Introdução às Redes de Computadores de Hoje

Versão BETA2

Gustavo Lopes de Oliveira Santos

Julho de 2009

Resumo

Esta obra foi feita por Gustavo Lopes de Oliveira Santos. Este livro/apostila/manual encontra-se em versão digital, formato PDF. Não encontra-se em formato impresso. Esta obra pode ser baixada, como está apresentada aqui, através do site:

http://planoemfoco.com

Tabela de conteúdos

Lista de tabelas

Lista de figuras

Prólogo

Blá blum.

Parte I

Fundamentos

Capítulo 1

Conceitos de Redes de Computadores

Vamos direto ao ponto: redes são necessárias. Senão, este livro não teria sido feito e muita gente não estaria ganhando dinheiro com a profissão de redista. Além disso, milhares de pessoas agora estariam morrendo porque não poderiam acessar o Orkut ou outras coisas que consideram importantes. A verdade é que as pessoas usam, durante todo momento, redes de computadores - seja navegando na internet ou assistindo televisão - e nem se dão conta do que acontece em oculto. A importância da transmissão da informação é subestimada: redes de computadores **não é o mesmo que** cabeamento, como muitos pensam. Redes de computadores é um assunto vasto, que envolve um profundo conhecimento de equipamentos, configuração dos mesmos, monitoramento da rede, manutenção, planejamento, escalabilidade... Redes de computadores inclui, sim, cabeamento, mas este assunto é apenas um tópico muito ínfimo se comparado ao conhecimento total de redes.

Este capítulo tem por objetivo explicar o que são redes de computadores. Não uma explicação enciclopédica que fará você sair por aí todo poderoso virando caminhões com o poder de um olhar, mas uma explicação básica, fundamental, leve e verde (gosto de verde), que fará você compreender a coisa. Depois você poderá virar caminhões com o poder de um olhar.

1.1. O mundo depende das redes

Isso mesmo. O mundo depende das redes de computadores. Não deixe-se intimidar por desenvolvedores de software (isto é, programadores... eles não gostam muito de serem chamados programadores, mas vamos chamá-los assim, pois é divertido vê-los com raiva). Deixando as brincadeiras de lado, todo profissional da informação é importante, até os programadores (hehe). A informação precisa ser gerada, armazenada, processada e **transportada**. Se a mesma for gerada mas não for armazenada, de nada vai adiantar. O mesmo acontecerá se for armazenada, mas não puder ser lida, e assim por diante. O transporte da informação cabe ao profissional de redes de computadores, ao "redista", como iremos chamá-lo algumas vezes neste estudo.

Os "clientes" do redista englobam todo tipo de pessoa. O usuário doméstico que quer ver notícias no computador, o adolescente que, curioso com a anatomia do corpo humano, estuda com muito esforço e afinco imagens e até mesmo vídeos esclarecedores - para os mais dedicados. As redes de televisão disponibilizam seus programas na internet, e as empresas estão adotando solução de telefonia sobre IP, um tipo de telefonia que usa a arquitetura das redes de computadores.

1.2. O que é uma rede de computador?

Se você tem dois computadores isolados num mesmo ambiente, estes funcionam, mas não conversam. Não batem papo. Talvez tivessem muitas coisas para combinar, mas como nunca se conheceram, não poderão ser amigos. Não temos uma rede ainda, pois os computadores não trocam informações.

Figura 1.1. Computadores em ilha. Rede *fail*.

Alguém poderia dizer: eles podem sim trocar informações. Basta uma pessoa plugar um pendrive no computador `a`, copiar dados, plugar o pendrive no... ah, você entendeu. Isso não é muito inteligente. É o mesmo que um casal de namorados estar brigado e pedir para que uma terceira pessoa dê recados um ao outro por eles. Vemos deixar a cena melhor:

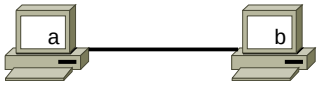

Figura 1.2. E viveram felizes para sempre.

Fim. Isso é uma rede de computadores, certo? Terminamos o curso. Vá para casa e ganhe muito dinheiro construindo redes. Uma vez que terminamos o curso mas não este livro, vou contar a história dos três porquinhos e o lobo mau. Era uma vez...

Não, não terminou. O computador `a` está ligado a `b`, mas isso não significa que eles podem trocar informações. Talvez eles falem linguagens diferentes (ou protocolos diferentes; é a mesma coisa). Ninguém tem certeza de que a informação passará pelo cabo. Temos aí, uma rede em sentido físico, visto que as duas máquinas estão conectadas; é o começo de uma rede de computadores. Entretanto, não é tudo. Para que esses computadores possam marcar alguma coisa no próximo fim de semana, é preciso, no mínimo:

1. Que `a` e `b` possam ser acessados. Ou seja, eles precisam ter dispositivos, internos ou externos, conectados aos seus respectivos barramentos[1.1], que possibilitem a comunicação em rede; esses equipamentos chamam-se "placas de redes". Um computador pode ter uma ou várias delas. Além disso, as placas de rede precisam ter algum nome ou endereço, para que possam ser chamadas.

2. As placas de rede precisam falar a mesma linguagem, ou protocolo.

3. As placas de rede devem conseguir acessar o cabo de cobre, de fibra ou outro objeto que seja capaz de transmitir sinais (o termo técnico deste objeto é enlace) de forma que as duas máquinas consigam conversar de forma viável, isto é, sem muitos erros.

As três regras são importantes, mas não são as únicas. Vamos estudá-las um pouco mais a fundo, para que você tenha uma ideia mais específica do que seja essa coisa toda de transmissão de informações.

1.3. ENDEREÇAMENTO

Para que os computadores numa mesma rede possam ser acessados, é necessário que haja uma identificação. No caso dos seres humanos, atendemos quando alguém chama nosso nome ou apelido. Não atendemos quando chamam pelo nosso RG. Ou sim. Bom, no caso dos computadores, essa identificação é algo que está contido na placa de rede. Cada placa de rede tem um endereço.

1.1. [Kurose & Ross], pág. 331.

Considere a figura abaixo:

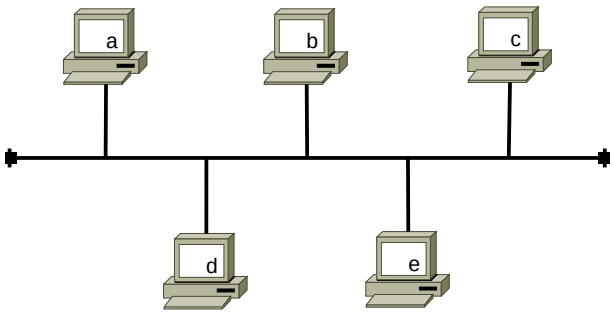

Figura 1.3. Uma rede com cinco computadores.

No desenho, cada computador está nomeado como `a`, `b`, `c` etc. Assuma que esses sejam os endereços das placas de rede dos respectivos computadores. Assim, é possível haver comunicação se, e somente se, os computadores souberem com quem falar. O computador cuja placa de rede tem endereço `a` pode enviar dados para o computador cujo endereço da placa de rede é `e`. `a` também pode desejar falar com todos os outros. Todavia, `a` não pode falar com `f`, pois este não existe, ou está inacessível.

Observe que a rede mostrada no desenho é composta de cinco computadores conectados a um único meio físico (enlace). Assuma que este enlace são cabos com fios de cobre. O enlace no qual estão conectados é próprio para transportar tais sinais elétricos. Os sinais elétricos são codificados de forma que, quando recebidos pelo computador destinatário, este decodifica o sinal para interpretá-lo. Se acontecer alguma coisa no enlace que altere o sinal elétrico, significa que quando a máquina destinatária receber o sinal, vai interpretá-lo de forma errada. Por isso, não devem acontecer interferências.

Agora, pense um momento sobre um fato interessante: se as máquinas que falam (as máquinas remetendes) colocam sinais elétricos no enlace, bom... todo o enlace será eletrificado. Se `a` deseja falar com `e`, não vai acontecer a mágica de só o caminho de `a` para `e` ser eletrificado: todo o enlace será. E a lógica diz que todos os outros computadores receberão os sinais elétricos. Observe a figura abaixo para perceber o que estou falando.

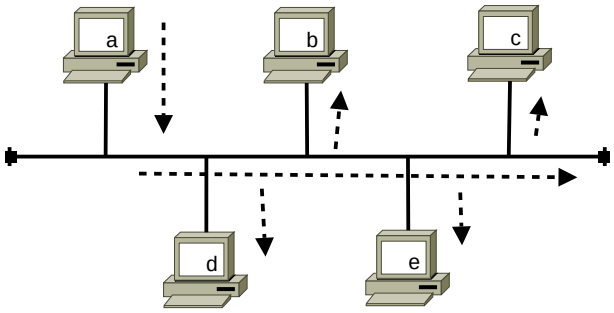

Figura 1.4. Todos os computadores, exceto quem envia, recebem os sinais elétricos.

Desagradável, não? Eis aí outro motivo para a existência do endereçamento: somente a máquina destinatária correta saberá que os sinais elétricos são para ela. “Como assim?”, você pergunta.

O computador `a` fala. Ele quer enviar uma informação para `e`. Assim, na sua fala, está contido o endereço do destinatário. `a`, então, eletrifica a rede, e todas as outras máquinas exceto ela própria recebem o sinal. Quando cada computador (ou melhor, cada placa de rede) recebe o sinal, interpreta-o, e verifica se ele é o destinatário dos sinais elétricos. Se for, aceita; caso contrário, nega. Simples. As placas de rede são programadas para obedecerem a essa regra[1.2]: "recebam apenas os sinais elétricos que são destinados a vocês".

Em suma: um envia, todos recebem e interpretam os sinais elétricos, mas só o destinatário trabalha os sinais. É lógico que pode haver mais de um destinatário, uma vez que `a` pode desejar falar com todos, por exemplo.

1.4. PROTOCOLO

Não vamos gastar 4.000 páginas explicando o que são protocolos, uma vez que você já sabe que é o mesmo que linguagem. Entretanto, poderíamos gastar 4.000 páginas apresentando centenas, talvez milhares de protocolos existentes para comunicação em redes de computadores, sendo que você nunca na vida usaria todos eles, e essa discussão seria inútil.

DEFINIÇÃO 1.1. *Protocolo. Protocolo é uma linguagem e também um conjunto de boas maneiras que define como os computadores devem falar, e também, ouvir.*

Educação é primordial em redes de computadores, como ficará claro neste exemplo: suponha que, em nossa rede de cinco computadores, dois deles queiram falar ao mesmo tempo. Claro, os computadores não são tão grosseiros assim, e querem falar com colegas diferentes: `a` quer falar com `e` e `c` quer falar com `d`. Observe o que acontece:

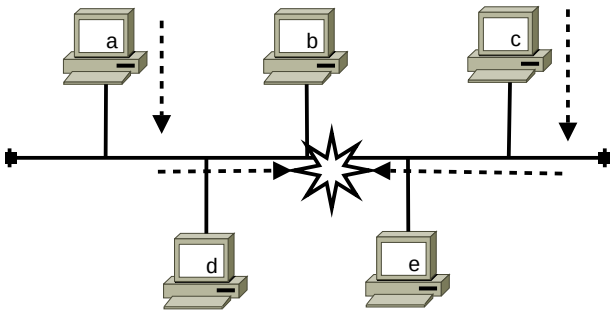

Figura 1.5. Uma colisão.

Cabum! Uma colisão. Uma explosão de se ouvir no outro quarteirão... prédio em chamas, computadores em curto circuito e bombeiros desesperadamente tentando salvar o pobre cachorrinho na janela.

Está bem, não chega a tanto, mas não é uma coisa muito agradável ver toda a rede sem conexão cada vez que isso acontece. Mesmo que o estrago seja mínimo (o estrago real é apenas as máquinas não conseguirem se comunicar por alguns milissegundos), não queremos que isso fique acontecendo constantemente!

Esse problema é causado simplesmente porque todo o enlace é eletrificado quando alguma máquina fala. Assim, se duas máquinas falam ao mesmo tempo o enlace é duplamente eletrificado, e os sinais elétricos, é claro, são totalmente alterados e se tornam ilegíveis para a máquina destinatária. Agora, imagine uma rede com 200 computadores! Quantas colisões, ein?

1.2. Sim, toda regra tem exceção.

Entra no enredo o protocolo, destemido e desbravador herói de óculos escuros que vem dar fim aos problemas da comunicação. Vimos que além de ser uma linguagem (duas máquinas podem conversar se usam o mesmo protocolo), também é um conjunto de regras que definam a boa educação na rede. Um protocolo poderia definir, por exemplo, as seguintes regras para comunicação:

1. A máquina que quer falar deve primeiro escutar o enlace, para ver se alguém já está falando por meio dele.
2. Se o enlace estiver ocupado, então espera mais um pouco e escuta novamente mais tarde.
3. Se o enlace estiver desocupado, então começa a falar nele.
4. Se for percebido que houve uma colisão, então a máquina termina de falar, espera um pouco e escuta a rede. Ou seja, volta ao ponto 1.

É um exemplo de protocolo. Começarei a falar de protocolos específicos mais tarde neste curso. Por enquanto, este exemplo deixa claro que um protocolo é uma linguagem e um conjunto de regras para comunicação em redes de computadores. Não existe um, e sim muitos protocolos, cada um com suas próprias regras, gostos gastronômicos e modos de se vestir.

1.5. Pra que tantos cabos?

Conforme as redes foram crescendo, achou-se difícil fazer manutenção na mesma, devido a quantidade de cabos espalhados por aí. Pessoas tropeçando, muito dinheiro gasto na substituição dos mesmos... uma caca. Então inventaram o repetidor, ou hub[1.3]: um equipamento que funciona como um cabo na qual outros cabos são conectados.

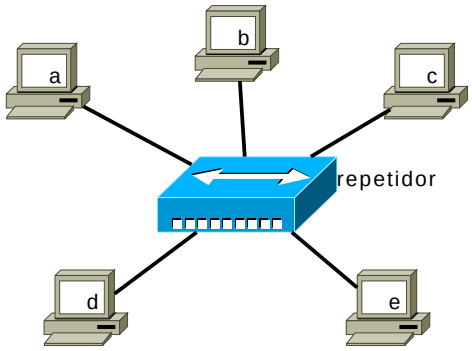

Figura 1.6. Uma rede com repetidor.

A invenção do repetidor foi uma introdução às redes locais modernas. Hoje temos equipamentos de rede muito mais sofisticados, como veremos ao longo deste estudo. O repetidor resolveu o problema dos milhares de metros de cabos, mas apenas isso foi resolvido; o velho problema da colisão continua: quando uma máquina fala, todo o meio (o repetidor e os outros cabos) ficam ocupados; ainda é necessário um protocolo para regular a comunicação em redes que usam repetidor.

1.3. Ou, ainda, concentrador. Mas concentrador é uma palavra muito genérica... um comutador não seria, também, um concentrador?

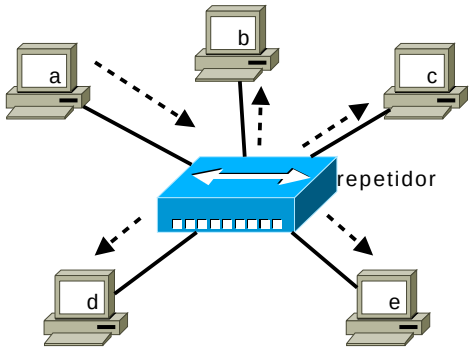

Figura 1.7. Uma rede com repetidor, com o velho problema do enlace ocupado.

Você deve estar se perguntando: "será que há uma maneira de uma máquina que fala ocupar apenas os enlaces específicos com quem quer falar?". Ou seja, será que existe a possibilidade de que, se `a` quer falar com `e`, somente o enlace que liga `a` ao equipamento central, e o enlace que liga o equipamento central a `e`, fiquem ocupados? A resposta é: sim. Hoje as redes são assim; já não usam um único cabo ou um repetidor para fazer a comunicação. Isso é coisa do passado, é brega, coroa. Ainda existem masoquistas que gostam da velha maneira, mas hoje em dia, as redes locais usam *comutadores*. Estudaremos sobre comutadores em breve; segure sua curiosidade por um momento. Ainda nos resta falar um pouco sobre a internet, a grande rede.

1.6. CONECTANDO-SE A INTERNET

A internet é uma rede, mas diferente das redes que vimos neste capítulo. Até agora, vimos redes mais simples, em que todos os computadores têm em comum o mesmo enlace, seja este enlace um cabo único, ou um repetidor. Este tipo de rede chama-se rede local, ou LAN (de Local Area Network), e uma parte de nosso estudo concentra-se nelas. A internet, contudo, é um tipo de rede mais complexa, tanto pela abrangência geográfica (ela não ocupa apenas uma sala, um escritório ou uma empresa; ocupa o mundo todo), quando pela diversidade de tecnologias e protocolos. Enquanto uma rede local possui um único protocolo que dita o funcionamento das máquinas, a internet conecta diversas redes de protocolos diferentes, e a própria internet possui estruturas diversas. A internet é uma rede de longa distância, uma WAN (Wide Area Network), a maior das WAN's. Até agora, tudo que sabemos sobre internet neste curso é que seu computador liga-se a ela por meio de um cabo, que sai da sua casa em direção à próxima central telefônica ou outro tipo de provedor de acesso. A arquitetura da internet é uma nuvem nebulosa, um desconhecido.

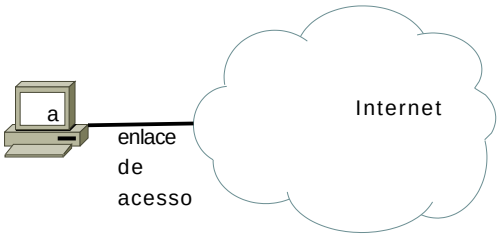

Figura 1.8. Um computador conectado à internet.

Dissemos que a internet interliga várias redes. Sim, isso mesmo, “redes”, e não, “máquinas”. Com o conhecimento que você tem até agora, pode começar a ter ideias mirabolantes... hum... que tal uma coisa assim:

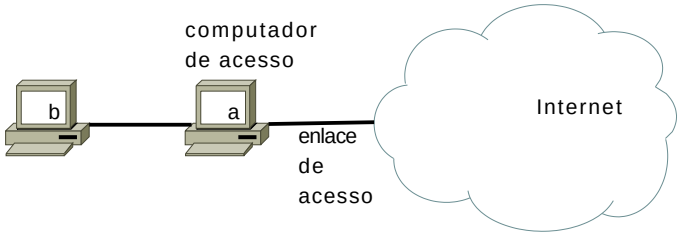

Figura 1.9. Dois computadores conectador à internet, mas pagando por um!

Na figura, temos um computador (o computador `a`) conectado diretamente à internet, e um segundo computador conectado ao computador `a`. Isso é possível, e você não precisa pagar duas conexões para as operadoras de telefonia. Basta o computador `a` ter duas placas de rede. A internet é como temperatura, e os computadores ligados a ela são como ótimos condutores: a nuvem da figura está quente, e o computador `a` está frio enquanto estiver desconectado da nuvem. Assim que estiver conectado, passa a ficar “quente”, ou seja, com internet. E o computador `b` também ficará “quente” ao ser conectado ao computador `a`. É contagioso!

Aí você pode começar a pensar alto: hum... e se eu tiver, em vez de um computador, uma rede completa ligada ao computador de acesso? Posso até ser um provedor de acesso! Muito bem, desbravador, este é o caminho!

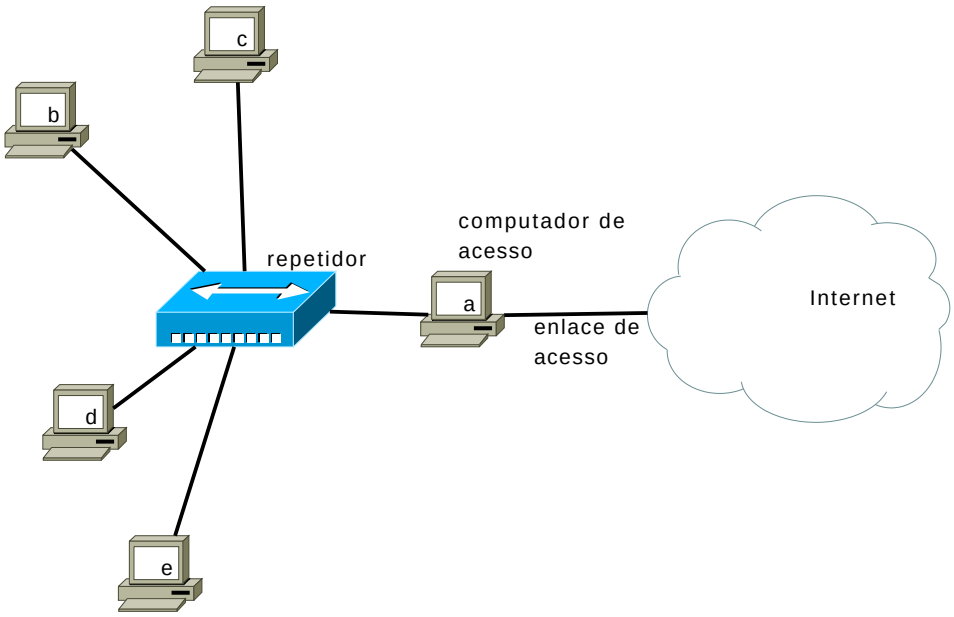

Figura 1.10. Estou rico!

Na figura, temos uma rede completa conectada à internet. Neste exemplo, o computador `a` provê acesso à internet para toda a rede. Os outros computadores são interligados ao repetidor. Não é porque a internet é uma nuvem nebulosa que não podemos explorá-la,

não acha? Mas faremos muito mais neste curso do que explorá-la. Nós compreenderemos seu funcionamento, desenvolveremos projetos, acharemos soluções... e o mesmo dizemos sobre as redes locais, sobre as quais você tem muito mais controle, pois os equipamentos e a infraestrutura pertencem a você.

1.7. ARQUITETURA DE REDE

Arquitetura é coletivo de protocolo. Um conjunto de protocolos forma uma arquitetura. A arquitetura mais usada atualmente é a TCP/IP: na internet, em muitíssimas redes locais... o nome deve-se a dois importantes protocolos desta arquitetura: o TCP e o IP. Claro, existem outras arquiteturas, mas vamos estudar TCP/IP neste curso, visto que a usamos hoje. A versão que usamos desta arquitetura é a 4 (Protocolo IPv4), de 1981[1.4], contudo estamos em fase de transição para a versão 6 (IPv6).

Podemos dividir as redes de computadores em camadas[1.5]. Neste capítulo você teve uma visão geral da camada mais baixa da rede, que são os enlaces físicos (cabos e o repetidor), e dissemos um pouco também sobre as regras da camada que está imediatamente sobre esta, que é a camada enlace. Você sabe: pelos enlaces transitam sinais elétricos. Todavia, você deve concordar também que os computadores possuem muito mais do que sinais elétricos: eles possuem informações. Os sinais elétricos são apenas informações convertidas para um formato que pode ser transmitido. As informações propriamente ditas são manipuladas por outros protocolos, diferentes daqueles protocolos que manipulam sinais elétricos. Não entendeu? Observe bem: o enlace físico, propriamente dito, não possui nenhum protocolo... ele apenas existe para transportar sinais elétricos. Existem, contudo, protocolos nas máquinas que dizem a elas como transmitir, como falar e como ouvir sinais elétricos. A inteligência não está no enlace, mas nas máquinas; estes protocolos são necessários para que seja possível a comunicação através do enlace físico. Assim, temos protocolos que operam sobre o enlace, mas não nele; operam em uma camada acima. Em redes, a camada mais baixa é a camada física: nela se encontram os enlaces físicos e equipamentos que fazem parte dela, como repetidores. A camada imediatamente acima da camada física é a camada enlace. Sim, isso mesmo, você pode achar um pouco estranho esse nome, uma vez que os enlaces estão na camada física; mas é isso mesmo: a camada física contém os enlaces, e a camada enlace contém os protocolos de acesso aos enlaces.

Mas a camada enlace apenas dita as regras para que a comunicação seja possível. A camada enlace tem o objetivo de "pegar" os dados da máquina que quer transmitir, e falar esses dados no enlace físico, obedecendo as regras do protocolo de camada enlace, que dita como a máquina deve falar no enlace físico. Na outra ponta da rede, a máquina destinatária também usará o protocolo de camada enlace para saber como deve ouvir a informação que vem pelo enlace físico. Assim que tiver recebido a informação, a camada enlace da máquina destinatária traduz os sinais elétricos e passa a informação para que a máquina trabalhe com ela. A camada enlace, portanto, existe em todas as máquinas de uma rede; bem como todos os protocolos desta camada. Observe o que foi dito, na figura abaixo:

1.4. [IPv6.br]; [RFC 791].

1.5. Nossa divisão é baseada naquela adotada por [Kurose & Ross]. As quatro camadas da internet estão sobre a camada física, portanto, temos 5 camadas ao todo. Em [Kurose & Ross], são reconhecidas 5 camadas, e não 4. O que acontece é que o livro não enfatisa a camada física. Todavia, comprova sua existência como camada distinta, conforme pode observar-se nas páginas 36 (figura 1.17), 37-38 etc. A internet e, por conseguinte, as redes de computadores de hoje, possuem uma arquitetura em 5 camadas segundo esta visão.

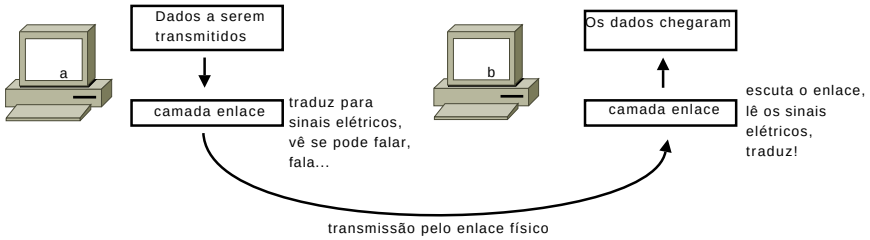

Figura 1.11. Máquina a transmite para b.

Em suma:

1. A máquina `a` tem dados a transmitir. Esses dados podem ser, por exemplo, um email digitado pelo usuário em uma aplicação própria.

2. A aplicação de email não possui acesso à placa de rede, e muito menos sabe traduzir o email para sinais elétricos. Sendo assim, o programa de email manda os dados para a camada enlace do computador, e fica despreocupado.

3. A camada enlace de `a` trata de trabalhar com os dados recebidos pelo programa de email. Ela vai, basicamente, traduzir os dados para sinais elétricos e enviar pelo enlace físico.

4. Os dados vão transitar pelo enlace físico até a máquina `b`.

5. A camada enlace da máquina `b`, ao receber os sinais elétricos, verificará se a destinatária é a máquina `b`. Se for, então processará os sinais elétricos, transformando-os novamente em informação lógica.

6. Finalmente, a camada enlace da máquina `b` passará os dados para o programa de emails próprio.

Claro, o passo-a-passo acima está muito simplista. Muito mais coisas acontecem além disso. Mas percebemos, ao menos, a existência de três camadas nesta rede: a camada física, a camada enlace, e a camada... bom, não demos um nome ainda, mas você sabe que é a camada que fica imediatamente acima da camada enlace, em cada máquina.

Agora, o tiro de misericórdia: você compreendeu basicamente como funciona a comunicação entre dois computadores, mas nossas redes modernas baseadas em TCP/IP não possuem apenas três camadas. Elas possuem cinco! Veja figura abaixo:

Figura 1.12. Redes modernas baseadas na arquitetura TCP/IP.

Já falamos sobre a camada física, e a camada enlace. A parte dos "dados" compreendem as outras três camadas, cada uma delas com seus próprios protocolos e funções.

1.8. COMUNICAÇÃO ENTRE CAMADAS

Dois computadores comunicam-se através do enlace físico; porém (e você compreendeu isso), a máquina remetente envia os dados das camadas superiores para a camada enlace, que por sua vez traduz em sinais elétricos para a camada física; e a máquina que escuta lê os dados elétricos da camada física, usa a camada enlace para traduzir os sinais elétricos em dados, e repassa para as camadas superiores. A figura abaixo deixa isso mais evidente:

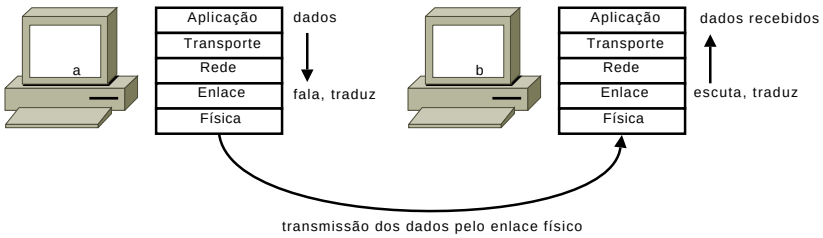

Figura 1.13. Transmissão de um computador para outro.

A camada aplicação contém os dados a serem enviados; você, na máquina `a`, digita um email em um programa de email, por exemplo, direcionado para a máquina `b`. Quando você clica em "enviar", o programa de email passa os dados para a camada transporte, que por sua vez, passa os dados para a camada rede, que passa para a camada enlace. A camada enlace usa de suas regras para ver se dá para transmitir no meio físico; ela, então, fala os dados no enlace físico, traduzindo-os em sinais elétricos.

Quando a informação chega pelo enlace físico à máquina `b`, a camada enlace desta, após escutar os sinais elétricos, traduz estes sinais e passa a informação para a camada rede, que passa para a camada transporte, que finalmente passa para a aplicação de email desta máquina. Você deve estar se perguntando para que tantas camadas; analisaremos em breve, neste curso. Para o momento, basta saber a ordem das coisas: a máquina remetente desce com os dados através das camadas; a destinatária, sobe com os dados.

Além do que foi dito até agora, cada camada manipula os dados à sua maneira, adicionando ou retirando informações. Na máquina que fala, cada vez que os dados vão descendo, as camadas adicionam informações a eles; na máquina destinatária, cada vez que os dados vão subindo, as camadas vão tirando informações deles. Mas não se preocupe: a informação da aplicação não é alterada. A camada transporte da máquina destinatária vai retirar somente a informação que a mesma camada na máquina que fala colocou. Este processo chamamos de encapsulamento.

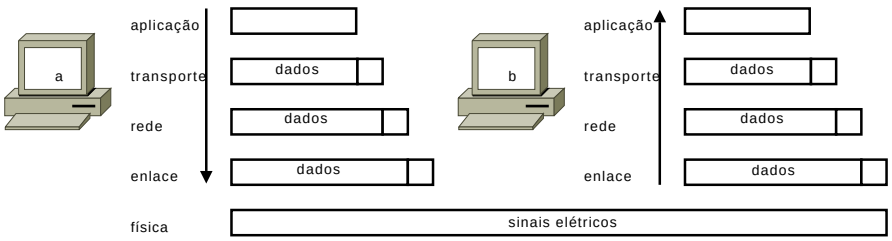

Figura 1.14. Encapsulamento.

Acontece que o que cada camada faz com os dados que recebe é parecido com o ato de empacotar ou desempacotar alguma coisa. Imagine uma brincadeira infantil em que as crianças põem uma carta dentro de um envelope, e este envelope dentro de outro envelope... é o que acontece nas redes de computadores. E isso, com um objetivo, que você entenderá ao longo deste curso.

A camada aplicação da máquina `a` envia os dados para a camada imediatamente inferior, que é a camada transporte. Esta camada recebe os dados, e adiciona mais dados aos dados existentes, sem alterá-los. É como se uma carta fosse envelopada. A camada transporte, então, envia tudo isso (os dados originais mais os dados que ela mesma colocou) à camada rede. Do ponto de vista da camada rede, os dados são tudo aquilo que ela recebeu da camada transporte. A camada rede não sabe diferenciar entre dados da camada aplicação e dados da camada transporte: *o todo são os dados*. Assim, esta camada também adiciona informações suas aos dados recebidos, envelopando o envelope mais uma vez, e passando para a camada enlace. Como você pode ver, na máquina que envia dados, conforme a informação vai descendo pelas camadas, seu tamanho vai aumentando.

Quando os sinais elétricos chegam à camada enlace da máquina `b`, esta camada traduz os dados, e retira os dados que a camada enlace da máquina `a` colocou. Após, sobe com os dados. A camada rede de `b` também retira os dados que a camada rede de `a` colocou, e sobe o pacote; isto continua até que os dados originais cheguem à camada aplicação da máquina `b`. Temos, assim, que as camadas das duas máquinas conversam entre si: a camada enlace das duas máquinas se entendem, conversam, pois uma coloca informação que a camada da outra máquina lê; a camada transporte da máquina `a` pode anexar uma piada, por exemplo, nos dados, para que quando a camada transporte da máquina `b` receber, leia e ria muito. A camada transporte da máquina `a` sabe que os dados adicionados por ela não serão lidos pelas camadas enlace ou rede da máquina `b`; por isso, pode adicionar coisas como "rede levou um tapa da namorada" ou "a camada enlace usa prótese". As camadas se entendem, se relacionam, se amam e marcam encontros sem que os outros se intrometam nas suas vidas. Bom para elas.

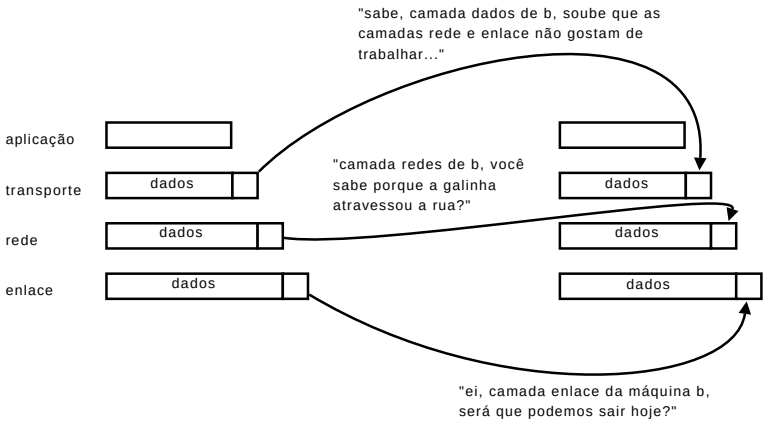

Figura 1.15. Comunicação de camadas em máquina diferentes.

1.9. Organização deste livro

Este livro é organizado de forma que você estude primeiro as camadas inferiores, e depois as camadas superiores das redes baseadas em TCP/IP. Este capítulos deu a você um entendimento básico do que é uma rede de computador. Agora podemos começar a nos aprofundar em nossos estudos.

No próximo capítulo, estudaremos sobre as duas camadas superiores, aplicação e transporte, visto que no âmbito deste curso, não são tão relevantes quanto as camadas rede e enlace. Depois, iniciaremos a parte que fala sobre comutação, que é uma função da camada enlace, e em seguida, perto do fim do curso, estudaremos sobre roteamento, uma função da camada rede. A figura abaixo mostra a ordem que se dará esse curso:

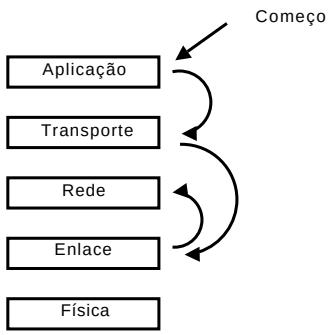

Figura 1.16. Estrutura do curso.

1.10. EXERCÍCIOS

Exercício 1.1. Defina, com suas palavras, o que é uma rede de computadores.

Exercício 1.2. E o que é um protocolo? Qual a utilidade do mesmo?

Exercício 1.3. Verdadeiro ou falso:

a) Em uma rede local, o endereçamento físico é um dos requisitos necessários para que haja comunicação entre as máquinas.

b) Quando uma máquina fala, somente a máquina destinatária recebe os sinais elétricos.

c) Em uma rede com meio físico compartilhado, é necessário um protocolo para regular a educação das máquinas.

d) A camada enlace da máquina destinatária recebe os sinais elétricos do enlace físico, e verifica se o destino físico é o correto.

Exercício 1.4. O que é uma colisão? Quando ela ocorre?

Exercício 1.5. Diferencie LAN de WAN.

Exercício 1.6. Verdadeiro ou falso:

a) Não é possível conectar uma rede inteira à internet usando-se um único computador; é necessário um enlace com acesso à internet para cada máquina.

b) Hoje em dia, muitas redes modernas usam a arquitetura TCP/IP.

c) Podemos dividir redes baseadas em TCP/IP em três camadas: cama física, camada de rede e camada do usuário.

d) Na camada enlace encontra-se o protocolo que dita as regras de comunicação das máquinas na rede local.

Exercício 1.7. Como ocorre a transmissão dos dados pelas cinco camadas, entre duas máquinas?

Exercício 1.8. De que forma duas camadas de máquinas diferentes trocam informações?

Capítulo 2

Funcionamento Básico de Redes TCP/IP

Este capítulo tem o objetivo de deixar mais claro na sua mente o conceito da arquitetura de redes em camadas. Além disso, na segunda parte deste capítulo discutiremos sobre as duas camadas superiores das redes baseadas em TCP/IP: a camada aplicação e a camada transporte. Você verá que computadores possuem tanto um endereço físico quanto um endereço lógico: duas camadas (a camada enlace e a camada rede) são responsáveis pelo endereçamento. Você também entenderá o porquê da necessidade de dois endereços.

2.1. Revisão

Podemos dividir as redes em dois tipos: redes locais, ou LAN's, e redes de longa distância, ou WAN's. A internet é a maior das WAN's, e interconecta milhares de redes. Em redes locais, é necessário que as máquinas possuam endereços exclusivos, para que possam conversar na rede. Também em redes locais, é necessário o uso de um protocolo para possibilitar a comunicação das máquinas, e ditar as regras de boa educação. Até agora, você viu redes locais construídas usando-se um único enlace, que é compartilhado por todas as máquinas. Este enlace pode ser um cabo, ou um repetidor. Quando qualquer das máquinas da rede fala, o enlace inteiro é eletrificado, e todas as máquinas escutam. Porém, somente a máquina destinatária captura os sinais elétricos, transformando-os em dados e enviando para cima.

A máquina que envia os dados desce com eles pelas cinco camadas. A máquina que recebe os dados sobe com eles pelas cinco camadas. Quando cada camada, na máquina remetente, recebe um dado da camada superior, ela adiciona informações aos dados que recebe. Essa informação adicionada será lida e retirada pela camada equivalente na máquina destinatária. Assim, camadas de máquinas diferentes podem conversar durante a transmissão.

Uma máquina pode ser conectada à internet por meio de um enlace; mas não somente uma máquina: pode-se ter uma rede inteira ligada à internet. Você viu que uma máquina com conexão pode compartilhar esta conexão; temos, portanto, várias máquinas compartilhando um único enlace de conexão à internet. Como a internet é uma rede de longa distância (WAN), podemos chamar este enlace que liga a rede local à internet de *enlace WAN*. Por conseguinte, os enlaces que ligam as máquinas nas redes locais podem receber a denominação de enlaces LAN. É só uma questão de nomenclatura, você não vai morrer se não gravar isso. Mas convenhamos que é melhor falar "enlace WAN" do que "enlace que liga sua interessante e esplêndida máquina com processador legal e muita memória à rede de longa distância onde trafegam informações relevantes para a manutenção do planeta terra".[2.1]

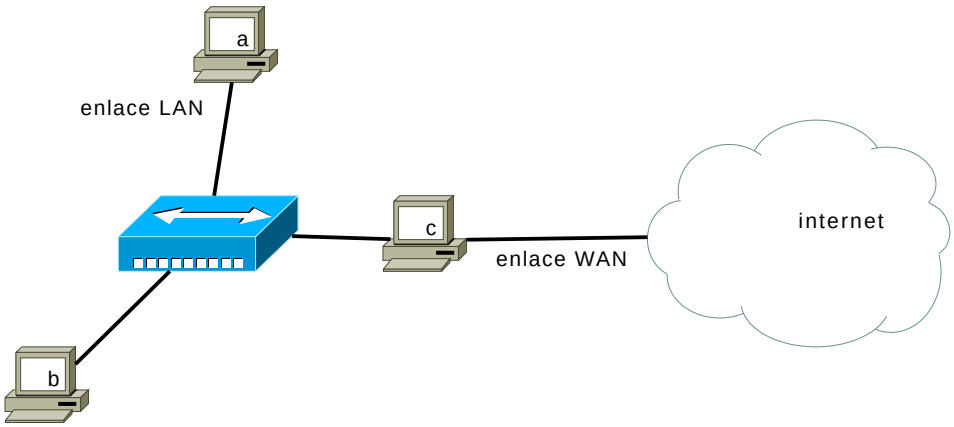

Figura 2.1. Enlaces LAN e WAN

2.2. DUPLO ENDEREÇAMENTO

Vamos começar essa seção psiquiátrica frustrando você. Você foi enganado, mentimos, burlamos seu cérebro. Uma máquina não possui apenas um endereço de rede: ela possui dois. Você pode pensar: "sim, óbvio, pois uma única máquina pode ter duas placas de rede...". Mas não é isso que estou falando. O que estou dizendo é que, uma máquina na rede precisa ter **obrigatoriamente** dois endereços: um endereço físico, e um endereço lógico. Com respeito ao endereço físico, você já tem uma noção. É um endereço que atua na camada enlace (lembra-se? quando uma máquina recebe sinais elétricos, ela verifica se o endereço destinatário é ela própria; isso é feito pela camada enlace, e, portanto, a camada enlace cuida do endereçamento físico). Porém, temos também um endereço que atua na camada imediatamente superior à camada enlace: o endereço lógico, na camada rede.

Isso significa que a máquina irá verificar não uma vez, mas duas vezes, para ter certeza de que aquela informação é para ela mesmo. "Isso é redundante", você pensa. Realmente é, mas tem um objetivo que vamos deixar claro daqui a pouco. Entretanto, vamos resumir o que acontece quando uma máquina recebe dados:

1. Os sinais elétricos chegam na placa de rede. A camada enlace entra em ação!

2. A camada enlace verifica se a máquina é destinatária dos dados. Se for, envia os dados para a camada rede.

3. A camada rede, que não está na placa de rede, e sim no sistema operacional do hospedeiro, verifica se a máquina é destinatária dos dados. Desta vez, em vez de verificar o endereço físico, verifica o endereço lógico de destino da informação. Se a máquina for realmente a destinatária, então, passa os dados para a camada transporte.

Duas verificações: uma feita no âmbito da camada enlace, e outra, no da camada física. Você está entendendo que quando a máquina que envia a informação fala, a camada de rede desta máquina escreve o endereço da camada de rede da máquina destinatária, e a camada

2.1. Na verdade, um enlace WAN é representado por uma linha em forma de raio; mas, para facilitar nossa discussão, vamos usar a linha simples por enquanto. E, em nosso exemplo, enlace WAN representa meramente o cabo que liga o computador à internet.

enlace escreve o endereço que a camada enlace da máquina destinatária lerá. Observe a figura abaixo para mais esclarecimentos:

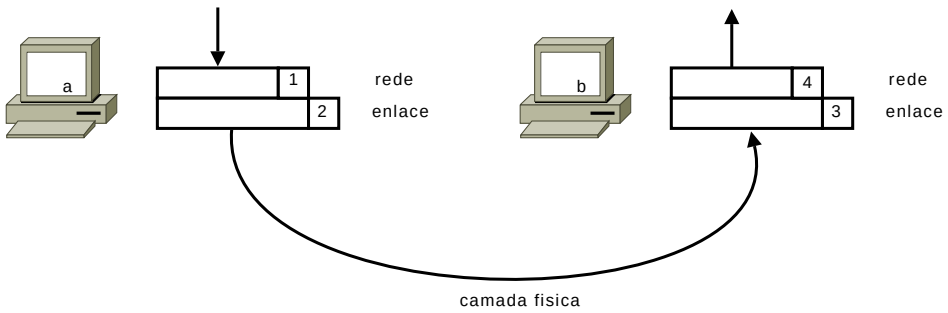

Figura 2.2. Transmissão dos dados.

Na figura acima, os números indicam a sequência das coisas:

1. O endereço lógico (endereço de camada rede) de destino é adicionado pela camada rede da máquina `a`.

2. O endereço físico (endereço de camada enlace) de destino é adicionado pela camada enlace da máquina `a`.

3. A camada enlace da máquina `b` lê o endereço físico (endereço de camada enlace) da informação que chega. Se o endereço for o desta máquina, então, retira os dados de camada enlace (portanto, sobram os dados da camada rede) e passa os dados para cima.

4. A camada rede, por sua vez, lê o endereço lógico. Se o endereço for o desta máquina, então, tudo bem, passa a informação para cima.

Exercício 2.1. Por essa lógica, existe a possibilidade de a camada enlace da máquina `b` aceitar a informação, e a camada rede negar? Justifique.

2.3. Pacotes e quadros

Até agora, estamos usando o termo "informação" para descrever os dados que chegam à camada enlace, e "sinais elétricos" para descrever os dados que trafegam pelo meio físico. Os termos técnicos, porém, passarão a ser usados: pacotes e quadros.

Definição 2.1. *Pacote. Chamamos de* **pacote** *os dados manipulados pela camada de rede. Lembre-se que tais pacotes contém, além de dados da aplicação, dados adicionados pela camada rede. Algumas literaturas chamam um pacote de datagrama*[2.2]*.*

2.2. Como [Kurose e Ross], por exemplo.

DEFINIÇÃO 2.2. *Quadro. Um* ***quadro*** *é um pacote mais as informações adicionadas pela camada enlace. Um quadro é maior que um pacote, portanto. É o quadro que trafega pelos enlaces físicos da rede.*

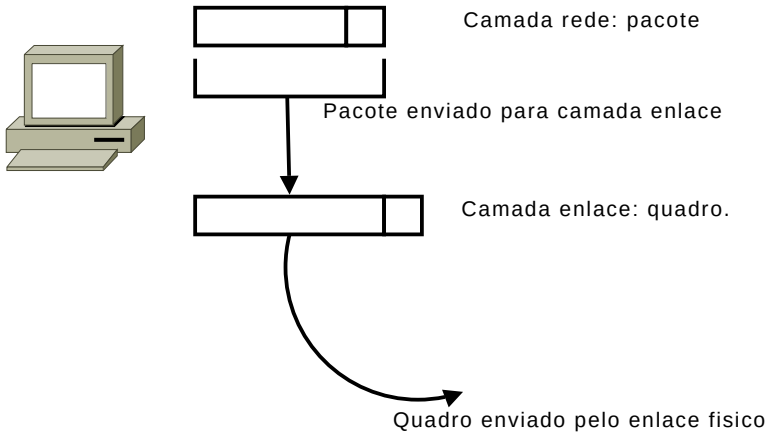

Figura 2.3. Pacote e quadro.

2.4. A NECESSIDADE DO ENDEREÇO LÓGICO

O endereçamento de camada rede (endereçamento lógico) é necessário. No momento, parece que é algo totalmente contra o cérebro humano, e coisa de masoquista, mas você compreenderá sua necessidade (embora seja algo totalmente contra o cérebro humano e masoquismo).

Em primeiro lugar, você já pensou se toda a internet recebesse todos os quadros que uma máquina enviasse para outra? Você sabe que em uma rede local (LAN) com enlace compartilhado (cabo único ou repetidor, por exemplo), quando uma máquina `a` envia dados para uma máquina `b`, todo o enlace é eletrificado, e todas as máquinas recebem o quadro. Agora, imagine se isso fosse verdadeiro também na internet: quando cada máquina do mundo falasse alguma coisa, todas as outras máquinas escutariam isso. Ouvido de tuberculoso. Muito desagradável você se deparar com uma almofada em forma de braço feminino (que provavelmente algum cara muito carente comprou) enquanto estiver lendo sobre cirurgias que deram errado na internet. Ainda bem que a internet não é assim. A internet é uma rede que interconecta muitas outras redes; não é uma coisa única, uma rede local gigantesca. Por isso, e preste bastante atenção pois seu pâncreas precisa disso para viver, máquinas na rede local só enxergam endereços físicos da rede local. Pegou?

DEFINIÇÃO 2.3. *Escopo do endereçamento de camada enlace. Máquinas numa rede local só conseguem enxergar endereços de camada enlace de máquinas que estejam na mesma rede local.*

Isso mesmo. É como numa sala de aula, em que o professor irritado chama o responsável pelo alfinete na sua cadeira: "Rachmaninov!". No caso, o professor se refere ao Rachmaninov que está na sala naquele momento, e não a algum outro no mundo. Abaixo, uma figura pra você ficar feliz (a máquina `a` quer falar com a máquina `f`).

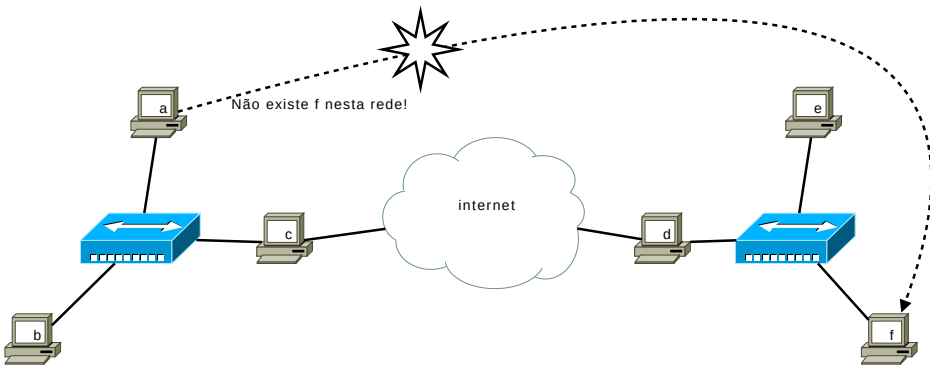

Figura 2.4. As máquinas só enxergam endereços físicos na mesma rede local.

Nesta figura, com sua visão semidivina de amplo espectro você consegue ver seis máquinas. Entretanto, do ponto de vista míope da máquina `a`, só existem mais dois endereços físicos além do próprio: `b` e `c`. O que acontecerá se a máquina `a` tentar enviar um quadro para o endereço `f`? Você acha que o quadro atravessará a internet, alegre e pimpolho, direto para o endereço físico `f`? Não, não é assim! Primeiro, porque somente o enlace da rede local será eletrificado, e não o enlace WAN. Segundo, porque `f` não pode ser localizado na rede local: está fora dela. E agora, José?

Observe a figura abaixo:

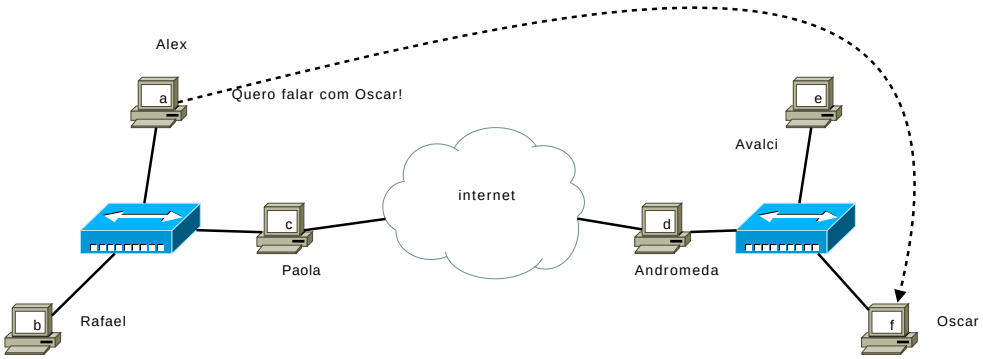

Figura 2.5. As máquinas enxergam endereços lógicos em redes diferentes.

