



## Camada de rede

Prof. Fábio Contarini Carneiro

### Descrição

Apresentação da camada de rede do Modelo OSI e o seu relacionamento com as camadas superior e inferior, bem como a sua implementação na Arquitetura TCP/IP (arquitetura utilizada na Internet).

### Propósito

Reconhecer o funcionamento da camada de rede para o desenvolvimento de aplicações modernas que tenham como requisito a comunicação em rede; realizar um endereçamento uniforme permitindo que equipamentos e aplicações possam trocar informações.

# Objetivos

Módulo 1

## Camada OSI

Reconhecer os conceitos básicos da camada de rede no Modelo OSI.

Módulo 2

## Configurações de rede IP

Aplicar configurações de rede IP.

Módulo 3

## Protocolos de controle

Classificar os protocolos de controle da Internet.

Módulo 4

## Algoritmos de roteamento

Selecionar os algoritmos de roteamento.



# Introdução

Imagine que você está na sua casa e deseja ir para um evento com seus amigos. Você tem o endereço do local de encontro, mas precisa determinar o caminho até chegar ao destino.

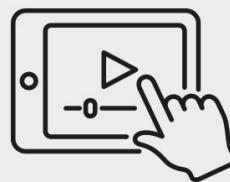
Provavelmente você utilizará algum aplicativo de GPS ou de transporte público que lhe fornecerá o caminho a percorrer ou conduções a serem utilizadas.

Esse exemplo que você viu acima, algo rotineiro no nosso dia a dia, é o que a camada de rede faz. Ela é responsável por determinar o caminho entre a máquina de origem e a máquina de destino, atividade que chamamos de roteamento.

Mas ela não faz apenas isso! A camada de rede e seus respectivos protocolos também são responsáveis por definir o formato do endereço a ser utilizado, definição de protocolos auxiliares para sinalizar possíveis erros que ocorram no processo de roteamento, entre outras funções.

Vamos agora desvendar como a camada de rede executa essas funções, entendendo os princípios de funcionamento e como os protocolos operam.

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.

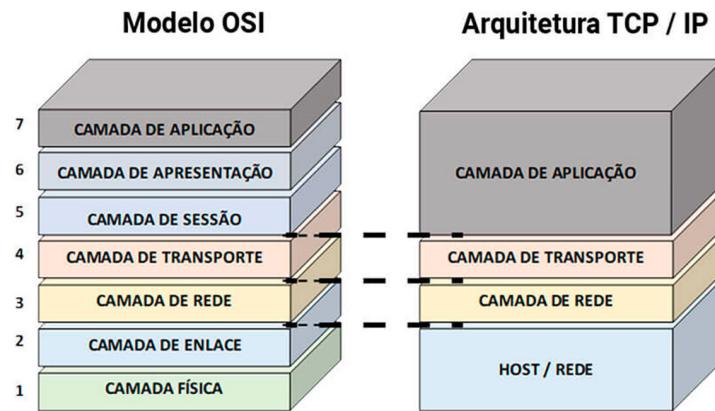


## 1 - Camada OSI

Ao final deste módulo, você será capaz de reconhecer os conceitos básicos da camada de rede no Modelo OSI.

# Camada de rede

No [Modelo OSI](#), a camada de rede situa-se logo acima da camada de [enlace](#), utilizando os serviços oferecidos por essa camada.



A **camada de enlace** tem como objetivo organizar a comunicação em nível de enlace, ou seja, ela deve garantir que um hospedeiro consiga se comunicar com o hospedeiro vizinho.

O objetivo da **camada de rede** é, a partir desse serviço de comunicação lado a lado, promover uma comunicação de âmbito global, permitindo que a informação que sai de um **hospedeiro** chegue a seu destino não importando em qual local da rede esteja esse destino. Para isso, ele deve realizar duas grandes funções: definir um esquema de endereçamento que seja aceito por toda a grande rede, e realizar o **roteamento**.

## Modelo OSI

O Modelo OSI (*Open System Interconnection*) foi criado pela International Organization for Standardization (ISO) com o objetivo de ser um padrão para construção de redes de computadores. Divide a rede em 7 camadas, em que cada camada realiza funções específicas, implementadas pelo protocolo da camada, prestando serviços para sua camada superior.

## Enlace

Ligação física entre dois ou mais dispositivos que permite a troca de dados entre os dispositivos participantes.

## Hospedeiro

Também conhecido como host, é qualquer equipamento conectado à rede capaz de trocar informações com outros equipamentos. Exemplos de hospedeiros são computadores, impressoras de rede, smartphones etc.

## Roteamento

Processo de determinar o caminho a ser seguido por um pacote de dados em uma rede para que chegue ao destino da melhor forma possível.

## Conceitos básicos da camada de rede

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



A quantidade máxima de dados que podem ser trocados entre entidades da camada de rede possui um tamanho limitado. Para que as mensagens maiores que o limite possam ser trocadas entre aplicações, a camada de transporte deverá segmentar tais mensagens em partes menores e colocar cada segmento em uma unidade de dados da camada de rede.

Essa unidade de dados trocada entre entidades da camada de rede recebe a denominação de **pacote**.

### Pacote

Porção de dados trocada por protocolos da camada de rede contendo dados mais informações de cabeçalho suficientes para que possam seguir seu caminho até o hospedeiro destino.

#### Comentário

É um trabalho diferente daquele realizado pela camada de enlace, que necessita apenas entregar a informação na outra extremidade do meio de transmissão.

A camada de enlace de dados cuida apenas da comunicação de um hospedeiro com outro que esteja diretamente conectado a ele por intermédio de um meio de comunicação. Já a camada de rede vai encaminhar os pacotes permitindo que ocorra a comunicação fim a fim.

Para conseguir realizar essa comunicação, a camada de rede deve:

- Conhecer toda a **topologia** da **sub-rede** de comunicação.
- Escolher rotas que evitem sobrecarregar partes da rede enquanto outras ficam ociosas.
- Compatibilizar a comunicação entre os diferentes tipos de sub-redes existentes.

Na origem, a camada de rede cria um pacote com dados provenientes da camada superior e acrescenta a ele um cabeçalho com informações de endereço origem e destino. Em sistemas com mais de uma **interface**

**de rede** deve também consultar suas **tabelas de repasse** para determinar por qual interface o pacote deve ser encaminhado.

Nos hospedeiros intermediários, a camada de rede deve receber o pacote, consultar suas tabelas de roteamento e enviar o pacote pela interface de rede apropriada.

No destino, a camada de rede deve verificar se o endereço destino do pacote é o mesmo do hospedeiro e, caso seja, entregar o pacote à camada superior (camada de transporte).

## Topologia

Layout da rede de computadores.

## Sub-rede

Subdivisão de uma grande rede de computadores. Os roteadores que fazem parte de uma sub-rede devem utilizar o mesmo software de roteamento para que possam trocar informações sobre rotas.

## Interface de rede

Dispositivo que permite que um hospedeiro possa enviar e receber dados por intermédio de uma rede de computadores. Exemplos de interface de rede são placas de rede ethernet, muito comuns em computadores desktop, e placas de comunicação Wi-Fi comuns em notebooks e smartphones.

