Universidade Federal do Piauí
Centro de Educação Aberta e a Distância

REDE DE COMPUTADORES:
OLHANDO PARA A AQUITETURA DA *INTERNET*

André Soares

Ministério da Educação - MEC
Universidade Aberta do Brasil - UAB
Universidade Federal do Piauí - UFPI
Centro de Educação Aberta e a Distância - CEAD

Rede de Computadores: Olhando para a Arquitetura da Internet

André Soares

UFPI
2012

PRESIDENTE DA REPÚBLICA	*Dilma Vana Rousseff Linhares*
MINISTÉRIO DA EDUCAÇÃO	*Fernando Haddad*
GOVERNADOR DO ESTADO	*Wilson Nunes Martins*
REITOR DA UNIVERSIDADE FEDERAL DO PIAUÍ	*Luiz de Sousa Santos Júnior*
SECRETÁRIO DE EDUCAÇÃO A DISTÂNCIA DO MEC	*Carlos Eduardo Bielshowsky*
PRESIDENTE DA CAPES	*Jorge Almeida Guimarães*
COORDENADOR GERAL DA UNIVERSIDADE ABERTA DO BRASIL	*João Carlos Teatini de S. Clímaco*
DIRETOR DO CENTRO DE EDUCAÇÃO ABERTA E A DISTÂNCIA DA UFPI	*Gildásio Guedes Fernandes*

COORDENADORES DE CURSOS

ADMINISTRAÇÃO	*Antonella Maria das Chagas Sousa*
CIÊNCIAS BIOLÓGICAS	*Maria da Conceição Prado de Oliveira*
FILOSOFIA	*Zoraida Maria Lopes Feitosa*
FÍSICA	*Miguel Arcanjo Costa*
MATEMÁTICA	*João Benício de Melo Neto*
PEDAGOGIA	*Vera Lúcia Costa Oliveira*
QUÍMICA	*Rosa Lina Gomes do Nascimento Pereira da Silva*
SISTEMAS DE INFORMAÇÃO	*Luiz Cláudio Demes da Mata Sousa*

EQUIPE DE DESENVOLVIMENTO

TÉCNICOS EM ASSUNTOS EDUCACIONAIS	*Ubirajara Santana Assunção*
	Zilda Vieira Chaves
COLABORADORA	*Cleidinalva Maria Barbosa Oliveira*
EDIÇÃO	*Roberto Denes Quaresma Rêgo*
PROJETO GRÁFICO	*Samuel Falcão Silva*
DIAGRAMAÇÃO	*Everton Oliveira de Araújo*
REVISÃO	*Lis Cardoso Marinho Medeiros*
REVISÃO GRÁFICA	*Maria da Penha Feitosa*

CONSELHO EDITORIAL DA EDUFPI

Prof. Dr. Ricardo Alaggio Ribeiro (Presidente)
Des. Tomaz Gomes Campelo
Prof. Dr. José Renato de Araújo Sousa
Profª. Drª. Teresinha de Jesus Mesquita Queiroz
Profª. Francisca Maria Soares Mendes
Profª. Iracildes Maria de Moura Fé Lima
Prof. Dr. João Renór Ferreira de Carvalho

© 2012. Universidade Federal do Piauí - UFPI. Todos os direitos reservados.

A responsabilidade pelo conteúdo e imagens desta obra é do autor. O conteúdo desta obra foi licenciado temporária e gratuitamente para utilização no âmbito do Sistema Universidade Aberta do Brasil, através da UFPI. O leitor se compromete a utilizar o conteúdo desta obra para aprendizado pessoal, sendo que a reprodução e distribuição ficarão limitadas ao âmbito interno dos cursos. A citação desta obra em trabalhos acadêmicos e/ou profissionais poderá ser feita com indicação da fonte.
A cópia desta obra sem autorização expressa ou com intuito de lucro constitui crime contra a propriedade intelectual, com sansões previstas no Código Penal.
É proibida a venda ou distribuição deste material.

Apresentação

Bem-vindos à primeira edição do livro Rede de Computadores: Olhando para a Arquitetura da Internet. A elaboração deste livro foi motivada pelo desafio de desenvolver um material voltado para a disciplina de Redes de Computadores do curso de Sistemas de Informação da Universidade Aberta do Piauí - UAPI. Este projeto surgiu a partir do convite do Prof. Carlos André Batista de Carvalho, coordenador do referido curso. É fundamental relatar que este livro tem fortes influências do livro do Prof. Kurose, chamado *Computer Networking: a Top-Down Approach.*

As redes de computadores atualmente constituem uma infraestrutura de comunicação indispensável. Um exemplo oportuno disso é o uso das rede de computadores para auxiliar o ensino a distância. As redes de computadores estão presentes em todos os setores e áreas do conhecimento. Qualquer pequena empresa (farmácia, padaria, restaurante), por menor que seja, faz uso de uma modesta rede para interconectar setor de pagamento (caixas) com o setor de vendas e o setor de estoque.

As redes de computadores também estão presentes na área da saúde em aplicações menos complexas (realização de laudos de exames e divulgação de seus resultados, ambos à distância) como em aplicações sofisticadas como por exemplo, a telecirurgia. Em 2001, foi realizada a primeira cirurgia a distância para retirada da vesícula biliar. O paciente estava em Strasbourg, na França e o cirurgião em Nova York, Estados Unidos. O Brasil atualmente está desenvolvendo uma rede de telemedicina chamada RUTE (http://rute.rnp.br). Seu objetivo é apoiar o aprimoramento da infraestrutura para telemedicina já existente em hospitais universitários, bem como promover a integração

de projetos entre as instituições participantes.

Além da área de saúde, as redes de computadores são utilizadas em outros setores. Na indústria, para interconexão de sensores e robôs, no campo para monitoramento de animais e plantações.

Na saúde tem-se criado uma rede de assessoramento para diagnóstico, tratamento, educação e outros mais, chamada Telessaude Brasil, Rede conectada à RUTE. Esta rede constitui-se de núcleos centrais que recebem todas as demandas e devolvem as soluções em curto intervalo de tempo, para melhorar a assistência ao usuário do Sistema Único de Saúde. O Piauí hoje faz parte desta rede e tem como coordenação estadual o Prof Gidásio Guedes Fernandes.

Indiscutivelmente, o melhor exemplo de rede de computadores é a Internet, presente em boa parte das escolas, universidades, hospitais, shoppings etc. Por isso, este livro utiliza os protocolos da arquitetura da Internet para apresentar ao leitor os principais conceitos e características das redes de computadores.

Bom Estudo !
André Soares

Sumário

UNIDADE 01

Introdução à Rede de Computadores

Resumindo

Este capítulo apresenta um panorama geral sobre redes de computadores. Isso é feito descrevendo importantes características da arquitetura de rede da internet e apresentando conceitos fundamentais, como por exemplo:

- Host;
- Computador;
- Arquitetura de rede;
- Protocolo;
- Rede de acesso;
- Meios físicos;
- Largura de banda;
- Atraso de transmissão e propagação.

Além disso, este capítulo caracteriza sucintamente as principais funções de cada uma das cincos camadas da arquitetura de rede da Internet.

Ao final é apresentado um rápido histórico sobre a Internet em nível mundial e nacional.

1 INTRODUÇÃO À REDE DE COMPUTADORES

INTRODUÇÃO

Nos dias de hoje, os computadores estão presentes nas mais diferentes áreas do conhecimento: educação, saúde, engenharias, física, química etc. Certamente, essa massificação dos computadores, hoje presentes em residências, escritórios, *shoppings centers*, aviões, trens, está associada à popularização da Internet.

A Internet é popularmente conhecida como a rede mundial de computadores ou a rede de redes, dentre outras definições. De fato, a Internet é uma rede de computadores que é definida em torno de uma arquitetura de rede chamada arquitetura TCP/IP. Neste livro, toma-se como exemplo a Internet para apresentar características importantes e discutir conceitos, mecanismos e tecnologias de uma rede de computadores.

Atualmente, pode-se afirmar que a principal funcionalidade de uma rede de computadores é suportar aplicações distribuídas. Quando dois ou mais processos em cooperação oferecem um serviço, pode-se dizer que eles compõem uma aplicação distribuída. Por exemplo, a WEB é certamente uma das aplicações distribuídas mais conhecidas no mundo, viabilizada através da cooperação de dois processos, servidor WEB (servidor WEB Apache ou *Internet Information Server* da Microsoft) e o *browser* do usuário (por exemplo, *Internet Explore* ou *Firefox*).

Seguindo o objetivo de dar suporte a aplicações distribuídas, pode-se dizer que o computador que executa o processo (parte da aplicação distribuída) é um hospedeiro de uma aplicação distribuída. Dessa forma, os computadores que fazem parte da arquitetura de rede da Internet são chamados de **hospedeiro** (*host*, em inglês).

Outro potencial de uma rede de computadores é a sua capacidade de compartilhar recursos (impressoras, scanners, gravadores de DVD) e de replicar dados e serviços com o objetivo de ser tolerante a falhas.

Para que dois computadores se comuniquem é necessário que ambos implementem a mesma arquitetura de rede. Uma arquitetura de rede é tipicamente dividida em camadas em que são definidas funcionalidades específicas para cada uma delas. As camadas inferiores normalmente oferecem serviços para camadas superiores.

Essa abordagem orientada em camadas é justificada pela complexidade do sistema. Normalmente, quando o objetivo é resolver um problema complexo pode ser interessante dividi-lo em vários problemas menores (dividir para conquistar). Esta conduta também é usada para projetar uma infraestrutura de rede de computadores.

Um conceito importantíssimo no universo das redes de computadores é o de **protocolo**. Segundo o Prof. Jim Kurose, um protocolo define o formato e a ordem das mensagens trocadas entre duas ou mais entidades comunicantes, bem como as ações realizadas na transmissão e/ou no recebimento de uma mensagem ou outro evento.

Um protocolo é um conjunto de regras que são utilizadas para que duas máquinas (ou melhor, dois processos que executam em máquinas diferentes) possam se comunicar de forma precisa.

VISÃO GERAL DA ARQUITETURA TCP/IP

Normalmente, uma arquitetura de rede possui em cada uma de suas camadas um ou mais protocolos responsáveis por implementar os serviços. A arquitetura de rede da Internet é dividida em 5 camadas, conforme ilustra a Figura 1.

Figura 1: Arquitetura TCP/IP.

Por exemplo, na arquitetura da Internet a **camada de aplicação** é responsável por dar suporte às aplicações propriamente ditas. Um exemplo de protocolo da camada de aplicação é o *Hyper Text Transfer Protocol* (HTTP). Ele define um conjunto de mensagens que viabilizam a aplicação WEB.

Existem diversos outros protocolos da camada de aplicação para a arquitetura TCP/IP, por exemplo *File Transfer Protocol* (FTP), *Simple Mail Transfer Protocol* (SMTP) etc. O protocolo FTP é utilizado para transferência de arquivos. O protocolo SMTP é necessário para envio de e-mails.

Veja um exemplo em que se deixa de utilizar um protocolo da camada de aplicação para utilização de outro protocolo também da camada de aplicação. Considere que um determinado usuário da Internet fecha o seu browser WEB (por exemplo Internet Explorer) ao término de uma pesquisa no Google sobre determinado assunto. Em seguida, ele passa a transferir documentos PDF (resultado da sua pesquisa no Google) para um determinado computador remoto. Para realizar a transferência de arquivos, o usuário deve utilizar o protocolo FTP e um cliente FTP, que é um programa específico para fazer a transferência de arquivos.

Ao trocar a aplicação WEB pela aplicação FTP, deixa-se de utilizar o protocolo HTTP na camada de aplicação e o protocolo FTP passa a ser utilizado. Este exemplo ilustra uma outra vantagem da abordagem em camadas, uma flexibilidade em substituir protocolos mantendo a funcionalidade da camada. Além disso, essa troca de protocolos da camada de aplicação é transparente para as demais camadas da arquitetura de rede.

A camada de aplicação tem como objetivo dar suporte às aplicações distribuídas. Para isso, os protocolos da camada de aplicação definem como os processos devem trocar mensagens para viabilizar tais aplicações. Por exemplo, para dois chefes de estado (presidentes do Brasil e dos EUA) conversarem a respeito de um tema de interesse internacional é necessário definir qual será o idioma utilizado (português, inglês ou espanhol). Da mesma forma, para viabilizar uma aplicação distribuída, dois processos precisam utilizar o mesmo protocolo para o sucesso na comunicação. Pode-se dizer de forma simplista que o idioma utilizado na comunicação entre dois processo que utilizam a arquitetura TCP é um protocolo da camada de aplicação.

Como dito anteriormente, os processos que cooperam para viabilizar uma aplicação distribuída são hospedados nos *hosts*. Portanto, os *hosts* precisam implementar todas as camadas da arquitetura TCP/IP, inclusive a camada de aplicação. Note que sem a camada de aplicação os processo não

conseguem trocar mensagens.

A Figura 2 ilustra a viabilização de uma aplicação distribuída que utiliza a arquitetura de rede da Internet, onde os processos A e B trocam mensagens no nível da camada de aplicação.

Figura 2: Viabilização de uma aplicação distribuída sob a Arquitetura TCP/IP.

Observe que sob cada um dos processos existe uma pilha de protocolos da arquitetura da Internet com todas as cinco camadas. Vale destacar que tais processos são executados nos *hosts* (hospedeiros de aplicações distribuídas).

A **camada de transporte** implementa uma comunicação lógica entre processos que normalmente são executados em hospedeiros diferentes. Esta comunicação acontece com o envio e recebimento de segmentos no nível da camada de transporte. As mensagens da camada de aplicação são transportadas dentro dos segmentos da camada de transporte.

A **camada de rede** implementa uma comunicação lógica entre hosts. Esta camada tem o objetivo de fazer o endereçamento dos computadores que fazem parte da rede. É através do endereçamento da camada de rede que são encaminhadas informações para um host específico da rede. Além disso ela é responsável pelo roteamento dos dados desde o host de origem até o destino, passando pelos dispositivos de interconexão.

Os componentes da Internet podem ser divididos em componentes de borda ou de núcleo. Uma rede de computadores pode ser vista como um conjunto de dispositivos (por exemplo, computadores, celulares, *Personal Digital Assistent* - PDA) que se comunicam seguindo as regras de semântica e sintaxe definidas nos protocolos de comunicação.

Os computadores conectados à Internet são os hospedeiros (*hosts*)

ou sistemas finais. A Internet é uma infraestrutura de *hardware* e *software* que promove a cooperação de processos que são executados em *hosts* remotos. Os hospedeiros são interligados por dispositivos de interconexão conforme ilustra a Figura 3.

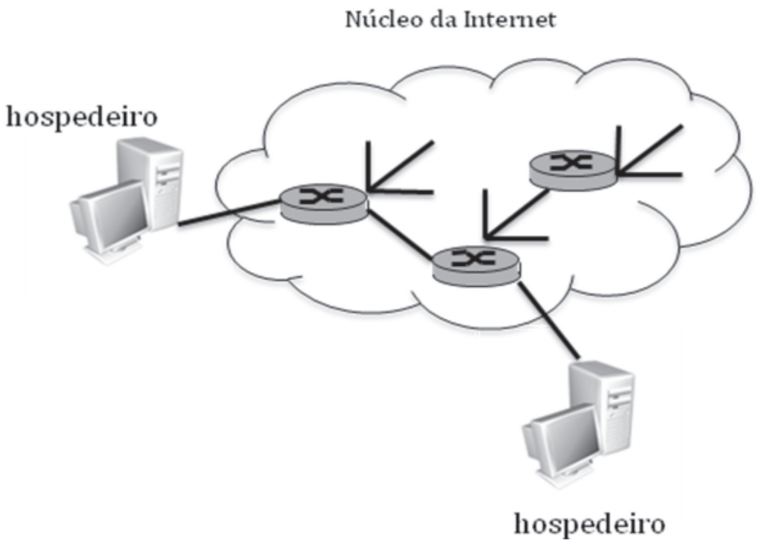


Figura 3: Componentes de borda e núcleo da Internet.

Os dispositivos de interconexão são os componentes do núcleo da Internet. Esses dispositivos são basicamente roteadores e comutadores.

A **camada de enlace** tem como objetivo criar um enlace de comunicação para que dispositivos adjacentes possam enviar dados. Por exemplo, na ilustração da Figura 1.2, cada *host* se liga fisicamente a um roteador através de um enlace. A camada de enlace da arquitetura TCP/IP tem como função transportar os pacotes IP entre as extremidades do enlace. O enlace pode ser entre hosts, entre roteadores ou entre um *host* e um roteador. A unidade de transporte da camada de enlace é o quadro. Portanto, os pacotes da camada de rede atravessam um enlace dentro dos quadros da camada de enlaces.

A função da **camada física** é transmitir *bits* entre elementos de rede adjacentes. Os quadros das camadas de enlace são representados através de *bits*. Os *bits* são representados através de uma codificação de sinais elétricos ou ópticos. Esses sinais são transmitidos por um dispositivo fonte e são recebidos e interpretados (decodificados) por um dispositivo de destino. O processo de decodificação é transformar os sinais que chegaram pelo meio

físico (por exemplo, cabos de cobre, fibra óptica etc) em *bits*. Dessa forma, o quadro transmitido pelo dispositivo fonte chega ao dispositivo de destino.

COMUTAÇÃO DE CIRCUITOS VERSUS COMUTAÇÃO DE PACOTES

No contexto das redes de computadores é fundamental a existência dos dispositivos de interconexão. Esses dispositivos estão presentes no núcleo da rede com a função de possibilitar um caminho ou rota por onde os dados seguirão de um *host* de origem até um outro *host* de destino. Os dispositivos de interconexão são comutadores e roteadores e possuem pelo menos duas interfaces de comunicação.

No exemplo da Figura 4 visualiza-se um roteador com quatro interfaces. O roteador é um dispositivo de interconexão de camada 3, isto é, a terceira camada de baixo para cima da arquitetura TCP/IP, a camada de rede.

Em termos gerais, a tarefa de comutação consiste em transferir os dados que chegam de uma interface para uma outra interface de saída. Por exemplo, os dados que chegam pela interface 1 podem ser comutados para as interfaces 2, 3 ou 4.

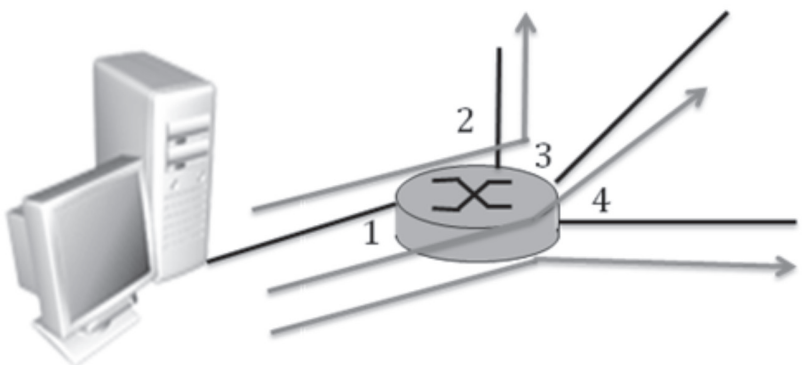


Figura 4: Função de comutação realizada pelos dispositivos de interconexão.

Basicamente existem dois paradigmas de comutação, a comutação de circuitos e a comutação de pacotes.

Na comutação de circuito, quando um circuito é solicitado, existe a necessidade de comunicação entre componentes da rede para saber se o

circuito requerido pode realmente ser atendido. Nesse processo é necessário saber se os comutadores do núcleo da rede possuem recursos disponíveis para viabilizar o circuito solicitado. Esse processo de verificação de disponibilidade e reserva de recursos é chamado de fase de **estabelecimento do circuito**.

Havendo a capacidade é feita a reserva desses recursos que ficam sob uso exclusivo daquele que solicitou o circuito até a sua finalização. Um circuito pode ser visto como um caminho fim a fim com recursos garantidos nos dispositivos de interconexão. Neste paradigma de comutação os dados fluem através do circuito estabelecido. Os dados somente podem ser enviados depois que o circuito está estabelecido.

Na comutação de pacotes não é feita reserva de recursos e portanto não existe a necessidade de um processo para estabelecimento da comunicação. Neste paradigma existe o conceito de pacote que é uma pequena unidade de transporte de dados. Ao invés de existir um circuito fim a fim por onde os dados são transmitidos, os dados são estruturados em pacotes que são comutados de forma independente. Assim, cada pacote precisa conter informações referentes ao seu *host* de destino. Tais informações são utilizadas no processo de comutação para saber qual interface de saída cada pacote deve ser encaminhado.

Diferentemente do paradigma de comutação de circuito, o encaminhamento dos dados nos dispositivos do núcleo da rede é feito sem nenhum acordo prévio. Como no paradigma de comutação de pacotes não existe uma sondagem em termos da capacidade dos comutadores, podem surgir congestionamentos nos dispositivos de interconexão. Isto é, podem haver situações onde a demanda de dados submetidos aos comutadores é maior do que sua capacidade de comutação. A consequência do congestionamento é a perda de dados. Entretanto, essa perda pode ser corrigida com a retransmissão dos dados perdidos.

Apesar da desvantagem associada à possibilidade de congestionamento nos comutadores das redes de comutação por pacote, existe uma vantagem da comutação de pacotes em relação a sua capacidade em absorver mais usuários. Por exemplo, no paradigma de comutação por circuito, se um circuito for estabelecido e os comutadores envolvidos passarem por um período de inatividade, os recursos da rede serão desperdiçados. Já no paradigma de comutação por pacote, o período de inatividade de um dado usuário da rede pode ser aproveitado para que outros usuários utilizem o recurso. Isso é possível porque na comutação de pacotes não

existe a reserva de recursos. Como quase tudo em computação, ambos os paradigmas possuem vantagens e desvantagens.

MEIOS DE TRANSMISSÃO E REDES DE ACESSO

Para que dois dispositivos com poder computacional (por exemplo, computadores, celulares, sensores sem fio etc) se comuniquem é necessário que ambos sigam a mesma arquitetura de rede. Normalmente, os computadores se interconectam utilizando cabos. É verdade também que mais recentemente as redes sofreram uma grande revolução e passaram a utilizar mais fortemente conexões sem fio (wireless).

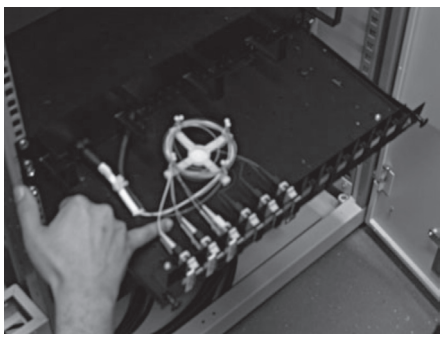


Figura 5. Exemplo de fibra óptica

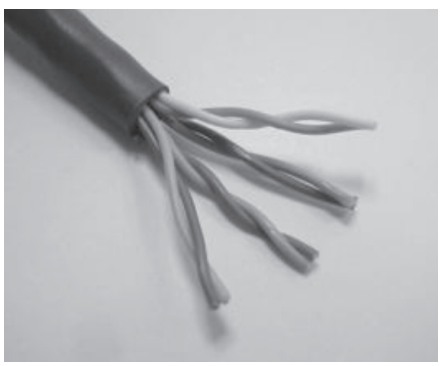


Figura 6. Exemplo de cabo de cobre

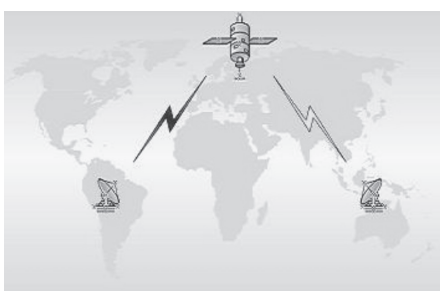


Figura 7. Exemplo de enlace de satélite

Para viabilizar uma comunicação através de um enlace, utiliza-se uma codificação de sinais (representando os bits zeros e uns) que são transmitidos de um computador fonte para um computador de destino. Esses sinais precisam ser transmitidos através de um meio físico e interpretados corretamente pelo receptor.

O meio físico pode ser classificado como guiado ou não guiado. Os meios guiados são meios sólidos que conduzem os sinais através de um cabo. Já nos meios físicos não guiados os sinais se propagam pela atmosfera. São exemplos de meios físicos:

- Fio de cobre (guiado)
- Fibra óptica (guiado)
- Atmosfera (não guiado)

As figuras 5, 6 e 7 ilustram, respectivamente, exemplos de utilização de fibra óptica, fio de cobre (par trançado) e enlaces de satélite que utilizam a atmosfera para propagar os sinais.

O tipo do meio físico influencia diretamente a tecnologia de enlace utilizada. Por exemplo, a velocidade de propagação dos sinais depende das propriedades físicas do meio. Cada meio físico tem uma velocidade de propagação específica.

Atraso de propagação é o tempo necessário para

um bit “viajar” de um extremo a outro do enlace.

A capacidade de transmitir informações (taxa de transmissão) através de um enlace é chamada de largura de banda. Ela indica a taxa de bits que uma dada tecnologia de enlace pode transmitir por segundo. Por exemplo, 10 Mbps (10 mega bits por segundo). Normalmente, a largura de banda é expressa em bits por segundo (bps, Kbps, Mbps, Gbps, Tbps etc).

Para recordar:

- Kb (Kilo bit) = 10^3 bits
- Mb (Mega bit) = 10^6 bits
- Gb (Giga bit) = 10^9 bits
- Tb (Tera bit) = 10^{12} bits

Vale destacar que o **b** (minúsculo) indica bit e **B** (maiúsculo) indica Byte.

A tecnologia da camada de enlace mais difundida atualmente é a Ethernet. Ela opera com diferentes taxas de transmissão 10/100/1000 Mbps. Portanto, considerando sua taxa mais modesta (10 Mbps), a tecnologia Ethernet precisa de 1 segundo para colocar 106 bits no enlace. Atraso de transmissão é o tempo que uma placa de rede gasta para transmitir 1 bit.

Transmitir é codificar os bits em sinais e colocá-los no enlace. Depois que os bits são transmitidos eles ainda precisam se propagar até a outra extremidade do enlace. O atraso de propagação é diferente de atraso de transmissão.

Conforme mencionado anteriormente a Internet é composta basicamente de *hosts* e roteadores. Para que um computador se torne um host e usufrua dos serviços da Internet ele precisa se conectar a um roteador de borda. Essa ligação é feita através de um provedor de acesso a Internet e é chamada de rede de acesso. Isso porque vai existir uma rede que vai interconectar um dado host a um roteador de borda da Internet.

As redes de acesso são implementadas em diferentes situações:

- Rede de acesso residencial
- Rede de acesso de um campus universitário
- Rede de acesso empresarial

O acesso residencial viabiliza a interconexão de computadores pessoais de residências à Internet. Esses acessos normalmente utilizam como meio físico a infraestrutura de cabos telefônicos já existentes. Neste

contexto existem basicamente duas tecnologias, **modem discado** e a **linha digital de assinante** (*Digital Subscriber Line* - DSL).

A Figura 8 ilustra a utilização da infraestrutura da rede de telefonia como rede de acesso, viabilizando a ligação de um host de uma residência com o roteador de borda do provedor de acesso a Internet.

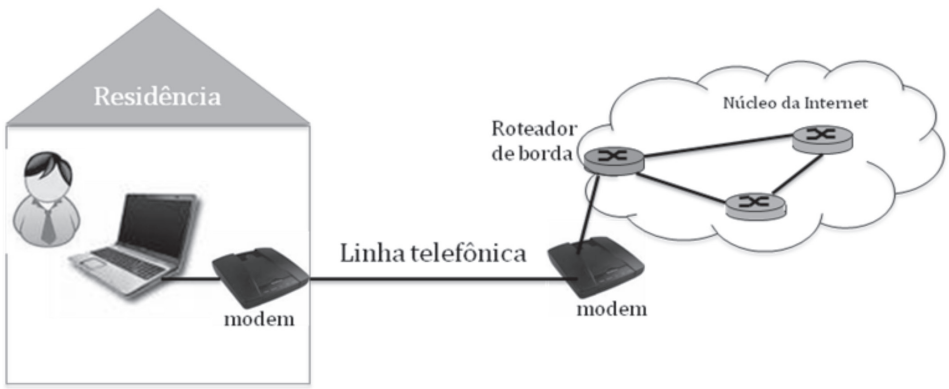


Figura 8. Exemplo de rede de acesso residencial utilizando a infraestrutura da rede de telefonia

O modem discado proporciona uma conexão de até 56 Kbps através de uma linha telefônica analógica. Nesta tecnologia não é possível navegar na Internet e falar ao telefone ao mesmo tempo.

O serviço de rede de acesso via DSL é normalmente oferecido pela operadora de telefonia. O modem utilizado na tecnologia DSL é similar ao modem discado. Uma limitação imposta em termos de distância entre os modens, pela tecnologia DSL, ocasiona a capacidade de operar com taxas de transmissão mais alta.

O serviço DSL é normalmente oferecido com assimetria (*Asymmetric Digital Subscriber Line* - ADSL) entre o *uplink* (envio de dados da residência para o provedor de Internet) e o *downlink* (envio de dados do provedor para a residência). Isso porque espera-se que um usuário residencial tenha um tráfego maior de descida (provedor para a residência) do que um tráfego de subida (residência para provedor). A tecnologia DSL permite que o usuário navegue na Internet e utilize o telefone ao mesmo tempo.

