SERVIÇOS DE REDES DE COMPUTADORES

UNIDADE 1 - COMO FUNCIONA UMA REDE IP?

Aline Izida

Introdução

A interligação entre redes tem o objetivo de integrar um sistema de comunicações. Para os usuários, parece uma enorme e única rede, a internet. No entanto, isso é uma ilusão, pois, na verdade, os *softwares* que implementam os protocolos escondem os detalhes. E mesmo que você saiba que existem vários dispositivos físicos conectados, seja por cabos ou pelo ar, parece que os *softwares* fazem tudo parecer uma única rede. Você já parou para refletir sobre isso?

Você já deve ter ouvido sobre endereço IP, mesmo antes de começar a estudar TI, no entanto, nunca soube o que significava? Realmente, não é algo trivial entender o papel de protocolos em uma rede de computadores, quanto mais na rede mundial de computadores. Nesta unidade, você poderá compreender o funcionamento básico do IP e porque ele é importante para que você navegue na internet.

Além disso, você já parou para pensar como os dados são transportados pela rede? Nesta unidade, você também verá o protocolo TCP, que está na base do entendimento sobre como os dados são transportados na internet. Calma, terão mais conceitos envolvidos! Mas, agora, você dará início a essa jornada de conhecimento.

Por fim, vamos fechar a unidade explanando sobre autenticação, um meio pelo qual muitas informações e dados podem ser assegurados diante dos constantes ataques maliciosos na internet. Ao final, é esperado que você tenha agregado um pouco mais de conhecimento sobre tecnologias que trabalham para a rede mundial de computadores funcionar. Vamos começar?

1.1 Redes TCP/IP – Protocolo IP

O *Internet Protocol* (IP) ou Protocolo de Internet é um dos protocolos mais importantes da internet. Antes de entendermos o que é o IP em específico, vamos analisar a figura a seguir, em que podemos fazer uma analogia com a comunicação humana para entendermos o que é um protocolo. Para que possamos nos comunicar, precisamos de uma pessoa, que é uma emissora da informação, e outra ou outras pessoas que irão receber a informação.

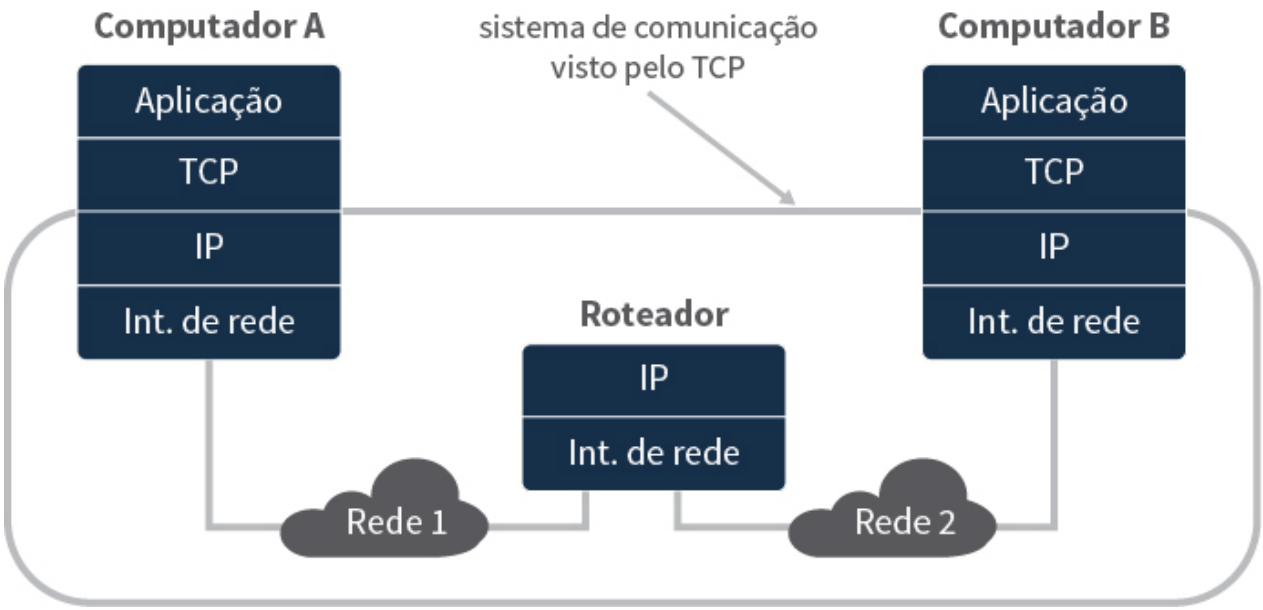


Figura 1 - Exemplo da base de um sistema de comunicação que precisa de um receptor, um emissor, um meio de transmissão e um protocolo.

Fonte: Elaborado pela autora, 2019.

Uma Rede de Computadores e Dispositivos Eletrônicos (RCDE) funciona como uma rede de comunicação humana, pois, se nós utilizamos uma língua para nos comunicar, uma RCDE utiliza o que chamamos de protocolo. No caso de uma língua, o meio de transmissão é o ar, já de uma RCDE pode ser desde cabos até o ar. É claro que, assim como a comunicação humana, existem regras e uma complexidade nessa comunicação. Por exemplo, se mandamos um "oi" para outra pessoa, devemos esperar outro cumprimento na mesma língua para que a conversa prossiga ou mesmo seja iniciada. Assim como existem diversas regras, que demandam conhecimentos, e possibilitam essa comunicação de forma coerente, a fim de transmitir uma informação.

De acordo com Kurose e Ross (2013, p. 7), "um protocolo define o formato e a ordem das mensagens trocadas entre duas ou mais entidades comunicantes, bem como as ações realizadas na transmissão e/ou no recebimento de uma mensagem ou outro evento". Uma RCDE faz uso dos protocolos constantemente, existindo, assim, diversos deles para realizar diferentes tarefas, desde as mais simples até as mais complexas.

1.1.1 Endereço IP: conceito

Cada camada do modelo OSI ou TCP-IP tem um protocolo desempenhando uma determinada tarefa, contudo, existem três categorias que abarcam esses protocolos, são elas: protocolos de aplicação, protocolos de transporte e protocolos de rede. O IP se encaixa na categoria de protocolo de rede. Nessa categoria, os protocolos operam nas camadas físicas, de enlace de dados e de rede, no caso do modelo OSI e nas camadas de acesso à rede e internet, no caso do modelo TCP-IP, sendo responsáveis por tarefas que transmitem informações sobre endereçamento e roteamento, verificação de erro e requisições de retransmissão (COMER, 2016).