## Tabelas de repasse

Tabela interna de um roteador montada por seu algoritmo de roteamento que possui entradas para todos os destinos conhecidos, com indicação por qual interface de saída deve ser enviado um pacote com base em seu endereço destino.

# Comutação na camada de rede

Neste vídeo, o especialista explica sobre comutação de pacotes.

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.

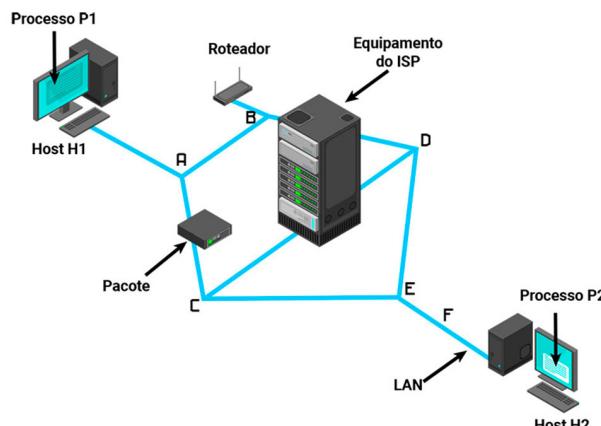


## Comutação de pacotes store and forward

Neste momento você deve estar se perguntando como o pacote vai da origem até o destino. Diretamente? Saltando de hospedeiro em hospedeiro?

Lembre-se de que, no Modelo OSI, uma camada utiliza os serviços da camada imediatamente inferior para realizar seus trabalhos e oferecer serviços para a camada superior. Assim, um hospedeiro com um pacote a enviar o transmite para o roteador mais próximo. O pacote é então totalmente recebido pelo roteador e, após ter chegado por completo, é conferido. Em seguida, ele é encaminhado para o próximo roteador ao longo do caminho até alcançar o hospedeiro de destino, onde é entregue. Esse mecanismo que tem como base armazenar totalmente o pacote antes de enviá-lo ao próximo hospedeiro é conhecido como comutação de pacotes store and forward.

Confira esse passo a passo na imagem abaixo:



Entenda as diferenças entre repasse e roteamento:



### Repasse

Um roteador é um equipamento que possui, normalmente, várias interfaces de rede. Quando um pacote chega por uma de suas interfaces, o roteador deve verificar o destino desse pacote e decidir por qual de suas linhas o pacote deve ser enviado para que consiga chegar ao destino. Chamamos de repasse o trabalho (local) de

escolher por qual das interfaces deve seguir o pacote que chega.



## Roteamento

Cabe à camada de rede decidir o caminho a ser seguido por um pacote para que este chegue ao seu destino, fazendo o melhor caminho possível. Para isso, os roteadores devem trocar informações entre si sobre o estado da rede, de forma a decidirem o melhor caminho para cada destino. Chamamos de roteamento o trabalho (global) de escolher os caminhos por uma rede para que o repasse possa ser realizado.

O algoritmo que realiza esta tarefa é conhecido como algoritmo de roteamento.

O repasse refere-se a uma ação realizada localmente por um roteador sobre um pacote que chega, enquanto no roteamento há uma ação global envolvendo todos os roteadores de uma sub-rede para a escolha do melhor caminho.

Você já deve ter percebido que existe uma relação bastante próxima entre repasse e roteamento, mas como é essa relação?

Vamos iniciar pelos algoritmos de roteamento. Para escolher a melhor rota, um conjunto de roteadores precisa trocar informações sobre a situação da rede em sua vizinhança, de modo que os roteadores do conjunto possam decidir sobre o caminho a ser seguido em cada situação. Diferentes algoritmos com diferentes propriedades podem ser utilizados nessa tarefa.

Como funcionam esses algoritmos e de que forma eles se diferenciam, estudaremos adiante. Por ora, o importante é que existem algoritmos capazes de encontrar o melhor caminho em uma rede.

O mesmo algoritmo é então executado nos roteadores da sub-rede, trocando informações de tráfego e calculando a melhor rota. Uma vez calculadas as rotas para cada destino, os algoritmos de roteamento criam as chamadas tabelas de repasse, as quais são indexadas pelos

possíveis endereços destino de um pacote e indicam a interface pela qual o pacote deve ser enviado com base em seu endereço destino.

Ao chegar um pacote em um roteador, e sendo necessário fazer seu repasse, o **processo** responsável por essa função verifica o endereço de destino do pacote, consulta a tabela de repasse utilizando o endereço destino como índice e obtém a interface pela qual o pacote deve seguir. Resumidamente, as tabelas de repasse são montadas pelos algoritmos de roteamento e consultadas pelo processo de repasse para determinar por qual interface deve seguir um pacote que chega.

## Processo

Um processo é um programa em execução, incluindo sua região de memória, valores de variáveis e seu contexto de hardware.

## Circuitos virtuais e datagramas

Há basicamente duas formas para a organização da camada de rede de uma sub-rede: uma utilizando **conexões** e a outra trabalhando sem conexão. Na camada de rede, um serviço orientado à conexão costuma ser chamado de circuito virtual, enquanto um serviço que realiza a entrega de pacotes independentes, **sem conexão**, costuma ser chamado de serviço de datagramas.

A camada de transporte também oferece serviços com conexão e sem conexão, mas estes se diferem dos serviços oferecidos pela camada de rede.

### Comentário

Pensando na arquitetura TCP/IP, a camada cliente da camada de transporte é a camada de aplicação, portanto, o serviço de transporte com conexão é um serviço aplicação a aplicação, ou seja, uma conexão de transporte conecta diretamente duas aplicações. Já a camada de rede é voltada para a interligação de hospedeiros, logo, uma conexão da camada de rede é uma conexão hospedeiro a hospedeiro.

# Encaminhamento por circuitos virtuais

## Circuitos virtuais

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



A ideia dos **circuitos virtuais** é evitar a necessidade de escolher uma nova rota para cada pacote que passa, sendo por isso utilizado em sub-redes com serviço de rede orientado à conexão. Ao se estabelecer uma conexão, é criada uma rota entre o hospedeiro origem e o hospedeiro destino como parte do estabelecimento da conexão. Essa rota é utilizada por todo o tráfego que flui pela conexão entre os hospedeiros. Quando a conexão é liberada, o circuito virtual deixa de existir.

Um circuito virtual (CV) consiste em:

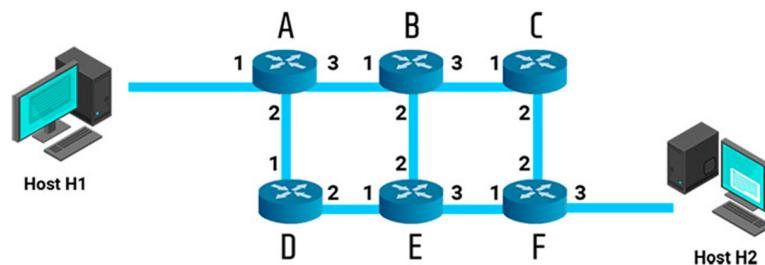
- Um caminho definido entre origem e destino;
- Números de identificação de circuitos virtuais, um para cada enlace ao longo do caminho;
- Registros em tabelas de **comutadores** ao longo do caminho.