Outra tecnologia empregada para prover acesso residencial é o *Hybrid Fiber Coax* – HFC, popularmente chamada de cabo modem. A tecnologia HFC tem esse no me por fazer uso de uma infraestrutura híbrida de fibra óptica e cabo coaxial. Como nas redes de acesso mencionadas anteriormente, o HFC também utiliza uma infraestrutura já implantada, infraestrutura utilizada para

distribuição de sinal de TV a cabo.

Mais recentemente, com a chegada da terceira geração de celular, as operadoras de celular estão provendo redes de acesso através de suas infraestruturas. Um diferencial dessas redes de acesso é que o usuário tem acesso a Internet com um certo nível de mobilidade e em diferentes lugares de uma mesma cidade.

As redes de acesso de empresas e campus universitário normalmente fazem uso de uma tecnologia de rede local como Ethernet para ligar os hosts aos roteadores de borda. Mais informações sobre a tecnologia Ethernet serão apresentadas no Capítulo 5.

CLASSIFICAÇÃO DAS REDES SEGUNDO DIMENSÕES GEOGRÁFICAS

As redes de computadores podem ser classificadas de acordo com a sua extensão. Apesar de não ser uma classificação muito precisa essa classificação ainda é utilizada nos dias de hoje. As primeiras categorias criadas foram:

LAN – *Local Area Network*
MAN – *Metropolitan Area Network*
WAN – *Wide Area Network*

Uma LAN tem abrangência de um laboratório de computadores, de uma rede residencial ou de uma empresa de porte pequeno localizada em um ponto de uma cidade. É uma rede que possui um pequena abrangência geográfica.

Uma MAN é uma infraestrutura de rede que tem o alcance de uma cidade, uma metrópole. É uma rede de proporções geográficas mediana. Uma rede metropolitana interconecta normalmente diferentes redes locais de uma mesma cidade. Atualmente está em desenvolvimento um projeto chamado de redes Comep. Esse projeto é da Rede Nacional de Pesquisa – RNP e visa a implantação de redes metropolitanas em todas as capitais do Brasil.

Por usa vez, uma rede WAN tem o alcance de uma grande área geográfica interconectando redes MANs. Normalmente as redes WAN são de operadoras de telecomunicações. WAN cobrem tipicamente as redes referente a países e continentes. A RNP possui uma rede WAN que interliga todos os estados do Brasil, chamada de Rede Ipê. O objetivo da RNP é interconectar as redes Comeps através da rede Ipê.

Outras classificações que surgiram posteriormente foram:
PAN – *Personal Area Network*
WLAN – Wireless LAN

A classificação PAN é utilizada para fazer alusão a uma rede pessoal que interconecta dispositivos em torno de uma pessoa. Em termos gerais, esse tipo de rede faz uso de um enlace sem fio. São exemplos de redes PANs, conexão de um *headfone* ou um *headset* a um celular, um celular ou um mp3 player ao sistema de som do carro, câmera fotográfica a uma impressora etc. Para esse tipo de rede é muito utilizada a tecnologia de enlace sem fio Bluetooth.

A classificação WLAN é empregada para redes locais que são criadas utilizando uma rede sem fio como WiFi.

Outra característica importante de uma rede de computadores é a sua topologia. Topologia é a forma como os nós de uma rede estão interconectados. A topologia de uma rede pode ser vista como um grafo G(N,E), em que N é o conjunto de nós da rede e E é o conjunto de enlaces. Atualmente as redes oferecem suporte a serviços importantíssimo. Portanto, é fundamental projetar uma topologia de rede que tenha um alto nível de tolerância a falhas. Neste caso, as topologias em anel e em malha atendem esses requisitos.

HISTÓRICO DAS REDES DE COMPUTADORES

Para contar a história das redes de computadores vamos partir das tradicionais redes de telefonia, que no início da década de 1960 dominava o mundo das telecomunicações. Essa tecnologia, ainda presente nos dias de hoje na grande maioria das casas e escritórios, é caracterizada pelo estabelecimento de circuitos para suportar comunicação de voz.

Em meados da década de 1960 diferentes grupos de pesquisa começaram a estudar a comutação de pacotes, o paradigma de comutação utilizado atualmente na Internet. Entre os pesquisadores da época podemos citar Leonard Kleinrock, que neste período era aluno de doutorado do MIT interessado em comutação de pacotes. Kleinrock e seus colegas do MIT, Lawrence Roberts e Linklinder, lideraram um programa de Ciência de Computação na *Advanced Research Projects Agency* (ARPA) nos Estados Unidos. A partir deste programa surgiu a ARPAnet, a primeira rede de

computadores baseada em comutação por pacote.

Os primeiros comutadores de pacotes foram chamados de processadores de mensagens de interface (*Interface Message Processors* - IMPs). O primeiro IMP foi instalado na Universidade da Califórnia em Los Angeles (UCLA) sob a coordenação de Kleinrock. A Figura 9 mostra uma imagem do roteador utilizado na primeira conexão da Internet em outubro de 1969.

Em 1969 a ARPAnet tinha 4 nós, em 1972 passou a ter 15 nós e foi apresentada ao público na conferência International Conference on Computer Communications.

O primeiro protocolo fim-a-fim entre sistemas finais foi chamado de protocolo de controle de rede (*Network-Control Protocol* - NCP). Esse protocolo foi definido na [RFC 001] permitindo a partir daquele momento a escrita de aplicações.

Em 1972, Ray Tomlinson escreveu o primeiro programa de e-mail.

Os protocolos essenciais da Internet de hoje IP, TCP e UDP foram concebidos (ainda no papel) no final da década de 1970. A ARPAnet no início era uma rede de computadores isolada. Na primeira metade da década de 1970 surgiram outras redes de comutação de pacotes como ALOHAnet no Havaí e Cyclades na França. Com o surgimento dessas outras redes de comutação foi dado início ao trabalho de criar uma rede de redes com o patrocínio da DARPA (*Defense Advanced Research Projetcs Agency*) idealizado por Vinton Cerf e Robert Kahn. O termo utilizado para referenciar esse trabalho foi ***Internetting***. Pode-se dizer que a partir desse trabalho foi concebida a Internet.

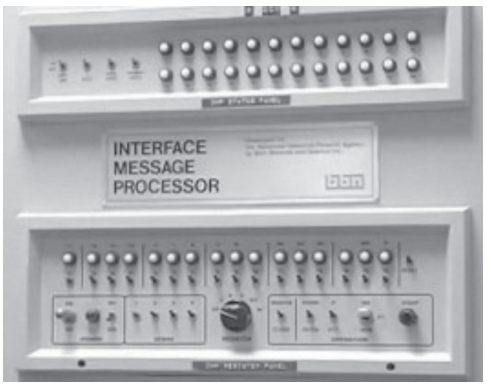


Figura 9: Imagem do roteador (IMP) utilizado na primeira conexão da Internet.

No Brasil, em 1988, surgiram iniciativas ligando centros de pesquisa do Rio de Janeiro, São Paulo e Porto Alegre a instituições nos Estados Unidos.

Em 1989 foi criada a Rede Nacional de Pesquisa – RNP com objetivo de unir tais redes embrionárias e formar um *backbone* (“espinha dorsal” da rede, principal infraestrutura) de alcance nacional.

Em 1991 foi inaugurado o primeiro *backbone* brasileiro de uso

exclusivamente acadêmico. Em seguida, em abril de 1995, o governo optou por abrir o *backbone*, dando conectividade a provedores de acesso comercial. Atualmente a RNP interconecta os 27 estados brasileiros através da Rede Ipê. A Figura 10 mostra a topologia da rede Ipê e as taxas de transmissão de seus enlaces.

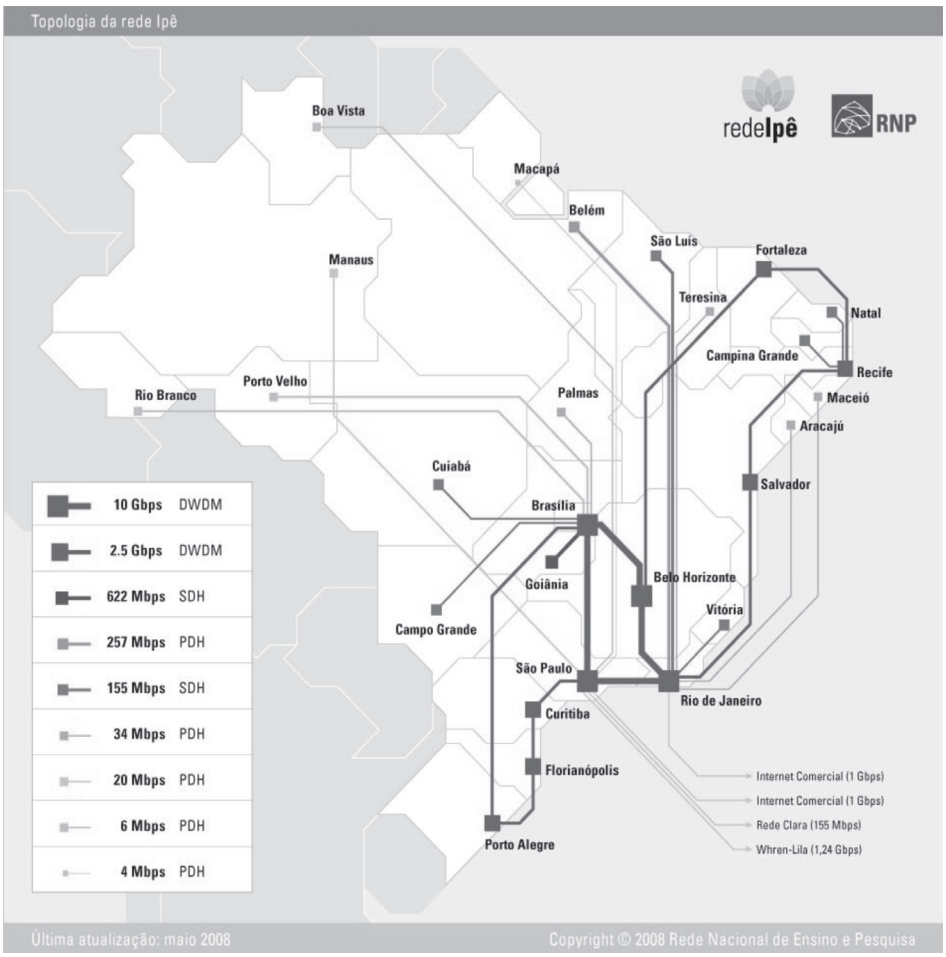


Figura 10: Topologia da rede Ipê.

EXERCÍCIOS

1) O que é um host?

2) Defina com suas palavras o que é um protocolo de comunicação.

3) Diferencie um roteador de um *host*.

4) Explique a divisão dos componentes da Internet em periferia e núcleo quanto à pilha de protocolos da arquitetura TCP/IP.

5) O que é uma rede de acesso? Cite e caracterize três tecnologias empregadas como rede de acesso.

6) Diferencie comutação de circuitos e comutação de pacotes. Discuta suas vantagens e desvantagens.

7) Explique a diferença entre TDM e FDM.

8) O que você entende por encaminhamento de pacotes feito pelos roteadores?

UNIDADE 02

Protocolos da Camada de Aplicação

Resumindo

Este capítulo está voltado para a camada de aplicação da arquitetura TCP/IP. A camada de aplicação oferece suporte direto às aplicações distribuídas. São listados e descritos os principais protocolos da camada de aplicação, são eles:

- HTTP;
- FTP;
- SMTP;
- POP3;
- IMAP e;
- DNS.

Tais protocolos são descritos com a ajuda de exemplos e explicações sobre cada passo no processo de comunicação através das mensagens da camada de aplicação.

2 PROTOCOLOS DA CAMADA DE APLICAÇÃO

INTRODUÇÃO

Um dos principais propósitos das redes de computadores são as aplicações distribuídas. Tomando a Internet como exemplo, é fácil identificar a sua relação com importantes aplicações distribuídas: WEB, e-mail, FTP, login remoto, dentre outras.

Sob o ponto de vista da arquitetura de rede da Internet (arquitetura TCP/IP), a camada de aplicação é responsável por dar suporte direto à aplicação distribuída. Ela define um conjunto de mensagens que serão trocadas entre os processos que cooperam para viabilizar a aplicação distribuída.

No contexto da Internet existem vários protocolos posicionados na camada de aplicação. Alguns dos mais conhecidos são: HTTP, FTP, SMTP, IMAP, POP3, DNS e DHCP.

A Quadro 1 relaciona cada um desses protocolos da camada de aplicação, suas aplicações e os protocolos da camada de transporte utilizados por eles.

A maioria desses protocolos de aplicação suporta aplicações que, em termos gerais, exigem um serviço com entrega confiável de dados. Por isso, dessa lista de protocolos da camada de aplicação, apenas os protocolos operam sobre o protocolo UDP da camada de transporte. Os protocolos da camada de transporte serão estudados no Capítulo 3. Até aqui é suficiente saber que o protocolo TCP implementa um serviço de entrega confiável de dados e o UDP não.

Como mostrado no quadro 1 os protocolos HTTP, FTP, DHCP, SMTP, POP3 e IMAP fazem uso do protocolo TCP da camada de transporte. As aplicações suportadas por esses protocolos requerem garantias em relação à entrega dos dados.

Já os protocolos UDP e SNMP optam pela simplicidade e agilidade de UDP em detrimento da garantia de entrega, que é provida apenas pelo TCP.

Quadro 1: Primeiro servidor WEB.

Camada de aplicação	Aplicação que o utiliza	Camada de transporte	Características
Hypertext Transfer Protocol (HTTP)	WEB	TCP	Utilizado para distribuição de conteúdo
File Transfer Protocol (FTP)	Transferência de arquivos	TCP	Transferência de arquivos entre hosts remotos
Domain Name Service - DNS	Traduação de nomes em endereços IPs	UDP	Necessário quando uma aplicação faz referência através de uma URL
Simple Mail Transfer Protocol (SMTP)	e-mail	TCP	Usado para o envio de e-mails
Post Office Protocol (POP3)	e-mail	TCP	Utilizado para baixar os emails em um agente de usuário
Internet Message Acess Protocol (IMAP)	e-mail	TCP	Utilizado para obter os emails em um agente de usuário
Dynamic Host Configuration Protocol DHCP	e-mail	TCP	Atribuição dinâmica de endereço IP
Simple Network Management Protocol (SNMP)	Gerenciamento da rede	UDP	Utilizado para fazer o gerenciamento dos recurso da rede

Note que algumas aplicações utilizam mais de um protocolo para serem viabilizadas. Por exemplo o e-mail.

O protocolo *Dynamic Host Configuration Protocol* - DHCP é utilizado para fazer atribuição dinâmica de endereço IP (endereço da camada de rede). Isso evita, por exemplo, que o endereço de rede de todas as máquinas de uma empresa seja configurada manualmente.

PROTOCOLO HTTP

Conforme já dito anteriormente, uma das aplicações mais conhecidas da Internet é a WEB. A ***World Wide Web*** foi proposta pelo físico inglês Tim Bernes-Lee no início da década de 90. Ele escreveu o primeiro cliente e servidor WEB. Sua invenção mudou a forma das pessoas viverem. Bernes-Lee não quis patentear sua invenção por considerar que ela foi uma obra social. A Figura 11 ilustra a máquina que executou o primeiro servidor WEB.

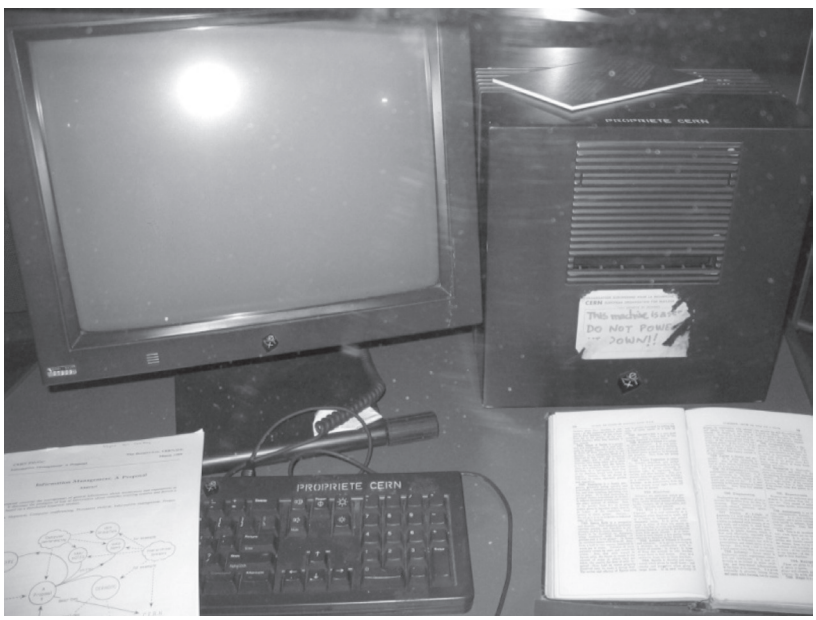


Figura 11: Primeiro servidor WEB.

A Figura 12 mostra a visualização da página WEB da UFPI através do browser WEB Safari, presente no sistema operacional MAC OS da Apple.

O protocolo da camada de aplicação que dá suporte direto à WEB é o *Hypertext Transfer Protocol* (HTTP) padronizado na RFC [2616]. Através de

mensagens de pedido HTTP um browser pode solicitar uma página WEB a um servidor WEB que responde utilizando uma mensagem de resposta HTTP.

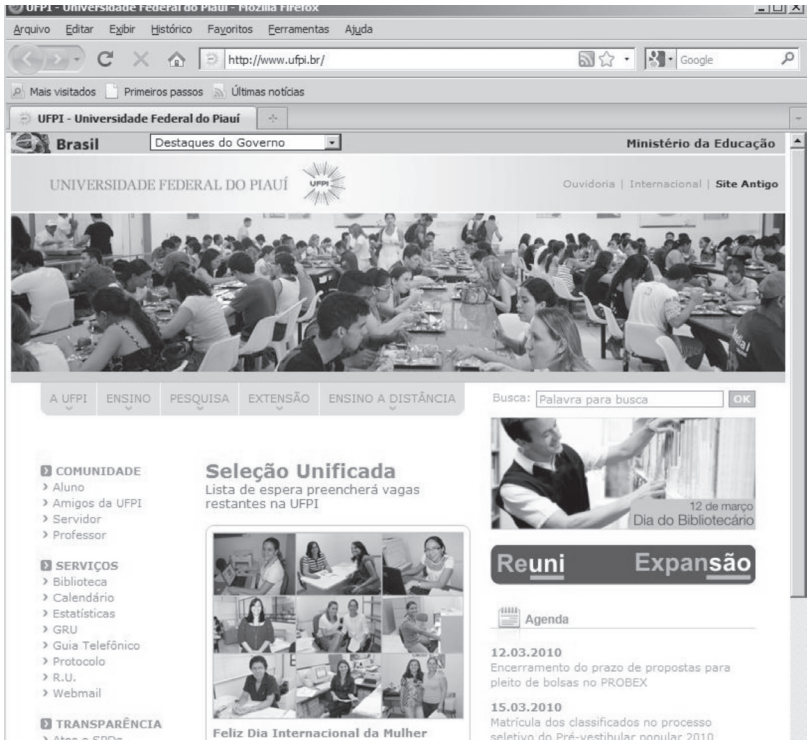


Figura 12: Exemplo de browser WEB.

Considere que um usuário digita uma *Uniform Resource Location* (URL) do site da UFPI (http://www.ufpi.br) no seu browser. No primeiro momento é requerida uma conexão TCP entre o browser WEB e o servidor WEB. Vale lembrar que esse processos em máquinas remotas interconectadas utilizando a infraestrutura da Internet.

Considerando que os computadores são multitarefas e executam mais de um processo “simultaneamente” usando artifícios de escalonamento de processador, como um sistema operacional sabe para qual processo devem ser encaminhados os dados que chegam pela rede? Essa pergunta será melhor respondida no Capítulo 3. Por enquanto, é suficiente saber que o sistema operacional utiliza como parte do endereçamento para identificar o processo um número chamado número de porta.

A conexão TCP para suportar mensagens HTTP é feita na porta 80 da máquina que hospeda o servidor WEB. O estabelecimento da conexão TCP

é um procedimento indispensável para a viabilização do serviço confiável e orientado à conexão da Internet. No Capítulo 3 descrever-se-á com mais detalhes o funcionamento do protocolo TCP.

Depois da viabilização da conexão TCP o Browser WEB envia uma mensagem solicitando a página que o usuário deseja. Ao receber a mensagem de pedido HTTP, o servidor WEB localiza a página no seu sistema de arquivo e encaminha a página solicitada encapsulada em uma mensagem de resposta HTTP. A Figura 13 ilustra este processo.

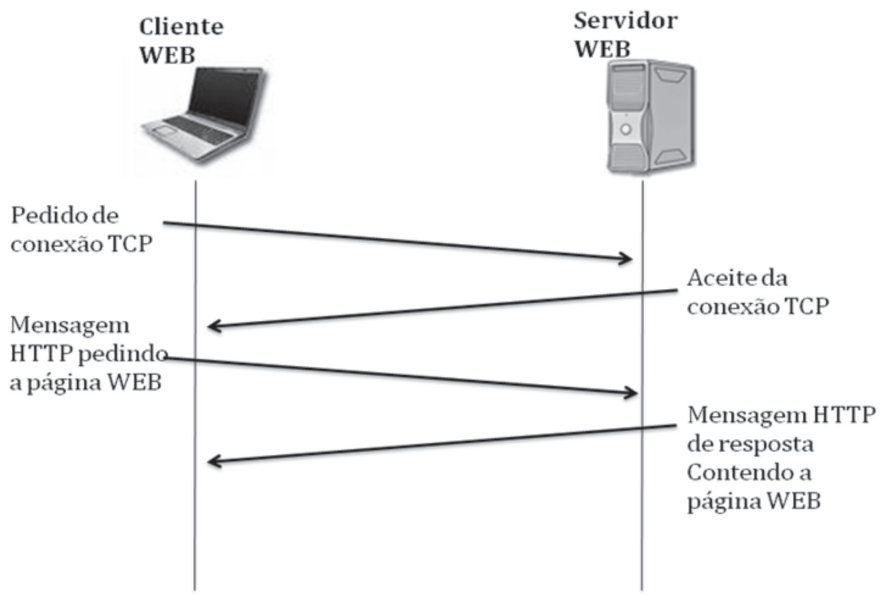


Figura 13: Troca de mensagens para solicitação de uma página WEB.

O protocolo HTTP possui basicamente as versões 1.0 e 1.1. A partir de 1998 o HTTP v1.1 passou a ser utilizado. No contexto WEB as informações são tipicamente disponibilizadas em página utilizando uma linguagem de marcação de hipertexto, *HyperText Markup Language* (HTML). As páginas HTML podem conter referências para outros arquivos como figuras, vídeos etc (todos chamados de objetos).

Ao digitar um URL no cliente WEB é enviada uma mensagem HTTP do cliente para o servidor WEB solicitando normalmente uma página HTML base. A Figura 14 ilustra uma página HTML visualizada pelo browser Internet Explore da Microsoft. Essa página HTML que faz referência a uma figura jpg e a um gif animado.

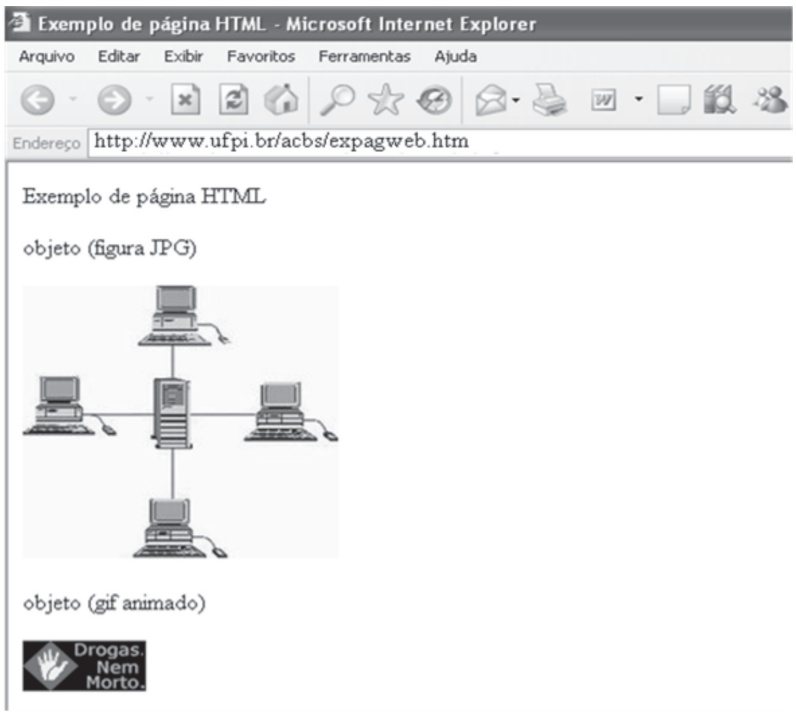


Figura 14: Exemplo de página HTML que faz referência para outros dois objetos.

O código HTML da página ilustrada na Figura 14 é apresentado na Figura 15.

Observe que a página HTML-base faz referência a outros objetos, o que requer o envio de outras duas mensagens HTTP de pedido e duas mensagens de resposta especificamente para viabilizar a transferência os outros dois objetos. No exemplo da Figura 2.5, os objetos são “obj1.jpg” “e “obj2.gif”. Suas referências no HTML base estão destacadas de vermelho.

Vale lembrar que o protocolo HTTP utiliza o protocolo TCP da camada de transporte que é orientado a conexão. Isso exige uma fase de estabelecimento da conexão, onde são trocadas pelo menos duas mensagens (uma do cliente para o servidor solicitando a abertura da conexão TCP e outra de resposta do servidor para o cliente aceitando o pedido de conexão) sem o envio efetivo dos dados úteis em termos de mensagem HTTP.

```
<!DOCTYPE HTML PUBLIC “-//W3C//DTD HTML 4.01 Transitional//EN”>
<html>
<head>
<title>Exemplo de p&aacute;gina HTML</title>
<meta http-equiv=”Content-Type” content=”text/html; charset=iso-8859-1”>
</head>

<body>
<p>Exemplo de p&aacute;gina HTML</p>
<p>objeto (figura JPG)</p>
<p><img src=”obj1.JPG” width=”210” height=”200”> </p>
<p>objeto (gif animado)</p>
<p><img src=”obj2.gif” width=”82” height=”37”> </p>
</body>
</html>
```

Figura 15: Código HTML da página WEB ilustra da Figura 14

Uma capacidade relevante do protocolo HTTP é a sua habilidade de ter uma conexão TCP persistente. No HTTP persistente vários objetos podem ser transmitidos através de uma única conexão TCP. No HTTP não persistente, apenas um objeto é transmitido por conexão TCP diferente. É fácil notar que o HTTP persistente é mais vantajoso, pois o custo para estabelecimento da conexão TCP é distribuído entre vários objetos transmitidos. O protocolo HTTP v1.1 tem como padrão a característica de persistência.

Outra característica relevante no protocolo HTTP 1.1 é a sua capacidade de solicitar vários objetos enviando mensagens assíncronas de pedido HTTP. Isto é, uma mensagem pode ser enviada logo após uma outra mensagem de pedido dentro de uma mesma conexão TCP, mesmo se a resposta da primeira solicitação ainda não tiver chegado. Essa capacidade é chamada de pipeline no jargão das redes de computadores. Esse termo faz alusão a um cano que é preenchido totalmente com a vazão da água. Na nossa analogia, as mensagens de pedido HTTP com flexibilidade de sincronismo em relação as suas respostas favorecem uma melhor utilização dos recursos da conexão TCP em curso.

As mensagens de pedido HTTP podem operar com os seguintes métodos: Get, Post, Head, Put e Delete. As mensagens Get são utilizadas para o pedido de objetos e admitem a passagem de parâmetro informados pelo usuários nos campos de formulário, conforme ilustrado na Figura 16. Note que o parâmetro passado do campo para pesquisa na página do Google é anexado na URL.