Na figura acima, cada máquina tem dois endereços: um físico (da camada enlace), e um lógico (da camada rede). A máquina cujo endereço de camada enlace é `a`, tem também um endereço de camada rede `Alex`; já a máquina com quem se quer falar, tem endereço de camada enlace `f`, e de camada rede `Oscar`.

Embora a camada enlace de máquina que está falando não possa localizar o endereço físico `f`, a camada rede da máquina que fala (`Alex`) consegue localizar o endereço lógico da máquina destinatária. E o endereço lógico, neste exemplo, é `Oscar`; mas a verdade é que, em redes modernas baseadas em TCP/IP, endereços lógicos são números. Não entraremos nesse mérito agora. O importante é compreender a diferença de um endereço para o outro.

	Endereço físico	**Endereço lógico**
Em que camada se localiza?	Camada enlace	Camada rede
Qual a visibilidade?	É visível apenas na LAN	Pode ser visível em WAN's

Tabela 2.1. Comparação entre endereço físico e endereço lógico.

Observe que dissemos "pode ser visível em WAN's". Isso porque, embora o endereço lógico possa ser público, o gerente de rede tem a possibilidade de não permitir que isso seja assim. Endereço físico é aquele que uma máquina na rede local usa para falar com outra máquina na rede local. Endereço lógico é aquele que seu navegador web favorito usa para acessar um site interessante, pois seu navegador web precisa enxergar mais do que apenas máquinas locais: servidores web estão espalhados pelo mundo todo.

2.5. ARQUITETURA CLIENTE-SERVIDOR

Todo mundo é cliente de alguém. Pode parecer uma frase polêmica, mas é verdade. E isso se aplica às redes também: a máquina que solicita algo é a cliente, e a que provê, o servidor. A sua máquina (cliente) atravessa a internet, até encontrar a máquina que provê o serviço desejado. Considere a figura abaixo:

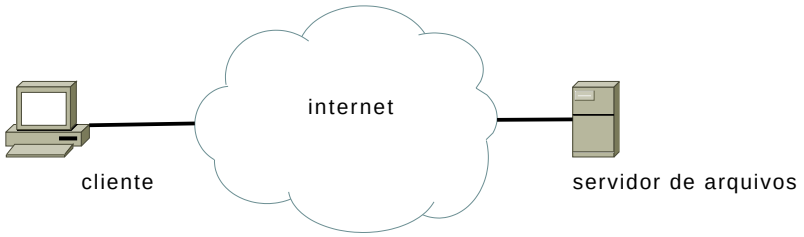

Figura 2.6. Uma máquina é a cliente, e a outra, o servidor.

Na figura, a máquina com a palavra "cliente" é a cliente (dã!). Sim, isso mesmo. Valente, intrépida e afoita, avança pelos sete mares em busca do tesouro escondido. Todavia, fique atento para o fato de que as máquinas não são apenas clientes, elas são clientes **de alguma coisa**. Com os servidores, o mesmo acontece: são servidores de alguma coisa. Na figura acima, por exemplo, temos uma máquina que é cliente de arquivos (ou seja, usa um programa que solicita uma conexão com um servidor de arquivos), e a outra máquina é o servidor de arquivos. Para ser mais específico, a verdade é que a máquina em si não é cliente ou servidora de nada; os programas que rodam nelas é que assumem o papel de cliente ou servidor.

Podemos ter um caso em que uma única máquina é cliente de duas coisas; por exemplo, cliente de arquivos e cliente web - o que significa que a máquina roda um aplicativo que conecta-se a um servidor de arquivos, e outro aplicativo que conecta-se a um servidor web, conforme figura abaixo:

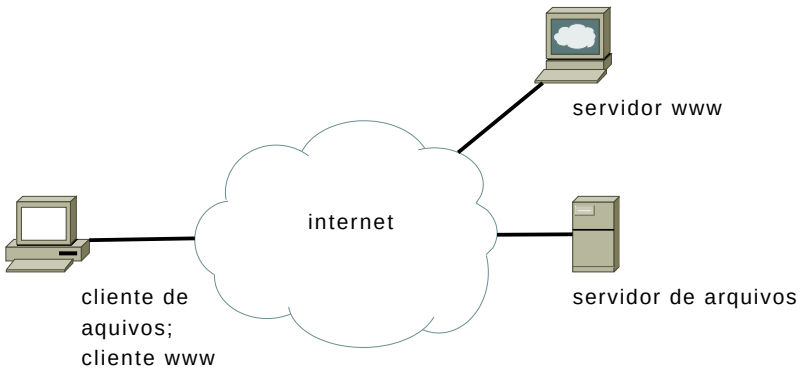

Figura 2.7. Uma máquina que é cliente de vários serviços.

Simplificando: uma única máquina pode ter vários programas clientes. Da mesma forma (embora isso não seja muito comum, nem muito recomendável), podemos ter uma máquina que é servidora de vários serviços (olha a redundância), isto é, roda vários programas servidores. Como um garçom que além de servir pizza, limpa a mesa e varre o chão após a festa.

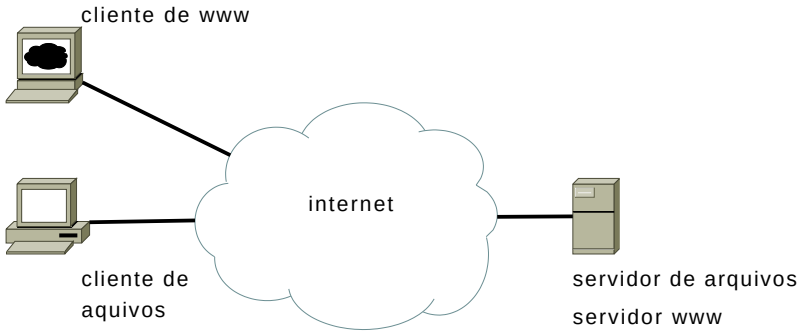

Figura 2.8. Uma máquina que é servidor de vários serviços.

Obviamente, uma máquina que tenha programas servidores tem a possibilidade de prover o serviço (ou os serviços) para várias máquinas. Ao mesmo tempo! Depende da capacidade do servidor; ou você pensa que é o único usuário conectado ao bate-papo nas madrugadas de sábado?

Preciso dizer também que é possível uma máquina cliente ser servidora. Por exemplo, em uma mesma máquina podem estar rodando um programa servidor web, e um cliente de email. Além disso, um servidor não precisa necessariamente estar na internet. Podemos ter servidores na rede local também.

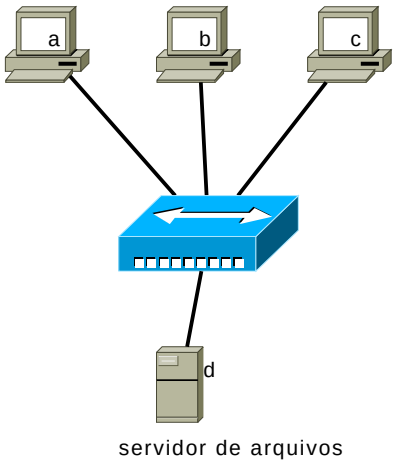

Figura 2.9. Um servidor na rede local.

Onde esses programas clientes e servidores operam? Operam na camada de aplicação, a camada que fica no topo da arquitetura TCP/IP.

2.6. Camada aplicação

Aplicações de rede são importantes para o funcionamento da mesma; as aplicações são a parte que interessa ao usuário. Entre as aplicações clientes (ou seja, que solicitam algo), temos os conhecidos navegadores web (Firefox, Opera, Safari, Konqueror, Internet

Explorer), programas de mensagem instantânea (GoogleTalk, MSN), clientes de email (Thunderbird, KMail, Outlook), e uma quase infinita quantidade de programas empresariais que conectam-se a uma máquina central para ler ou guardar informações.

Cada tipo de aplicação cliente exige um tipo de servidor. Por exemplo, para que você possa acessar uma página da internet com seu cliente web favorito (como o Firefox, por exemplo), é necessário que a máquina com a qual você se conecta esteja rodando um servidor www (como o Apache). Não adianta a máquina estar ligada, ou estar com outro servidor; é preciso o tipo específico de servidor para o tipo específico de aplicação cliente. Não se compra pão em açougue. Não se acessa páginas da web com um cliente de mensagens instantâneas.

Como se dá a comunicação entre aplicações de rede? Como a aplicação cliente fala com a aplicação servidora? Olhando a figura abaixo, você já pode ter uma ideia de como isso acontece: as camadas de aplicação das duas máquinas conversam entre si; para que isso aconteça, a máquina que envia informação desce com os dados da camada aplicativo para a camada física, e a máquina que recebe sobe com a informação. Essa história você já conhece.

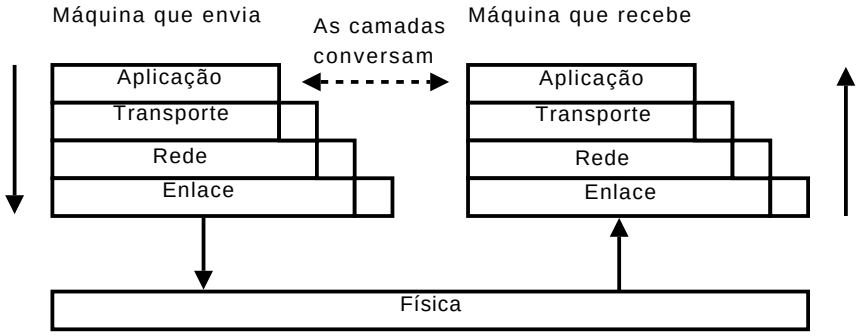

Figura 2.10. Comunicação entre aplicações através da arquitetura TCP/IP.

Você entende que as camadas equivalentes em máquinas diferentes conversam? Lembre-se: a informação que uma camada adiciona pode ser lida somente pela mesma camada da outra máquina. Assim, somente a camada aplicação da máquina que recebe conseguirá interpretar a informação que a camada aplicação da máquina que envia escreveu.

2.7. Camada transporte

Abaixo da camada aplicação, temos a camada transporte. Ela é a responsável não pelo transporte físico da informação (esta é uma função da camada física), e sim, pelo transporte lógico; na verdade, hum... digamos que esta camada faz o encaminhamento da informação da forma certa e para a aplicação correta.

Considere uma máquina que possua, neste exato momento, três programas clientes abertos: um programa que baixa arquivos de música, um programa de email e um navegador web. Todos eles estão em perfeita atividade, saudáveis etc. A máquina cliente, dinâmica como é, acessa três servidores distintos. Agora pense comigo: as informações vêm por um único meio, certo? Ou seja, os quadros das três aplicações vêm por um mesmo enlace, entrando na placa de rede da máquina. Após isso, a camada enlace interpreta o quadro, e passa para a camada rede. A camada rede também trabalha com o pacote e sobe com ele. E agora? Se não existisse a camada transporte, e os dados fossem jogados direto na camada aplicação, teríamos um erro muito estranho: as aplicações receberiam os dados de outras aplicações.

Porém, e Beethowen vai ter que concordar comigo (aquele surdo! nunca me ouve), a camada transporte está lá, bem vestida, olhando para você com um olhar atraente. Ela sabe dividir as coisas... além de ótima cozinheira, quando recebe os dados da camada rede, analisa as dados da camada transporte (que, adivinha, foi a camada transporte da máquina originária quem escreveu), e envia os dados da aplicação para o programa correto!

Observe a figura:

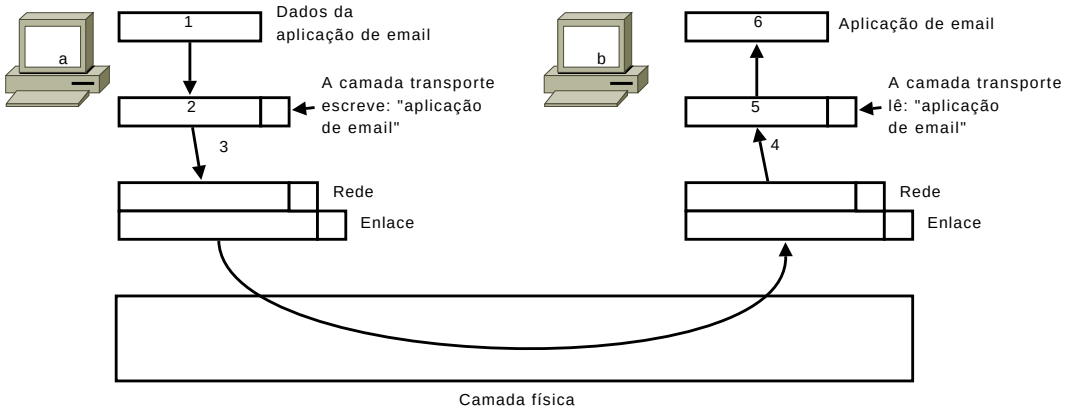

Figura 2.11. Camada transporte em ação.

Vejamos:

1. A máquina `a` envia informações de email para a máquina `b`. Por exemplo, o email "Olá, Boso!", que deverá surgir no programa de email na máquina `b`, deixando o usuário que o lerá, digamos assim, "bolado".

2. A camada aplicação envia os dados para a camada transporte. A camada transporte pega essa informação, e adiciona a ela sua própria informação de camada transporte; essa informação é um cabeçalho que diz "aplicação de email", indicando o que os dados contidos no interior deste "envelope" são.

3. A camada transporte desce, então, com os dados (o envelope) para a camada rede. A camada rede, por sua vez, faz o que tem de fazer, adicionando suas próprias informações, e enviando o pacote resultante à camada enlace. A camada também adiciona suas informações, e envia o quadro resultante para a camada física.

4. A camada enlace da máquina `b` recebe o quadro, interpreta-o, vê se ela é a destinatária do mesmo, e se for, retira as informações de enlace e passa o pacote resultante para a camada imediatamente superior, rede. A camada rede faz o que tem de fazer, retira os dados de camada rede do pacote e sobe com o "envelope" resultante para a camada transporte.

5. Agora é a hora da camada transporte trabalhar na máquina destinatária. Ela lê o conteúdo da informação de camada transporte (adicionada pela mesma camada na máquina remetente), e vê escrito: "aplicação de email". Assim, esta camada transporte sabe a que programa entregar os dados.

6. Finalmente, a camada transporte da máquina `b` retira as informações de camada transporte dos dados, e passa-os para a aplicação correta (ou seja, o cliente de email) na camada aplicação. O usuário lê, e fica bolado.

2.8. TRANSPORTE CONFIÁVEL E CONEXÃO

Além da função de encaminhar os dados corretamente, a camada transporte pode prover transporte confiável (ou não). Isso vai depender do protocolo usado. Na arquitetura TCP/IP, existem dois protocolos: o Protocolo de Datagrama do Usuário (UDP, de 1980[2.3]), e o Protocolo de Controle de Transmissão (TCP, de 1981[2.4]); o primeiro não provê um transporte confiável; ou seja, se alguma informação for perdida durante o trajeto, o protocolo não fará nada para corrigir essa perda. Já o TCP retransmite a informação se esta for perdida no caminho. Aí você pensa: "então o TCP é melhor que o UDP, pois garante que a informação chegará". Nem sempre. Se você estiver conectando-se a um banco, a informação precisa chegar integralmente ao destino; no entanto, se você está ouvindo uma música ou assistindo um filme pela internet, vai ser muito desagradável se a música ou o filme ficar pausando o tempo todo, devido à retransmissões: no caso da música e do vídeo, é preferível perder parte da informação. Veja a figura abaixo para compreender como acontece a transporte confiável de dados, usando o protocolo TCP:

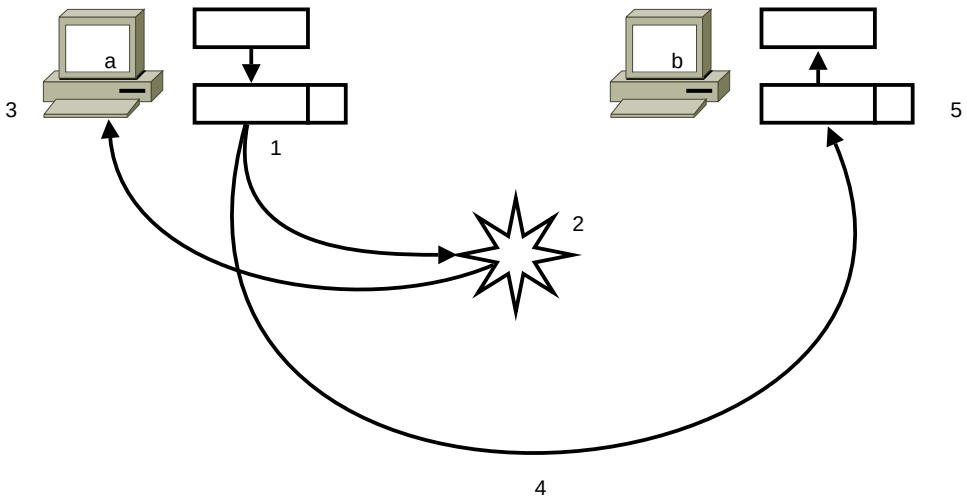

Figura 2.12. Transporte confiável.

1. A aplicação da máquina `a` escolheu o protocolo de camada transporte TCP para fazer a transmissão. Por isso, os dados da aplicação são passados para a camada transporte, e o protocolo TCP trabalhará com eles. Você já sabe o que acontece: a camada transporte desce com o envelope para a camada rede, que desce com o pacote resultante para a camada enlace, que por sua vez, desce com o quadro resultante para a camada física.

2. Aconteceu um imprevisto na camada física que impossibilitou o pacote de chegar ao destino. Não foi uma colisão, pois se fosse, a camada enlace retransmitiria o quadro; foi outra coisa, uma coisa qualquer que impossibilitou a chegada do quadro. Talvez um rato roeu o cabo em algum lugar (em roma).

2.3. [RFC 768].

2.4. [RFC 793].

3. A máquina `a` sabe que houve perdas na informação. Para ser mais específico, o protocolo TCP na camada transporte (o protocolo que enviou os dados), sabe que perdeu informação quando não recebe uma mensagem de confirmação da máquina `b`.

4. O que o TCP faz, então, na máquina `a`? Ele retransmite a informação.

5. O quadro chega à camada enlace da máquina `b`, que passa para a camada rede, depois para a camada transporte. A camada transporte passa os dados para a aplicação correta e informa à camada transporte da máquina `a` que recebeu a informação.

Outra característica dos protocolos de camada transporte é a orientação à conexão. UDP é um protocolo não orientado à conexão, o que significa que a máquina remetente não precisa de autorização da máquina destinatária para começar a enviar informações. Já o TCP precisa de autorização: é necessário que as máquinas se cumprimentem, sejam educadas, iniciem um diálogo e, aí sim, as informações poderão ser trocadas:

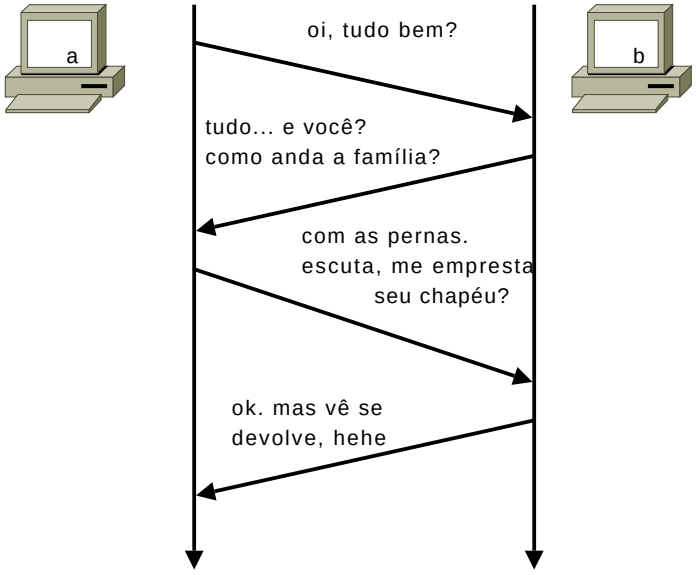

Figura 2.13. Conexão.

Faltou desenhar o chapéu voando de um hospedeiro para o outro na figura acima. Lembre-se que, quando duas camadas conversam, somente as mesmas camadas nas duas máquinas se entendem; na figura, é a camada transporte das duas máquinas que estão conversando e estabelecendo uma conexão de camada transporte; as camadas inferiores não entendem o que se passa, apenas cumprem suas funções. Abaixo, uma pequena tabela resumindo as características do TCP e UDP.

	UDP	**TCP**
Transporte	Não confiável	Confiável
Retransmite em caso de perda?	Não	Sim
Conexão	Não incia conexão	Inicia conexão
Velocidade comparada	Rápido	Lento
Indicado para	Áudio, vídeo	Email, web...

Tabela 2.2. Protocolos de camada transporte.

2.9. Conclusão

Neste capítulo, você viu a necessidade de dois endereços: um endereço físico e um lógico. Viu também que não é possível a máquina de uma rede local enxergar o endereço físico de uma máquina que esteja fora da LAN. Com isso, você teve uma visão geral das funções das camadas enlace e rede.

Além disso, você aprendeu que uma máquina pode ser tanto cliente quanto servidora, desde que os programas para isso estejam executando na máquina. Assim como uma máquina pode ter vários clientes, também pode ter vários servidores; cada programa cliente exige um tipo diferente de programa servidor.

O transporte das informações das aplicações é feita pela camada transporte; esta camada é responsável, entre outras coisas, pela entrega dos dados à aplicação correta na máquina destino, bem como a retransmissão dos dados para a máquina destinatária, se o protocolo assim configurado estiver sendo usado[2.5]. Fechamos, portanto, este capítulo, tendo visto de tudo um pouco.

2.10. Exercícios

Exercício 2.2. Diferencie LAN de WAN.

Exercício 2.3. Qual a diferença entre enlace LAN e WAN?

Exercício 2.4. Defina pacote e quadro.

Exercício 2.5. Por que são necessários dois endereçamentos?

Exercício 2.6. Verdadeiro ou falso:

a) Uma máquina pode ser cliente de um ou mais serviços.

b) Uma máquina pode acessar vários servidores.

c) Um servidor provê serviço a somente a uma única máquina.

d) Uma aplicação servidora provê somente serviço para um tipo de aplicação cliente.

e) Uma máquina não pode ter aplicativos clientes e servidores executando.

f) Um servidor não pode estar localizado na rede local.

Exercício 2.7. Cite exemplos de aplicações clientes.

Exercício 2.8. O que é transporte confiável de dados? Qual protocolo da camada transporte provê esse serviço?

Exercício 2.9. E o que é conexão, em se tratando de camada transporte?

Exercício 2.10. Verdadeiro ou falso:

a) A camada transporte da máquina destinatária recebe os dados da camada rede e passa para a camada aplicação.

b) A camada transporte tem a função de verificar se o endereço lógico do pacote é o da máquina destinatária.

2.5. Para detalhes de todas as funções assumidas pela camada transporte, em especial pelo protocolo TCP, consulte [RFC 793], pág. 3, tópico 1.5 - "Operation" em diante.

c) Vídeo é um exemplo de informação que poderia ser transmitida pelo protocolo UDP.

Capítulo 3

Fundamentos de comutação e roteamento

Neste capítulo você entenderá o que é comutação na camada enlace, e roteamento. A comutação na camada enlace funciona na camada enlace (dã!), enquanto o roteamento é uma função da camada rede. Existe comutação na camada rede também; todavia, como este capítulo só trata de comutação na camada enlace, usaremos simplesmente o termo "comutação" para designar isso.

3.1. Revisão

Você se lembra dos desenhos de redes locais vistos até o momento neste curso? Os computadores da rede local compartilham de um mesmo enlace, seja este um cabo único, ou um repetidor. Estudaremos mais sobre os tipos de cabo usados em redes locais na parte "Comutação na camada enlace". Você também se lembra do problema constante em redes locais, quando várias máquinas tentam falar ao mesmo tempo? Sim isso mesmo. Há o que chamamos de colisão de quadros no meio físico - lembrando que "quadro" é o nome que se dá aos dados transmitidos pela camada enlace.

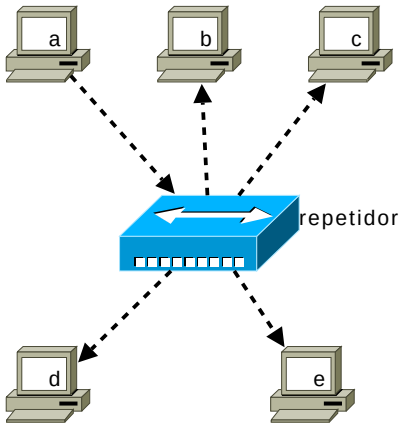

Figura 3.1. Uma máquina fala, todas escutam.

Em redes locais com enlace físico compartilhado, quando uma máquina fala, todas escutam. Por isso, é necessário que o protocolo de camada enlace (ou seja, protocolos que funcionam nas máquinas transmissoras e receptoras, na placa de rede dessas máquinas) dite as regras

para a boa educação na rede. Na rede local, o que vale é o endereço físico das máquinas: a camada enlace das máquinas se comunicam entre si por meio desse endereço.

Você também estudou sobre redes WAN. A internet é a maior rede WAN, porém existem outras, como redes de grandes companhias multinacionais. A internet interliga muitas redes; assim sendo, as máquinas de cada rede local não enxergam máquinas de outras redes por meio do endereço físico. É necessário um endereço lógico, um endereço de camada rede.

A grande maioria das redes hoje é baseada na arquitetura TCP/IP. Essa arquitetura é um conjunto de protocolos que estão localizados nas três camadas superiores: aplicação, transporte e rede - as duas camadas inferiores não são definidas pela arquitetura, embora a componham. Todas as máquinas do mundo que usam TCP/IP têm em comum esses protocolos das camadas superiores. Isso significa que uma aplicação de uma máquina no Japão é capaz de se comunicar com o servidor desta aplicação em uma máquina que esteja no Brasil, por exemplo.

Já com as camadas inferiores, é outra história. Os protocolos da camada enlace e também a camada física (cabos, repetidores e outros equipamentos) variam de rede para rede. Assim, temos neste mundo por aí afora protocolos TCP/IP sendo transportado sobre diversos protocolos de camada enlace, e meios físicos diversos. A arquitetura de redes locais mais usada no mundo chama-se Ethernet. Entretanto, nem todas as redes usam Ethernet; a própria WAN não utiliza esta arquitetura, visto que não é rede local. Veja um exemplo simples: sua rede local usa Ethernet nas camadas enlace e física, mas se você se conecta à internet por conexão assíncrona (modem, linha telefônica), o protocolo de camada enlace que você usa é o PPP - bem diferente do que você usa na sua rede local. Aí você pergunta: então como é possível que eu me conecte à uma aplicação na internet? A resposta é simples: embora as camadas enlace e física mudem, as camadas superiores permanecem as mesmas. A máquina com a qual você se conecta na internet possui protocolos de camada enlace diferentes da sua; mas o protocolo da camada rede (a camada do endereço lógico) é o mesmo protocolo que sua máquina usa: a camada rede daquela máquina é a mesma da sua. Camadas iguais em máquinas diferentes conversam entre si. O mesmo ocorre com a camada transporte e a camada aplicação.

3.2. O QUE É COMUTAÇÃO?

Vamos voltar ao maravilhoso e utópico passado da telefonia[3.1]. Como aconteciam as coisas? Você tinha uma senhora que fazia a comutação manual dos circuitos. Ou seja, você ligava para a central telefônica, a senhora com voz bonita atendia você, que dizia: quero falar com o Papa. Então, a senhora ligava o seu circuito ao circuito do Papa e você podia falar com ele: marcar uma festa de arromba na sua casa, coisas assim. Todos os seus colegas ligavam para o Papa, pois o Papa é pop. Isso é comutação: o ato de a senhora fechar o circuito entre você e o Papa. A senhora, no caso, era a comutadora.

3.1. Ver excelente discussão sobre o funcionamento das redes de telefonia em [Davidson, Peters] págs. 36-43.

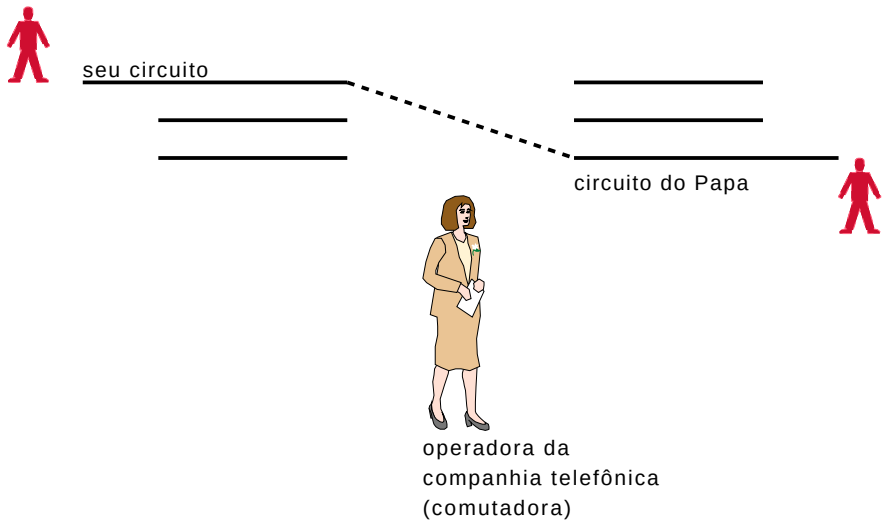

Figura 3.2. Comutação na época da vovó.

Hoje as coisas mudaram: você já não pode ligar para o Papa como ligava antigamente; ele é um cara meio ocupado... cortou o cabelo e vendeu o Opala. E também, aposentaram todas as senhoras que faziam as ligações. Entretanto, os comutadores continuam por aí, só que não fazem compras em supermercados: são aparelhos eletrônicos. A ideia da comutação é muito simples: interligar duas máquinas (no caso de comutadores de redes) e não permitir que a informação trocada por elas vaze por toda a rede local. Se você quer falar com o Papa, somente o Papa vai ouvir você falar; se a máquina `a` quer falar com `e`, somente `e` ouvirá a máquina falar; e o restante da rede ficará livre para quem quiser conversar com outra máquina. Veja a imagem abaixo, e observe que o desenho que representa o comutador é diferente daquele que representa um repetidor.

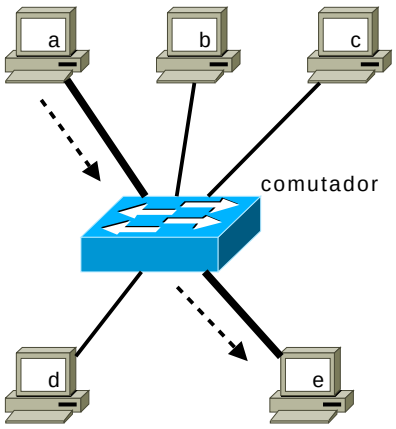

Figura 3.3. Um comutador na rede local.

O comutador é capaz de comutar vários circuitos ao mesmo tempo. Isso significa que várias máquinas podem falar, sem que haja colisão de quadros.

DEFINIÇÃO 3.1. *Comutação de quadros. Comutar quadros é o mesmo que criar um caminho, dentro do comutador, entre a máquina de origem e a máquina de destino; os quadros passam por este caminho específico, não ecoando por toda a rede.*

Observe a figura abaixo:

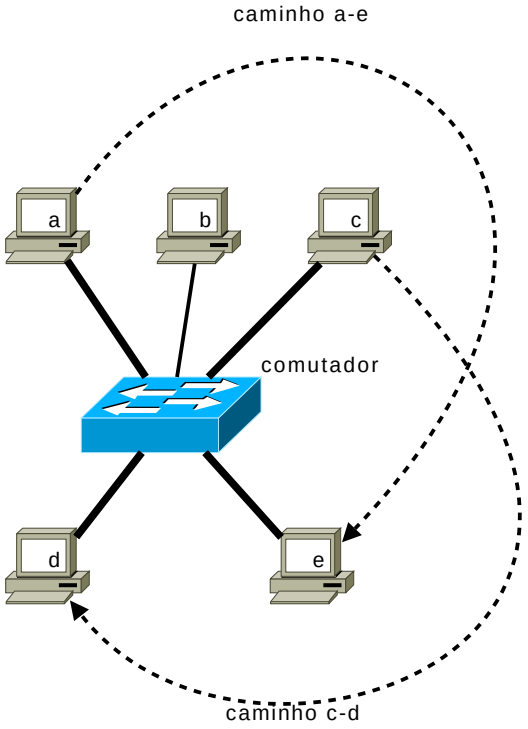

Figura 3.4. Várias máquinas falando ao mesmo tempo.

Na figura, as linhas tracejadas representam os caminhos, ou enlaces virtuais, de uma máquina para outra. Observe que a máquina `a` conversa com a máquina `e` por meio do caminho `a-e`; e, ao mesmo tempo, a máquina `c` conversa com a máquina `e` por meio do caminho `c-d`. Todavia, os quadros passam todos pelo comutador, que não se confunde: trabalha como um polvo em um restaurante, servindo vários pratos ao mesmo tempo com seus tentáculos (péssima metáfora!).

Em uma rede que usa comutador, e também tem apenas uma máquina ligada por porta no roteador, não é necessário um protocolo na camada enlace das máquinas para controlar a educação das máquinas: o próprio comutador trata de fazer isso. Estudaremos muito mais sobre comutadores neste curso; para o momento, basta você compreender que o comutador tem a função de `comutar` quadros da camada enlace, na rede local. E comutar é a mesma coisa que criar um caminho entre a origem e o destino, de modo que os quadros não ecoem por toda a rede.

3.3. O QUE É ROTEAMENTO?

Não, roteamento não é a arte ou ciência de arrotar. Isso se chama arrotamento. Roteamento é a arte ou ciência de criar rotas. Roteamento ocorre na camada rede, e trabalha com endereços lógicos. Imagine duas redes em prédios distintos. Você sabe que uma máquina da rede local não enxerga o endereço físico da máquina. Para que a comunicação seja possível, você coloca um roteador para interligar as duas LAN's. Confira na figura abaixo:

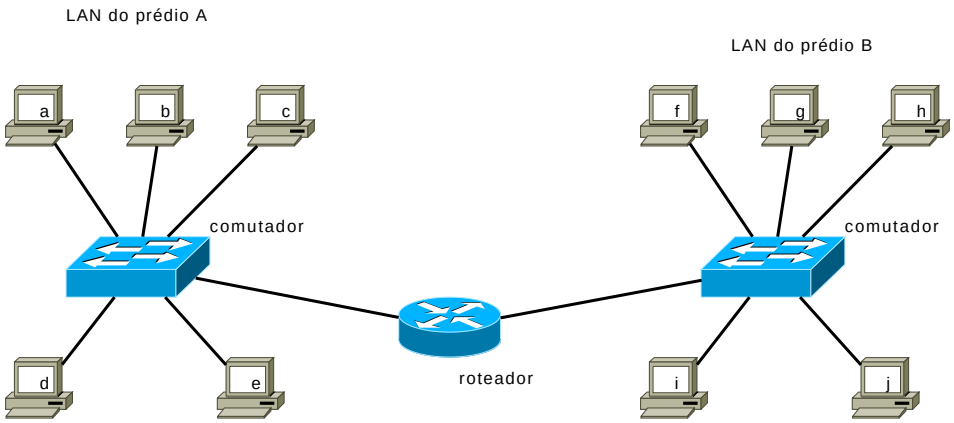

Figura 3.5. Duas LAN's ligadas por um roteador.

Analisando:

Primeiro, você sabe que a máquina `a` enxerga o endereço de camada enlace da máquina `e`. Assim, se a máquina `a` quisesse falar com a máquina `e` usando o endereço físico desta, a comunicação seria possível.

Segundo, se alguma máquina do prédio `A` quisesse falar com outra máquina do prédio `B` usando o endereçamento de camada enlace, não seria possível.

Exercício 3.1. Por quê não seria possível a máquina `a` falar com `j` usando o endereçamento físico?

Terceiro, se a máquina `a`, do prédio `A`, tentasse falar com a máquina `j`, do prédio `B`, usando endereçamento de camada rede, seria possível, e você sabe por quê: embora os enlaces mudem, e os protocolos de camada enlace também, o protocolo de camada rede não muda, e camadas iguais em máquinas distintas conversam entre si. Assim, as duas máquinas poderiam comunicar-se usando o endereçamento lógico. Para conectar essas duas redes, usa-se o roteador, pois ele conecta redes diferentes (diferente do comutador, que conecta máquinas de uma mesma rede).

O roteador opera na camada rede, o que significa que ele encaminha pacotes (lembre-se: pacotes são dados da camada rede). Você deve estar pensando que a internet deve estar cheia de roteadores, e é verdade; senão, seria impossível você conectar-se à uma máquina na internet. Estudaremos muito mais sobre roteadores neste curso. Para o momento, basta você saber qual a função básica de um roteador: interligar redes distintas, encaminhando[3.2] pacotes entre elas.

Definição 3.2. *Roteamento. Roteamento é uma função de camada rede que tem por objetivo encaminhar pacotes de uma para outra rede. Roteador é o equipamento que assume essa função, interligando redes distintas.*

3.4. Formatos de endereçamento

Toda máquina da rede possui ao menos um endereço físico, e outro lógico, e os motivos você já sabe quais são. Até o momento, usamos nomes como a, b, c etc para simplificar as coisas; entretanto, os endereços não são esses: eles possuem um formato predefinido.

Primeiro, o endereço de camada enlace. Onde ele é armazenado? Na placa de redes do computador. Por quê? Porque é a placa de rede que possui os protocolos de camada enlace. Por quê? Porque assim foi definido pelo ciclope que vive em Marte. Quantas perguntas!

3.2. "Encaminhar" é algo diferente de "rotear", como ficará claro em neste curso. Consulte [Kurose & Ross] págs. 236, 237.

A grande maioria das redes locais neste planeta usa Ethernet. O que é Ethernet? Ethernet é tanto o tipo de rede, quanto o protocolo que essa rede usa; existem, contudo, muitos outros protocolos para redes locais, que são menos usados do que Ethernet. E existem protocolos de camada enlace próprio para WAN's, por isso, fique ciente de que Ethernet não é a única coisa que existe no mundo.

Um endereço Ethernet (ou seja, endereço de camada enlace) é composto por doze dígitos hexadecimais. Dizemos que eles são hexadecimais (em vez de decimais) pois podem assumir dezesseis valores: 0, 1, 2, 3, ..., 9, A, B, C, D, E e F. Para facilitar as coisas, decidiu-se (quem decidiu? resposta: o famigerado ciclope que vive em Marte) agrupar os dígitos em dois. Eis um exemplo de endereço físico: 00:1d:92:a5:69:f4.

Cada dígito equivale a 4 bits, pois são necessários 2^4 valores para formar um dígito hexadecimal. Portanto...

Exercício 3.2. Qual o tamanho, em bits, do endereço Ethernet de camada enlace?

a) 6 bits

b) 24 bits

c) 36 bits

d) 48 bits

e) 64 bits

O endereço físico muitas vezes é chamado de endereço MAC. O MAC é gravado na placa de rede, de forma que não pode ser mudado. Claro, existe um truque para enviar quadros com outro endereço MAC de origem, porém o endereço gravado ***na*** placa não pode ser trocado. Cada placa de rede tem um endereço único: os seis primeiros dígitos indicam o fabricante da placa, e os três últimos são dígitos gerados pelo fabricante para diferenciar uma placa da outra. Desse modo é possível que hajam muito mais endereços do que o número de placas de rede existentes no mundo.

Nunca esqueça de que este endereço físico de camada enlace pertence ao protocolo Ethernet; existem outros protocolos de camada enlace, que podem possuir formatos de endereço diferentes.

Agora vamos falar de endereço lógico, que se localiza na camada rede. As redes TCP/IP - e, portanto, a internet - usam o protocolo IP na cama de rede. Existem hoje duas versões do protocolo IP: a versão 4 e a versão 6. A versão 4 ainda continua sendo muito usada, mas está perto do fim da vida; a versão 6 é o futuro, porém ainda está pouco implementada no mundo real. Para esta explicação, vamos usar o IP versão 4, pois em todo lugar que você for e que haja redes TCP/IP, esta é a versão usada. Abordaremos IP versão 6 em um momento posterior deste curso.

Pois bem. Um endereço IP (versão 4) é composto de um número decimal composto de quatro octetos. Dizemos que eles são octetos porque ocupam oito bits. Os valores possíveis para cada octeto são 2^8, ou seja, 256 valores. São quatro octetos, e não um só. Os valores vão de 0 a 255 (e não, como você pode estar pensando, de 1 a 256). Existem algumas regras para formação de endereços IP, que estudaremos na parte deste livro que fala sobre roteamento. Eis um exemplo de endereço IP válido: 192.168.0.1.

Os octetos, como você pode observar, são separados por ponto. "Mas por quê? Por que não separaram com uma tralha?", pergunta você. É aquela velha questão do ciclope marciano. Abaixo, a figura mostra máquinas em uma rede local com endereços MAC e endereços IP válidos. Também mostra um servidor na internet. Observe que a máquina que conecta a rede à internet possui dois endereços IP: um para a rede interna, e outro, que pode ser

visto a partir de fora da rede. Embora esta máquina possua, também, dois endereços de camada enlace, colocamos apenas o endereço Ethernet para a rede local, outros endereços de camada enlace (especificamente, para WAN) ainda não foram explicados neste curso.

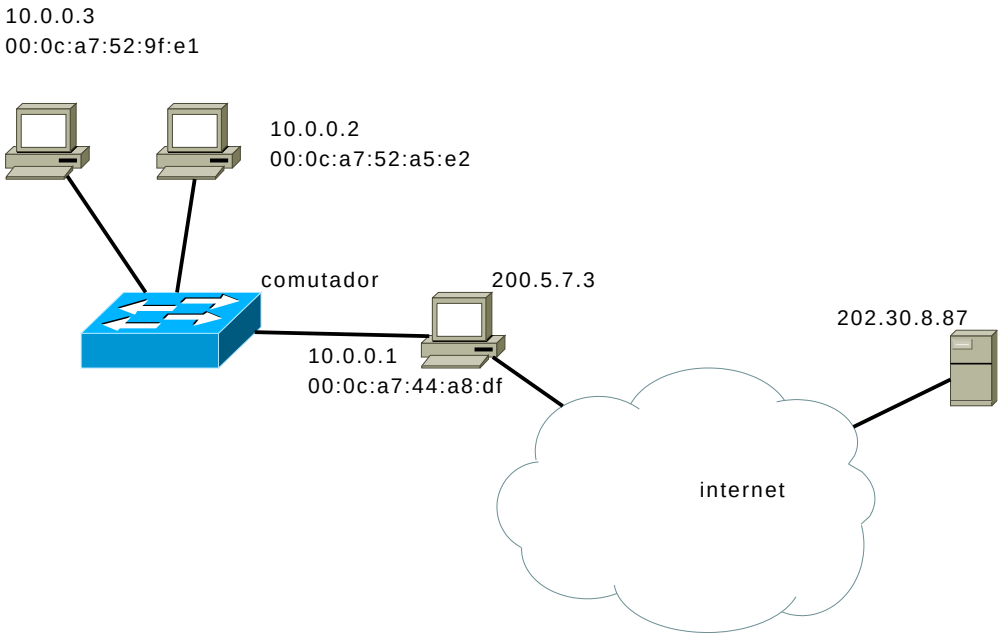

Figura 3.6. Demonstração de endereçamentos físicos e lógicos.

Observe também que na figura acima, as máquinas da LAN têm um endereço IP com o mesmo formato - ou seja, 10.x.x.x. Isto faz sentindo, pois como as máquinas pertencem à mesma rede, devem possuir endereços IP's com o mesmo formato; diferente do endereço de camada enlace, endereços de camada rede podem ser alterados.

3.5. Backbone

Observe a figura abaixo:

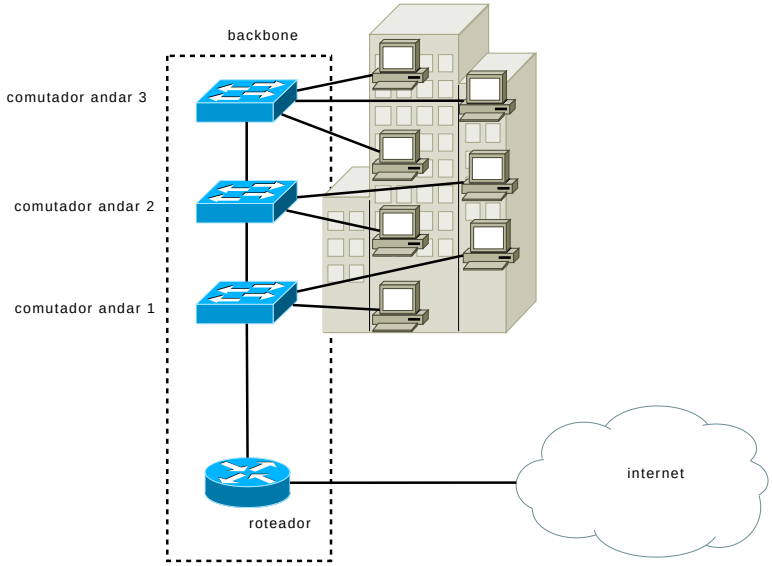

Figura 3.7. Desenho do backbone de um campus.

Backbone é simplesmente uma palavra muito intelectual em língua estrangeira (para você chamar atenção das gringas....), que designa a área em que o **tráfego** da rede se concentra.

DEFINIÇÃO 3.3. *Backbone. Backbone é a parte da rede onde trafegam grandes quantidades de dados. Em uma rede de campus, backbone pode também designar a área física onde os aparelhos comutadores e roteadores se localizam.*

Na figura acima, temos um prédio de três andares, cada andar com uma LAN. Obviamente, o comutador (ou comutadores) da cada andar estão concentrados em um local específico, talvez dentro de um armário etc. Para que haja interconexão entre os vários andares, podemos usar comutadores de camada enlace para comutar dados entre eles; e para que o prédio se conecte à internet, usamos um roteador na saída. Esta área específica onde estão os comutadores e o roteador (o roteador está do lado de fora do prédio na figura, apenas para ilustrar o fato; ele fica dentro do prédio) é chamada de espinha dorsal da rede, ou *backbone* (osso de trás; que vocabulário pobre!). Então, quando lhe disserem que há um problema no backbone da internet, significa que o mundo acabou.

O backbone precisa ser implementado de tal forma, que haja o mínimo de gargalos possíveis. Um gargalo acontece quando o tráfego no enlace físico é maior do que a capacidade do mesmo. Por exemplo, um enlace com capacidade para transportar 10Mbps (megabits por segundo) recebe uma requisição para transportar 10Mb de, por exemplo, 5 máquinas da rede. São 50Mb concorrendo por um enlace de 10Mbps; alguém sairá prejudicado.

3.6. CONCLUSÃO

Neste capítulo, estudamos os fundamentos de comutação e roteamento. Você viu que comutação é uma função da camada enlace, exercida por comutadores; os comutadores das famosas redes Ethernet (padrão mais usado no mundo para LAN's) recebem quadros Ethernet e encaminham esses quadros à máquina de destino, com base no endereço físico, ou endereço de camada enlace. O endereço Ethernet é formato por doze dígitos hexadecimais (que vão de 0 à F), separados em duplas para facilitar a leitura. Um exemplo de endereço físico válido é 00-0a-3c-4d-ee-f4. Esse endereço possui 48 bits, e os primeiros seis dígitos indicam quem é o fabricante do comutador. Podem ser chamados também de endereços MAC. Não existem dois endereços MAC iguais no mundo, e não é possível mudá-lo na placa de rede.

Estudamos também sobre roteadores. Roteadores são dispositivos que operam na camada rede da arquitetura TCP/IP; assim, eles encaminham pacotes, com base no endereço lógico, ou endereço IP. Um roteador interliga duas ou mais redes distintas, e pode servir também para interligar uma rede local à internet. O endereço IP possui quatro octetos de 8 bits (ou seja, o tamanho total é de 32 bits); cada octeto pode assumir valores que vão de 0 à 255, seguindo algumas regras, que ainda não estudamos. Um exemplo de endereço IP válido é 10.5.4.230.

Backbone é o núcleo da rede; deve-se planejar com cuidados backbones de grandes redes, pois é neles que o tráfego pesado passa. Enfim, neste capítulo fizemos uma prévia de tudo que iremos estudar neste curso. A partir de agora, nos aprofundaremos em comutação e roteamento, estudando protocolos e funcionamento das redes sob diversas circunstâncias.

3.7. EXERCÍCIOS

Exercício 3.3. Defina comutação de camada enlace.

Exercício 3.4. Defina roteamento.

Exercício 3.5. Verdadeiro ou falso (comutação):

a) Comutadores trabalham com pacotes da camada rede.

b) Comutadores trabalham com quadros da camada enlace.

c) Comutadores são dispositivos que assumem funções da camada transporte.

d) Comutadores são dispositivos que assumem funções da camada enlace.

e) A camada enlace trabalha com endereço físico.

f) Um exemplo de endereço válido de camada enlace é 10.13.2.5.

g) O protocolo mais usado em redes locais, é o protocolo Ethernet.

Exercício 3.6. Defina backbone.

Exercício 3.7. Observe a figura abaixo:

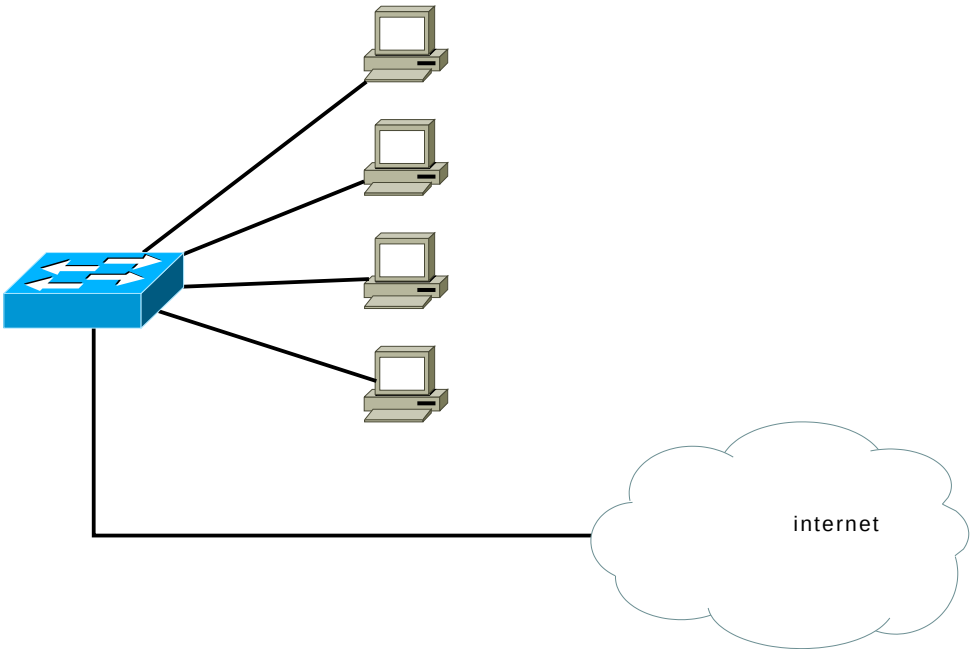

Figura 3.8. Rede local conectada à internet. Será?

Essa rede funciona?

a) Sim, pois a todas as máquinas da rede local estão conectadas ao roteador, que por sua vez está conectado à internet.

b) Sim, pois o comutador, embora não seja um roteador, provê acesso à internet para as máquinas da rede local.

c) Não, pois comutadores nem roteiam pacotes, e nem podem assumir função de roteadores.