Considere a rede da figura a seguir. Os números em cada extremidade dos enlaces representam o número da interface do comutador naquele enlace.

## Comutadores

Dispositivos utilizados em redes de computadores para o encaminhamento de dados com base em seu endereço destino ou circuito virtual.

Diferenciam-se dos roteadores por causa do trabalho realizado e por serem mais rápidos no encaminhamento de dados.



O hospedeiro H1 solicita à rede que estabeleça um circuito virtual até o hospedeiro H2, e a rede estabelece o caminho H1-A-B-E-F-H2, atribuindo respectivamente os seguintes circuitos virtuais: 23, 8, 37, 22, e 16 (numerações aleatórias).

Com base nessas informações são montadas as seguintes tabelas de repasse em cada um dos comutadores ao longo do caminho:

Tabela de repasse de A			
Entrada	Saída		
Linha	CV	Linha	CV
1	23	3	8

Tabela de repasse de B			
Entrada	Saída		
Linha	CV	Linha	CV
1	8	2	37

Tabela de repasse de E			
Entrada	Saída		
Linha	CV	Linha	CV
2	37	3	22

Tabela de repasse de F			
Entrada	Saída		
Linha	CV	Linha	CV
1	22	3	16

Sempre que um novo circuito virtual (CV) é estabelecido através de um comutador, um registro para este CV é adicionado à tabela do comutador. De forma análoga, sempre que um CV termina, suas informações são removidas das tabelas do comutador ao longo do percurso.

A utilização de diferentes identificadores para os CVs, ao longo do caminho, simplifica o trabalho, uma vez que, a utilização de um mesmo CV durante todo o trajeto iria requerer a concordância de todos os comutadores participantes da rota. O importante para que os comutadores possam definir o caminho é que o número de saída do CV de um comutador seja o mesmo número de entrada do CV do comutador seguinte.

No exemplo, o circuito estabelecido entre os comutadores E e F possui CV 22 (o CV 22 sai pela linha 3 de E e entra como CV 22 pela linha 1 de F). Da mesma forma, o CV entre A e B possui valor 8 e o CV entre B e E possui valor 37. Reveja as figuras com os circuitos em destaque.

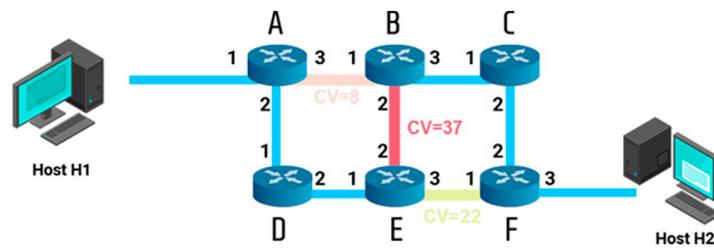


Tabela de repasse de A			
Entrada	Saída		
Linha	CV	Linha	CV
1	23	3	8

Tabela de repasse de B			
Entrada	Saída		
Linha	CV	Linha	CV
1	8	2	37

Tabela de repasse de E			
Entrada	Saída		
Linha	CV	Linha	CV
2	37	3	22

Tabela de repasse de F			
Entrada	Saída		
Linha	CV	Linha	CV
1	22	3	16

Ao estabelecer um caminho dessa forma, é como se fosse colocado um circuito físico (como um cabo de rede) ligando diretamente os hospedeiros, por isso o nome circuito virtual.

Outra característica da rede de circuitos virtuais é que, como os pacotes seguem sempre pelo mesmo caminho, eles chegam ao destino na mesma ordem em que saíram da origem.

Existem 3 fases que podem ser identificadas em um circuito virtual:



### Estabelecimento do CV

Nesta fase, os comutadores estabelecem os parâmetros da conexão e o caminho pelo qual os pacotes irão seguir durante a fase de transferência de dados.



### Transferência de dados

Fase seguinte ao estabelecimento do CV, durante a qual ocorre a transferência de pacotes desde a origem até o destino. Todos os pacotes seguem pelo caminho definido durante a fase de estabelecimento do CV.



### Encerramento do CV

Fase na qual o circuito virtual é desfeito. Necessária para que os comutadores sejam avisados do encerramento e possam retirar de suas tabelas as informações sobre o circuito virtual encerrado.

## Encaminhamento por datagramas

### Datagramas

Em uma sub-rede de datagramas, **nenhuma rota é previamente definida**. Ao ser transmitido, um pacote passa por uma série de roteadores, e cada roteador, ao longo do caminho, utiliza o endereço destino para determinar por qual interface de saída enviar o pacote (realiza o repasse).

Como não há estabelecimento de circuitos para as transferências, os roteadores, ao longo do caminho, deverão manter tabelas de repasse baseadas no endereço destino do pacote, qualquer que seja o destino. Significa dizer que, se as tabelas de repasse forem alteradas durante

uma transferência, os datagramas de uma mensagem podem fazer caminhos diferentes até o destino.

### Exemplo

Suponha que você esteja assistindo a um filme pela rede e que o vídeo seja enviado até seu equipamento por uma sub-rede de datagramas. Parece real? Sim, é real. Atualmente, assistimos a filmes pela Internet e ela é uma rede de datagramas.

O filme possui cerca de 90 minutos. Será que as rotas de uma rede mudam durante 90 minutos? Sim, essa é uma possibilidade. Isso significa que, em determinado momento, as tabelas de repasse mudarão, então, os datagramas passarão a fazer outro caminho até chegarem em seu equipamento. Assim, vemos que, ao contrário dos circuitos virtuais, os pacotes em uma rede de datagramas não seguem sempre pelo mesmo caminho.

Por que a rota é alterada?

Uma possibilidade é que os pacotes estavam passando por uma região que ficou sobrecarregada, e os roteadores descobriram outra região com melhores condições de tráfego, preferindo enviar os pacotes por essa nova região. Uma consequência disso é que os primeiros pacotes que estão seguindo pelo novo caminho podem chegar ao destino antes de alguns dos pacotes que seguiram pelo caminho antigo.

Assim sendo, existe a possibilidade de que alguns pacotes cheguem ao destino fora de ordem.

Os problemas que podem acontecer em uma rede de datagramas são:



### A perda de datagramas



### Datagramas chegarem com erro



### Datagramas chegarem fora de ordem



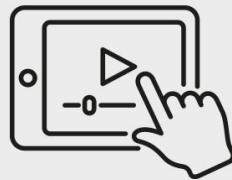
### Datagramas serem duplicados



## Protocolo ARP

Para saber mais sobre protocolo ARP, assista ao vídeo abaixo:

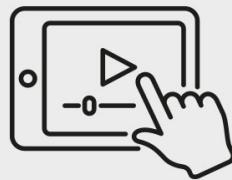
Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



## Endereçamento na camada de rede

### Endereçamento

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



Alguns tipos de rede permitem que sejam criados vários endereços por hospedeiro, enquanto outros tipos de rede permitem a utilização de apenas um por hospedeiro. Porém, qualquer que seja o tipo de rede, **o endereçamento deve ser completamente independente do endereçamento dos protocolos de outras camadas**. A camada de rede tem a função de unificar toda a comunicação da rede, portanto ela deve definir uma forma de identificação dos hospedeiros que seja aceita por toda a rede.