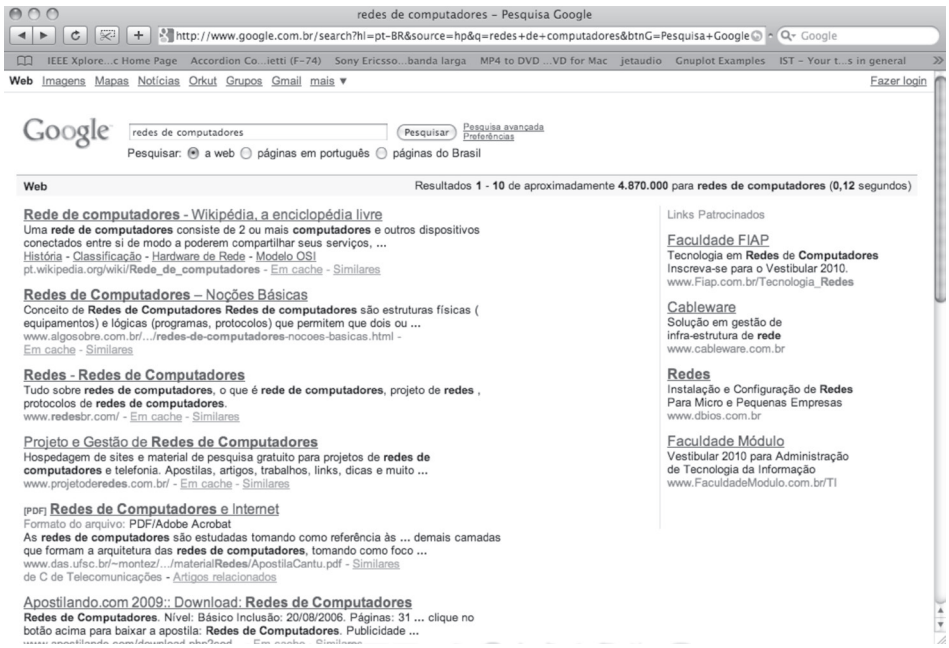


Figura 16: Exemplo de mensagem de pedido HTTP usando método Get.

No método **Post** os valores de campos de formulários são encapsulados na mensagem de pedido. O método **Head** é empregado como uma espécie de depurador para desenvolvedores de páginas WEB. Ao usar o método **Head** o servidor responde sem enviar o objeto solicitado na mensagem de pedido.

Os métodos **Put** e **Delete** foram inseridos no HTTP versão 1.1. A finalidade desses métodos é fazer upload de arquivos no servidor Web e permitir que um usuário apague arquivos no servidor WEB.

E-mail: protocolos SMTP, POP3 e IMAP

Uma das aplicações mais antigas na Internet é o e-mail. A data da primeira RFC é de 1982. O e-mail é uma aplicação assíncrona igual ao serviço de correio comum. Dessa forma, um usuário lê ou escreve seus

e-mail no momento que for conveniente para ele. Não é requerido que os dois interessados na comunicação estejam conectados ao mesmo tempo como ocorre por exemplo em ligações telefônicas.

A aplicação de e-mail é composta basicamente do agente de usuário, servidores de correio e dos protocolos da camada de aplicação responsáveis por enviar e baixar os e-mails. O agente de usuário é o aplicativo onde o usuário escreve, lê, armazena e organiza as suas mensagens. A Figura 17 ilustra o agente de usuário Eudora. Outros agentes de usuário são Outlook, Mail, Thunderbird.

Basicamente, o agente de usuário é subdividido em 3 janelas, a esquerda é visualizada as pasta do usuário onde o usuário organiza, classifica e armazenas seus e-mail. Existe também pastas específicas para guardar os e-mails já enviados. A janela direita superior mostra a lista de e-mails de uma das pastas selecionadas. A janela direita inferior permite a visualização de um e-mail propriamente dito. Além disso, o agente de usuário possui um editor de texto interno que conta com corretor ortográficos entre outros de recursos apropriados.

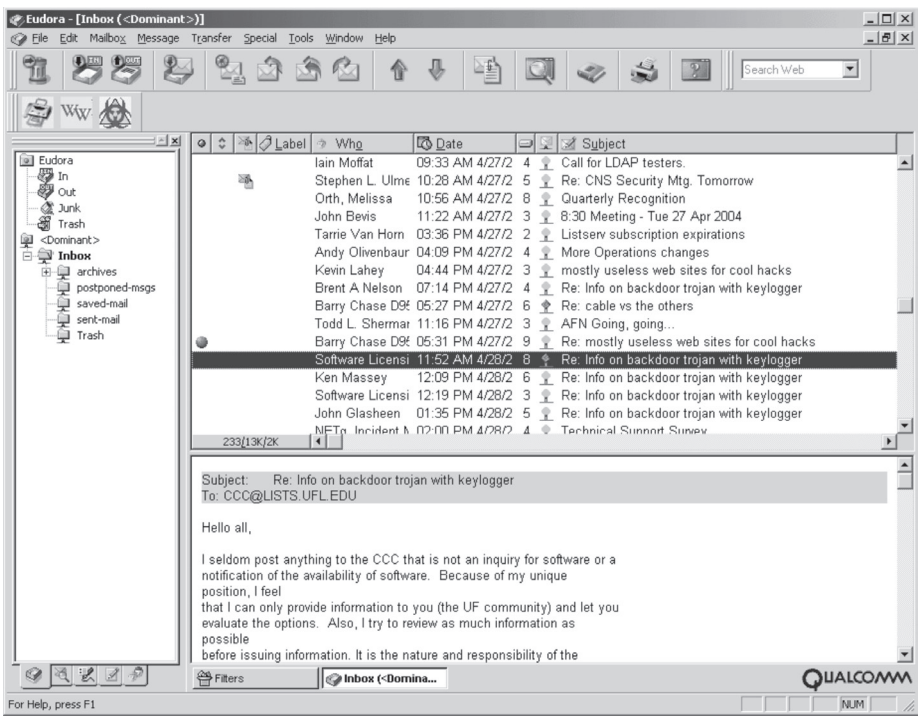


Figura 17: Visualização do agente de usuário Eudora.

Todo usuário de e-mail possui um servidor de correio onde fica sua caixa de correio. As mensagens enviadas para um dado usuário de e-mail são encaminhadas para a caixa de correio dele. Essas mensagens ficam na caixa de correio do usuário até que o mesmo leia e/ou apague as mensagens.

Há alguns anos a capacidade das caixas postais era muito modesta. Isso obrigava os usuários a ler e apagar suas mensagens com certa frequência. Caso contrário a caixa postal ficava cheia e novas mensagens não poderiam ser armazenadas. Isso também pode acontecer atualmente, entretanto, como a capacidade das caixas postais é alta essa ocorrência é menos provável.

Para apresentar todo o processo de envio e recebimento de e-mail vamos considerar dois usuário de e-mail, um remetente $\mathbf{U_r}$ e outro destinatário $\mathbf{U_d}$. Os servidores de correio de $\mathbf{U_r}$ e $\mathbf{U_d}$ serão representados respectivamente por $\mathbf{SC_r}$ e $\mathbf{SC_d}$. $\mathbf{AU_r}$ e $\mathbf{AU_d}$ são os agentes de usuário de $\mathbf{U_r}$ e $\mathbf{U_d}$, respectivamente.

O $\mathbf{U_r}$ redige um e-mail no seu agente de usuário ($\mathbf{AU_r}$) e em seguida pressiona o botão “enviar” para enviá-lo para o $\mathbf{U_d}$. Apesar do destino final ser o $\mathbf{U_d}$ o e-mail não é enviado diretamente para $\mathbf{U_d}$. Existem intermediários neste processo.

A Figura 18 mostra a caixa de composição de mensagem do agente de usuário chamado mail presente no sistema operacional da Apple.

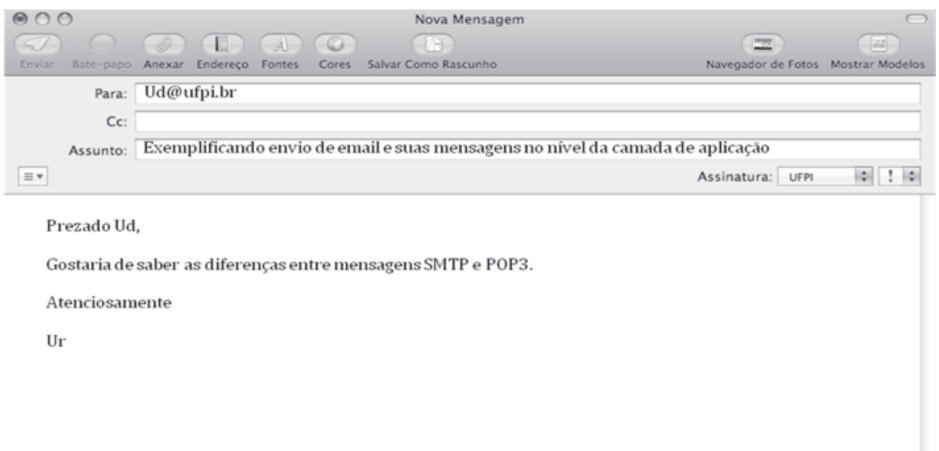


Figura 18: Visualização da tela de composição de e-mail do agende usuário mail presente nos sistemas operacionais da Apple.

Observe que na composição de um mensagem de e-mail o remetente precisa, obviamente, inserir o endereço de e-mail do destinatário. No nosso exemplo o endereço do destinatário é **Ud@ufpi.br.** O usuário remetente ($\mathbf{U_r}$) deve informar o assunto da mensagem e o texto da mensagem propriamente dita. Além disso, o remetente pode enviar um arquivo em anexo (neste caso ele pressiona o botão anexar para selecionar o arquivo a ser enviado, veja Figura 19) como também enviar uma cópia da mensagem para outros destinatários. Para isso o remetente deve inserir os endereços dos outros destinatário no campo Cc: logo abaixo o campo Para:

Feito isso, o agente de usuário do Ur envia o e-mail para o seu próprio servidor de correio ($\mathbf{SC_r}$) através de mensagens SMTP na porta 25. Os passos do envio de um e-mail do Ur para o Ud são ilustrados nas figuras 19, 20, 21 e 22.

Figura 19: Envio do e-mail do agente de usuário do remetente para o seu servidor de correio através de mensagens SMTP.

O $\mathbf{AU_r}$ solicita uma conexão TCP com o SCr na porta 25 e encaminha o e-mail através de mensagens SMTP. Considerando que o $\mathbf{SC_r}$ possui também outros usuários, é importante destacar que os e-mail de todos os usuário que chegam para serem encaminhados pelo $\mathbf{SC_r}$ são enfileirados. Seguindo a ordem da fila os servidor de correio encaminha tais e-mail para os seus respectivos destinos.

Vale ressaltar que essa troca de mensagem SMTP no primeiro momento é feita com o servidor de correio do próprio remetente do e-mail. Uma vez o e-mail estando no $\mathbf{SC_r}$ o e-mail será enviado para o $\mathbf{SC_d}$ também através de mensagens SMTP, conforme ilustra a Figura 20.

Figura 20: Envio do e-mail do servidor de correio do remetente para o servidor de correio do destinatário através de mensagens SMTP.

Note que o envio de e-mails conta com dois intermediários que são os $\mathbf{SC_r}$ e $\mathbf{SC_d}$. Tais intermediário são fundamentais para que o serviço de comunicação seja assíncrono. Esses intermediários são servidores que em tese ficam ligados 24 horas por dia. Assim, o usuário $\mathbf{U_r}$ pode desligar seu computador logo após transferir o e-mail para o seu servidor de correio. Os servidores de correio $\mathbf{SC_r}$ e $\mathbf{SC_r}$ são indispensáveis e precisam funcionar de forma ininterrupta para que as mensagens sejam enviadas e recebidas sem que os computadores do remetente e destinatário.

A Figura 21 descreve o último passo que é o recebimento do e-mail no agente de usuário do destinatário. Essa transferência é feita com o protocolo POP3 ou IMAP. Isto porque agora o e-mail não será enviado do SCd para o AUd, mas sim baixado do SCd para o AUd. Este procedimento é iniciado pelo agente de usuário (quando o Ud quiser checar seus e-mail) e não pelo servidor de correio.

Figura 21: Agente de usuário do destinatário baixando os e-mail da sua caixa postal através de mensagens POP3 ou IMAP.

Os protocolos POP3 [RFC 1939] e IMAP [RFC 2060] têm basicamente a mesma finalidade, permitir que um agente de usuário de um destinatário baixe suas mensagens do seu servidor de correios.

O protocolo POP3 pode operar em dois modos diferente, baixar e apagar ou baixar e manter as mensagens no servidor de correios. Já o protocolo IMAP é preparado para permitir que usuário crie pastas no seu servidor de correios contando com a funcionalidade de organizar as mensagens por pastas. Com isso as mensagens ficarão sempre organizadas independente de onde o usuário esteja acessando. Com o uso do protocolo POP3 o usuário somente pode organizar suas mensagens em pastas locais do seu agente de usuário. Se o usuário estiver em transito, utilizando um computador de um amigo, por exemplo, as mensagens não estarão organizadas. Por essas facilidade o protocolo IMAP acaba sendo mais complexo do que o POP3.

A Figura 22 apresenta todos os 3 passos no processo de envio e recebimento de e-mails.

Figura 22: Agente de usuário do destinatário baixando os e-mail da sua caixa postal através de mensagens POP3 ou IMAP.

Em 1990 o Hotmail propôs um agente de usuário baseado na WEB. Essa proposta foi muito aceita pelos usuários da Internet e é amplamente utilizada nos dias de hoje. Isso se deve a facilidade de ler e escrever e-mail em qualquer computador que tenha acesso a Internet.

Ao utilizar um WEBmail o protocolo SMTP é utilizando apenas na comunicação entre servidores de correio, ilustrado no passo 2 da Figura 23. Nos passos 1 e 3 (processo de envio e recebimento de e-mail) é utilizado o protocolo HTTP e não mais os protocolos SMTP, POP3 e IMAP

PROTOCOLO FTP

A aplicação FTP, que é baseada no protocolo FTP (File Transfer Protocol) da camada de aplicação, consiste em viabilizar a transferência de arquivos entre hosts remotos. Tal funcionalidade é fundamental para atividades como:

- A preparação e manutenção de web sites. Geralmente a pessoa que desenvolve o site WEB não tem acesso direto ao servidor WEB para transferir as páginas WEB desenvolvidas. Isso é feito normalmente com uso de uma aplicação FTP;
- Enviar e baixar arquivos com imagens, documentos, filmes e músicas para amigos;
- Compartilhar arquivos em geral com colaboradores ou então fazer cópias de segurança (backup) de arquivos locais ou remotos.

O protocolo FTP define um conjunto de mensagens para que servidor e cliente FTP, que executam em hosts remotos, possam enviar ou receber arquivos viabilizando um mecanismo de transferência de arquivos. Tipicamente um usuário faz uso de um cliente FTP que é um aplicativo para auxiliar o processo de transferência de arquivos.

A Figura 23 mostra um exemplo de cliente FTP (SmartFTP). A tela do cliente FTP é dividida em duas parte, à esquerda visualiza-se a árvore de diretórios da máquina local e do lado direito observa-se a árvore de diretórios da máquina remota.

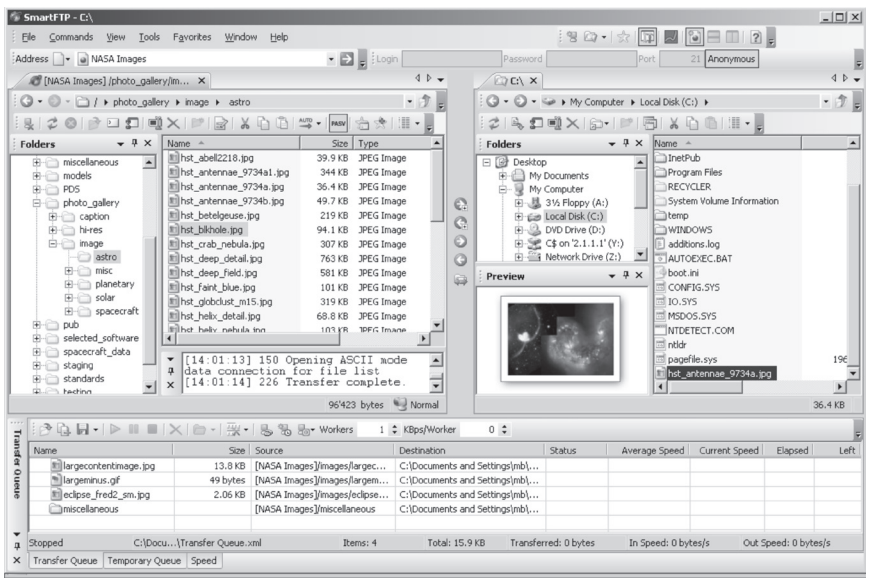


Figura 23: Visualização da tela do cliente de FTP (SmartFTP).

Ao transferir um documento de um diretório do lado direito para o esquerdo é feita uma transferência de um arquivo da máquina remota para a máquina local.

O protocolo FTP opera com duas conexões TCP distintas direcionadas para as portas 21 e 20 no servidor FTP. A conexão na porta 21 é utilizada para fazer sinalização. Por exemplo, quando um usuário estabelece uma comunicação via FTP ele precisa saber quais os arquivos existem em um dado diretório remoto. Para isso o cliente envia uma mensagem solicitando a lista dos arquivos que pertencem ao diretório em questão. Tal solicitação é feita com o envio de uma mensagem LIST. Para armazenar um arquivo no diretório remoto é utilizada uma mensagem STOR. Tais mensagens são geradas pelo cliente/servidor de e-mail e são especificadas dentro do próprio protocolo FTP. Essas mensagens são enviadas utilizando a conexão na porta 21.

Ao solicitar a transferência de arquivo via conexão na porta 21 é criada uma outra conexão na porta 20 exclusivamente para transferência do arquivo. Após a transferência do arquivo a conexão é finalizada. Por isso diz-se que o FTP é um protocolo que faz sinalização fora de banda, uma vez que os dados e as mensagens de controle não utilizam a mesma conexão TCP.

A Figura 24 ilustra o processo de transferência de arquivos entre sistemas de arquivos de hosts remotos viabilizado pelo protocolo TCP e a infraestrutura da Internet.

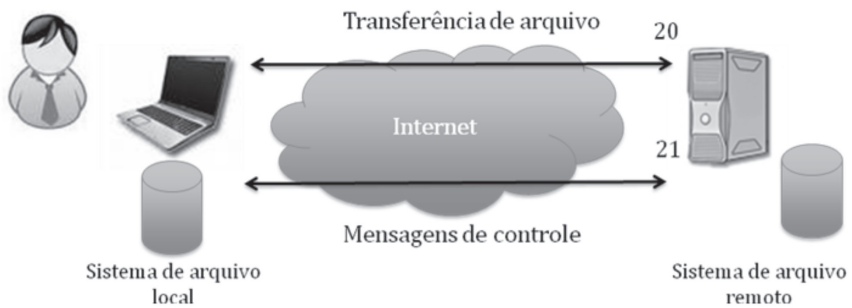


Figura 24: Sinalização fora de banda do protocolo FTP

PROTOCOLO DNS

Nos próximos capítulos estudar-se-á o funcionamento do protocolo IP posicionado na camada de rede da arquitetura TCP/IP. Antes desse estudo é importante mencionar que todos os hosts da Internet são referenciados através de um endereço, chamado endereço IP. O endereço IP é composto de 4 bytes (32 bits), cada byte é representado por um número de 0 a 255.

Considere, por exemplo, um cliente WEB que informa uma URL com

o interesse de acessar a página HTML de uma determinada empresa. Para fazer referência a um host que hospeda um site WEB é necessário utilizar o endereço IP do servidor WEB. Isso porque, na arquitetura TCP/IP, não existe outra forma de contactar um host que não seja através do seu endereço IP. Assim, para enviar uma mensagem HTTP de pedido é requerido o endereço IP do servidor WEB.

Uma situação similar ocorre no relacionamento de nós os seres humanos. Aqui no Brasil todas as pessoas são identificadas unicamente através do número do RG ou do CPF. Entretanto, nós não escrevemos um e-mail ou conversamos com as pessoas utilizando tais números. Para nós seres humanos é mais fácil decorar ou aprender um nome (Adriando) ao invés de número (CPF 123.456.789-01).

Motivado por tal facilidade os hosts que hospedam serviços na Internet possuem um nome além de um endereço IP. Existe um serviço chamado de sistema de nomes de domínios (Domain Name System - DNS) que traduz um nome (URL) para o respectivo endereço IP. O serviço DNS normalmente é usado quando são utilizadas aplicações WEB, E-mail e de transferência de arquivos.

O serviço de nomes é apoiado pelos seguintes componentes: base de dados distribuída, servidor DNS local e o protocolo DNS. O protocolo DNS utiliza o serviço UDP da camada de transporte e opera na porta 53.

A base de dados distribuída segue a hierarquia apresentada na Figura 25.

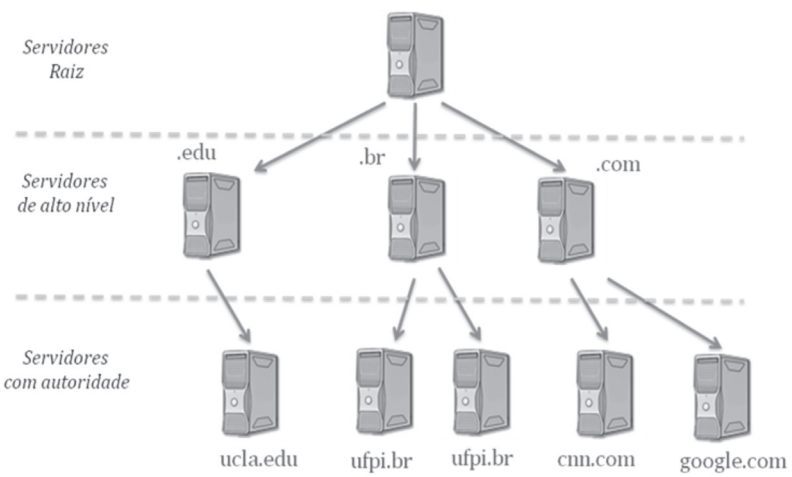


Figura 25: Hierarquia distribuída de servidores DNS.

Ao invés de utilizar um único servidor centralizado para tradução de nomes em endereços IPs o serviço DNS utiliza uma hierarquia de servidores distribuídos. As principais motivações para essa escolha são a capacidade de tolerância a falhas, facilidade de manutenção e ter um servidor DNS próximo de praticamente todos os hosts da Internet. Note que se tal solução fosse centralizada o servidor ficaria próximo de uma minoria e distante da maioria dos hosts. Nesta conduta distribuída os servidores são hierarquicamente organizados em servidores raiz, servidores de alto nível ou servidores com autoridade para um determinado domínio.

Existem um total de 13 servidores raiz (rotulados de “A” até “M”). A maioria desses servidores estão nos Estados Unidos. Vale ressaltar que cada um dos 13 servidores raiz possuem réplicas posicionadas em outras localidades. Por exemplo, existe uma réplica do servidor F localizado na cidade de São Paulo. As motivações de para criação de tais réplicas são: i) diminuição do tempo de resposta de consultas DNS a servidores raiz, ii) realizar um balanceamento de carga e iii) obter um sistema mais seguro e confiável. Os servidores raiz possuem o endereço dos servidores de domínio de alto nível.

Os servidores de alto nível, como o próprio nome sugere, são responsáveis pelos domínios de alto nível. Por exemplo, .com, .br, .fr, .uk, .edu etc. Por usa vez, um servidor de alto nível conhece o endereço dos servidores com autoridade para os seus subdomínios.

A navegação nessa hierarquia de servidores de DNS é intermediada por um outro servidor, chamado servidor DNS local (veja Figura 27) que reside normalmente na mesma rede local do usuário que deseja a tradução do nome em endereço IP.

Veja o exemplo de um usuário que deseja descobrir endereço IP do host que hospeda o servidor WEB da UFPI. Logo após digitar a URL www.ufpi.br no browser WEB entra em cena o serviço DNS. Então o browser faz uma consulta para o servidor DNS local afim de descobrir o endereço IP correspondente ao nome www.ufpi.br. Vale destacar que o endereço IP do servidor DNS local é definido na configuração do endereço IP de cada host que tem acesso a Internet. Isso é necessário para que os hosts possam solicitar aos seus servidores DNS locais traduções de nomes para endereços IPs. Considerando que o servidor DNS local ainda não conhece tal mapeamento ele vai intermediar um consulta com a hierarquia de servidores DNS.

O primeiro passo é fazer uma consulta ao servidor DNS raiz para saber qual o endereço IP do servidor de alto nível para o .br. Com essa resposta o servidor DNS local solicita ao servidor de alto nível .br o endereço para o servidor com autoridade para o domínio ufpi.br. De posse da resposta o servidor local pergunta ao servidor com autoridade para ufpi.br qual o endereço IP que corresponde ao nome www.ufpi.br. Ao descobrir o mapeamento, o servidor DNS local armazena no cache e repassa a resposta para o browser WEB.

Caso surja uma nova consulta DNS para www.ufpi.br o servidor DNS local responderá sem precisar consultar novamente toda a hierarquia distribuída de servidores DNS.

Deve-se destacar que as informações contidas no cache do servidor DNS local normalmente possuem um período de validade. Depois desse tempo tais mapeamentos são apagados do cache.

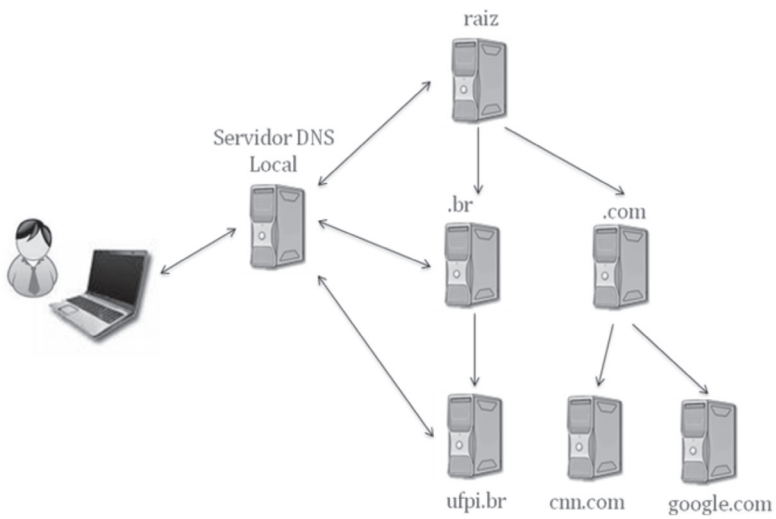


Figura 26: Processo de tradução de nome em endereço IP.

EXERCÍCIOS

1) Quais as funções dos protocolos da camada de aplicação?

2) Cite 5 importantes protocolos da camada de aplicação da arquitetura TCP/IP.

3) Diferencie os protocolos HTTP 1.0 e HTTP 1.1.

4) Diferencie os protocolo POP e SMTP.

5) Qual a função do protocolo DHCP e em qual camada da arquitetura TCP/IP ele está posicionado?

6) Quais os protocolos utilizados nas camadas de aplicação e transporte para dar suporte à aplicação WEB?

7) Qual a vantagem do uso do servidor DNS local na tradução de nomes de host para endereços IPs?

8) Quais os principais componentes do serviço DNS?

9) Explique o conceito de sinalização fora de banda utilizado pelo protocolo FTP.

10) O que é um socket?

UNIDADE 03

Protocolos da Camada de Transporte

Resumindo

Este capítulo apresenta a camada de transporte da pilha de protocolos da arquitetura de rede de Internet. Neste contexto são descrito os protocolos TCP e UDP, evidenciando os serviços, vantagens e desvantagens oferecidos por cada um deles.

Será discutido o processo realizado na camada de transporte, fundamental para que os dados sejam entregues para os processos corretos.

No âmbito do TCP destacam-se os seguintes serviços:

- Entrega confiáveis dos dados;
- Controle de congestionamento;
- Controle de fluxo;
- Gerenciamento de conexão.

3 PROTOCOLOS DA CAMADA DE TRANSPORTE

INTRODUÇÃO

A camada de transporte é a segunda camada de cima para baixo da arquitetura TCP/IP, posicionada imediatamente abaixo da camada de aplicação. Ela promove uma comunicação lógica entre processos, viabilizando a troca de mensagens de protocolos da camada de aplicação.

A unidade de transporte da camada de transporte é o “segmento”. Portanto, o segmento da camada de transporte carrega nos seus campos de dados as mensagens da camada de aplicação. Por exemplo, um pedido de página HTML previsto pelo protocolo HTTP.

A Figura 27 ilustra a comunicação lógica entre processos promovida pela camada de transporte.

Figura 27: Comunicação lógica entre processo provida pela camada de transporte

Note que na Figura 27 a camada de transporte representa uma via por onde são enviadas e recebidas as mensagens da camada de aplicação.

A camada de transporte da arquitetura TCP/IP dispõe de basicamente dois serviços que são oferecidos pelos protocolos *Transmition Control Protocol* - TCP e *User Datagram Protocol* - UDP.

O protocolo UDP oferece um serviço não orientado a conexão e não confiável. O termo não confiável faz alusão a sua não capacidade de recuperar eventuais erros no envio dos seus segmentos. Tais erros pode ser um segmento UDP corrompido ou um segmento que não foi entregue ao *host* de destino em função de um descarte em um roteador intermediário congestionado. Se um segmento UDP for perdido o protocolo UDP não se encarrega de reenviá-lo.

O protocolo TCP oferece um serviço orientado a conexão e confiável. Antes do envio dos dados através do protocolo TCP é requerido o estabelecimento da conexão TCP. Isto porque ele é um protocolo orientado a conexão.

Com relação ao seu serviço confiável, o TCP utiliza mecanismos para fazer uma entrega confiável dos seus segmentos, diferente do que ocorre no protocolo UDP. Se um segmento TCP for perdido o protocolo se encarrega de retransmiti-lo. Para isso são utilizados alguns mecanismos como: temporizador, avisos de reconhecimento (Ack), número de sequência e soma de verificação. Tais mecanismos serão detalhados na Seção 3.5.1.