Parte II

Redes Locais

Capítulo 4

Camada física das redes locais cabeadas

4.1. Introdução

Já foi dito anteriormente que redes de computadores não é o mesmo que cabeamento. Cabeamento inclui redes, mas não é a mesma coisa. Para adquirir um bom conhecimento de cabeamento, seria necessário um curso próprio para isso. Esse curso trata de redes. Todavia, é necessário ter uma noção de cabeamentos usados em redes locais, pois algum dia você certamente precisará lidar com eles, e o cabeamento faz parte da camada física das redes TCP/IP.

Neste capítulo, estudaremos a camada física das redes locais cabeadas. Dizemos "cabeadas" porque também existem redes locais não cabeadas (sem fio), que estudaremos separadamente neste curso. Cabeamentos usados em redes de longa distância também serão tratados em momento posterior.

4.2. Transmissão na camada física

Assuma a partir de agora que toda vez que você ler "camada física", isso significa que estamos fazendo referência aos cabos de rede. Isso é assim porque este capítulo trata explicitamente da camada física das redes locais (LANs) cabeadas.

Primeiramente, uma revisão. Você lembra-se de como acontece a transmissão de dados pela rede? O que acontece quando uma máquina quer transmitir dados da aplicação, isto é, *datagramas*?

Tente responder sem ler abaixo. Tente mais um pouco.

O datagrama de camada aplicação é enviado para baixo na pilha de protocolos TCP/IP. A camada imediatamente abaixo da aplicação é a transporte. O datagrama de aplicação é portanto encapsulado em um *segmento*. O segmento de transporte é enviado para a camada rede e encapsulado em um *pacote*. Por sua vez, o pacote é enviado para baixo, para a camada enlace, sendo encapsulado em um *quadro*. Este quadro precisa ser enviado à camada física, ou seja, ao cabo ligado à placa de rede da máquina. Na outra ponta do cabo temos um comutador provavelmente, pois hoje em dia este é o dispositivo que concentra todo o cabeamento e interliga as máquinas.

Pois é exatamente este *envio à camada física, ou seja, à placa de rede da máquina* o objeto de estudo deste capítulo. A placa de rede converte o quadro Ethernet em bits, zeros e uns, sinais elétricos modulados que passam pelos fios de cobre de um cabo de par trançado.

Tenha calma. Veremos isso vagarosamente. Observe por agora a figura abaixo.

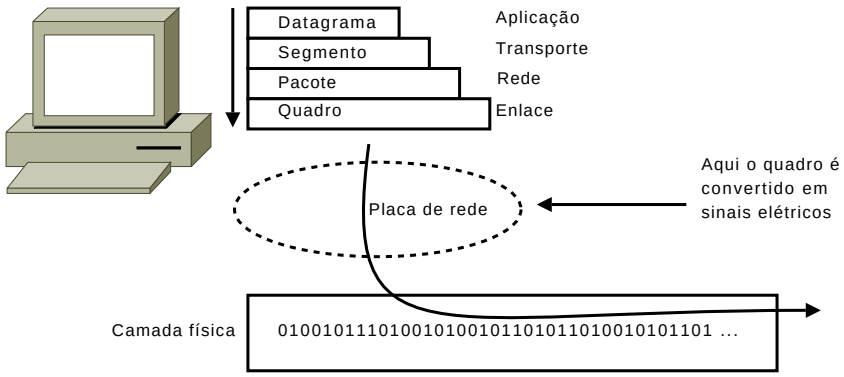

Figura 4.1. Placa de rede da máquina emissora convertendo um quadro em bits.

Até agora, nas figuras apresentadas anteriormente neste curso, você sempre via uma seta indo da camada enlace para a camada física diretamente. Porém, você pode notar que há uma "coisa", um dispositivo que tem a função de converter o quadro de camada enlace para bits de camada física. Por quê? Ora, quando um datagrama da aplicação é encapsulado em um segmento de transporte, ou quando um pacote de rede é encapsulado em um quadro de enlace, não ocorre conversão (mudança, tradução) de dados. O que ocorre é a adição de dados (lembre-se: estamos na máquina que envia), sem alterá-los[4.1]. Consegue compreender isso? Se, por exemplo, o pacote de rede contém a seguinte informação:

```
xxxx xxxx xxxx xxxx xxxx
```

Ele é encapsulado totalmente dentro de um quadro, que contém suas próprias informações. Suponha que a informação acima tenha sido encapsulada em um quadro; informações do quadro serão representadas com a letra y:

```
yyyy yyyy xxxx xxxx xxxx xxxx xxxx yyyy
```

Observe que os dados da camada rede (`xxxx xxxx` etc) não foram alterados. Foram encapsulados como estavam dentro do pacote. E mesmo que hipoteticamente os dados fossem alterados de uma camada para outra, essa conversão seria feita pelo sistema operacional. Ou seja, seria uma conversão lógica, não física.

Contudo, não é isso que acontece com dados que descem da camada enlace para a camada física. Primeiro porque os quadros enlace não são encapsulados. Segundo, porque eles realmente *são* convertidos. Terceiro, não é uma conversão lógica, e sim uma conversão física: quadros são convertidos em sinais elétricos.

Pense um pouco em como a informação pode ser representada. Por exemplo, um email. Um email pode ser representado por números. A letra a pode ser representada, por exemplo, por 10. A letra b por 15, a letra c por 20, a letra e por 25 e a letra t por 55. Assim, "abacate" pode ser representado por 10 15 10 20 10 55 25.

Prosseguindo com o raciocínio, a representação usada pelos computadores é binária: usa apenas 0 ou 1. Por exemplo, para representar a letra a poderíamos ter 00001010. A letra b poderia ser 00001111, a letra c, 00010100, a letra e, 00011001 e a letra t, 00110111. "abacate" poderia ser representado assim:

```
00001010 00001111 00001010 00010100 00001111 0110111 00011001
```

Agora, você deve compreender que qualquer informação da máquina transmissora que passe pela placa de rede é primeiramente convertida em binário. Ou seja, o quadro de camada enlace, ao passar pela placa de rede, é convertido em binário. "Ah", pensa você, "então isso

4.1. A excessão é o protocolo de camada enlace PPP, que coloca controles de escape no pacote de camada rede.

é simplesmente uma conversão lógica! A informação foi convertida de um modo para outro logicamente". Calma gafanhoto! A melhor parte vem agora.

A informação é convertida em binário, porém não fica armazenada na máquina. Esses 0s e 1s são enviados para o cabo de rede. E eles são enviados em forma de sinais elétricos, pois os fios do cabo são feitos de cobre e transportam somente sinais elétricos. Pense nisso.

A placa de rede sinaliza esses 0s e 1s não de forma lógica, como se estivesse salvando o arquivo resultante binário na memória do computador, e sim de forma física: esses bits 0s e 1s saem da placa de rede em forma de eletricidade. Um exemplo de codificação que as placas de rede poderiam usar é: o bit 1 é representado por um "choque" de 5v no cabo. Já o bit 0 é representado pela ausência de choque (0v), ou ainda, um choque de -5v.

Uma placa de rede é capaz de transmitir muitos bits por segundo dessa forma. Para você ter uma idéia, as redes Ethernet mais lentas que existem operam a 10Mbps, ou seja, 10 milhões de bits por segundo. Isso significa que em um único segundo, tal placa de rede é capaz de alternar choques de 5 e 0 (ou -5) volts *dez milhões* de vezes! Em um único segundo! E estamos falando das placas de rede *mais lentas*!

Podemos representar a informação na camada física por meio de uma figura que mostra o formato de onda dos sinais elétricos. O formato de onda digital tem apenas dois estados: 0 e 1, como já explicamos. Na figura abaixo, você pode ver a representação da informação 10101010 sendo transmitida pelo meio físico.

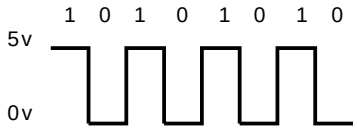

Figura 4.2. Transmissão em bits usando sinais elétricos.

Não entraremos em mais detalhes sobre codificação de camada física. Para os propósitos deste curso, o que você aprendeu é o suficiente para prosseguir sem problemas. Os bits da camada física em redes locais cabeadas atuais são trasmitidos em cabos de par trançado.

4.3. O cabo de par trançado

Na figura abaixo, você vê os fios de um cabo de par trançado desencapado. São 8 fios no total, trançados dois a dois. Cada fio possui uma cor distinda (ou duas cores), para que possam ser facilmente distinguidos dos outros.

Figura 4.3. Representação de cabo de par trançado retirado da Wikipedia.

Este é o cabo de par trançado sem blindagem (*Unblinded Twisted Pair - UTP*) usado nas redes locais. Apesar de ter oito fios, na maioria das vezes só usamos quatro deles como veremos adiante. Todos os fios têm a mesma capacidade de transmissão, assim sendo, você pode usar qualquer dos fios para transmitir ou receber, porém um único fio não pode transmitir e receber ao mesmo tempo. "Então isso quer dizer que uma máquina não pode transmitir e receber ao mesmo tempo?". Bom, se alguns fios (dois por exemplo) forem usados para transmitir e outros para receber, sim, é possível. Contudo, se a máquina usa todos os fios do cabo, então apenas é possível transmitir quando os fios estiverem desocupados; caso contrário, teremos uma colisão.

Assuma que em redes locais que usam esse cabo, a máquina usa dois fios para transmitir, e dois para receber. É isso que acontece na vida real.

4.4. Transmissão nos fios do cabo de par trançado

Não importa a ordem dos fios, se o primeiro fio é azul ou verde etc; a placa de rede sempre tratará o primeiro fio como o fio 1, o segundo como fio 2 e assim por diante. Todos os fios têm capacidade igual. Mais tarde veremos algumas normas que ditam quais devem ser a ordem dos fios; contudo, mesmo esta norma não sendo seguida (o que não é aconselhável), a placa de rede irá transmitir e receber pelos fios correspondentes.

Para início de conversa, analise a figura abaixo.

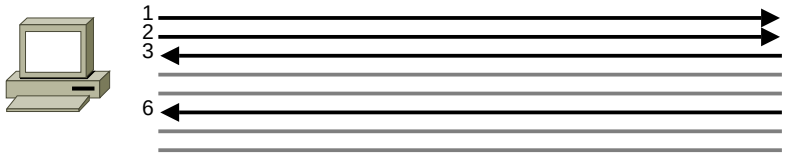

Figura 4.4. Fios úteis usados no cabo de par trançado.

Destacamos quatro fios na figura acima. Os outro quatro são irrelevantes para as transmissões: se você cortá-los, a transmissão ocorrerá sem problemas, pois só quatro fios são usados. Os fios 1 e 2 são usados para *transmitir, eviar* bits. Os fios 3 e 6 são usados para *receber*. Os fios 4, 5, 7 e 8 são irrelevantes.

Uma coisa não está óbvia na figura acima. Se esta máquina está conectada diretamente a outra máquina semelhante a ela mesma, então esta outra máquina também vai transmitir pelos fios 1 e 2. E se as duas transmitirem por esses fios, então haverá uma colisão, certo?

É isso que aconteceria se as máquinas fossem ligadas com esse fio. A comunicação seria impossível. Os fios 1 e 2 transmitem; os fios 3 e 6 recebem.

4.5. Transmissão com fios trocados

Para que duas máquinas semelhantes se comuniquem, elas não podem transmitir pelos mesmos fios. É necessário que os fios sejam *trocados*. Ou seja, os fios que em uma ponta servem para transmissão, na outra ponta devem ser os de recepção. Em outras palavras:

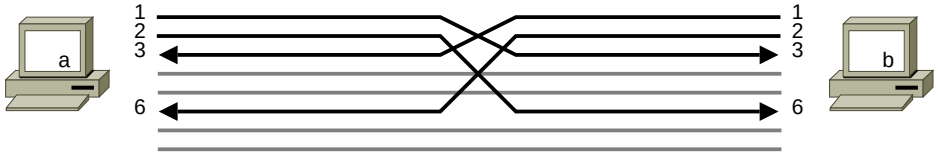

Figura 4.5. Transmissão entre dois computadores.

Os fios 1 e 2 de um lado do cabo são os mesmos fios 3 e 6 do outro lado. Certo? Mas isso e só para casos em que máquinas semelhantes são ligadas entre si. O que queremos dizer com máquinas semelhantes? Queremos dizer máquinas que enviam pelos fios 1 e 2, e recebem pelo 3 e 4. Entre essas máquinas estão os *computadores* e os *roteadores*. Ao fazer ligação entre essas máquinas, o cabo deve ter os fios trocados (cross-over).

4.6. Transmissão com fios diretos

Se as máquinas estão ligadas a um comutador, os fios não precisam ser trocados: podem ser fios diretos (straight-over), pois o comutador faz a troca internamente. Isso significa que os dados recebidos pelos fios 1 e 2 pela porta de origem são enviados pelos fios 3 e 6 na porta de destino, conforme mostra a figura abaixo.

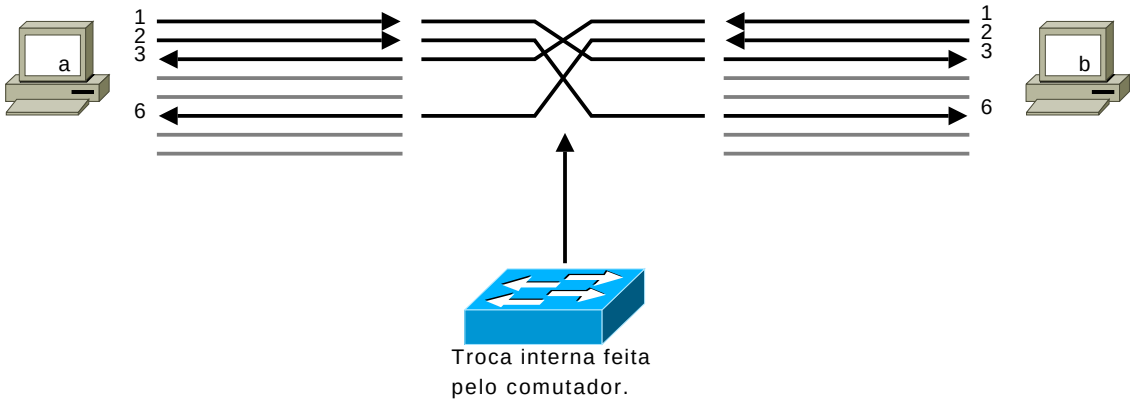

Figura 4.6. Transmissão entre uma máquina, um comutador e outra máquina.

Máquinas, sejam elas computadores ou roteadores, interligadas a comutadores, usam transmissão com fios diretos. Comutadores recebem pelos fios 1 e 2 e transmitem pelos fios 3 e 6: ou seja, operam de forma inversa se comparado às outras máquinas. Comutadores ligados entre si devem usar um cabo com fios invertidos (cross-over), pois o normal de um comutador é receber pelos fios 1 e 2. Pense sobre isso.

Tipo da máquina	Tipo da máquina	Tipo de cabo usado
Envia por 1 e 2	Envia por 1 e 2	Fios trocados (cross-over)
Envia por 1 e 2	Envia por 3 e 6	Fios diretos (straight-over)
Envia por 3 e 6	Envia por 3 e 6	Fios trocados (cross-over)

Tabela 4.1. Resumo da ligação entre máquinas

Máquina origem	Máquina destino	Tipo de cabo usado
Computador	Computador	Fios trocados (cross-over)
Computador	Roteador	Fios trocados (cross-over)
Computador	Comutador	Fios diretos (straight-over)
Roteador	Roteador	Fios trocados (cross-over)
Roteador	Comutador	Fios diretos (straight-over)
Comutador	Comutador	Fios trocados (cross-over)

Tabela 4.2. Exemplo de ligações entre máquinas

Isso pode ser meio difícil de decorar no começo, mas não se preocupe, as coisas vão piorar, porque pelo menos até agora é uma questão de lógica saber o tipo de cabo usar: basta saber por que fios a máquina envia e por quais ele recebe bits. Difícil de decorar é a ordem dos fios baseados na cor. Difícil, porém essencial.

4.7. As cores dos fios

Para tornar mais lógica a memorização, primeiro vamos focar nos fios relevantes, aqueles que são usados para alguma coisa em redes locais. Depois focaremos nos fios menos relevantes, aqueles que nem sequer são usados.

Os fios usados para transmissão são os verdes e os laranjas. O primeiro padrão é o EIA/TIA 568A. Por este padrão, o fio 1 é o verde e branco (ou verde claro). O fio 2 é o verde, o fio 3 é o laranja e branco, e o fio 6 é o laranja. Se você tem um computador e um comutador, você vai usar o padrão 568A em ambas as pontas do cabo: ou seja, transmissão por fios diretos.

O segundo padrão é o EIA/TIA 568B. Este padrão é usado para transmissões com fios trocados (cross-over). De um lado você tem o 568A, cujo fio 1 é o verde e branco, e o fio 2 é o verde. Pois é: como o 568B trata de ligações com fios trocados, os fios para transmissão não podem ser os mesmos do 568A. Assim, pelo 568B, o fio 1 é o laranja e branco, o 2 é o laranja, o 3 é o verde e branco e o 6 é o verde. Observe as tabelas abaixo.

1	Verde e branco
2	Verde
3	Laranja e branco
6	Laranja

Tabela 4.3. Padrão 568A

1	Laranja e branco
2	Laranja
3	Verde e branco
6	Verde

Tabela 4.4. Padrão 568B

4	Azul
5	Azul e branco
7	Marrom e branco
8	Marrom

Tabela 4.5. Os fios menos relevantes

Os fios menos relevantes sempre são os mesmos em ambos os padrões: o fio 4 é o azul, o fio 5 é azul e branco, o 7 é marrom e branco e o 8 é o marrom. Na tabela abaixo você pode comprar os dois padrões.

568A	**568B**
O primeiro fio é o *verde e branco*	O primeiro fio é o *laranja e branco*
Usa-se o par verde para transmissão	Usa-se o par laranja para transmissão
Usa-se o par laranja para recepção	Usa-se o par verde para recepção
É usado tanto com 568A quanto com 568B	É usado apenas com um 568A

Tabela 4.6. Diferença entre os padrões

4.8. O conector RJ-45

Por fim, os fios são colocados em um conector. O conector para máquinas de rede é o RJ-45. É um conector, obviamente, com 8 pinos: um para cada fio. São feitos de vidro.

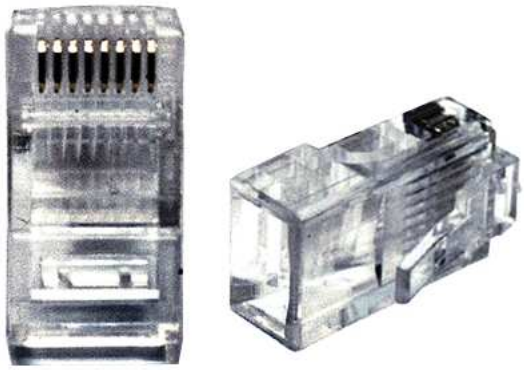

Figura 4.7. Conector RJ-45.

Na figura acima, olhando para o conector que está em pé, o pino que está mais a esquerda é onde será encaixado o fio 1. Ou seja, o verde e branco pelo padrão 568A, ou laranja e branco no 568B.

A ferramenta que usamos para fixar os fios no conector é o alicate de crimpagem. Diz-se "crimpagem" porque "crimpar" é a palavra que define fixar os fios no conector.

Figura 4.8. Alicate de crimpagem.

Na figura acima, o alicate possui três lâminas. Duas delas você pode ver, pois estão na parte de cima do alicate. A outra você não pode ver. Essas duas lâminas não se juntam, ficando sempre um espaço entre elas mesmo quando o alicate está fechado. Elas servem para desencapar o cabo. Você coloca o cabo ali, fecha o alicate e gira em torno do cabo, puxando-o em seguida. O alicate da figura contém uma catraca, que o impede de abrir enquanto você faz isso. Aconselho que você compre um alicate de crimpagem *com catraca*. O que resulta é os fios do cabo aparecendo, ou seja, desencapados.

A lâmina que está na parte de baixo do alicate é única, porém, quando o alicate é fechado, ela chega até o final. Serve para cortar o cabo, ou aparar os fios.

Este alicate tem duas entradas. A maior é para o conector RJ-45. A menor, para conectores RJ-11, usados em cabos telefônicos. Você deve inserir os fios no conector antes de colocá-lo no cabo, e os fios devem estar aparados para isso. Depois, crimpa-se o cabo, ou seja, você deve apertar o alicate para que os pinos de metal fixem-se nos fios. O alicate mostrado na figura permite que você faça isso sem usar muita força. Contudo, existem alicates baratos que são tão primitivos, por assim dizer, que é preciso imprimir uma força extraordinariamente prejudicial para a próstata (ou útero) do indivíduo.

4.9. Conclusão

Neste capítulo você estudou sobre a camada física das redes locais cabedas. Vimos que cabos transmitem bits, 0s e 1s, em forma de sinais elétricos. As placas de rede podem sinalizar 1 com 5v e 0 com 0v ou -5v. Placas de rede da máquina que envia convertem um quadro de camada enlace em bits e os transmite pela camada física (isto é, os cabos); a placa de rede da máquina de destino recebe os bits e os traduz devolta em um quadro de camada enlace.

O quadro de camada enlace não é encapsulado em nenhuma outra coisa pela placa de rede. A placa apenas traduz.

O tipo de cabo normalmente usado em redes locais Ethernet é o cabo de par trançado sem blindagem, ou UTP. Este cabo contém 8 fios, sendo que destes 8 fios, somente 4 são usados: dois para envio e dois para recepção.

Algumas máquinas de rede, como os computadores e os roteadores, enviam bits pelos fios 1 e 2 do cabo, e recebem pelos fios 3 e 6. Outras máquinas de rede, como os comutadores, fazem o contrário: recebem pelos fios 1 e 2 e enviam pelos fios 3 e 6.

Você deve estar atento a isso quando fizer ligações em equipamentos de rede. Na maioria das vezes você usará um cabo com fios diretos.

O nome do conector usado nesses cabos é RJ-45. Você pode crimpar um cabo com um alicate de crimpagem.

Embora qualquer ordem dos fios funcione, desde que estejam coerente, existem dois padrões para serem seguidos; o mundo inteiro os usa, e você também deveria usar. São os padrões EIA/TIA 568A (cujo fio 1 é verde e branco) e o 568B (cujo fio 1 é laranja e branco).

4.10. Exercícios

Exercício 4.1. O que acontece com a informação, quando ela passa da camada enlace para a camada física?

a) É encapsulada

b) É desencapsulada

c) É convertida

d) Nenhuma das alternativas

Exercício 4.2. Qual o padrão EIA/TIA usado para conexões com fios diretos, cujo fio 1 é o verde e branco?

Exercício 4.3. Quais são os fios menos relevantes?

a) 1, 2, 3, 6

b) 1, 3, 5, 7

c) 4, 5, 6, 7

d) 4, 5, 7, 8

Exercício 4.4. Quais são os fios usados para **recepção** em computadores e roteadores?

a) 1, 2

b) 3, 6

c) 1, 3

d) 2, 6

Exercício 4.5. Quais são os fios usados para **envio** em comutadores?

a) 1, 2

b) 3, 6

c) 1, 3

d) 2, 6

Exercício 4.6. Se estamos usando o padrão EIA/TIA 568B, quais as cores dos fios usados para **enviar** informações? (fios 1 e 2) (marque duas alternativas)

a) Verde e branco

b) Verde

c) Laranja e branco

d) Laranja

Exercício 4.7. Se estamos usando o padrão EIA/TIA 568A, quais são as cores dos fios usados para **receber** informações? (marque duas alternativas)

a) Verde e branco

b) Verde

c) Laranja e branco

d) Laranja

Exercício 4.8. Analise a figura abaixo, e escreva nas ligações se estas são diretas ou trocadas.

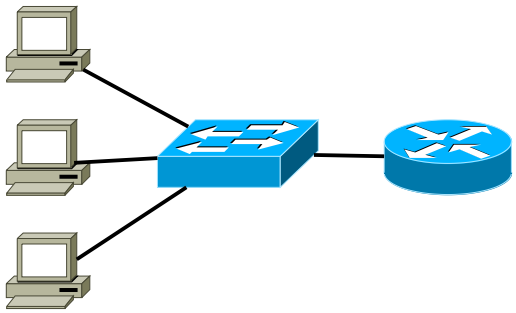

Figura 4.9. Qual o tipo de cabeamento usado?

Exercício 4.9. Continue com a figura abaixo.

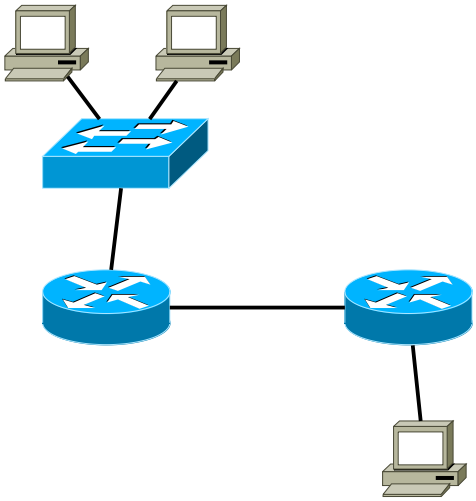

Figura 4.10. Qual o tipo de cabeamento usado?

Exercício 4.10. Continue com a figura abaixo. Assuma que um repetidor (isto é, um **hub**) funciona da mesma maneira que o comutador.

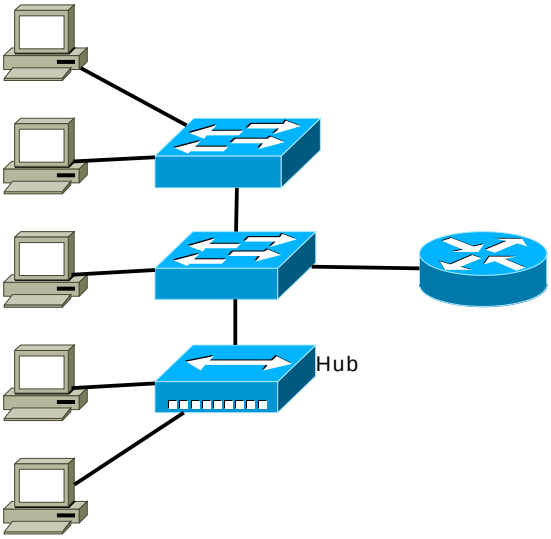

Figura 4.11. Qual o tipo de cabeamento usado?

Capítulo 5
DHCP e DNS

5.1. Introdução

Neste capítulo, você aprenderá sobre dois serviços da camada aplicação de redes: DHCP (Protocolo de Configuração Dinâmica de Máquina[5.1], *Dynamic Host Configuration Protocol*), e DNS (Sistema de Nomes de Domínio, *Domain Name System*); verá qual a função desses protocolos e o funcionamento básico dos mesmos.

Como já foi dito, ambos os protocolos estão na camada aplicação da arquitetura TCP/IP; isso significa que seus dados são encapsulados na camada transporte, que por sua vez, são encapsulados na camada rede, depois na camada enlace, e enfim transmitidos pelo enlace físico (camada física). Embora atuem na camada aplicação, ambos os protocolos alteram informações nas máquinas, que dizem respeito ao endereçamento lógico de camada rede. Ou seja, são protocolos de camada aplicação que configuram a camada rede das máquinas.

Você pode inicialmente achar estranho a camada aplicação alterar dados da camada rede do máquina, mas isso é absolutamente comum e corriqueiro. Veja um exemplo prático: no Linux, usamos o comando `ifconfig <nome_da_interface>` para exibir informações da camada rede da interface[5.2] indicada. O programa `ifconfig` obtém as informações da camada rede, gerenciadas pelo sistema operacional (no caso, o Linux) e mostra-as na tela. Ou seja, uma aplicação exibindo informações da camada rede da máquina local. Abaixo, mostramos um exemplo de retorno do comando:

Exemplo 5.1. Comando ifconfig, para obter informações da camada rede.

```
[nomedamaquina nomedousuario]# ifconfig ppp0
ppp0          Link encap:Protocolo Ponto-a-Ponto
              inet end.: 189.66.160.21 P-a-P:10.64.64.64 Masc:255.255.255.255
              UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Métrica:1
              RX packets:10 errors:0 dropped:0 overruns:0 frame:0
              TX packets:11 errors:0 dropped:0 overruns:0 carrier:0
              colisões:0 txqueuelen:3
              RX bytes:178 (178.0 b) TX bytes:211 (211.0 b)
```

A saída deste comando mostra o endereço lógico da máquina (`inet end.: 189.66.160.21`), mostra também o protocolo de camada enlace na qual os pacotes são encaulados (`Link encap:Protocolo Ponto-a-Ponto`), e o endereço da máquina na qual esta máquina está diretamente conectada para acessar a internet (`P-a-P:10.64.64.64`), bem como a máscara de rede (`Masc:255.255.255.255`) e outras informações úteis. São informações da camada rede da máquina.

5.1. Ou hospedeiro.

5.2. A interface nem sempre é uma placa de rede; pode ser um modem USB HSDPA como no exemplo mostrado, um modem de cabo, um modem sem fio e assim por diante.

Assim, é natural que a camada aplicação mostre, e até mesmo altere informações da camada rede. Quando a máquina recebe as informações de DHCP ou DNS, o que ela faz? Ela vai desencapsulando a informação e subindo com ela até chegar à camada aplicação; quando os dados chegam à camada aplicação, a aplicação própria obtém esses dados e, conforme seja, modifica a camada rede com essas informações.

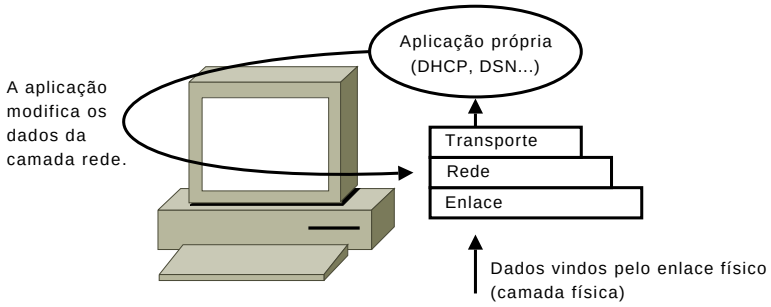

Figura 5.1. Como uma aplicação modifica dados da camada rede.

Agora que você compreende como isso é natural, podemos começar a explicação do funcionamento desses protocolos.

5.2. Objetivo do DHCP

DHCP é o Protocolo de Configuração Dinâmica de Máquina. Seu objetivo é configurar, sem intervenção do técnico ou administrador de rede, as configurações de camada rede da máquina. Isso signigica que se você tem 500 máquinas na rede local, você não precisará ir em uma a uma configurar os itens da camada rede; basta ligá-las, e elas os configurarão dinamicamente, se existir na rede local um servidor DHCP. Os itens que podem ser configurados dinamicamente via DHCP são:

- Endereço lógico (IP) e máscara de rede.
- Endereço do Gateway padrão.
- Opcionalmente, porém recomendado, o endereço do servidor (ou dos servidores, se for mais de um) DNS.

Por ser o DHCP um protocolo da arquitetura TCP/IP, ele é encapsulado em um protocolo IP, trabalhando, portanto, com endereços IPs.

5.3. Funcionamento do servidor DHCP

Qualquer máquina na rede local pode ser um servidor DHCP. Se existirem duas máquinas, as duas trabalharão nessa função. Bom, para facilitar as coisas, assuma que nossa rede local possui um servidor DHCP, conforme ilustrado na figura abaixo.

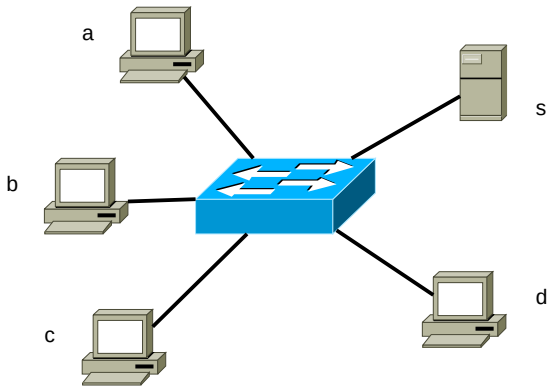

Figura 5.2. LAN com servidor DHCP.

Na figura, temos cinco máquina, sendo que uma delas (a máquina `s`) é o servidor de DHCP. A máquina `s`, portanto, é a única que terá as informações da camada rede configuradas estaticamente; ou seja, você, o cara da rede, vai lá configurar o endereço lógico, que no caso é um endereço IP, a máscara, o endereço do Gateway padrão que a máquina usa (discutiremos sobre Gateway padrão em breve neste curso) e o endereço do servidor DNS. Abaixo, a tabela mostra um exemplo do que será configurado na máquina `s`.

Endereço IP, Máscara	192.168.0.1, 255.255.255.0
Endereço do Gateway padrão	192.168.0.254
Endereço do DNS	10.10.10.10

Tabela 5.1. Exemplo de configuração no servidor DHCP.

Essas são configurações normais e corriqueiras de camada rede de uma máquina; todas as outras máquinas desta rede local também terão esses itens, com a diferença de que não será preciso você ir nelas para configurá-las: elas serão configuradas dinamicamente.

Ainda são necessárias duas outras configurações no servidor DHCP, configurações estas que apenas o servidor DHCP possui: primeiro, a ativação de um servidor DHCP, que é um programa próprio que fará a máquina executar as funções para que ela foi designada (sem um servidor DHCP, esse "servidor" é apenas uma máquina comum, como todas as outras); e segundo, um intervalo de IPs que serão disponibilizados para as outras máquinas da rede local.

Geralmente, para ativar o programa servidor de DHCP na máquina servidora, basta um único comando, como, por exemplo, `service dhcpd start` em máquinas Linux. Na verdade, máquinas configuradas para isso iniciam o serviço automaticamente toda vez que são ligadas. A segunda configuração, que é o intervalo de endereços lógicos, define quais IPs serão "doados" para as outras máquinas de rede. Por exemplo, você poderia definir um intervalo que iria de 192.168.0.2 (começamos deste número porque o servidor já está usando o endereço 192.168.0.1) até 192.168.0.253 (o IP 192.168.0.254 já está sendo usado pelo Gateway padrão, conforme configuração na tabela acima).

5.4. Funcionamento do cliente DHCP

Nos sistemas operacionais atuais, quando uma máquina não possui endereço lógico configurado, ela é um cliente DHCP que fará de tudo para conseguir preencher o que lhe falta: coração vazio, em busca do amor desconhecido por vales e padrarias. Eis o estado da camada rede da máquina cliente assim que é ligada:

Endereço IP, Máscara	Em branco, Em branco
Endereço de Gateway padrão	Em branco
Endereço do DNS	Em branco

Tabela 5.2. Estado inicial da camada rede da máquina.

Pobre máquina. Neste exemplo, usaremos a máquina `a`.

Como esta máquina está na rede local, lhe é permitido enviar um pacote de camada enlace com destino broadcast (lembra-se?), à procura do servidor DHCP. Neste caso, a camada aplicação desta máquina solitária e sem vontade de viver enviará dados de camada aplicação procurando pelo servidor DHCP. Este dado é algo assim: "ei, você é um servidor DHCP?". A pergunta é encapsulada em um segmento de camada transporte, que é encapsulado em um pacote de camada rede, sem endereço IP de origem definido (pois, logicamente, não há um endereço IP), e com endereço IP de destino como broadcast (pois a máquina nada sabe sobre o universo onde vive).

O pacote é encapsulado em um quadro da camada enlace, com endereço de origem da placa de rede de máquina `a` (pois o endereço físico vem de fábrica gravado na placa), e com um endereço físico de destino como broadcast. Esse quadro é enviado para o enlace físico, ou seja, para a chamada camada física, e navega pela rede, à procura das américas.

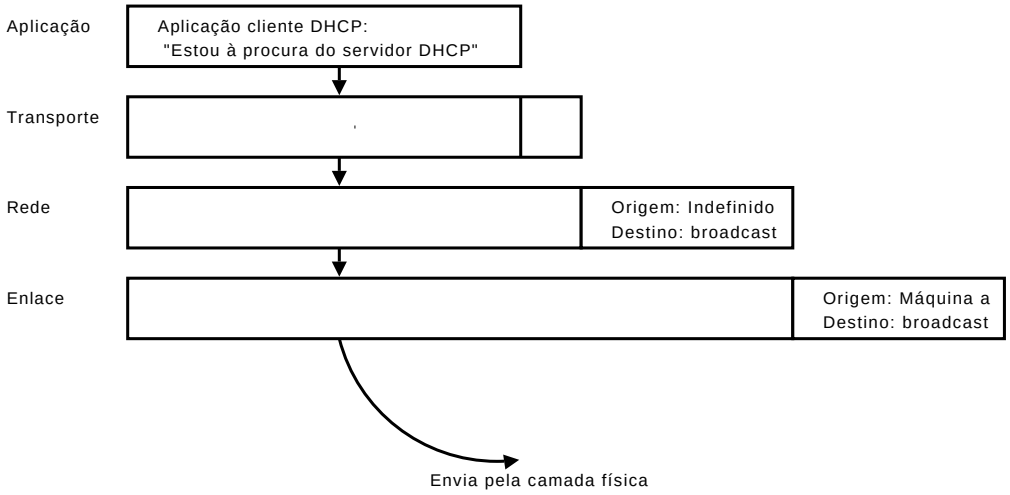

Figura 5.3. Máquina cliente `a` requisitando dados.

Como o quadro é de destino broadcast, ele vai parar em todas as placas de rede da rede local. E também, cada uma das placas de rede vai processar o quadro, pois todas têm esperança de que ao desembrulhar o quadro, receberão uma caixa de bombons ou outro agradável presente. Ao abrir o quadro, surge o pacote, com endereço lógico de destino broadcast. Todas as máquinas ficam empolgadas, ansiosas, algums gritam e outras vibram, achando que a mensagem é mesmo para elas.

Ao desembrulhar o pacote, percebem o segmento de camada transporte, e neste segmento há um campo, e neste campo há um número, por exemplo 68. Este número indica o tipo de aplicação que receberá os dados. Todas as máquinas chegam nesse ponto, pois o pacote/frame é basicamente o mesmo em todas elas (destino: broadcast). Contudo, apenas a máquina servidora DHCP possui uma aplicação escutando na porta 68. Assim, a única máquina a desencapsular o segmento de transporte e passar os dados para a camada aplicação, é a servidora DHCP. A partir daí, você já deve descobrir o que acontece.

O programa servidor responde dizendo algo do tipo “ei, eu sou o servidor DHCP! Você quer um endereço de IP? Tudo bem! Eu tenho aqui... er... deixe-me ver... o número 5 já está sendo usado... o número 4 está com aquele maluco da esquina... ah, sim! Eu tenho aqui disponível o IP 192.168.0.30!”.

Essa informação vai descendo a pilha de protocolos, pela camada transporte, depois pela camada de rede (que escreve no pacote seu próprio IP, isto é, o IP de `s`, como origem e o destino broadcast), a seguir pela camada enlace (que escreve no quadro seu próprio endereço físico como origem, ou seja, endereço físico de `s`, e o endereço físico da placa de rede da máquina destinatária, que é `a`, pois a máquina sabe quem é `a`), e envia o quadro pela camada física.

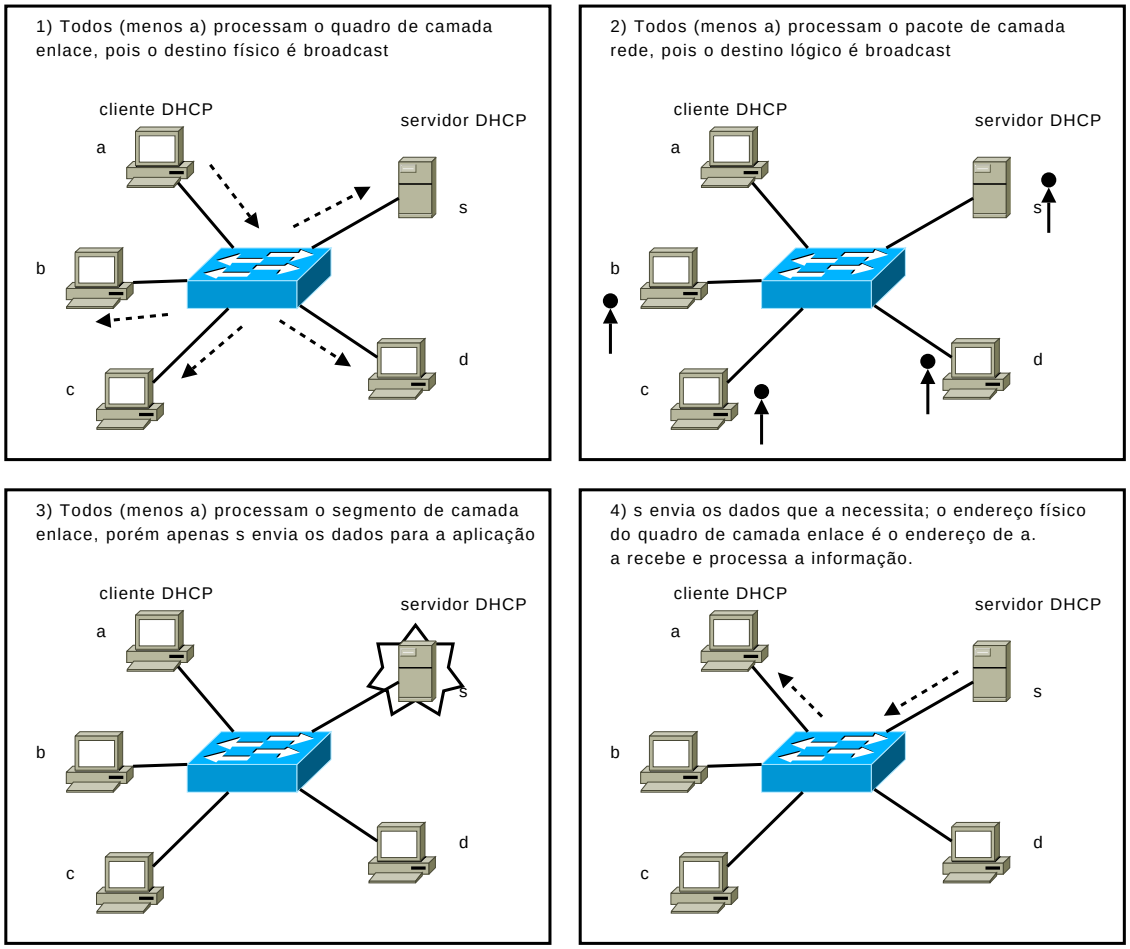

Figura 5.4. Resquisição e resposta DHCP.

A máquina `a` recebe o quadro, desencapsula, e vê o pacote. Neste pacote, a origem é o endereço de IP da máquina `s`. O destino é broadcast, assim, ela desembrulha o pacote. Ela vê o segmento de camada transporte, com o número de porta da apicação cliente DHCP (a mesma aplicação que, no começo de nossa aventura, buscou por informações de camada rede). Como há uma aplicação rodando nesta porta, a camada transporte envia os dados para esta aplicação. Nós dissemos que o servidor DHCP `s` respondeu com um endereço IP, mas na verdade, ele responde isso e mais alguma coisa, a saber:

- Máscara de rede.
- Gateway padrão.

- DNS (se estiver configurado no servidor).
- Tempo de vida. Isso informa quando tempo aquele endereço IP será alocado à máquina `a`. O padrão é 24h, mas isso pode ser configurado no servidor.

Então a aplicação, ao ver estes dados, configura a camada rede da máquina com essa informação.

Endereço IP, Máscara	192.168.0.30, 255.255.255.0
Endereço de Gateway padrão	192.168.0.254
Endereço do DNS	10.10.10.10

Tabela 5.3. Estado final da camada rede da máquina.

Agora a máquina pode conversar com outras máquinas da rede, lembrando que todas as outras máquinas clientes DHCP passam pelo mesmo processo, afim de obter informações de camada rede para sentirem-se realizadas na vida. O servidor não atribui o mesmo endereço IP a duas máquinas diferentes, pois ele sabe a quem delegou os IPs. Em linguagem mais direta, ele sabe quais endereços físicos possuem os IPs.

Nota 5.2. Agora faz sentindo o motivo de se configurar estaticamente os dados de camada rede no servidor DHCP; se ele for o único servidor na rede local, não haverá de quem buscar essas informações.

Agora, podemos passar para o DNS.

5.5. Objetivo do DNS

DNS é o Sistema de Nomes de Domínios. Seu objetivo é traduzir nome de máquinas em endereços lógicos. Como o DNS é um protocolo da arquitetura TCP/IP, ele traduz nomes de domínios em endereços IPs, e endereços IPs em nomes de domínio. Um domínio pode ser uma máquina ou uma rede. Para facilitar as coisas, todos os exemplos dados serão com máquinas inidividuais.

Assim como acontece com o DHCP, um cliente consulta um servidor para obter informações de camada aplicação. No caso do DNS, o servidor roda em uma máquina servidora que pode estar na rede local ou não (ou seja, pode estar fora da rede). Todas as outras máquinas que não sejam servidoras DNS são clientes; elas possuem uma tabela curta na camada rede, que é manipulada pela aplicação cliente. Essa tabela lista alguns poucos nomes de máquinas e seus respectivos endereços IP; caso a máquina cliente não encontre o nome da máquina nessa tabela local, ela requisita isso do servidor DNS, que possui uma tabela muito maior.

Graças ao servidor DNS você pode nevegar pela internet. Você não precisa saber de todos os endereços IPs dos servidores que você navega; basta saber um nome, como *www.google.com*, bem mais fácil de decorar do que um IP.

5.6. Tabela DNS local

Seria muito difícil você decorar o endereço lógico dos servidores que visita. A mente humana não se dá bem com números, tanto, que se você for normal, não chama seu melhor amigo pelo CPF dele. Da mesma forma, é melhor digitar o nome de um servidor do que seu número IP. O DNS faz a tradução para você: ele basicamente traduz um nome em um endereço IP.

Observe a tabela abaixo. Ela é um exemplo de uma tabela DNS local, em uma máquina qualquer:

Nome da máquina	Endereço IP
braço	192.168.0.31
sangue_bão	192.168.0.50
mano	200.10.20.21
nóix_é_déix	195.5.120.14

Tabela 5.4. Exemplo de tabela DNS local.

No exemplo acima, temos duas máquinas na rede local (braço e sangue_bão), uma máquina fora da rede, em São Paulo, por exemplo, e outra também fora da rede, no Rio de Janeiro. Sabemos que um pacote de camada rede não pode colocar "nóix_é_déix" no campo destino do pacote (pois esse campo só aceita endereçamento IP), o cliente DNS da máquina local vai consultar essa sua pequena tabela DNS para ver se encontra o endereço IP correspondente. E o cliente DNS encontrou: é 195.5.120.14. Este endereço é escrito no campo destino do pacote, e o pacote desce pela pilha e é enfim enviado. Simples assim.

Essa tabela pode ser um arquivo, lido com qualquer editor de textos simples.

5.7. Obtendo IP de máquina a partir do servidor

A tabela de DNS local não é infinita. Ela não contém, e nem poderia conter, todos os possíveis nomes mapeados para seus respectivos IPs. Por exemplo, imagine que esta máquina queira enviar um pacote para "trem_bão" (Minas Gerais). O cliente DNS procuraria esse nome na tabela, e não encontraria. A vontade de comer queijo minas é maior, então, o cliente DNS na máquina local passaria para o próximo passo, que é consultar um servidor DNS. Você está se lembrando, na explicação sobre DHCP, de que o servidor DHCP pode prover o endereço do servidor DNS para as máquinas na rede local?

Nota 5.3. Todas as máquinas da rede, se quiserem falar com outras máquinas pelo nome, precisam saber o endereço do servidor DNS. Esse endereço pode ser configurado estaticamente na máquina, ou pode ser aprendido via DHCP.

Pois bem. Essa máquina sabe qual é o endereço do servidor DNS, e envia uma mensagem de camada aplicação para lá. A mensagem diz mais ou menos: ei, você sabe qual o IP da máquina cujo nome é "trem_bão"?

Essa mensagem desce pela pilha de protocolos da máquina, através da camada transporte, em seguida, camada rede (com IP de destino sendo o endereço da máquina DNS), camada enlace e assim por diante. Ao chegar na máquina servidora DNS, a camada transporte desta encaminha os dados da aplicação para a aplicação correta (no caso, uma aplicação servidora DNS), que procura por "trem_bão" na sua enorme tabela DNS.

Se encontrar, ela retorna uma mensagem parecida com: achei! O endereço IP de "trem_bão" é 15.30.45.90. Essa informação desce pela pilha de protocolos e é destinada à máquina que pediu. Por sua vez, a máquina que pediu sobe com os dados pela pilha de protocolos e os envia para a aplicação cliente DNS, que escreve no arquivo DNS local o nome da máquina e o endereço aprendido. Agora, a máquina pode falar com "trem_bão", pois já sabe para qual IP enviar.

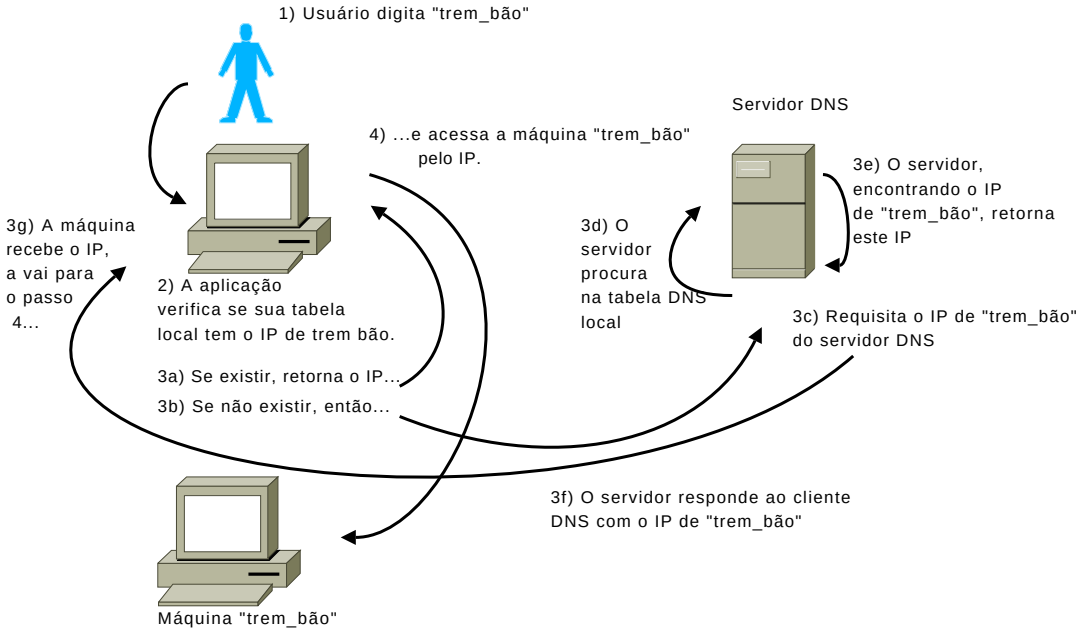

Figura 5.5. Funcionamento do DNS.

Uma observação: quando a máquina é desligada, ou quando passa-se algum tempo, o endereço aprendido que fica armazenado na tabela DNS local é perdido, e a máquina deve fazer uma nova requisição DNS ao servidor.

Sem um servidor DNS, é possível acessar a internet, mas você precisa decorar os IPs dos servidores, e isso não é lá muito agradável. Por isso dizemos que a configuração do IP do servidor DNS não é obrigatória, mas altamente recomendada.