O IP é responsável por identificar seu dispositivo eletrônico (que também inclui os computadores) ou mesmo um *site* conectado a uma rede (tal como a de internet), de modo que essa identificação seja única e, por isso, chamamos de endereço IP. O endereço IP conta com duas versões, IPv4 e IPv6. O IPv4 é composto por uma sequência de números, divididos em quatro blocos de 8 *bits* (separados por um ponto final), que são representados por 32 *bits* ou 4 *bytes*, por exemplo: 189.58.55.99.

O IPv6 é formado por 128 *bits*, permitindo utilizar 2^{128} endereços, equivalente a 79 octilhões de vezes a mais a quantidade de endereços IPv4 e, portanto, muito mais endereços que o IPv4, que suporta 2^{32} endereços e que, por sua vez, foi idealizado quando não se imaginava que a rede mundial de computadores teria tantos usuários conectados a ela através de variados dispositivos eletrônicos (KUROSE; ROSS, 2013).

Com o IPv6 resolvemos o problema de limitação. O IPv6 é representado por números em hexadecimal, seu formato conta com 32 caracteres organizados em oito quartetos de 16 *bits* separados por dois pontos, por exemplo, 8888:9999:AAAA:BBBB:CCCC:DDDD:EEEE:FFFF. Um exemplo de IPv6 válido na internet seria 2001:0DB8:AD1F:25E2:CAFE:CADE:F0CA:84C1. Na notação hexadecimal, cada caractere possui 4 *bits*, isto é, 16 combinações, o que nos faz ter números de 0 a 9 e também caracteres de A a F, representando os números de 10 a 15, respectivamente (COMER, 2016).

VOCÊ SABIA?

Os 128 *bits* que compõem o endereço IPv6 permitem que apenas ele seja utilizado na internet, no entanto, os *softwares* e *hardwares* que possibilitam a comunicação na rede não são totalmente compatíveis com essa versão do protocolo IP. Logo, tanto o padrão IPv4 quanto o IPv6 devem coexistir.

Cada provedor de serviço de internet (do inglês *Internet Service Provider* – ISP) tem uma cota de endereços IP e quem distribui essas cotas e julga conflitos é uma entidade chamada *Internet Corporation for Assigned Names and Numbers* (ICANN). Assim, o endereço IP tem a função de rotear e entregar pacotes dentro da rede. Ele pode ser público ou privado, pode ser dinâmico, isto é, atribuído temporariamente a cada conexão a uma rede, ou pode ser estático, isto é, fixo. A capacidade de identificar um dispositivo eletrônico em uma rede contribui para que haja comunicação e permite identificar a origem de atividades de usuários que, inclusive, podem estar ligadas a crimes.

1.1.2 Endereço IP: funcionamento

A primeira parte do endereço IP identifica uma rede específica e a segunda identifica o *host* conectado a esta rede. Um *host* é qualquer dispositivo eletrônico conectado a uma rede, desde computadores, *smartphones*, impressoras, TVs, até um roteador. As partes que identificam a rede e o *host* variam de acordo com a classe a que este endereço pertence. Clique em cada uma classes e saiba mais.

Classe A

0.0.0.0 a 127.255.255.255 (por exemplo: 17.2.23.12). Permite até 128 redes, cada uma com até 16.777.216 dispositivos eletrônicos conectados.

Classe B

128.0.0.0 a 191.255.255.255 (por exemplo, 185.4.11.48). Permite até 16.384 redes, cada uma com até 65.536 dispositivos eletrônicos conectados.

Classe C

192.0.0.0 a 223.255.255.255 (por exemplo, 192.14.11.10). Permite até 2.097.152 redes, cada uma com até 256 dispositivos eletrônicos conectados.

A figura abaixo demonstra essa separação de classes.

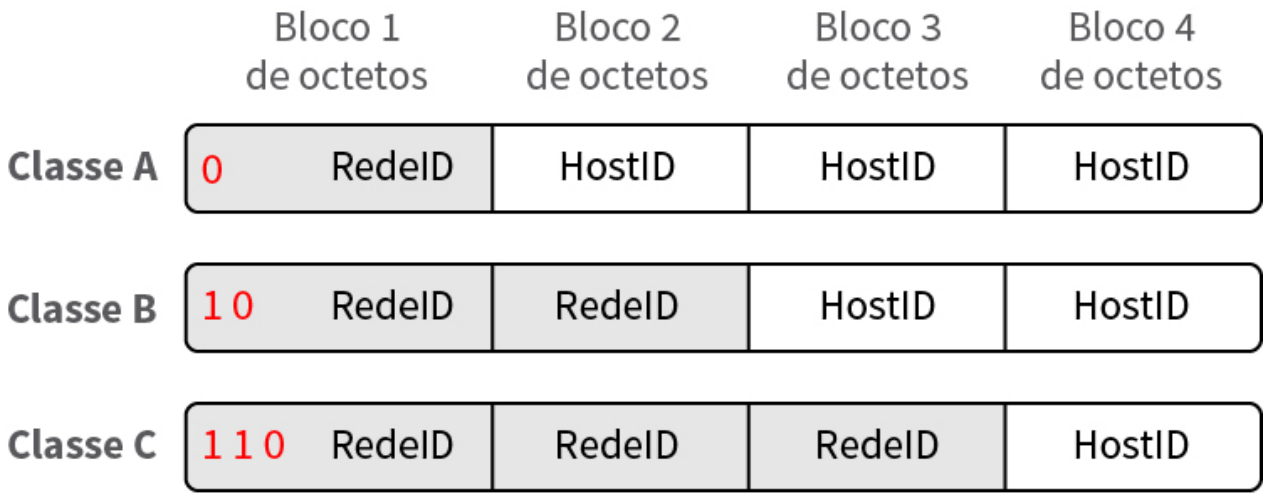


Figura 2 - Representação da separação de blocos de 8 bits, que compõem os 32 bits de um endereço IP, e definem as classes A, B e C de acordo com a utilização dos blocos para definição de identificador de rede e blocos para definição de identificador de hosts.
Fonte: Elaborado pela autora, 2019.