Quando uma aplicação distribuída está sendo desenvolvida o desenvolvedor especifica qual serviço da camada de transporte é desejado, TCP ou UDP.

MULTIPLEXAÇÃO E DEMULTIPLEXAÇÃO

Atualmente, a maioria dos sistemas operacionais é multitarefa e executa mais de um processo, permitindo que tais processos compartilhem o processador. Considere, por exemplo, que uma determinada máquina A está executando três processos. Quando a máquina A recebe uma informação da rede essa informação deve ser encaminhada para qual dos três processos?

A ação de entregar as informações transportadas pelos segmentos (unidade de transporte da camada de transporte) para um processo específico é chamada de demultiplexação. O processo inverso, encaminhar segmentos de diferentes processos em um host de origem através da camada de transporte é chamado de multiplexação.

Como a camada de transporte fornece uma comunicação lógica entre processos, os protocolos da camada de transporte precisam implementar a multiplexação e a demultiplexação. A Figura 28 mostra um exemplo de demultiplexação.

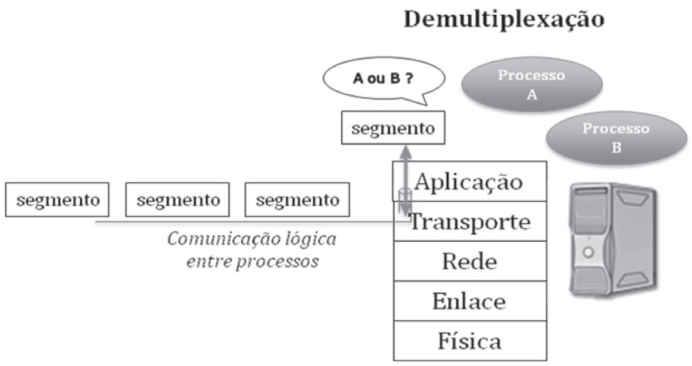


Figura 28: Ilustração do processo de demultiplexação da camada de transporte.

A camada de transporte da arquitetura TCP/IP define dois campos de endereçamento nos cabeçalhos dos segmentos (TCP e UDP) que são utilizados no processo de multiplexação e demultiplexação. Esses campos são os endereços de porta de origem e de destino que podem assumir valores de 0 a 65535. Os valores de 0 a 1023 são reservados a serviços bem conhecidos como HTTP, FTP, UDP, SMTP, POP3, DNS etc.

Quando um pedido HTTP chega em uma máquina que hospeda um servidor WEB a demultiplexação deve ser feita de forma a garantir que o segmento TCP (que transporta a mensagem HTTP) seja encaminhado ao processo servidor WEB. Isso é feito utilizando o endereço de porta de destino 80. A Quadro 2 mostra um mapeamento de porta para serviços bem conhecidos da Internet.

Quadro 2 - Números de porta utilizados por importantes serviços da Internet.

Serviço	Porta
http	80
FTP	20 e 21
DNS	53
SMTP	25
POP3	110

Se os serviços da Internet (HTTP, FTP, DNS etc) não possuíssem uma porta previamente conhecida, todos os clientes seriam obrigados a descobrir em qual porta cada serviço está operando. Para evitar esse problema existe um mapeamento padrão entre o número da porta e os serviços bem conhecidos.

Vale ressaltar que a multiplexação e a demultiplexação não utilizam apenas os endereços de portas origem e destino. Além desses, são utilizados também os endereços de origem e/ou de destino da camada de rede (endereços IPs).

O protocolo UDP faz a demultiplexação baseado no par de endereços (porta de destino, endereço IP de destino). Já o protocolo TCP faz a demultiplexção de acordo com a quádrupla (porta de origem, porta de destino, IP de origem, IP de destino).

SOCKET

Conforme já foi discutido, a camada de transporte implementa uma comunicação lógica entre processos. A comunicação entre dois processo exige que os dados trocados atravessem o núcleo da rede passando pelas camadas inferiores (rede, enlace e física). Assim, pode-se dizer que um processo envia e recebe dados diretamente da rede.

Para um processo que coopera com outro, a fim de viabilizar uma aplicação distribuída, a infraestrutura de rede é abstraída por uma interface entre a camada de aplicação e a camada de transporte. Essa interface é chamada de Socket. Quando um processo deseja encaminhar uma informação para outro processo, ele encaminha os dados para um socket. O socket é uma espécie de portão onde tudo que chega ou sai de ou para um processo deve passar por ele.

A Figura 29 ilustra os sockets de dois processos que cooperam via rede.

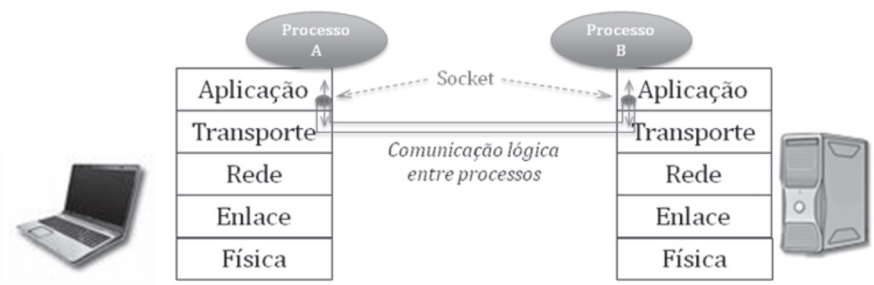


Figura 29: Exemplo de funcionamento do Socket.

No processo de desenvolvimento da aplicação o desenvolvedor utiliza uma *Application Program Interface* (API) para implementar o socket. Note que a abstração da infraestrutura de rede utilizando o conceito do socket simplifica

significativamente o trabalho do desenvolvedor da aplicação distribuída. Ele apenas precisa criar o socket e utilizar as referências apropriadas para se comunicar com o processo correto do outro lado da rede.

O desenvolvedor não precisa saber por quais roteadores os dados serão envidados, qual a tecnologia empregada nos enlaces da rota utilizada ou mesmo qual a rota utilizada. Tal abordagem potencializa o desenvolvimento de novas aplicações, uma vez que é muito simples viabilizar a comunicação entre dois processos via rede utilizando sockets.

PROTOCOLO UDP

O protocolo *User Datagram Protocol* – UDP é definido na RFC 768 e implementa um serviço não orientado a conexão e não confiável. Esse protocolo da camada de transporte implementa mecanismos de verificação de erros e a multiplexação e demultiplexação.

Quando um desenvolvedor opta pelo uso do serviço UDP da camada de transporte ele deseja fazer uso da sua simplicidade e regra geral não possui interesse em garantir uma entrega confiável de dados. Esse é o caso do serviço DNS que faz uso do protocolo UDP. Deve-se destacar que, por não ser orientado a conexão, o protocolo UDP é mais ágil do que o protocolo TCP. O protocolo TCP requer uma troca inicial de informações de controle entre os processos interessados para o estabelecimento da conexão. A Figura 31 apresenta o formato do segmento UDP.

Os campos de porta de origem e destino são utilizados para multiplexação e demultiplexação e cada um possui 16 bits. O campo de comprimento indica o tamanho de todo o segmento UDP (cabeçalho + dados) em Bytes.

Figura 30: Formato do segmento UDP.

O UDP implementa um mecanismo de verificação de erros com o objetivo de detectar bits corrompidos em todo o segmento. O segmento UDP é dividido em palavras de 16 bits. As palavras são somadas e ao final calcula-se o complemento de 1.

A Figura 31 ilustra o cálculo do campo de soma de verificação (*checksum*) considerando apenas duas palavras de 16 bits.

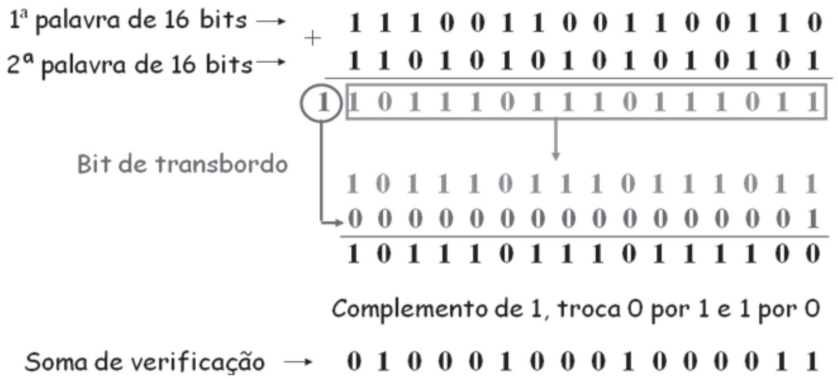


Figura 31: Cálculo do campo de soma de verificação no remetente.

Observe que no exemplo da Figura 31 ocorreu o transbordo do bit 1 que passa a fazer parte de um terceira palavra de 16 bits (0000000000000001).

Quando o segmento chega no receptor mais uma vez ele é dividido em palavras de 16 bits, incluindo o campo *checksum* que foi calculado na origem. O resultado da soma das palavras de 16 bits no destino deve ser também uma palavra com 16 bits 1.

A Figura 32 apresenta o processo de verificação no destino de acordo com o cálculo do *checksum* obtido no exemplo da Figura 31.

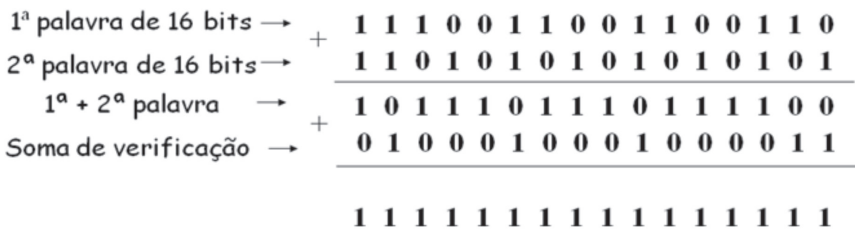


Figura 32: Verificação da corretude do segmento no destino

PROTOCOLO TCP

Um dos protocolos mais importantes da arquitetura da Internet é o protocolo *Transmission Control Protocol* (TCP) da camada de transporte definido nas RFCs 793, 1122, 1323, 2018 e 2581. Certamente por isso ele emprestada seu nome para compor o nome da arquitetura da Internet (arquitetura TCP/IP).

O protocolo TCP implementa um amplo conjunto de serviços: entrega confiável de dados, controle de congestionamento e controle de fluxo. Por ser o protocolo da camada de transporte que implementa um serviço de entrega confiável de dados ele passa a ser utilizado por todos os serviços da Internet que não toleram perdas. Por exemplo, HTTP, FTP, SMTP, POP3 TELNET etc. Apesar de ser orientado a conexão o TCP não viabiliza um circuito físico fim a fim.

A conexão TCP é controlada exclusivamente pelos processos interessados na comunicação. Processo origem ou remetente é aquele que envia segmentos para o processo destino ou destinatário. No decorrer das próximas seções será tratado dos mecanismos do TCP visualizando apenas um sentido da conexão TCP. Entretanto, deve-se destacar que o mesmo ocorre no sentido inverso.

Lembre-se que as aplicações distribuídas viabilizam uma cooperação de processo utilizando a infraestrutura da rede. Por exemplo, uma solicitação de página WEB é feita no sentido browser para o servidor WEB, mas a resposta do servidor WEB para o Browser é no sentido oposto. Considerando essa necessidade a conexão TCP é full-duplex. Isto é, ocorre a comunicação do cliente com o servidor em paralelo com a comunicação do servidor com o cliente. Por simplicidade serão apresentado os conceitos e mecanismos do TCP visualizando apenas o primeiro sentido da comunicação, cliente (origem) para o servidor (destino).

Entrega confiável de dados

Por que é necessário um serviço de entrega confiável de dados na camada de transporte? No modelo baseado em camadas a camada inferior sempre presta serviço para camada superior. Portanto, na arquitetura TCP/IP o protocolo IP (camada de rede) presta serviço para os protocolos da camada de transporte. O protocolo IP será detalhado no Capítulo 4, mas para um

melhor entendimento sobre as necessidades do serviço de entrega confiável de dados é interessante conhecer algumas características do protocolo IP.

O protocolo IP segue o paradigma de comutação por pacote onde não existe reserva de recursos. De forma simplista, nesse modelo, um pacote IP pode precisar atravessar um roteador congestionado. Por razões de congestionamento o roteador pode não ter a capacidade de processar o pacote. O pior caso em um cenário de congestionamento é chamado de descarte de pacote.

O pacote que não pode ser tratado é literalmente descartado. Portanto, os dados transportados (segmentos da camada de transporte) no pacote IP descartado pelo roteador congestionado são efetivamente perdidos. O pior é que protocolo IP não fará nada para recuperar essa perda.

Diante do exposto, é fundamental um serviço na camada de transporte para reagir às condições de descarte de pacotes IP. Isso é feito pelo protocolo TCP. Além da possibilidade de descarte podem haver erros em função bits corrompidos. Por exemplo, por alguma instabilidade um bit 1 pode passar a ser zero e vice-versa.

Para superar os erros ocasionados por bits corrompidos ou por descartes de pacotes o protocolo TCP emprega as seguintes técnicas: soma de verificação, número de sequência, reconhecimento de segmentos recebidos e temporizadores.

Considera-se inicialmente um contexto onde segmentos TCP chegam no destino com bits corrompidos. Esses erros ocorrem geralmente em componentes físicos da rede no processo de transmissão do pacote IP que contem o segmento TCP. Ao fazer o cálculo da soma de verificação com complemento de 1 no destino (é a mesma técnica utilizada no protocolo UDP) detecta-se que um segmento possui bits corrompidos. Para reagir, o TCP solicita a retransmissão do segmento com erro. Isso é feito enviando um segmento de reconhecimento negativo. Quando um segmento chega em perfeito estado o destino sinaliza com um reconhecimento positivo. A mensagem de reconhecimento (positivo ou negativo) é um mecanismo de *feedback* para o processo de origem saber se um dado segmento enviado foi recebido ou não com sucesso.

No contexto de redes de computadores essa sinalização é chamada de ack (*acknoledgment*) para um *feedback* positivo e nack para uma sinalização negativa. O ack é também chamado de mensagem de reconhecimento.

O TCP opera apenas com ack, sinalização de *feedback* positivo. Na verdade, para fazer um feedback negativo o TCP envia uma confirmação positiva para o último segmento recebido com sucesso.

O uso de ack não resolve o problema em cenários onde o segmento é perdido em consequência do descarte de pacotes IPs. Isso porque nada chegará no destino e por conseguinte não é possível realizar uma soma de verificação.

Para reagir a situações de descarte de pacotes IP o TCP utiliza um mecanismo de temporização no remetente. Se depois de um determinado tempo a sinalização de reconhecimento (ack) não chegar na origem é porque provavelmente o segmento enviado (ou o seu feedback) foi descartado.

Em situação de descate de pacote, o mecanismo de temporização na origem detecta que um tempo maior do que o esperado passou e mensagem ack não chegou. Sua reação é retransmissão do segmento em questão. Neste caso o TCP interpreta que o pacote foi descartado em algum roteador congestionado. Portanto, sua reação é retransmitir o segmento cujo ack não chegou dentro do tempo esperado. Vale ressaltar que o TCP não recebe informação alguma da rede sobre congestionamento. Ele na verdade desconfia do congestionamento apos um evento de estouro de temporizador, por exemplo.

É importante destacar que o tempo limite definido no temporizador é baseado em uma estimativa do *Round Trip Time* (RTT). Esse é o tempo entre o envio de um segmento (do processo remetente para o processo destino) e recebimento de um ack (enviado do processo destinatário para o processo remetente). A Figura 33 ilustra o tempo do RTT.

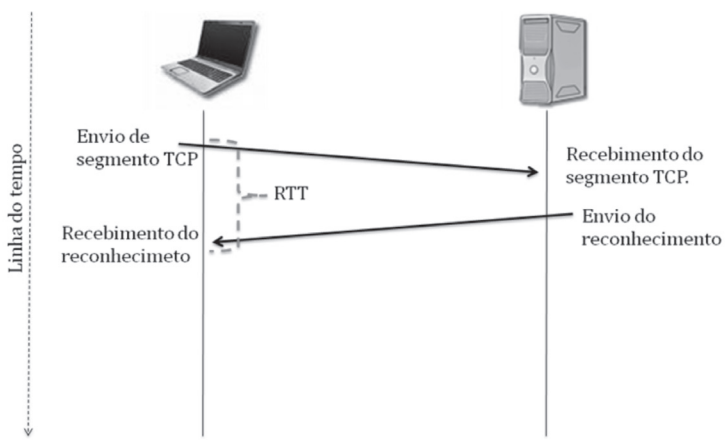


Figura 33: Round Trip Time - RTT.

Com base nos RTTs medidos para segmentos/reconhecimentos enviados anteriormente, o protocolo TCP estima um valor para o próximo RTT. Essa é uma estimativa do RTT para o próximo segmento a ser transmitido. O tempo máximo até o estouro do temporizador é baseado na estimativa do RTT. O tempo de espera até o estouro do temporizado conta com um folga (estimativa do RTT + folga) afim de evitar retransmissões prematuras.

Uma retransmissão prematura ocorre normalmente quando pacotes IP sofrem um atraso significativo em função de filas nos roteadores. Esse também é um cenário de congestionamento, porém, menos caótico do que o cenário de descarte de pacotes. Em função desse atraso inesperado, um segmento/reconhecimento (transportado dentro de um pacote IP) demora mais do que o esperado. O temporizado estoura antes do segmento TCP ou seu ack chegar. Neste caso ocorre uma retransmissão desnecessária, dois ou mais segmentos repetidos chegarão no destino. Em situações como essa é que se faz necessário o uso dos números de sequência.

O número de sequência é uma identificação para diferenciar segmentos TCPs que pertencem a uma mesma conexão. Assim, em casos de retransmissão prematura o destinatário identifica segmento iguais e descarta o segmento repetido.

É importante relembrar que, de fato, o TCP não utiliza nacks. Ao invés de fazer um *feedback* negativo usando *nacks* o TCP envia um ack para o segmento imediatamente anterior recebido com sucesso. Por exemplo, o TCP utiliza acks duplicados para um segmento de número de sequência 5 com objetivo de informar que o segmento de número de sequência 6 não foi recebido com sucesso.

A Figura 34 ilustra a sinalização negativa feita pelo protocolo TCP através de acks duplicados.

Inicialmente, no exemplo da Figura 34, é enviado e recebido com sucesso o segmento com número de sequência 5 o que acarreta naturalmente um sinalização positiva, uma ack para o segmento de número de sequência 6 (nSeq = 6). Note que o feedback positivo feito pelo protocolo TCP é na verdade a solicitação para o próximo byte pendente. Isso é equivalente a uma mensagem do receptor para o transmissor dizendo: mande-me agora o próximo segmento. Então, quando o receptor deseja sinalizar que recebeu o segmento nSeq=5 ele envia uma ack para o próximo segmento (nSeq=6).

Continuando no exemplo da Figura 34, após receber o ack nSeq=6 (que significa que nSeq=5 foi recebido com sucesso) a origem envia o

segmento nSeq = 6. Esse por sua vez chega corrompido no destino. Então, para fazer um feedback negativo, do destino para a origem, é enviado um ack para o nSeq = 6. Note que nSeq = 6 é o número de sequência que ainda continua sendo aguardado pelo receptor.

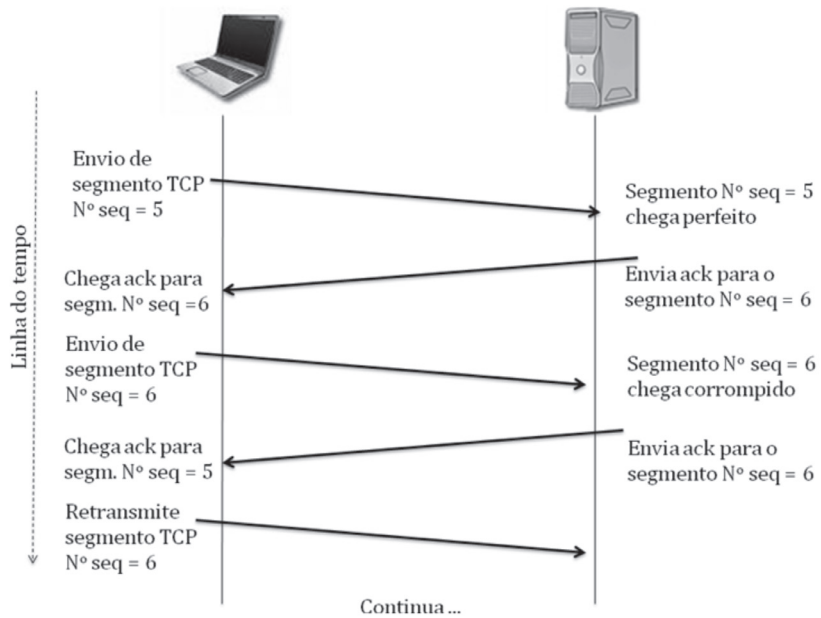


Figura 34: Reconhecimento negativo através de acks duplicados para último segmento recebido com sucesso.

Esse procedimento acarreta no recebimento de acks duplicados para o segmento nSeq = 6. Na origem da conexão TCP a interpretação para os acks duplicados para o nSeq=6 é que o segmento nSeq = 6 ainda não foi recebido corretamente. Portanto, a origem da conexão retransmite o segmento nSeq = 6.

Os exemplos apresentados sobre entrega confiável de dados são de cenários simplistas como objetivos didáticos. De fato, o TCP é um pouco mais sofisticado mas segue os princípios aqui apresentados.

Controle de congestionamento

Outro serviço implementado exclusivamente pelo protocolo TCP na camada de transporte é o controle de congestionamento. Esse serviço visa mitigar o desperdício de recursos em função de congestionamentos nos roteadores da Internet.

Conforme discutido anteriormente, por fazer uso da comutação de pacotes são esperadas condições de congestionamento quando for submetida ao roteador uma demanda maior do que a sua capacidade de encaminhar pacotes IP.

No início do congestionamento ocorre a formação de filas. Se ao chegar um pacote IP no roteador já houver um outro pacote (recebido anteriormente) sendo processado, o pacote que acabou de chegar fica aguardando sua vez em um fila. A fila é na verdade um buffer que armazena pacotes IPs que aguardam para serem atendidos.

Se o desequilíbrio entre demanda e a capacidade nos roteadores for mantido por um período significativo de tempo, as filas do roteador tendem a crescer de forma a ocupar toda a capacidade do buffer. Assim, o próximo pacote que chegar não poderá ser armazenado. Consequentemente esse pacote e o segmento que ele transporta serão descartados.

Considere que se um dado pacote IP (que transporta um segmento TCP com nSeq = x) é descartado no terceiro roteador de sua rota até o destino. Como o protocolo TCP está sendo utilizado na camada de transporte, o segmento TCP nSeq=x descartado junto com o pacote IP será retransmitido após um evento de estouro de temporizador. Entretanto, deve-se atentar que a capacidade utilizada na primeira transmissão do pacote IP (contendo o segmento nSeq=x) nos dois primeiros rotadores foi totalmente desperdiçada, uma vez que o segmento nSeq=x será retransmitido.

Diante das considerações acima é fácil notar que a condição de congestionamento desperdiça o uso dos recursos em função do alto número de retransmissões. A reação do TCP mediante a detecção de congestionamento é diminuir a taxa de envio de segmentos.

O TCP controla a taxa de segmentos enviados no escopo de uma conexão através da janela de transmissão de segmentos. Essa janela indica quantos segmentos TCP (na verdade indica quantidade máxima de bytes) podem ser transmitidos sequencialmente sem o recebimento do ack para o segmento mais antigo. Portanto, quando o protocolo TCP identifica uma situação de congestionamento a sua reação é diminuir a janela de transmissão. Isso diminui a carga da conexão TCP em questão sobre os roteadores congestionados. Tal procedimento tende a descongestionar os roteadores sobrecarregados. Uma pergunta ainda não foi respondida é, como o TCP identifica que roteadores estão congestionados?

Efetivamente, o protocolo TCP não sabe que o roteador está congestionado, ele desconfia do congestionamento quando ocorre um evento que está associado a congestionamento, por exemplo, um estouro de temporizador. Após a ocorrência de eventos que sugerem congestionamento nos roteadores utilizados por uma determinada conexão TCP, a reação do TCP é diminuir a janela de transmissão.

Todas as conexões TCP que utilizarem os roteadores congestionados e perceberem eventos ligados a congestionamento reduzirão suas janelas de transmissão. Isso certamente provocará o descongestionamento dos roteadores afetados.

A Figura 35 ilustra o funcionamento da janela de transmissão do protocolo TCP, também conhecido como protocolo de janelas deslizantes.

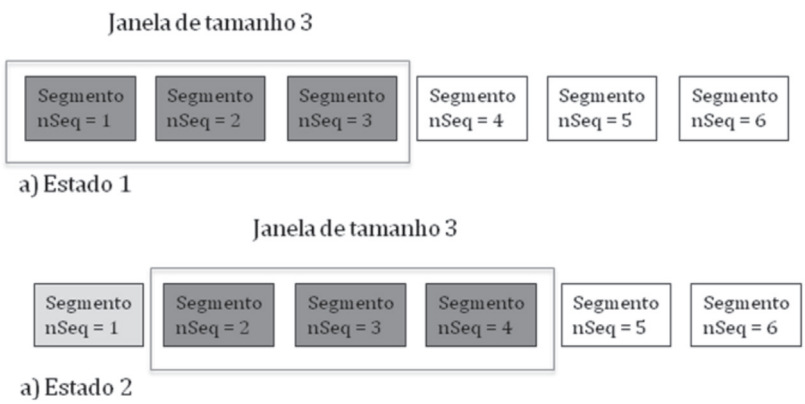


Figura 35: Janela de transmissão de segmentos.

No exemplo da Figura 36, observa-se que o processo cliente possui 6 segmentos (com número de sequência de 1 a 6) prontos para serem enviados ao processo servidor (do outro lado da conexão TCP). Entretanto, no primeiro momento (Estado 1), somente os 3 primeiros segmento são enviados em função do tamanho da janela de transmissão. Após a chegada do ack referente ao segmento mais antigo enviado e ainda não confirmado (nSeq=1) a janela desliza para direita permitindo o envio de mais um segmento (nSeq=4). Note que, se o tamanho da janela for reduzido naturalmente diminui a quantidade de dados enviados por unidade de tempo. Portanto, a diminuição da janela de transmissão tem impacto positivo mitigando o congestionamento.

O leitor mais curioso que desejar conhecer o funcionamento do controle de congestionamento do TCP com detalhes pode ler o livro do Prof. Kurose.

Controle de fluxo

O controle de fluxo é outra função implementada pelo protocolo TCP. Seu objetivo é evitar que um processo rápido inunde um outro processo receptor mais lento.

Ao estabelecer uma conexão TCP são reservados buffer na origem e no destino. Os buffers da conexão TCP servem, por exemplo, para armazenar segmentos que chegaram do outro lado da conexão TCP mas ainda não foram lidos pelo processo. Nesse contexto, um processo remetente mais rápido pode esgotar o buffer de um processo destinatário mais lento. Para impedir isso o TCP implementa um serviço de controle de fluxo.

O protocolo TCP implementa uma conexão full-duplex. Portanto, quando um processo A envia segmentos com dados da aplicação para o processo B ele sinaliza quanto de memória disponível ele (processo A) tem. Assim, o lado da conexão TCP do processo B não envia mais segmentos do que o lado do processo A consegue armazenar. O mesmo é feito no sentido inverso. Com isso o TCP utiliza a informação sobre quanto de memória disponível o outro lado da conexão TCP dispõe para armazenar segmentos. Essa informação é utilizada para saber o tamanho da janela de transmissão. De fato, o tamanho da janela de transmissão é definido em termos do controle de fluxo e do controle de congestionamento. Note que a solução para ambos os controles é diminuir ou aumentar a taxa de transmissão de segmentos, logo diminuir ou aumentar a janela de transmissão.

Estabelecimento e finalização de conexão TCP

Conforme já dito o protocolo TCP é orientado à conexão. O estado de uma conexão TCP é conhecido e gerenciado apenas pelas respectivas camadas de transporte dos *host* interessados na comunicação. Os roteadores do núcleo da rede desconhecem as conexões TCP.

Para estabelecer uma conexão TCP o processo cliente envia um segmento especial pedindo a abertura de uma conexão. Esse segmento especial é conhecido como segmento SYN. Ele não transporta dados da aplicação, sua função é apenas informar que o processo cliente está interessado em abrir uma conexão.

Para aceitar o pedido de conexão o processo servidor responde com um segmento SYN ack (esse segmento também não contem dados

da aplicação). Ao receber o SYN ack o cliente reserva buffers e variáveis para gerenciar a conexão TCP e envia um ack para o SYN ack enviado do servidor. Esse reconhecimento pode conter dados da aplicação, por exemplo, uma mensagem de pedido HTTP.

O procedimento para estabelecimento de uma conexão TCP também é conhecido como apresentação de três vias. Note que somente depois de um RTT (envio de um segmento SYN e recebimento de um segmento SYN ack) o processo pode enviar dados da aplicação para o servidor.

Utiliza-se o exemplo de um servidor WEB para explicar alguns detalhes do processo de demultiplexação de segmentos TCP.

