTINUE

Conclusão da Unidade

Nessa terceira Unidade apresentamos as camadas de mais alto nível do modelo TCP/IP: camada de transporte e camada de aplicação.

Na camada de transporte vimos quais as principais funções que ela implementa e porque ela é considerada como uma camada complementar à camada de rede. Além disso apresentei como que um aplicativo de rede implementa uma transmissão de dados: através do uso do socket.

Na camada de aplicação – básica vimos quais os primeiros protocolos que foram desenvolvidos para implementar as funções mais básicas utilizadas até hoje nas redes de computadores.

Por fim, na camada de aplicação – suplementar apresentamos o mais importante protocolo de troca de informações pela internet: o HTTP. Além dele, vimos como a internet se adaptou para suportar o novo tipo de tráfego de dados, cada dia mais importante na internet: o conteúdo multimídia de áudio/vídeo em tempo real.

 YOUTUBE





Rede de Computadores - Vídeo Conclusão

Professor: Francisco Adell Péricas Gravação e Edição: Gustavo Bruch Féo

VISUALIZAR NO YOUTUBE >

CONTINUE

Referências

O estudo das camadas do modelo de referência TCP/IP abordadas nesta parte do livro pode ser encontrado em:

KUROSE, James F; ROSS, Keith W. **Redes de computadores e a internet**: uma abordagem top-down. 6 ed. São Paulo: Pearson Education do Brasil, 2013.

MENDES, Douglas R. **Redes de Computadores**: teoria e prática. São Paulo: Novatec, 2010.

Tópicos específicos sobre telefonia IP e sobre as técnicas de transmissão de multimídia sobre a internet são apresentados em:

HERSENT, Oliver; GUIDE, David; PETIT, Jean-Pierre. **Telefonia IP**: comunicação multimídia baseada em pacotes. São Paulo: Prentice Hall, 2002.

Tópicos específicos sobre linguagens para desenvolvimento de aplicativos de rede: SILVA, Osmar J. **JavaScript avançado**: animação, interatividade e desenvolvimento de aplicativos. São Paulo: Érica, 2003.

ROCHA, Cerli Antônio da. **Desenvolvendo Web Sites dinâmicos**: PHP, ASP, JSP. Rio de Janeiro: Campus, 2003.

YNEMINE, Silvana Tauhata. **Conhecendo o JavaScript**. 2 ed. Florianópolis: Visual Books, 2005.

As normas referenciadas nessa unidade podem ser obtidas diretamente da página da internet dos respectivos organismos de padronização:

- **IANA:** www.iana.org
- **ICANN:** www.icann.org
- **IETF:** www.rfc-editor.org/rfc-index.html
- **ISO:** www.iso.org/standards-catalogue/browse-by-ics.html
- **ITU-T:** www.itu.int/pub/T-REC
- **W3C:** www.w3.org/standards

CONTINUE

Créditos

Reitora

Profª. Ma. Marcia Cristina Sardá Espindola

Vice-Reitor

Prof. Dr. Marcus Vinicius Marques de Moraes

Pró-Reitor de Ensino de Graduação, Ensino Médio e Profissionalizante

Prof. Dr. Romeu Hausmann

Pró-Reitor de Administração

Prof. Me. Jamis Antônio Piazza

Pró-Reitora de Pesquisa, Pós-Graduação, Extensão e Cultura

Profª. Drª. Michele Debiasi Alberton

Divisão de Modalidades de Ensino Chefia da Divisão

Profª. Drª. Clarissa Josgrilberg Pereira

Professores Autores

Prof. Me. Francisco Adell Péricas

Design Instrucional

Profª. Drª. Clarissa Josgrilberg Pereira

Prof. Dr. Maiko Rafael Spiess

Prof. Me. Francisco Adell Péricas

Marcia Luci da Costa

Me. Wilson Guilherme Lobe Junior

Revisão Textual

Me. Wilson Guilherme Lobe Junior

Laura Cristina Zorzo

Roteirização

Laura Cristina Zorzo

Produção de Mídia

Gerson Luís de Souza

Gustavo Bruch Féo

Equipe de Design Gráfico

Amanda Kannenberg

Camylle Sophia Teske

Laura Cristina Zorzo

Nicolle Sassella

Renan Diogo Depiné Fiamoncini

Diagramado por Amanda Kannenberg em 13
de Fevereiro de 2023

CONTINUE