Devido à definição dos intervalos que separam as classes, a classe A inicia com um *bit* 0, a classe B com um *bit* 1 e um 0 e a classe C com dois *bits* 1 seguido de um *bit* 0. Cada bloco tem 8 *bits*, que podem variar entre os valores em decimal 0 (caso todos *bits* sejam 0) e 255 (caso todos os *bits* sejam 1). Essa divisão em octetos facilita a organização da rede, como o endereço de uma casa ou prédio, que é composto de uma rua e um número. Assim, a rede pode ser comparada ao bairro, enquanto o número da casa pode ser comparado ao *host* dentro da rede. A figura a seguir traz exemplos de números de IP em binário e os decimais correspondentes (KUROSE; ROSS, 2013).

Número binário de 32 bits	Notação decimal pontilhada equivalente
10000001 00110100 00000110 00000000	129.52.6.0
11000000 00000101 00110000 00000011	192.5.48.3
00001010 00000010 00000000 00100101	10.2.0.37
10000000 00001010 00000010 00000011	128.10.2.3
10000000 10000000 11111111 00000000	128.128.255.0

Figura 3 - Exemplos de números binários de 32 bits e seus equivalentes na notação decimal usados no IPv4.
Fonte: COMER, 2016, p. 306.

Existe também a classe D (224.0.0.0 até 239.255.255.255) usada para comunicação *multicast* (para um grupo determinado de dispositivos), e a classe E (240.0.0.0 até 255.255.255.255), reservada para futuras aplicações ou experimentos. O endereço 127.0.0.1, utilizado para se referir ao próprio *host*, é chamado de *localhost*. Em geral, quando um endereço começa com 127 indica uma rede reservada para testes. Além disso, o endereço 255.255.255.255 é utilizado como *broadcast*, isto é, para propagar mensagens para todos os *hosts* de forma simultânea. Ainda temos o IP 169.254.0.0, que é o IP que o Sistema Operacional atribui ao computador quando ele não está conseguindo receber um IP válido (não consegue acessar a rede), sendo, por exemplo, um indicador de que o modem ou roteador não está funcionando corretamente.

Podemos verificar que a classe A deve ser utilizada quando desejamos poucas redes, mas temos uma quantidade grande de *hosts* a serem conectados a elas. Já os IPs da classe B podem ser utilizados quando desejamos criar quantidades semelhantes de redes e de *hosts*. Enquanto os IPs de classe C podem ser utilizados para quando existe a necessidade de criar muitas redes com poucos *hosts*.

VOCÊ QUER VER?

Os fundamentos do endereçamento IP podem ser estudados de uma forma descontraída no canal *Curso em Vídeo*, no Youtube. O conteúdo está dividido em duas partes (GUANABARA; JÚNIOR, 2019):

parte 1: https://youtu.be/q65kHIvtWxg

parte 2: https://youtu.be/ee5htpGdWHY

Alguns intervalos de endereços IP das classes A, B e C são privados, usados em redes internas, significando que não podem ser utilizados pela internet. Clique nos itens e veja quais são os intervalos:

Classe A: 10.0.0.0 a 10.255.255.255.
Classe B: 172.16.0.0 a 172.31.255.255.
Classe C: 192.168.0.0 a 192.168.255.255.

Notoriamente, você pode verificar que pequenas empresas ou instituições iniciam suas redes com 192.168, já que geralmente não precisam de muitos *hosts* em cada rede (Classe C).
Além disso, cada rede IP pode ser dividida em sub-redes para vários fins de administração facilitada, melhor organização e questões de segurança. Você pode saber a divisão de classes porque decorou, mas o computador sabe qual é a classe do endereço IP por causa da máscara de rede. Ela é usada para determinar onde termina o endereço da rede e onde inicia o endereço do *host*, principalmente em sub-redes. No entanto, quando não há divisão por sub-redes, utilizamos as máscaras *default* para cada classe. Assim, se a máscara de rede é 255.0.0.0, o computador entende que a classe na qual está inserido é a A, se a máscara de rede é 255.255.0.0 pertence a classe B e se a máscara de rede é 255.255.255.0 pertence a classe C (MORAES, 2010).

CASO

Vamos supor um endereço classe C válido na internet como 200.10.1.x. É preciso dividir esse endereço classe C em duas sub-redes, uma para o setor de RH e outra para o restante de usuários da empresa. Como resolver esse problema?

Para isso, precisamos criar uma máscara de rede que permita dividir a rede em duas. A máscara *default* para uma classe C em binário é:

11111111. 1111111. 1111111. 00000000

255.255.255.0

Para que possamos dividir o endereço classe C da empresa em dois, vamos precisar adicionar 1 *bit* à máscara, uma vez que a regra usada é:

$N = 2^X$

Em que N = número de sub-redes e X = número de *bits* que vamos adicionar à máscara.

Portanto, a máscara de sub-rede vai ficar com 1 *bit* a mais, ou seja:

11111111.11111111.11111111.10000000 255.255.255.128

A primeira sub-rede vai começar no endereço 200.10.1.0 (Id da rede), tendo como endereços válidos 200.10.1.1 a 200.10.1.126. O *broadcast* será o último endereço da sub-rede 200.10.1.127.

A segunda sub-rede vai começar no endereço 200.10.1.128 (Id da rede), tendo como endereços válidos 200.10.1.129 a 200.10.1.254. O *broadcast* será o último endereço da sub-rede 200.10.1.255.

A importância do IP se dá ao passo que ele é o centro da comunicação através da internet, porque todos os aplicativos utilizam o IP e ele funciona sobre todas as tecnologias de redes. A internet parece uma enorme e única rede por causa dos *softwares* que implementam os protocolos e, para isso, todos os computadores e dispositivos devem usar um esquema de endereçamento uniforme, além de único. De acordo com Comer (2016), para garantir isso, o IP define um esquema de endereçamento independente dos endereços MAC. A diferença é que enquanto os endereços MAC são usados como destinos em uma LAN, os endereços IP são utilizados como destinos na internet. Desse modo, para enviar um pacote pela internet, um *software* que implementa o protocolo IP utiliza o endereço IP para transportar pacotes entre emissor (origem) e receptor (destino).