Considere três processo, 2 cliente WEB (C1 e C2) e 1 servidor WEB (S) que estão em execução em três *hosts* distintos. Antes de qualquer coisa o processo servidor está pronto para receber pedidos HTTP na porta 80.

Considere que inicialmente C1 solicita uma conexão TCP com S afim de fazer solicitações HTTP. Lembre-se de que inicialmente é necessário o estabelecimento da conexão antes de trocar quaisquer informações referentes a aplicação propriamente dita. Outro lembrete é que a comunicação entre processos é utiliza-se interface socket de cada lado da conexão TCP. Cada uma dessas interfaces é a abstração da conexão TCP para os respectivos processos nas extremidades da conexão. Então, C1 envia através do seu socket um segmento SYN para o servidor S. O servidor S recebe o segmento TCP SYN através do seu socket padrão associado a porta 80. Ao aceitar a conexão TCP com C1 o servidor S cria um novo socket que será responsável exclusivamente pela conexão TCP criada entre C1 e S.

Quando o cliente C2 solicitar uma conexão TCP com S haverá a criação de um terceiro socket no servidor S. O primeiro socket (que entra em operação logo que o serviço WEB é inciado) é responsável apenas por atender os pedidos de conexão, ele aguarda segmentos TCP SYN. Os outros dois sockets foram criados a partir do primeiro para gerenciar duas conexões TCP distintas.

Para gerenciar a demultiplexação de segmentos TCP que chegam no servidor S utiliza-se uma quádrupla de endereços (porta de origem, porta de destino, endereço IP de origem, endereço IP de destino). Essa chave composta é utilizada para encaminhar segmentos para o socket correto. Note que dessa forma é possível diferenciar duas conexões TCP tendo como origens processos de um mesmo *host,* cada uma originada por uma instâncias do browser WEB diferente (solicitando páginas diferentes).

Para finalização de uma conexão TCP utiliza-se o segmento especial FIN. Quando um cliente deseja finalizar uma conexão TCP ele envia para o servidor um segmento TCP FIN em seguida o servidor responde com um ack para o segmento TCP FIN. No segundo momento o servidor envia um segmento TCP FIN para o cliente que em seguida envia um ack reconhecendo o segmento TCP FIN enviado pelo servidor. Depois disso a conexão TCP é finalizada.

Formato do segmento TCP

O formato do segmento TCP é ilustrado na Figura 36.

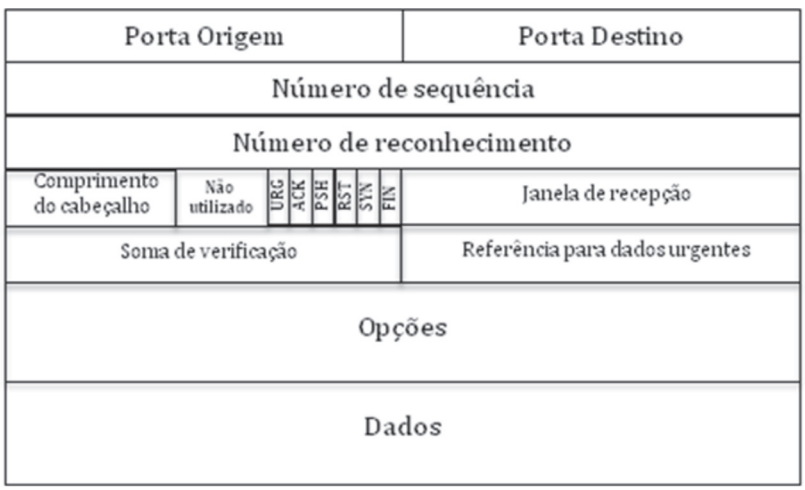


Figura 36: Campos do segmento TCP.

Os campos de porta de origem e destino (16 bits) do segmento TCP, assim como segmento UDP, são utilizados no processo de multiplexação e demultiplexão. A porta de origem é a porta utilizada pelo processo que envia o segmento e a porta de destino é a porta utilizada pelo processo destinatário.

Deve-se ressaltar que um processo é identificado na demultiplexação do TCP pela quadra (porta origem, porta destino, endereço IP de origem, Ip de destino).

O campo de número de sequência (32 bits) é utilizado para identificar um segmento dentro de uma conexão TCP. Essa funcionalidade é necessária para prover o serviço de entrega confiável de dados. Ainda relacionado ao número de sequência é necessário fazer um esclarecimento. Ao invés do TCP utilizar números incrementais, por exemplo, nSeq =1, nSeq = 2, nSeq =3, ele opera da seguinte forma.

Suponha que para enviar uma mensagem da camada de aplicação de 5000 bytes serão necessários 5 segmento TCP cada um de 1000 bytes. O TCP utiliza o número do primeiro byte de cada segmento (relativo a posição de toda a mensagem) para representar o número de sequência. O primeiro segmento TCP terá nSeq = 0 o segundo segmento terá nSeq = 1000, o terceiro nSeq=2000 e o quinto segmento terá nSeq = 4000. Isso porque o TCP implementa um serviço de entrega confiável para um fluxo de bytes a ser transmitido por sua conexão.

Note que o primeiro byte do segmento 1 transporta o byte zero da mensagem de 5000 bytes. O primeiro byte do segmento 5 transporta o byte 4000 da mensagem de 5000 bytes.

O número de reconhecimento (32 bits) serve para sinalizar um ack. Considera-se mais uma vez o exemplo da transferência de uma mensagem de aplicação de 5000 bytes. O lado do destino da conexão TCP sinaliza que recebeu corretamente o segmento nSeq = 1000 (que transporta 1000 bytes) enviando para o outro lado da conexão um segmento TCP com o número 2000 no campo de reconhecimento. Isto equivale a dizer, mande-me agora o segmento com nSeq=2000 pois recebi o nSeq=1000 com sucesso. Como a conexão TCP é full-duplex esse tipo de sinalização pode ser enviado de carona. Por exemplo, quando um processo A envia um segmento para o processo B, pode ser feito o reconhecimento de um segmento que o processo B enviou anteriormente para A. Essa carona é conhecida com *pingback*.

O campo comprimento do cabeçalho, como o próprio nome diz define o tamanho da cabeçalho do segmento TCP. O tamanho máximo da carga útil (dados da camada de aplicação, também conhecido como *payload*) do segmento TCP é chamado de *Maximum Segment Size* (MMS). O MSS é negociado na fase de estabelecimento da conexão TCP.

O cabeçalho TCP conta com 6 bits de sinalização. O bit SYN quando ligado indica que o segmento é um pedido de conexão TCP. O bit FIN indica o pedido de finalização da conexão. O bit ACK sinaliza que o valor transportado no campo de número de reconhecimento é valido. O bit RST sinaliza não é possível atender a conexão pois não existe um processo esperando pedido de conexão na porta informada. Isto é, houve um erro no processo de abertura de conexão pois não existe processo “escutando” a porta informada no pedido de conexão. Em situações desse tipo o TCP retorna para a origem (processo que solicitou a conexão) um segmento RST sinalizando a reiniciacialização. O bit PSH indica que o destinatário deve passar os dados imediatamente para

a camada superior. O bit URG sinaliza que há dados urgentes no segmento. Na prática os bits PSH e URG não são utilizados.

O campo de opções é utilizado quando um remetente negocia com o destinatário o tamanho do MSS. O campo de opções tem tamanho variável.

EXERCÍCIOS

1) Ilustre e explique detalhadamente como é feita a multiplexação e demultiplexação na camada de transporte.

2) Cite as principais diferenças entre os protocolos TCP e UDP.

3) Cite pelo menos duas aplicações que utilizam o protocolo UDP e outras duas aplicações que utilizam o protocolo TCP.

4) Descreva o propósito dos campos do protocolo UDP.

5) Descreva o propósito dos campos do protocolo TCP.

6) O que é controle de congestionamento?

7) O que é controle de fluxo?

8) O que é um protocolo de janelas deslizantes?

9) Quais os principais mecanismos utilizados pelo protocolo TCP para implementar um serviço de entrega confiável de dados.

10) Como é estabelecida e finalizada uma conexão TCP?

UNIDADE 04

Protocolos da Camada de Rede

Resumindo

Este capítulo apresenta conceitos da camada de rede da arquitetura TCP/IP. Em se tratando da internet a camada de rede é representada especialmente pelo protocolo IP, os roteadores IP e os algoritmos de roteamento. Neste cenário, este capítulo trata das seguintes questões relevantes:

- Endereçamento IP;
- Funções da camada de rede;
- Protocolos IPv4 e IPv6;
- Nat;
- Algoritmos de roteamento.

4 PROTOCOLOS DA CAMADA DE REDE

INTRODUÇÃO

A unidade de transporte da camada de rede é o pacote. O protocolo da camada de rede é o *Internet Protocol* (IP). O pacote IP tem como obrigação transportar segmentos da camada de transporte (seja TCP ou UDP) de um host de origem para um host de destino. Por isso, diz-se que a camada de rede fornece uma comunicação lógica entre *hosts*. Para ser mais exato, a camada de rede viabiliza uma comunicação lógica entre interfaces de rede (placas de rede), isso porque um endereço da camada de rede é definido para cada interface de rede de um dado host.

Quando alguém deseja, por exemplo, acessar o servidor WEB da UFPI é necessário transportar a mensagem de pedido HTTP dentro de segmentos TCPs que por sua vez viajam dentro de pacotes IP. Esses pacotes devem ser endereçados com o endereço IP da interface de rede do servidor WEB da UFPI.

A Figura 37 ilustra a verificação da interface de rede do *host* que hospeda o site da UFPI.

Figura 37: Ilustração do processo de verificação de operabilidade da interface de rede do *host* www.ufpi.br através do programa ping.

Esse processo de verificação é feito com o auxílio do programa chamado ping. Normalmente, de um outro *host* (chamado neste exemplo de *host* verificador) é enviado um pedido de eco para o endereço IP da placa de rede que se deseja verificar. Portanto, quando o *host* (www.ufpi.br) receber o pedido de eco ele envia uma resposta para o *host* verificador. Ao receber a resposta de eco o *host* verificador constata a operabilidade de *host* (www.ufpi.br).

A Figura 38 mostra a tela com a execução do programa ping no processo de verificação do host www.ufpi.br.

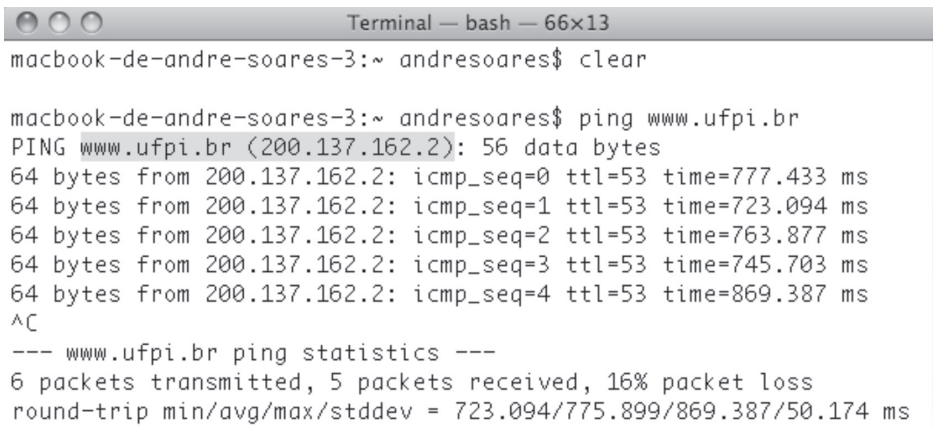


Figura 38: Tela com resultado do programa ping direcionado para host www.ufpi.br.

Observe que o endereço IP do *host* www.ufpi.br é 200.137.162.2. O comando ping foi disparado para a url www.ufpi.br. No primeiro momento entra em cena o serviço DNS responsável por traduzir o nome www.ufpi.br para o IP 200.137.162.2. Note que o *host* verificador enviou para www.ufpi.br 6 pacotes IP com 56 bytes cada um. Desses 6 pacotes enviados com pedido de eco apenas um não foi respondido por www.ufpi.br. Dos pacotes que foram respondidos com eco observa-se o tempo gasto em milissegundos entre o envio e recebimento do eco.

Para enviar qualquer informação para o *host* que hospeda o site WEB da UFPI, deve-se utilizar o seu endereço IP (200.137.162.2). Da mesma forma que quando se deseja enviar uma carta (via correios) para alguém colocamos o endereço dessa pessoa no campo do destinatário. Portanto, no âmbito da camada de rede da Internet utiliza-se o endereço IP de um *host* para enviar algo para ele.

A Figura 39 mostra o envio de pacotes IPs do host do host A cujo endereço IP é 201.93.6.97 para o host B com endereço IP 200.137.162.2.

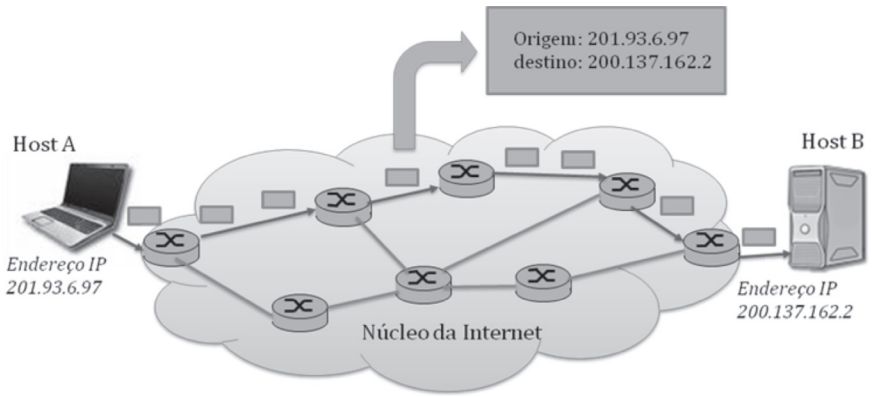


Figura 39: Processo de envio de pacotes IPs baseado no endereço do host de destino.

A Figura 39 ilustra a principal função da camada de rede da arquitetura TCP/IP, comunicação lógica entre *host* baseada nos seus endereços IPs. Note que em cada pacote enviado do *host* A para o *host* B tem o endereço 200.137.162.2 no campo de destino. Esse endereço (200.137.162.2) é utilizado no processo de encaminhamento dos pacotes IPs realizado pelos roteadores que compõem a rota do *host* A para o *host* B.

Observou-se no capítulo anterior que o protocolo TCP implementa um serviço de entrega confiável de dados porque o protocolo IP não é confiável. Então, uma pergunta interessante seria: qual o serviço provido pelo protocolo IP? O modelo de serviço da Internet é conhecido como serviço de melhor esforço (*Best effort*). Esse modelo de serviço da Internet está fortemente ligado ao protocolo IP.

O protocolo IP segue o paradigma de comutação de pacotes. Nesse paradigma não existe reserva de recursos. Os pacotes são enviados via camada de rede (protocolo IP) sem a garantia de que serão entregues aos seus hosts de destino. Essa não garantia de entrega está associada a um problema que chamamos de congestionamento ou sobrecarga dos roteadores. Como o protocolo IP não realiza reserva de recursos, os *host* podem enviar um volume de pacotes maior do que os roteadores conseguem encaminhar. Isso provoca a formação de filas nos roteadores, podendo ocorrer também o descarte de pacotes. Essa característica do protocolo IP impede que ele forneça qualquer tipo de garantia. Então, a expresão “melhor esforço” seria o mesmo que dizer: “vou trabalhar dentro das minhas possibilidades, se não conseguir ter um bom resultado, paciência... “.

PROTOCOLO IPv4

O protocolo *Internet Protocol* versão 4 (IPv4), definido na RFC 791, é o protocolo utilizado na camada de rede da arquitetura TCP/IP. Deve-se destacar que existe outra versão do protocolo IP, como por exemplo o *Internet Protocol* versão 6 (IPv6).

O protocolo IPv6 é um evolução do protocolo IPv4 com o objetivo de aumentar o número de endereços IPs. Além disso, o IPv6 elimina funções técnicamente desnecessárias feitas no IPv4 com o objetivo de melhorar o desempenho do protocolo IP.

Além do IPv6 é possível citar o *Internet Protocol Security* (IPSec) como uma variação do funcionamento normal da camada de rede da arquitetura TCP/IP. O IPSec implementa mecanismos de segurança para viabilizar uma *Virtual Private Network* (VPN). O IPSec é um protocolo padrão de camada 3 projetado pelo IETF cujo objetivo é oferecer um serviço de transferência segura (confidencialidade, autenticação e integridade) de informações fim a fim através de rede IP pública ou privada. Essencialmente, ele recebe pacotes IP privados, realiza funções de segurança de dados como criptografia e então encapsula esses pacotes protegidos em outros pacotes IP para serem transmitidos.

A Figura 40 ilustra o tunelamento de pacotes IPSec através de pacotes IP.

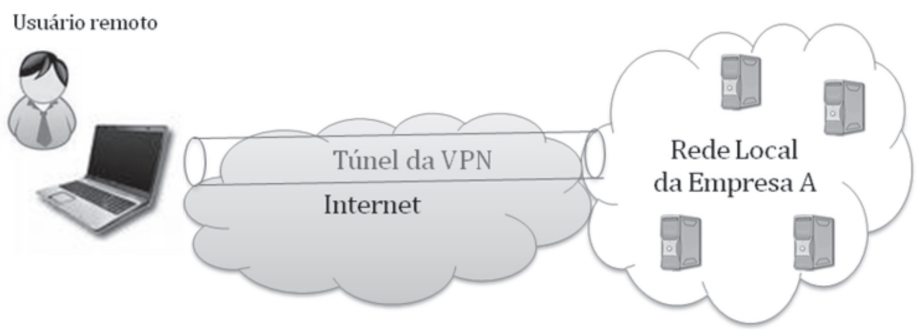


Figura 40 : ilustração do túnel com características de segurança de uma VPN.

Conforme dito anteriormente, os pacotes IP transportam pacotes IPSec que implementam características como confidencialidade, autenticação e integridade. Uma visualização desse serviço seria um túnel seguro viabilizado através da Internet.

ENCAMINHAMENTO

Para que uma camada de rede possa implementar uma comunicação lógica entre *hosts*, ela realiza duas importantes funções: encaminhamento e execução de algoritmos de roteamento.

Encaminhamento é uma tarefa desempenhada pelos roteadores. Ao chegar um pacote IP em um roteador ele analisa o endereço de IP do *host* de destino do pacote e comuta o pacote para uma porta de saída específica. Essa comutação é feita de acordo com uma tabela que contém um mapeamento entre o endereço IP de destino e uma interface de saída do roteador. Essa tabela é chamada de tabela de encaminhamento ou de repasse.

A Figura 41 representa duas comunicaçãos lógicas entre os *hosts*. Uma entre os *hosts* H1 e H2 e outra entre os hosts H1 e H3.

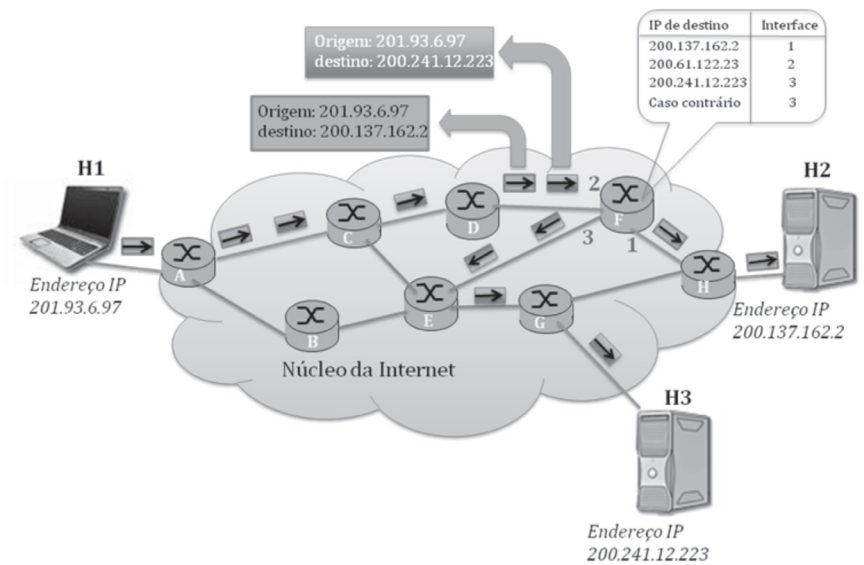


Figura 41: Exemplo de encaminhamento de pacote IP.

Na Figura 41 observa-se o envio de pacotes IPs do host H1 para o host H2 como também de H1 para H3. Os roteadores são dispositivos de interconexão situados no núcleo da rede e que possuem mais de uma interface de rede. Esses dispositivos são responsáveis pelo processo de encaminhamento dos pacotes, isto é, decidir por qual interface de saída do roteador um dado pacote IP deve ser encaminhado.

No exemplo da Figura 41 o roteador F, por exemplo, possui 3 interfaces de rede. A tabela de encaminhamento do roteador F indica que

os pacotes que chegarem com destino ao endereço IP 200.137.162.2 (host H2) devem ser encaminhados pela interface de saída 1. Já os pacotes que possuem como destino o endereço IP 200.241.12.223 devem ser comutados para interface 3.

As regras de encaminhamento são definidas através dos protocolos de roteamento executados pelos roteadores. Os algoritmos de roteamento trocam informações sobre seus enlaces e realizam cálculos visando a configuração das tabelas de encaminhamento.

Formato do pacote IPv4

A Figura 42 ilustra o formato do pacote IPv4 identificando cada um dos seus campos.

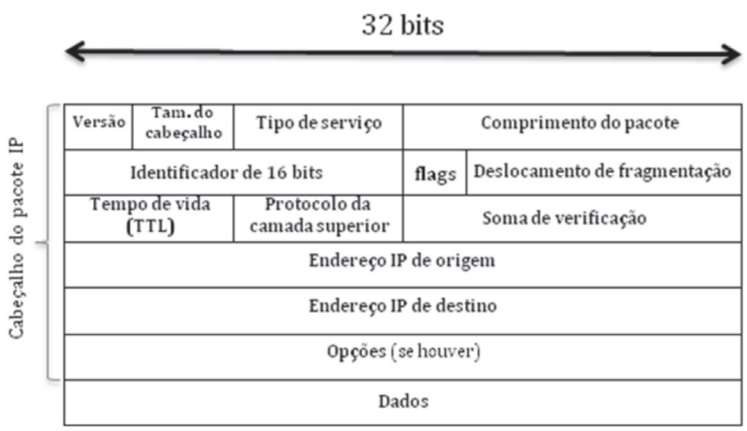


Figura 42: Formato e campos do pacote IPv4.

O primeiro campo do cabeçalho do pacote é a versão do protocolo (4 bits). Quando um pacote chega a um roteador ele precisa ser processado. Considerando que um processador pode executar duas versões diferentes do protocolo IP (por exemplo, IPv4 e IPv6), é fundamental saber qual protocolo deve ser utilizado para processar um pacote que chega a esse roteador.

Por exemplo, se o campo versão indicar o protocolo IPv4 aplicam-se as regras do IPv4. Se o campo versão especificar IPv6, o pacote é processado segundo as regras do IPv6. Note que esse campo foi pensado pelos projetistas dor protocolo IP visando uma possível evolução, como o surgimento de novas versões do protocolo.

O segundo campo do protocolo especifica o **tamanho do cabeçalho** do pacote IPv4 (4 bits). Logo, o tamanho do cabeçalho IPv4 é variável. O protocolo IPv4 prevê um campo opcional chamado de **opções**. Quando esse campo é utilizado o tamanho do cabeçalho IP aumenta. Em função dessa

variação do tamanho do cabeçalho, logo após a chegada de um pacote o roteador precisa saber qual o tamanho do cabeçalho para em seguida processar suas informações. Isso justifica a necessidade do campo do tamanho do cabeçalho. Tipicamente, sem o uso do campo opções, o tamanho do cabeçalho TCP é de 20 Bytes. O valor definido no campo tamanho do cabeçalho é expresso em Bytes.

O campo **tipo de serviço** (8 bits) foi concebido pelos projetistas do IPv4 com o objetivo de classificar os pacotes IPs. A ideia seria criar privilégios baseado na classe de cada pacote IP. Entretanto, normalmente esse campo não é utilizado. Alguns fabricantes de roteadores (por exemplo: cisco) utilizam esse campo para criar um tipo de prioridade entre pacotes. Para isso é necessária a definição de uma política de prioridades pelos administradores da rede.

O **tamanho do pacote** IP (16 bits) informa ao roteador o tamanho total do pacote IP, cabeçalho + carga útil. Esse valor é expresso em bytes, portanto, o tamanho máximo do pacote IP é 65535 bytes. Deve-se atentar que o pacote IP é transportado dentro de um quadro da camada de enlace. A tecnologia de enlace mais comum atualmente é Ethernet (IEEE 802.8). Como um quadro dessa tecnologia de enlace transporta no máximo 1500 bytes, em geral, o tamanho dos pacotes IPs não podem exceder esse limite.

Os campos **identificador** (16 bits), **flags** (1 bit) e **deslocamento de fragmentação** (1 bit) são utilizados para os procedimentos de fragmentação e remontagem do pacote IP. Isso é necessário quando o roteador possui placas de redes que utilizam tecnologias de enlace diferentes. Supondo que um dado roteador utiliza placas de duas tecnologias de enlace diferentes, A e B. (a tecnologia A, que é capaz de transportar até pacotes IP com até 1500 Bytes; a tecnologia de enlace B, empregada na outra interface de rede do roteador, que é capaz de transportar 4000 bytes), o que deve ser feito se chegar um pacote de 4000 bytes pela interface da tecnologia de enlace A e o roteador encaminhá-lo para a interface que utiliza tecnologia de enlace B, que suporta pacotes de até 1500 Bytes?

Em situações como essa o pacote IP precisa ser fragmentado para caber na tecnologia de enlace utilizada na interface de saída. Além disso, os fragmentos gerados no processo de fragmentação do pacote original precisam ser remontados no destino, antes de entregar o segmento para o protocolo da camada de transporte no *host* de destino.

O campo chamado de **tempo de vida**, *Time To Live* – TTL, (8 bits) informa quantos roteadores um dado pacote IP pode atravessar até a chegada no *host* de destino. Quando um pacote IP sai do host de origem é definido um inteiro para o valor do TTL. Toda vez que esse pacote atravessar um roteador o valor do TTL é decrementado de uma unidade. Isso é feito pelo próprio roteador. Se um pacote chegar a um dado roteador e tiver o valor do TTL decrementado para zero, o roteador é obrigado a descartá-lo. Esse procedimento evita que pacotes "zumbis" (pacotes que provavelmente não chegarão aos seus destinos) fiquem perambulando pela rede gerando apenas sobrecarga nos roteadores.

O campo **protocolo da camada superior** (8 bits) sinaliza qual o protocolo utilizado na camada de transporte. Essa informação é utilizada basicamente quando o pacote IP chega no *host* de destino. O valor 6 indica que o segmento transportado é TCP. Já o valor 17 faz referência para o protocolo UDP. Mensagens do protocolo ICMP, utilizadas pelo programa *ping* ilustrado anteriormente, também são transportadas pelo protocolo IP. Quando isso ocorre o campo protocolo da camada superior é 1. Lembre-se de que, uma VPN o protocolo IP transporta pacotes IPSec. Na verdade o IPSec é um conjunto de protocolos para fornecer por exemplo confidencialidade (utiliza-se o protocolo *Encapsulation Security Payload* - ESP), autenticação (utiliza-se *Authentication Header* - AH) e integridade. Quando o protocolo IP transporta um pacote do protocolo ESP utiliza-se o valor 50 no campo protocolo da camada superior.

A **soma de verificação** (16 bits) é utilizada para identificar bits corrompidos apenas no cabeçalho do pacote IP. Como o valor do TTL é alterado em cada roteador, o valor do campo *soma de verificação* deve também ser recalculado. Caso contrário, o próximo roteador detecta que o cabeçalho do pacote foi corrompido. Essa técnica de verificação é praticamente a mesma técnica utilizada nos protocolos da camada de transporte. É feita uma soma considerando cada 2 bytes do cabeçalho e utiliza-se complementos de 1 dessa soma. Esse valor é armazenado no campo soma de verificação. Quando um erro é detectado no cabeçalho o pacote é descartado pelo roteador.

Os campos **endereço IP de origem e destino** (32 bits cada campo) informam, respectivamente, os endereços IP do *host* que enviou o pacote IP e o *host* de destino.

O campo **opções** (múltiplos de 4 bytes) foi projetado considerando que futuramente poderia ser necessário o envio de outras informações

adicionais. A maioria dos pacotes IP não utiliza o campo opções. Sem o campo de opções o cabeçalho IP possui 20 Bytes.

FRAGMENTAÇÃO DO PACOTE IPV4

Conforme apresentado no Capítulo 1, os quadros da camada de enlace são responsáveis por transportar os pacotes IP no enlace (ligação física entre nós adjacentes) em questão. Cada tecnologia da camada de enlace possui uma unidade máxima de transmissão (*Maximum Transmission Unit* - MTU), que é o tamanho máximo de carga útil que o quadro da camada de enlace pode transportar.