Basicamente dois tipos de endereçamento são possíveis:

#### Endereçamento hierárquico

No **endereçamento hierárquico**, o endereço de uma entidade é constituído de acordo com os endereços correspondentes aos vários níveis da hierarquia da qual faz parte. O endereço hierárquico é o método sugerido pelo ITU-T para interconexão de

redes públicas de telefonia. Nessa recomendação, os endereços são números decimais formados por três campos: um código do país, um código para a rede, e um campo para o endereçamento dentro da rede.

**Observação:** setor da ITU (International Telecommunication Union) responsável pela elaboração de padrões e normas consensuais sobre tecnologia que garantam o funcionamento, a interoperabilidade e a integração dos sistemas de comunicação em todo o mundo, com a finalidade de facilitar o acesso das indústrias aos diferentes mercados de cada país.

### Endereçamento horizontal

No **endereçamento horizontal**, os endereços não têm relação alguma com o lugar onde estão as entidades dentro da rede. Um exemplo comum desse tipo de endereçamento são os endereços utilizados nas placas de rede ethernet, que são gravados durante seu processo de fabricação e serão sempre os mesmos, não importando em qual lugar do mundo a placa seja utilizada.

**Observação:** placa de rede ethernet é um dispositivo de hardware que permite a ligação de um computador a uma rede de computadores padrão ethernet.



Considerações sobre o roteamento indicam vantagens na utilização de endereçamento hierárquico, uma vez que este contém informações explícitas sobre o local onde se localizam as entidades. Já o endereçamento horizontal permite uma mobilidade das entidades sem reconfiguração.

Apesar do endereçamento universal ser realizado pela camada de rede, o envio dos pacotes é realizado por camada inferior que possui seu próprio esquema de endereçamento específico da tecnologia da rede física em uso. Ocorre, portanto, um mapeamento entre os endereços físicos e de rede, e essa tarefa deve ser realizada pela camada de rede.

Existem duas técnicas usuais para essa conversão: **resolução por meio de mapeamento direto e resolução por meio de vinculação dinâmica.**



### Mapeamento direto

A estação sabe como computar o endereço de enlace por intermédio de uma função que mapeia o endereço de rede no endereço de enlace. Por exemplo, o endereçamento hierárquico onde o campo de endereçamento dentro da rede corresponda exatamente ao endereço físico da estação. Conversões mais complexas podem ser realizadas por meio de tabelas de conversão e técnicas de acesso rápido a essas tabelas.



### Vinculação dinâmica

Para evitar o uso de tabelas de conversão, uma vinculação dinâmica pode ser efetuada entre o endereço de rede e o físico por intermédio da utilização de algum protocolo de resolução. A exemplo, temos o protocolo ARP, o qual estudaremos mais adiante.

Falta pouco para atingir seus objetivos.

Vamos praticar alguns conceitos?

#### Questão 1

Quando um pacote chega para ser encaminhado por um roteador, o processo responsável pelo repasse deve

**A**

consultar outros roteadores para determinar a interface de saída a ser utilizada.

**B**

decidir pela interface de saída com base no endereço origem do pacote.

**C**

montar a tabela de repasse com base nos dados obtidos pelo algoritmo de roteamento.

**D**

escolher a interface de saída com base em sua tabela de repasse.

**E**

encaminhar o pacote para todas as interfaces do roteador.

Parabéns! A alternativa D está correta.

O algoritmo de roteamento de um roteador troca constantemente informações com outros roteadores sobre o estado da rede e, com base nessas informações, monta a tabela de repasse que determina por qual interface deve seguir um pacote que chega. Quando o pacote chega, cabe ao processo de repasse apenas consultar a tabela de repasse que definirá por qual interface o pacote deve seguir.

## Questão 2

Em uma rede de circuitos virtuais

**A**

a identificação do circuito virtual permanece a mesma ao longo de todo o caminho.

**B**

os pacotes sempre chegam ao destino na ordem em que foram enviados.

C

a rota é definida pelos comutadores antes mesmo do estabelecimento da conexão.

D

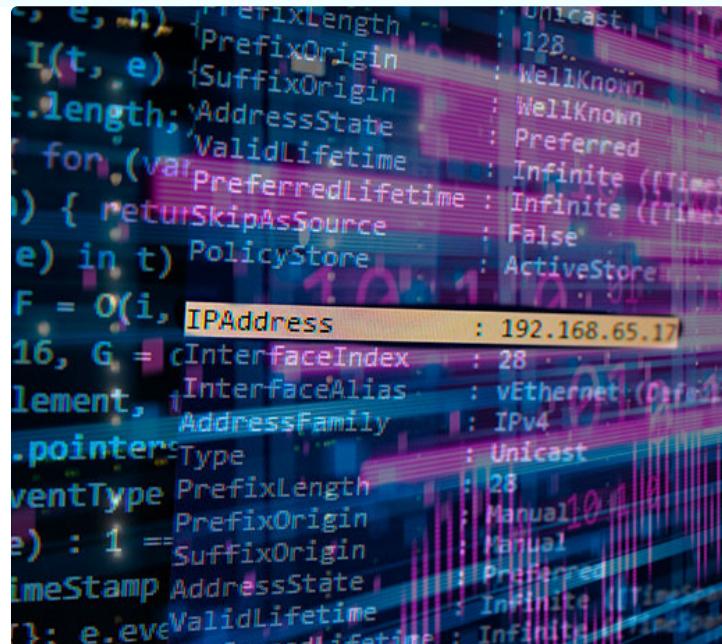
a rota pode ser alterada se o algoritmo de roteamento encontrar um caminho melhor.

E

é garantida a entrega de todos os pacotes enviados.

Parabéns! A alternativa B está correta.

Em uma rede de circuito virtual, a rota a ser seguida pelos pacotes é definida no momento do estabelecimento da conexão e nunca muda. Como os pacotes seguem sempre pelo mesmo caminho, eles mantêm a posição um em relação ao outro, chegando ao destino na mesma ordem em que foram enviados.



```
, e, n), PrefixLength : Unicast, IP(t, e) PrefixOrigin : WellKnown, SuffixOrigin : WellKnown, t.length; AddressState : Preferred, for (var ValidLifetime : Infinite ([Times])) { PreferredLifetime : Infinite ([Times]) } returns SkipAsSource : False, e) in t) PolicyStore : ActiveStore, F = O(i, IPAddress : 192.168.65.17, 16, G = cInterfaceIndex : 28, element, InterfaceAlias : vEthernet (Physical Adapter), AddressFamily : IPv4, pointers.Type : Unicast, eventPrefixLength : 28, PrefixOrigin : Manual, SuffixOrigin : Manual, timeStamp AddressState : Preferred, ValidLifetime : Infinite ([Times]), PreferredLifetime : Infinite ([Times])}; e.eventValidLifetime : Infinite ([Times]), PreferredLifetime : Infinite ([Times])}
```

## 2 - Configurações de rede IP

Ao final deste módulo, você será capaz de aplicar configurações de rede IP.