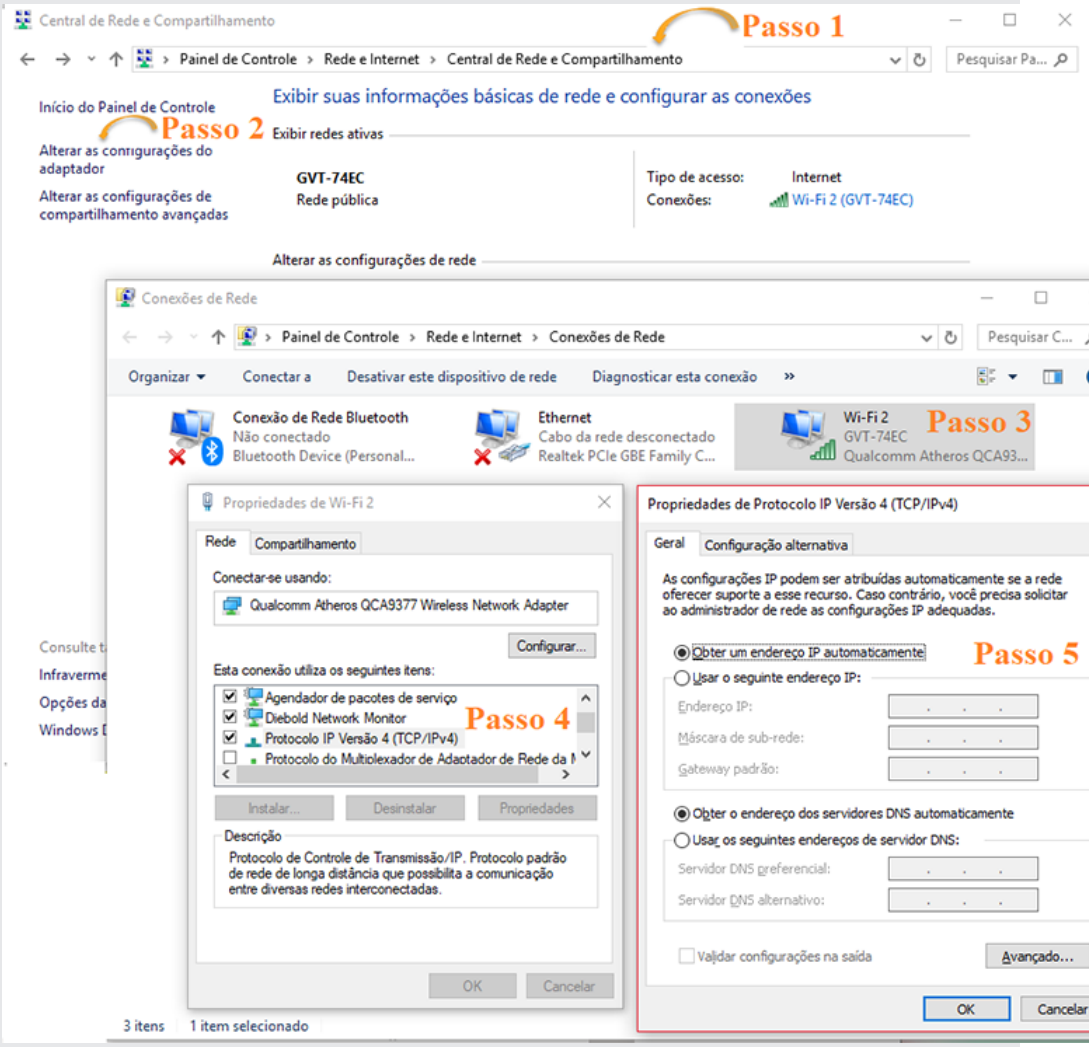
Em suma, os aplicativos se comunicam através de endereços IP. Portanto, quem fornece os endereços IP são os *softwares* que os implementam e não a rede. E isso acontece sem a necessidade de os aplicativos saberem o tipo de *hardware* de rede ou endereços MAC que estão sendo usados pelos aplicativos de origem e destino.

VAMOS PRATICAR?

Vamos supor um IP Classe C – 192.168.0.1. Como você iria distribuir IPs para os dispositivos na sua casa de forma manual? Primeiro, você precisa saber que os três primeiros blocos de octetos representam a rede a qual o seu computador está inserido, portanto, não podem ser mudados. O último bloco de octeto representa os *hosts*, logo, é este que você pode mudar para endereçar cada dispositivo em sua residência. Se há mais dois dispositivos, você irá atribuir os IPs de forma sequencial, sendo 192.168.0.2 e 192.168.0.3. Mas esse IP corresponde a uma rede privada, então, como você vai conseguir acessar à internet? Simples, basta utilizar um modem e um roteador *Wi-Fi* que irão receber a conexão e possibilitar o compartilhamento de internet com os dispositivos em sua casa. Esse modem é um dispositivo, logo, terá um IP que possibilitará a comunicação com um servidor (outro dispositivo) de internet.

Você pode verificar qual é o seu IP na internet acessando o seguinte *site* (2019): https://www.meuip.com.br/.No Sistema Operacional Windows, você pode verificar seu IP de rede privada (local) ao abrir o aplicativo Prompt de Comando e digitar "ipconfig". Você verá seu endereço de IPv4, IPv6, além da máscara de rede e do *gateway*.

Os IPs da sua casa provavelmente são distribuídos automaticamente. Para verificar essa configuração, siga os passos de acordo com a figura:



No passo 1, acesse Painel de Controle > Rede e Internet > Conexões de Rede. No passo 2, clique em "Alterar as configurações do adaptador". No passo 3, clique com o botão direito do mouse na sua *Wi-Fi*. No passo 4, selecione a opção "Protocolo IP Versão 4 (TCP/IPv4)" e clique no botão Propriedades, logo abaixo. No passo 5, você pode verificar que a configuração está para obter o endereço de IP automaticamente e você tem a opção de marcar "Usar o seguinte endereço IP" e então configurar manualmente.

No próximo tópico, vamos abordar o protocolo TCP que, juntamente com o IP, são fundamentais para o funcionamento da comunicação na rede mundial de computadores, a internet.