5.8. Conclusão

Neste capítulo, você aprendeu sobre DHCP e DNS. O DHCP configura dinamicamente as informações de camada rede das máquinas na rede local. É necessário que o servidor DHCP esteja na rede local; de outro modo, as máquinas nunca poderiam obter as informações que precisam. Também é necessário que o servidor DHCP tenha suas informações de camada rede previamente configuradas pelo técnico ou administrador da rede, pois se não for assim, é impossível que ele próprio consiga os dados que precisa a partir do nada.

O DNS é o sistema de nomes de domínio. Sua função é basicamente mapear nomes de máquinas em endereços físicos (IP). Toda máquina possui uma pequena tabela DNS local para consultas rápidas; o nome das últimas máquinas aprendidas ficam nessa tabela, e duram até que a máquina seja desligada ou, então, o tempo de armazenamento seja estourado. Se o cliente DNS rodando na máquina não conseguir mapear um nome em endereço lógico a partir desta pequena tabela local, ele então obtém essa informação do servidor DNS.

É altamente recomendado que as máquinas da rede local saibam o endereço do servidor DNS. Esse endereço pode ser configurado tanto estaticamente quando via DHCP. Lembre-se que no servidor DHCP é possível configurar as informações da camada rede do próprio servidor, bem como as informações que serão disponibilizadas aos clientes DHCP; entre essas útlimas, encontram-se um intervalo de IPs, uma máscara, que é igual à máscara do servidor; um endereço de Gateway padrão, que pode ser ou não o mesmo do servidor, e, por fim, o DNS, que pode ser o não o mesmo do servidor, mas na maioria das vezes o é.

Podemos configurar mais de um endereço DNS, sendo um o primário, e, caso este falhe, o secundário ou terciário.

A seguir neste curso, estudaremos o que é o Gateway padrão de uma rede local, e porque é importante configurá-lo no servidor DHCP.

5.9. Exercícios

Exercício 5.1. A respeito do DHCP, marque a(s) alternativa(s) correta(s):

a) A função do DHCP é basicamente traduzir nomes de máquinas em endereços lógicos.

b) A função do DHCP é basicamente traduzir nomes de máquinas em endereços físicos.

c) A função do DHCP é basicamente configurar dinamicamente informações de camada rede nas máquinas.

d) A função do DHCP é basicamente configurar estaticamente informações de camada rede nas máquinas.

e) A função do DHCP é basicamente configurar dinamicamente, ou seja, através de um servidor DHCP na rede local, dados da camada enlace das máquinas clientes DHCP.

Exercício 5.2. A respeito do DNS, marque a(s) alternativa(s) correta(s):

a) A função do DNS é basicamente traduzir nomes de máquinas em endereços lógicos.

b) A função do DNS é basicamente traduzir nomes de máquinas em endereços físicos.

c) A função do DNS é basicamente configurar dinamicamente informações de camada rede nas máquinas.

d) A função do DNS é basicamente configurar estaticamente informações de camada rede nas máquinas.

e) A função do DNS é basicamente configurar dinamicamente, ou seja, através de um servidor DNS na rede local, dados da camada enlace das máquinas clientes DNS.

Exercício 5.3. O que costuma-se configurar no servidor DHCP? (marque todas as corretas)

a) IP do próprio servidor.

b) Máscara do próprio servidor.

c) DNS do próprio servidor.

d) Gateway padrão do próprio servidor.

e) Intervalo de IPs das máquinas na rede local.

f) DNS que serão usados nas máquinas.

g) Gateway padrão que será usado nas máquinas.

h) O tempo que o servidor alocará um IP para si mesmo.

Exercício 5.4. Quais informações de camada rede das máquinas clientes DHCP costumar ser as mesmas do servidor DHCP? (marque todas as corretas)

a) O endereço IP

b) A máscara de rede

c) O Gateway padrão

d) O DNS

Exercício 5.5. Marque o(s) aspecto(s) de diferença(s) entre o servidor DHCP e o servidor DNS:

a) Um servidor DNS deve estar, orbigatoriamente, na rede local. Um servidor DHCP não.

b) Um servidor DHCP deve estar, orbigatoriamente, na rede local. Um servidor DNS não.

c) Para máquinas sem IP passarem a se comunicar, é necessário um servidor DNS. Servidor DHCP é opcional para que elas se comuniquem.

d) Para máquinas sem IP passarem a se comunicar, é necessário um servidor DHCP. Servidor DNS é opcional para que elas se comuniquem.

Capítulo 6

Gateway padrão e portas do roteador

6.1. Introdução

Neste capítulo, você aprenderá o que é necessário configurar nas máquinas da rede local para que estas possam comunicar-se entre si e com o mundo externo (isto é, fora do domínio da rede local). Veremos também o que é um Gateway padrão, e para finalizar, diferenciaremos portas dos roteadores, em LAN e WAN.

6.2. Configuração das máquinas na LAN

Observe a figura abaixo.

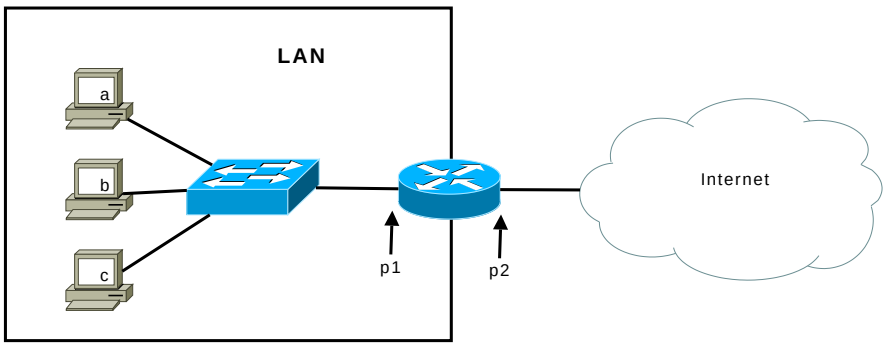

Figura 6.1. Máquinas da LAN e portas do roteador.

Nesta figura, você vê uma rede local com três computadores, um switch e um roteador. Nomeamos as portas dos computadores como `a`, `b` e `c`, e a porta do roteador como `p1`. Vamos fazer um breve resumo do que você aprendeu neste curso: a rede local possui quatro portas que se enxergam. Podemos perceber que o roteador possui outra porta, `p2`, que não é visível pelas portas das redes locais. Esta porta está fora do escopo da LAN. Porém, ela é visível a partir da internet.

Você já sabe que para `a` falar com `p1`, tudo que ele precisa é saber o endereço lógico de `p1` e enviar diretamente para ela. Assim também acontece se `c` quiser falar com `b`, e assim por diante. No centro da LAN, temos um comutador, que, verificando o endereço físico de destino do quadro que chega, irá encaminhá-lo pela porta correta.

Para que as máquinas de LAN se comuniquem entre si, é preciso que cada porta possua:

- Um endereço IP único da rede local
- Uma máscara de rede

Estas configurações devem ser aplicadas inclusive a porta `p1`, pois esta porta do roteador também pertence à LAN e deve comunicar-se com as máquinas da mesma. Para configurar as portas dos computadores, você pode usar o DHCP para não precisar trabalahar como peão configurando todas as 500 máquinas da rede local. Nesta rede você deve ter um servidor DHCP, que pode ser qualquer uma das máquinas, ou, como veremos adiante, o roteador.

A porta `p1` do roteador terá um IP estático, imutável, por causa do conceito de Gateway padrão.

6.3. GATEWAY PADRÃO

DEFINIÇÃO 6.1. *Gateway padrão. Gatewau padrão é um termo que define a máquina que será usada para encaminhar pacotes cujo endereço não consta na rede local.*

Pela definição acima, você deve entender que todas as máquinas da rede local devem saber quem é o gateway padrão da LAN. Cada LAN com uma conexão externa precisa ter um Gateway padrão. Observe a figura abaixo.

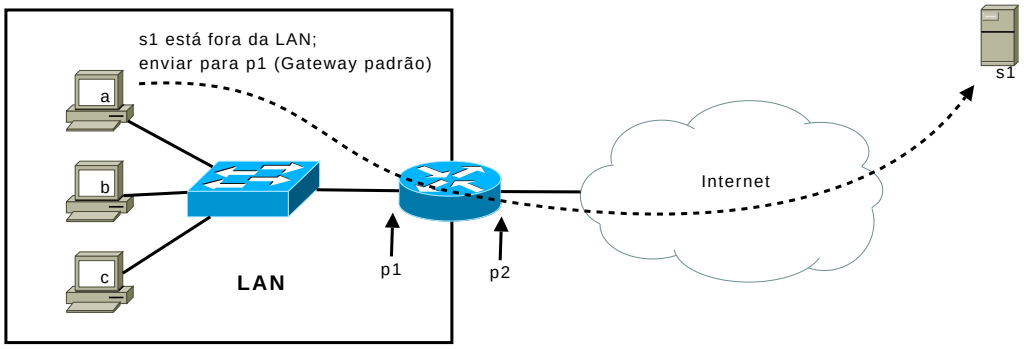

Figura 6.2. Máquina da LAN usando Gateway padrão.

Neste exemplo, se `a` quer enviar um pacote para `s1`, que está fora da rede local, vai usar o endereço lógico da porta `p1` do roteador. Assim, `p1` é o Gateway padrão. Todas as máquinas precisam saber isso. Na tabela abaixo, você pode ver a lógica que a máquina `a` usa para enviar pacotes para máquinas que estejam fora da rede local.

Destinatário	O que fazer?
b	Enviar diretamente para b (endereço físico de b)
c	Enviar diretamente para c (endereço físico de c)
p1	Enviar diretamente para p1 (endereço físico de p1)
s1	Enviar diretamente para p1 (endereço físico de p1)

Tabela 6.1. Lógica da máquina `a` ao enviar pacotes.

Ou seja, caso o endereço lógico de destino esteja na rede local, a máquina deve enviar diretamente para o endereço físico do destinatário. Caso o endereço lógico de destino esteja fora da rede local, a máquina deve enviar para o endereço físico de seu Gateway padrão, porém o endereço lógico do pacote continuará sendo o endereço da máquina destinatária.

Quando o Gateway padrão receber o quadro de camada enlace, ele o processará, pois sabe que é para ele. A camada rede do Gateway padrão analisará o pacote (lembre-se que nosso Gateway é um roteador, neste exemplo), e verá que o endereço lógico encontra fora da rede local. O roteador fará, então, o que tem de fazer: encapsular em um novo quadro de camada enlace e encaminhá-lo pela porta correta.

Você está percebendo o quão importante é todas as máquinas da LAN saberem quem é o Gateway padrão? Sem ele, não é possível se comunicar com outras redes. O roteador pode até estar corretamente configurado, mas se as máquinas locais não souberem que ele é o Gateway padrão, só comunicação local entre as máquinas da rede será possível.

Nota 6.2. Toda rede local tem um Gateway padrão.

Vamos complicar um pouco as coisas. Observe a figura abaixo.

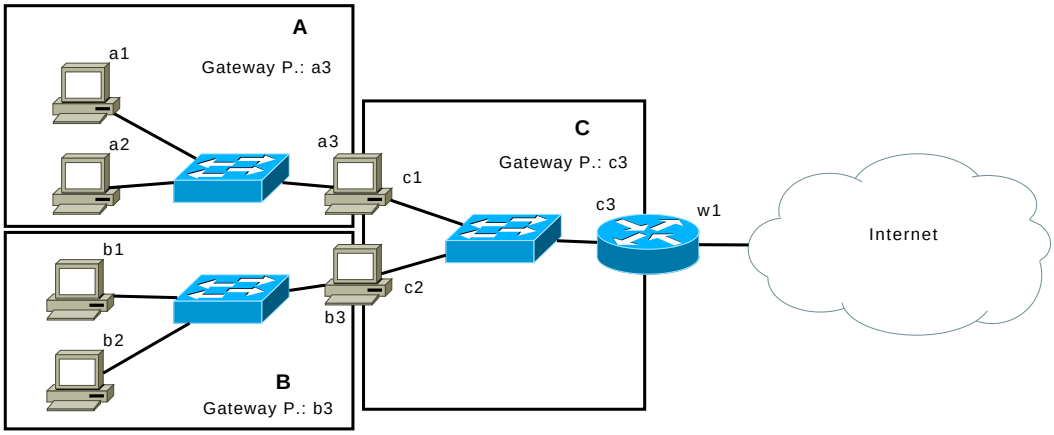

Figura 6.3. Demonstração de como o Gateway padrão é relativo à rede local.

Na figura acima, você vê três redes locais. A rede A, cujo Gateway padrão é a porta a3, a rede B, cujo Gatewey padrão é a porta b3, e... hum? Como assim? Rede C? Que palhaçada é essa?

É vero, meu caro... até mesmo seu Gateway padrão pode fazer parte de uma LAN. Neste exemplo você percebe que as máquinas da rede C são apenas máquinas da rede local. A rede C possui três portas, `c1`, `c2` e `c3`, sendo que o Gateway padrão desta rede é `c3`. Se a porta `c1` quer comunicar-se com `c2`, ela envia diretamente para ela. Caso `c1` queira comunicar-se com uma máquina na internet, então enviará para? Para quem? Para `c3`, seu Gateway padrão.

Cada rede local tem um Gateway padrão que as máquinas usam para comunicar-se com a rede externa. Na figura, temos três redes e, portanto, três Gateways padrão. Perceba que um Gateway padrão não precisa ser um roteador... pode ser um computador com duas placas de rede. Portanto, uma segunda definição seria...

Definição 6.3. *Gateway padrão (2). Uma máquina específica na rede local, seja ela um roteador ou um computador, que será usada para enviar pacotes com endereços lógicos fora da rede local E QUE TAMBÉM estejam mais para fora do campus.*

É uma definição meio estranha, ainda mais pela frase "E QUE TAMBÉM estejam mais para fora do campus". O que significa esse "mais para fora"?

Antes de continuarmos, observe uma coisa importante:

NOTA 6.4. Um roteador encaminha pacotes de camada rede pelas portas adequadas, de acordo com o endereço lógico de destino. Já um computador comum só sabe fazer duas coisas: enviar o quadro de camada enlace para a máquina local, ou enviar o quadro de camada enlace para o Gateway padrão.

Um roteador é mais poderoso do que um computador comum[6.1], primeiro pela quantidade de portas, e segundo, pela capacidade de roteamento. Um roteador toma muitas decisões, analisa o endereço de IP destino e decide por qual das portas vai encaminhar o pacote. Já um computador comum possui apenas uma lógica simples, que diz: pacotes de camada rede destinados à esta rede (local) são enviados diretamente para a máquina. Caso o endereço não esteja na rede local, envie para o Gateway padrão.

Um roteador pode ter, por exemplo, cinco decisões diferentes para o endereço lógico: endereços começados com 10 devem ser enviados para a máquina X; endereços começados com 20 devem ser enviados para a máquina Y; endereços começados com 30 devem ser bloqueados, e endereços que estejam na rede local, devem ser enviados pela porta p3 do roteador. Um roteador é complexo.

Já um computador comum, possui apenas duas lógicas. Se o endereço lógico for a rede loca, envie diretamente; caso contrário, envie para o Gateway padrão. Pronto. Um computador comum possui uma lógica bem mais simples que um roteador.

Bom, agora que você está ciente disso, o que aconteceria se, em nosso exemplo, a máquina cuja porta é `a1` quisesse falar com a máquina cuja porta é `c2`. Seria possível?

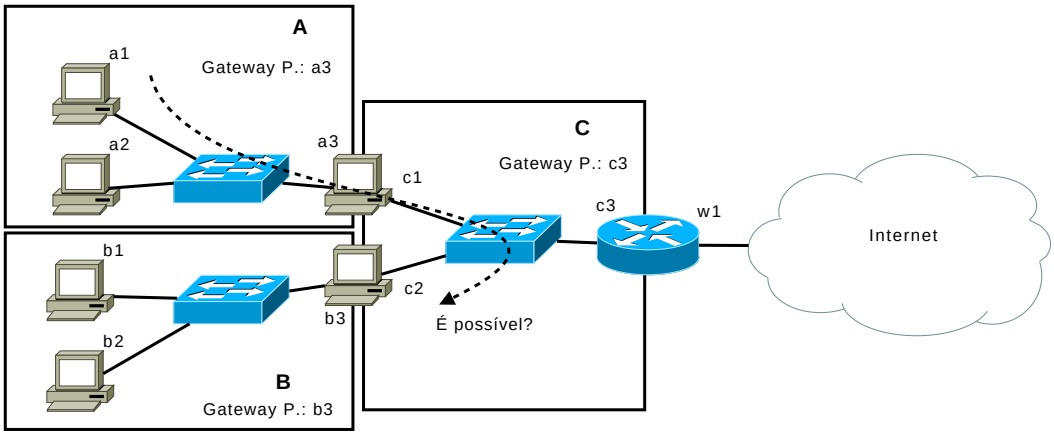

Figura 6.4. Máquina `a1` falando com `c2`.

A resposta é: sim, seria possível:

1. A máquina `a1` quer falar com `c2`. Então, ela monta um pacote de camada rede cujo endereço de destino é `c2`. Entretanto, a máquina `a1` sabe que o endereço físico não encontra-se na rede local; assim, ela constrói um quadro de camada enlace cujo endereço de destino é seu Gateway padrão, ou seja, `a3`, e envia o pacote para a rede local.

6.1. Dizemos "computador comum" para enfatizar que computadores também podem ser roteadores, desde que o sistema operacional do mesmo esteja devidamente configurado.

2. O Gateway padrão `a3` recebe o quadro cujo endereço destino é ele próprio, e processa-o. Ao encontrar o pacote de camada rede, vê o endereço de destino. Esta máquina, que é um computador comum, e não um roteador, possui duas portas, sendo quem uma delas está na rede `C`. Como o endereço lógico de destino está na rede local C, a máquina envia o quadro diretamente pela porta `c1` (a porta que está na rede `C`) com destino à porta `c2`.

Vejamos agora a cenário da figura abaixo:

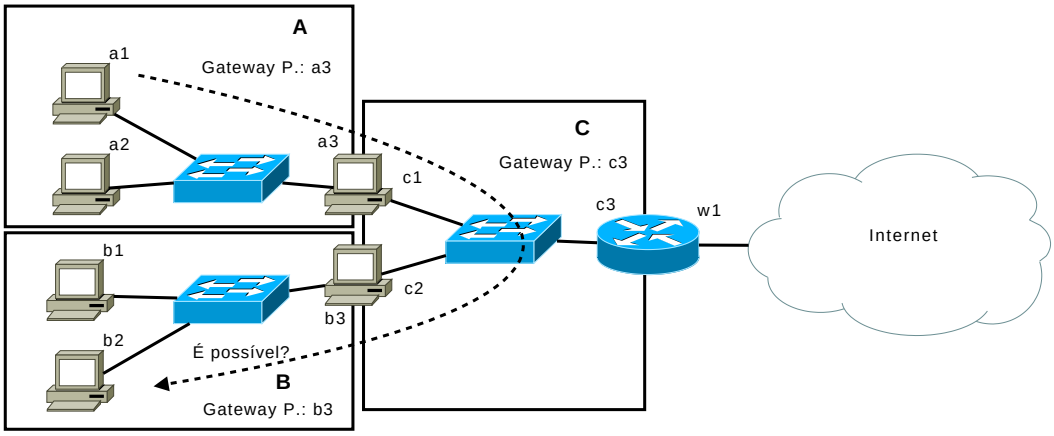

Figura 6.5. `a3` falando com `b2`: é possível?

E então? A resposta pode tanto ser **sim**, como **não**.

1. A máquina `a1` cria um pacote com endereço lógico de destino igual a `b2`. Este pacote é encapsulado em um quadro de camada enlace, cujo endereço de destino é `a3`. Por quê? Porque a máquina sabe que `b2` não está na rede local, e nesses casos, envia-se o quadro para o Gateway padrão.

2. `a3` recebe o quadro, vê o pacote e percebe que o destino não está na rede local. O que esta máquina faz, então? Lembre-se de que esta máquina não é um roteador, e sim, um computador comum. Bom, ela envia um quadro para o Gateway padrão, através da porta `c1`. E quem é o Gateway padrão de `c1`? É `c2`? Não! É, na verdade, a porta `c3` do roteador. Por aí você percebe que há um erro de encaminhamento: o quadro não é encaminhando para `c2`.

3. O roteador (que é o Gateway padrão da rede `C`) recebe o quadro destinado a ele. Ele vê o pacote de camada rede. Ele vê o endereço destino do pacote. O destino está na rede `B`. Não sabemos (porque não nos foi informado) se o roteador tem rota para a rede `B`. Se ele estivesse comunicando-se com outros roteadores, em vez de com computadores comuns, poderíamos supor que ele soubesse para qual roteador enviar o pacote, visto que roteadores podem comunivar-se por meio de protocolos próprios. Contudo, não é este o caso. Se o roteador não tem rota para a rede `B`, então ele simplesmente descarta o pacote.

4. Contudo, se o roteador sabe que a rota para a rede `B` é `c2`, ele enviaria um quadro para esta porta. Como esta máquina tem ua porta na rede `B`, e o endereço `b2` é local nesta rede, a máquina encaminharia um quadro para o tal endereço.

Assim, você pode perceber que um Gateway padrão nem sempre é um roteador; e que computadores comuns possuem uma lógica mais simples do que roteadores.

6.4. PORTAS DO ROTEADOR

Um roteador pode ter uma, duas ou várias portas. Para facilitar nosso estudo, usaremos inicialmente um exemplo com duas portas, sendo que uma delas está ligada à rede local, e a outra, à internet (ou seja, é uma porta WAN).

NOTA 6.5. Nos roteadores descritos neste capítulo, as portas não vêm de fábrica designadas como LAN ou WAN; você, o operador do roteador, é quem as configura para assim funcionarem. Portanto, em um roteador com quatro portas, você definirá quais terão comportamento de porta LAN e quais, de porta WAN.

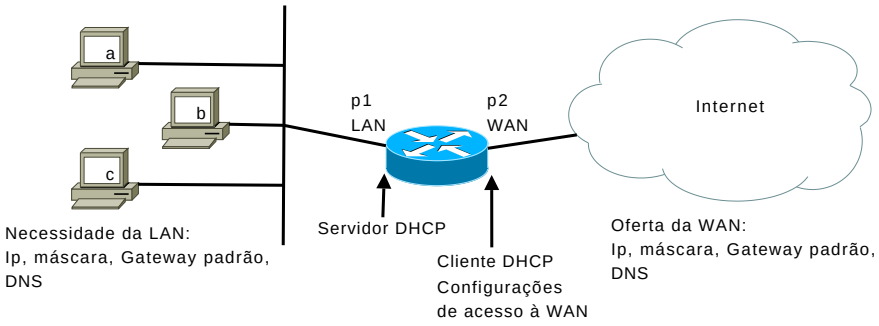

Figura 6.6. Representação abreviada das portas do roteador.

Na figura acima, você vê que o roteador tem duas portas, `p1` e `p2`. O roteador vem de fábrica com as portas desconfigurados, ou seja, sem IP, máscara ou outra coisa. Você deve escolher qual das portas será a porta LAN e qual será a WAN. Qualquer uma delas serviria, pois ambas possuem a mesma capacidade de configuração, Na verdade, as duas poderiam ser portas LAN, conectando duas LANs diferentes, ou, ainda, as duas poderiam ser WANs. Mas nesses exemplo, uma delas é LAN (no caso, a `p1`) e a outra, WAN.

Quais as configurações típicas para uma porta LAN?

- Endereço IP configurado estaticamente. Ou seja, você, o operador do roteador, definirá um endereço IP para a porta LAN. Uma máscara de rede também é configurada estaticamente.

- A porta LAN será o Gateway padrão da rede local; isso faz sentido, certo? Os computadores encaminharão pacotes para fora da rede através do roteador. Você não precisa fazer nenhuma configuração na porta para que esta seja o Gateway padrão; as máquinas da rede é que precisam saber disso.

- A porta LAN poderá estar configurada para ser um servidor DHCP. Você poderia ter qualquer outra máquina na rede local com essa função, mas os roteadores já têm essa capacidade.

- Na figura, você pode observar qual a necessidade de cada máquina na LAN: elas necessitam de um IP, de uma máscara, de um Gateway padrão e, opcionalmente, de um endereço para DNS. Tudo isso pode ser fornecido pela porta `p1` do roteador, pois esta tem um servidor DHCP rodando. Em suma, você deve configurar na porta `p1` do roteador todos os dados que as máquinas necessitam.

Assim, graças à porta LAN do roteador, que está devidamente configurada, qualquer máquina que esteja ligada é capaz de se comunicar com outra máquina na LAN, e com máquinas fora da LAN, pois elas sabem o endereço do Gateway padrão (que é `p1`).

Já com a porta WAN, é diferente. Esta porta não está provendo serviços à rede local. Ao contrário, ela recebe informações do provedor de internet; na vida real, conforme mostra a figura, é o provedor quem define o IP da porta `p2`, a máscara, o endereço do DNS etc.

Nota 6.6. Como o roteador recebe o endereço do servidor DNS por `p2`, este mesmo endereço pode ser usado no servidor DHCP rodando na porta `p1`, para que todas as máquinas da LAN saibam, também, o endereço do DNS.

A porta `p2`, portanto, na maioria das vezes, será um cliente DHCP. Outras configurações de portas WANs veremos posteriormente neste curso.

Observe, agora, um caso de um roteador com quatro portas:

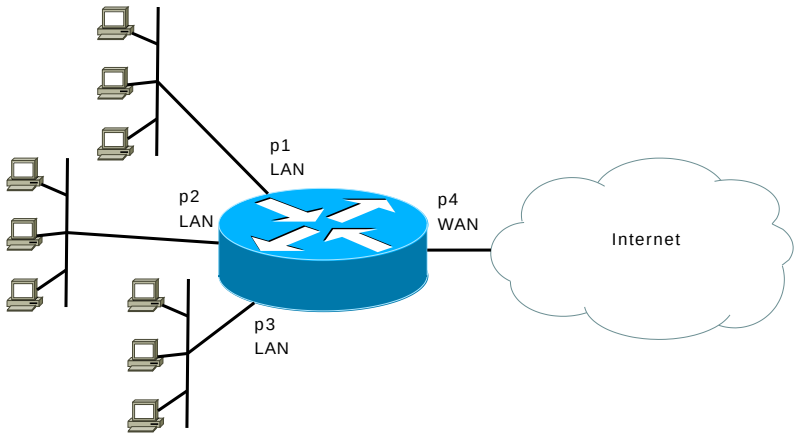

Figura 6.7. Roteador com 4 portas, com uma porta configurada para WAN.

Na figura acima, vemos que uma das portas foi configurada como porta WAN: a porta `p4`. Poderia ser qualquer outra porta: o roteador não faz distinção entre elas. As portas `p1`, `p2` e `p3` estão configuradas para, *cada uma delas independentemente*, serem portas de LAN. Assim, temos três portas ligadas à três LANs distintas.

- Cada uma das portas LAN tem um endereço IP e uma máscara de rede diferentes, já que as portas estão em redes diferentes.
- Cada uma das portas LAN é o Gateway padrão de suas respectivas redes. Isso significa, por exemplo, que quando uma máquina da rede ligada à `p1` enviar um pacote para fora da rede local, o pacote será enviado para `p1`.
- Cada uma das portas LAN roda um servidor DHCP independente; esses servidores proverão os dados de IP, máscara, Gateway padrão e DNS para suas respectivas redes locias.
- Não se esqueça de temos um roteador na figura. Assim, se uma máquina na rede ligada à `p1` quiser falar com uma máquina na rede ligada à `p3`, ela enviará um quadro com endereço físico de destino para o gateway parão, que é `p1`, o roteador encaminhará o pacote para a porta `p3`, naturalmente. A porta `p3`, por sua vez, encapsulará o pacote em um quadro de camada enlace e o encaminhará diretamente para a máquina na respecitiva rede local.

- Caso alguma máquina de uma das redes locais queira falar com alguém que esteja na internet, também não há problemas: a máquina enviará um quadro para seu Gateway padrão (que é uma porta do roteador), e este encaminhará o pacote para a porta WAN.

Em suma: na figura, temos quatro redes locais interligadas entre si e à internet através do roteador. Lembre-se que roteadores não encaminham quadros de camada enlace cujo endereço físico destino seja broadcast ou desconhecido. As quatro redes locais são independentes, isoladas entre si: duas máquinas só podem comunicar-se através de roteamento, como o mostrado acima.

6.5. Observações sobre roteadores domésticos

Joãozinho foi à feira e encontrou um roteador de cinco portas. Olhando na embalagem do produto, ele vê que o roteador tem quatro portas LAN e uma porta WAN e pensa: "ual! um roteador de cinco portas! Vou poder interligar quatro redes entre si e à internet!". Ele também vê que o preço do roteador é muito acessível, e compra. Todavia, embora não saiba, ele não comprou um roteador "tradicional", digamos assim... ele comprou um roteador doméstico: e nesses tipos de roteadores, não importa quantas portas LAN você tenha: você terá apenas *uma* rede local.

Acompanhe a explicação para saber o porquê. Abaixo, uma figura representativa do chassi de um roteador doméstico.

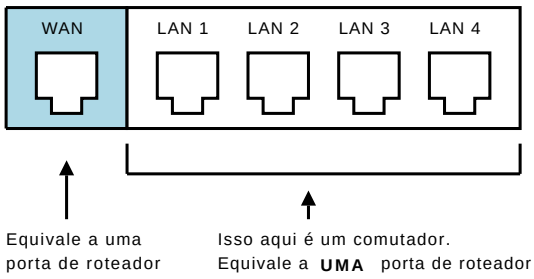

Figura 6.8. Represetação de um roteador doméstico.

Roteadores doméstico, sem dúvida, são muitíssimo úteis. Têm um custo benefício muito bom para quem tem alguns computadores em casa, e deseja compartilhar a conexão à internet. Podem ser usados em ambientes domésticos, ou quem sabe, em um pequeno escritório com duas, três ou quatro máquinas. Se esse é seu objetivo, vale mesmo a pena comprar um. Contudo, se você pensa que com esse tipo de roteador poderá ligar várias redes, está enganado. Embora o roteador doméstico na figura tenha quatro portas LAN, isso não significa (embora pareça) que você poderá ligar quatro LANs nele... e sim, que você poderá ligar quatro máquinas. Este roteador, na verdade, é um acoplamento de um roteador de duas portas e um comutador. As quatro portas LAN são portas de um computador, portanto. O que isso significa? Bom, significa muita coisa.

Primeiro, isso significa que um roteador doméstico com quatro portas LAN e uma WAN não é a mesma coisa que um roteador "de verdade", digamos assim. As quatro portas LAN são uma única porta de roteador, possuem domínio de broadcast compartilhado, possuem um único servidor DHCP rodando e, enfim, são portas de um comutador, em uma única rede. É como se você tivesse um comutador de quatro portas ligado a uma porta do roteador.

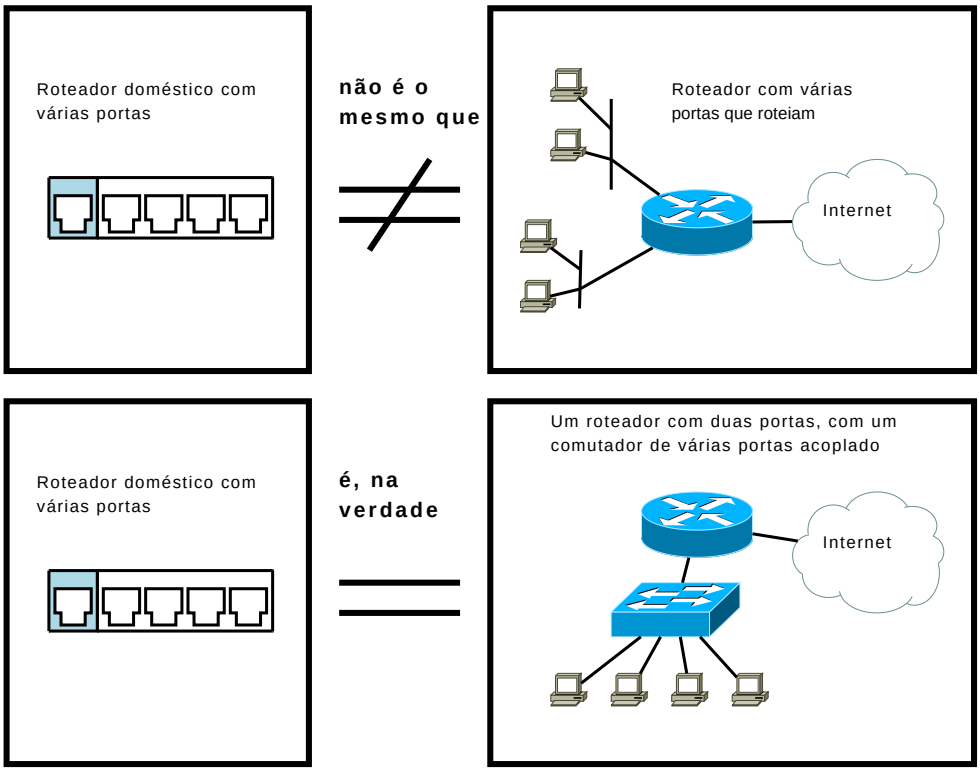

Figura 6.9. O que um roteador doméstico é, e o que não é.

Um roteador doméstico é, na verdade, a junção de um roteador de duas portas, e um comutador de quatro portas; o comutador está internamente ligado à segunda porta do roteador. No roteador doméstico de quatro portas LAN, você configurará apenas um servidor DHCP, e não quatro. A configuração da porta WAN, contudo, continua a mesma.

Outra diferença de roteadores "tradicionais" e roteadores domésticos é que, em roteadores tradicionais, você pode configurar qualquer porta como WAN ou LAN; em roteadores domésticos, a porta WAN já vem de fábrica assim designada, bem como as portas (ou seja, a porta) LAN. Você não pode usar as portas (a porta) LAN para agir como WAN; também não pode configurar a porta WAN a agir como uma porta LAN, com um servidor DHCP rodando etc.

Enfim, lembre-se sempre:

Nota 6.7. Roteadores domésticos possuem apenas duas portas, uma WAN e uma LAN, independente de quantas "portas LAN" ele alega ter. As várias portas LAN de um roteador doméstico nada mais são do que portas de um comutador, o que equivale a apenas uma porta com capacidade de roteamento.

6.6. Conclusão

Neste capítulo você aprendeu o que é um Gateway padrão. Não é obrigatório ter um Gateway padrão em uma rede, mas se você quiser que as máquinas da LAN falem com o mundo além da rede local, é necessário que todas elas saibam quem é o Gateway padrão.

Um Gateway padrão pode ser qualquer porta, seja ela de um computador ou de um roteador, desde que esta porta seja visível na rede local.

Existe uma diferença básica entre computadores comuns e um roteador. A capacidade de encaminhamento de um roteador é bem maior da dos computadores comuns; logicamente, um computador comum pode funcionar como um roteador se o sistema operacional for capaz e assim estiver configurado.

Um roteador pode ter uma, duas ou mais portas. Cada uma das portas pode ser configurada independente da outra, e podem agir como portas LAN ou WAN. Por exemplo, em um roteador com quatro portas, você pode configurar uma delas como WAN e as outras como LAN, ou duas como WAN, ou, ainda, todas como LAN; a configuração depende de você, operador do roteador; o roteador não vem de fábrica com as portas configuradas. Todas as portas são iguais e possuem a mesma capacidade de configuração.

Geralmente, a porta LAN do roteador é configurada com IP estático, bem como a máscara; e é também configurado para funcionar como servidor de DHCP, afim de distribuir as informações que as máquinas da rede local precisam. E essas informações são: IPs e máscara, Gateway padrão e, opcionamente porém recomendado, endereço do DNS. Logicamente, o endereço do Gateway padrão fornecido pelo servidor DHCP do roteador é o endereço lógico da própria porta LAN que está na rede local onde se encontram as máquinas clientes.

Diferente da porta LAN do roteador, na maioria das vezes na vida real, a porta WAN é configurada dinamicamente, sendo um cliente DHCP. Entretanto, pode acontecer, uma vez ou outra, que o operador do roteador precise configurar estaticamente configurações de IP na porta WAN; isso é raro, contudo. A porta WAN, por meio do DHCP, obterá, do provedor de acesso à internet, o IP, a máscara, o Gateway padrão e o DNS. Esse endereço de DNS aprendido pela porta do roteador quase sempre será repassado para o servidor DHCP da porta LAN, para que as máquinas da rede local também conheçam o DNS. Isso implica que as máquinas da rede local usam o mesmo DNS que o roteador usa.

Finalmente, existe uma grande diferença entre roteadores "tradicionais" e roteadores domésticos. Roteadores tradicionais são ideais para interligar redes entre si e à internet; roteadores domésticos são ideais para interligar computadores da rede local entre si e à internet. Não importa quantas portas LAN um roteador doméstico alega ter; na verdade, ele possui uma única porta LAN interna, ligada a um comutador; existe uma única rede local; existe um único servidor DHCP rodando. É nesse comutador que as máquinas são ligadas. Roteadores domésticos são ideais para ambientes domésticos ou pequenos escritórios, tendo um ótimo custo-benefício.

Para ambientes empresariais e para interligar várias redes, como já foi dito, é melhor um roteador tradicional, como todas as portas com capacidade de roteamento.

6.7. EXERCÍCIOS

Exercício 6.1. Observe a figura abaixo.

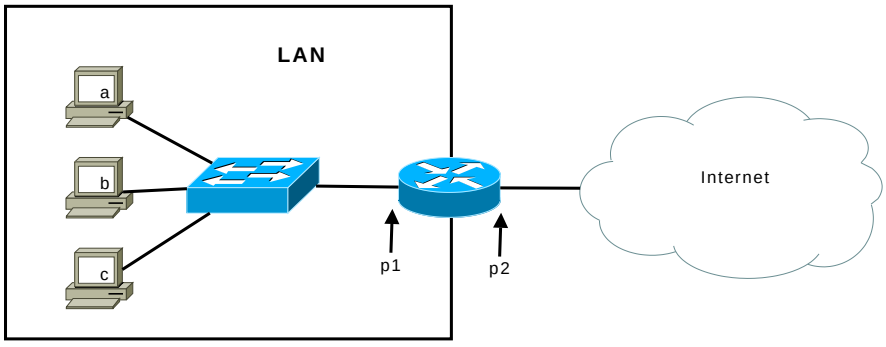

Figura 6.10. Representação de LAN e WAN.

Marque a(s) alternativa(s) correta(s):

a) O Gateway padrão da rede local é `a`.

b) O Gateway padrão da rede local é `p1`.

c) O Gateway padrão da rede local é `p2`.

Exercício 6.2. Ainda usando a figura do exercício anterior, marque a(s) alternativa(s) correta(s):

a) Se `a` quer falar com `b`, ele envia um quadro para o Gateway padrão.

b) Se `p1` quer falar com `b`, ele envia um quadro para o Gateway padrão.

c) Se `b` quer falar com `a`, ele envia um quadro para `a`.

d) Se `c` quer falar com um computador na internet, ele envia um quadro para o Gateway padrão.

Exercício 6.3. Observe a figura abaixo:

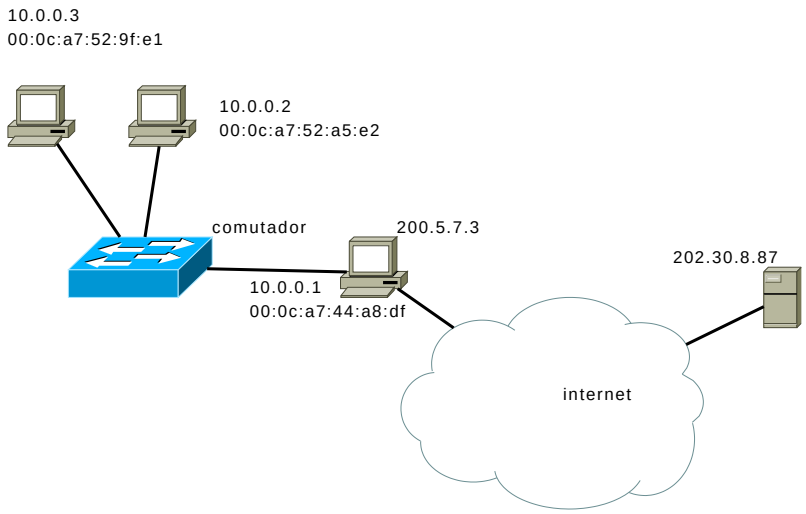

Figura 6.11. LAN conectada à internet através de um computador.

Marque a(s) alternativa(s) correta(s):

a) Para que a máquina cujo endereço lógico é 10.0.0.3 fale com a máquina de endereço lógico 10.0.0.2, um quadro de camada enlace é enviado contendo o endereço físico destino igual a 00:0c:a7:52:a5:e2.

b) Para que a máquina cujo endereço lógico é 10.0.0.3 fale com a máquina de endereço lógico 202.30.8.87, um quadro de camada enlace é enviado contendo o endereço físico destino igual a 00:0c:a7:44:a8:df, ou seja, o endereço do Gateway padrão.

c) Para que a máquina cujo endereço lógico é 10.0.0.3 fale com a máquina de endereço lógico 202.30.8.87, um pacote de camada rede é enviado contendo o endereço lógico de destino igual a 10.0.0.1, ou seja, o endereço do Gateway padrão.

d) Para que a máquina cujo endereço lógico é 10.0.0.3 fale com a máquina de endereço lógico 202.30.8.87, um pacote de camada rede é enviado contendo o endereço lógico de destino igual a 202.30.8.87.

Exercício 6.4. Marque a(s) alternativa(s) correta(s) quanto à capacidade de roteamento de um computador comum e um roteador.

a) Um computador comum pode tomar muitas decisões, enquanto um roteador toma apenas duas decisões: encaminhar diretamente para uma máquina da rede local, ou encaminhar para o Gateway padrão.

b) Um roteador pode tomar muitas decisões, enquanto um computador comum toma apenas duas decisões: encaminhar diretamente para uma máquina da rede local, ou encaminhar para o Gateway padrão.

c) Qualquer computador pode vir a funcionar como um roteador quanto ao número de deciões: basta o sistema operacional suportar e estar configurado para isto.

d) Não há diferenças entre um computador comum e um roteador.

Exercício 6.5. O que geralmente configuramos na porta LAN de um roteador? (marque uma ou mais alternativas)

a) Cliente DHCP

b) Servidor DHCP

c) Ip estático

d) Máscara dinâmica

e) Não precisamos configurar nada, o roteador vem de fábrica com uma configuração funcional

Exercício 6.6. O que geralmente configuramos na porta WAN de um roteador? (marque uma ou mais alternativas)

a) Cliente DHCP

b) Servidor DHCP

c) Ip estático

d) Máscara dinâmica

e) Não precisamos configurar nada, o roteador vem de fábrica com uma configuração funcional

Exercício 6.7. Observe a figura abaixo:

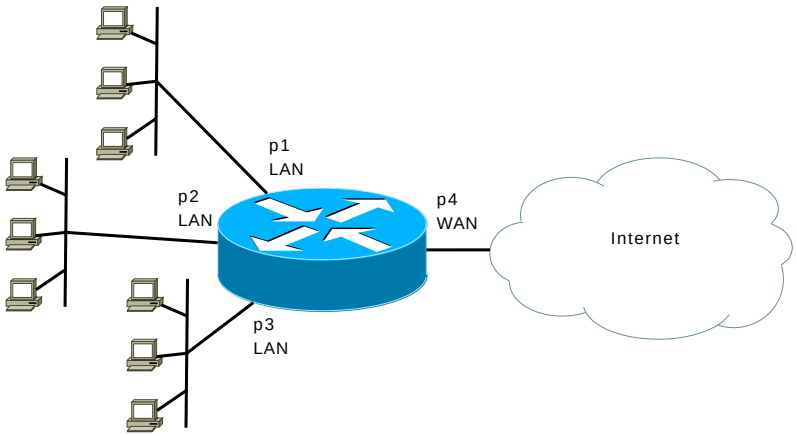

Figura 6.12. Várias LANS interconectadas entre si e à internet.

Marque a(s) alternativa(s) correta(s):

a) O esquema representa o comportamento de um roteador tradicional.

b) O esquema representa o comportamento de um roteador doméstico.

c) Provavelmente, existem três servidores DHCP rodando.

d) Existe certamente um único servidor DHCP rodando, visto que o servidor DHCP roda um serviço por roteador.

e) Existe provavelmente um cliente DHCP rodando.

f) O roteador poderia ter qualquer uma das portas configuradas como LAN ou WAN, pois roteadores tradicionais não fazem diferença entre elas, e todas possuem a mesma capacidade de configuração.

Exercício 6.8. Observe a figura abaixo, que representa um roteador doméstico com cinco portas, sendo uma porta WAN e quatro portas LAN.

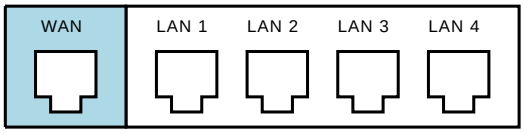

Figura 6.13. Roteador doméstico.

Marque a(s) alternativa(s) correta(s):

a) A porta WAN pode funcionar como cliente DHCP.

b) Este roteador pode interligar até quatro LANs diferentes.

c) Este roteador pode interligar até quatro máquinas, entretanto elas estarão na mesma LAN.

d) Este roteador pode rodar até quatro servidores DHCP.

Exercício 6.9. Qual a melhor definição para roteador doméstico com uma porta WAN e quatro portas LAN?

a) É um roteador de cinco portas com capacidade de rotear em todas elas (cinco redes).

b) É um roteador de duas portas, sendo uma para WAN e outra interna, ligada com um comutador de quatro portas.

c) É um roteador de cinco portas, com capacidade de rotear em todas elas (cinco redes), contudo, uma porta é reservada para configurações de WAN.

Capítulo 7

Padrões de Redes Locais

Capítulo 8

Protocolo de Camada Enlace Ethernet

Parte III

WANs IPv4

Capítulo 9

Conceitos de IPv4

9.1. Introdução

Segundo consta, quando a arquitetura TCP/IP foi lançada, os dois protocolos eram uma coisa só. Aí veio a versão 2, a 3, e finalmente os protocolos separaram-se na versão 4. A versão 4 destes protocolos, notavelmente do protocolo IP, é a versão mais usada em todo mundo. A internet é IP versão 4: não porque isso foi planejado, e sim porque as coisas desenvolveram-se naturalmente sobre o IP versão 4.

Neste capítulo, estudaremos a versão 4 do IP. Você, caríssimo redista contemporâneo, está vivendo um momento de já-ainda-não, um momento de transição do IPv4 para a nova versão do IP, versão 6. O número de endereços IPv4 disponíveis está prestes a acabar, e a nova versão (a versão 6) é de pouco conhecimento geral. Mais do que isso, esse é um momento complicado para você pois terá de aprender as duas versões: a que está em uso, e a que será usada. Redistas anteriores a você precisavam saber apenas sobre IPv4. Redistas do futuro estudarão apenas IPv6. Mas você precisará saber as duas versões, explicar sobre as duas, implementar as duas... em suma: a barra está pesada pro seu lado.

Neste capítulo, que é básico sobre o IPv4, analisaremos os seguintes aspectos:

- Formato de endereçamento
- Divisão de rede e máquina
- Comunicação dentro e fora da rede local
- E outros...

9.2. Formato de endereçamento

IP é abreviação para Internet Protocol. A versão 4 deste protocolo usa um formato de endereço que consiste em quatro octetos. Cada octeto contém 8 (dã) bits, o que significa que o tamanho total do endereço IP é de 32 bits (faça os cálculos). Cada octeto é separado do outro por um ponto simples. Um exemplo de endereço IP é 192.168.0.30, como mostrado na figura abaixo.

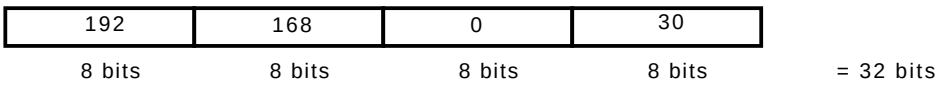

Figura 9.1. Exemplo de endereço IPv4.

Os valores que cada octeto pode assumir são de 0 até 255, ou seja, um total de 256 valores. Isso significa que o número total de endereços IP versão 4 possíveis é de 256^4, ou 2^{32}, ou ainda, 4.294.967.296 (4 bilhões e alguma coisa). Aparentemente, isso é endereço pra caramba.

Endereço IP versão 4 é usado na internet; as máquinas ligadas à internet usam esse endereço para conversarem. Teoricamente (sim, você vai ler a palavra "teoricamente" muitas vezes neste capítulo) cada máquina precisa de um endereço IP para conversar com outra, um endereço único. Assumindo que temos capacidade para 4 bilhões e alguma coisa de endereços, e assumindo também que o número de pessoas em 2009 no mundo é de aproximadamente 7 bilhões, e ainda, o total de pessoas conectadas à internet é de 2 bilhões e esse número cresce assustadoramente, começamos a perceber que a quantidade de endereços IP versão 4 não é tão grande assim.

Agora, vamos voltar à parte técnica não-antropológica do endereçamento IPv4: o seguinte endereço:

```
200.259.5.300
```

Não é válido, pois o segundo octeto (259) e o último (300) extrapolam a capacidade do octeto, que vai de 0 a até 255.

9.3. DIVISÃO DE REDE E MÁQUINA

Logicamente, toda máquina faz parte de uma rede. Se você pensar em cada máquina do mundo tendo um endereço IP aleatório, logo perceberá o caos que é: endereços sem nenhum tipo de relacionamento com os outros. Agora, se você pensar que os endereços são organizados por rede, perceberá como fica fácil gerenciar o endereçamento IP. Por exemplo, endereços semelhantes, como 192.168.0.30 e 192.168.0.3, podem significar (e significam!) máquinas que estejam na mesma rede. Da mesma forma, 192.168.0.30 e 192.200.0.30 podem significar máquinas que estejam em redes diferentes, mas que pertencem a uma única empresa, ou estejam em uma única localidade.

O endereçamento IP versão 4 divide-se em duas partes: a parte que identifica a rede, e a parte que identifica a máquina. Essas duas partes podem assumir diverssos tamanhos. No exemplo abaixo, temos um endereço IP que aloca 3 octetos (24 bits) para identificar a rede, e 1 octeto (8 bits) para identificar a máquina:

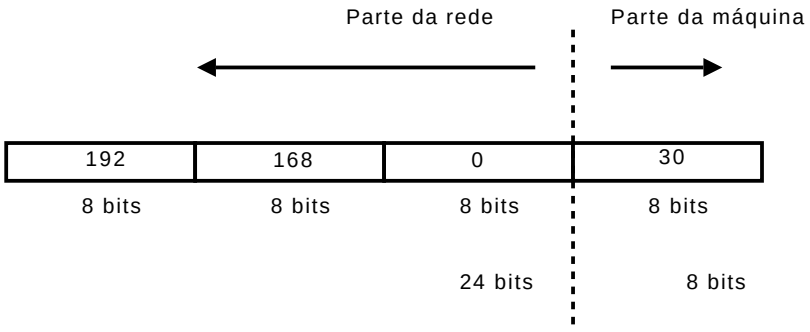

Figura 9.2. Parte de rede e parte da máquina.