Um roteador pode ter diferentes tecnologias de enlace em suas interfaces. Portanto, tais tecnologias podem ter MTUs de tamanho diferente. Nesse contexto, suponha que um dado roteador utiliza duas tecnologias diferentes A e B. A MTU da tecnologia A é 4000 bytes e da B é 1500 bytes. O que fazer quando um pacote IP de 4000 bytes chega através da tecnologia A e deve ser encaminhado para a tecnologia B?

A Figura 43 ilustra esse cenário.

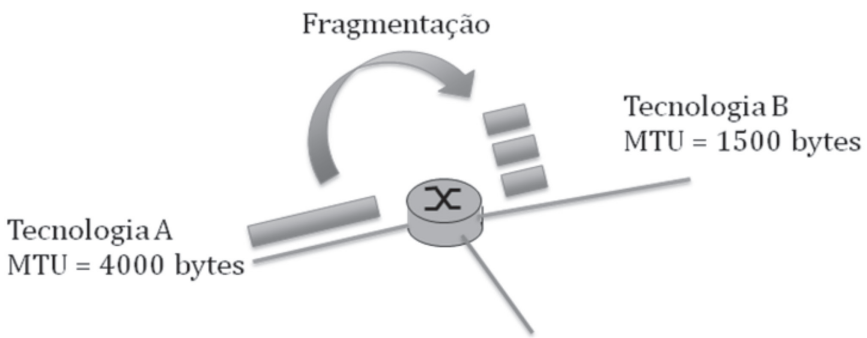


Figura 43: Fragmentação do pacote IPv4.

Seja P1 o pacote que chega pela tecnologia de enlace A com 4000 bytes sendo 20 bytes de cabeçalho. Todo pacote IP possui um campo identificador utilizado para diferenciar os pacotes IP. No processo de fragmentação, o cabeçalho do pacote original P1 deve ser replicado em cada um dos seus fragmentos. Portanto, se a MTU da tecnologia B é de 1500 bytes a carga útil deve ser de 1500 – 20 1480 bytes. Para saber quantos em quantos fragmentos P1 será dividido basta dividir a carga útil de P1 pela carga útil máxima dos fragmentos. Logo, 3980 / 1480 ≅ 2,69 o que significa 3 fragmentos.

O campo **identificador** é copiado de P1 para cada um dos fragmentos. O campo **deslocamento de fragmentação** é utilizado para saber qual a ordem dos fragmentos para remontar o pacote original (P1 no nosso exemplo). Já o campo flag sinaliza qual o último fragmento. Percebe-se que o uso dos campos identificador e deslocamento de fragmentação apenas não indica no destino se todos os fragmentos já chegaram. A remontagem do pacote é feita apenas no host de destino.

ENDEREÇAMENTO IP

O endereço IPv4 é composto de 32 bits que são representados em 4 campos de 1 byte expressos em base decimal, por exemplo, endereço IP 200.137.162.2. A Figura 44 ilustra as representações binária e decimal.

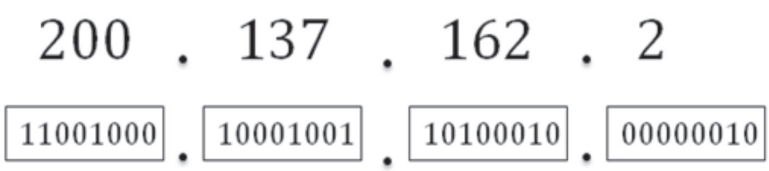


Figura 44: Exemplo de endereço IP.

O endereço IP é dividido em duas partes ou em dois sub-endereços. Endereços de sub-rede e de *host*. Uma sub-rede pode ser vista como um agregado de *host*. Por exemplo, em um laboratório de computadores de uma universidade normalmente todos os host pertencem a mesma sub-rede. A Figura 45 separa o endereço 200.137.162.2 em endereços de sub-rede e de host.

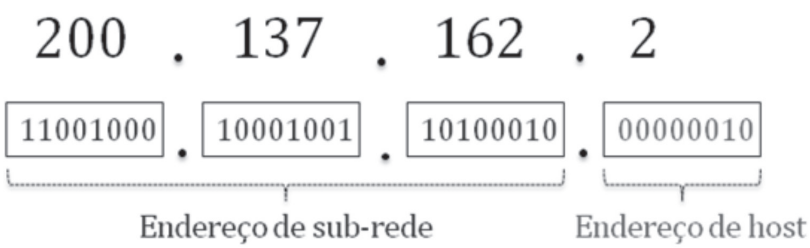


Figura 45: Sub-divisão do endereço IP em endereços de sub-rede e de *host*.

Para sinalizar essa separação, utiliza-se uma representação igual ao do endereço IP, chamada de máscara de rede. Os bits 1s expressos na

máscara de sub-rede indicam quantos bits mais significativos serão utilizados para compor o endereço de sub-rede. A Figura 4.10 ilustra a máscara de rede utilizada para separar o endereço de sub-rede do endereço de host do exemplo da Figura 50.

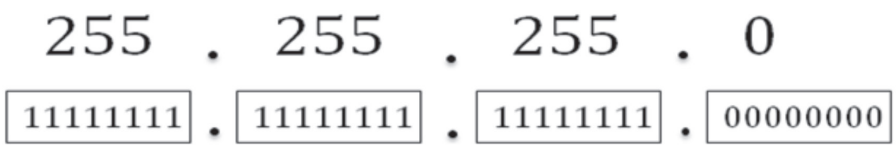


Figura 46: Máscara de sub-rede.

Observe que os 24 bits mais significativos estão ligados. Logo, a máscara ilustrada da Figura 46 separa os 24 bits mais significativos do endereço 200.137.162.2 para fazer o endereçamento da sub-rede. Essa representação também pode ser feita com a notação \24. Os outros 8 bits menos significativos são empregados para fazer o endereçamento de um *host* dentro dessa sub-rede.

Seja n o número de bits do endereço de *host*, o número máximo de *hosts* nesta sub-rede é dado por 2n – 2. A combinação de todos os bits zeros na parte de endereço de *host* é o endereço da sub-rede e todos os bits uns é o endereço de *broadcast*.

O endereço de *broadcast* é utilizado para enviar um pacote para todos os *host* da sub-rede em questão. Por exemplo, considere o endereço da sub-rede 200.137.162.x com máscara 255.255.255.0. Essa sub-rede tem 8 bits separados para endereçamento de host (n=8). O endereço 200.137.162.0 (todos os oito bits menos significativos, isto é, os bits destinados a endereço de *host* são zero) é o endereço de sub-rede. O endereço 200.137.162.255 (todos os oito *bits* menos significativos, *bits* destinados a endereço de host, são um) é o endereço de *broadcast*. Portanto do universo de 2n=8 combinações deve-se subtrair 2 endereços, os endereços da sub-rede e de *broadcast*.

O endereço IP de um *host* evidentemente deve seguir as regras definidas para a sua sub-rede. A configuração do endereço IP pode ser feita manualmente ou através do *Dynamic Host Configure Protocol* (DHCP). O DHCP é um protocolo utilizado para configura dinamicamente o endereço IP de *host*. Esse processo simplifica significamente o trabalho do administrador da rede. Para saber mais detalhes sobre o protocolo DHCP consulte o livro

do Prof. Kurose.

A Figura 47 ilustra a tela de configuração do endereço IP de um host que utiliza o sistema operacional MAC OS.

Figura 47: Visualização da configuração do endereço IP de um host.

A Figura 47 mostra que a tecnologia da camada de enlace utilizada é a Ethernet. A configuração do endereço IP é feita via DHCP. O endereço é 10.36.9.50 com máscara de sub-rede 255.255.0.0. Isto é, os 16 bits mais significativos endereçam a sub-rede. O endereço IP da placa do roteador que faz parte da sub-rede 10.36.0.1 é 10.36.255.250. O servidor DNS local possui o endereço IP 10.42.2.3. Observe que o servidor DNS local pertence a outra sub-rede.

O conceito de sub-rede é utilizado para agregar um conjunto de hosts. Assim, é possível criar regras de encaminhamento baseadas apenas nos endereços das sub-redes. Por exemplo, considere a sub-rede 10.137.162.0 (/24). Ao invés de ter uma regra para cada um dos 28 - 2 = 254 hosts dessa sub-rede, é definida nos roteadores apenas uma regra que é baseada no endereço de sub-rede. Isso é chamado endereçamento hierárquico. A mesma idéia é empregada nos serviços de correspondência comum (correios). Por

exemplo, para fazer a entrega de correspondências em Teresina o carteiro analisa primeiro o bairro. Uma vez estando no bairro ele se dirige para a rua específica. Somente depois de estar na rua ele procura pelo número da casa ou do edifício do destinatário.

De forma semelhante, no processo de envio de um pacote IP, o núcleo da rede primeiro se preocupa em encaminhar o pacote IP para a sub-rede de destino. Uma vez na sub-rede, utiliza-se o endereço do host para entregar o pacote ao host de destino.

A Figura 48 ilustra um roteador com 3 interfaces de rede. Normalmente, cada interface do roteador pertence a uma sub-rede diferente.

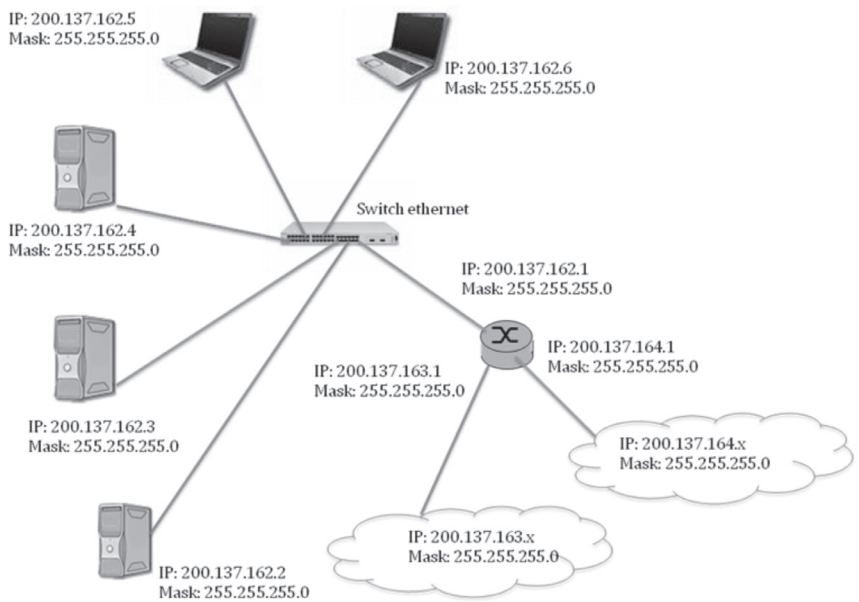


Figura 48: Exemplo de um roteador pertencendo a 3 sub-redes diferentes.

NAT (*NETWORK ADDRESS TRANSLATION*)

Já há alguns anos é comum encontrar na literatura de redes de computadores a seguinte afirmação: os endereços IPv4 estão se esgotando. Essa afirmação significa dizer que existem (ou chegará um momento em que irão existir) mais *hosts* do que endereços IP. Então, como é possível conectar um novo *host* na Internet se não existe mais endereço IP disponível?

Diante desse cenário foi proposto um artifício chamado *Network Address Translation* (NAT). O NAT consiste de um mapeamento de vários endereços IP privados em um único endereço IP público. Isso é feito com

o auxílio de uma tabela (tabela de tradução NAT) que utiliza endereços da camada de transporte para fazer o mapeamento entre diversos endereços IP privados de uma instituição em um único endereço IP público. Portanto, com o NAT, vários *hosts* (65535) podem ter acesso a Internet pendurados em um único endereço IP público.

ALGORITMOS DE ROTEAMENTO

Conforme mencionando anteriormente, uma das principais funções da camada de rede é executar os algoritmos de roteamento. Considerando um conjunto de roteadores interconectados por enlaces, um algoritmo de roteamento deve descobrir um caminho ou rota que viabilize a comunicação entre dois *hosts*. A partir das rotas identificadas pelos algoritmos de roteamento são montadas as tabelas de encaminhamento que serão utilizadas para encaminhamento de pacotes nos roteadores da Internet.

Os algoritmos de roteamento podem ser classificados em roteamento global ou descentralizado. Na classe de roteamento global os algoritmos necessitam conhecer o estado de toda a rede para calcular as rotas. Já os algoritmos descentralizados operam conhecendo apenas parte do estado da rede.

Pode-se ainda classificar os algoritmos de roteamento quanto a frequência de mudanças ou atualizações de rotas. Os algoritmos dinâmicos mudam suas rotas em função de alterações dos estados da rede. Por exemplo, um algoritmo de roteamento que considera a intensidade de tráfego dos enlaces da rede como custo. Neste caso, se houver uma sobrecarga significativa em um dos enlaces da rede o algoritmo provavelmente deve alterar as rotas que utilizam o enlace sobrecarregado.

Os principais algoritmos de roteamento da Internet são o algoritmo de vetor distância e o algoritmo baseado em estado de enlaces. Esses algoritmos são implementados, respectivamente, pelos procolos de roteamento RIP e OSPF. O algoritmo vetor distância é baseado na equação de Belman-Ford e o algoritmo de estado de enlace é baseado no algoritmo de menor caminho de *Dijkstra*.

EXERCÍCIOS

1) Cite e explique as duas principais funções da camada de rede.

2) Explique o que é, para que serve e como funciona a fragmentação e remontagem prevista no protocolo IP.

3) Por que é necessário recalcular o *header checksum* para cada pacote IP em todos os roteadores?

4) Por que a remontagem de datagramas fragmentados é feita somente no destino final?

5) Explique o funcionamento do NAT.

6) Quais as diferenças e mudanças do IPv6 em relação ao IPv4?

7) Explique com detalhes o funcionamento do Traceroute (tracert).

8) Descreva as funcionalidades dos campos do pacote IP.

9) Qual a utilidade de um algoritmo de roteamento?

10) Diferencie um *host* de um roteador.

UNIDADE 05

Camada de Enlace

Resumindo

Este capítulo apresenta conceitos ligados à camada de enlace da arquitetura de rede da Internet. Inicialmente, são elencados os serviços previstos na camada de enlace da arquitetura TCP/IP e os tipos de enlace previstos.

Posteriormente, são discutidos protocolos de acesso ao meio, protocolo ARP e o funcionamento de dispositivos de interconexão.

5 CAMADA DE ENLACE

INTRODUÇÃO

Conforme apresentado e discutido nos capítulos anteriores, a arquitetura de rede da Internet é composta basicamente de componentes de borda (hosts) e de núcleo (roteadores). Sob a óptica da camada de enlace tal separação é menos relevante, uma vez que o enlace viabiliza a interconexão dos dois tipos de componentes. Por isso, no escopo deste capítulo utilizaremos o termo **nó da rede**, ou simplesmente nó, para fazer alusão a qualquer um desses dois tipos de componentes, seja de núcleo e de borda. A Figura 49 faz essa tradução entre componente de borda e núcleo em nó da rede.

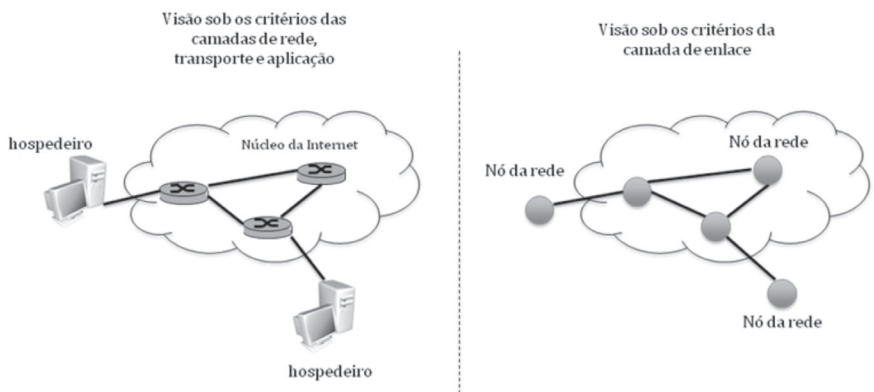


Figura 49: Visão dos componentes de borda e núcleo sob os critérios da camada de enlace.

Observa-se que o enlace viabiliza simplesmente a interconexão entre dois componentes sem qualquer distinção.

A camada de enlace (ou um enlace) da arquitetura TCP/IP tem como obrigação transportar os pacotes IP entre nós adjacentes. Um enlace pode

ser ponto a ponto (point to point) ou de difusão (broadcast). Em um enlace de broadcast muitos nós da rede fazem uso do mesmo enlace. Por isso é fundamental o uso de regras para gerenciar o acesso ao enlace compartilhado, isto é, quando cada máquina pode transmitir. Em um enlace ponto a ponto apenas dois nós são interconectados. Por exemplo, uma conexão entre dois roteadores no núcleo da Internet ou a ligação entre um usuário que utiliza uma conexão discada e o roteador do provedor de acesso. É fácil notar que o gerenciamento do acesso em um enlace ponto a ponto é bem mais simples do que o de um enlace de broadcast. A Figura 50 ilustra um enlace *broadcast* e outro ponto a ponto.

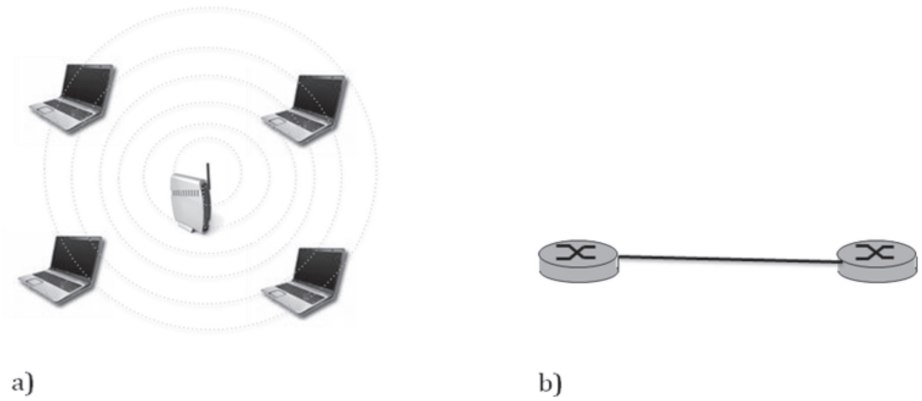


Figura 50: a) Enlace *broadcast* de uma rede sem fio. b) Enlace ponto a ponto entre dois roteadores.

Em um enlace *broadcast* deve haver uma regra para definir qual dos computadores deve transmitir em um dado instante de tempo. Vale destacar que se dois nós transmitem simultaneamente na mesma frequência haverá um colisão dos dados transmitidos. Neste caso a informação não pode ser entendida no destinatário. Uma analogia seria você conversar com um amigo em um ambiente onde existem várias outras pessoas falando alto ou gritando. Nesse ambiente as outras pessoas (mal educadas) certamente vão atrapalhar a comunicação entre você e seu amigo.

SERVIÇOS PREVISTOS NA CAMADA DE ENLACE

A unidade de transporte da camada de enlace é o quadro. Assim, no contexto da arquitetura TCP/IP, os quadros da camada de enlace são responsáveis por transportar pacotes IP entre as extremidades de um enlace. Para isso, a camada de enlace realiza de forma geral as seguintes tarefas:

- Enquadramento de dados;
- Acesso ao enlace;
- Entrega confiável;
- Controle de fluxo;
- Detecção e correção de erros e
- Transmissão *half-duplex* ou *full-duplex.*

Enquadramento de dados: A camada de enlace precisa encapsular os pacotes IP em quadros antes de transmití-los. Assim como no pacote da camada de rede, o quadro da camada de enlace possui informações de controle que compõem um cabeçalho além da parte de carga útil (aonde o pacote IP é inserido). O formato do quadro da camada de enlace é especificado no protocolo da camada de enlace.

Acesso ao enlace: As regras que especificam como um dado nó da rede deve acessar o meio físico do enlace para transmitir um quadro são definidas no protocolo de acesso ao meio. Esse conjunto de regras também conhecido como protocolo MAC (Médium Access Control protocol). Evidentemente, o protocolo MAC em enlaces ponto a ponto é trivial quando comparados com um protocolo MAC requerido em um enlace broadcast.

Entrega confiável: O serviço de entrega confiável de dados é praticamente o mesmo serviço implementado pela camada de transporte no protocolo TCP. O objetivo é garantir que um quadro transmitido por um dado nó de origem seja entregue com sucesso na outra extremidade do enlace. Esse serviço não é implementado por todas as tecnologias da camada de enlace. Tal serviço se faz necessário quando o meio de transmissão possui uma alta probabilidade de interferência o que acarreta uma alta taxa de erros, como no caso das redes sem fio IEEE 802.11 (também chamadas de redes Wi-Fi).

Controle de fluxo: Esse é outro serviço que também é implementado pelo protocolo TCP da camada de transporte. Esse serviço é fundamental para regular e controlar a taxa de envio de dados entre os nós que compõem um dado enlace. Considere por exemplo um nó de uma extremidade do enlace é capaz de transmitir e receber 10 Gbps enquanto o outro nó transmite e recebe no máximo 100 Mbps. Para o bom funcionamento do enlace nivela-se a capacidade de transmissão por baixo. Isto é, mesmo tendo uma capacidade de 10 Gbps o nó com maior capacidade de transmissão vai ser obrigado a transmitir a 100 Mbps.

Detecção e correção de erros: A detecção de erros é a capacidade do nó destinatário descobrir se algum bit do quadro transmitido foi alterado. A alteração de bits gera consequência de atenuações do sinal e de ruídos eletromagnéticos. A detecção de erros é desejada para evitar que um quadro defeituoso seja passado a frente. Isso evita desperdício da rede, uma vez que o quadro defeituoso não poderá ser interpretado corretamente no destino final. Vale ressaltar que para implementar a detecção de erros é necessário utilizar bits específicos para tal finalidade. Uma função mais elaborada neste contexto seria o enlace, pois além de detectar que bits foram alterados ele se capaz de corrigir os erros. A detecção de erros é implementada na maioria das tecnologias de enlace. Entretanto, o serviço de correção de erros é menos comum. Uma tecnologia de enlace que não implementa correção de erros apenas descarta o quadro defeituoso.

Transmissão half-duplex e full-duplex: Quando um enlace opera com transmissão full-duplex os nós das extremidades do enlace podem transmitir e receber informações ao mesmo tempo. Já na transmissão half-duplex um nó não pode transmitir e receber quadros simultaneamente.

PROTOCOLOS DE ACESSOS MÚLTIPLOS

Em enlaces do tipo *broadcast* (com vários transmissores acessando um enlace compartilhado) é indispensável um conjunto de regras para coordenar o acesso ao meio. Conforme dito anteriormente, esse conjunto de regras é chamado de protocolo MAC.

Um enlace de *broadcast* pode ser comparado a um ambiente de sala de aula. Em uma sala de aula com vários alunos e um professor é fundamental ter um conjunto de regras para impedir que mais de uma pessoa fale simultaneamente. Geralmente, quem coordena o direito de falar é o professor. Se dois ou mais alunos falam ao mesmo tempo é difícil entender a conversa ou os seus questionamentos. Normalmente um aluno levanta a mão para sinalizar que deseja falar, pelo menos deveria ser assim! Cabe ao professor conceder ou não a palavra ao aluno que solicitou. Outra questão é: quanto tempo o aluno terá para fazer o seu questionamento? Isso também deve ser gerenciado pelo professor.

Esse problema também existe nas redes de computadores e é chamado de problema do acesso múltiplo. Essa realidade está presente nos enlaces broadcast. Para coordenar os acessos utilizam-se protocolos de acessos múltiplos que podem ser classificados como:

- Protocolo de divisão de canal;
- Protocolo de acesso aleatório ou
- Protocolo de revezamento.

Protocolo de divisão de canal

Essa classe de protocolo de acessos múltiplos trabalha basicamente com a divisão do canal no domínio da frequência ou no domínio do tempo. Um termo mais apropriado para fazer alusão à divisão do canal é multiplexação. A multiplexação por divisão do tempo (*Time Division Multiplexing* – TDM) consiste em um ciclo composto de n intervalos de igual período de tempo. A Figura 51 ilustra um cenário onde quatro pares de máquinas desejam se comunicar utilizando um mesmo enlace compartilhado. Neste exemplo a máquina 1t (transmissor) deseja enviar dados para a máquina 1r (receptor); 2t deseja transmitir para 2r e assim sucessivamente.

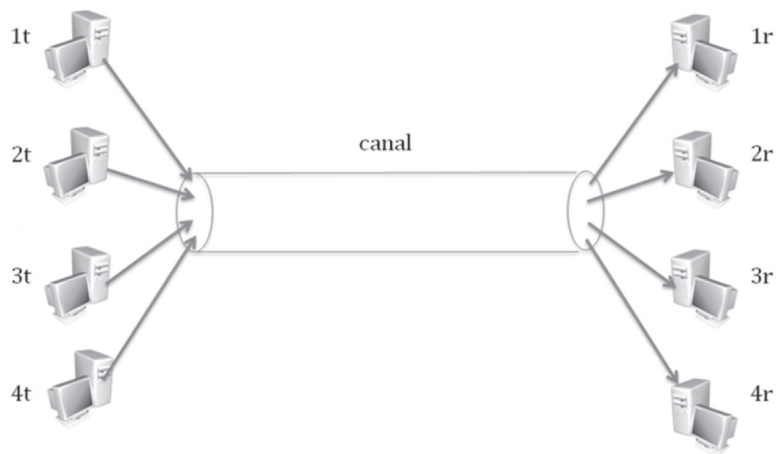


Figura 51: Problema de múltiplo acesso em um enlace *broadcast*.

Observe que se os pares de máquinas transmitirem no mesmo intervalo de tempo e na mesma frequência ocorrerão colisões. Esse termo é utilizado para denominar situações em que mais de uma máquina transmitem ao mesmo tempo e na mesma frequência em um enlace compartilhado. Quando isso ocorre pode-se dizer que os dados colidem no meio físico e isso inviabiliza a interpretação dos mesmos pelos destinatários. Quando ocorre uma colisão os dados são deformados e perdem o seu significado. A Figura 52 mostra um exemplo de enlace compartilhado que emprega multiplexação TDM. Neste exemplo considera-se que o sentido de transmissão é da

esquerda para a direita. Portanto, o enlace é compartilhado para transmissão apenas pelas máquinas 1t, 2t, 3t e 4t.

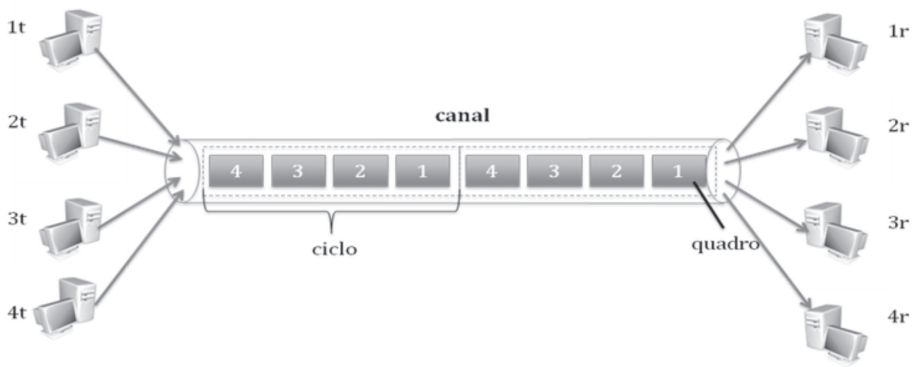


Figura 52: Protocolo de divisão de canal TDM.

Neste exemplo o tempo de transmissão é dividido em 4 partes (slots) iguais. Ciclo é o período de tempo em que todas as 4 máquinas terá um *slot* de tempo para transmitir. No intervalo de um ciclo todas as máquinas terão a oportunidade de transmitir durante um *slot* de tempo. Portanto, o número de *slots* do ciclo é igual ao número de máquinas que compartilham o enlace transmitir utilizando o enlace compartilhado. Cada máquina tem um *slot* reservado no ciclo. Se uma das máquinas não tiver algo para transmitir nenhuma outra máquina poderá transmitir durante o seu *slot* de tempo.

A Figura 53 ilustra a multiplexação por divisão de frequência. Note que, agora, todas as máquinas transmitem efetivamente em paralelo. Entretanto, cada uma delas utiliza apenas ¼ da capacidade do enlace.

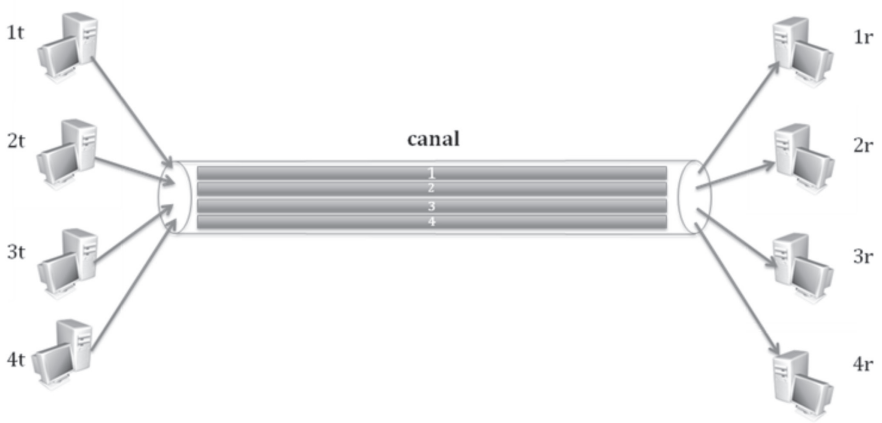


Figura 53: Protocolo de divisão de canal FDM.