1.2 Redes TCP/IP – Protocolo TCP

O *Transmission Control Protocol* (TCP) é o principal protocolo de transporte utilizado na internet. Diferente do protocolo UDP, o TCP promove o transporte e a entrega confiável de dados. O protocolo TCP desempenha uma tarefa incrível, pois utiliza um serviço de datagramas, que são mensagens enviadas sem conexão e sem confirmação oferecidas pelo protocolo IP, e promove a entrega de dados confiável para programas de aplicação. Isso significa que o TCP pega mensagens não confiáveis e transporta-as de forma confiável até o destino. Para promover essa entrega confiável, o TCP compensa perdas, atrasos, duplicações, pacotes fora de ordem, tudo isso sem sobrecarregar a rede e os roteadores. Assim, o TCP define explicitamente o estabelecimento da conexão, a transferência de dados e o fechamento para oferecer um serviço orientado à conexão (KUROSE; ROSS, 2013).
Contudo, existe a nomenclatura TCP-IP, que se trata, na verdade, de uma pilha de protocolos de comunicação de computadores e dispositivos em rede, que recebe esse nome devido aos seus dois principais protocolos. O TCP-IP é, portanto, como um modelo, semelhante ao modelo OSI, em que cada camada é responsável por determinadas tarefas. Cada camada oferece uma série de serviços para a camada superior. A figura a seguir mostra as camadas do modelo TCP-IP.

Figura 4 - Modelo TCP-IP de pilha de protocolos, em que cada camada fornece serviços para a camada superior, destacando a camada de Transporte, onde o protocolo TCP atua.
Fonte: Elaborado pela autora, 2019.

O agente responsável pela camada de aplicação é o aplicativo, enquanto pelas camadas de transporte e de rede é o sistema operacional e o adaptador de rede é o agente responsável pela camada de interface. É importante entender que o TCP trabalha na internet e, por isso, em cima do modelo TCP-IP, que possui quatro camadas e trabalha com diferentes protocolos em cada uma delas. No entanto, o foco aqui é descrever os serviços e o funcionamento do protocolo TCP que trabalha na camada de Transporte. Assim, embora a base do modelo TCP-IP seja necessária, nosso objetivo, neste momento, é entender quais são as funções deste importante protocolo que é o TCP.

1.2.1 O serviço oferecido pelo TCP

O TCP é um protocolo que oferece um transporte de dados com vários benefícios para o bom funcionamento de uma rede de computadores e dispositivos na internet. Para entender como isso é possível, é importante entender as seguintes características do serviço oferecido pelo TCP às aplicações, segundo Comer (2016), clique na interação a seguir.

Orientado à conexão: primeiramente, o aplicativo deve solicitar uma conexão com o destino para depois utilizá-la para transferência de dados. Origem e destino devem se apresentar.

Comunicação ponto a ponto: cada conexão TCP tem especificamente dois pontos finais, isto é, *unicast*, entre uma única origem e um único destino.

Confiabilidade completa: há a garantia de que os dados que serão enviados através da conexão sejam entregues com integridade e em ordem.

Comunicação nos dois sentidos (*full-duplex*): permite a comunicação dos dados fluindo em qualquer direção, tanto da origem para o destino como o contrário, e que os dados sejam enviados a qualquer momento pelas aplicações.

Interface de fluxo (*stream*): fornece uma interface de fluxo na qual um aplicativo envia uma sequência contínua de octetos através de uma conexão. Não agrupa dados em registros ou mensagens e não garante a entrega dos dados em tamanhos iguais aos que foram enviados pelo aplicativo de origem.

Início de conexão confiável: o início da conexão das duas aplicações é confiável.

Finalização de conexão suave: antes de terminar a conexão de fato, há a garantia de que todos os dados tenham sido entregues e que ambos os lados concordaram em encerrar a conexão.

Portanto, o TCP deve oferecer um serviço fim a fim (entre um aplicativo em um computador e outro aplicativo em outro computador) confiável e eficiente. Para isso, deve tratar duplicatas, entregas fora de ordem, lidar com pacotes perdidos, evitar repetições, evitar inundação de dados e evitar congestionamento. Esses problemas são descritos a seguir, conforme nos traz Comer (2016). Clique nas abas.

- **Comunicação não confiável**

  O TCP deve lidar com mensagens perdidas, duplicadas, corrompidas, entregues com atraso ou fora de ordem.

- **Reinicialização de um ponto envolvido**

  Não deve haver confusão entre as sessões caso um dos dois pontos seja reinicializado por motivo de travamento, por exemplo.

- **Máquinas heterogêneas**

  Um aplicativo funcionando em um computador equipado com um processador muito potente pode gerar dados de forma muito rápida a ponto de inundar (ou transbordar) uma aplicação que está rodando em um computador com processador lento.

- **Congestionamento na internet**

  A transmissão de dados muito rápida faz com que computadores e roteadores intermediários possam ser abarrotados de pacotes, causando o que chamamos de congestionamento.

Os protocolos de transporte são responsáveis por lidar com esses problemas complexos que podem atrapalhar a comunicação de forma efetiva. A seguir, é descrito brevemente como ocorre a comunicação ao utilizar o TCP e, em seguida, são descritos alguns mecanismos do TCP utilizados para resolver tais problemas.

1.2.2 Protocolo TCP: comunicação

Consideramos que cada mensagem é encapsulada no que chamamos de datagrama para serem transportadas através da internet. Desse modo, o IP transfere o conteúdo para o TCP quando o datagrama chega ao destino, contudo o TCP não lê e nem interpreta essas mensagens. Assim, o IP trata cada mensagem TCP como dados que precisam ser transportados. Enquanto o TCP trata o IP como um sistema de comunicação orientado a pacotes que fornece a comunicação entre os módulos TCP em cada extremidade de uma conexão. A figura a seguir mostra como o TCP enxerga a internet. Podemos verificar que a internet toda é um sistema de comunicação que transmite e recebe mensagens sem interpretar o conteúdo delas.