Assim, 192.168.0 é a parte da rede, e as máquina usam o último octeto para se identificarem. No caso, o número da máquina é 30. Poderíamos ter outras máquinas nesta mesma rede: basta manter idêndica a parte da rede, e mudar a parte da máquina (pois, como você sabe, as máquinas devem ter endereços exclusivos, que não se repitam):

```
192.168.0.50
```

```
192.168.0.240
```

```
192.168.0.1
```

Agora, o seguinte endereço IP:

```
192.168.0.277
```

Não é válido, pois extrapola a capacidade de um octeto, que vai de 0 a 255. E o endereço abaixo:

```
192.168.1.55
```

Embora seja um endereço IP válido, a máquina não está na mesma rede, pois a parte da rede não é 192.168.0.

9.4. Máscara de rede

Bom, nem todas as redes são como as do exemplo mostrado anteriormente: três octetos para rede e um octeto para a máquina. Na verdade, podemos ter quantos octetos[9.1] forem necessários para rede. Mas então, como a máquina vai saber qual a parte de rede e qual a parte de máquina (host)? Através da máscara de rede. Cada máquina da rede possui um endereço IP e uma máscara: são duas informações de camada rede indispensáveis!

A máscara de rede possui o mesmo formato do endereço IP: quatro octetos de 8 bits cada. Quando o octeto é 255, significa que a parte do endereço IP correspondente é rede. Se o octeto da máscara for um 0, significa que o octeto do IP correspondente é máquina. Por exemplo, veja a figura abaixo:

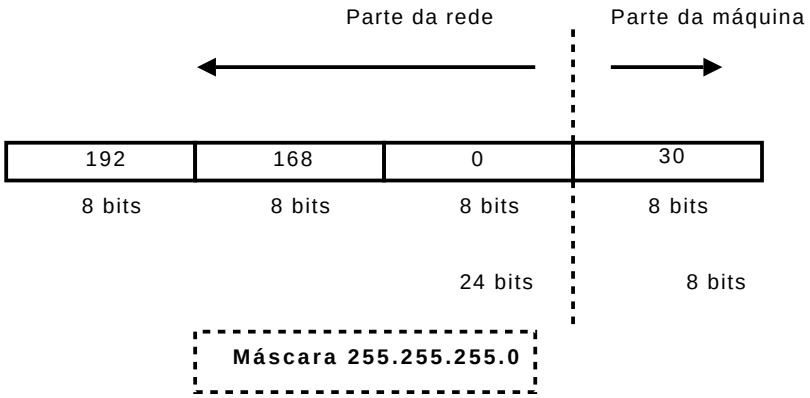

Figura 9.3. Máscara de rede 255.255.255.0.

9.1. "quantos BITS forem necessários".

Temos o mesmo endereço de rede do exemplo, e a máscara cujos três primeiros octetos são 255. Assim, 192.168.0 são a parte do endereço que corresponde à rede, visto que a máscara de rede nessas posições é 255. Agora, "30" é a parte da máquina, pois na máscara, temos um 0.

A lógica nos diz que não podemos ter a máscara de rede 255.0.255.255, pois isso complicaria muito. Dividimos o endereço em parte de rede, e em parte de máquina. A parte de rede vem primeiro, e a parte de máquina vem depois.

```
Endereço IP:      192.168.0.30
Máscara:          255.255.255.0
Parte da rede:    192.168.0 (pois a máscara é 255 nestes octetos)
Parte da máquina: 30 (pois a máscara é 0 neste octeto)
```

Observe também, que uma rede cuja máscara é 255.255.255.0 pode conter até 256 endereços (de 0 a 255) para máquinas **teoricamente**, pois um octeto tem essa capacidade.

Você deve estar pensando: bom, será que se eu modificar a máscara de rede, poderei aumentar ou diminuir a parte do endereçamento que representa a rede? A resposta é sim! Observe a figura abaixo:

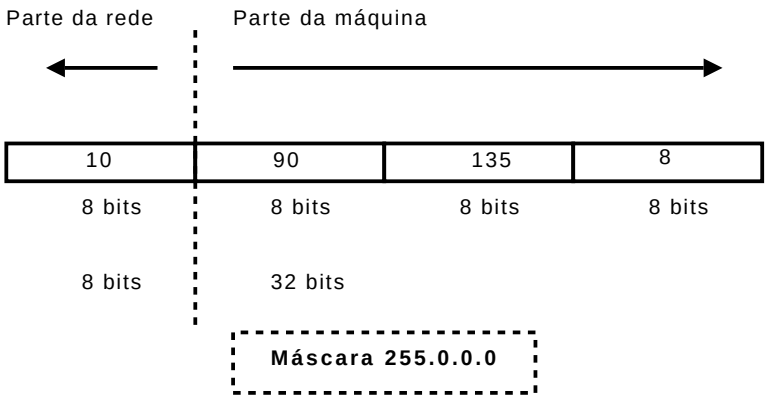

Figura 9.4. Máscara de rede 255.0.0.0.

Neste exemplo, a máscara é 255.0.0.0. Somente um octeto é reservado para a rede; três são reservados para máquinas.

```
Endereço IP:      10.90.135.8
Máscara:          255.0.0.0
Parte da rede:    10 (pois a máscara é 255 neste octeto)
Parte da máquina: 90.135.8 (pois a máscara é 0 nestes octetos)
```

Observe que a rede deste exemplo (a rede 10) tem capacidade para 256^3, ou 2^{24}, ou ainda, 16.777.216 (dezesseis milhões e alguma coisa), pois essa é a capacidade de três octetos. Se a máscara fosse 255.255.0.0, teríamos 256^2, ou 2^{16}, ou ainda, 65.536 endereços para máquinas, **teoricamente**.

Aviso 9.1. Embora você possa dividir o endereço IP em duas partes (rede e máquina), não se esqueça de que, ao referenciar a máquina, o endereço completo é usado, isto é, todos os quatro octetos.

Você pode ver as informações de camada rede (IP, máscara e outras coisas) no seu computador Linux usando o comando `ifconfig`, se for root, ou `/sbin/ifconfig`. Basta digitar e analisar a saída do comando. Abaixo, um exemplo de saída.

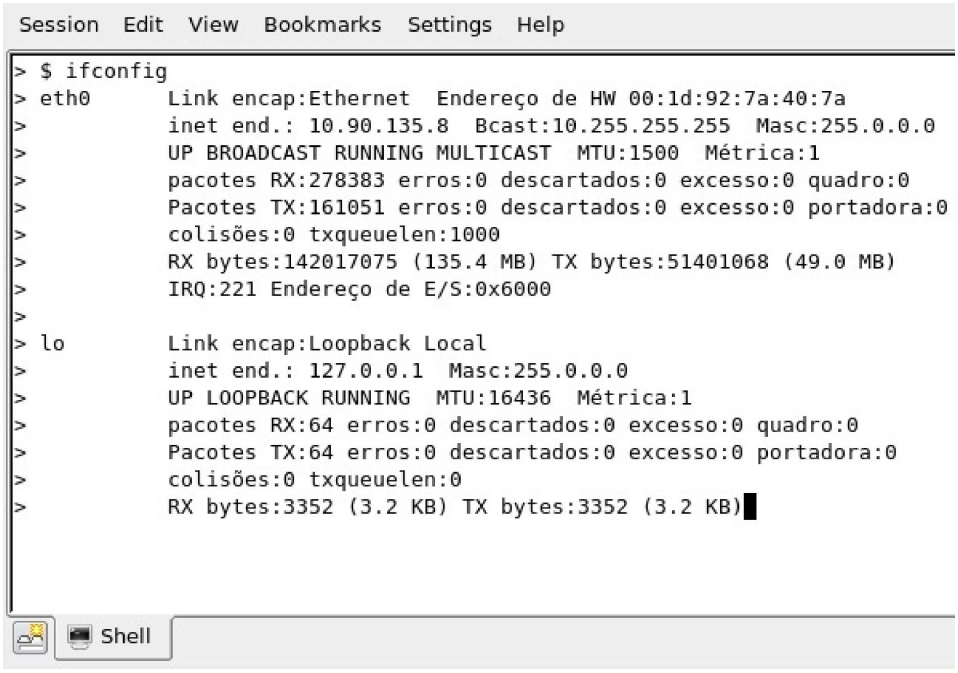

Figura 9.5. Saída do comando ifconfig no Linux.

A placa de rede "real" da máquina, neste exemplo, é a `eth0`. Você pode ver o endereço físico (MAC) da placa de rede, bem como o endereço IP (inet end) 10.90.135.8, e a máscara, que é 255.0.0.0. Você já é capaz de descobrir qual é a parte do endereço que representa a rede, e qual a parte que representa a máquina. Faça isso.

Existem outras informações na saída do comando `ifconfig` que ainda não vimos, como "`Bcast 10.255.255.255`". Bcast é abreviação de broadcast.

9.5. Endereço de rede e de broadcast

Nem todos os IPs disponíveis em uma rede podem ser usados. Por exemplo, considere o endereço de IP abaixo com a respectiva máscara:

```
Endereço: 192.168.0.30

Máscara:  255.255.255.0
```

```
Parte da rede: 192.168.0

Parte da máquina: 30
```

Segundo consta, resta um octeto (256 endereços) para atribuição de máquinas. Esse endereço pode ir, teoricamente (sei, isso já está enjoando) de 192.168.0.**0** até 192.168.0.**255**, certo? Mas o primeiro endereço da rede não é usado, pois representa a própria rede. E o último endereço também não pode ser usado, pois é um endereço de broadcast, ou seja, representa todas as máquinas da rede.

Tá complicado?

No exemplo acima, poderíamos dizer que o endereço IP da rede é 192.168.0.0, pois este é o primeiro endereço de host. Assim, se lhe perguntasemm a que rede pertence a máquina 192.168.0.30 máscara 255.255.255.0, você poderia responder: pertence à rede 192.168.0.0. da mesma forma, se lhe perguntassem em qual rede está a máquina 10.90.135.8 máscara 255.0.0.0, você responderia que está na rede 10.0.0.0. O número da rede sempre é o primeiro endereço da parte de máquina. Observe o esquema abaixo:

```
Endereço completo da máquina: 10.90.135.8

Máscara: 255.0.0.0

Parte da rede: 10

Parte da máquina: 90.135.0

Primeiro endereço disponível na parte da máquina: 0.0.0

Endereço de rede: 10.0.0.0
```

Já o endereço de broadcast é um endereço que pode ser usado para referenciar todas as máquinas da rede. Por exemplo, considere a rede 192.168.0.1, cuja máscara é 255.255.255.0. O último endereço da parte de máquina é o endereço que representa todas as máquinas da rede, isto é, broadcast. Há duas situações em que uma máquina processa o pacote e desencapsula dados do mesmo, enviando-o à camada imediatamente superior:

1. Quando o endereço destino é igual ao endereço da própria máquina.

2. Quando o endereço destino é broadcast.

Logo, quando uma máquina envia um pacote de broadcast, o pacote é ecoado para toda a rede, e todas as máquinas o aceitam, pois o sistema operacional das máquina foi programado para (seguindo a norma) aceitar pacotes broadcasts. Felizmente, tais pacotes não saem da rede local, pois senão a internet seria um caos.

Analisemos, pois a situação abaixo, passo por passo, para facilitar as coisas.

```
Endereço completo da máquina: 192.168.0.1

Máscara: 255.255.255.0

Parte da rede: 192.168.0
```

```
Parte da máquina: 1
Primeiro endereço disponível na parte de máquina: 0
Último endereço disponível na parte da máquina: 255.
Endereço de rede: 192.168.0.0
Endereço de broadcast: 192.168.0.255
```

Agora, para você gravar o que foi feito:

Exercício 9.1. Complete a lista abaixo

```
Endereço completo da máquina: 15.5.88.139
Máscara: 255.0.0.0
Parte da rede: _______________________________
Parte da máquina: ____________________________
Endereço de rede: ____________________________
Endereço de broadcast: _______________________
```

Exercício 9.2. Continue completanto (observe a máscara de rede)

```
Endereço completo da máquina: 110.10.1.101
Máscara: 255.255.0.0
Parte da rede: ________________________________________
Parte da máquina: _____________________________________
Endereço de rede: _____________________________________
Endereço de broadcast: ________________________________
```

Definição 9.2. *Endereço de rede. É o endereço que representa a rede, o primeiro endereço desta; não pode ser aplicado em máquinas.*

Definição 9.3. *Endereço de broadcast. Em IPv4, é o endereço que representa todas as máquinas da rede. No IPv6 não existe. É usado (obviamente, somente no IPv4) quando se quer enviar um pacote para todas as máquinas, sem ter de digitar endereço por endereço. É o último endereço da rede.*

Como não podemos aplicar o endereço de rede, nem o endereço de broadcast em uma máquina da rede, você sempre deve diminuir 2 endereços quando contar quantas máquinas podem existir na rede. Por exemplo, embora a rede 192.168.0.0 máscara 255.255.255.0 possa **teoricamente**(!) conter 256 endereços (pois esta é a capacidade de um octeto, de 0 até 255), a verdade é que na vida real só poderão existir 254 máquinas nesta rede: de 1 até 254.

A tabela abaixo ilustra três exemplos da capacidade das redes. Observe que sempre diminuimos dois endereços.

Rede, máscara	Capacidade
9.0.0.0, 255.0.0.0	$256^3 - 2$, ou $16.777.216 - 2$, ou 16.777.214.
122.25.0.0, 255.255.0.0	$256^2 - 2$, ou $65.536 - 2$, ou 65.534.
201.8.75.0, 255.255.255.0	$256 - 2$, ou 254.

Tabela 9.1. Capacidade das redes.

Agora, volte à figura 1.5, que mostra a saída do comando ifconfig no Linux. Lá consta o endereço de broadcast da rede, embora não conste o endereço da rede propriamente dito.

9.6. COMUNICAÇÃO DENTRO E FORA DA REDE LOCAL

As máquinas usam a máscara de rede, o próprio endereço IPv4 e o endereço de destino para saber se o pacote deve ser enviado para alguma máquina da rede local, ou fora da rede. "Mas como é que as máquinas conseguem fazer isso?".

Observe a figura abaixo:

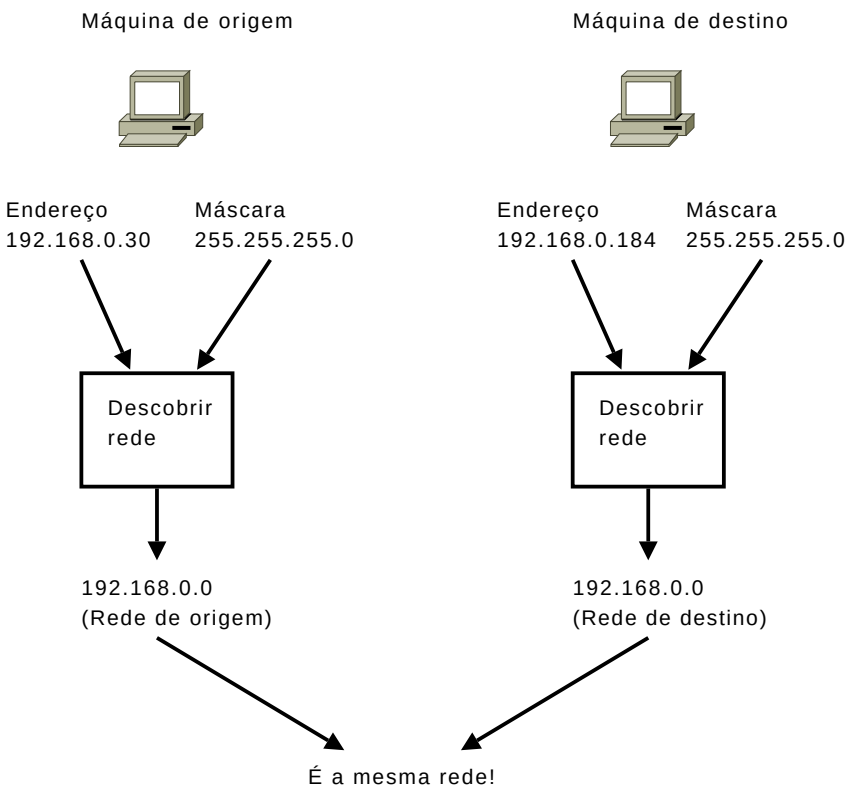

Figura 9.6. Enviando para uma máquina da mesma rede.

Neste exemplo, o ip da máquina de origem é 192.168.0.30, máscara 255.255.255.0, o que significa que o endereço de rede desta máquina é 192.168.0.0, conforme ilustrado na figura. A máquina executa uma função (representada por um retângulo na figura) para descobrir qual a própria rede.

O segundo passo é descobrir a rede da máquina destinatária. Pega-se o ip de tal máquina, a máscara **da própria rede que a máquina remetente se localiza**, joga-se na função e descobre-se em que rede a máquina destinatária se localiza. Em nosso exemplo, a rede de origem e a rede de destino são as mesmas: 192.168.0.0. portanto o pacote será enviado diretamente para a máquina destinatária, que está na rede local.

Observe, agora, a figura abaixo para um cenário diferente:

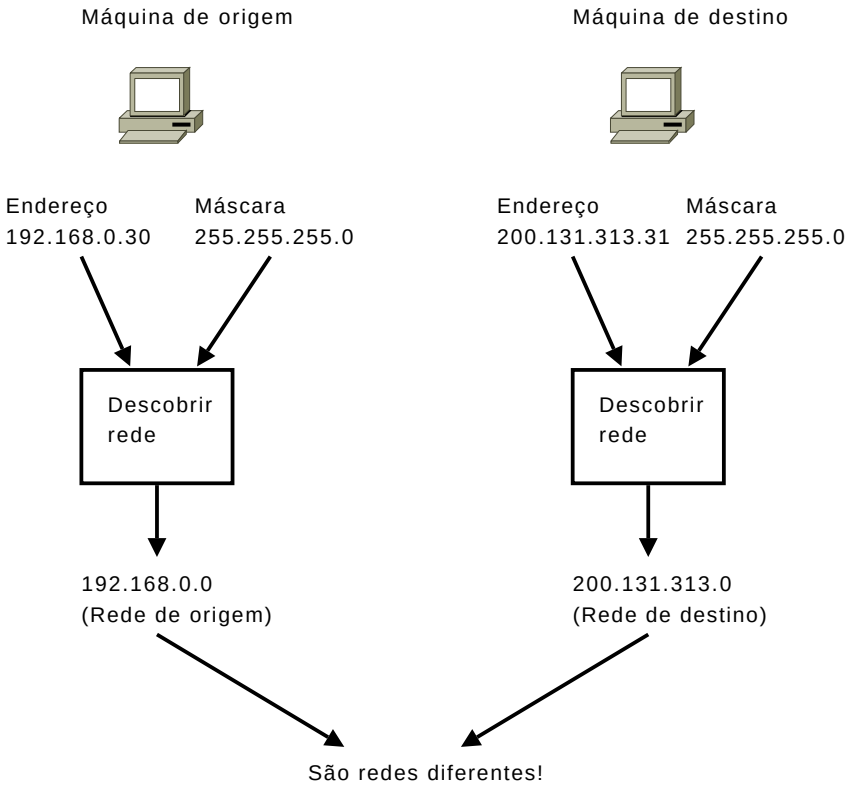

Figura 9.7. Enviando para uma máquina em uma rede diferente.

O exemplo acima está claro. A máquina de origem descobre, através da função representada pelo retângulo, a própria rede e a rede em que a máquina destinatária se localiza. Como são redes diferentes, então a máquina não envia o pacote diretamente para a máquina, e sim, para o Gateway padrão. A máquina destinatária está fora da rede.

9.7. O pacote IPv4 - Explicação introdutória

Como você já deve imaginar, o pacote IPv4 contém, dentro de si, um segmento de camada transporte. Pacotes IP são interpretados por roteadores; sendo assim, quando um roteador recebe um pacote, ele deve saber para onde enviá-lo. Um pacote precisa, portanto, de um endereço de destino; também de um endereço de origem, para que a máquina receptora possa responder à máquina remetente. Um pacote não precisa de máscara de rede, pois ele não tem inteligência própria; quem precisa de máscaras são as máquinas da rede.

Uma última coisa que o pacote precisa é um campo para que a máquina de destino saiba para onde enviar as informações na camada superior. Ou seja, o pacote tem um campo com um código de protocolo de camada transporte, para saber se deve enviar as informações via TCP ou UDP, ou outro protocolo que exista[9.2]. Existem outros campos, que não analisaremos neste capítulo, como o *checksum*, o campo *versão*, o campo *tamanho*, entre outros. Adiantando, o campo versão indica qual a versão do protocolo IP do pacote; para redes IPv4, a versão é, obviamente, 4.

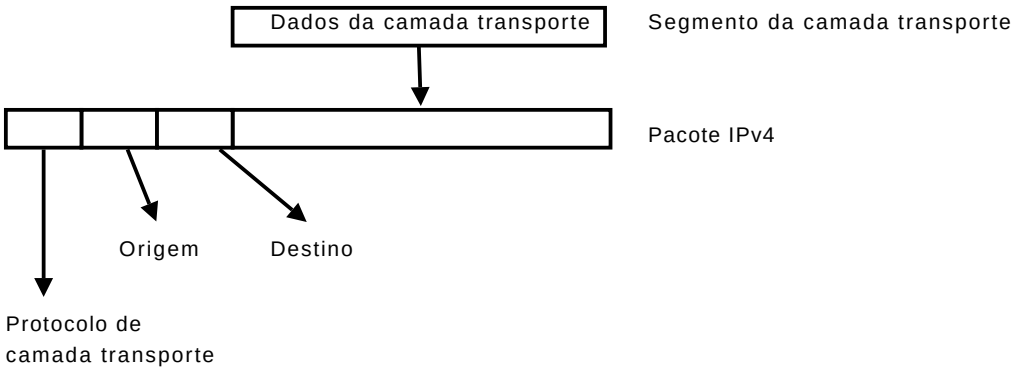

Figura 9.8. Pacote IP resumido (campos foram propositalmente ocultados).

Posteriormente neste curso, estudaremos mais campos do protocolo IP.

9.8. CONCLUSÃO

Neste capítulo, tivemos uma bela introdução ao IP versão 4. Muita informação, não acha? E isso é só o começo.

Você viu o formato de endereçamento IPv4. Um endereço IPv4 é formado por quatro octetos de oito bits; você verá em breve neste curso o que significa isso. Por agora, você já sabe que esse octetos tem capacidade para 256 números cada: de 0 a 255. Você também viu que pode reservar alguns octetos para a rede, e outros para as máquinas. Quanto mais octetos, mais capacidade de endereçamento temos. por exemplo, se for usado um único octeto para a parte da rede e três para as máquinas, temos capacidade de ter dezesseis milhões e alguma coisa de máquinas em uma única rede.

Pois é... a máscara de rede é uma coisa muito útil, para que se possa definir a parte da rede e a parte de máquina. Você viu que quando na máscara o octeto é 255, o mesmo octeto do endereço IP é parte de rede; se na máscara o octeto for 0, o mesmo no endereço equivale à máquina. As máquinas, quando querem enviar um pacote, inserem o endereço dela própria e a máscara em uma função, afim de descobrir o endereço de rede da mesma; faz a mesma coisa com a máquina detsinatária. Se as máquinas estiverem na mesma rede, então envia-se o pacote diretamente; caso contrário, envia para o Gateway padrão.

O primeiro endereço de rede é sempre o endereço da própria rede, enquanto que o último endereço, é o endereço de broadcast.

9.2. Protocolos multimídia para streaming etc.

9.9. Exercícios

Exercício 9.3. Observe a figura abaixo e marque um X nas frases que sejam verdadeiras:

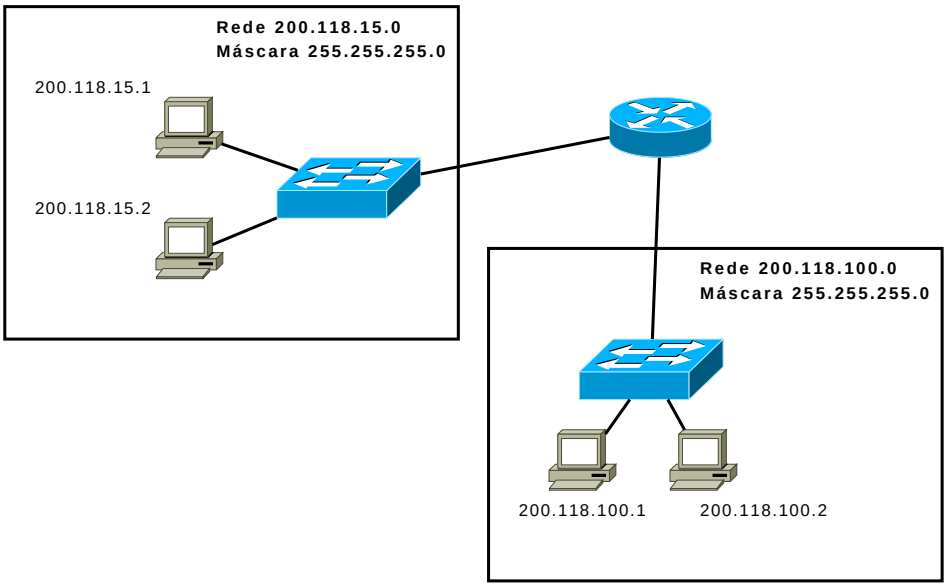

Figura 9.9. Duas redes conectadas por um roteador.

a) A máquina 200.118.15.1 envia um pacote diretamente para a máquina 200.118.15.2

b) A máquina 200.118.15.1 envia um pacote diretamente para a máquina 200.118.100.1

c) A máquina 200.118.15.1 NÃO envia um pacote diretamente para 200.118.15.2, e sim, para o Gateway padrão da rede 200.118.15.0

d) A máquina 200.118.15.1 NÃO envia um pacote diretamente para 200.118.15.2, e sim, para o Gateway padrão da rede 200.118.100.0

e) A máquina 200.118.15.1 NÃO envia um pacote diretamente para 200.118.100.1, e sim, para o Gateway padrão da rede 200.118.15.0

f) A máquina 200.118.15.1 NÃO envia um pacote diretamente para 200.118.100.1, e sim, paea o Gateway padrão da rede 200.118.100.0.

Exercício 9.4. Informados os endereços ip e as máscaras, escreva o endereço de rede e de broadcast:

a) 10.13.5.8 255.255.255.0

b) 15.3.3.4 255.0.0.0

c) 129.30.50.1 255.255.0.0

d) 202.101.55.2 255.255.255.0

Exercício 9.5. Qual(is) opção(ões) abaixo é um campo que não é necessário no pacote IPv4?

a) Origem

b) Destino

c) Protocolo de camada enlace

d) Máscara

Capítulo 10

O Sistema de Numeração Binário

10.1. tenho mesmo que estudar isso?

Sim. O sistema de numeração binário está envolvido com o endereçamento IPv4 e quiçá (gosto desta palavra) IPv6. Para montar redes robustas e estáveis, é preciso saber dimensioná-las, dividi-la em subredes, e para isso você deve conhecer o sistema de numeração binário. Quando eu digo que binários são inseparáveis de endereços IP, é porque é verdade. Um redista que não sabe binário é como hamburguer sem gordura, escova sem dentes, macaco sem banana, ping sem pong, "tan-taranan-tan" sem o "tan-tan" e por aí vai. Você consegue viver num mundo triste assim? Não! Não foi isso que nossos pais nos ensinaram! Portanto, estude binário.

10.2. Introdução

Não tenha medo. Não é uma aula de matemática que vai torrar seu cérebro e fundir seu pâncreas. É apenas uma aula de redes... que irá torrar seu cérebro e fundir seu pâncreas, sem dúvida. O sistema de numerção binário é um problema para você, pois em vez de usar dez algarismos, usa apenas dois. Isso se torna um problema pois você tem dez dedos, e aprendeu a pensar de forma 10. Temos duas soluções propostas:

a) Ampute quatro dedos de cada mão e pé; ou

b) Torre seu cérebro para entender o conceito de numeração binária.

Vamos começar com calma para você não ficar nervoso. Considere os 10 algarismos que temos: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. Embora termine em 9, o total são 10, porque contamos o 0 também. O que acontece quando terminamos o número 9? Ora... adicionamos (concatenamos, juntamos, noivamos e casamos) números já existentes. Neste caso, usamos 10! Depois do 19, temos o 20, e por aí vai, até que chegamos ao 99. E agora? Fazemos a mesma coisa! Concatenamos mais uma vez, colocando um 1 e dois zeros para formar 100. E assim, concatenando números cada vez que as possibilidades acabam, podemos formar combinações infinitas.

Com o sistema de numeração binário é a mesma coisa, porém temos apenas dois algarismos, 0 e o 1. Assim, depois do 1, como não há outros algarismo, temos um 10 (não, não leia "dez" pois o certo é "um zero"). Em seguida, temos 11, 100, 101, 110, 111, 1000 e assim por diante. Podemos fazer uma tabela de correspondência entre números do sistema decimal e números do sistema binário:

Decimal	Binário		Decimal	Binário
0	0		7	111
1	1		8	1000
2	10		9	1001
3	11		10	1010
4	100		11	1011
5	101		12	1100
6	110		256	100000000

Tabela 10.1. Tabela de conversão.

Agora, com esse conceito básico (leia o texto acima quantas vezes for necessário), vamos começar a explorar o sistema de numeração binário, pois ele é importante para se trabalhar com endereçamento IPv4. Você deve parar esta leitura quando começar a ver binários por todos os lados, pois se continuar, corre o risco de 001 1 100101 101 0101010 10 1001 1 01 010101 1 010.

10.3. O bit

Em um número binário, como por exemplo o número abaixo:

```
010110
```

cada algarismo é um bit. Assim, o número binário acima tem 6 bits. Pegou?[10.1]

Cada bit tem capacidade para dois valores: 0 e 1. Já dois bits possuem capacidade para quatro valores. Por exemplo, 00, 01, 10 e 11: quatro valores formados com 2 bits.

Continuando, com três bits, temos 8 valores: 000, 001, 010... ah, você entendeu. Como descobrir qual a quantidade de valores que cabem em tantos bits? É fácil. Basta você desenhar o número 2 bem bonitinho em uma folha de papel, e do lado do número 2, um número pequeno, que é a quantidade de bits do número. Aí você eleva 2 a esse número. Por exemplo, olhe denovo o número binário do exemplo acima, de 6 bits... quantos números podemos formar com 6 bits? Basta elevar 2 à sexta potência:

`Quantos números formo com 6 bits?` 2^6 `= 64 números.`

`Quantos números formo com 9 bits?` 2^9 `= 512 números.`

Lembrando o capítulo anterior, agora você deve saber por que um octeto é chamado assim. É chamado assim porque ele possui oito bits. E quantos números é possível formar com 8 bits?

`8 bits ->` 2^8 `= 256 números.`

Você pode usar uma calculadora para converter números decimais em binários e vice-versa, mas é aconselhável que você aprenda a fazer isso mentalmente, como explicaremos mais tarde. Por agora, veja a versão binária de um conhecido endereço IP:

```
Endereço IPv4 em decimal: 192.168.0.1
```

10.1. Tá bom, tá bom! Eu sei que essa não é a definição de bit, mas como vou explicar isso de uma maneira que o leitor não tenha vontade de pular de uma ponte?

`Endereço IPv6 em binário: 11000000.10101000.00000000.00000001`

O endereço IPv4 tem quatro octetos de oito bits cada, totalizando 32 bits. Mágica (ou não), esse é exatamente o número de algarismos constantes na versão binária do endereço IP! Caramba...

10.4. Máscara de rede em binário

Agora um assunto delicado. Você já sabe o que é uma máscara de rede, certo? A máscara define uma parte para rede, e outra para máquina. A máscara também pode ser convertida em binário, conforme tabela abaixo:

	Decimal	Binário
End IPv4	192.168.0.1	11000000.10101000.00000000.00000001
Másc	**255.255.255.0**	**11111111.11111111.11111111.0000000**
End rede	192.168.0.0	11000000.10101000.00000000.00000000
End broad	192.168.0.255	11000000.10101000.00000000.11111111

Tabela 10.2. Exemplo de máscara de rede convertida em binário

Como você deve ter observado, todos os octetos que são 255 na máscara, são convertidos como 00000000 em binário. Assim, você pode ver que a conversão é muito fácil. Todavia, pode acontecer de a máscara de rede em decimal não ter octetos iguais a 255. Para você entender como isso é possível, devemos relembrar a relação entre a máscara e a quantidade de máquinas que a rede suporta.

Por exemplo, a famosa rede 192.168.0.0, de máscara 255.255.255.0. Nesta rede, teoricamente temos capacidade para 256 endereços. Podemos observar isso tão-somente observando a máscara em binário. Vejamos com calma:

`Máscara em decimal: 255.255.255.0`

`Maáscara em binário: 11111111.11111111.11111111.00000000`

`Quantidade de bits reservados para máquinas: 8 bits`

`Quantidade de endereços possíveis na parte de máquina:` 2^8`, ou seja, 256.`

Você entendeu? O que fizemos no exemplo acima? Observamos a máscara em binário e vimos quantos zeros ela tinha (é latinha?). Tem 8 zeros. Ou seja, oito bits reservados para parte de máquina; e 2 elevado a 8 bits é igual a 256. Observe agora o exemplo abaixo:

`Máscara em binário: 11111111.11111111.00000000.00000000`

`Quantidade de bits reservados para máquinas: 16`

`Quantidade de endereços possíveis na parte de máquina:` 2^{16}`.`

Está mais do que claro que a quantidade de máquinas "cabíveis" em uma rede é o número 2 elevado à quantidade de bits 0s na máscara em binário. Para converter a máscara acima em decimal, é fácil: basta lembrar que oito 1s é 255 em binário. A máscara decimal é 255.255.0.0. O endereço de rede não nos interessa, por enquanto.

Agora, observe a máscara de rede em binário do exemplo abaixo:

```
Máscara em binário: 11111111.11111111.11111111.11000000
Quantidade de bits reservados para máquinas: _____
Quantidade de endereços possíveis na parte de máquina: ______
```

Você conseguiu completar? Muito simples. A quantidade de bits é 6; logo, o número de endereços reservados para máquinas é 2^6, isto é, 64. Você pode observar, portanto, que podemos ter muito mais máscaras do que as três que vimos até agora; neste exemplo, a máscara convertida para decimal é 255.255.255.192. Vide tabela abaixo:

Número em binário	Número em decimal
00000000	0
10000000	128
11000000	192
11100000	224
11110000	240
11111000	248
11111100	252
11111110	254
11111111	255

Tabela 10.3. Máscaras possíveis em um octeto

Com o auxílio da tabela acima, é possível converter facilmente máscaras de binário para decimal. Vide um último exemplo abaixo:

```
Máscara em binário: 11111111.11111111.11100000.00000000
Bits reservados para máquinas: 13
Quantidade de endereços possíveis para máquinas: 2^13
Máscara convertida para decimal: 255.255.224.0
```

Conclusão: nem sempre as máscaras são 255 e 0; às vezes, elas comportam octetos com valores diferentes disto, pois podemos ter mais ou menos bits reservados para máquinas. Nos exercícios deste capítulo você poderá exercitar melhor isso.

10.5. Endereço de rede em binário

Definição 10.1. *O endereço de rede consiste de um número que, em binário, a parte do endereço que se refere às máquinas possui todos os bits com valor 0.*

A definição acima simplesmente diz o que será exemplificado abaixo:

```
Endereço de rede em decimal: 192.168.15.0
Máscara: 255.255.255.0
```

```
Endereço de rede em binário: 11000000.10101000.00001111.00000000
Máscara de rede em binário:  11111111.11111111.11111111.00000000
```

Olhando o endereço de rede e o a máscara, você vê que a parte do endereço de rede que se refere às máquinas sempre é zero em decimal. Olhando o mesmo endereço e a mesma máscara em binário, você pode observar que a parte onde os bits da máscara são 0 também são 0 no endereço de rede. Então, temos que: *no endereço de rede, os bits que se referem à parte de máquina são sempre 0.* Sempre.

Agora, vejamos um exemplo em que a máscara de rede não possui 255 em um dos octetos:

```
Endereço de rede em decimal: 122.14.184.0
Máscara: 255.255.248.0
Endereço de rede em binário: 11111010.00001110.10111000.00000000
Máscara de rede em binário:  11111111.11111111.11111000.00000000
```

Observou? Sempre, eu disse SEMPRE que a máscara em binário contiver o bit 0, o endereço de rede também o conterá. Como último exemplo, no exemplo abaixo descobrimos, a partir do endereço completo da máquina e da máscara, o endereço de rede.

Primeiro, os dados:

```
Endereço da máquina: 156.56.65.87
Máscara: 255.255.252.0
```

Agora, vamos converter ambos para binário:

```
Endereço de máquina: 10011100.00111000.01000001.01010111
Máscara em binário:  11111111.11111111.11111100.00000000
```

Até aqui, podemos observar que, na máscara, 10 bits são reservados para o endereçamento de máquina. Ou seja, temos capacidade para 2^{10} endereços, ou 1024. Para descobrir o endereço da rede, basta escrever 0 nos bits que se referem ao endereço de máquina. Ou seja, pegue o endereço de máquina em binário e escreva 0 nos 10 últimos bits. Depois, basta converter o resultado para decimal, e você terá o endereço de rede em decimal:

```
Endereço de rede em binário: 10011100.00111000.01000000.00000000
Endereço de rede em decimal: 156.56.64.0
```

Pronto, temos a reposta. A máquina 156.56.65.87 máscara 255.255.252.0 está contida na rede 156.56.64.0. Pode parecer estranho que o terceiro octeto seja diferente no endereço de máquina e de rede, mas é assim mesmo, não se preocupe.

10.6. Endereço de broadcast em binário

O endereço de broadcast sempre é o último endereço da rede.

Definição 10.2. *O endereço de broadcast consiste de um número que, em binário, a parte do endereço que se refere às máquinas possui todos os bits com valor 1.*

```
Endereço de rede em decimal: 192.168.15.0
Máscara: 255.255.255.0
Endereço de rede em binário: 11000000.10101000.00001111.00000000
Máscara de rede em binário:  11111111.11111111.11111111.00000000
End. broadcast em binário:   11000000.10101000.00001111.11111111
End. broadcast em decimal:   192.168.15.255
```

Viu? No endereço de broadcast em binário, todos os bits na parte reservada para as máquinas é 1.

Agora, vejamos um exemplo em que a máscara de rede não possui 255 em um dos octetos:

```
Endereço de rede em decimal: 122.14.184.0
Máscara: 255.255.248.0
Endereço de rede em binário: 11111010.00001110.10111000.00000000
Máscara de rede em binário:  11111111.11111111.11111000.00000000
End. broadcast em binário:   11111010.00001110.10111111.11111111
End. broadcast em decimal:   255.255.191.255
```

Pode parecer estranho o terceiro octeto do endereço de brodcast ser diferente o terceiro octeto do endereço de rede, mas não se preocupe: é assimo mesmo.

Agora, vamos descobrir o endereço de broadcast da rede a partir de um endereço de máquina. Vamos lá, passo por passo.

```
Endereço da máquina: 156.56.65.87
Máscara: 255.255.252.0
```

Precisamos converter isso para binário, a fim de ordenar o caos.

```
Endereço de máquina: 10011100.00111000.01000001.01010111
Máscara em binário:  11111111.11111111.11111100.00000000
```

Agora, para obter o endereço de rede, os dez últimos bits são 1 (pois a máscara indica que esta é a parte que representa a máquina). E aí converta isso em decimal e deixe alguém feliz hoje:

```
Endereço de broadcast em binário: 10011100.001110000.01000011.11111111
Endereço de broadcast em decimal: 156.56.67.255
```

Enfim, a resposta: a máquina de endereço 156.56.65.87 máscara 255.255.252.0 está contida em uma rede cujo endereço de broadcast é 156.56.67.255.

10.7. Descobrindo intervalos de endereços

Às vezes, é importante saber se a máquina está realmente naquela rede. Por exemplo, precisamos saber se a máquina cujo endereço é 10.13.3.51 máscara 255.255.254.0 está contida na rede 10.13.2.0. E agora? Sim ou não?

Primeiro, vamos descobrir o endereço de rede e de broadcast:

```
Endereço da máquina em decimal: 10.13.3.51
Máscara em decimal: 255.255.254.0
Endereço da máquina em binário: 00001010.00001101.00000011.00110011
Máscara em binário:             11111111.11111111.11111110.00000000
Endereço de rede em binário:    00001010.00001101.00000010.00000000
Endereço de broadc. em binário: 00001010.00001101.00000011.11111111
Endereço de rede em decimal:    10.13.2.0
Endereço de broadc. em decimal: 10.13.3.255
```

Ou seja, a resposta é sim. A máquina 10.13.3.51 máscara 255.255.254.0 está contida na rede 10.13.2.0, pois o intervalo de endereço nesta rede começa em 10.13.2.0 e vai até 10.13.3.255.

Vejamos um problema mais complexo. Será que as duas máquinas abaixo estão na mesma rede?

```
Endereço da máquina 1 em dec.: 155.10.44.8
Endereço de máquina 2 em dec.: 155.10.55.9
Máscara das máquinas em dec.:  255.255.192.0
```

Parece complexo, mas você tendo calma e fazendo por etapas, chegará ao resultado.

Primeiro, precisamos descobrir a rede de uma das máquinas, por exemplo, da máquina 1. Depois de descobrir a rede, devemos achar o intervalo de endereços desta rede, e ver se a máquina 2 está contida em tal intervalo.

```
Endereço da máquina 1 em dec.: 155.10.44.8
Máscara da máquina em decimal: 255.255.192.0
Endereço da máquina 1 em bin.: 10011011.00001010.00101100.00001000
Máscara da máquina em binário: 11111111.11111111.11000000.00000000
Endereço de rede em binário:   10011011.00001010.00000000.00000000
Endereço de broadc. em bin.:   10011011.00001010.00111111.11111111
Endereço de rede em decimal:   155.10.0.0
Endereço de broadc. em dec.:   155.10.63.255
Intervalo da rede: vai de 155.10.0.0 até 155.10.63.255
```

Bom, se o endereço da máquina 1 é 155.10.44.8 e o da máquina 2 é 155.10.55.9, então elas estão na mesma rede, como você pode concluir.

10.8. Lembrete sobre o número real de máquinas

O número real de máquinas que podem estar contidas em uma rede não é o mesmo que o número total de endereços possíveis. Você deve lembrar que os endereços de rede e de broadcast não podem ser atribuidos às máquinas. Assim, embora uma rede de máscara 255.255.255.128 (7 bits para a parte das máquinas) possua capacidade para 128 endereços, somente 126 desses endereços podem ser usados, pois o primeiro endereço é de rede, e o último, de broadcast. Da mesma forma, uma rede de máscara 255.255.254.0 (9 bits para a parte das máquinas) possui capacidade para 2^9 endereços, mas apenas $2^9 - 2$ podem ser atribuídos para máquinas.

Para os propósitos deste capítulo, considere, ao fazer os exercícios, o número **total** de endereços possíveis, sem subtrair os dois endereços que não podem ser usados.

10.9. Exercícios

Exercício 10.1. Dadas as máscaras em binário abaixo, converta-as para decimal e informe quantos endereços para máquinas são possíveis.

a) Máscara em binário: 11111111.00000000.00000000.00000000

Máscara em decimal: _____._____._____._____

Quantidade de endereços possíveis para máquinas: _____

b) Máscara em binário: 11111111.11111111.11111000.00000000

Máscara em decimal: _____._____._____._____

Quantidade de endereços possíveis para máquinas: _____

c) Máscara em binário: 11111111.11111110.00000000.00000000

Máscara em decimal: _____._____._____._____

Quantidade de endereços possíveis para máquinas: _____

d) Máscara em binário: 11111111.11111111.11111111.11000000

Máscara em decimal: _____._____._____._____

Quantidade de endereços possíveis para máquinas: _____

Exercício 10.2. Agoras, dadas as máscaras em decimal, informe quantos endereços para máquinas são possíveis (isso, SEM ESCREVER a máscara em binário).

a) Máscara: 255.255.0.0

Quantidade de endereços possíveis para máquinas: _____

b) Máscara: 255.255.255.192

Quantidade de endereços possíveis para máquinas: _____

c) Máscara: 255.255.254.0

Quantidade de endereços possíveis para máquinas: _____

d) 255.255.224.0

Quantidade de endereços possíveis para máquinas: _____

Exercício 10.3. Você está projetando uma rede para alguém. Esta pessoa pede a você uma rede em que sejam possíveis instalar 30 computadores. Qual é a máscara que **melhor** permite este número de máquinas?

a) 11111111.11111111.11111111.00000000

b) 11111111.11111111.00000000.00000000

c) 11111111.11111111.11111111.11100000

d) 11111111.11111111.11111111.11111000

Exercício 10.4. Agora, para uma rede que contenha 110 computadores, qual a máscara que **melhor** permite este número de máquinas? (responda em binário e também em decimal)

Exercício 10.5. Fulano tem uma rede com 17.000 máquinas. Qual(is) da(s) máscara(s) abaixo serviriam para esta rede?

a) 255.0.0.0

b) 255.192.0.0

c) 255.255.192.0

d) 255.255.255.0

Exercício 10.6. Ainda para a rede de 17.000 máquinas, qual é a máscara que **melhor** permite este número de máquinas? (responda apenas em decimal)

Exercício 10.7. Informados o endereço da máquina e a máscara em decimais, converta-os para binário e descubra o endereço de rede da mesma, tanto em decimal quanto em binário.

a) End. máq. dec.: 10.13.5.2
   Masc. dec. 255.128.0.0
   End. máq. bin.: _____._____._____._____
   Masc. bin.: _____._____._____._____
   End. rede bin.: _____._____._____._____
   End. rede dec.: _____._____._____._____

b) End. máq. dec.: 129.12.199.226
   Masc. dec. 255.255.224.0
   End. máq. bin.: _____._____._____._____
   Masc. bin.: _____._____._____._____
   End. rede bin.: _____._____._____._____
   End. rede dec.: _____._____._____._____

c) End. máq. dec.: 197.91.203.16
   Masc. dec. 255.255.255.254
   End. máq. bin.: _____._____._____._____
   Masc. bin.: _____._____._____._____
   End. rede bin.: _____._____._____._____
   End. rede dec.: _____._____._____._____

d) End. máq. dec.: 201.100.245.98
   Masc. dec. 255.255.255.192

End. máq. bin.: _____._____._____._____

Masc. bin.: _____._____._____._____

End. rede bin.: _____._____._____._____

End. rede dec.: _____._____._____._____

Exercício 10.8. Informados o endereço de máquina e a máscara de rede em decimais, descubra o endereço de broadcast, informando-os em decimais.

a) End. máq. dec.: 10.13.5.2

Masc. dec. 255.128.0.0

End. broad. bin: _____._____._____._____

End. broad. dec: _____._____._____._____

b) End. máq. dec.: 129.12.199.226

Masc. dec. 255.255.224.0

End. broad. bin: _____._____._____._____

End. broad. dec: _____._____._____._____

c) End. máq. dec.: 197.91.203.16

Masc. dec. 255.255.255.254

End. broad. bin: _____._____._____._____

End. broad. dec: _____._____._____._____

d) End. máq. dec.: 201.100.245.98

Masc. dec. 255.255.255.192

End. broad. bin: _____._____._____._____

End. broad. dec: _____._____._____._____

Exercício 10.9. Neste exercício, você deverá descobrir o intervalo dos endereços da rede. São informados os endereços de máquinas. (tudo está em decimal)

a) Endereço de máquina: 9.18.27.36

Máscara de rede: 255.128.0.0

Intervalo: de _____._____._____._____ até _____._____._____._____

b) Endereço de máquina: 18.27.36.45

Máscara de rede: 255.224.0.0

Intervalo: de _____._____._____._____ até _____._____._____._____

c) Endereço de máquina: 18.27.36.45

Máscara de rede: 255.255.240.0

Intervalo: de _____._____._____._____ até _____._____._____._____

d) Endereço de máquina: 200.100.50.25

Máscara de rede: 255.255.255.248

Intervalo: de _____._____._____._____ até _____._____._____._____

Exercício 10.10. Marque um X nas alternativas em que as duas máquinas apresentadas pertencem à mesma rede:

a) Máquina 1: 192.168.0.30; Máquina 2: 192.168.0.70; Máscara: 255.255.255.192

b) Máquina 1: 192.168.0.30; Máquina 2: 192.168.0.70; Máscara: 255.255.255.0

c) Máquina 1: 150.30.56.8; Máquina 2: 150.30.60.10; Máscara: 255.255.224.0

d) Máquina 1: 30.15.5.10; Máquina 2: 30.15.6.10; Máscara 255.255.254.0

Capítulo 11
Atribuição de Endereços IPv4

11.1. Introdução

Neste capítulo, você verá como são atribuidos os endereços IP hoje em dia globalmente, o que é endereçamento com classes, e qual a diferença entre endereços públicos e privados. Atentaremos também para a exaustão do número de IPs versão 4 disponíveis hoje em dia para atribuição, e como este problema é aliviado usando-se o NAT.

11.2. Atribuição de IPs na Internet

Você já deve ter observado que números de telefone próximos possuem o mesmo prefixo. Por exemplo, se Fulano e Cicrano são visinhos, o telefone de Fulano é 9991-XXXX e de Cicrano 9991-YYYY. Atenção, estes números são apenas exemplos hipotéticos; se eles existirem, não é de minha responsabilidade que eles passem trote e não quero ser processado!

Com os endereços IP é a mesma coisa. As pessoas não podem simplesmente escolher um número IP aleatoriamente e sair por aí navegando na internet... tem regulamento a parada. Na vida real, organizações regionais possuem bocos de IP que disponibilizam para pessoas daquela região. Assim. por exemplo, todos os IPs que começam com 202.XXX.XXX.XXX pertencem a um bloco, e nunca poderão ser encontrados em outro bloco. No mapa abaixo, você pode ver o nome dos RIRs (Registros Regionais de Internet) que controlam os IPs em suas respectivas regiões.

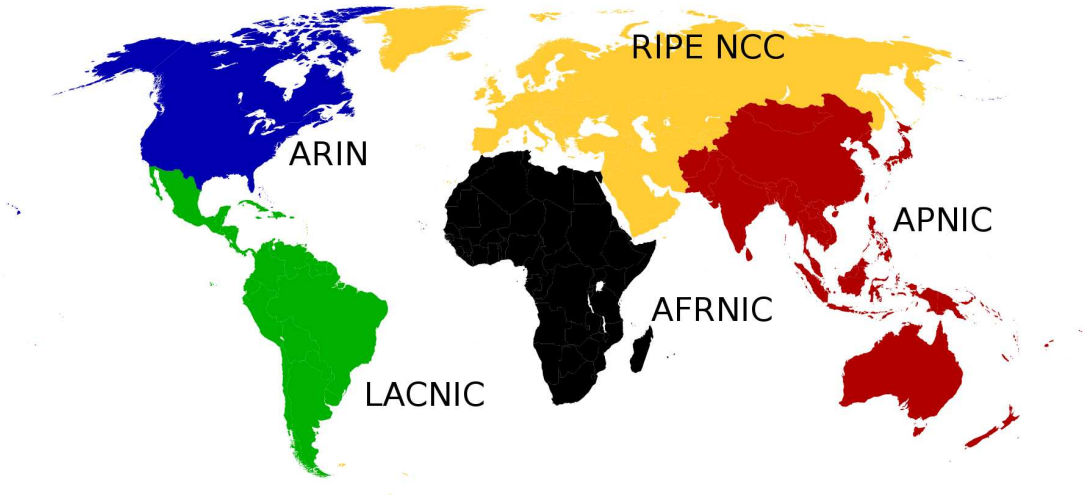

Figura 11.1. Registros Regionais de Internet (RIR - Regional Internet Registry) no mundo

Na América Latina e Caribe, a LACNIC é a responsável pela distribuição dos IPs. Observe que os nomes das RIRs são bastante legais. AFRNIC, por exemplo... poderia ser um nome de um filho de alguém, não acha?