# A camada de rede na Internet

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.

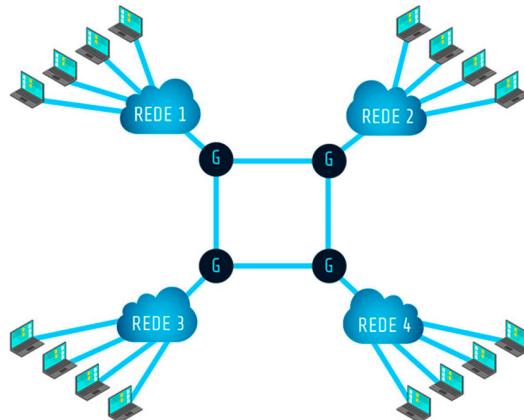


Na camada de rede, a Internet pode ser vista como um conjunto de sub-redes ou Sistemas Autônomos (SA) conectados entre si. Não existe uma estrutura real, mas diversos **backbones** que interligam redes regionais, que por sua vez estão conectadas às redes locais.

O protocolo que mantém a Internet unida é o **protocolo de camada de rede IP** (Internet Protocol). A tarefa do IP é fornecer a melhor forma de transportar datagramas da origem para o destino, independentemente de esses hospedeiros estarem na mesma rede ou em outras redes intermediárias.

## Backbones

Linhos de comunicação de dados que formam a espinha dorsal de um sistema de comunicação, provendo meio de comunicação entre redes de computadores. Normalmente, possuem tecnologia com elevado desempenho para transmissão de dados.



Imagine que você está solicitando a transferência de dados de seu computador para um hospedeiro remoto (está fazendo o upload do arquivo). A camada de transporte de seu computador receberá o fluxo de dados referente à transferência de seu arquivo. Como provavelmente esse fluxo é muito grande para caber em um único pacote IP, ele será quebrado em partes menores pela camada de transporte e cada parte desta será entregue para a camada de rede (cada um será colocado em um datagrama IP).

Esses datagramas serão transmitidos pela Internet, podendo ainda serem fragmentados ao longo do caminho. Quando todos os fragmentos de um datagrama chegam ao destino, o datagrama original

é remontado pela camada de rede do destino sendo entregue à camada de transporte que recriará o fluxo original para o processo de recepção.

# Protocolo de endereços IPv4

## Protocolo IP versão 4 (IPv4)

Na Internet, a camada de rede trata cada pacote de forma independente, não tendo qualquer relação com pacotes anteriores ou posteriores. Pacotes com mesma origem e destino podem, inclusive, passar por diferentes rotas.

O IPv4 é definido pela [RFC 791](#), sendo atualizado pelas RFC 1349, RFC 2474 e RFC 6864.

## RCF

*Request for Comments* são documentos públicos mantidos pela Internet Engineering Task Force.



### Datagramas

São unidades básicas de informações que passam pela Internet. Dentro do datagrama está seu cabeçalho, que contém informações sobre sua origem e seu destino, assim como informações sobre para qual protocolo o IP deverá passar os dados. Também fazem parte dele os dados que devem ser entregues à camada de transporte no destino.



### Fragments

São partes de um datagrama muito grande que foi quebrado em unidades menores. Como os dados precisam caber dentro da porção de dados da rede física, pode ser que seja necessário fragmentar os dados da aplicação para que eles possam ser

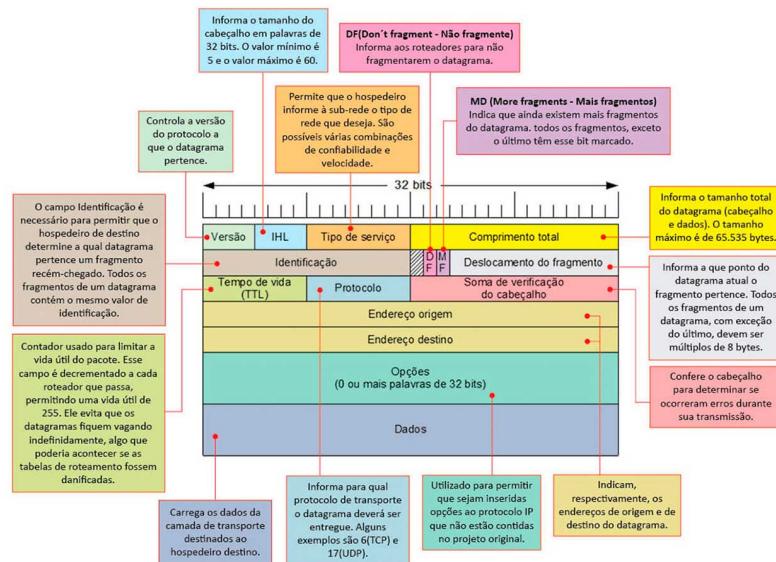
Camada de rede  
transportados pela rede. O tamanho do fragmento é determinado pelo MTU da interface de hardware da rede. O IPv4 especifica que a fragmentação ocorre em cada roteador baseado na **MTU** da interface pela qual o datagrama IP deve passar.

## MTU

*Maximum Transmission Unit*. Refere-se à quantidade máxima de dados que pode ser enviada em uma mensagem de um protocolo de rede.

Quando um datagrama é quebrado em fragmentos, cada um desses fragmentos passa a ser um novo datagrama que está passando pela rede.

Um datagrama IP consiste em duas partes: cabeçalho e dados. O cabeçalho tem uma parte fixa de 20 bytes e uma parte opcional de tamanho variável.



## Fragmentação

Para saber mais sobre como funciona a fragmentação, assista ao vídeo:

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



## Endereço IPv4

### Endereçamento IPv4

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



Um hospedeiro normalmente possui apenas uma interface com a rede de computadores. Quando o protocolo IP de um hospedeiro quer enviar um datagrama, ele o faz por meio dessa interface.

**A interface de rede é a fronteira entre o hospedeiro e a rede física.**

Considere, neste momento, um roteador e suas interfaces. Como a tarefa de um roteador é receber um datagrama por uma interface e repassá-lo por outra interface, ele estará ligado a duas ou mais redes físicas. A fronteira entre o roteador e qualquer uma dessas redes também é denominada uma interface. Assim, um roteador tem múltiplas interfaces, uma para cada uma das redes às quais está conectado.

#### Atenção!

Independentemente de quantas interfaces tenha um hospedeiro, o IP exige que cada uma delas tenha seu próprio endereço IP de modo que este, tecnicamente, esteja associado a uma interface, e não ao hospedeiro que a contém.

Cada endereço IP tem comprimento de 32 bits (4 bytes), havendo um total de 232 endereços possíveis (aproximadamente 4 bilhões de endereços IP), os quais são escritos em notação decimal separada por pontos, na qual cada byte do endereço é escrito em sua forma decimal e separado dos outros bytes por um ponto.