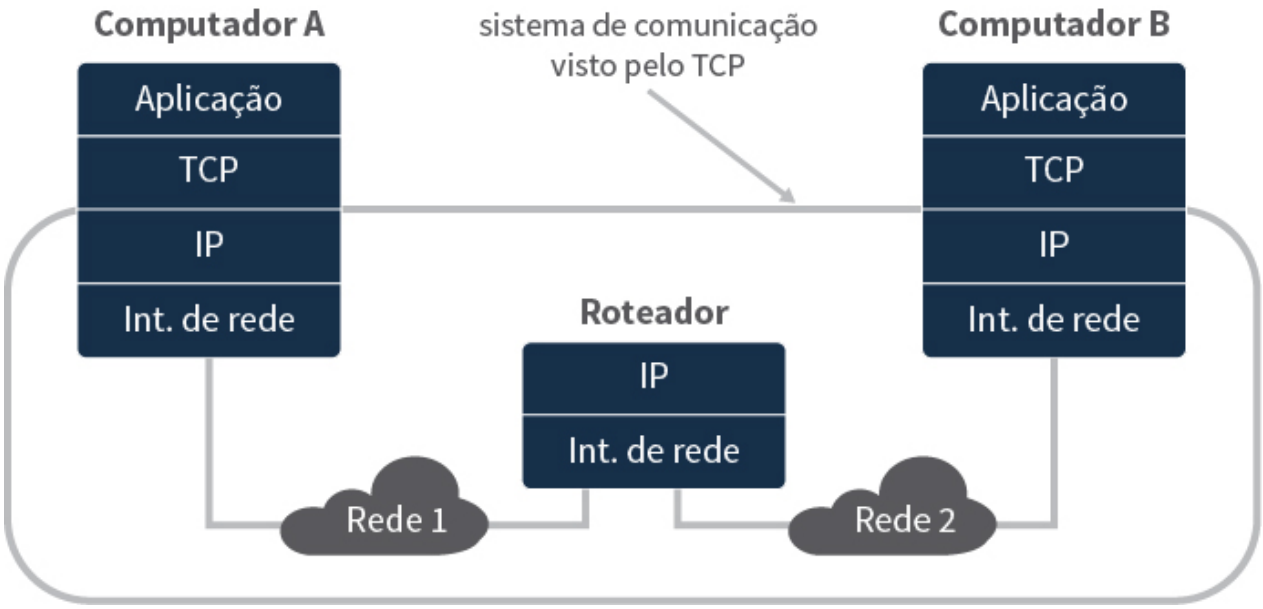


Figura 5 - Ilustração de como o TCP vê a Internet em que cada ponto terminal de uma conexão virtual precisa de um software TCP, porém os roteadores intermediários não precisam.
Fonte: COMER (2016, p. 372).

Primeiramente, como o TCP é orientado à conexão, existe o estabelecimento de uma conexão entre os processos de origem e destino para que a comunicação possa, de fato, acontecer. Em seguida, as comunicações podem acontecer nos dois sentidos em simultâneo, pois as conexões TCP são *full-duplex*. No entanto, as conexões poderão ser estabelecidas somente entre dois processos da rede, não existindo comunicações *multicast* ou *broadcast*.

Podemos abstrair o conceito de *sockets* para estabelecimento de uma conexão, em que um servidor TCP cria um *socket* para que fique esperando a solicitação dos clientes de conexão. É comum que o servidor crie um processo (ou *thread*) a cada nova conexão, assim, pode tratar a troca de mensagens por meio da nova conexão enquanto o processo original volta a esperar uma nova conexão. Enquanto isso, do lado do cliente, um *socket* também é criado para solicitar a conexão com o servidor. O cliente informa o endereço de rede do servidor e a porta utilizada para estabelecer tal conexão (KUROSE; ROSS, 2013).

VOCÊ QUER LER?

O protocolo TCP é definido no documento RFC 793. RFC significa *Request for Comments* ou Pedido para Comentários. São documentos técnicos desenvolvidos e mantidos pela instituição *Internet Enginnering Task Force* (IETF), que especifica os padrões implementados e utilizados na internet. Conheça o documento (INFORMATION SCIENCE INSTITUTE, 1981) em:

https://tools.ietf.org/pdf/rfc793.pdf

Portanto, podemos concluir que o modelo de comunicação adotado pelo TCP é o cliente-servidor, contudo, ele também é utilizado como suporte para implementar outros modelos de comunicações em níveis superiores, como o modelo *peer-to-peer*.

1.2.3 Mecanismos de resolução de problemas do TCP

Para resolver duplicatas e entregas fora de ordem é utilizada a técnica de sequenciamento, em que a origem adiciona um número de sequência a cada pacote e o destino armazena o número de sequência do último pacote recebido em ordem e armazena também uma lista dos pacotes que chegaram fora de ordem. Quando chega um pacote, o destino checa qual é o número de sequência. Se o pacote é o próximo esperado, de acordo com a ordem que deve chegar, o *software* do protocolo entrega o pacote para a próxima camada mais alta e verifica a sua lista para ver se algum pacote adicional também pode ser entregue. Se o pacote chegou fora de ordem, o *software* do protocolo acrescenta o pacote na lista dos pacotes que chegaram fora de ordem.

O problema de duplicação também é resolvido com o sequenciamento, em que um destino verifica a existência de duplicatas ao analisar o número de sequência do pacote que chegou. Se o pacote já foi entregue ou se o número de sequência corresponde a um dos pacotes esperando na lista, o *software* descarta a nova cópia (COMER, 2016).

Para reenviar pacotes perdidos na rede o TCP utiliza do mecanismo de retransmissão. Esse mecanismo também é útil quando os pacotes chegam de forma errada. Para isso, ele analisa o *checksum* de cada pacote, em que para a identificação é analisado o campo de número de sequência e para a confirmação de entrega correta ao destino existe um campo de confirmação.

De acordo com Carissimi, Rochol e Zambenedetti (2009), no TCP, a origem possui um *buffer* (temporizador) de transmissão que armazena os pacotes já transmitidos, porém que ainda não foram confirmados. Logo que as confirmações de entrega dos pacotes são recebidas pela origem, as cópias dos pacotes no *buffer* de transmissão são eliminadas, ao passo que o destino já recebeu o pacote. Assim, caso aconteça de o pacote ser perdido ou descartado, não há confirmação de recebimento enviado do destino para a origem. Após um período máximo estipulado para espera, a origem detecta que o pacote foi perdido e então toma a providência de retransmitir o pacote.