Pois bem... sabendo que cada RIRs é responsável pelo endereçamento na sua área, como é feita a distribuição de IPs na internet mundial? Bom, existe uma organização chamada IANA que atribui blocos de endereços IP para cada RIR, conforme mostra a figura abaixo.

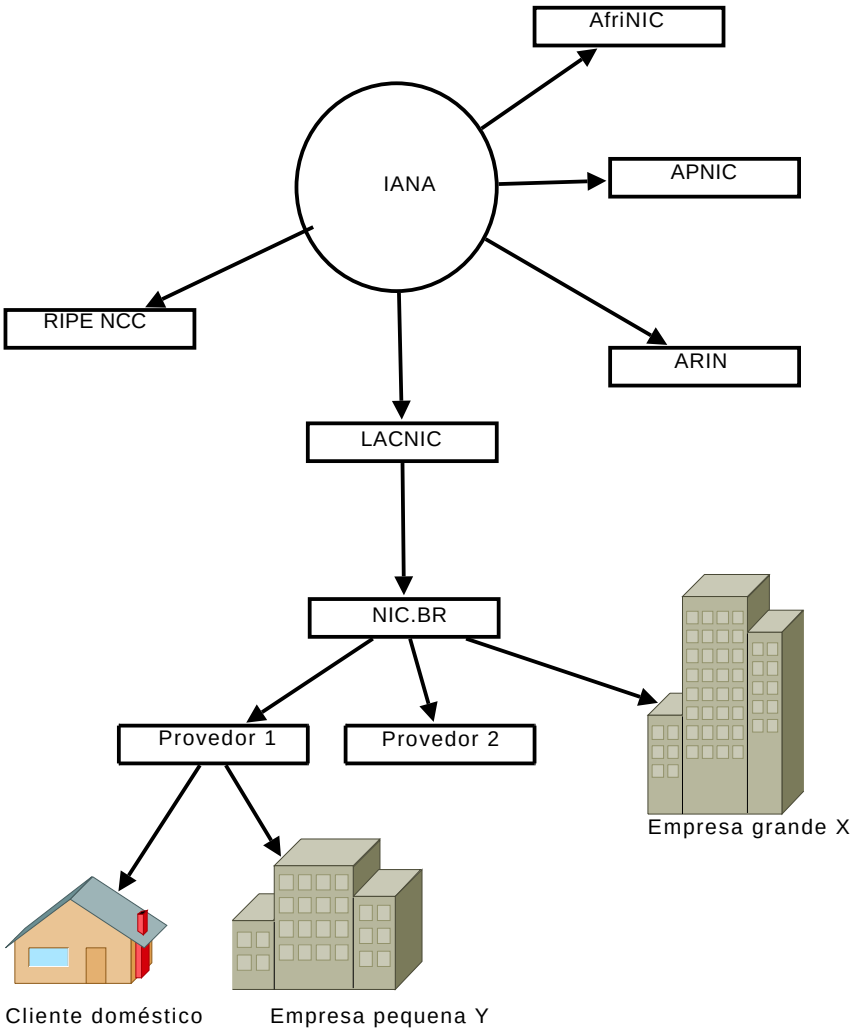

Figura 11.2. Atribuição hierárquica de IPs

Por exemplo, a IANA atribui o bloco 189.0.0.0 máscara 255.0.0.0 para a LACNIC, RIR da américa latina. Assim, a LACNIC dispõe do intervalo de IPs que vai de 189.0.0.0 até 189.255.255.255. Por sua vez, a LACNIC atribui ao NIC.BR, órgão responsável pela atribuição de IPs no Brasil, o bloco de IPs 189.40.0.0 máscara 255.255.0.0. Ou seja, a LACNIC pegou sua faixa de IPs e dividiu-as em redes menores, atribuindo para países diferentes da América Latina. O bloco 189 inteiro é da LACNIC, porém somente a rede menor 189.40 é do Brasil.

O NIC.BR, aqui no Brasil, pega sua rede 189.40.0.0, cujo intervalo vai de 189.40.0.0 até 189.40.255.255, e atribui uma rede menor ainda para um provedor ou uma empresa grande. Por exemplo, suponha que o endereços 189.40.84.0 máscara 255.255.252.0 seja atribuído para um provedor de acesso à internet. O provedor, de posse dessa rede cujo intervalo vai de 184.40.84.0 até 184.40.85.255, distribui para um cliente empresarial pequeno a rede 184.40.85.0 máscara 255.255.255.0, que é uma rede com capacidade para 256 endereços, dentre os quais 254 são atribuíveis às máquinas.

Lógico que isso é apenas um exemplo, pois provedores possuem um intervalo de IPs muito amplo, bem como o NIC.BR, com capacidade de atribuir redes para muitas redes grandes e provedores. O objetivo desta explicação é desenvolver a idéia de como os endereços são atribuídos: blocos de IPs cada vez menores são atribuídos, conforme descem na hierarquia.

Primeiro a IANA, depois o RIR, depois o país, os provedores, clientes, clientes dos clientes (sim, um cliente pode atribuir endereços para seus próprios clientes) e o ciclo pode continuar por muito tempo. Cada entidade faz o que bem entender com o grupo de endereços IPs que possui.

11.3. Endereçamento com classes

Não sabemos se isso foi um erro (muito provavelmente foi), mas funcionou por um bocado de tempo: o endereçamento com classes. No início da internet, o IETF, órgão responsável pelas normas, padrões e evolução da própria Internet, redes em geral e muitas outras coisas, não vendo que a internet poderia crescer tanto assim como hoje, criou o conceito de endereçamento com classes. Classe aqui não significa gente fina, mas sim o seguinte: redes de classe A são aquelas cujo primeiro octeto é reservado para rede. Classe B significa que o primeiro e o segundo octatos são reservados para rede. Classe C significa que o primeiro, o segundo e o terceiro octetos referem-se à rede. Em resumo: redes de classe A possuem máscara 255.0.0.0, classe B possui máscara 255.255.0.0 e classe C 255.255.255.0.

Até aí tudo bem. Agora, vem o manual de "como desperdiçar endereços IP". Essa norma de classes definiu que as redes 1.x.x.x até 127.x.x.x (ou seja, metade dos endereços IPs disponíveis) seriam redes de Classe A. "E daí?", você pergunta. E daí que essas redes foram dadas para empresas. Isso mesmo. Essas redes gigantescas, com capacidade para mais de seis milhões de IPs, foram dadas para redes que nunca teriam essa quantidade de máquinas. Por quê? Porque na época a quantidade de computadores conectados à internet era ínfima; ninguém tinha idéia de que 4 bilhões de endereços seriam usados; ninguém imaginava que dispositivos portáteis acessariam a internet; ninguém imaginava que uma única pessoa poderia ter três ou quatro dispositivos de conexão à internet, visto que naquela época o preço de um computador era a coisa mais absurda para uma pessoa comum. Por isso essas redes foram dadas para empresas.

Uma vez que essas redes de classe A foram dadas (ou vendidas, que seja) a essas empresas, não pode-se mais obteâ-la denovo.

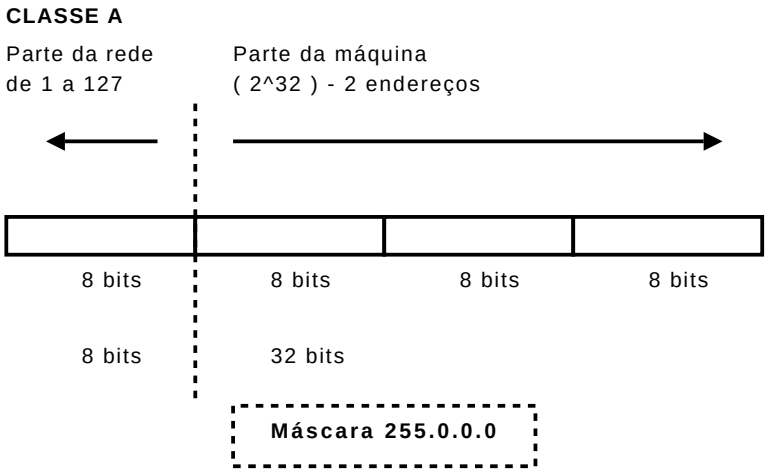

Figura 11.3. Classe A

A norma continua. As redes de classe B são aquelas cuja máscara é 255.255.0.0, e vai de 128.0.x.x até 191.255.x.x (16 mil redes, aproximadamente). Isso equivale a 1/4 dos IPs disponíveis, e cada rede tem capacidade para mais de 32 mil endereços. É difícil uma empresa chegar a isso, embora possível.

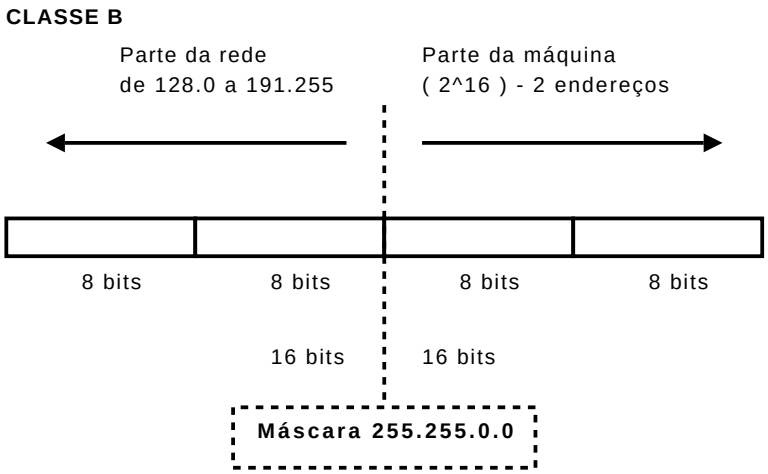

Figura 11.4. Classe B

Enfim, a classe C diz que suas redes possuem máscara 255.255.255.0. As redes de classe C vão de 192.0.0.x até 223.255.255.x. Isso equivale a 1/8 dos endereços IPs disponíveis, e cada rede de classe C possui capacidade para 254 máquinas (256 menos os endereços de rede e broadcast). Essas redes são as mais prováveis de existirem; contudo, como mostramos acima, apenas 1/8 do total de IPs estão nas duas milhões de redes classe C; metade dos quatro bilhões de endereços IP estão em apenas 127 redes de classe A! *Pouco mais de 100 empresas possuem redes de Classe A, e tais empresas nunca usarão todos os 16 milhões de endereços possíveis em suas redes*. Como você pode observar, há uma desigualdade gritante aí.

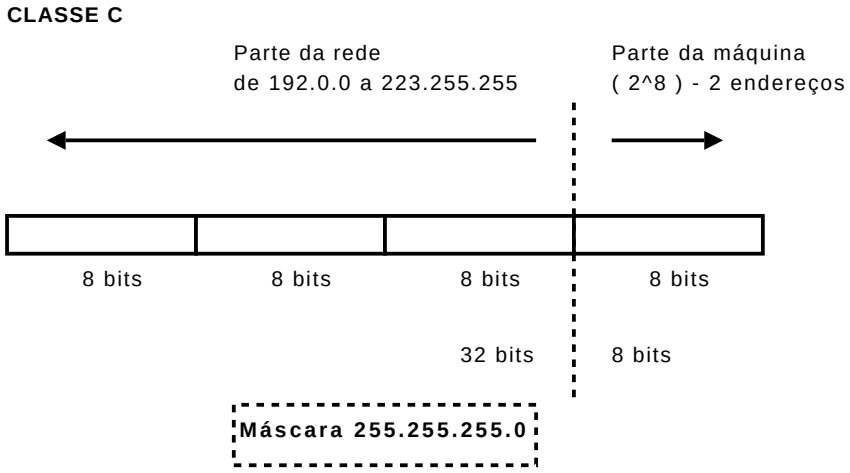

Figura 11.5. Classe C

NOTA 11.1. Neste capítulo, nas tabelas abaixo, indicamos os endereços das redes. Quando eventualmente nos referimos a redes hipotéticas da classe D ou E, usamos redes com a máscara 255.255.255.0.

Na tabela comparativa abaixo, você pode ver um resumo de tudo que falamos até agora sobre quantidade de redes e endereços atribuíveis a máquinas em cada rede (já subtraímos os endereços de rede e de broadcast).

Classe	Faixa	Número de Redes	Número de máquinas por rede
A	1.0.0.0 a 127.0.0.0	127	16.777.214
B	128.0.0.0 a 191.255.0.0	16.384	65.534
C	192.0.0.0 a 223.255.255.0	2.097.152	254

Tabela 11.1. Comparativo entre as classes

Nota 11.2. Neste livro, consideramos a rede 127.0.0.0 como uma rede de classe A, embora seja privada. Por isso a quantidade de redes na classe A apresentada aqui é 127, e não 126. Além disso o número de redes de classe B e C pode ser diferente de outras fontes, com duas redes a mais, pois presumimos que o não-uso da sub-rede zero é coisa do passado.

Pois é. A Classe A ocupa 50% dos endereços; a classe B, 25% e a classe C 12,5%. A soma disso é 87,5%. O que aconteceu com os outros 12,5% de endereços disponíveis no mundo? Bom, o IETF definiu ainda duas classes de endereços, que não poderão ser atribuídos à rede, porém possuem suas utilidades. São as classes D e E. A Classe D é reservada para endereços multicast, que estudaremos posteriormente neste curso. A classe E foi reservada para uso futuro, mas é usada para testes hoje em dia.

Classe	**Faixa**	**Utilidade**
D	224.0.0.0 a 239.255.255.0	Multicast
E	240.0.0.0 a 247.255.255.0	Reservado para uso futuro

Tabela 11.2. Classes D e E

Agora, atente para o seguinte fato: nem todas as redes podem ser usadas. "Claro, pois elas pertencem à empresas!". Não, não é isso. O IETF definiu algumas redes que não poderiam ser usadas por ninguém, pois tem finalidade de rede privada. Explicaremos este conceito adiante.

11.4. Endereços Privados

Logo notou-se que, com a velocidade que a Internet crescia, logo o mundo ficaria sem endereços IP. Por isso, foram criadas faixas de endereços que não seriam, observe bem, **não seriam** roteadas na internet. São endereços que podem ser usados apenas em empresas. Os roteadores da internet não encaminhariam pacotes destinados a eles. Chamamo-os de endereços privados.

Definição 11.3. *Endereço privado: é o endereço IP versão 4 que não é roteado na internet, apenas em redes no âmbito de uma mesma companhia.*

O escopo dos endereços privados é local a uma empresa; assim, ao contrário do que acontece com endereços públicos, os endereços privados podem ser usados por várias companhias diferentes, com repetição - só não podem ser repetidos, é claro, dentro de uma mesma companhia, em tese.

A tabela abaixo mostra onde estão esses endereços.

Classe	**Faixa**
A	10.0.0.0 (uma rede)
B	172.16.0.0 a 172.31.0.0 (32 redes)
C	192.168.0.0 a 192.168.255.0 (256 redes)

Tabela 11.3. Endereços privados

Como usá-los? Bom, você pode usá-los como quiser, pois não precisa pedir permissão. São endereços livres da necessidade de coordenação por algum órgão superior. A LACNIC não vai processá-lo, você não terá que solicitar um intervalo de endereços ao NIC.br e ninguém vai morrer por causa disso. Nenhuma outra empresa que estiver usando, dentro dela, o intervalo 10.0.0.0 vai ter dificuldade se você usar o mesmo intervalo. Lembre-se sempre de que endereços privados são válidos apenas na companhia local: ou seja, se você for um milionário com uma empresa enorme, pode usar o intervalo 10.0.0.0 para endereçar suas máquinas internamente.

Um fato interessante é que, teoricamente, não é possível acessar a internet (que usa endereços públicos) a partir de uma máquina que usa endereços privados. Isso acontece porque quando a máquina de endereço privado envia um pacote para uma máquina na internet, o pacote chega à máquina de destino (pois seu IP é público); porém, quando a máquina envia uma resposta para quem solicitou informação, o pacote não chega nesta máquina, pois o IP da mesma é privado e na internet (observe como estamos repetindo este fato) os roteadores não encaminham pacotes de endereços privados.

Para esclarecimentos, observe a figura abaixo.

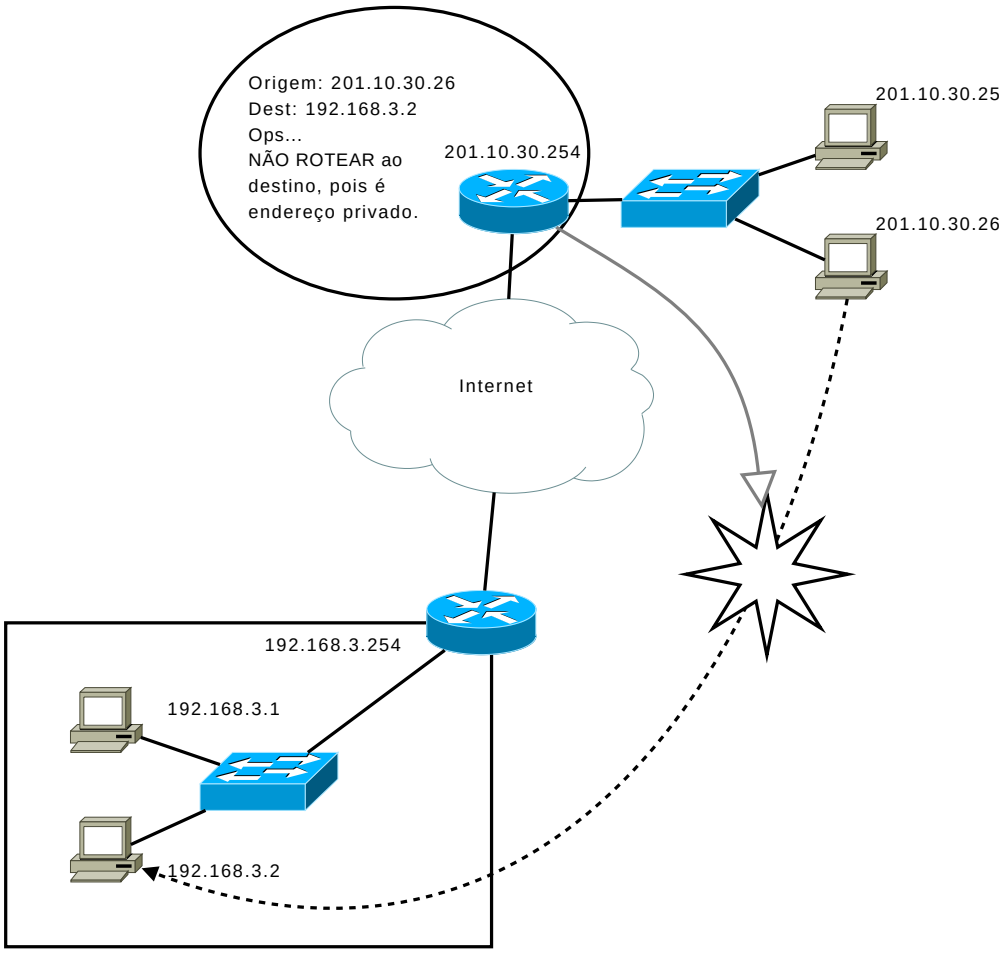

Figura 11.6. Falha ao enviar para máquina em rede privada

Primeiro, observe que na parte superior da figura, a máquina cujo IP é 201.10.30.26 tenta, mas somente tenta essa infeliz, entrar em contato com uma máquina da rede local na parte inferior da figura, cujo IP é 192.168.3.2. A máquina 201.10.30.26 está na internet, como

pode-se concluir a partir de seu IP - ele é público. Já a máquina na parte inferior da figura está em alguma rede local. Isso não seria problema, pois ela poderia estar na rede local e mesmo assim ter um endereço de IP público; contudo, além de estar na rede local, ela possui um endereço de IP privado.

Então, quando a máquina com IP público envia um pacote para o roteador que está ligado à internet... adivinha o que acontece. O presidente dos EUA tem diarréia? Não. O presidente do Brasil tem uma diarréia? Não! Não há diarréias. "Ah, já sei! Alguém toma um remédio anti-diarréia".

Bom, o que acontece na verdade é que o roteador descarta o pacote. Isso mesmo. Já era. Perdeu preibói. E isso acontece porque (olha a lavagem cerebral) roteadores da internet, ou até mesmo conectados a ela, não encaminham pacotes cujo IP de destino é privado. Se depois de tanta insistência nisso alguém ainda persistir em teimar...

"Ué... mas alguém já me disse que tem como uma máquina com IP privado se conectar à internet". Sim, tem, mas isso não é roteamento. Isso definitivamente não tem nada haver com roteamento, basta! Basta! Pá!

Isso é um recurso chamado tradução de endereços. Esse recurso permite, de forma gambiarrática mas incrivemente funcional, que máquinas privadas falem com máquinas na internet; para isso, o IP privado é transformado em um IP público. Ou seja, máquinas com IP privados ainda continuam sem poder falar com máquinas na internet, mas seu IP privado é traduzido pelo Gateway padrão. Estudaremos isso em breve ainda neste capítulo (na seção de NAT e PAT), mas não esqueça:

Tradução não é roteamento. IPs públicos não conseguem falar com IPs privados, porque os roteadores na internet não encaminham. Contudo, com a tradução de endereços, um IP privado é transformado em um IP público, e aí sim, a comunicação é possível.

11.5. Exaustão dos Endereços IPv4

Que os enderços IPv4 disponíveis vão acabar não é surpresa. Contudo, o fim está mais próximo do que se pensa. Por isso foram criados os endereços privados. Sem eles, cada uma das máquinas do planeta terra precisaria ter um IP público. Graças ao endereço privado, cada empresa, de centenas de computadores, precisa ter apenas um endereço público, e internamente, usar endereços privados.

Penso que o ideal seria você começar a se preparar para a mudança com o IPv6. Use-o na sua casa, na sua rede caseira interna, na sua empresa, no seu país comunista, se for ditador. Assim, quando ele for implantado definitivamente no mundo, você não sofrerá um colapso do coração (e no caso do ditador o povo o aclamará como um herói, ou não).

11.6. NAT

O NAT, Tradução de Endereços de Rede (Network Address Translation) é um recurso da arquitetura TCP/IP introduzido na RFC 1631, datada de 1994. É implementado na camada rede da arquitetura, funcionando em sistemas operacionais que obedecem a essa RFC. O NAT é comumente usada no Gateway padrão da rede, que pode ser um computador com Linux, por exemplo, ou um roteador. Hoje em dia, NAT e PAT (estudado adiante) são referidos como NAT apenas; porém você verá que os dois termos referem-se a coisas diferentes. NAT é tradução de endereços; PAT é tradução de portas.

Primeiro, a tradução de endereços. Observe a figura abaixo:

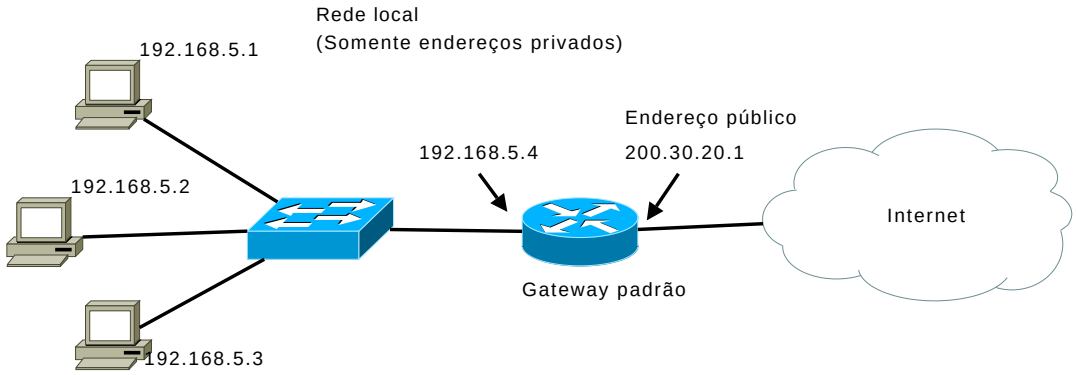

Figura 11.7. Gateway padrão usando endereço público na porta WAN.

Na rede local da figura, temos três hospedeiros interligados por um comutador. O comutador é ligado à porta LAN do roteador (Gateway padrão), cujo IP é 192.168.5.4. Observe que este IP é privado, ou seja, não é roteável pela Internet. O Gateway possui uma porta WAN, cujo IP é público: 200.30.20.1.

Como uma máquina da rede local pode falar com uma máquina na internet? Eles não possuem IP para isso. Bom, a tradução de endereços serve para isso: a porta WAN do Gateway possui um IP público, certo? Pois bem. Quando uma máquina da LAN quiser falar com alguém na internet, esta máquina usa o IP público do Gateway padrão. Isso mesmo! Observe a figura abaixo.

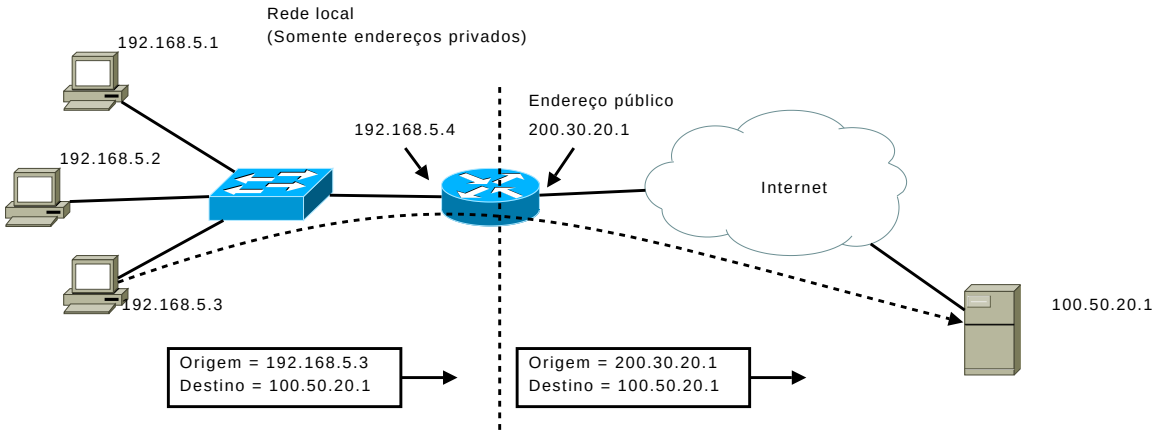

Figura 11.8. Tradução de IP privado para IP público.

A máquina cujo IP é 192.168.5.3 (rede local) quer falar com o servidor 100.50.20.1, certo? Então, um pacote que sai da rede local em direção à internet (linha pontilhada na figura) passa por uma tabela de tradução NAT no Gateway:

Endereço origem	Origem traduzida
192.168.5.3	200.30.20.1
100.50.20.1	Não precisa

Tabela 11.4. Lógica do NAT no Gateway padrão.

Ou seja, “quando o endereço de origem for 192.168.5.3, traduza-o para 200.30.20.1 (endereço da porta WAN do Gateway). Este será o novo endereço de origem”. Assim, quando o servidor na internet receber o pacote, ele o receberá de 200.30.20.1, que é roteável (por ser público), e enviará uma resposta para ele. O pacote de resposta chegará, obviamente, à porta WAN do Gateway, que traduzirá o endereço de destino para 192.168.5.3, entregando-o à máquina local de endereço privado.

Você pode estar pensando: bom, só temos um endereço público para a rede inteira. O que acontece se várias máquinas quiserem acessar a internet? Pela lógica, não é possível duas máquinas usarem um único endereço público através do NAT. Por isso, temos o PAT.

11.7. PAT

PAT significa Tradução de Endereços de Portas (Port Address Translation). Ele expande as possibilidades do NAT. Com ele, é possível várias máquinas com endereços IP privados acessarem a internet, por meio de um único endereço público - a saber, o endereço público da porta WAN do Gateway padrão.

Para conseguir isso, o PAT associa (a) um endereço privado e (b) uma porta à (c) um endereço público e (d) à uma porta, conforme tabela abaixo.

End. de máq. na Lan	Porta da máq. na Lan		End. público	Porta
192.168.5.1	2033		200.30.20.1	2050
192.168.5.2	1988		200.30.20.1	2051
192.168.5.3	2033		200.30.20.1	2052

Tabela 11.5. Tabela PAT no Gateway padrão

As coisas parecem confusas? Se parecem, é porque precisamos relembrar o que é uma porta. Porta é um conceito de camada transporte da arquitetura TCP/IP, camada imediatamente acima da “rede” a abaixo da “aplicação”. A camada rede trata do endereçamento lógico do hospedeiro, enquanto a camada transporte trata da identificação da aplicação TCP/IP cliente ou servidora. Veja a figura abaixo:

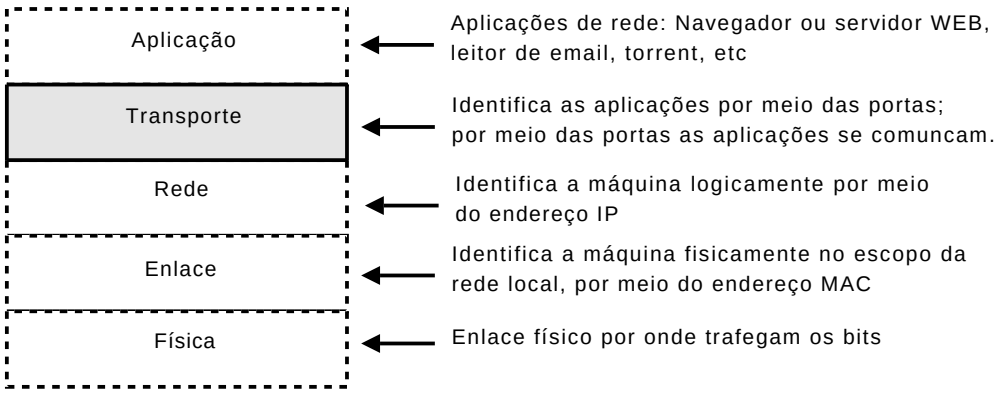

Figura 11.9. Resumo da arquitetura TCP/IP.

A tradução de endereços NAT funciona unicamente na camada Rede, pois traduz um endereço lógico para outro. Já a tradução de portas PAT atua nas em duas camadas: Transporte e Rede. Você deve se lembrar do que estudamos no capítulo 2, seção 2.7 ("Camada transporte"). Se não lembra, leia.

O meio pelo qual as aplicações se identificam é através dos números das portas. Por exemplo, o cliente WEB na máquina abre uma conexão WEB com um servidor em outra máquina. Assim, o cliente pode abrir a porta 2055, por exemplo, na camada transporte da máquina. E o servidor WEB mantém a porta 80 aberta para receber conexões. Se o mesmo cliente abre outra aplicação, esta aplicação irá abrir uma nova porta, 2056, por exemplo. E os servidores sabem que devem responder aos clientes com o IP de destino do cliente, e porta de destino equivalente àquela que o cliente abriu. Para entender melhor, analise a figura abaixo:

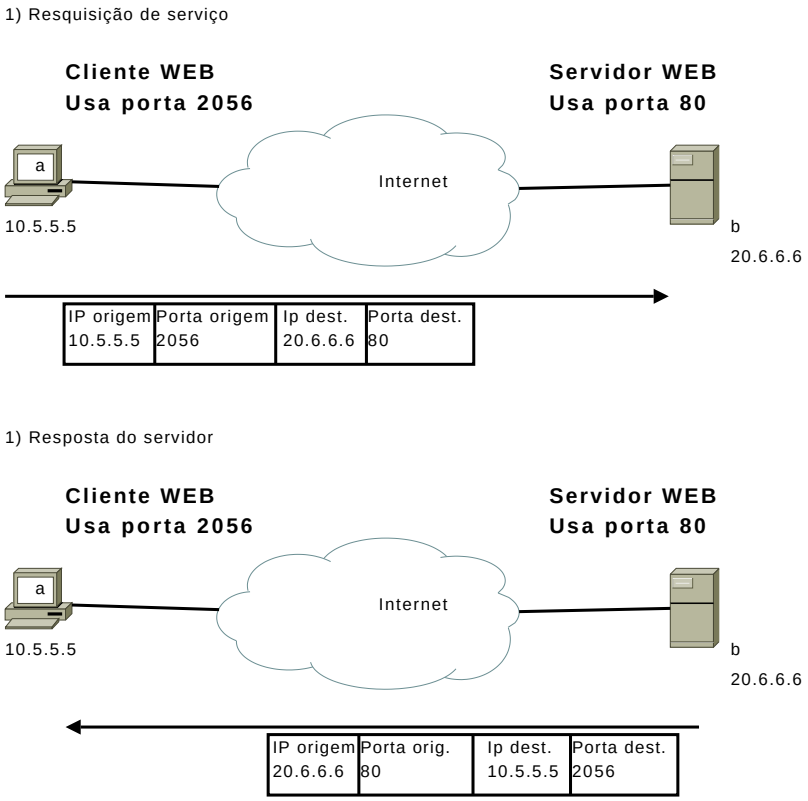

Figura 11.10. Requisição e resposta: as aplicações usam portas para identificar-se.

No passo 1, o cliente conecta-se ao servidor. A porta que a aplicação web usa para identificar-se é 2056. Poderia ser qualquer outra porta que não estivesse sendo usada. É o sistema operacional da máquina a que escolhe qual porta associar a essa aplicação. O usuário digita um endereço no cliente web, e o cliente conecta-se ao servidor na porta 80 (pois convencionou-se que servidores web sempre usam a porta 80 por padrão para serviços web).

No passo 2, quando o servidor recebe a requisição, ele responde para a máquina que requisitou, e para a aplicação que requisitou, usando, logicamente, o endereço da máquina `a` como destino e a porta que identifica a aplicação cliente, que é 2056 neste exemplo.

NOTA 11.4. Lembre-se que, além do número da porta, a aplicação deve escolher por qual dos protocolos da camada transporte deve transmitir os dados: TCP ou UDP.

Se o usuário da máquina `a` abrisse outro cliente web chamando o mesmo servidor, os dados entre as duas aplicações abertas não seriam trocados, pois embora o endereço ip de origem seja o mesmo, o número das portas são diferentes, e o servidor sempre faz distinção. Ou seja, se a máquina `a` tivesse outra aplicação, usando a porta 2057, e se conectasse ao mesmo servidor, o servidor teria ainda duas conexões, conforme tabela abaixo. Seriam conexões distintas, com dados distintos, sem mistura.

Nº da conexão	Ip origem	Porta origem	Ip destino	Porta destino
1	10.5.5.5	2056	20.6.6.6	80
2	10.5.5.5	2057	20.6.6.6	80

Tabela 11.6. Duas conexões partindo de uma mesma máqina

Em suma, o que identifica, do lado servidor, uma conexão, é a dupla "Ip origem" e "Porta de origem". Assim é possível que uma única máquina possua diversas conexões com o servidor. Aproveitando-se desse fato, a tradução de portas possibilita que o Gateway padrão da rede local se conecte *em nome* das outras máquinas da rede.

Analise a figura abaixo:

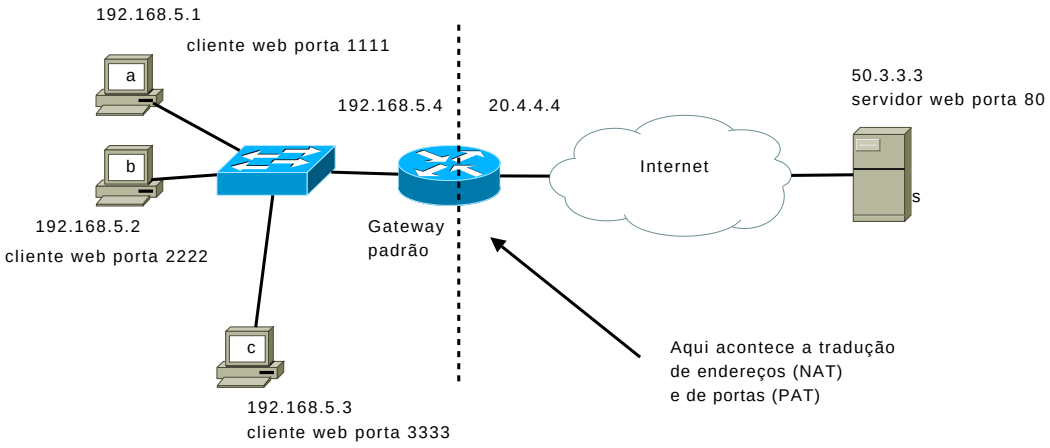

Figura 11.11. Esquema NAT/PAT.

Temos, na rede local, três máquinas, todas elas conectando-se ao servidor `s`. Contudo, essas máquinas precisam ter o endereço IP traduzido, visto que seus atuais endereços IPs são privados. Qual será o novo endereço delas após a tradução? Será 20.4.4.4, pois este é o endereço IP WAN do Gateway. Pois bem, essas máquinas possuem uma aplicação web rodando nas portas indicadas na figura, e o servidor `s` tem um programa servidor web rodando na porta 80, como é natural que aconteça.

Quando a aplicação cliente da máquina `a`, por exemplo, quer falar com o servidor `s`, o pacote contém as seguintes informações:

```
Endereço de origem: 192.168.5.1

Porta de origem: 1111

Endereço de destino: 50.3.3.3

Porta de destino: 80
```

Todavia, quando este pacote passa pelo Gateway, ocorre a tradução de endereço e também de porta. A tabela abaixo demonstra como está a lógica do Gateway:

Nº da conexão	Ip orig.	Port. orig.	Ip traduzido	Porta traduzida
1	192.168.5.1	1111	20.4.4.4	2000
2	192.168.5.2	2222	20.4.4.4	2001
3	192.168.5.3	3333	20.4.4.4	2002

Tabela 11.7. Lógica PAT/NAT do Gateway

Como você pode observar, todas as três máquinas usam um único IP público. Se não existisse a tradução de portas PAT, mas somente a tradução de endereços, seria imposivel que as três máquinas falassem com o servidor ao mesmo tempo. Hoje em dia, usamos sempr o NAT junto com o PAT. Não é comum usar apenas o NAT, pois isso impossibilitra muitas conexões concorresntes, como também não é comum usar apenas o PAT.

Os redistas contemporâneos chamam o conjunto NAT e PAT de NAT. Assim, quando alguém fala NAT, muito provavelmente está referindo-se às duas traduções.

11.8. Conclusão

Este capítulo foi o terceiro, de um total de quatro, que fala sobre o protocolo IP versão 4. Nele, você viu como Registros Regionais de Internet (RIRs) distribuem o espaço de endereços pelo mundo. Viu que as entidades em cada país alocam blocos de IPs a provedores de serviço ou a grande empresas, que por sua vez, alocam blocos menores para outras entidades ou pessoas, e assim sucessivamente.

Abrimos um parêntese para falar sobre o endereçamento com classes, que divide o bloco de IPs em basicamente três classes: classe A, cujo primeiro octeto pertence à rede; classe B, cujos dois primeiros octetos pertencem à rede; e a classe C, cujos três primeiros octetos pertencem à rede. Vimos também o grande desperdício causado por isso, pois no começo da internet, grandes blocos foram designados a empresas, ficando definitivamente ocupados. A próxima internet, que funcionará sobre o protocolo IPv6 (estudaremos adiante neste curso), não divide (ainda bem!) o total de endereços em classes.

Uma vez tendo um IP, ou ainda, um bloco de IPs designado, uma pessoa ou uma companhia deve decidir como escalonar o espaço de IPs disponíveis em sua empresa. Ele o responsável pela rede na companhia receber um bloco de endereços IP de máscara 255.255.255.0, ele possui cerca de 254 endereços (isto é, 256 menos o endereço de rede e o endereço de broadcast) para alocar em suas máquinas, e todos eles são endereços públicos, isto é, endereços roteáveis na internet.

Caso o redista tenha recebido um único endereço de IP, entretanto possua mais do que uma máquina em sua casa ou escritório, deve usar, dentro da rede local, endereços privados, isto é, não roteáveis na internet. Com isso, o redista pode alocar endereços às suas máquinas sem medo de uma catástrofe universal.

Se essas máquinas, cujo endereço de rede é privado, quiserem falar na internet, é preciso que haja uma tradução de endereços. Isso é feito pelo NAT. O NAT permite que os pacotes das máquinas na LAN tenham o endereço traduzido, de privado para público e vice-versa quando a resposta vier. Para casos mais reais em que muitas máquinas com endereço privado falem na internet, além no NAT precisamos do PAT, tradução de portas. Hoje em dia, quando alguém fala NAT, geralmente está se referindo aos dois, NAT e PAT.

No próximo capítulo, analisaremos um pacote IP e como é feito o roteamento do mesmo.

11.9. Exercícios

Exercício 11.1. Em uma cidade existem duas companhias com 100 máquinas cada. Observa-se que, em uma das máquinas da companhia A, o IP é 15.30.2.5. Já em uma máquina da companhia B, o IP é 15.30.20.10. Os dois primeiros octetos são iguais. Responda com suas palavras o porquê dessa semelhança.

Exercício 11.2. O que é um Registro Regional de Internet e qual sua função?

Exercício 11.3. Um grande provedor de internet aloca um bloco a um pequeno provedor regional. O bloco é: 10.20.0.0 máscara 255.255.0.0. Quantos clientes este pequeno servidor regional pode ter, se cada cliente exigir um endereço de máscara 255.255.255.0?

Exercício 11.4. E quantos clientes o pequeno servidor regional da questão anterior pode ter, se os mesmos alocarem um endereço IP público cada?

a) O mesmo que 2^8

b) O mesmo que 2^{16}

c) O mesmo que 2^{24}

d) O provedor não pode fazer isso

Exercício 11.5. Qual atributo abaixo diferencia um endereço IP público de um privado?

a) Endereços públicos podem ter diferentes máscaras de rede; endereços privados devem ter apenas máscara 255.255.255.0.

b) Endereços públicos não podem ser roteados; endereços privados podem.

c) Endereços públicos podem ser roteador na internet; já endereços privados não podem ser roteados na internet, apenas em roteadores de uma mesma companhia (rede de campus).

d) Endereços públicos são gratuitos; endereços privados são pagos.

Exercício 11.6. O que é endereçamento com classes? Qual a desvantagem do mesmo?

Exercício 11.7. Das situações abaixo, quando é necessário usar o NAT?

a) Quando uma máquina na LAN cujo IP é público quer falar com um servidor na Internet.

b) Quando uma máquina na LAN cujo IP é privado quer falar com um servidor na Internet.

c) Quando duas máquinas na LAN cujo IP é público querem se falar.

d) Quando duas máquinas na LAN cujo IP é privado querem se falar.

Exercício 11.8. Das situações abaixo, quando é necessário usar o PAT?

a) Quando uma única máquina na LAN de IP é privado quer falar com um servidor na Internet.

b) Quando duas máquinas na LAN de IP privado querem falar simultaneamente com um servidor na Internet.

c) Quando duas máquinas na LAN de IP privado querem falar entre si.

d) Quando uma única máquina na LAN cujo IP é público, e que é ligada ao Gateway padrão da LAN, quer falar com um servidor na Internet.

Exercício 11.9. Embora, hoje em dia, os redistas falem NAT para se referir a duas coisas, NAT e PAT, qual a verdadeira diferença entre elas?

Exercício 11.10. De qual(is) cadama(s) da arquitetura TCP/IP o NAT tira funcionalidades?

a) Aplicação

b) Transporte

c) Rede

d) Enlace

e) Fìsica

Exercício 11.11. E o PAT?

a) Aplicação

b) Transporte

c) Rede

d) Enlace

e) Física

Capítulo 12

Roteamento IPv4

12.1. Introdução

Agora vamos falar de roteamento. Começaremos este capítulo com uma descrição de um pacote IPv4, passaremos ao funcionamento das tabelas dos roteadores e como essas tabelas são atualizadas, e por fim, discutiremos sobre a fragmentação do IPv4 e porque isso foi abandonado no IPv6.

12.2. Montando um pacote

A figura abaixo mostra o conteúdo de um pacote IPv4, bem como o tamanho, em bits, dos mesmos.

bit offset	0–3	4–7	8–15	16–18	19–31
0	Version	Header length	Differentiated Services	Total Length	
32	Identification			Flags	Fragment Offset
64	Time to Live		Protocol	Header Checksum	
96	Source Address				
128	Destination Address				
160	Options				
160 or 192+	Data				

Figura 12.1. Cabeçalho do IPv4; retirado de http://en.wikipedia.org/wiki/IPv4.

O primeiro campo é o **version**, pois o roteador precisa saber se deve trabalhar com o pacote como IPv4 ou IPv6. Em seguida, temos o campo **Header lengh**, largura do cabeçalho. Isso acontece porque o tamanho do cabeçalho IPv4 pode variar - e acredite, para um roteador, isso não é nada bom. O cabeçalho IP tem a capacidade de colocar um maior ou menor número de opções, alterando seu tamanho.

O campo **Differentiaed Services** mudou muito ao longo do tempo. Sua intenção inicial[12.1] era prover um meio de identificação do tipo de serviço contido no pacote. Para quê? Ora, para que os roteadores beneficiassem pacotes com tipos de serviços prioritários. Ou seja, se um pacote de uma aplicação essencial para o funcionamento da rede precisasse passar por um roteador congestionado, com pacotes na entrada e na saída, o roteador daria prioridade a esse pacote, permitindo-o furar a fila.

A seguir temos o campo **Total Length**, que informa o tamanho total do pacote, incluindo o cabeçalho. Isso é útil para determinar onde o pacote termia.

12.1. [RFC 791] pág. 29.

O campo **Identification** serve para dizer que este pacote é, na verdade, um pedaço de um pacotão que foi anteriormente desfragmentado (ou despedaçado). Agora para tudo! Precisamos explicar isso com calma.

Os roteadores na internet possuem diferentes configurações. Uma das coisas que se configura em um roteador é o tamanho máximo do pacote que passará por uma determinada porta. Considere a figura abaixo:

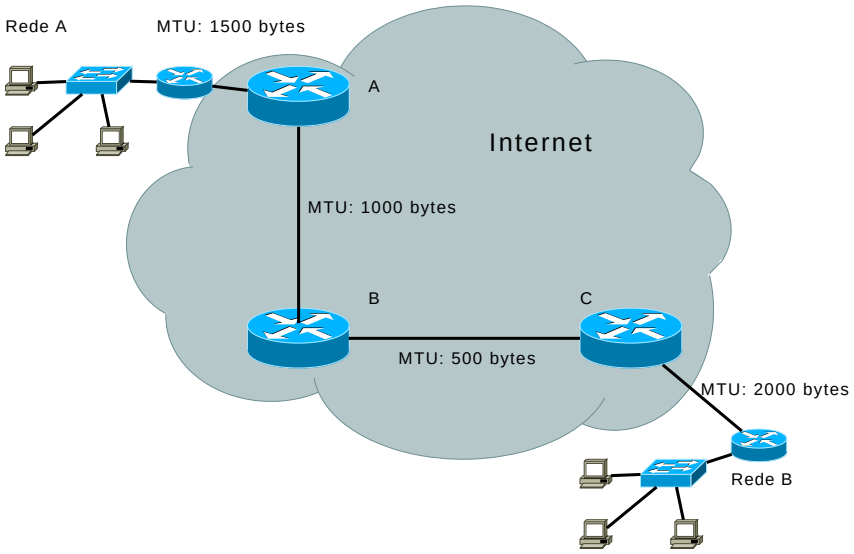

Figura 12.2. Pacote trafegando pela internet.

Temos duas redes locais (Rede A e Rede B) interconectadas através da internet, representada pela nuvem cinzenta. Os rotadores dentro desta nuvens pertencem aos provedores de acesso. Os detalhes de endereçamento IP não importam para esta discussão. Observe que coloquei as palavras MTU seguidas de um número de bytes em cada enlace que liga roteadores a roteadores ou roteadores a LANs. MTU significa Tamanho Máximo de Transmissão (Maximum Transmission Unit), isto é, o tamanho máximo que o pacote IPv4 pode ter naquele enlace.

Você deve se esforçar para perceber a importância do que direi agora: é melhor um pacotão do que muitos pacotinhos, pois cada pacote desperdiça espaço com o cabeçalho. Ou seja, se tenho 3000bytes para transmitir, e uso um único pacotão para isso, meus dados da aplicação serão postos em um cabeçalho IPv4, que normalmente tem 20bytes, e o tamanho total do pacote será 3020bytes. Ou seja, pouco desperdício.

De outra forma, se para transmitir meus 3000bytes de dados usam-se pacotinhos de 300 bytes, cada um desses terá 20bytes de cabeçalho IPv4. Como serão precisos 10 pacotinhos para transmitir toda a informação, serão gatos 200bytes, e o número de bytes percorrido na rede será de 3200.

Porém, existem enlaces que não são capazes de transmitir pacotões. Você precisa saber que a internet é imensa, tem muitos tipos de rede, e cada enlace que interliga duas redes pela WAN pode ter um MTU diferente, como no caso da figura mostrada para explicar o que será dito adiante.

Suponha que na figura, uma máquina da *Rede A* tem um pacote de 1500bytes para transmitir (este é o tamanho total do pacote contido no campo Total Length). O MTU do enlace aceita este valor, então tudo bem, o pacote é enviado inteiramente para o roteador A da internet.

O enlace que liga o roteador A ao roteador B tem apenas 1000bytes. Tudo bem, o pacote é dividido em dois - um de 1000 e outro de 520bytes (20bytes do novo cabeçalho), por exemplo - e enviado para o roteador B. Contudo, esses pacotes não são distintos entre si: eles precisam ser remontados em algum momento do percurso, e quando isso acontecer é preciso saber que os dois pacotes pertencem a uma mesma sequência. Para isso serve o campo **Identification**.

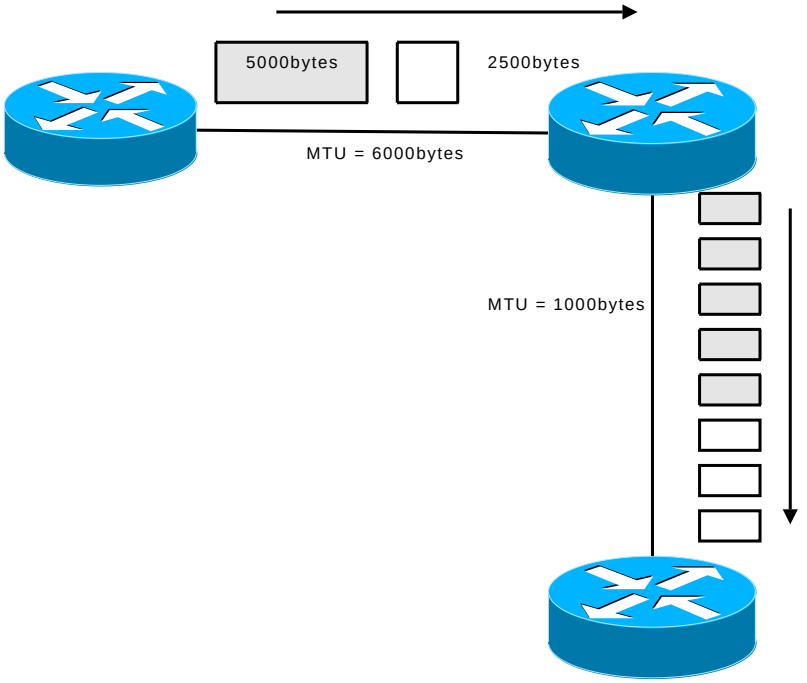

Figura 12.3. Segmentação e identificação.