Considere o endereço IP 192.168.23.67. O 192 é o número decimal equivalente aos primeiros 8 bits do endereço; o 168 é o decimal equivalente ao segundo conjunto de 8 bits do endereço, e assim por diante. Em notação binária, o endereço 192.168.23.67 fica:

**11000000 10101000 00010111 01000011**

**11000000 10101000 00010111 01000011**

$11000000 = 2^7 + 2^6 = 128 + 64 = 192$

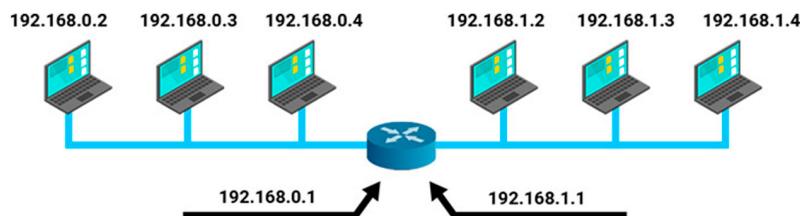
$10101000 = 2^7 + 2^5 + 2^3 = 128 + 32 + 8 = 168$

$00010111 = 2^6 + 2^1 + 2^0 = 64 + 2 + 1 = 67$

$01000011 = 2^4 + 2^2 + 2^1 + 2^0 = 16 + 4 + 2 + 1 = 23$

Cada interface em cada hospedeiro da Internet precisa de um endereço IP globalmente exclusivo, ou seja, não pode haver duas interfaces com o mesmo endereço IP na Internet válida.

A figura, a seguir, fornece um exemplo de endereçamento IP e interfaces em que um roteador com 2 interfaces é usado para interconectar os 6 hospedeiros. Observando os endereços dos hospedeiros da esquerda e da interface do roteador que os liga, percebe-se que todos possuem endereço IP na forma 192.168.0.x, enquanto os hospedeiros e a interface do roteador localizados na parte direita da figura possuem endereço na forma 192.168.1.x. Diz-se que cada uma dessas redes que interconecta 3 hospedeiros mais a interface do roteador formam uma sub-rede.



Como pode-se observar, no exemplo, os 3 campos mais à esquerda do endereço (formados por 24 bits) nunca mudam dentro da sub-rede. Desse modo, temos à esquerda a sub-rede 192.168.0.0/24 e à direita a sub-rede 192.168.1.0/24. O "/24" indica que os 24 bits mais à esquerda do endereço definem a **sub-rede**.

**A estratégia de atribuição de endereços da Internet é conhecida como Roteamento Interdomínio sem Classes (Classless Inter-domain Routing - CIDR).**

O CIDR generaliza a noção de endereçamento de sub-rede. O endereço IP de 32 bits é dividido em 2 partes sendo representado por A.B.C.D/X, em que X indica o número de bits existentes na primeira parte do endereço.

Os X bits mais significativos de um endereço na forma A.B.C.D/X constituem a parcela da rede do endereço IP e normalmente são denominados prefixo (ou prefixo de rede). Esse prefixo determina qual parte do endereço IP identifica a rede. Uma organização normalmente recebe um bloco de endereços contíguos, ou seja, uma faixa de endereços com um prefixo comum. Assim, os endereços IP de hospedeiros dentro da organização compartilharão o prefixo comum, o que reduz consideravelmente o tamanho da tabela de repasse nos roteadores, visto que um único registro da forma A.B.C.D/X será suficiente para transmitir pacotes partindo de fora para qualquer destino dentro da organização.

Os últimos (32-X) bits de um endereço podem ser considerados como os bits que distinguem os hospedeiros dentro da sub-rede. Esses bits menos significativos podem (ou não) ter uma estrutura adicional de sub-rede tal como aquela discutida anteriormente. Por exemplo, suponha que os primeiros 24 bits do endereço A.B.C.D/24 especificam o prefixo da rede da organização e são comuns aos endereços IP de todos os hospedeiros da organização. Os 8 bits restantes, então, identificam os hospedeiros específicos da organização.

A estrutura interna da organização poderia ser tal que esses 8 bits mais à direita seriam usados para criar uma sub-rede dentro da organização, como discutido. Por exemplo, A.B.C.D/28 poderia se referir a uma sub-rede específica dentro da organização.

Anteriormente ao CIDR, os bits reservados para indicar a parte da sub-rede possuíam exatamente 8, 16 ou 24 bits. Esse esquema é conhecido como endereçamento de classes cheias, e as sub-redes com endereços de sub-rede de 8, 16 e 24 eram conhecidas, respectivamente, como redes de classe A, B e C. Tal divisão em classes cheias se mostrou problemática para suportar o rápido crescimento da quantidade de organizações com sub-redes de pequeno e médio portes.

Uma sub-rede de classe C (/24) poderia acomodar apenas  $2^8 - 2 = 254$  hospedeiros (dois dos  $2^8 = 256$  endereços são reservados para uso especial). Essa sub-rede é muito pequena para muitas organizações, por outro lado, uma sub-rede de classe B (/16), que suporta até  $2^{16} - 2 = 65.534$  hospedeiros, seria demasiadamente grande. Com o endereçamento de classes cheias, uma organização com 2.000 hospedeiros receberia um endereço de sub-rede de classe B (/16), o que resultava no rápido esgotamento do espaço de endereços de classe B e na má utilização do espaço de endereço alocado.

Os 2 endereços especiais de uma sub-rede que não podem ser utilizados são o primeiro e o último endereço da faixa de endereços da organização. O primeiro é reservado para o endereço de rede, que

identifica a rede como um todo. Nele, todos os bits que não fazem parte do prefixo de rede recebem o valor 0.

Já o último endereço é utilizado como endereço de **difusão** (*broadcast*). Roteadores não repassam mensagens de difusão, portanto, em uma rede IP a difusão fica limitada ao segmento de rede limitado pelo roteador. No endereço de difusão, todos os bits que não fazem parte do prefixo de rede recebem o valor 1.

O 255.255.255.255 é um endereço especial de difusão em que a mensagem é entregue a todos os demais hospedeiros que estão na mesma sub-rede do hospedeiro que enviou a mensagem.

No endereçamento de classes cheias, **o número total de redes e hospedeiros para cada classe é:**

## Difusão

Endereço lógico utilizado em redes de computadores que deve ser colocado no campo destino de uma mensagem quando se deseja que os dados sejam entregues a todos os hospedeiros da sub-rede.

## Classe A

126 redes com aproximadamente 16 milhões de hospedeiros cada.

## Classe B

16.384 redes com até 64K hospedeiros cada.

## Classe C

Cerca de 2 milhões de redes com até 254 hospedeiros cada.

## Classe D

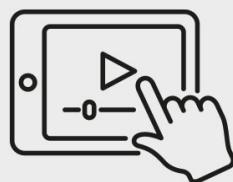
Cerca de 268 milhões de grupos multicast.



A distribuição de endereços IP é controlada pela ICANN (Internet Corporation for Assigned Names and Numbers).

## Sub-Redes

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



O mundo exterior a uma organização enxerga sua rede como sendo única, e nenhuma informação sobre sua estrutura interna é necessária. Porém, sem a utilização de sub-redes, o espaço de endereçamento pode se tornar muito ineficiente.