Pode, ainda, haver erros de repetição quando ocorrem atrasos muitos longos na transmissão. Comer (2016, p. 374) explica:

> Para evitar repetição, protocolos marcam cada sessão com uma identificação única (por exemplo, o horário no qual a sessão foi estabelecida) e requerem que essa identificação única esteja presente em cada pacote. O software do protocolo descarta qualquer pacote que chega com uma identificação incorreta. Para evitar repetição, uma identificação não deve ser reutilizada até ter passado um tempo razoável (por exemplo, horas).

Para evitar que um pacote transmitido por uma origem muito potente sobrecarregue (inunde) o destino que é mais lento, o TCP implementa o controle de fluxo. No TCP existe um campo de tamanho de janela que indica para a origem quantos *bytes* o destino possui disponíveis na sua janela de recepção. Isso significa que o destino pode informar à origem quantos dados ele ainda pode enviar sem que haja congestionamento. Existem técnicas para controle de fluxo, tais como *stop-and-go* e de janela deslizante. O *stop-and-go*, por exemplo, funciona com a origem transmitindo somente quando recebe uma mensagem de controle do destino, avisando quando está pronto para receber o próximo pacote.
O próximo tópico traz algumas informações básicas sobre autenticação na internet, mostrando quais são os tipos de autenticação e como lidamos com elas em relação ao cuidado de manter a segurança em rede.

1.3 Segurança da informação: autenticação

Em Ciência da Computação, a autenticação é um dos principais meios pelo qual se procura manter a segurança nos sistemas. Existe, de acordo com Stallings (2015), a autenticação de usuário e a autenticação por mensagem. Clique nas abas a seguir para saber mais.

Autenticação por usuário	Processo no qual é verificado se alguém é quem diz ser. Isso é feito geralmente no momento do login (acesso) em um software, inclusive sistemas operacionais.
Autenticação por mensagem	Processo em que as partes que se comunicam verificam se o conteúdo de uma mensagem recebida não foi alterado e se a origem é quem deveria ser, isto é, autêntica.

No que diz respeito à autenticação por usuário, que verifica a identidade deste, Stallings (2015) afirma que pode ser feita de quatro maneiras, podendo ser apenas uma ou uma combinação delas, sendo:

- algo que o indivíduo sabe: pode ser uma senha, algum número de identificação pessoal que vem no seu *smartphone* novo, por exemplo (PIN – *Personal Identification Number*) ou resposta secreta a uma pergunta que você escolhe entre algumas pré-estabelecidas, como, por exemplo, "qual foi o seu primeiro animal de estimação?" Neste último caso, existem alguns tipos de desafios-respostas que os processos de autenticação utilizam; os bancos, por exemplo, utilizam a data de nascimento do proprietário da conta;
- algo que o indivíduo possui: chaves criptográficas, cartões de senha eletrônica, *smart cards*, chaves físicas. Esse tipo de autenticador é conhecido como *token*;
- algo que o indivíduo é (biometria estática): reconhecimento por impressão digital, retina e face;
- algo que o indivíduo faz (biometria dinâmica): reconhecimento por padrão de voz, características de escrita manual e ritmo de digitação.

Uma quinta maneira que pode ser utilizada para autenticação de usuário é onde o usuário está (localização). O banco, por exemplo, pode considerar estranho você estar fazendo uma compra em um país diferente do de origem e bloquear seu cartão por segurança. Existe aplicativo de banco que permite você marcar uma opção quando está fora do país ou viajando, assim, você poderá fazer quantas comprar quiser, pois o banco saberá que é você, já que avisou que iria viajar ao marcar determinada opção no aplicativo.
E outro modo que não se encaixa em nenhuma dessas categorias acima, mas é utilizada para atestar que não tem *bot* algum tentando acessar determinado sistema, é o *captcha*. Uma sequência de caracteres dentre maiúsculos e minúsculos são exibidos em uma imagem e você deve digitá-los para poder prosseguir a navegação.

1.3.1 Mecanismos de defesa a ataques de segurança

Cada maneira de autenticação tem seus problemas. Um invasor pode adivinhar ou roubar uma senha, como também pode forjar ou roubar um *token*. O próprio usuário pode esquecer uma senha ou perder um *token* e alguém o utilizar indevidamente, se passando pelo proprietário. Além disso, os sistemas também podem estar vulneráveis a falhas de segurança. Quanto aos leitores biométricos, digitais podem não estar legíveis, pode haver falso positivo e falso negativo, além de um custo maior. E quando falamos de rede, as autenticações mais comuns são chaves criptográficas e algo que o indivíduo sabe, principalmente senhas.

Figura 6 - Inserção de senha para realizar login. Uma autenticação exigida para acesso a diversos sistemas web e aplicativos.
Fonte: mangpor2004, Shutterstock, 2019.

Na internet, há algum tempo, criar senhas se tornou algo mais valorizado pelas empresas e elas mesmas implantaram mecanismos que avaliam se sua senha, no momento de criação, é fraca, média ou forte. Assim, não é possível criar uma senha fraca. Com o avanço da capacidade de ataques, se tornou comum exigir autenticação de dois ou mais fatores, em que, além da senha, algum *token* ou mesmo a opção de biometria por digital e em casos mais sofisticados, de reconhecimento facial. As fabricantes de *smartphones* já utilizam esses dois últimos mecanismos. Enquanto os bancos utilizam em seus aplicativos, além da biometria de digital, opção de envio de SMS com um código para confirmar transações *on-line*.

VOCÊ QUER LER?

O Google revelou que exigir autenticação de dois fatores (2FA) bloqueia 100% dos ataques advindos de *bots*. Nesse caso é exigido, além de uma senha, também um *token* de segurança, que é um código enviado em tempo real por SMS ou aplicativo. Vale ressaltar que são ataques vindos de *bots*, outros tipos podem ocorrer, por isso ainda há vulnerabilidade. Você pode ler a notícia na íntegra em (OH, 2019):

https://canaltech.com.br/seguranca/google-revela-que-autenticacao-de-2-fatores-bloqueia-100-dos-ataques-via-bots-140005/

Além de todas as maneiras de autenticação por usuário, existem técnicas de criptografia que buscam proteger não apenas senhas, como conteúdo em arquivos, pastas e sistemas. No entanto, dificilmente alguma técnica é 100% segura.