A capacidade do canal é dividida em 4 intervalos de frequências independentes (1, 2, 3 e 4). Um dado par de máquinas (transmissora, receptora) precisa sintonizar em um intervalo de frequência independente. Por exemplo, as máquinas (1t, 1r) transmitem no subcanal 1, as máquinas (2t, 2r) utilizam o subcanal 2 e assim por diante. Vale destacar que, ambas as técnicas de multiplexação dividem a capacidade do canal de acordo com o número n de máquinas. No exemplo das Figura 55 e 56 onde n=4, se considerar que o canal pode transmitir R bps, cada máquina utiliza R/4 bps. Assim como na multimplexação TDM, no FDM se uma máquina ficar ociosa, a capacidade de transmissão relativa ao seu subcanal será desperdiçada.

Protocolo de acessos aleatórios

Vê-se que na classe de protocolos que operam com particionamento do canal existe uma criação de subcanais utilizando multiplexação no tempo ou na frequência. Esse procedimento impede a ocorrência de colisões. Já na classe de protocolos de acesso aleatórios cada máquina transmite utilizando toda a capacidade do canal. Nos protocolos de acessos aleatórios não existe a alocação de subcanais. Por isso, duas máquinas podem transmitir ao mesmo tempo e na mesma frequência, o que acarreta uma colisão. Protocolos de acesso aleatórios têm com objetivo evitar transmissões simultâneas e consequentemente as colisões.

Um dos protocolos clássicos da classe de aceso aleatórios é o *Carrie Sence Multiple Access* (CSMA), protocolo de múltiplos acesso com detecção de portadora. As máquinas que utilizam o CSMA possuem a capacidade de detectar se o meio de transmissão está ou não em uso. Assim, uma máquina somente transmite se detectar que o meio de transmissão está livre. Essa abordagem já diminui significativamente as colisões. Entretanto, elas ainda podem ocorrer.

Considere três máquinas **A**, **B** e **C**. Assuma que a máquina **A** começa a transmitir no instante de tempo t_0 e logo em seguida a máquina **B** deseja transmitir. Se o sinal da transmissão da máquina A ainda não tiver propagado até a máquina **B**, **B** não detecta que o meio está ocupado e por conseguinte B transmite. Neste caso haverá uma colisão.

Outra situação em que ocorre colisão mesmo com a detecção de portadora é quando coincidentemente duas máquinas começam a transmitir praticamente ao mesmo tempo. Ambas detectam que o meio está disponível e portanto iniciam a transmissão.

Frente a essas duas situações surge uma questão. O que as máquinas que se envolveram na colisão devem fazer após a colisão? Retransmitir logo em seguida o quadro que sofreu a colisão?

Após uma colisão as máquinas envolvidas precisam, evidentemente, retransmitir os seus quadros. Para minimizar futuras colisões o protocolo CSMA em cada uma das máquinas envolvidas na colisão sorteiam um número aleatório. As retransmissões devem acontecer depois desse tempo definido aleatoriamente. Por exemplo, após uma colisão de quadros das máquinas **A** e **B** elas sorteiam respectivamente os números 3 e 7. Então a máquina **A** retransmite depois de 3 segundos e a máquina **B** retransmite depois de 7 segundos. Essa medida separa as retransmissões impedindo novas colisões. Vale ressaltar que antes de retransmitir cada máquina verifica se o meio está livre.

Considere que no exemplo anterior foi sorteado em cada máquina que teve quadro colidido um número de 1 a 7. Considere agora uma outra situação em que 8 máquinas estiverem envolvidas em uma mesma colisão. Inevitavelmente, se for sorteado um número de 1 a 7 haverá uma nova colisão na retransmissão. Para reagir a situações como essa o CSMA aumenta o conjunto de possibilidade desse número aleatório em função de colisões consecutivas do mesmo quando essa reação é tomada em situações de colisões consecutivas. Nestes casos o CSMA interpreta que existe um número significativo de máquinas envolvidas. Então, para diminuir a probabilidade de uma nova colisão deve ser considerado um universo maior de números que podem ser sorteados.

O protocolo CSMA possui algumas variações, como o CSMA-CD (*Colision Detection* – CD), que possui detecção de colisão e o CSMA-CA (*Colision Avoidance* – CA), que opera evitando colisões. O CSMA-CD possui a capacidade de detectar a colisão. Com o uso do CSMA-CD as máquinas envolvidas na colisão abortam a transmissão do quadro logo após a detecção da colisão. Isso diminui o desperdício de largura de banda uma vez que o período de duração da colisão é encurtado.

O protocolo CSMA-CD é utilizado na tecnologia de Enlace Ethernet (IEEE 802.3) que será apresentada na próxima seção. O protocolo CSMA-CA é utilizado na tecnologia de enlace sem fio, Wi-Fi (IEEE 802.11). Isso porque é difícil detectar a colisão em redes sem fio.

Protocolo de revezamento

Na classe de protocolo de revezamento existe um mecanismo capaz de conceder a oportunidade de transmitir para cada máquina que compartilha o enlace. A idéia é que as máquinas que compartilham o enlace façam um espécie de revezamento para que todas tenha oportunidade de transmitir. Enquanto uma máquina transmite as outras ficam impedidas de fazê-lo. Quando uma máquina transmite ela utiliza toda a largura de banda do enlace compartilhado. Isso é feito utilizando um artifício centralizador. Um exemplo seria o gerenciamento do uso da palavra em um sala de aula, apresentado anteriormente. O professor é responsável por conceder e tomar a palavra em um ambiente de sala de aula. Os alunos quando querem usar a palavra precisam solicitar ao professor.

Dois protocolos de revezamento bem conhecidos são: protocolo de seleção e o protocolo de passagem de permissão. No protocolo se seleção um dos nós que compartilham o enlace é eleito ou designado como nó mestre. Os outros nós somente poderão transmitir no enlace sob coordenação do nó mestre. Normalmente, o nó mestre cria uma ordem circular entre os outros nós. Inicialmente o nó mestre envia uma mensagem para o nó 1 dizendo que ele pode transmitir um determinado numero de quadros. Em seguida, o nó mestre faz o mesmo para o nó 2 e assim por diante. No protocolo de seleção o revezamento é conduzido com o auxílio de um nó mestre que determina a ordem do revezamento.

O outro protocolo de revezamento mencionado é o de passagem de permissão. Neste protocolo o revezamento é feito com a utilização de uma ficha (também chamada de token). Cada máquina que compartilha o enlace somente poderá transmitir quadros se estiver de posse da ficha. A ficha precisa ser repassada de forma cíclica para todas máquinas que pertencem ao enlace. Quando uma máquina recebe a ficha ele tem o direito de transmitir um numero máximo de quadros. Feito isso ela deve encaminhar a ficha para a próxima máquina. Como existe somente uma ficha, nunca duas máquinas poderão transmitir ao mesmo tempo. Logo, essa solução não gera colisão de dados. A deficiência desses tipos de protocolos (revezamento) é a sua vulnerabilidade mediante as falhas. Se a máquina que está transmitindo falhar o enlace para, pois a máquina que falhou não terá como repassar a ficha para as outras máquinas.

ETHERNET

A tecnologia de enlace (com fio) mais difundida no mundo no contexto de redes locais é certamente a Ethernet (IEEE 802.3). A Ethernet foi inventada em meados da década de 70 por Bob Metcalfe e David Boggs. Eles criaram a empresa 3Com em 1979. Em 1990 ele deixou a 3Com que tinha nessa época receita de 400 milhões de dólares.

Inicialmente, a Ethernet usava um barramento para interconectar até 256 máquinas e operava a 2,94 Mbps. Posteriormente, a Ethernet evoluiu para 10 Mbps, 100 Mbps (Fast Ethernet). Mais recentemente a Ethernet sofreu outra evolução operando a 1 Gbps (Gigabit Ethernet) e 10 Gbps (10 Gigabit Ethernet). A Ethernet utiliza o protocolo CSMA-CD para fazer acesso ao meio.

O quadro Ethernet é ilustrado na Figura 54.

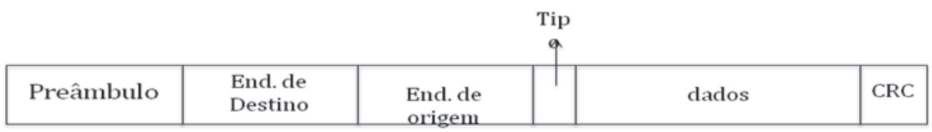


Figura 54: Formato do quadro Ethernet.

O campo **preâmbulo** é composto de 8 bytes e tem a função de sincronizar o relógio do remetente com o relógio da placa de rede do destinatário. Ele utiliza os primeiros 7 bytes no formato 10101010 e o último byte no formato 10101011. O preâmbulo pode ser visto como um mecanismo para “acordar” a placa de rede. Seria algo como “fica atenta pois estão chegando dados”.

Os campos **endereço de destino** e de **origem** contêm 6 bytes cada um. Eles são utilizados para identificar cada placa de rede individualmente. Similar ao que ocorre na camada de rede, é necessário um endereço da camada de enlace para saber, no nível da camada de enlace, qual máquina está enviando e para qual máquina. Esses endereços são chamados de endereços MAC e seguem a notação hexadecimal.

O **tipo** é um campo que utiliza 2 bytes e serve para indicar qual protocolo está sendo transportado pelo quadro Ethernet. Certamente, na maioria das vezes o quadro Ethernet transporta um pacote IP. Entretanto, outros protocolos podem ser transportados, por exemplo, Appletalk ou Novell IPX. O protocolo Ethernet também deve ser capaz de transportar mensagens

e respostas de protocolo ARP. Esse protocolo é responsável por fazer uma tradução de endereço IP em endereço MAC. Mais a frente serão apresentados detalhes sobre o protocolo ARP e o seu funcionamento.

Dados é um campo que vai efetivamente transportar o pacote IP entre as extremidades do enlace. O tamanho máximo de campo de dados (Maximum Transmition Unit - MTU) do Ethernet é de 1500 Bytes. Portanto, pacotes IP não podem ter mais do que 1500 bytes se forem ser transmitidos pela tecnologia de enlace Ethernet.

ENDEREÇAMENTO MAC

De acordo com o exposto na seção anterior, vê-se que existe um endereço específico para a camada de rede e outro para a camada de enlace. É natural o surgimento da pergunta: por que dois endereços? Não poderia ser utilizado apenas um endereço para as duas camadas?

O endereço MAC é o endereço do adaptador de rede, isto é, da placa de rede. Na maioria das tecnologias de redes locais o endereço MAC possui 6 bytes e é expresso em notação hexadecimal (por exemplo, 00-0f-ea-9a-cf-c4). Esse espaço de endereçamento viabiliza 248 combinações diferentes de endereços MAC. O seu objetivo é identificar unicamente os adaptadores de rede. Para isso, os endereços MAC são gerenciados pelo IEEE com o objetivo de impedir que duas placas de rede tenham o mesmo endereço MAC. Todo fabricante de placas precisa comprar faixas de endereçamentos MAC do IEEE. Portanto, pelo menos em tese, não existem duas placas no mundo como o mesmo endereço MAC.

Os endereços IP possui uma estrutura hierárquica. Se um dado pacote tiver como destino o endereço IP A, os roteadores do núcleo da Internet devem encaminhá-los para a região geográfica da sua sub-rede. De forma simplista, pode-se dizer que em um endereço IP tem uma “interpretação geográfica”. Isso porque todos os pacotes que possuírem o endereço IP A como destino serão encaminhados para mesma região que é a região da sub-rede do endereço IP A. Por exemplo, qualquer host do planeta terra que enviar um pacote IP com o endereço de destino 200.137.162.2 será encaminhado para o servidor WEB da UFPI que fica em Teresina – PI.

O endereçamento MAC, da camada de enlace, não é hierárquico como o endereçamento IP e sim linear. Considere um aluno de computação utiliza o seu laptop inicialmente na rede da UFPI. Assume-se que esse host

utiliza um único endereçamento, diferente do que ocorre na Internet que opera com endereço MAC e endereço IP. Posteriormente esse aluno se dirige para o aeroporto de Teresina. Chegando no aeroporto ele liga o seu laptop para acessar seu e-mail. Neste exemplo tem-se um mesmo computador, como a mesma placa de rede, que se conecta a Internet através de duas redes diferentes em momentos diferentes. Nota-se que se fosse possível o usuário utilizar o endereço IP da rede da UFPI no aeroporto para solicitar páginas WEB suas páginas seriam enviadas para a UFPI e não para o aeroporto. Isso porque o endereço IP é utilizado para seguir um caminho através dos roteadores de núcleo da Internet em direção a um determinado host. Note que neste exemplo não seria possível utilizar um único endereço para fazer endereçamento da camada de rede/enlace.

Considere agora o inverso, isto é, o uso apenas do endereço MAC. Imagine que a placa de rede do servidor WEB da UFPI queima após a queda de um raio no período das chuvas de verão. Ao trocar a placa de rede do servidor o endereço MAC passa a ser outro, uma vez que cada placa de rede possui um endereço MAC único. Se apenas o endereço MAC fosse utilizado seria necessário reprogramar os roteadores do núcleo da Internet para que eles fiquem ciente de que os pacotes com destino ao servidor WEB da UFPI agora tem outro endereço. Somente após tal reprogramação eles poderiam encaminhar os dados baseados no novo endereço MAC. Mais uma vez nota-se que isso não é viável.

Diante dos exemplos expostos, fica mais evidente a necessidade de trabalhar com endereçamento da camada de enlace e endereçamento da camada de rede.

PROTOCOLO ARP

O protocolo ARP (*Address Resolution Protocol*) tem a função de fazer o mapeamento entre endereço IP e endereço MAC. A nível da camada de enlace os quadro precisam ser endereçado utilizando os endereços MAC. Portanto, quando um nó A deseja enviar um quadro para um nó B o nó A precisa saber qual o endereço MAC de B.

Considera-se dois nós A e B que pertencem a mesma sub-rede IP. O nó A deseja enviar um pacote IP para o nó B. Para isso o pacote IP deve ser encapsulado em um quadro da tecnologia de enlace que por vez, deve conter como destino o endereço MAC do nó B. O problema é que a princípio

o nó A não sabe o endereço MAC do nó B. Esse problema é resolvido pelo protocolo ARP. Neste caso, o nó A vai mandar um *query* ARP em broadcast para todos os nós que pertencem sua sub-rede. Esta *query* ARP equivale a seguinte pergunta: “qual o endereço MAC da placa que está associada com o endereço IP B? Como essa pergunta é enviada em broadcast, todos os nós que pertencem a sub-rede vão recebê-la, inclusive o nó B. Essa pergunta (*query* ARP) é enviada dentro de um quadro da camada de enlace contendo como o endereço MAC de A como no campo de endereço de origem. Quando o nó B recebe a query ARP ele aprende o endereço MAC do nó A. Em seguida ele responde diretamente para o nó A o seu endereço MAC. Depois desse processo o nó A aprende qual o endereço MAC deve ser utilizado para enviar um quadro o nó B.

Os endereços MAC que um nó aprende são mantidos temporariamente em uma tabela chamada de tabela ARP. Depois de um período de tempo essa informações expiram e são apagadas da tabela.

A Figura 55 ilustra uma tabela ARP que faz mapeamento entre endereços IP e endereço MAC. Para visualizar a tabela ARP em um *host* utilize o comando “arp -a”.

A Figura 55 mostra dois mapeamentos entre endereços IP e endereços MAC. O endereço IP 192.168.0.176 está associado ao endereço MAC 00-0f-ea-9a-cf-c4. Já o endereço 192.168.0.1 está ligado ao endereço MAC 00-15-e9-e2-0e-25.

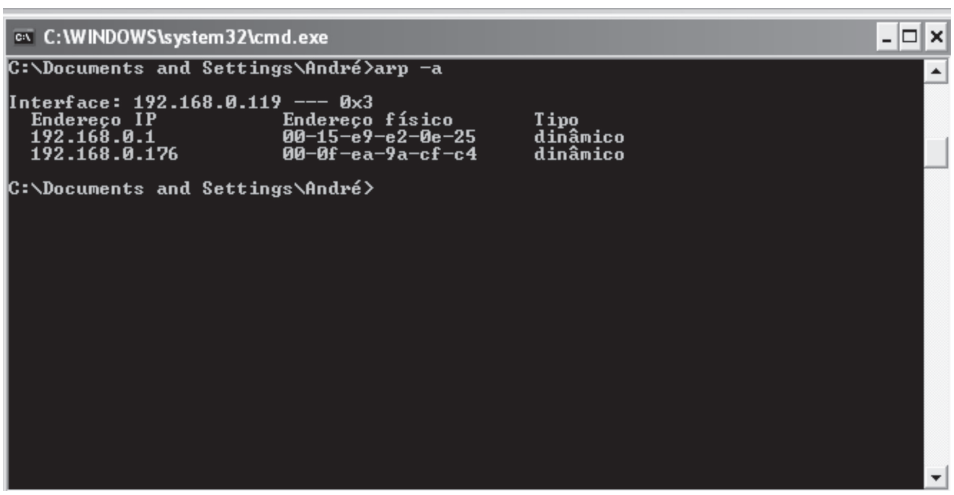


Figura 55: Exemplo de tabela ARP.

DISPOSITIVOS DE INTERCONEXÃO

Atualmente existem basicamente três tipos de dispositivos de interconexão: roteadores, comutadores (switches) e hubs.

Os roteadores são comutadores de camada 3 e já foram apresentados no contexto do Capítulo 4. Seu objetivo é comutar pacotes de rede realizando o encaminhamento de pacotes.

Hubs são repetidores de sinal que interconectam máquinas diretamente a um barramento compartilhado. Esse dispositivo era muito empregado no início da tecnologia Ethernet, como podemos ver na figura abaixo:

Figura 56: Ilustração da interconexão interna de um hub.

A Figura 56 ilustra a interconexão interna de um hub. Entretanto, um hub é uma caixa preta com porta para conexão de cabos. Ele visualmente é similar a um roteador e a um *switch*. Apesar dessa aparente similaridade eles são bem diferentes. O hub nada mais é do que um barramento onde todas as máquinas ficam conectadas diretamente. Portanto, quando uma máquina transmite o sinal é propagado para todas as outras máquinas. Por isso, diz-se que um hub coloca todas as máquinas ligadas a ele em um mesmo domínio de colisão. Sempre que duas máquinas ligadas ao hub transmitirem ao mesmo tempo ocorrerá um colisão.

Os *Switchs* são comutadores de camada 2, comutadores de quadros da camada de enlace. Esse dispositivo é essencial em redes Ethernet. Conforme apresentando anteriormente o Ethernet utiliza um protocolo aleatório de acesso ao meio, o CSMA-CD. Diferente do hub o switch apresenta um comportamento inteligente. Ele é capaz de aprender e mapear os endereços MAC a cada uma de suas portas.

Considere um cenário (Figura 57) com um switch que possui 4 portas. Em cada uma dessas portas está uma máquina. Na porta 1 do switch está conectada a máquina com endereço MAC A, na porta 2 a máquina com endereço MAC B e assim por diante.

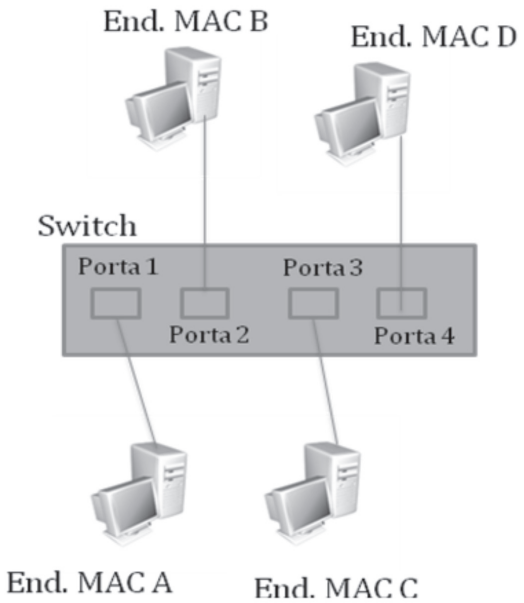


Figura 57: Exemplo da interconexão de um switch.

Depois que o *switch* é ligado ele aprende qual endereço MAC está associado a qual porta. Assim, quando um máquina transmite um quadro com um dado endereço MAC de destino o *switch* envia o quadro apenas para a porta específica. Esse comportamento isola as máquinas em diferentes domínios de colisão. Com essa flexibilidade o *switch* diminui significativamente o número de colisões da rede, aumentando o seu desempenho. Algo que não ocorre quando um hub é utilizando como dispositivo de interconexão. Vale destacar que o *switch* implementa o protocolo CSMA-CD em cada uma de suas portas. Então, um quando somente será transmitido por uma porta do *switch* se o meio estiver livre. Se ainda assim houver uma colisão o *switch* segue o comportamento d o protocolo CSMA-CD já apresentado.

EXERCÍCIOS

1) Cite as principais funções da camada de enlace.

2) Explique o que é multiplexação no contexto da camada de enlace.

3) Explique qual a finalidade e o funcionamento do protocolo ARP.

4) Cite e explique as classes de protocolos de acesso ao meio.

5) Diferencie TDM do FDM.

6) Descreva o funcionamento do protocolo CSMA-CD.

7) Diferencie os seguintes dispositivos de interconexão: hub, switch e roteador.

8) O que é um domínio de colisão em uma rede Ethernet

9) Explique como um *switch* fragmenta suas portas em diferentes domínios de colisão.

10) Descreva o propósito dos campos do quadro Ethernet.

UNIDADE 06

Redes de Alto Desempenho: Circuitos Ópticos

Resumindo

Este capítulo apresenta um exemplo de tecnologia de rede de alto desempenho, redes de circuitos ópticos transparentes. Essa nova tecnologia de rede é baseada em uma infraestrutrura óptica com grande largura de banda e visa atender a crescente demanda de tráfego da Internet.

Além de ter uma maior capacidade de transmissão, a tecnologia de circuitos ópticos é capaz de oferecer um serviço com garantia de qualidade de serviço. Isso porque ela emprega o paradigma de comutação de circuito.

Em especial, este capítulo descreve o principal problema das redes ópticas transparentes, o problema do roteamento e alocação de comprimento de onda e alternativas para resolvê-lo.

6 REDES DE ALTO DESEMPENHO: CIRCUITOS ÓPTICOS

INTRODUÇÃO

O crescimento do número de usuários da Internet e o surgimento de novas aplicações envolvendo voz e vídeo, como vídeo sob demanda, teleconferência, imagens médicas de alta resolução etc., têm provocado um aumento considerável da demanda de banda passante nas redes de transporte que constituem os *backbones* dos provedores de serviços de telecomunicações.

A Figura 58 mostra a taxa de crescimento anual do tráfego internacional de Internet (valores médio e máximo) como também a taxa de crescimento da largura de banda da Internet entre 2005 e 2008.

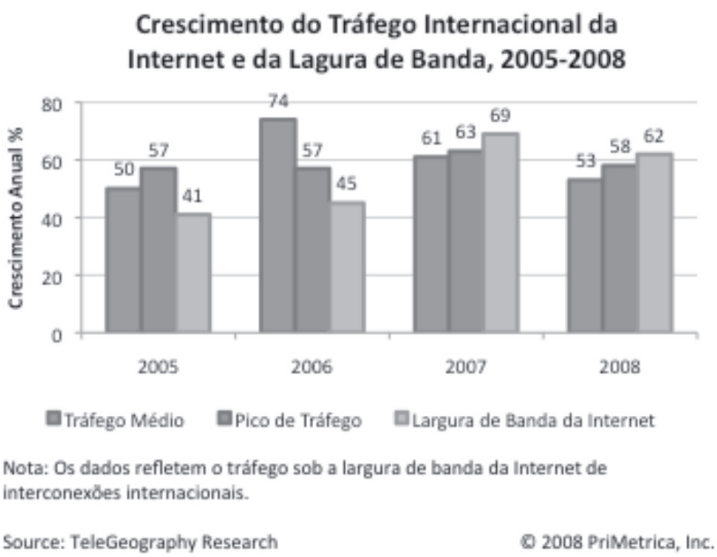


Figura 58 - Taxa de crescimento do tráfego internacional da Internet (valores médio e máximo) e da largura de banda entre 2005 e 2008).

Com a análise do tráfego médio, por exemplo, percebe-se um crescimento de 74% no período de 2006. Mesmo no período de menor crescimento (ano de 2005) identifica-se um aumento relevante de 50% no tráfego médio. Estes números evidenciam um crescimento significativo da demanda por largura de banda no âmbito da Internet nos últimos anos.

Recentemente surgiu uma nova geração de aplicações voltadas ao aproveitamento dos recursos globais de computação, armazenamento e de rede, para viabilizar o desenvolvimento de pesquisas colaborativas de forma global a grandes distâncias.

O termo *e-science* tem sido aplicado para referenciar aplicações desta natureza voltadas a estudos científicos que dependem da utilização de recursos computacionais de alto desempenho e geram grandes quantidades de dados, por exemplo, aplicações de física nuclear e de alta energia, astrofísica, energia de fusão, bioinformática etc. Em geral, tais aplicações necessitam transferir grandes volumes de dados sob rigorosos requisitos de Qualidade de Serviço (*Quality of Service* – QoS) das redes envolvidas. Exemplos de projetos de e-science desenvolvidos em diferentes países são: *Exploitation Switching Lightpath for E-Science Application* – ESLEA, *Global Lambda Integrated Facility* - GLIF, HOPI - *Hybrid Optical and Packet Infrastructure Project*.

Com o objetivo de atender essa crescente demanda de tráfego com garantia de QoS tem sido desenvolvida uma nova geração de redes de transporte de telecomunicações baseada em uma infraestrutura óptica mais inteligente.

A infraestrutura óptica é justificada por características das fibras ópticas como grande largura de banda, baixa perda, imunidade a ruídos e a interferências eletromagnéticas, entre outras. Nesse contexto, espera-se uma maior capacidade dos elementos da rede óptica em termos de processamento de protocolos de controle com o objetivo de permitir flexibilidade e agilidade na provisão de serviços para atender a essa nova realidade dos usuários.

A multiplexação por divisão de comprimentos de onda (*Wavelength Division Multiplexing* - WDM) é uma tecnologia que utiliza de forma mais eficiente a banda passante das fibras ópticas. Vale lembrar que a tecnologia WDM é um tipo de multiplexação FDM, discutida no Capítulo 5.

Em uma única fibra óptica são estabelecidos, simultaneamente, múltiplos canais ópticos que operam em diferentes comprimentos de onda. Cada comprimento de onda pode atingir atualmente taxas de transmissão da ordem de 40 Gbps com equipamentos disponíveis comercialmente.

Atualmente, existem basicamente duas variações de sistemas WDM: multiplexação densa (*Dense Wavelength Division Multiplexing* - DWDM) ou multiplexação esparsa (*Coase Wavelength Division Multiplexing* - CWDM) de comprimento de onda. A diferença básica entre as variações DWDM e CWDM é a densidade de comprimentos de onda multiplexados em uma única fibra.

Os sistemas DWDM são utilizados principalmente em redes WAN (*Wide Area Network*) de alta capacidade enquanto que os sistemas CWDM, mais simples e baratos, são usados tipicamente em redes MAN.

As redes ópticas WDM podem ser classificadas em opacas ou transparentes. As redes ópticas opacas realizam o roteamento de comprimentos de onda no domínio eletrônico. Neste tipo de rede óptica são necessários conversores Opto-Eletro-Óptico - OEO responsáveis por converter o sinal óptico em sinal elétrico e vice-versa em cada nó da rede. Conversores OEO têm o inconveniente de inserir atrasos de processamento, além de aumentar significativamente o custo dos equipamentos. Nas redes ópticas transparentes, o roteamento de comprimentos de onda é realizado no domínio óptico, eliminando a necessidade de conversores OEO e suas limitações.

Em uma rede óptica transparente o sinal óptico é transmitido ao longo de nós intermediários sem a realização de conversão para o domínio eletrônico. Desta forma, o custo associado de uma comutação de alta velocidade em meio eletrônico é eliminado. Diferentes tecnologias de comutação foram desenvolvidas com o objetivo de viabilizar o uso de redes ópticas WDM sem a necessidade de se realizar o processamento eletrônico intermediário.

As alternativas existentes atualmente para a comutação em redes ópticas transparentes são:

- Comutação de circuitos ópticos (*Optical Circuit Switching* - OCS);
- Comutação de pacotes ópticos (*Optical Packet Switching* - OPS) ;
- Comutação de rajadas ópticas (*Optical Burst Switching* - OBS).

A tecnologia OCS é caracterizada pela reserva de recursos (comprimentos de onda) e pela configuração das matrizes de comutação, dos nós envolvidos, na fase de estabelecimento do circuito óptico (também chamados de lightpath). Apesar da ineficiência em comunicações de curta duração, a comutação OCS permite a reserva de recursos com garantia de QoS para comunicações no circuito óptico.

A tecnologia OPS pode ser considerada uma alternativa mais eficiente quando o tráfego for caracterizado por um alto dinamismo e pela multiplexação estatística de pacotes de dados com tamanho variável. Entretanto, na comutação OPS não existe uma reserva de recursos e, consequentemente, é difícil garantir qualidade de serviço.