A fim de tornar mais eficiente a utilização da rede, é comum sua divisão em várias sub-redes. O mecanismo que permite tal divisão é conhecido como **máscara de rede**.

Assim como um endereço IP, uma máscara de rede possui 32 bits divididos em 4 campos com 8 bits cada, seguindo o padrão:



### Bits da rede - bit 1



### Bits da sub-rede - bit 1



### Bits do hospedeiro - bit 0

Os bits da sub-rede são utilizados para especificar quais bits no campo do hospedeiro são usados para especificar as sub-redes de uma rede.

A máscara padrão (default) para cada classe é dada por:

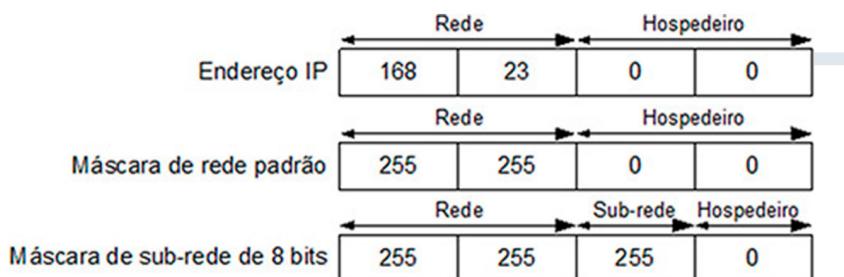
Classe	Decimal	Notação IDR
A	255.0.0.0	/8
B	255.255.0.0	/16

Classe	Decimal	Notação IDR
C	255.255.255.0	/24

A máscara padrão (default) para cada classe.

Fábio Contarini Carneiro.

Veja o exemplo abaixo:



Ao se dividir uma rede em sub-redes, deve-se alocar os bits para a sub-rede a partir dos bits de mais alta ordem (bits mais à esquerda) do campo do hospedeiro. A tabela a seguir mostra os valores usados no campo do hospedeiro quando se divide uma rede em sub-redes.

128	64	32
1	0	0
1	1	0
1	1	1
1	1	1
1	1	1
1	1	1
1	1	1

Campo do hospedeiro.

Fábio Contarini Carneiro.

Uma rede pode ser dividida em diversas partes para uso interno, continuando a ser vista como uma única rede externamente. Essas partes são as sub-redes.

Existem diversas razões para se dividir uma rede em sub-redes.

Algumas destas razões são:

- Isolar o tráfego de uma sub-rede, reduzindo assim o tráfego total da rede.
- Proteger ou limitar o acesso a uma sub-rede.
- Permitir a associação de uma sub-rede com um departamento ou espaço geográfico específico.



## Cálculo das máscaras de rede

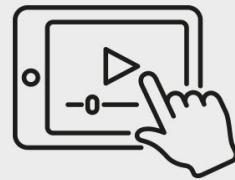
Para saber mais sobre como calcular máscaras de rede, assista ao vídeo.

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



## Vamos exercitar o cálculo de endereços?

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.

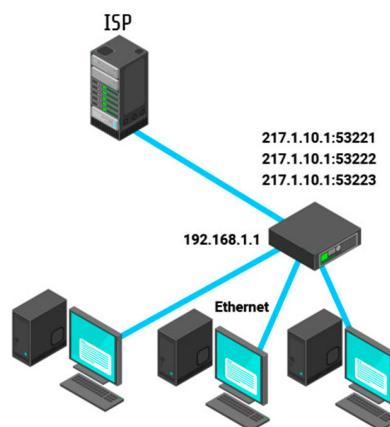


## Tradução de endereços

NAT

Atualmente, **endereços IP são escassos** e esse esgotamento não é um problema teórico que pode ocorrer em algum momento no futuro distante, ele já está acontecendo, e a solução atual para esse problema é o **NAT (Network Address Translation** – Tradução de Endereço de Rede), descrito na RFC 3022.

Com essa técnica, uma organização pode utilizar internamente uma faixa de endereços que não é válida na Internet, e quando é necessário fazer acesso externo, o dispositivo responsável pelo NAT faz a tradução do endereço da rede interna para o endereço válido da organização. Dessa forma, a organização poderá obter internamente uma quantidade maior de hospedeiros do que endereços disponíveis para utilização da Internet.



A ideia básica do NAT é atribuir a cada organização uma pequena quantidade de endereços IP para tráfego na Internet. Dentro da organização, todo computador obtém um endereço IP exclusivo (também conhecido como endereço IP privado) usado para roteamento do tráfego interno e, quando um pacote sai da organização e vai para a Internet, ocorre uma conversão do endereço.

Para tornar esse esquema possível, três intervalos de endereços IP foram declarados como endereços privativos (reservados) e as organizações podem utilizá-los internamente da maneira que quiserem, a única regra é que nenhum pacote contendo esses endereços pode aparecer na própria Internet. Os três intervalos reservados são:

Faixa	Máscara	Classe
10.0.0.0 a 10.255.255.255	255.0.0.0	A
172.16.0.0 a 172.31.255.255	255.255.0.0	B

Faixa	Máscara	Classe
192.168.0.0 a 192.168.255.255	255.255.255.0	C

Intervalos reservados.

Fábio Contarini Carneiro.

Dentro da organização, toda máquina tem um endereço exclusivo que não é válido na Internet. Quando um pacote deixa a organização, ele passa por uma caixa NAT (normalmente um roteador), que converte o endereço de origem no endereço IP válido da organização. Desse modo, o pacote poderá transitar sem problemas pela Internet, porém, a resposta do pacote voltará para o endereço IP válido da organização, e não para a máquina que fez tal requisição. Por causa disso, a caixa NAT deverá manter uma tabela na qual poderá mapear a máquina que enviou a requisição para Internet, de forma que, quando a resposta voltar, ela possa ser mapeada para a máquina correta.

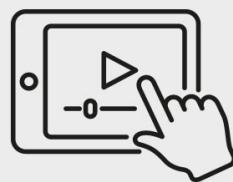
Na técnica de NAT, os endereços utilizados pelos hospedeiros de uma organização não são válidos na Internet, não há como tais equipamentos receberem acesso direto da rede externa, conferindo um certo grau de proteção aos hospedeiros da rede interna.



## Funcionamento do NAT

Para saber mais sobre o funcionamento do NAT, assista ao vídeo.

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



Falta pouco para atingir seus objetivos.

Vamos praticar alguns conceitos?

## Questão 1

No CIDR (*Classless Inter-domain Routing – Roteamento Interdomínio sem Classes*), o prefixo de rede é utilizado para

- A identificar a interface de rede do hospedeiro.
- B informar a quantidade de bytes do endereço IP.
- C identificar o hospedeiro que deve receber o datagrama.
- D determinar qual parte do endereço IP identifica a rede.
- E definir o default gateway da rede.

Parabéns! A alternativa D está correta.

Um endereço IP possui 32 bits. Quando se utiliza o CIDR, é definido um prefixo, sendo este utilizado para dividir o endereço IP em 2 partes: rede e hospedeiro. O prefixo CIDR determina a quantidade de bits a ser alocada na identificação da rede, ou seja, determina a parte do endereço IP que identifica a rede.