Uma rede sem fio (*Wi-Fi*), por exemplo, pode ser o berço de diversos ataques sem que você perceba. Por exemplo, um criminoso pode simular ser um *hotspot* de uma *Wi-Fi* legítima, podendo observar como está o tráfego na rede e as chaves de segurança dos usuários conectados. Assim, eles acessam dados confidenciais dos usuários. Existem meios de autenticação para redes sem fio, tais como chave WEP, WPA, WPA2, autenticação em um banco de dados central, autenticação de fator duplo, entretanto, manter uma *Wi-Fi* a mais segura possível ainda é um desafio para os profissionais de TI.

Outro modo muito utilizado é o *phishing*, quando o criminoso manda um *e-mail*, parecendo ser originalmente do seu banco, por exemplo, solicitando que você acesse *links* que o levam a inserção de dados confidenciais através de formulários.

Muitas das formas de tentar quebrar a segurança de autenticação são de tentativa e erro. Desse modo, *softwares* podem ser criados para tentar, na força bruta, acessar um sistema, isto é, tentando diversas combinações de caracteres (letras, números e símbolos). Existem programas para quebrar senhas específicas da internet que utilizam protocolos HTTP, FTP e outros de *e-mail*, existem programas que quebram senhas de arquivos advindos de editores de texto, de compactadores de arquivos, de arquivos em pdf.

VOCÊ O CONHECE?

Kevin Mitnick é um dos *crackers* mais famoso já descoberto. Ele invadiu diversos sistemas e servidores por anos, tais como sistemas de empresas de telefonia, a fim de copiar e alterar dados confidenciais. Ele também invadiu contas de *e-mail*, sendo capaz de roubar senhas e ler conteúdos pessoais. Ele foi preso em 1995 pelo governo americano e depois de 20 anos trabalha em uma empresa de segurança, isto é, do outro lado do jogo, ajudando as empresas a proteger seus sistemas (TAKEDOWN, 2019).

As maneiras de autenticar usuários e proteger conteúdo, mesmo que em constante evolução pelos profissionais de TI e empresários, possuem falhas e, por isso, os usuários podem contribuir para essa segurança das informações ao tomar determinadas atitudes. Tais atitudes podem ser: criar senhas fortes, aderir ao acesso com

biometria, utilizar aplicativos que bloqueiam o acesso aos aplicativos importantes com senha, não acessar ou realizar *download* em *sites* suspeitos, verificar se o *link* do *site* é realmente oficial, ficar atento a ligações que dizem ser de bancos que solicitam informações que os bancos informam que não solicitam por telefone ou *e-mail* , verificar os endereços de remetente de *e-mails* adequadamente para não clicar em *links* que podem exigir informações confidenciais ou mesmo instalar *malwares* em seu dispositivo, dentre outros cuidados.

Síntese

Concluímos a unidade e agora você deve ter entendido que o IP é como um número de RG ou CPF para nós, humanos, só que para os dispositivos conectados à rede ou mesmo aplicativos como páginas *web* (*sites*). O IP serve para identificação desses dispositivos e aplicativos. Já o TCP é responsável por ditar as regras para que os conteúdos trafeguem na rede e cheguem sem falhas ao aplicativo de destino. A autenticação é um fator essencial para manter a segurança dos dados pessoais e confidenciais que trafegam na internet e sem ela não poderíamos, por exemplo, fazer compras *on-line* ou acessar e fazer transações em nossos aplicativos de banco.
Nesta unidade, você teve a oportunidade de:

- aprender que os protocolos são regras e procedimentos de comunicação;
- identificar que o endereçamento IP é fundamental para comunicação de computadores e dispositivos na internet, pois através do endereço IP é possível identificar unicamente cada *host* em uma rede;
- compreender que o TCP fornece um serviço de transporte orientado à conexão, *full-duplex* e confiável, que permite a dois programas de aplicação criarem uma conexão, enviarem dados em qualquer direção e, em seguida, terminarem a conexão. Cada conexão TCP é iniciada de forma confiável e finalizada suavemente.
- descobrir que para criar senhas fortes contra os ataques de segurança em uma rede, é preciso utilizar, além das senhas, outras maneiras de autenticação e tomar atitudes para não dar abertura para os ataques.

Bibliografia

CARISSIMI. A. da. S.; ROCHOL, J.; ZAMBENEDETTI, G. **Redes de computadores**. Porto Alegre: Bookman, 2009.

COMER, D. E. **Redes de computadores e internet**. 6. ed. Porto Alegre: Bookman, 2016.

GUANABARA, G.; JÚNIOR, A. **Endereçamento IP (Parte 1).** Direção: Canal em Vídeo. YouTube. Brasil: Canal em Vídeo, 2019. Disponível em: https://www.youtube.com/watch?v=q65kHIvtWxg&feature=youtu.be. Acesso em: 27 jun. 2019.

GUANABARA, G.; JÚNIOR, A. **Endereçamento IP (Parte 2)**. Direção: Canal em Vídeo. YouTube. Brasil: Canal em Vídeo, 2019. Disponível em: https://www.youtube.com/watch?v=ee5htpGdWHY&feature=youtu.be. Acesso em: 27 jun. 2019.

INFORMATION SCIENCE INSTITUTE. **Transmission Control Protocol**. Darpa Internet Program. Protocol Specification. University of Southern California, 1981. Disponível em: https://tools.ietf.org/pdf/rfc793.pdf. Acesso em: 27 jun. 2019.

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

MEU IP. **Painel de Controle**. 2019. Disponível em: https://www.meuip.com.br/. Acesso em: 27 jun. 2019.

MORAES, A. F. **Redes de computadores:** fundamentos. 7. ed. São Paulo: Érica, 2010.

OH, S. H. Google revela que autenticação de 2 fatores bloqueia 100% dos ataques via bots. **Canal Tech**, 24 mai. 2019. Segurança. Disponível em: https://canaltech.com.br/seguranca/google-revela-que-autenticacao-de-2-fatores-bloqueia-100-dos-ataques-via-bots-140005/. Acesso em: 27 jun. 2019.

STALLINGS, W. **Criptografia e segurança de redes:** princípios e práticas. 6. ed. São Paulo: Pearson Education do Brasil, 2015.

TAKEDOWN. **Biography Kevin Mitnick**. 2019. Disponível em: http://www.takedown.com/bio/index.html#Kevin. Acesso em: 27 jun. 2019.