Nesse tipo de comutação, os dados e as informações de controle são enviados tipicamente na mesma banda (cabeçalho + carga útil), havendo a necessidade de armazenar o pacote óptico e processar as informações do cabeçalho. Somente depois disso o pacote óptico pode ser encaminhado pelos nós intermediários. Em função do avanço ainda limitado nas áreas de processamento e armazenamento óptico essa tecnologia de comutação óptica ainda não está suficientemente madura.

A tecnologia de comutação OBS trabalha com o envio de rajadas ópticas, uma espécie de container que agrupa pacotes de dados a serem encaminhados para um mesmo destino. Na comutação OBS um pacote de controle é enviado antes da rajada óptica com a intenção de reservar a banda necessária e configurar as matrizes de comutação ao longo do caminho.

Tipicamente, a rajada óptica é enviada sem a confirmação da reserva dos recursos. Isto significa que sempre a rajada óptica será enviada sem a garantia do sucesso no seu encaminhamento. A comutação de rajadas ópticas pode ser uma alternativa para a transmissão de dados através de uma rede óptica transparente com o objetivo de prover uma infraestrutura de transporte flexível para o encaminhamento de pequenos volumes de tráfego quando os requisitos de QoS são baixos.

Uma variação das tecnologias acima citadas é o uso de arquiteturas híbridas que permitem em uma mesma rede a comutação de circuitos ou de rajadas ópticas. Neste tipo de arquitetura de comutação óptica é necessária a identificação prévia da característica do tráfego para decidir qual técnica de comutação deve ser utilizada.

Dentre essas alternativas, a tecnologia de comutação óptica mais amadurecida atualmente é a comutação de circuitos ópticos. Sua capacidade de reservar recursos é uma característica que posiciona a tecnologia OCS em vantagem quando o objetivo é garantir QoS.

ROTEAMENTO E ALOCAÇÃO DE COMPRIMENTO DE ONDA

Uma rede óptica transparente é formada basicamente por nós ópticos e enlaces de fibras ópticas. Os nós ópticos são interconectados por enlaces

de fibras ópticas formando uma topologia física que é utilizada para prover o estabelecimento de circuitos ópticos, também chamados de *lightpaths*. A Figura 59 a ilustra a rede óptica transparente com comutação de circuitos.

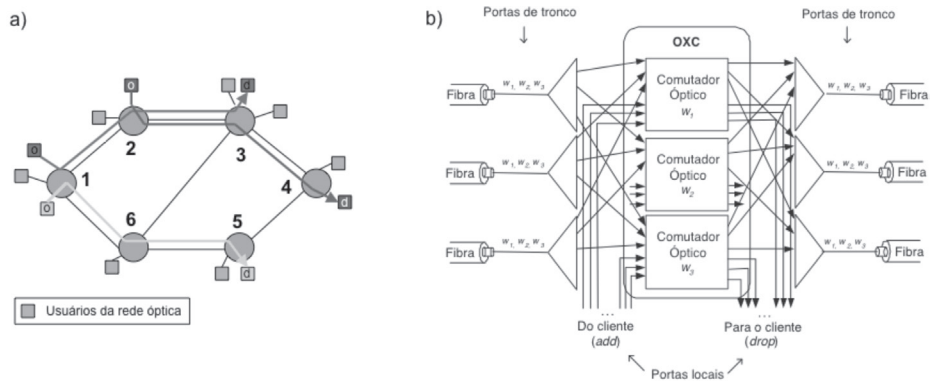


Figura 59: a) Rede Óptica transparente. b) Nó óptico e seus principais componentes.

Um circuito óptico pode ser visto como a utilização efetiva de um ou alguns comprimentos de onda em um conjunto de enlaces, provendo uma conexão de um nó de origem para um nó de destino. Esse par de nós (origem, destino) utiliza a rede óptica para transmissão e recepção de dados. Os circuitos ópticos são simples canais de grande largura de banda que transportam dados em altas taxas de transmissão (por exemplo 40 Gbps).

Um nó da rede óptica é composto basicamente por portas de tronco, portas locais, multiplexadores/demultiplexadores e um OXC, como ilustra a Figura 62 b.

Um nó óptico pode iniciar, finalizar ou ser intermediário de um circuito óptico. As portas locais são utilizadas para que os usuários da rede óptica possam originar (add) ou ser o destinatário (drop) de um circuito óptico. Um nó óptico utiliza as portas locais somente quando for origem ou destino de um circuito óptico.

As portas de tronco fazem a interconexão entre nós adjacentes de acordo com a topologia da rede. Essas portas são utilizadas para que o sinal óptico possa sair de um determinado nó e chegar a um outro nó adjacente e vice-versa. Os comprimentos de onda multiplexados que chegam por uma porta de entrada precisam ser separados espacialmente para que possam ser comutados de forma independente (Fig. 62 b), uma vez que eles podem seguir rotas diferentes. Da forma inversa, após a realização da comutação, os comprimentos de onda que compartilham uma mesma saída precisam ser novamente multiplexados em uma única fibra óptica.

O OXC é um elemento de rede óptica que tem a função de comutar um comprimento de onda de uma porta de entrada para uma outra porta de saída. Esta função é fundamental para definir o caminho/rota a ser utilizada por um circuito óptico. OXCs de nós intermediários de um circuito óptico realizam a comutação entre as portas de tronco. OXCs de nós de origem ou de destino de um circuito óptico realizam comutação entre portas locais e portas de tronco para que um usuário da rede óptica envie (add) ou receba (drop) informações através dos circuitos ópticos.

Cada enlace óptico pode suportar um número específico de comprimentos de onda. Um padrão usual definido atualmente pela ITU-T (ITU-T Recommendation G.698.1) especifica um máximo de 40 comprimentos de onda para cada fibra.

Para se estabelecer um circuito óptico em uma rede óptica transparente é necessário escolher uma rota e um comprimento de onda que ligue o nó de origem ao nó de destino. Este problema é conhecido como *Routing and Wavelength Assigment* - (RWA). Rotear e alocar comprimentos de onda são fatores importantes na tentativa de otimizar a utilização dos recursos das redes ópticas transparentes.

O tráfego em uma rede óptica transparente pode ser classificado em estático ou dinâmico. No caso de tráfego estático os pares de nós origem e destino são conhecidos previamente. Neste caso, os recursos necessários (comprimentos de onda) para se estabelecer cada circuito óptico são alocados *off-line* numa fase de planejamento da rede óptica.

No início da operação da rede todos os circuitos ópticos já estão estabelecidos e assim permanecem. Por outro lado, no caso de tráfego dinâmico, os circuitos ópticos são estabelecidos e finalizados dinamicamente, de acordo com as requisições dos usuários da rede óptica. Tal dinamismo impossibilita o conhecimento de quais e quantos circuitos serão requisitados em um determinado instante de tempo. Logo, a demanda de circuitos ópticos pode ser momentaneamente superior à capacidade da rede óptica em termos de enlaces ópticos (fibras e comprimentos de onda).

Uma requisição de circuito óptico pode ser atendida somente se existirem recursos disponíveis na rede. No processo de estabelecimento de um circuito óptico, um comprimento de onda é alocado e deve permanecer ocupado durante o período de tempo que o circuito estiver ativo. Devido ao número limitado de comprimentos de onda, alguns circuitos ópticos podem não ser estabelecidos, gerando o bloqueio por ausência de comprimentos de

onda livre.

O problema RWA pode ser classificado em RWA estático ou RWA dinâmico segundo o tipo de tráfego. O objetivo do RWA estático é minimizar os recursos (comprimentos de onda) necessários para atender a um conjunto finito de circuitos ópticos conhecidos previamente. Na verdade, o problema RWA estático é um problema de dimensionamento de recursos em redes ópticas com conexões estabelecidas de maneira estática. Já no problema RWA dinâmico, o objetivo é rotear e alocar comprimentos de onda minimizando a probabilidade de bloqueio de futuras conexões, dado um conjunto finito de comprimentos de onda disponíveis.

No RWA dinâmico os algoritmos devem apresentar baixa complexidade computacional, uma vez que as escolhas das rotas e dos comprimentos de onda são feitas em tempo de execução.

O problema RWA estático pode ser formulado como um problema de programação linear inteira mista que é um problema NP-completo. O Problema RWA dinâmico é ainda mais difícil de ser resolvido, e por isso geralmente são utilizados métodos heurísticos caberia uma nota de rodapé esclarecedora. Por motivos de simplificação, o problema RWA é geralmente dividido em dois subproblemas, o problema de roteamento e o problema de alocação de comprimento de onda.

A Figura 60 ilustra cada um dos subproblemas. Nesse exemplo é feita uma requisição de circuito óptico do Nó 1 para o Nó 3.

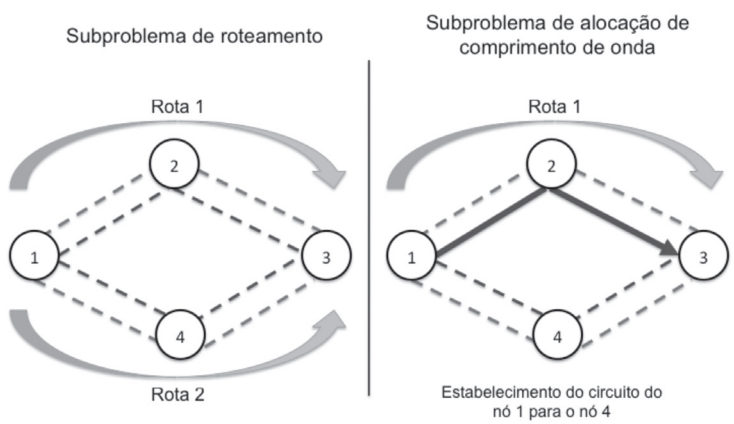


Figura 60: Exemplo ilustrando o Problema RWA dinâmico.

Para o estabelecimento do circuito é necessário, inicialmente, decidir qual rota deve ser utilizada. No exemplo da Figura 60, existem duas alternativas de rota (Nó 1, Nó 2 e Nó 3 ou Nó 1, Nó 4 e Nó 3). A escolha da rota é feita por um algoritmo de roteamento. Após escolher a rota, o próximo

passo é alocar um comprimento de onda dentre os comprimentos de onda livres (vermelho e azul) na rota escolhida. A escolha do comprimento de onda é função do algoritmo de alocação de comprimento de onda.

No exemplo da Figura 60 , a Rota 1 (Nó 1, Nó 2 e Nó 3) e o comprimento de onda azul são escolhidos para o estabelecimento do circuito em questão, viabilizando a conexão do Nó 1 para o Nó 3.

Uma conexão representa logicamente a comunicação entre dois nós da rede óptica. Tal conexão é de fato viabilizada por um circuito óptico (ligthpath) que interconecta fisicamente esse par de nós. Ao ser viabilizada uma conexão entre um par de nós origem e destino é também estabelecido um circuito óptico entre esses nós.

ROTEAMENTO DE COMPRIMENTO DE ONDA

O problema de roteamento de comprimento de onda é atualmente tratado com as seguintes classes: roteamento fixo, roteamento alternativo e o roteamento exaustivo.

Para estabelecer um circuito óptico utilizando uma rota específica entre um par de nós origem e destino é necessário o uso de protocolos de sinalização para (i) obter informações sobre quais os comprimentos de onda disponíveis por enlace, (ii) quais os comprimentos de onda disponíveis em todos os enlaces da rota e (iii) reservar um comprimento de onda na rota especificada. Estes protocolos de sinalização introduzem sobrecargas de comunicação e atrasos no estabelecimento do circuito óptico.

No roteamento fixo, cada (o,d) possui apenas uma rota fixa que é definida previamente. Isto significa menos sobrecarga de comunicação e menos atraso no estabelecimento de um circuito óptico quando comparada com as outras classes de roteamento.

A Figura 61 ilustra uma rota fixa 1, 6, 5 definida para atender a conexão C(1,5). Se surgir uma requisição para a conexão C(1,5) e os recursos da rota 1, 6, 5 estiverem ocupados, o estabelecimento da C(1,5) será bloqueado, pois neste tipo de roteamento não são utilizadas rotas alternativas. O roteamento fixo é muito simples de ser implementado, mas possui desvantagens como intolerância à falhas e geralmente apresenta maior probabilidade de bloqueio quando comparado com as outras classes de roteamento que consideram o uso de rotas alternativas.

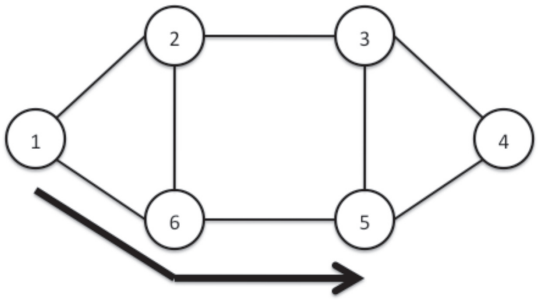
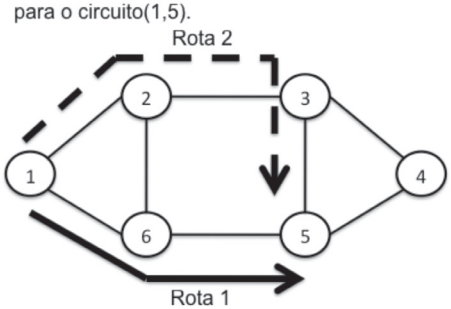


Figura 61 Exemplo de roteamento fixo.

A classe de roteamento alternativo é caracterizada pela existência de um conjunto fixo de rotas definidas previamente para cada (o,d). O roteamento alternativo pode ainda ser subdividido em duas categorias, roteamento fixo alternativo e roteamento dinâmico alternativo. No roteamento fixo alternativo, cada nó da rede possui uma tabela com as rotas predefinidas e ordenadas em função do custo (segundo algumas métricas) para alcançar cada possível destino. Na tentativa de atender uma requisição de conexão óptica, a lista de rotas pré-selecionadas é percorrida na ordem crescente de custo com o objetivo de identificar a rota de menor custo da lista que tenha a capacidade de estabelecer o circuito óptico requisitado. Se nenhuma das rotas pré-definidas tiver recursos disponíveis, o circuito óptico será bloqueado. A rota de menor custo para um dado (o,d) é chamada de rota primária e as outras rotas para esse mesmo par são chamadas de rotas alternativas.

A Figura 62 a) ilustra um exemplo de roteamento fixo alternativo com 2 rotas (rota primária + 1 rota alternativa) no atendimento da C(1,5).

a) Topologia e rotas altarnativas para o circuito(1,5).



b) Tabela de rotas alternativas do nó 1 para todos os destinos.

Destino	Rota 1	Rota 2
2	1 → 2	1 → 6 → 2
3	1 → 2 → 3	1 → 6 → 5 → 3
4	1 → 2 → 3 → 4	1 → 6 → 5 → 4
5	1 → 6 → 5	1 → 2 → 3 → 5
6	1 → 6	1 → 2 → 6

Figura 62 Exemplo de roteamento fixo alternativo.

Considerando o custo da rota como o número de saltos, a rota primária é 1, 6, 5. Se a rota primária não possuir recursos disponíveis a rota alternativa 1, 2, 3, 5 será utilizada. O roteamento fixo alternativo possui um certo nível

de tolerância a falhas sendo mais flexível do que o roteamento fixo. Observa-se que essa flexibilidade depende da topologia da rede e do número de rotas alternativas para cada (o,d). As rotas podem ser computadas utilizando um algoritmo de menor caminho como o algoritmo de menor caminho de Dijkstra ou o de Bellman-Ford.

O roteamento dinâmico alternativo também opera com um conjunto de rotas pré-definidas para cada (o,d). Entretanto, antes de tentar estabelecer o circuito óptico utilizando uma dessas alternativas, são coletadas informações sobre a utilização dos comprimentos de onda dos enlaces que pertencem à lista de rotas pré-definidas. Assim, com base no atual estado da rede e de acordo com uma função objetivo definida, uma rota será selecionada dentre o conjunto de rotas pré-definidas. Um exemplo de roteamento dinâmico alternativo é o LCP, roteamento pelo caminho menos sobrecarregado

Nesse tipo de roteamento o custo de um enlace equivale à medida do seu congestionamento, dada pelo número de comprimentos de onda ocupados no enlace. O congestionamento de uma rota é representado pelo custo do enlace mais sobrecarregado. Quando surge uma requisição de circuito óptico, é utilizada a rota menos congestionada dentro de um conjunto de rotas definidas previamente. Havendo um empate pode ser utilizado o caminho mais curto (i.e. menor número de saltos) como critério de desempate.

O *Least Loaded Routing* - LLR é um outro exemplo de roteamento dinâmico alternativo. Esta técnica seleciona a rota com o maior número de comprimentos de onda livres com continuidade em todos os enlaces da rota.

Em geral, tem sido mostrado que o roteamento dinâmico alternativo apresenta menor probabilidade de bloqueio de conexão do que o roteamento fixo alternativo. Entretanto, os algoritmos de roteamento dinâmico alternativo geram maior sobrecarga com sinalizações e trocas de informações sobre o estado da rede do que os algoritmos de roteamento fixo alternativo

A terceira classe de roteamento é a do roteamento exaustivo. As rotas não são escolhidas de um conjunto de rotas pré-definidas. Qualquer uma das possíveis rotas que interligam os nós de origem e destino pode ser utilizada no atendimento do circuito óptico. As rotas são escolhidas dinamicamente em função do atual estado da rede. Uma exemplo de roteamento exaustivo é o ASCP, roteamento adaptativo de menor custo. Apesar de qualquer uma das rotas possa ser selecionada nesta técnica, prioriza-se o uso das rotas de menor custo, onde o custo pode ser o número de saltos da rota. Um enlace livre tem um custo 1 e um enlace ocupado tem custo ∞.

A Figura 63 ilustra um exemplo de escolha da rota para atender C(1,3) considerando o algoritmo de roteamento ASCP.

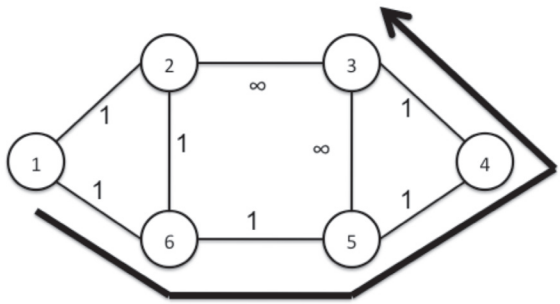


Figura 63. Exemplo de roteamento exaustivo.

Apesar da classe de roteamento exaustivo sugerir uma maior tolerância a falhas e uma probabilidade de bloqueio inferior quando comparada com as classes de roteamento fixo e roteamento alternativo, ela apresenta uma alta complexidade computacional, devido à alta sobrecarga ocasionada por frequentes atualizações das informações sobre o estado da rede. A classe de roteamento exaustivo tem que considerar todas as possibilidades de rotas para cada (o,d), isto é, a conexão será bloqueada somente se nenhuma dessas rotas possuir recursos disponíveis.

Observa-se que alguns autores consideram os roteamentos dinâmico alternativo e exaustivo como algoritmos de uma mesma classe de roteamento chamada de roteamento adaptativo, em virtude de ambos trabalharem em função do estado atual da rede, o que sugere um comportamento adaptativo.

ALOCAÇÃO DE COMPRIMENTO DE ONDA

As redes ópticas transparentes possuem características diferentes das tradicionais redes de circuito comutado devido à necessidade de respeitar a propriedade de continuidade obrigatória de comprimento de onda. Isto significa que um circuito óptico não pode utilizar qualquer comprimento de onda nos enlaces da rota definida. É necessário utilizar o mesmo comprimento de onda em todos os enlaces da rota escolhida pelo algoritmo de roteamento. Em função desta característica, um algoritmo de alocação de comprimento de onda tem o objetivo de selecionar um comprimento de onda para o estabelecimento de um circuito óptico tentando minimizar a probabilidade

de bloqueio de futuras requisições de circuitos ópticos. A propriedade de continuidade obrigatória de comprimento de onda pode ser desrespeitada se a rede fizer uso de dispositivos chamados conversores de comprimento de onda. Entretanto a tecnologia desse dispositivo ainda não está suficiente madura e esses possui um alto custo.

A Figura 64 ilustra uma rota 1, 2, 3, 4 e os estados dos seus enlaces.

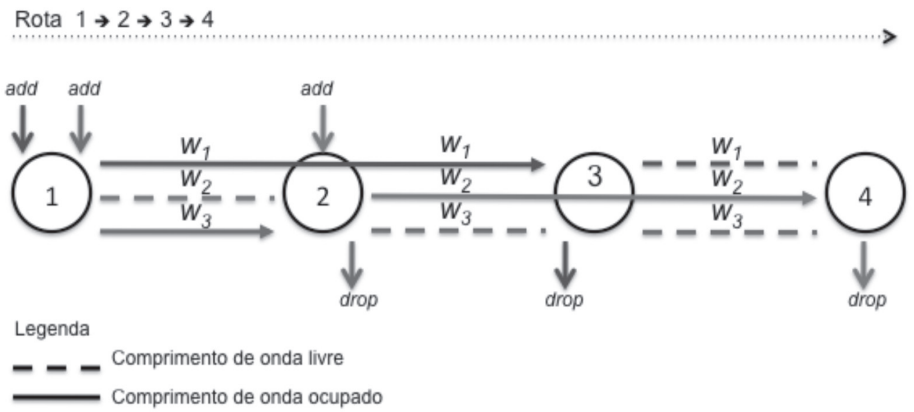


Figura 64. Exemplo do problema de alocação de comprimento de onda

Estão estabelecidas a C(1,3) no comprimento de onda azul, a C(2,4) no comprimento de onda vermelho e a C(1,2) no comprimento de onda verde. A partir desse estado da rede considera-se uma requisição de C(1,4) que deve utilizar a rota 1, 2, 3, 4.

De acordo com a propriedade de continuidade obrigatória de comprimento de onda, o estabelecimento da C(1,4) será bloqueado, pois não existe um comprimento de onda contínuo livre na rota 1, 2, 3, 4. Por outro lado, se ao invés de ser estabelecida no comprimento de onda verde C(1,2) fosse estabelecida no comprimento de onda vermelho, C(1,4) poderia ser estabelecida no comprimento de onda verde. Esse exemplo ilustra o impacto do algoritmo de alocação de comprimento de onda na probabilidade de bloqueio de futuras conexões.

Vários trabalhos foram realizados propondo novos algoritmos e/ou comparando o desempenho dos principais algoritmos de alocação de comprimento de onda (First-Fit, *Random Wavelength Assignment*, *Most Used*, *Least Used*, *Max-Sum*, *Relative Capacity Loss* dentre outros) considerando diferentes topologias e características de tráfego.

O *First-Fit* (FF) enumera todos os comprimentos de onda e aloca o

comprimento de onda disponível de menor número. *Random Wavelength Assignment* (RD) escolhe aleatoriamente um comprimento de onda dentre os disponíveis. *Most Used* (MU) escolhe o comprimento de onda disponível mais utilizado na rede. *Least Used* (LU) é exatamente o inverso do MU, ele escolhe o comprimento de onda disponível menos utilizado na rede. *Max-Sum* (MS) aloca um comprimento de onda, minimizando a perda de capacidade total. *Relative Capacity Loss* (RCL) aloca um comprimento de onda minimizando a perda de capacidade relativa.

De maneira geral, dentre os algoritmos de alocação de comprimento de onda citados, o *First-Fit* apresenta o melhor desempenho, uma vez que ele obtém baixa probabilidade de bloqueio e faz parte do grupo de algoritmos com menor complexidade (O(E.W) sendo E o número de enlaces da rota escolhida e W o número de comprimentos de onda). Diante dessas características, a grande maioria dos estudos de redes ópticas transparentes utiliza o algoritmo *First-Fit*.

EXERCÍCIOS

1) Explique as motivações para estudar redes ópticas.

2) Qual a diferença entre redes ópticas opacas e transparentes.

3) No que consiste a propriedade de continuidade obrigatória de comprimento de onda (*Wavelength Continuit Constrant*)?

4) Defina o problema RWA dinâmico e o diferencie do problema RWA estático.

5) Descreva o funcionamento dos algoritmos First-Fit, Random, Most Used, Least Used e Max Sum. Classifique-os em ordem crescente de complexidade.

6) Cite e explique as classes de algoritmos de roteamento em redes ópticas.

7) Como ocorre o estabelecimento de um circuito óptico em uma rede óptica transparentes?

CONSIDERAÇÕES FINAIS

O desenvolvimento de redes ópticas transparentes para compor a futura infraestutrura de transporte dos *backbones* dos provedores de serviços de telecomunicações é uma tendência mundial. Os recursos básicos dessas redes são caminhos ópticos compostos de comprimentos de onda multiplexados em WDM nas fibras ópticas. Esses recursos devem suportar a oferta de uma gama variada de serviços de telecomunicações com um aprovisionamento dinâmico sob demanda e otimizado para um volume de tráfego crescente.

Os algoritmos de roteamento e alocação de comprimento de onda apresentados constituem-se em instrumentos valiosos na otimização desses recursos. Principalmente quando a tecnologia de conversores de comprimento de onda ainda não está amadurecida e competitiva.

Referências

KUROSE, J. F. & ROSS, K. W. **Computer Networking**: a Top-Down Approach. 5. ed. Addison Wesleys, 2009.

TANEMBAUM, A. S. **Redes de computadores**. 4. ed. São Paulo: Editora Campus, 2003.

STALLINGS, W. **Redes e Sistemas de Comunicação de Dados**, São Paulo: Campus, 2005.

COMER, D. E. **Interligação de Redes com TCP/IP**: princípios, protocolos e arquitetura. 5. ed.,São Paulo: Campus, 2006. vol 1.

R. Jain, The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling. Wiley-Interscience, Apr 1991.

H. Zang, J. P. Jue, and B. Mukherjee, “A review of routing and wavelength assignment approaches for wavelength-routed optical WDM network,” SPIE Optical Networks Magazine, Jan 2000.

R. Ramaswami and K. N. Sivarajan, Optical Network - A Practical Perspective, 2nd ed. Morgan Kaufmann Publishers, 2002.

J. R. de Almeida Amazonas, **Projeto de Sistemas de Comunicações Ópticas**, 1st ed. Manole, 2005.

W. Giozza, E. Conforti, and H. Waldman, Fibras Ópticas - **Tecnologia e Projeto de Sistemas**, 1st ed. Makron, McGraw-Hill, 1991.

André Castelo Branco Soares, **“Uma Metodologia para Planejamento de**

Redes de Circuitos Ópticos Transparentes e Dinâmicos com Garantia de Qualidade de Serviço," Tese de Doutorado, Cin-UFPE, 2009.

C. Qiao and M. Yoo, "Optical burst switching (OBS) - A new paradigm for an optical Internet," Journal of High Speed Networks, pp. 69 – 84, Jan 1999.

C. Xin, C. Qiao, Y. Ye, and S. Dixit, "A hybrid optical switching approach," in IEEE GLOBECOM 2003, Dec 2003, pp. 3808 – 3812.

Steven E. Butner and Moji Ghodoussi, "Transforming a Surgical Robot for Human Telesurgery," in IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION 2003, Oct 2003, Vol 09, N. 19, pp. 818 –824.

REFERÊNCIAS NA WEB

Site do Prof. Jim Kurose
http://www-net.cs.umass.edu/personnel/kurose.html

Site do Prof. Leonard Kleinrock
http://www.lk.cs.ucla.edu/

Browser Internet Explorer
http://www.microsoft.com/brasil/windows/internet-explorer/

Fabricantes de equipamentos de interconexão
http://lat.3com.com/br (3COM)
http://www.cisco.com/web/BR/index.html (CISCO)
http://www.dlinkla.com/home/index.jsp (Dlink)

Provedores de rede de acesso
http://www.oi.com.br/ (Oi)
http://www.gvt.com.br (GVT)
http://www.vivo.com.br/ (Vivo)
http://www.claro.com.br/ (Claro)

Site da Rede Nacional de Pesquisa
http://www.rnp.br/noticias/imprensa/2002/not-imp-marco2002.html

A RNP e a História da Internet Brasileira
http://www.rnp.br/noticias/imprensa/2002/not-imp-marco2002.html

Página do inventor da WEB
http://www.w3.org/People/Berners-Lee/

Página com informações e uma entrevista com Tim Berners-Lee
http://veja.abril.com.br/especiais/tecnologia_2006/p_040.html

Página do browser WEB Safári
http://www.apple.com/safari/

Página do servidor WEB IIS da Microsoft
http://www.iis.net/

Página do servidor WEB Apache
http://httpd.apache.org/

Página do agente de usuário para e-mail Thenderbird
http://br.mozdev.org/thunderbird/

Página que explica a configuração do agente de usuário mail
http://support.apple.com/kb/HT1508?viewlocale=pt_BR&locale=pt_BR

Página do cliente FTP chamado SmartFTP
http://www.smartftp.com/

Projeto ESLEA
http://networks.internet2.edu/hopi/hopi-documentation.html

Projeto Glif
http://www.glif.is/publications/info/brochure.pdf

Projeto HOPI
http://networks.internet2.edu/hopi/hopi-documentation.html

www.uapi.ufpi.br