Na figura acima, o primeiro roteador envia dois pacotes ao segundo: um pacote cinza, de 5000bytes, e um pacote branco, de 2500bytes. Pois bem, o enlace que liga o segundo ao último roteador tem MTU = 1000bytes, e isso significa que haverá segmentação dos pacotes grandes em pacotes menores. Observe que na ilustração, os pacotes menores seguem o mesmo esquema de cores, e você percebe que os pacotinhos brancos pertencem ao pacotão branco etc. Você teve essa percepção através das cores. Um roteador também tem uma percepção, baseada no conteúdo do campo Identification, para saber a qual pacotão pertence um pacote menor. Mais adiante entraremos em detalhes sobre isso.

O campo **flags** contém códigos que indicam algumas opções, como, por exemplo, "não fragmentar esse pacote".

O campo **Fragment Offset**, ou simplesmente **offset**, indica qual a posição do fragmento, para que, quando o pacote for remontado, os fragmentos sejam inseridos corretamente nas posições de origem. Isso é ilustrado na figura abaixo.

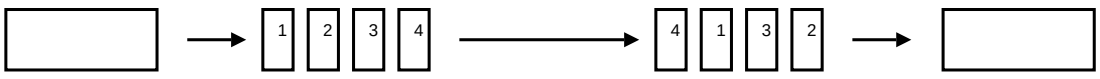

Figura 12.4. Função do campo offset.

Sem o campo Offset, seria impossível ordenar os pacotes fragmentados.

O campo **Time to Live** conta o tempo de vida do pacote. Cada vez que o pacote passa por um roteador, esse número é diminuído. Se chegar a zero, o roteador descarta o pacote e envia uma mensagem para o hospedeiro remetente informando o descarte do pacote. Isso é útil para que não hajam pacotes trafegando eternamente pela internet. A figura abaixo ilustra o funcionamento desse campo.

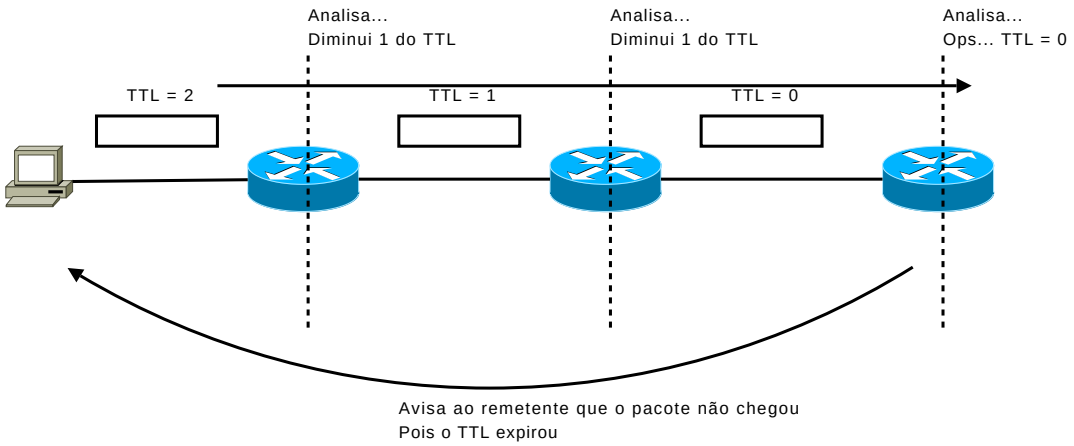

Figura 12.5. Time To Live.

O campo **Protocol** indica qual o protocolo de camada superior (isto é, camada transporte) que está contido no pacote IP. Isso é útil? Sim. Quando a máquina destinatária receber o pacote, deverá saber se deve enviar para o protocolo TCP, UDP ou outro, lembra-se? Um fato interessante é que, atualmente, este campo protocolo pode indicar protocolos de camada Rede, como o próprio IP[12.2]. Isso é útil quando temos um pacote IP encapsulado dentro de outro. Você pode não entender, neste momento, como isso pode ser útil, porém irá entender quando falarmos sobre segurança de redes.

O campo **Header Checksum**, como o próprio nome diz, contém o código de chegagem do cabeçalho. É semelhante ao campo de verificação dos quadros de camada enlace. Porém, o campo Checksum em pacotes IP verifica apenas a integridade do cabeçalho do IP, e não dos dados.

A seguir, temos os campos irmãos **Source Address** e **Destination Address**, que indicam a máquina que está enviando o pacote, e a máquina que o receberá. Você já está barbudo de saber como isso acontece, não?

O polêmico e indesejado (nos dias de hoje) campo **Options** é opcional, e possibilita a inserção de algumas opções que os roteadores ao longo do caminho ou o destino lerão. Essas opções não são relevantes para o entendimento do roteamento.

12.3. COMO ROTEADORES TRABALHAM

Roteadores são equipamentos de camada Rede, que tem a função de analisar um pacote e tomar uma decisão baseado, normalmente, no destino IP. Contudo, existem roteadores que tomam decisões muito mais complexas. Por exemplo, atualmente, está se tornando comum um roteador tomar decisões baseado em rótulos, em vez de endereços IPv4. Esturemos sobre isso em um momento posterior deste curso.

12.2. Uma lista dos valores que podem ser usados no campo Protocol encontra-se em [Wikipedia IPv4Protocols].

Abaixo, temos a representação de um pacote entrando em um roteador.

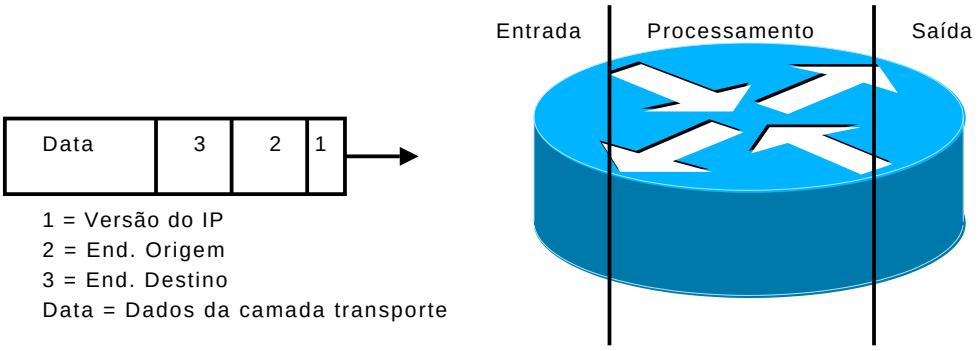

Figura 12.6. Um pacote entrando em um roteador.

O pacote acima encontra-se abreviado, com apenas 3 campos e os dados. Observe que a ordem de leitura do cabeçalho IPv4 é a mesma mostrada na figura. Podemos dividir basicamente o roteador em três: entrada, processamento e saída[12.3].

- Entrada: é onde porta onde o pacote chega.
- Processamento: é onde funciona a lógica do processador. As decisões são tomadas aqui.
- Saída: é a porta de onde o pacote sairá.

Uma única porta pode ser de entrada e saída, como acontece em todos os roteadores. Quando o pacote chega pela porta de entrada, pode encontrar duas situações: ou a porta está absolutamente livre, ou está ocupada.

- **Situação 1: a porta de entrada está livre**. Nessa situação, o pacote é encaminhado diretamente para o processamento do roteador.
- **Situação 2: a porta de entrada está com pacotes.** Isso acontece porque a capacidade de recepção dos pacotes do roteador é menor do que a recepção propriamente dita. Por exemplo, suponha que a velocidade que a porta de entrada é de 1Mbps. Se forem enviados 2Mbps para essa porta, irá acontecer um gargalo. Os roteadores possuem uma memória de armazenamento temporária para guardar pacotes que chegam na porta de entrada, caso esta esteja ocupada. Dessa forma, se dois pacotes chegam na porta de entrada, esta armazenará um deles na memória temporária (também chamada de *buffer*) e enviará o primeiro pacote para o processamento. Depois disso, poderá encaminhar o pacote armazeado, liberando a memória.

A respeito do que foi dito acima, saiba que a memória temporária das portas é finita. Por isso, se a porta de entrada não conseguir esvaziar a fila de pacotes que se formam, novos pacotes serão descartados[12.4]. Além disso, a velocidade de operação da porta de entrada não é a única coisa que gera filas; o próprio processamento do roteador pode causar isso. As portas de entrada só podem enviar pacotes para o processamento se este estiver liberado.

12.3. Uma explicação mais profunda pode ser encontrada em [Kurose & Ross] págs. 247-255.

12.4. [Kurose & Ross] pág. 252.

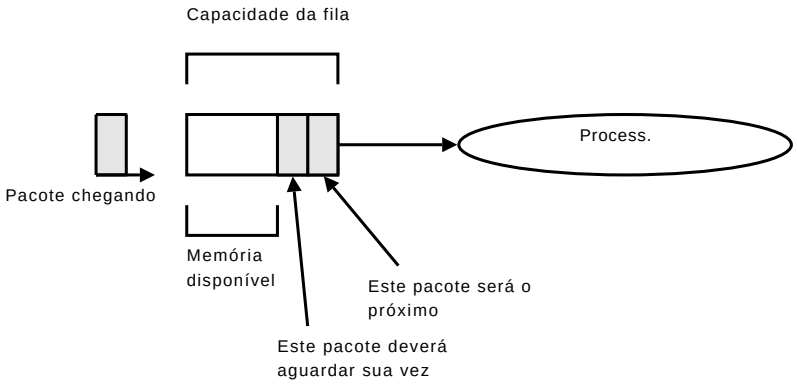

Figura 12.7. Formação de filas na porta de entrada.

É no processamento do roteador que ocorre a tomada de decisões. A primeira decisão que o roteador tomará é se deve trabalhar com um pacote IPv4 ou IPv6, pois roteadores de hoje são compatíveis com ambos os protocolos. O processador lerá o campo *version* e deciriá como deve proceder daí por diante.

Depois, o processador do roteador consulta uma tabela, criada estaticamente no roteador ou atualizada de alguma forma pela rede, que contém regras quanto o encaminhamento de pacotes.

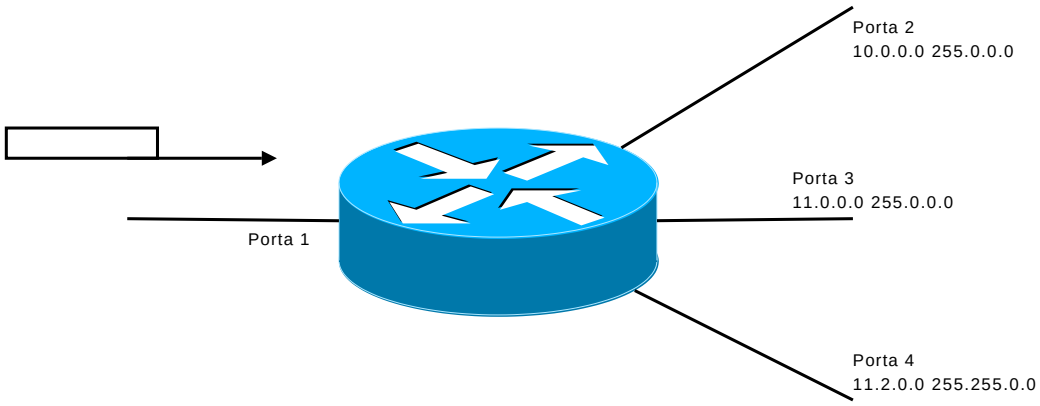

Figura 12.8. Tomando uma decisão.

Observando a figura acima, observe que as portas 2, 3 e 4 (que representam a saída) contém um endereço de IP e uma máscara de rede. O roteador, na fase de processamento, vai analisar o endereço de destino do pacote para ver se combina - isto é, se pertence a rede associada à porta de saída.

Bom, o roteador **sabe** (e você também, pois já estudou números binários) que o destino 10.3.5.2 pertence a rede 10.0.0.0 máscara 255.0.0.0, e irá encaminhar o pacote pela porta 2. Pronto problema resulvido.

Vejamos agora um pacote com destino 11.2.8.7. Aí surge um problema, pois esse pacote pertence tanto à rede 11.0.0.0 máscara 255.0.0.0 quanto à rede 11.2.0.0 máscara 255.255.0.0. O que o roteador decide? Encaminhar pela porta 3 ou pela porta 4?

Bom, nesses casos, será encaminhado pela porta 4, pois a lógica do roteador diz para usar a regra mais restritiva, e sabemos que a rede 11.2.0.0 255.255.0.0 é mais restrita (por ser menor) do que a outra rede.

O processado enviará o pacote para a porta correta, somente se a porta de saída estiver disponível ou haja espaço na memória temporária dela. A figura abaixo ilustra esse fato, dando ênfase na probabilidade de ocorrer filas e descarte de pacotes ao longo de todo o roteador.

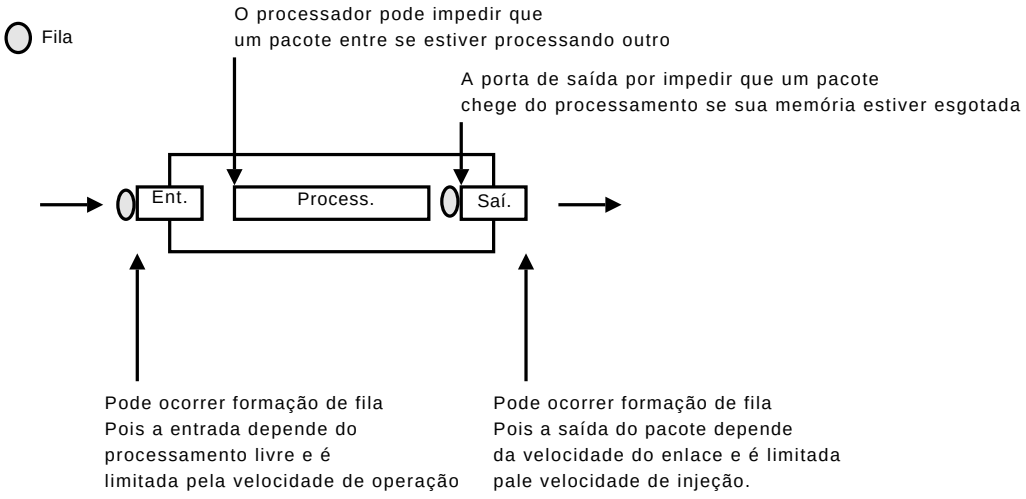

Figura 12.9. Motivos que levam à formação de filas.

A parte mais falada do roteador é o processamento. Falamos de processamento baseado no endereço de destino; todavia, existem outros tipo de processamento. Programação de roteadores é um assunto bastante extenso, mas vale a pena estudar sobre isso. Quem sabe farei isso em outro livro? (olha a propaganda!)

12.4. Introdução à Lógica de Roteamento

Bom, você já viu que um roteador possui uma tabela de roteamento. Com base nela, é roteador sabe para que porta de saída deve encaminhar o pacote que está sendo processado. É possível inserir manualmente no roteador essa tabela, mas em um ambiente que existem dezenas de roteador, seria um tanto trabalhoso fazer isso em cada um deles - sem contar que, quando alguma rota da rede fosse alterada, todos os roteadores deveriam ser alterados manualmente.

O fato é que, hoje em dia, os processadores se comunicam por meio de protocolos de roteamento. Isso mesmo, roteadores têm uma linguagem própria. Eles trocam informações entre si. Entretanto não existe apenas um protocolo, e sim, muitos, cada um com seu comportamento distinto. Para um exemplo simples, suponha que temos três roteadores em uma rede[12.5] empresarial pequena:

12.5. Rede aqui é um termo geral.

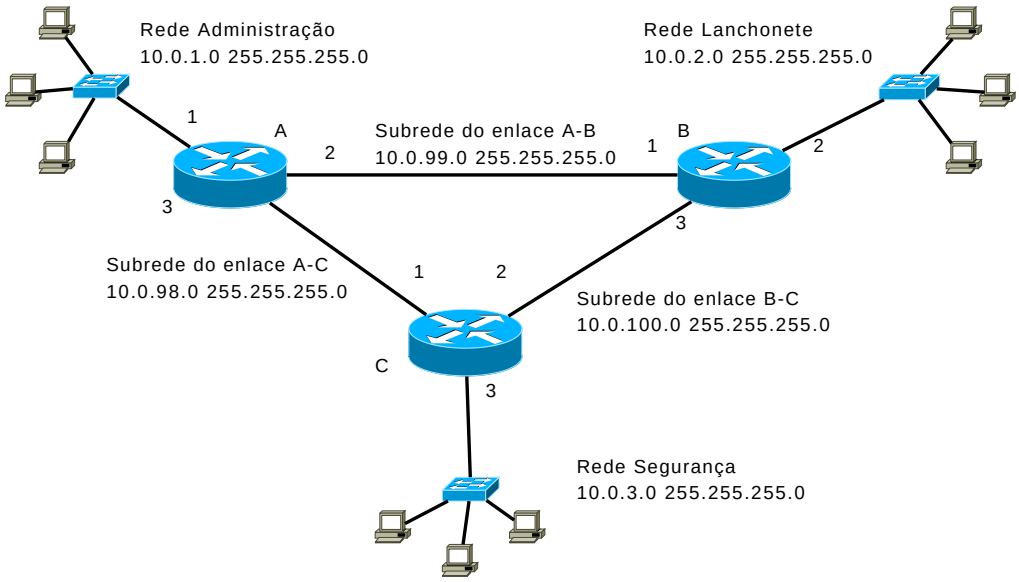

Figura 12.10. Exemplo de rede com três roteadores.

Observe que todo enlace saindo dos roteadores é uma rede diferente. Até mesmo enlaces sem hospedeitos (ou seja, enlaces que interligam roteadores) possuem um endereço de rede e uma máscara. Sabe porque? Porque é necessário que cada porta do roteador tenha um endereço IP e uma máscara. Os roteadores não conversam entre si por meio de endereços MAC, e sim por meio de IP's. Um chama pelo IP do outro. Você deve estar questionando-se: "é um grande desperdício dedicar uma rede para conectar roteadores". Sim, no nosso caso desperdiçamos 252 endereços atribuíveis porque escolhemos uma máscara que forma subredes grandes. Poderíamos usar subredes menores, mas isso é outra história. Vamos focar no fato de que cada roteador está conectado a duas subredes.

A primeira regra é: a primeira coisa que um roteador aprende são as conexões diretas a ele. Isso é óbvio, pois se cada porta do roteador possui um endereço IP e uma máscara, ele vai saber a que subredes está conectado. Então, em um primeiro momento, cada tabela de roteamento irá exibir três linhas, conforme abaixo:

	Roteador A	
IP	Máscara	Porta
10.0.1.0	255.255.255.0	1
10.0.99.0	255.255.255.0	2
10.0.98.0	255.255.255.0	3

	Roteador B	
IP	Máscara	Porta
10.0.99.0	255.255.255.0	1
10.0.2.0	255.255.255.0	2
10.0.100.0	255.255.255.0	3

	Roteador C	
IP	Máscara	Porta
10.0.98.0	255.255.255.0	1
10.0.100.0	255.255.255.0	2
10.0.3.0	255.255.255.0	3

Tabela 12.1. Tabelas de roteamento.

Com essas tabelas de roteamento, os roteadores já conseguem trabalhar. Por exemplo, se uma máquina de rede Administração quiser falar com alguma máquina que seja temporariamente conectada à rede 10.0.99.0, essa máquina encaminhará o pacote para seu Gateway Padrão (o roteador A), e este irá encaminhar o pacote pela porta 2. O problema é que nõa há nenhuma máquina nessa rede.

Se um hospedeiro da rede Administração quiser falar com um hospedeiro da rede Segurança, não iraá conseguir. Por quê?

- O hospedeiro em Administração irá enviar o pacote para o Gateway padrão.
- O Gateway padrão, que é o roteador A, analisará o destino do pacote; por exemplo, 10.0.3.25 (ou seja, rede Segurança).
- O roteador A procurará por essa entrada em sua tabela de roteamento, e não encontrará. Então, irá descartar o pacote e informar à máquina remetente que o destino está inalcançável.

"Mas o destino está alcançável!", diz você. Sim, está, você sabe pois está tendo uma visão semidivina das coisas, mas o roteador A não sabe disso. Ele só sabe quais são as redes que estão diretamente conectadas a ele. Para que ele saiba que existem outras redes no prédio, é necessário que algum outro roteador fale com ele sobre isso. O mesmo acontece com os roteadores B e C.

12.5. Atualização das tabelas

Sabemos que as tabelas precisam ser atualizas de alguma forma. Tendo como base a figura da seção anterior, e as tabelas dos roteadores A, B e C, é possível adicionar estaticamente uma rota no roteador A afim de que ele saiba que a rede Segurança existe. Como isso seria feito?

Bom, seria feito através de um comando dado no roteador. Que comando é esse, foge do escopo desse livro, mas ao término do comando a tabela do roteador A ficaria assim:

	Roteador A		
IP	Máscara	Porta	Tipo
10.0.1.0	255.255.255.0	1	Direto
10.0.99.0	255.255.255.0	2	Direto
10.0.98.0	255.255.255.0	3	Direto
10.0.3.0	255.255.255.0	3	Estático

Tabela 12.2. Tabela do roteador A depois da atualização.

Observe que adicionei a coluna "tipo" à tabela, com o objetivo de mostrar como aquele endereço foi parar ali. "Direto" significa que o roteador aprendeu o endereço que estava diretamente conectado a ele; "estático" significa que alguém colocou, manualmente, através de algum comando, aquela rota ali.

Será que, agora, um pacote originário da Administração chegaria à Segurança? Será que a máquina remetente obteria alguma resposta? Fiz duas perguntas aqui. Vamos analisar:

- O hospedeito, cujo IP é 10.0.1.50 (Administração), envia um pacote com destino 10.0.3.25 (Segurança). O pacote é enviado do hospedeito para seu Gateway padrão, que é o roteador A.
- O roteador A verifica em sua tabela de roteamento se tem uma entrada para aquela rede. Sim, tem. A rede 10.0.3.0 máscara 255.255.255.0 combina com o pacote 10.0.3.25, pois esse pacote pertence a essa rede. A porta configurada para isso é a 3. O roteador envia, sem pensar, o pacote pela porta 3.

- O pacote trafega no enlace que liga o roteador A ao roteador C; inevitavelmente chegará ao roteador C. Este recebe o pacote, e analisa o endereço de destino. Embora nada tenha sido alterado na tabela de roteamento de C, ele sabe que deve enviar esse pacote pela porta 3, pois a rede 10.0.3.0 está diretamente conectada a ele.

Então, o pacote chega ao seu destino.

Agora, vamos para a segunda parte:

- O destino vai responder a quem lhe enviou um pacote. Agora a máquina da Segurança cria um pacote cujo endereço de origem é seu próprio endereço, e o endereço de destino é 10.0.1.50.

- Após criar o pacote, a máquina da Segurança envia-o para seu Gateway padrão, que é o roteador C. Ele analisa sua tabela, e constata que:

	Roteador C		
IP	Máscara	Porta	Tipo
10.0.98.0	255.255.255.0	1	Direto
10.0.100.0	255.255.255.0	2	Direto
10.0.3.0	255.255.255.0	3	Direto

Tabela 12.3. Tabela do roteador C.

Sim, isso mesmo. O roteador não sabe para onde deve enviar o pacote, então descarta-o e manda uma mensagem "destino inalcançável" para a máquina da Segurança. Trágico. O pior é que a máquina da Administração ficará esperando eternamente, pois não recebeu nenhuma mensagem de erro, e crê que eu destino foi alcançado.

Se você está pensando em um comutador agora, deverá opinar: "bem, porque o roteador C não inseriu uma linha automaticamente na tabela quando recebeu o pacote que vinha da Administração?". Isso é uma operação de comutadores, e não de roteadores. Roteadores preferem conversar com seus vizinhos através de protocolos de roteamento, e não tomar decisões com base em suposições. Se nenhuma informação oficial é recebida, então o roteador não muda sua tabela de roteamento.

"Mas neste exemplo não há nenhum protocolo de roteamento em operação!". Exatamente. O jeito seria adicionar estaticamente mais uma linha na tabela de roteamento de C, como foi feito com A. Na verdade, para que a rede funcionasse sem problemas, o mesmo deveria ser feito no roteador B.

"Isso é bastante trabalhoso, mesmo nessa rede de apenas três roteadores!". Sim, é verdade. Por isso, roteadores anunciam o que sabem através de protocolos de roteamento.

12.6. ANUNCIANDO AOS VIZINHOS

Existem muitos protocolos de roteamento, mas não falaremos de nenhum aqui. Só entenda a importância e o funcionamento básico do mesmo. A figura abaixo mostra um exemplo de protocolo de roteamento.

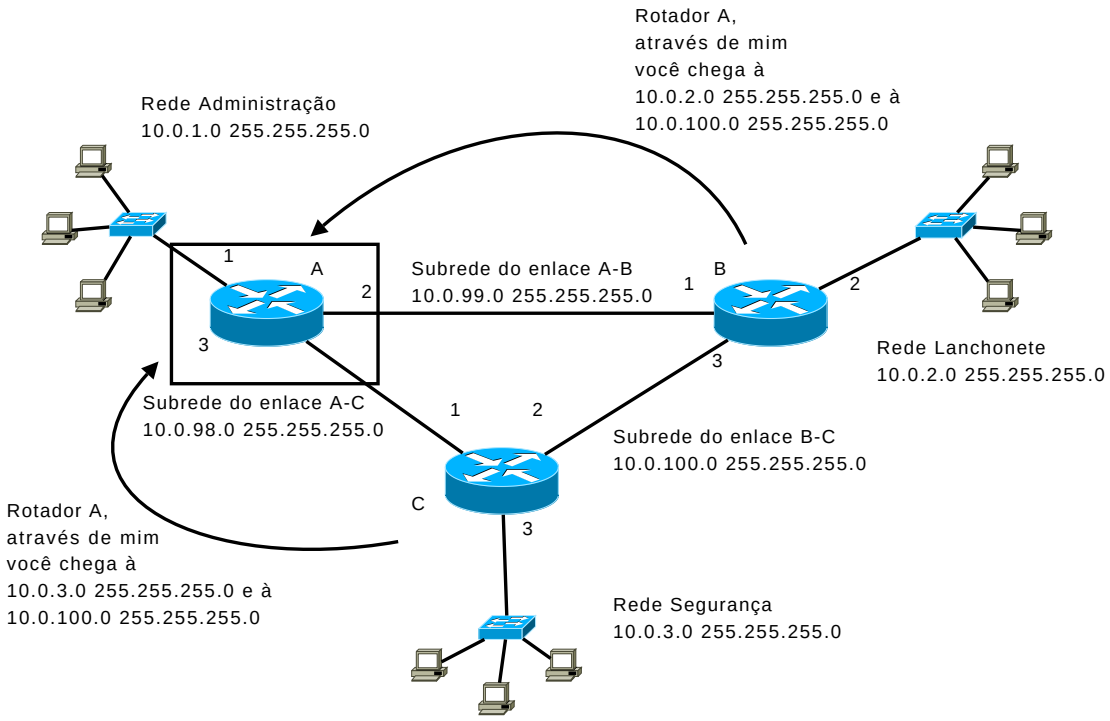

Figura 12.11. Roteadores B e C enviando atualizações para roteador A.

A definição de nosso protocolo hipotético diz que cada roteador deve enviar informações do que sabe para todos os seus vizinhos. Para facilitar as coisas, na figura acima é a vez dos roteadores B e C enviarem o que sabem para o roteador A. Quando A receber essas informações, irá atualizar sua tabela de roteamento, ficando assim (excluímos propositalmente a entrada estática adicionada pelo programador do roteador):

IP	Máscara	Porta	Tipo	De quem recebeu?
10.0.1.0	255.255.255.0	1	Direto	Local
10.0.99.0	255.255.255.0	2	Direto	Local
10.0.98.0	255.255.255.0	3	Direto	Local
10.0.3.0	255.255.255.0	3	Protocolo	Roteador C
10.0.2.0	255.255.255.0	2	Protocolo	Roteador B
10.0.100.0	255.255.255.0	3	Protocolo	Roteador C
10.0.100.0	255.255.255.0	2	Protocolo	Roteador B

Tabela 12.4. Tabela de A após atualização através de protocolo.

Um fato interessante é que o roteador A recebeu duas rotas distintas para o mesmo destino: ou seja, 10.0.100.0 via B e via C. Qual dessas o roteador A escolherá para enviar pacotes? Qualquer um, pois não há diferença entre elas. Hoje em dia, roteadores usam um conceito de métricas e distâncias administrativas para determinar qual a melhor rota, mas isso está fora do escopo desse capítulo.

Observe que, como a informação recebida não é estática, ela pode ser alterada assim que uma nova rota surgir na rede. Basta que o novo roteador a ser adicionado informe aos seus vizinhos o que sabe.

Um fato interessante precisa ser dito: protocolos de roteamento mais antigos não enviavam a máscara através das mensagens de roteamento. Isso tornava impossível trabalhar com subredes como usamos em nossos exemplos. Felizmente, hoje em dia, os protocolos de roteamento enviam as máscaras, juntamente com os IP's das redes.

AVISO 12.1. Existe muita, muita coisa para se falar sobre protocolos de roteamentos. Infelizmente, o espaço aqui é curto, e seria muito difícil abordar configuração de roteadores e análise dos protocolos atuais, como OSPF, por exemplo. Isso ficará para um outro curso de redes, mas avançado e posterior a esse.

12.7. CONCLUSÃO

Neste capítulo demos uma pincelada no conceito de rotemento IPv4. Você viu os campos que formam o cabeçalho do IP, e qual a função de cada um. Viu que o roteador analisa campo por campo para saber o que deve fazer com o pacote. A fragmentação consiste em dividir um pacote grande em vários pacotes pequenos, mas isso vai gerendo um problema na medida que desperdiça-se espaço para isso como novos cabeçalhos IP's. A fragmentação ocorre porque em alguma parte do trânsito do pacote o enlace não suportava que o pacote fosse transportado em sua totalidade. Cada enlace tem um MTU diferente. Quem junta o pacote fragmentado não são os roteadores, e sim, o hospedeiro que recebe-o.

Você viu também como um roteador trabalha. Dividimos o roteador em três partes: entrada, formado pelas portas de entrada, processamento, formado pelo processador e sua lógica de decisão, e saída, formado pelas portas de saída. É possível haver formação de filas na porta de entrada e de saída, mas não no processador do roteador, pois enquanto o provcessador estiver ocupado, os pacotes devem esperar na fila de entrada. A fila é armazenada em uma memória limitada; se a fila transbordar, novos pacotes recebidos são descartados. Nem sempre é assim, pois alguns roteadores possuem várias filas lógicas por portas, a fim de diferenciar tráfego mais e menos prioritário.

Para encaminhar um pacote corretamente, o roteador consulta sua tabela de roteamento. A primeira coisa que um roteador aprende são suas rotas diretamente conectadas. É possível que um operador de redes adicione uma linha à tabela de roteamento, usando, para isso, da linguagem de programação do roteador. Embora possível, o desejável é que roteadores atualizem dinamicamente suas tabelas através de algum protocolo de roteamento.

Este capítulo foi o último sobre IPv4. O IPv4 será substituído pelo IPv6, por isso, ele será o tema dos capítulos que se encontram adiante neste livro.

12.8. EXERCÍCIOS

Exercício 12.1. Defina a função do campo *Version*.

Exercício 12.2. Defina a função dos campos *Source Address* e *Destination Address*.

Exercício 12.3. Os únicos campos processados pelo roteador são *Version, Source Address e Destination Address*.

a) Verdadeiro

b) Falso

Exercício 12.4. O conteúdo do campo *Data* é processado pelo roteador.

a) Verdadeiro

b) Falso

Exercício 12.5. A função do campo *Protocol* é... (marque uma alternativa)

a) Informar ao roteador qual o protocolo usado na camada rede.

b) Informar ao roteador qual o protocolo usado na camada enlace.

c) Informar ao hospedeiro destinatário qual o protocolo usado na camada transporte.

d) Informar o tempo de vida do pacote.

Exercício 12.6. Qual a função do campo TTL (*Time To Live*)? (marque duas alternativas)

a) Informar ao roteador se o pacote deve prosseguir ou deve ser descartado.

b) Descartar o pacote quando assim que seu tempo de vida informado no campo, em segundos, expirar.

c) Descartar o pacote quando o número no campo for igual a 0.

d) Avisar para o destinatário a duração da viagem do pacote, em segundos.

Exercício 12.7. Onde podem acontecer filas no roteador?

a) Nas portas de entrada

b) Nas portas de saída

c) No processador

d) Nas tabelas de roteamento

Exercício 12.8. Qual a diferença entre atualizar estaticamente e dinamicamente a tabela de roteamento?

Exercício 12.9. Existem vários protocolos de roteamento.

a) Verdadeiro

b) Falso

Exercício 12.10. Observe a figura abaixo.

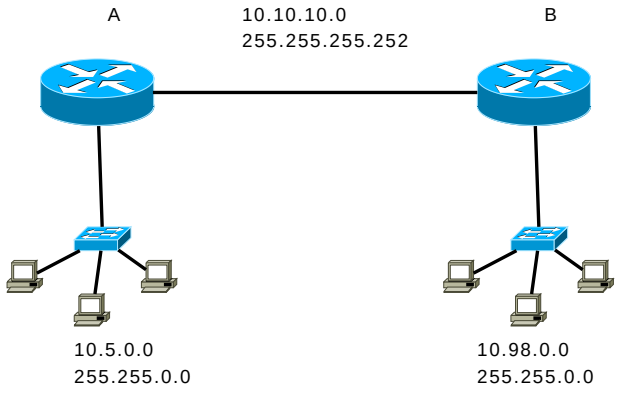

Figura 12.12. Rede com dois roteadores.

Preencha as tabelas abaixo, seguindo as seguintes regras:

a) Existe um protocolo de roteamento na rede e os roteadores já trocaram informações.

b) No campo tipo, preencha com “Direto” para rotas diretamente conectadas e “Dinâmico” para rotas aprendidas pelo protocolo.

c) No campo origem, preencha com “Local” ou o nome do roteador, conforme o caso.

IP	Máscara	Tipo	Origem

Tabela 12.5. Tabela do roteador A.

IP	Máscara	Tipo	Origem

Tabela 12.6. Tabela do roteador B.

Parte IV

Internet

Capítulo 13

Conexão ADSL

13.1. Introdução

Hoje em dia, muitas pessoas que têm banda larga em casa usam ADSL. Claro, as provedoras de serviço não dizem que sua conexão é, pois se os usuários soubessem disso, comprariam seus próprios modems em vez dos da operadora.

No Rio de Janeiro temos um serviço chamado Velox da Oi; em São Paulo, o Speedy da Telefônica; em outros estados, onde outras operadoras têm um monopólio das comunicações locais, temos outros nomes para serviços de banda larga, com aproximadamente as mesmas taxas de velocidade e os mesmos preços, bem como a quantidade de reclamações de usuários insatisfeitos.

O serviço, como dissemos, é o mesmo. Para conectar um computador à internet usando-se ADSL, é preciso um modem ADSL, um roteador comum, e, é claro, o computador propriamente dito. *Não é necessário um modelo específico de modem.* As operadoras, contudo, tentam manter os usuários ignorantes a esse respeito, para assim venderem seus serviços alugando um modem por um preço ridicularmente alto (todo aluguél é caro se compardo à compra), ou então obrigam o usuário a assinar um contrato para que este venda sua alma.

A primeiro coisa que você precisa saber é: qualquer modem compatível com os serviços ADSL pode ser usado para conectar-se à internet (até porque modems compatíveis com ADSL no Brasil têm um selo de homologação da Anatel).

13.2. O Modem

O que é um modem? Modem é simplesmente um conversor.

Definição 13.1. *Modem. Modem é abreviação de dois termos:* **mo***dulador e* **dem***odulador; sua função é modular, ou seja, conventer sinais elétrico analógico para sinais elétricos digitais, e demodular, ou seja, converter sinais elétricos digitais para sinais elétricos analógicos, afim de que dois dispositivos digitais possam comunicar-se por um meio físico onde dados são transferidos em modo analógico.*

Você se lembra de como funciona a transmissão dos dados na rede local pela camada física? Depois que o computador de origem encapsula os dados pela pilha de protocolo até chegar na camada enlace, o que acontece? Já estudamos sobre isso. Vamos relembrar.

A placa de rede, neste momento, está com um quadro de camada enlace para transmitir. Este quadro contém essencialmente o endereço físico de origem, o endereço físico de destino, os dados da camada imediatamente superior (ou seja, um pacote da camada rede) e um campo para que a placa de rede de destino verifique se os dados estão com alguma corrupção. Em redes locais, o protocolo de camada enlace normalmente usado para conversação é o Ethernet. A placa de rede de origem, então, transmite esse quadro pelo enlace físico, que normalmente são cabos de par trançado, feitos de cobre. Em outras palavras, a placa de rede dá choques nos fios de cobre; esses choques são sentidos pela placa de rede destinatária (pois em redes atuais que usam comutador, só a máquina de destino recebe as informações). Os choques que a máquina destino recebe são traduzidos pela placa de rede em quadros de camada enlace.

Os bits 1 são representados por choques de 5v; os bits 0 são representados pela ausência de choques, ou por choques de -5v.

Pois bem. Quando você acessa a internet banda larga, usando um serviço ADSL (ou seja, Velox, Speedy etc), você não está mais usando uma placa de rede Ethernet para comunicar-se com uma máquina na outra ponta. Você está, na verdade, usando um enlace físico e equipamentos físicos da companhia telefônica; e esse meio físico, o cabo telefônico da sua casa até a companhia telefônica, não transporta informações como fazemos nas redes locais. Esse meio físico transporta informações de uma forma diferente da que sua placa de rede trabalha.

A sua placa de rede emite apenas dois tipos de sinais elétricos, como já vimos; um deles sinaliza os bits 0, e outro sinaliza os bits 1. Ou seja, é uma sinalização elétrica binária: ou um, ou zero. Ou sim, ou não. Chamamos esse sinal elétrico de sinal digital. Você lembra qual é o formato de um sinal digital?

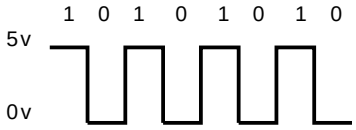

Figura 13.1. Sinal digital.

Fora da rede local, no enlace físico que vai da sua casa até a companhia telefônica, não se usa sinais digitais. Em vez disso, usam-se sinais analógicos. Seu telefone usa sinais analógicos para enviar e receber informações da companhia telefônica. Isso acontece porque é mais fácil gerar um sinal analógico do que um sinal digital: sua voz é analógica, e tudo que um telefone precisa fazer é converter esta onda analógica mecânica que é sua voz, em um sinal elétrico analógico, e enviar para a central. O sinal analógico tem formato de onda.

Figura 13.2. Sinal analógico.

E chegamos no ponto onde um modem é necessário. Seu computador entende e fala sinal digital; o enlace físico que vai da sua casa até a companhia telefônica transporta apenas sinais alaógicos... e agora José? O natural é alguém ter inventado um equipamento físico que converte um sinal para outro, um equipamento capaz de *modular* o sinal digital em analógico e, logicamente, *demodular* o sinal analógico em digital. Este equipamento é o modem.

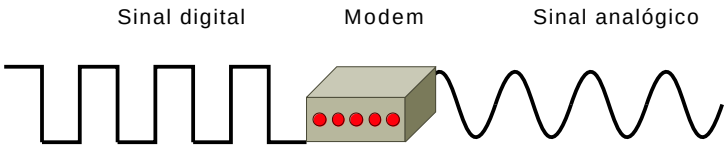

Figura 13.3. Modem: um conversor.

Dessa forma, se dois equipamentos digitais querem conversar, contudo o meio físico é analógico, os dois equipamentos se conectarão a um modem cada um e conversarão com eles por meio de sinais digitais; os modems, por sua vez, se conectarão ao meio físico analógico, falando analogicamente... pegou o trocadilho?

Vamos voltar ao contexto do ADSL. Você tem uma linha telefônica por perto, sabe que é um meio de transmissão analógico, mas mesmo assim, quer conectar-se à internet através dela. O que você fará? Comprará um modem. Existem vários tipos de modem, e um modem para conexões ADSL deve ser um modem ADSL, óbvio (não vá comprar um modem comum achando que ele é ADSL!). O chassi de um modem ADSL normalmente possui duas portas: uma RJ-11 e outra RJ-45[13.1].

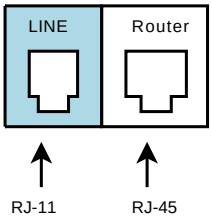

Figura 13.4. Representação do chassi de um modem.

A porta RJ-11 é menor do que a RJ-45, e serve para conectar a linha telefônica. É, digamos assim, a porta WAN, embora essa definição seja incorreta, visto que um modem não opera na camada rede da arquitetura TCP/IP e não tem, portanto, informações de IP, máscara etc. O modem é um equipamento de camada física; ele apenas traduz sinais elétricos; nem sequer os interpreta: bits de camada física continuam sendo bits de camada física, sem se transformar em quadros de camada enlace.

A porta RJ-45 deve ser ligada a um cabo de par-trançado usado em redes locais; você já está familiarizado com este tipo. Para um único computador conectado à internet, o cabo que sai da porta RJ-45 do modem é conectado à placa de rede do computador. Veremos mais sobre conexões adiante; por agora, é importante que você tenha mentalizado o que é, e qual a função básica de um modem: converter sinais.

13.3. Multiplexação por divisão de frequência

Antes de continuar, você deve prestar atenção para um fato simples: conexão ADSL não é a mesma coisa que conexão discada. Na conexão discada, quando você está na internet a linha fica ocupada e não é possível telefonar; ou telefona, ou acessa[13.2]. Já na ADSL, é possível telefonar para o Papa e acessar a internet; na verdade, em ADSL não é necessário sequer discar para o provedor de internet: a conexão está sempre disponível.

13.1. Modems tradicionais que não são ADSL geralmente não possuem uma porta RJ-45 para placa de rede de computador; em vez disso, possuem uma porta serial, para ser ligada na porta serial do computador.

13.2. É claro que não é isso distingue conexão discada de ADSL; isso é apenas um benefício. Existem muitas características técnicas que distinguem os dois tipos de conexão.

A pergunta é: como é possível, em um mesmo meio físico que é o cabo telefônico da sua casa até a central, transportar ao mesmo tempo dados (ou seja, pacotes da camada enlace) e voz? A resposta é: através da multiplexação por divisão de frequência.

DEFINIÇÃO 13.2. *Multiplexação por divisão de frequência, ou FDM (Frequency Division Multiplexing) é a técnica de se distinguir informações em diferentes frequências, encapsulando todas elas em um mesmo meio físico. O oposto disso, a demultiplexação, consiste em separar essas mesmas informações que vêm pelo meio físico.*

Multiplexar significa juntar. Demultiplexar significa separar. Mas não é uma "junção" porca: é uma junção higiênica, pois embora se unam informações, elas não se misturam. E porquê? Porque frequências diferentes não se misturam.

O que distingue uma frequância de outra é a largura da onda. Quando você usa o telefone e fala com alguém, seu telefone converte sua voz em uma onda e transmite tal onda pelos fio de cobre do cabo telefônico. Porém, se alguém que tenha uma voz bem mais aguda que você (uma criança, por exemplo) falar ao mesmo tempo, a ligação não vai cair! O fio de cobre do cabo telefônico irá transmitir a sua voz e a voz da criança ao mesmo tempo. A voz da criança tem um formato de onda mais curto que o seu, e o telefone converte isso em uma onda de frequência mais alta. As duas ondas (a que corresponde à sua voz e à voz da criança) são transmitidas ao mesmo tempo pelo cabo telefônico, sem interrupção, pois as ondas não se misturam. Teoricamente, se você tivesse mil ondas em frequências diferentes, todas elas seriam transmitidas pelo cabo telefônico. As ondas não se misturam.

Nossa audição tem uma limitação. Só conseguimos escutar ondas sonoras até 4000 Hertz (ou seja, 4 Kilohertz, Khz). Se você é um pouco surdo por ter ouvido muito MP3 no volume máximo, talvez escute menos que isso. Quando mais a frequência aumenta, mais agudo (ou seja, fino) fica o som. Um apito tem uma frequência alta. Um apito mais fino ainda, tem uma frequência mais alta ainda, de forma que quanto mais alta a frequência, mais agudo é o som que você escuta. Chega um momento em que a frequência é tão alta, e o som é tão agudo, que você não escuta-o mais. Porém o som continua aí, no ar. Ele ainda existe; você é que não consegue percebê-lo. É por isso que dizemos que frequência acima de 4000Hz (isto é, 4Khz) não são "audíveis".

Os projetistas do ADSL tinham noção disso. Veja o que eles observaram:

- A audição humana escuta até 4Khz.
- Um telefone converte a voz de quem fala (onda mecânica) em um sinal elétrico, e o transmite pelo cabo. O sinal convertido tem a mesma frequência da onda sonora; assim, uma pessoa que fale, por exemplo, na frequência sonora 2Khz tem essa onda sonora convertida em sinal elétrico de 2Khz. Na outra ponta, o telefone reconverte o sinal elétrico analógico em onda sonora, também na mesma frequência.
- Um cabo telefônico tem capacidade de transportar sinais elétricos em frequências além de 4Khz. Na verdade, a capacidade do cabo telefônico é muito maior do que o usado pela audição humana.

Em conexões discadas, a linha telefônica é ocupada, pois tanto o computador quanto o telefone usam toda a capacidade do cabo para transmitir sinais analógicos; o cabo é inteiramente ocupado. Assim, mesmo que você fale e escute abaixo de 4Khz, em linhas analógicas comuns todo o cabo é usado, impedindo que outras informações sejam transmitidas.

Já em linhas ADSL, há uma organização das frequências: as frequências abaixo de 4Khz, somente as que estao abaixo disso, são usadas pelo telefone. E as frequâncias acima de 4Khz, e somente as acima disso, são usadas para transmissão de dados. Lembre-se de que esse uso é possível, visto que as frequências não se misturam: os telefones escutam na faixa que vai até 4Khz; os computadores, na faixa que está acima desse limiar.

Além disso, dentro da faixa de frequências reservadas para dados, há ainda uma subdivisão para recebimento de dados (download) e envio (upload). A faixa reservada para recebimento é bem maior do que a reservada para envio, pois normalmente usuários baixam mais arquivos da internet do que enviam.

Abaixo, uma representação da multiplexação por divisão de frequência no meio físico analógico que vai da sua casa até a central telefônica.

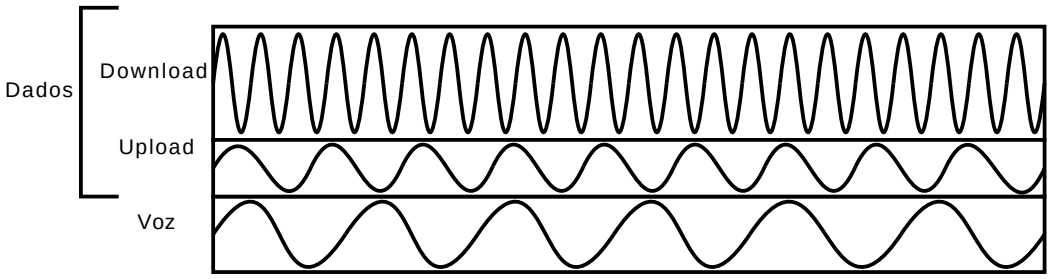

Figura 13.5. Divisão de frequência na linha ADSL.

Você deve ter compreendido que a multiplixação por divisão de frequência é o ato de a origem das informações unir, em um único meio físico, sinais analógicos diferentes (ou seja, de frequências diferentes). Isso é possível porque eles não se misturam. De forma similar, o destino demultiplexa, ou seja, separa os sinais analógicos: sinais abaixos de 4Khz vão para os equipamentos que processam voz, e sinais acima de 4Khz vão para equipamentos que processam dados.

Em linhas discadas comuns, sempre precisava-se discar para um número, afim de estabelecer chamadas; mesmo que você não quisesse telefonar, e sim acessar o provedor de internet. Isso era necessário porque o circuito era fechado, e quando você tirava o telefone do gancho, abria um circuito entre você e a operadora de telefonia; ao discar para o número, o circuito era comutado até o destino.

Em linhas ADSL, é necessário discar apenas se você quer usar o telefone ($<$ 4Khz); para frequências acima de 4Khz, o circuito já está estabelecido, e não é necessário discar um número; apenas enviar quadros de camada enlace para o provedor de acesso. Ou seja, o provedor já "está na linha", por assim dizer; a comutação do circuito está previamente estabelecida.

Até agora, explicamos o conceito de multiplixação, mostrando como é possível dados e voz trafegarem por um único meio físico. Mas ocultamos uma coisa importante desse sistema. Observe que o telefone de uma linha comum é o mesmo telefone usado em linhas ADSL, e esse tipo de telefone ocupa toda a linha, e não apenas frequências abaixo de 4Khz. O modem pode até ser inteligente o suficiente para saber em que frequências deve falar e escutar, mas o telefone continua se intrometendo nas frequências mais altas.

Para resolver este problema, antes do telefone (e do modem) instalamos um pequeno equipamento chamado "separador", ou *splitter*.

DEFINIÇÃO 13.3. *Separador (splitter). Um separador tem a função de filtrar e separar sinais analógicos em uma linha ADSL. Para sinais que vêm da operadora de telefonia, o separador encaminha frequências abaixo de 4Khz para o telefone e acima disso para o modem. Para sinais que vão em direção à operadora de telefonia, o separador filtra o sinal do telefone, descartando sinais com frequencias superiores a 4Khz, e encaminhando apenas sinais inferiores a isso.*

Para melhor entendimento, analise a figura abaixo.

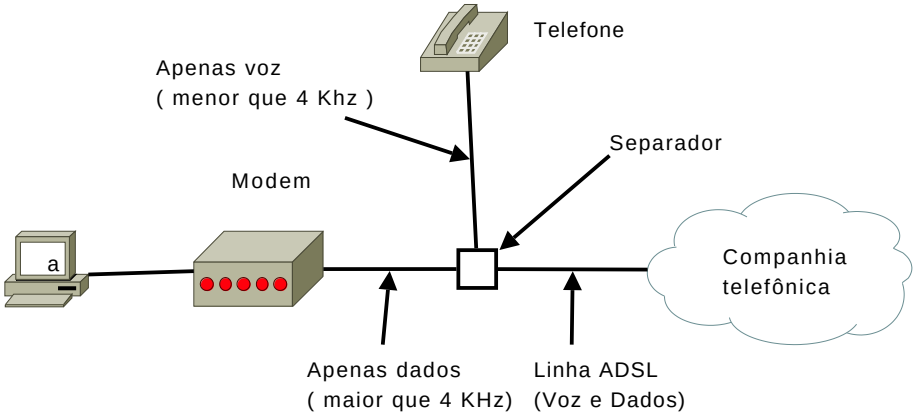

Figura 13.6. Frequências usadas para dados e voz em uma linha telefônica.

Como você pode ver na figura, o separador fica antes do modem e do telefone. Dessa forma, o telefone recebe apenas frequências abaixo de 4Khz, e o modem, apenas frequências acima disso. O oposto também é verdadeiro: frequências vindas do telefone que ultrapassem esse valor são filtradas pelo separador. A linha ADSL depois do separador contém todos os sinais analógicos. Você deve imaginar, com razão, que o separador contém três portas: uma para a linha que vem da operadora, outra para conexão do telefone, e outra porta, para conexão com o modem.

Tenha noção de que o modem, portanto, só recebe sinais analógicos de frequências superiores a 4Khz. Para facilitar o entendimento, a partir de agora neste capítulo ocultaremos o telefone e o separador, mas tenha noção de que eles continuam presentes.

NOTA 13.4. Um separador (splitter) não é a mesma coisa que um filtro.

13.4. COMPUTADOR CONECTADO À ADSL

Agora, você já tem o modem ADSL, e uma linha ADSL que vem da operadora de telefonia. A linha já passou pelo separador e o que resta é uma linha com conector RJ-11, pronta para ser ligada ao modem. O modem liga-se, também, ao computador, conforme mostra a figura abaixo.

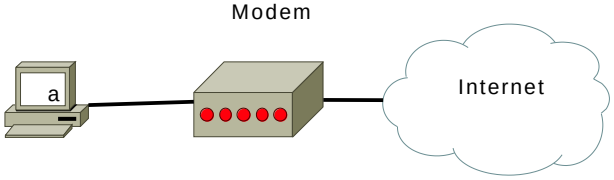

Figura 13.7. Acesso à internet através de modem.

Na figura acima, o computador está habilitado para falar com qualquer máquina na internet, desde que a camada enlace esteja funcionando corretamente. O provedor de acesso possui um servidor DHCP, de modo que a máquina da figura aprenderá dinamicamente tudo que precisa para se falar e ouvir na internet. O cabo que vem do modem conecta-se à sua placa de rede, e esta pode conversar normalmente usando sinalização digital. Qualquer quadro de camada enlace que o computador `a` enviar será traduzido para analógico pelo modem, encaminhado para fora; de forma similar, qualquer informação anaogica que chega ao modem será convertido para digital, e enviado para a placa de rede do computador.

A comunicação é digital, do ponto de vista da placa de rede da máquina `a`. O quadro de camada enlace usado para comunicação com o provedor é uma variação do Ethernet chamada Point-to-point Over Ethernet, ou simplesmente, PPPoE. Falaremos ainda neste capítulo sobre o PPPoE.

13.5. LAN conectada à ADSL

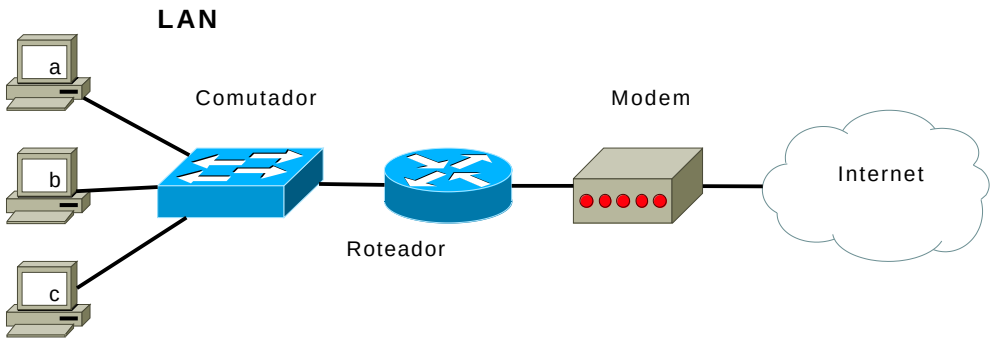

Figura 13.8. LAN conectada à internet através de um roteador.

Se você deseja conectar uma rede local à internet, em vez de apenas um computador, pode usar seu velho amigo roteador para isso. No caso, é a porta WAN do roteador que irá aprender as informações de camada rede oferecidas pelo provedor de acesso. Você está familizarado com a figura acima? A única coisa nova é o modem! Lembre-se de que quando o roteador tem acesso à internet, todo o resto da rede tem.

A porta LAN do roteador, como você pode estar imaginando, servirá de servidor DHCP para as máquinas da rede local. Algumas informações qua você preencherá no servidor DHCP foram aprendidas pelo roteador através da porta WAN (como por exemplo, o IP dos servidores DNS). Observe também que o Gateway padrão da rede local é a porta LAN do roteador.

13.6. Acoplamento de equipamentos

A indústria é inteligente. O papai noel sempre está arranjando um jeitinho de carregar menos presentes no seu trenó. Como? Ora essas, acoplando um brinquedo em outro. Um menino queria um cãozinho, um macaco e uma girafa. O velho Noel foi lá e lhe deu uma girafa de brinquedo com cara de macaco que latia. Um técnico de redes queria um modem ADSL e um roteador; o velho e bom Noel, então, criou um roteador que possui um modem ADSL em uma de suas portas. O técnico de redes ficou feliz, mas a criança - triste criança - cresceu traumatizada.

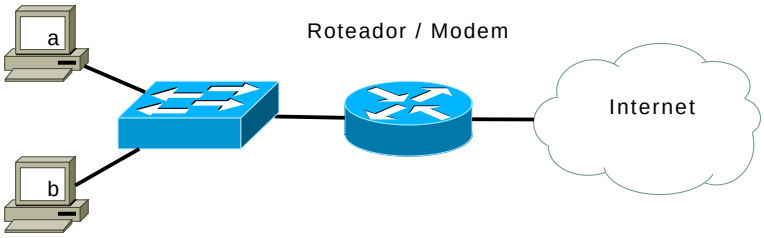

Figura 13.9. Roteador com modem ADSL acoplado.

Isso significa muito mais que economia de espaço. Isso significa economia de dinheiro. Para que gastar dois chassis se tudo pode ser acoplado em um único equipamento? E isso (o roteador com modem ADSL) é o tipo de coisa que vende que nem água. Você deve tomar cuidado é para não comprar um roteador "ADSL", pensando que ele é um modem também. Não se engane! TODO roteador serve para ADSL, desde que você tenha um modem. Contudo, nenhum roteador, nem os "roteadores ADSL", vêm com modem. Precisa tomar cuidado com esse tipo de propaganda tendenciosa. O que você precisa é comprar um roteador **com modem** ADSL acoplado. Bastante atenção com isso!

Uma dica simples é: se na embalagem do "roteador ADSL" não estiver escrito que ele possui um modem, não compre, pois é um roteador comum (que, como todos os outros roteadores comuns, serve para ADSL). Agora, se você ver escrito claramente que este roteador é também um modem, então pode comprar. Na verdade, no Brasil modems ADSL são homologados pela Anatel, comprovando que estão em conformidade com o padrão ADSL usado no Brasil. Se o tal roteador ADSL tiver escrito na caixa que é também um modem, e contiver um selo da Anatel, então significa que é um roteador com modem realmente. Já roteadores não precisam do tal selo.

Ou talvez você queira comprar um roteador "ADSL" que não tem modem (ou seja, é apenas um roteador comum que a palavrinha mágica ADSL escrito na embalagem para enganar desinformados), e em seguida comprar um modem ADSL homologado pela Anatel; contudo, atenção ao fazer isso, pois é meio complicado hoje em dia encontrar modems que sejam apenas modems ADSL. O que você encontrará são roteadores ADSL com modems ADSL.

Nunca é demais repetir: atenção! Verifique se o roteador possui também um modem ADSL, pois não basta ser roteador, tem que ser modem. Se você já tiver um modem ADSL e precisar de um roteador, tudo bem, mas se não tiver, PRESTE ATENÇÃO! E veja se seu candidato a roteador com modem acoplado é homologado pela Anatel, pois se ele for, ele realmente é um modem, tem o selo da Anatel na caixa ou no chassi, e foi testado e comprovado que está de acordo com os padrões ADSL usados no Brasil. Entendeu?

Normalmente, você encontra um modem com roteador ADSL com duas portas: uma para WAN, e outra para a LAN. A porta WAN tem formato RJ-11, ou seja, para linha telefônica, pois esse roteador contém um modem. A outra porta (a LAN) é RJ-45, para que você possa ligar seu computador, ou um comutador. Esse tipo de roteador com modem embutido pode ser configurados para servir de servidor DHCP na rede local. Nada mais natural.

Um outro equipamento bastante útil para ambientes domésticos ou pequenos escritórios é o roteador "doméstico" (ou caseiro) com modem. Ele é um roteador com modem, contudo com várias portas LAN. Essas portas LAN não interligam redes locais diferentes, mas sim, computadores de uma mesma rede local; é o típico roteador doméstico que já foi descrito neste curso. Tem sua utilidade, é excelente por já vir com modem ADSL embutido, entretanto não é um roteador tradicional, que interliga várias redes. Tanto é assim, que só roda um servidor DHCP para todas as portas LAN.

Esse tipo de roteador doméstico equivale a um roteador de duas portas, sendo a porta WAN conectada internamente a um modem ADSL, que por sua vez tem a porta RJ-11 para fora do chassi; e a porta WAN conectada a um comutador, que tem (no caso da figura abaixo) as três portas de comutação para fora do chassi.

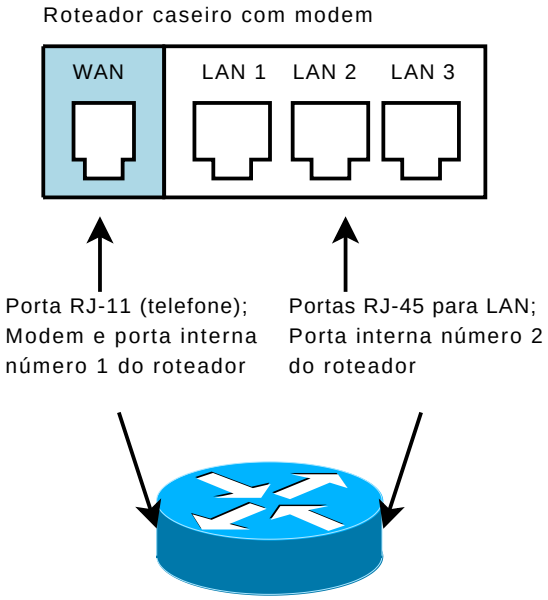

Figura 13.10. Roteador doméstico com modem.

13.7. Camada enlace ADSL: PPPoE

Já sabemos o protocolo de camada enlace usado na rede local: Ethernet. Contudo, a porta WAN do roteador (que está ligada ao modem e comunica-se com o provedor de acesso) usa um protocolo diferente para isso. Esse protocolo é o Point-to-Point over Ethernet (Ponto-a-Ponto sobre Ethernet), ou simplesmente PPPoE. A máquina conectada diretamente à internet deve saber encapsular pacotes da camada enlace no PPPoE, seja esta máquina um computador ou um roteador.

A função deste protocolo não é apenas prover comunicação de camada enlace entre duas máquinas; é, além disso, prover um meio de autenticação. Ou seja, o provedor precisa saber quem está se conectando; e o usuário deve fornecer uma senha ao provedor, para habilitar a conexão de camada rede. Por isso usa-se o PPPoE, pois este protocolo permite isso. Observe, na figura abaixo, onde usamos Ethernet e onde usamos PPPoE em ambientes ADSL.

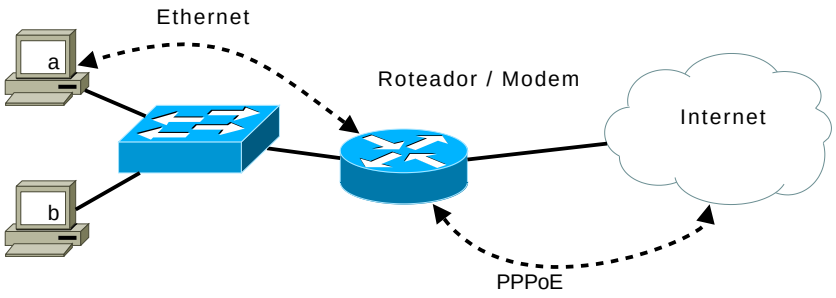

Figura 13.11. Protocolos de camada enlace usados na LAN e na conexão ADSL.

Na máquina conectada à LAN, você normalmente configura o nome do usuário e a senha. Estas informações são enviadas pela máquina ao provedor de acesso em um quadro de camada enlace PPPoE. Se o nome do usuário constar no banco de dados do provedor, e a senha for correta, E TAMBÉM o usuário tiver permissão para isso, então mais quadros de camada enlace PPPoE são trocados, e a conexão é estabelecida. A partir daí, a máquina ligada à internet (computador ou roteador) pode encapsular e desencapsular pacotes da camada rede, mas sempre tendo o PPPoE como protocolo de camada enlace.

Se por algum motivo a conexão de camada enlace precisar ser finalizada (o usuário não pagou a conta, ou está há algum tempo sem transmitir informações), a máquina do usuário (roteador ou computador) e o provedor de acesso trocam quadros de camada enlace PPPoE afim de fechar a conexão. O funcionamento detalhado do PPPoE está fora do escopo deste curso; todavia, é importante você saber que este protocolo é usado para conexões ADSL entre a máquina do cliente e o provedor de acesso. No caso de um computador diretamente conectado ao modem ADSL, a placa de rede deve falar PPPoE; no caso de um roteador, a porta WAN deve falr PPPoE, enquanto a porta LAN deve continuar falando Ethernet, que é o padrão das redes locais.

13.8. CONCLUSÃO

Enfim, terminamos este capítulo. Você deve estar sentindo-se um pouco cansado após essa leitura. Desta vez, analise bem a figura abaixo, olhe-a por alguns minutos e tente explicar a si mesmo, mentalmente, qual a função de cada equipamento, e porque ele está ali. Depois de analisá-la, leia o texto abaixo.

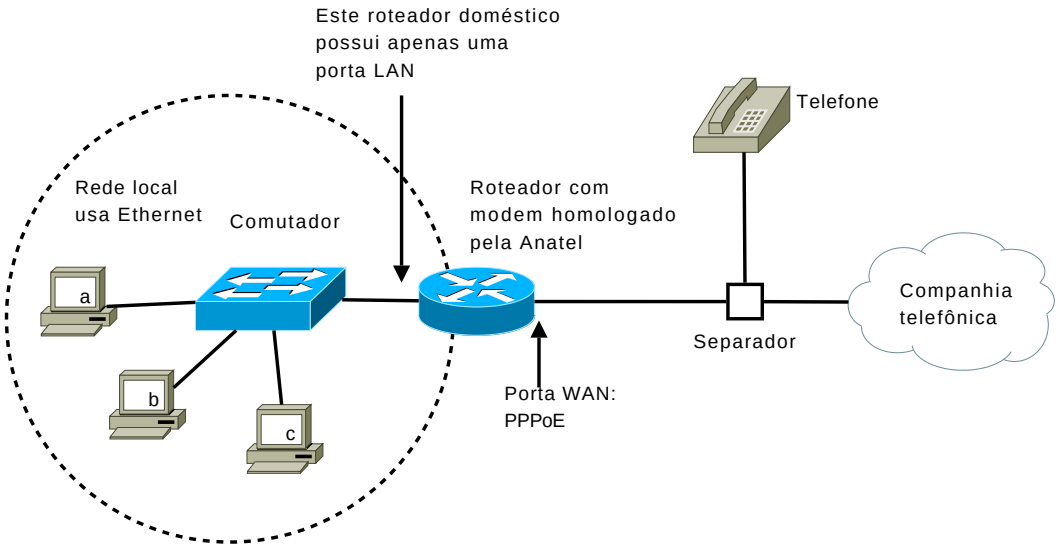

Figura 13.12. Ambiente doméstico ADSL.

Nesta figura, você tem uma rede local. Esta rede possui três computadores, `a`, `b` e `c`, e uma porta do roteador, a porta LAN, interligados por um comutador. As máquinas `a`, `b` e `c` são clientes DHCP. A porta LAN do roteador é um servidor DHCP. O protocolo de camada enlace usado na rede local é o Etherne, isto é, as máquinas usam o Ethernet para conversarem.

A porta WAN do roteador possui um modem acoplado, como mostra indicação na figura. Se esta porta não possuisse um modem ADSL, este seria um roteador comum, e precisaria estar conectado a um modem ADSL externo. Todavia, não é isso que acontece: a porta WAN tem um modem acoplado. Isso significa que a porta WAN é, na verdade, uma porta RJ-41, para linha telefônica. Esta porta WAN do roteador é cliente DHCP, e fala o protocolo PPPoE para comunicar-se com o provedor. Como existe um modem na porta WAN, os sinais elétricos enviados e recebidos por esta porta são analógicos. Se esse roteador não tivesse um modem na porta WAN, ele enviaria sinais elétricos, ainda usando o protocolo PPPoE.

A porta WAN do roteador é um modem, não esqueça disso. O modem envia sinais analógicos, assim como o telefone. A diferença está na frequência. O modem envia sinais acima de 4Khz. O telefone, desinteligente como é, envia sinais em todas as frequências. Entretanto, o separador "barra" os sinais vindos do telefone que estejam acima de 4Khz, deixando passar para a linha somente sinais abaixo disso. Os sinais que vem do modem, sempre acima de 4Khz, são postos na mesma linha telefônica.

Lembra-se qual o termo que define o ato de juntar sinais de várias frequências em um único meio físico? É multiplixação por divisão de frequência. Todos os sinais elétricos analógicos, de diferentes frequências, vão pelo mesmo cabo telefônico sem misturarem-se, pois frequências diferentes não se misturam. No outro lado, o separador da companhia telefônica vai demultiplexar os sinais, encaminhando sinais abaixo de 4Khz para os equipamentos que processam voz, e sinais acima disso para equipamentos que processam dados.

13.9. Exercícios

Exercício 13.1. Com base na figura exibida na conclusão, desenhe um ambiente ADSL que contenha:

- Um computador
- Um modem ADSL
- Um separador
- Um telefone

Esse ambiente não conterá roteador nem comutador.

Exercício 13.2. Ainda com base na figura exibida na conclusão, desenhe um ambiente ADSL que contenha:

- Um roteador doméstico com modem ADSL embutido, com três portas LAN
- Três computadores
- Um separador
- Um telefone

Esse ambiente não conterá um comutador separado.

Exercício 13.3. Marque a alternativa correta sobre modem ADSL:

a) A função do modem é servir de cliente DHCP do provedor de acesso.

b) A função do modem é servir de servidor DHCP para a rede local.

c) A função do modem é separar sinais abaixo de 4Hhz de sinais acima.

d) Nenhuma das alternativas.

Exercício 13.4. Como se chama o ato de transformar sinais elétricos digitais em sinais elétricos analógicos?

a) Multiplexar

b) Demultiplexar

c) Modular

d) Nenhuma das alternativas.

Exercício 13.5. Como se chama o equipamento que faz o que foi dito na questão anterior?

a) Roteador

b) Separador

c) Modem

d) Nenhuma das alternativas.

Exercício 13.6. Marque a alternativa que define **mais precisamente** a multiplexação explicada neste capítulo.

a) É o ato de transmitir por um mesmo meio físico dados e voz sobre pacotes de camada enlace.

b) É o ato de transmitir por um mesmo meio físico dados e voz.

c) É o ato de transmitir por um mesmo meio físico sinais de diferentes frequências.

d) Nenhuma das alternativas define multiplexação.

Parte V

Apêndices

Apêndice A

Repostas dos exercícios

Para exercícios dissertativos, as respostas oferecidas neste apêndice são apenas sugestões.

A.1. Capítulo 1

1. Rede de computadores é um conjunto de máquinas ligadas entre si, e que conseguem trocar informações.

2. Protocolo é um conjunto de regras para comunicação entre as máquinas. Um protocolo serve para que as máquinas saibam como devem comunicar-se entre si com educação. Sem um protocolo, ou ainda, se as máquinas usassem protocolos diferentes e incompatíveis entre si, a comunicação seria impossível.

3.

    a) Verdadeiro. É necessário um endereço físico para que as máquinas enviem pacotes umas para as outras; os pacotes usam o endereço físico.

    b) Falso. Isso nem sempre é verdade, pois em redes que usam um único enlace físico compartilhado entre todas as máquinas, o pacote de camada enlace chega em todas elas, porém só é interpretado pela máquina destinatária..

    c) Verdadeiro. Como o enlace físico é compartilhado, se as máquinas enviassem dados como bem entendessem, haviriam colisões e outros problemas. Por isso o protocolo: ele tem a função reguladora, pois dita que as máquinas devem ouvir antes de falar e falar somente quando o meio estiver disponível, parando de falar se for detectada uma colisão.

    d) Verdadeiro. A camada enlace recebe os dados pela enlace físico (ou seja, recebe os dados da camada física). Ao receber os dados, converte-os em um quadro de camada enlace, e analisa o endereço físico de destino. Se o endereço físico de destino foi igual ao endereço físico da máquina, então desencapsula-se o pacote d ecamada rede e envia para cima.

4. Colisão é a junção de informações de duas máquinas; o resultado da junção é ilegível, e não é possível interpretá-lo. Ocorre quando informações de duas máquinas trafegam pelo mesmo meio físico que tenha capacidade de transportar somente a informação de uma máquina por vez. Quando duas máquinas enviam pelo mesmo mesmo físico, há uma colisão.

5. LAN é rede local. WAN é rede de longa distância. LAN é sigla para Local Area Network. WAN é sigla de Wide Area Network. Um conjunto de computadores em um escritório, ou no andar de uma empresa, é LAN. A internet é WAN.

6.

a) Falso. Com apenas um enlace que esteja com internet, é possível conectar uma rede inteira à internet.

b) Verdadeiro. Muitas, para não dizer todas as redes usam protocolos da arquitetura TCP/IP.

c) Falso. Camada do usuário não existe, o que é existe é a camada Aplicação, que é usada pelo usuário. São cinco camadas: Aplicação, Transporte, Rede, Enlace e Física. Entretanto, esquemas mais antigos representam a arquitetura TCP/IP em quatro camadas, sendo as camadas Enlace e Física uma só. Apesar de representarmos a arquitetura TCP/IP hoje em dia em cinco camadas, o domínio desta arquitetura é principalmente nas três camadas superiores. As camadas Enlace e Física podem ser usadas com outros protocolos que não os da pilha TCP/IP.

d) Verdadeiro. Um exemplo de protocolo de camada enlace é o Ethernet.

7. Na máquina que envia os dados, funciona assim: a camada aplicação encapsula seus dados (ou seja, o datagrama) na camada inferior, ou seja, transporte. A camada transporte adiciona seus próprios dados ao datagrama, formando assim o segmento, e o envia para a camama inferior, que é a rede. A camada rede adiciona seus próprios dados ao segmento, formando assim um pacote, e o envia para a camada inferior, a enlace. A camada enlace adiciona seus próprios dados ao pacote, formando assim o quadro, e o envia para a camada física, que é o enlace físico no qual as máquinas da rede local estão conectadas.

   Na máquina que recebe os dados, acontece o seguinte: a camada enlace recebe os dados da camada física, e analisa o quadro. Se esta máquina for o destino do quadro, então retira as informaçõa da camada enlace e envia o pacote resultante para a camada superior, a rede. A camada rede analisa o pacote, retira as informações de camada rede dele e envia o segmento resultante para a camada transporte. A camada transporte analisa o segmento, retira os dados de camada transporte e envia o datagrama resultante para a camada aplicação.

   O ato de uma camada receber um pacote de cima e adicionar dados da própria camada se chama encapsular. O ato de a camada receber um pacote de baixo e retirar dados da própria camada se chama desencapsular.

8. Duas camadas de máquinas diferentes podem trocar informações adicionando informações (a máquina de origem) e lendo essas informações (a máquina de destino). Por exemplo, a camada transporte adiciona dados de camada transporte ao datagrama, e estes dados serão lidos somente pela camada transporte da máquina de destino.

A.2. CAPÍTULO 2

1. Sim. Apesar de a camada enlace aceitar a informação, a camada rede pode negar. por exemplo, supondo que a máquina na rede local envie um quadro cujo endereço físico seja broadcast, contudo o endereço lógico do pacote encapsulado seja um IP qualquer. Todas as máquinas da rede local receberão o quadro de camada enlace, mas apenas a máquina cujo endereço lógico de destino é o correto deixará o pacote de camada rede passar.

2. LAN é rede local. WAN é rede de longa distância.

3. Enlace LAN é o enlace físico usado em redes locais. Enlace WAN é o enlace físico usado em redes de longa distância.

4. Pacote é a informação da camada rede. Um pacote contém em seu interior um segmento de camada transporte. Já um quadro é a informação da camada enlace. um quadro contém em seu interior um pacote.

5. O endereçamento físico é necessário para conversar com uma máquina na rede local. Endereço lógico é necessário para conversar com uma máquina que não esteja na rede local. As camadas enlace das máquinas comunicam-se através do endereço físico. As camadas rede das máquinas comunicam-se através do endereço lógico. Os quadros usam endereço físico, enquanto os pacotes usam endereços lógicos.

6.

   a) Verdadeiro. Por exemplo, uma única máquina pode estar acessando ao mesmo tempo uma página da Web (ou seja, é cliente Web) e um servidor de email (é cliente de email).

   b) Verdadeiro. Uma máquina pode estar ao mesmo tempo acessando uma página Web de um servidor que esteja no Brasil, e outra página que esteja na China.

   c) Falso. Um servidor pode prover serviço, desde que configurado para isso, para qualquer máquina que solicite.

   d) Verdadeiro. Se uma aplicação é servidora de páginas Web, esta aplicação só oferecerá esse serviço à máquinas que possuam uma aplicação cliente Web. um servidor Web não pode oferecer informações, por exemplo, para clientes de email.

   e) Falso. Uma máquina pode ter, ao mesmo tempo e em execução, aplicativos clientes e servidores. Por exemplo, uma máquina servidora Web pode ser cliente de um servidor de emails.

   f) Falso. A localização do servidor não influi no funcionamento da aplicação servidora. As máquinas só precisam ter como acessá-lo.

7. Um cliente de email, como o Firefox; um cliente de emails, como o Thunderbird, ou o Kmail; um cliente de torrent, como o BitTorrent, ou o KTorrent; qualquer aplicação empresarial que usa um banco de dados, como aplicações de agendamento de viagens das companhias aéreas.

8. Transporte confiável de dados é o ato de duas máquinas trocarem informações de forma segura: isso significa que a informação não será corrompida no caminho, por exemplo. Se a informação for corrompida, a máquina de destino saberá disso e não encaminhará o erro adiante: em vez disso, ele pedirá para a máquina de origem para reenviar a informação. Um protocolo de camada transporte que oferece confiabilidade é o TCP.

9. Conexão de camada transporte é o ato de duas máquinas conversarem antes de começarem a enviar informações uma para a outra.

10.

    a) Verdadeiro. A máquina destinatária "sobe" com os dados.

    b) Falso. A camada transporte não verifica endereços. Quem faz isso é a camada enlace (endereço físico) e a camada rede (endereço lógico).

    c) Verdadeiro. Vídeo tolera pequenas perdas de pacotes.

A.3. CAPÍTULO 3

1. Porque J não se encontra na rede local. Máquinas só enxergam endereços físicos que estejam na rede local. Fora da rede local, só é possível se comunicar usando endereço lógico.

2. d)

3. Comutação de camada enlace é o ato de um aparelho no centro da rede ligar logicamente a máquina de origem à máquina de destino, de forma que a rede inteira não fique ocupada, somente os enlaces que se comunicam.

4. Roteamento é o ato de um aparelho ligar duas redes distintas (duas redes, e não duas máquinas). O roteamento encaminha pacotes de camada rede, deixando para trás quadros de camada enlace.

5.

    a) Falso.

    b) Verdadeiro.

    c) Falso. Não deixe a expressão "transporte" confundir você.

    d) Verdadeiro.

    e) Verdadeiro.

    f) Falso. Este endereço é um endereço IP versão 4, que funciona na camada rede. Ou seja, é um endereço lógico.

    g) Verdadeiro. Os sistemas operacionais de hoje geralmente já vêm com o protocolo Ethernet instalado; as placas de rede já vêm com um endereço MAC instalado, que é o endereço usado pelo protocolo Ethernet.

6. Backbone é o núvleo da rede, onde a informação passa de forma intensa. Por exemplo, em um edifício com uma rede local por andar, o Backbone é o conjunto de equipamentos que interligam essas redes.

7. c)

A.4. Capítulo 4

1. c). A resposta **a** está errada, pois, embora a informação esteja descendo a pilha de protocolos, ela não é encapsulada em um "embrulho" de camada física, simplesmente porque a placa de rede não adiciona nenhuma informação ao quadro de camada enlace.

2. EIA/TIA 568A.

3. d)

4. b)

5. b)

6. a, b). O padrão EIA/TIA 568B usa os fios 3 e 6 para enviar. Esses fios, na outra ponta do cabo, são os mesmos 1 e 2 do padrão 568A.

7. c, d) São os fios 3 e 6.

8.

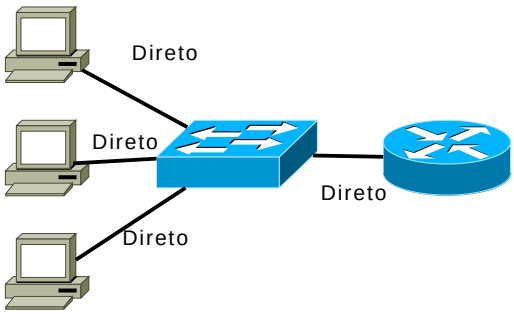

Figura A.1. Resposta do exercício 8.

9.

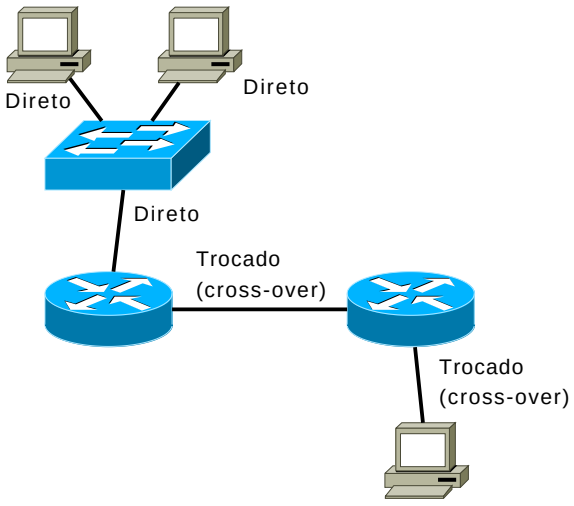

Figura A.2. Resposta do exercício 9.

10.

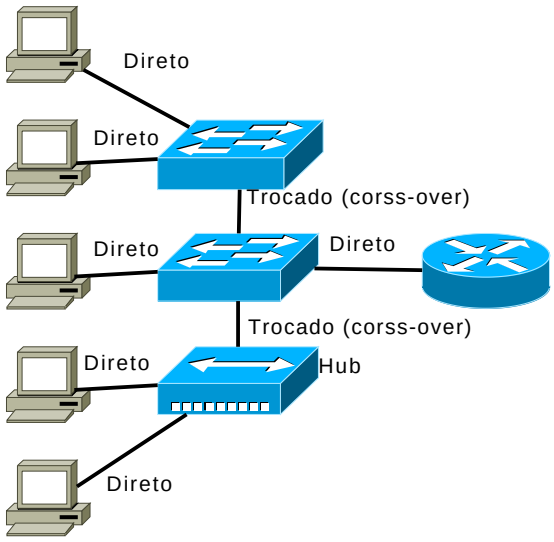

Figura A.3. Resposta do exercício 10.

A.5. CAPÍTULO 5

1. c). A alternativa **a** e a alternativa **e** estão erradas porque, para começo de conversa, DHCP opera com informações de camada rede. A alternativa **d** está errada porque DHCP configura dinamicamente essas informações nas máquinas clientes, não sendo necessário o redista ir de máquina em máquina configurá-las. Por fim, a alternativa **b** está errada pois tradução de nomes é função do DNS.

2. a). A alternativa **b** está incorreta, pois os nomes são traduzidos para IPs, que são endereços lógicos.

3. a, b, c, d, e, f, g). A alternativa **h** está incorreta, pois o IP do servidor é configurado estaticamente, de modo que ele não expira.

4. b, c, d). A letra **c** poderia não estar marcada, pois hoje em dia o servidor de DHCP é a porta de um roteador, que por conhecidência é o Gateway padrão.

5. b, d). As máquinas não precisam de um servidor DNS para se comunicarem, pois elas conseguem conversar usando apenas IPs em vez de nomes.

A.6. CAPÍTULO 6

1. a). A alternativa **b** está incorreta, pois embora `p2` seja uma porta do roteador, ela está fora da rede local e não é enxergada pelas máquinas: nunca poderá ser servidor de DHCP. O mais natural é `p2` ser cliente, afim de obter informações de camada rede do provedor de acesso.

2. c, d). A alternativa **a** está incorreta, pois se `a` e `b` estão na mesma rede local, não é necessário usa-se o Gateway padrão para isso. Em vez disso, o endereço físico de destino do quadro de camada enlace é o endereço de `b`. A alternativa `c` está incorreta pelos mesmos motivos.

3. a, b, d). A alternativa `c` está incorreta, pois embora o enderço físico do pacote deva ser a do Gateway padrão, o endereço lógico permanece o endereço da máquina fora da rede local. lembre-se que a camada enlace (endereço físico) só enxerga a rede local; já a camada rede enxerga além da rede loca.

4. b, c)

5. b, c). A alternativa `a` está incorreta porque a porta LAN do roteador irá prover serviço de camada rede às máquinas da rede local. A alternativa `d`, máscara dinâmica, não existe.

6. a). A alternativa `e` pode até estar correta porque o roteador já pode vir configurado para determinado tipo de conexão WAN. Mas isso depende do roteador.

7. a, c, e, f). A alternativa `d` está incorreta, pois roda-se um servidor DHCP para cada porta LAN. Este roteador tem três servidores DHCP nas portas LAN e provavelmente um servidor DHCP rodando na porta WAN.

8. a, c). A alternativa `b` está incorreta porque as portas LAN não são portas de roteador, mas sim de comutador. Temos uma única LAN. A alternativa `d` está incorreta porque, como trata-se de uma única LAN sendo ligada em suas portas, há necessidade de apenas um servidor DHCP.

9. b). A alternativa `c` se refere a um roteador "tradicional".

A.7. Capítulo 7

A.8. Capítulo 8

A.9. Capítulo 9

A.10. Capítulo 10

Apêndice B

Redes Legadas

A grande maioria dos cursos de rede hoje em dia citam alguns tipos de redes pré-históricas como se fossem a coisa mais importante do mundo. Não vou dizer que esses assuntos não são importantes. Sim, são importantes (o médico disse que não podemos contrariar quem fala isso)! Tudo no mundo é importante, informação nunca é demais, blá blá blá. É interessante ver alguém que saiba muito. Imagine só, um redista moderno conhecer de redes legadas! Um redista conhecedor de literatura latina! Um redista conhecedor do processo de combustão dos gases no estômago de uma baleia branca!

Porém, o mundo não vai acabar se isso não for ensinado em sala de aula. O redista conseguirá montar uma rede local, deixá-la funcionando e fará seu patrão feliz mesmo sem nunca ter ouvido falar em "topologia anel". Agora, o patrão não ficará muito feliz se o redista não souber como alocar o espaço de endereços IPs em sua empresa. Ou se não souber como configurar as informações de camada rede do servidor Linux mais importante do setor.

Se os cursos básicos de redes tivessem 4 mil horas de conteúdo, poderíamos falar sobre redes legadas, sistemas legados, como configurar o IP naquele sistema operacional de 20 anos atrás. Entretanto os cursos básicos de rede possuem uma carga horária limitada: portanto, somente o mais relevante, o mais relevante mesmo precisa ser dito. Se, e somente **se** sobrar tempo, algo que acho difícil, aí sim, devemos falar sobre redes legadas.

Essas redes não são mais usadas hoje. Hoje todo mundo (ou ao menos a grande maioria das empresas) usa redes locais Ethernet. Neste apêndice, estaremos explorando essas redes do passado.

B.1. As designações da topologia: física e lógica

Antes que você pergunte o que é topologia, saiba que existem duas designações para ela. A topologia física, aquela que malha na academia e pratica natação, e a lógica, aquela que lê livros e escreve poesias.

Brincadeiras de lado, topologia é...

Definição B.1. *Topologia: a maneira que as máquinas estão dipostas na rede. A forma que as máquinas estão arrumadas na rede.*

Por exemplo, você tem uma sala com quatro máquinas, e liga-as por meio de um cabo. Elas formam um quadrado. Abaixo, uma figura que representa as máquinas ligadas na sala, formando um quadrado:

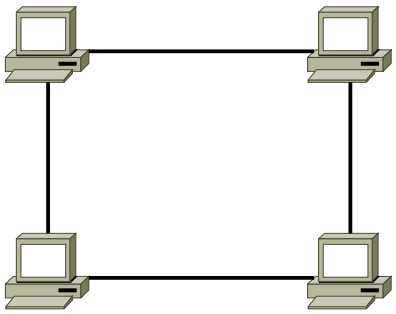

Figura B.1. Topologia do quadrado.

Assim, nós poderíamos chamá-la de "topologia do quadrado" ou "topologia em quadrado", porque é assim que as máquinas estão dispostas nessa rede. Além disso, é uma topologia física, pois as máquinas estão fisicamentes dispostas assim.

Uma topologia lógica é quando a rede funciona (logicamente) como se fosse uma topologia do quadrado. Em breve, esclareceremos este ponto, mas por agora, basta você saber que uma rede cuja topologia física seja um quadrado pode funcionar logicamente como um triângulo (expressão de espanto!).

Agora, uma verdade chocante: a topologia do quadrado não existe. Foi apenas uma mentirinha para este explicação. Contudo, existe uma topologia chamada "topologia em anel", que se parece bastante com a do quadrado, só que é assim:

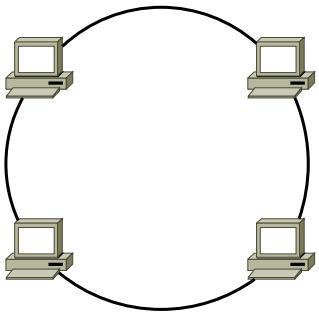

Figura B.2. Topologia em anel: essa existe.

Uma verdade mais chocante ainda: a topologia do quadrado, na figura 1.1, é na verdade a topologia em anel. São a mesma coisa. "Mas um quadrado não tem forma de anel", questiona você, catedrático como é no assunto de quadrados e anéis. Entretanto, a topologia é chamada de anel não porque os cabos estão em forma de anel, etc, e sim, porque a topologia age fisicamente de uma forma que convencionou-se chamar de anel. E também, logicamente.

Vamos falar de topologias a partir de agora.

B.2. ANEL

Na figura abaixo, todas as três redes possuem topologia lógica em anel, mesmo que a topologia física não seja (ou não se pareça com um) anel.

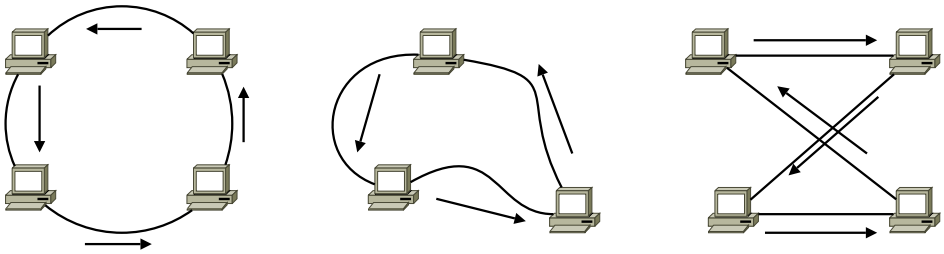

Figura B.3. Topologia lógica em anel.

Nos referimos à topologia lógica simplesmente por topologia. Assim, se alguém der um grito no restaurante "topologia em anel!", pense na topologia lógica. A topologia física não é tão importante, pois é a topologia lógica que dita as regras do funcionamento da rede.

A topologia (lógica) em anel funciona resumidamente da seguinte forma: uma máquina fala, e em seguida, a próxima máquina fala, assim por diante, em um ciclo infinito, conforme mostra a figura 1.3. As setas indicam a direção da informação.

Cada uma das máquinas tem uma cenoura, digo, um token, que é como se fosse uma permissão para que esta máquina fale na rede. Ela pode falar com quem quiser enquanto estiver com essa permissão. Essa permissão dura apenas um período muito ínfimo de tempo para cada máquina. Quando o tempo acaba, esta máquina passa o token para a seguinte, que usa-o para falar com quem quiser na rede. No fim, todo mundo tem permissão para falar, é bem democrático. Além disso, não há colisões nesse tipo de rede, pois apenas uma máquina fala por vez (a que está com o token, thanan!).

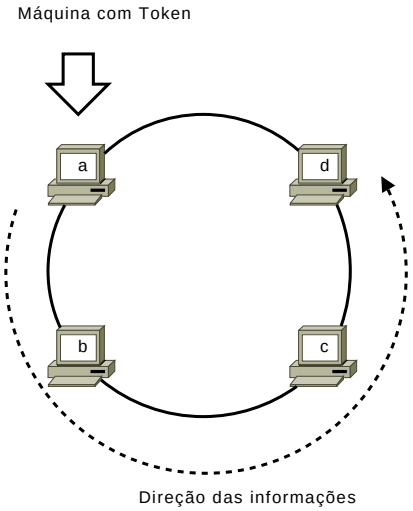

Figura B.4. Esquema da comunicação em Anel.

As informações nesse tipo de rede seguem apenas uma direção. Assim, se, por exemplo, a máquina `a` quer falar com a máquina `d`, a informação vai percorrer todo o percurso indicado. Além disso, `a` não vai receber resposta imediata de `d`, pois `d` não tem permissão para falar. `a` terá de esperar o token passar por `b` e `c`, para que enfim `d` fossa responder. Não se preocupe: o token passa por todo o anel milhares de vezes por segundo.

Está tudo maravilhoso, não é mesmo? Mas nem tudo é perfeito. Com essa democratização toda, algumas máquinas vão alocar o token pelo tempo normal mesmo que não tenha nada a transmitir. Ou seja, se apenas a máquina `a` estiver transmitindo a informação na rede, a rede perderá 25% do seu tempo sem fazer nada, pois o token estará em máquinas que não têm nada a falar.

NOTA B.2. A topologia em Anel é também chamada de "Token Ring".

B.3. BARRA

A topologia em barra funciona da seguinte forma:

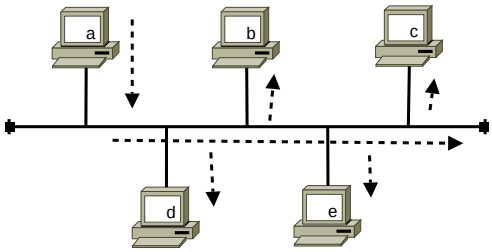

Figura B.5. Topologia em barra.

Você já estou sobre isso no capítulo 1. Não se lembra? Leia novamente.

Na figura, quando qualquer das máquinas fala, todas escutam, mas somente a verdadeira destinatária recebe. Pode haver colisão nessa topologia se duas máquinas falam mais ou menos ao mesmo tempo. Nesta topologia entra em funcionamento o protocolo CSMA/CD, para minimizar a ocorrência de colisões ou os prejuízos advindos delas.

NOTA B.3. Redes em barra usavam tipicamente cabos coaxiais.

B.4. TOPOLOGIA FÍSICA EM ESTRELA

Pode acontecer de, fisicamente, as máquinas da rede estarem conectadas a um único dispositivo central, seja ele um repetidor (hub) ou comutador (switch), conforme a figura abaixo:

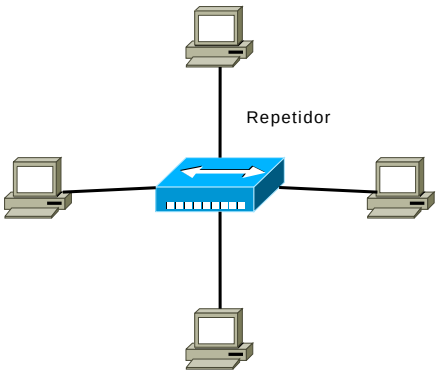

Figura B.6. Topologia física em estrela.

Na figura, temos um repetidor. Embora as máquinas estejam fisicamente dispostas conforme na figura, ou seja, topologia física em estrela, o funcionamento não é em estrela. Ou seja, a topologia lógica é em **barra**. Por quê? Ora, porque um repetidor não faz comutação de quadros, e sim, age como se fosse um único enlace compartilhado, tal como a topologia barra. Se você está com dificuldades para entender isso, releia o capítulo 1.

Hoje em dia, nas redes contemporâneas, usamos topologias física e lógica em estrela, isto é, o funcionamento é em estrela: as máquinas enviam pacotes para um dispositivo central e este envia somente para o destinatário: o comutador age como esse dispositivo.

Comentário B.4. Podemos resumir o que foi dito da seguinte forma: uma rede em que as máquinas são ligadas a um dispositivo central possui topologia física em estrela. Se este dispositivo é um repetidor, a topologia lógica é barra, pois a rede funciona da mesma forma que a topologia física barra. Se o dispositivo é um comutador, a topologia lógica é estrela.

Observe figura abaixo:

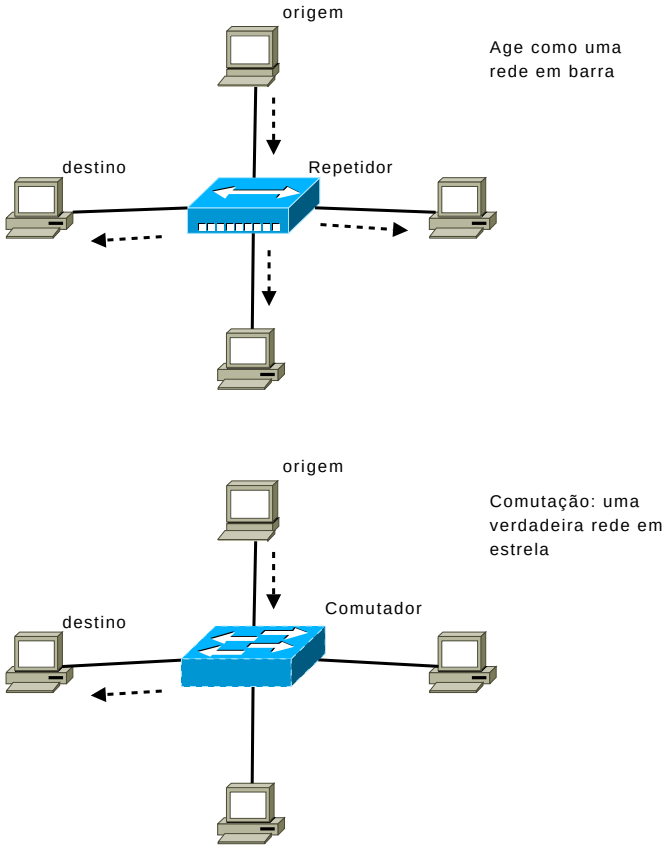

Figura B.7. Topologias lógicas em barra e estrela, respectivamente.

B.5. Topologia física em estrela, lógica em anel

Uma rede em anel pode assumir uma topologia física em estrela, ou seja, com todas as máquinas conectadas a um dispositivo central na rede. Porém, ao contrário do que acontece em redes Ethernet, este dispositivo não é nem um repetidor, nem um computador. É um MAU, uma Unidade de Acesso de Mídia (Media Access Unit), conforme ilustra a figura abaixo.

Figura B.8. MAU: Media Access Unit

Este dispotivo centraliza o gerenciamento do Token (ou permissão) em um único ponto da rede. Tudo que as máquinas precisam é estarem conectarem

B.6. TOPOLOGIA HÍBRIDA

"Híbrida" é uma palavra que significa mista. Topologia híbrida é igual a topologia mista, pegou? Não é uma topologia, e sim, uma mistura de topologias, como mostra a figura abaixo:

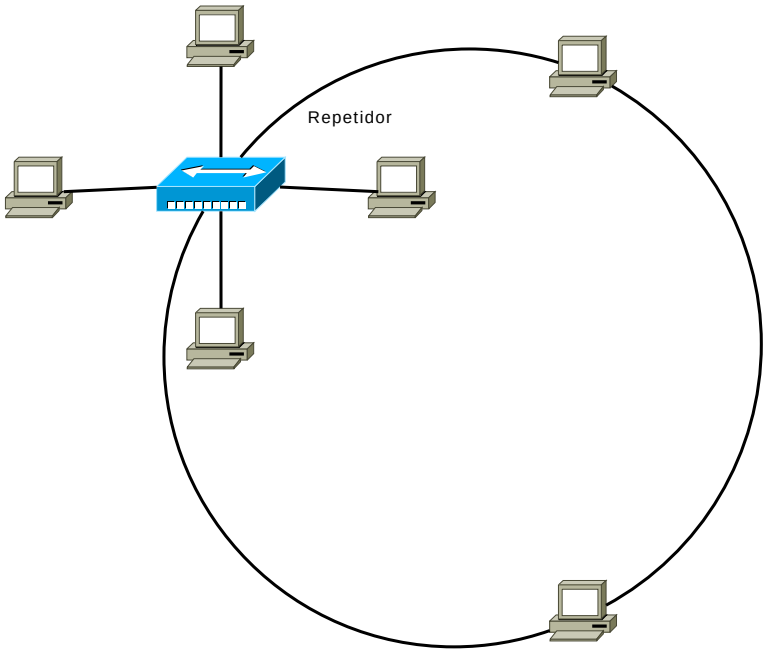

Figura B.9. Topologia híbrida anel-estrela.

Bibliografia

[Bruno & Kim] Bruno, A.; Kim, J. *CCDA Guia de Certificação do Exame - O Guia oficial de certificação para o exame DCN #640-441*, Alta Books (Cisco Press), 2003, Brasil.

[CESPE Anatel 2008] CESPE UnB, Prova específica para Concurso Público ANATEL, 2008, Cargo 10: Analista Administrativo - Área: Tecnologia da Informação - Especialidade: Redes e Segurança. Data de aplicação: 8/3/2009.

[CESPE Serpro 2008] CESPE UnB, Prova específica para Concurso Público SERPRO, 2008, Cargo 16: Analista - Especialização: Redes. Data de aplicação: 7/12/2008.

[Davidson, Peters] Davidson, J.; Peters, J.; Bhatia, M.; Kalidindi, S.; Mukherjee, S. *Fundamentos de VoIP - Uma abordagem sistêmica para a compreensão dos fundamentos de Voz sobre IP*, 2ª edição. Bookman (Cisco Press), 2008, Brasil.

[IPv6] NIC.br; http://ipv6.br, acessado em 11/06/2009.

[Kurose & Ross] Kurose, J. F.; Ross, K. W. *Redes de computadores e a Internet - Uma abordagem top-down*, 3ª edição. Pearson Education do Brasil, 2006, Brasil.

[Osborne & Simha] Osborne, E.; Simha, A. *Engenharia de tráfego com MPLS - Projeto, configuração e gerenciamento do MPLS TE para otimização de desempenho de rede*. Editora Campus (Cisco Press), 2003, Brasil.

[RFC 768] IETF; *User Datagram Protocol*, http://www.ietf.org/rfc/rfc0768.txt, acessado em 11/06/2009.

[RFC 791] IETF; *Internet Protocol*, http://www.ietf.org/rfc/rfc0791.txt, acessado em 11/06/2009.

[RFC 793] IETF; *Transmission Control Protocol*, http://www.ietf.org/rfc/rfc0793.txt, acessado em 11/06/2009.

[Tate, Lucchese & Moore] Tate, J.; Lucchese, F.; Moore, R.. *Introduction to Storage Area Networks*. Redbooks (IBM), 2006, EUA.

[Wikipedia IPv4] Wikipedia; *http://en.wikipedia.org/wiki/IPv4*, acessado em 30/06/2009.

[Wikipedia IPv4Protocols] Wikipedia; http://en.wikipedia.org/wiki/List_of_IP_protocol_numbers, acessado em 03/07/2009