## Questão 2

A técnica de NAT foi desenvolvida para

- A realizar a fragmentação de datagramas IP.
- B calcular a máscara de rede.

**C**

associar uma sub-rede a um espaço geográfico específico.

**D**

resolver o problema da escassez de endereços IP.

**E**

realizar o roteamento de endereços públicos.

Parabéns! A alternativa D está correta.

A quantidade de endereços IP disponíveis para utilização na Internet já é escassa, e a cada dia mais organizações e pessoas desejam se conectar à Internet. Dessa forma, foi criada a técnica de NAT (*Network Address Translation*), que permite que uma organização possua um conjunto de endereços internos que não são válidos na Internet mas que, quando um hospedeiro deseja acessá-la, seu endereço é traduzido em um endereço válido, permitindo, assim, a possibilidade de conectar mais hospedeiros à Internet do que a quantidade de endereços disponíveis para isso.



### 3 - Protocolos de controle

Ao final deste módulo, você será capaz de classificar os protocolos de controle da Internet.

# Principais protocolos de controle

## Protocolo de controle da Internet

O protocolo IP é o responsável por manter a unidade de toda a Internet, porém, ele não consegue realizar sozinho todo o trabalho necessário. Para auxiliá-lo nessa tarefa, foram desenvolvidos protocolos auxiliares na camada de rede, conhecidos como protocolos de controle.

Neste módulo, estudaremos os principais protocolos de controle que trabalham em conjunto com o protocolo IP.

### ARP (*Address Resolution Protocol*)

Embora na Internet cada hospedeiro possua um ou mais endereços IP, na verdade eles não podem ser usados diretamente para o envio de informações pela rede, pois o hardware da camada de enlace de dados não entende os endereços Internet.

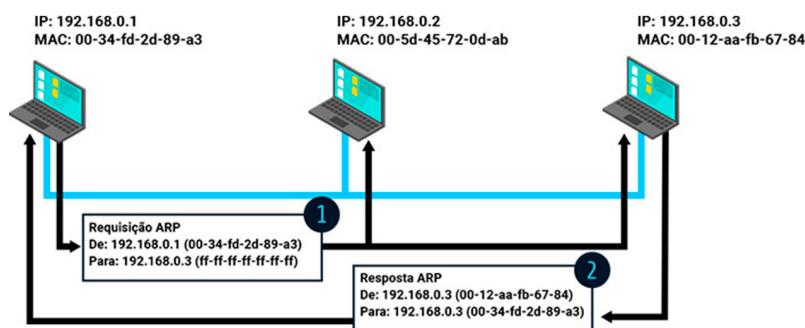


Suponha que você esteja utilizando um notebook e que ele enviará dados para um servidor na mesma rede. O software IP do notebook receberá os dados a serem enviados e construirá um pacote com o endereço IP do servidor no campo ENDEREÇO DESTINO do datagrama para transmissão. O software IP descobrirá que o servidor está em sua própria rede e que, portanto, deverá transmitir os dados diretamente a ele.

Neste ponto existe um problema. A placa de rede trabalha com endereços físicos da camada de enlace, ela não reconhece endereços IP. Logo, é necessário encontrar uma forma de mapear os endereços de rede nos endereços de enlace.

A solução adotada é fazer com que o hospedeiro que precise descobrir o endereço de enlace do destinatário envie um pacote por difusão perguntando: "A quem pertence o endereço IP A.B.C.D?". Então, o hospedeiro destinatário responderá com seu endereço de enlace. O protocolo que faz essa pergunta e obtém a resposta é o ARP (Address Resolution Protocol, que é definido na RFC 826).

No exemplo da imagem a seguir, a requisição ARP é enviada para o endereço de enlace ff-ff-ff-ff-ff-ff, o que permite que a camada de enlace promova uma difusão (entregue a requisição a todos os hospedeiros ligados ao meio de transmissão).



Continuando em nosso exemplo, o software IP do notebook constrói um quadro de enlace endereçado ao servidor, coloca o pacote IP no campo de carga útil e o envia. A placa de rede do destinatário detecta esse quadro, reconhece-o como um quadro destinado a ela e o recolhe. O software de enlace extraí o pacote IP da carga útil e o repassa para o software IP, que verifica se ele está corretamente endereçado e o processa.

**São possíveis várias otimizações para tornar o arp mais eficaz. Em primeiro lugar, depois que uma máquina executa o ARP ela armazena o resultado em um cache.**

Em muitos casos, o destinatário precisará enviar uma resposta, o que forçará também a execução do ARP para determinar o endereço de enlace do transmissor. Essa difusão do ARP pode ser evitada fazendo-se com que o destinatário inclua em sua tabela o mapeamento entre os endereços de rede e de enlace do transmissor.

Se um hospedeiro precisar enviar dados para outro hospedeiro que se encontra em uma sub-rede diferente da sua, esse sistema de entrega

direta não funcionará. Nesse caso, o hospedeiro transmissor precisará enviar seus dados para um roteador, que deverá providenciar o encaminhamento do datagrama para a sub-rede do destinatário. Poderá ser necessário que o protocolo ARP seja utilizado mais de uma vez ao longo do caminho.

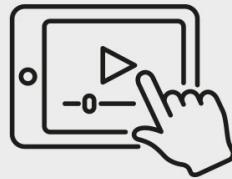
Para saber mais sobre o funcionamento do ARP quando existe um roteador no meio do caminho, assista ao vídeo.



## Protocolo ARP

Para saber mais sobre protocolo ARP, assista ao vídeo.

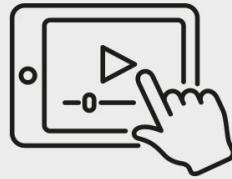
Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



## Configuração automática de endereços

DHCP (*Dynamic Host Configuration Protocol*)

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



Para ter acesso à Internet, um hospedeiro precisa ser configurado com alguns parâmetros.

Os parâmetros mínimos a serem configurados são:

## Endereço IP

Para permitir que o hospedeiro possa ser endereçado e receber dados de outro hospedeiro.

## Máscara de sub-rede

Para que o hospedeiro possa determinar qual a sua sub-rede.

## Endereço do gateway (roteador)

Para que o hospedeiro possa enviar informações para fora de sua sub-rede.

## Endereço do servidor DNS

Para que o hospedeiro possa acessar outros hospedeiros por seus nomes em vez de precisar conhecer seus endereços IP.

Esses parâmetros devem ser configurados em todos os hospedeiros da organização, o que pode ser feito manualmente pelos administradores da rede. Isso funcionaria muito bem em um ambiente relativamente estático onde os hospedeiros raramente fossem mudados de lugar ou trocados.

E no caso de ambientes onde existe uma grande variação nos hospedeiros presentes na rede?

Imagine, por exemplo, como seria na praça de alimentação de um shopping que oferece acesso à Internet. Seria viável solicitar que todo o dispositivo móvel que chegasse nesse shopping e quisesse acessar a Internet fizesse um pré-cadastro para poder ter esse acesso? Como cada dispositivo precisa de um endereço único no momento do acesso, como seria administrar essa distribuição por todos os possíveis clientes?



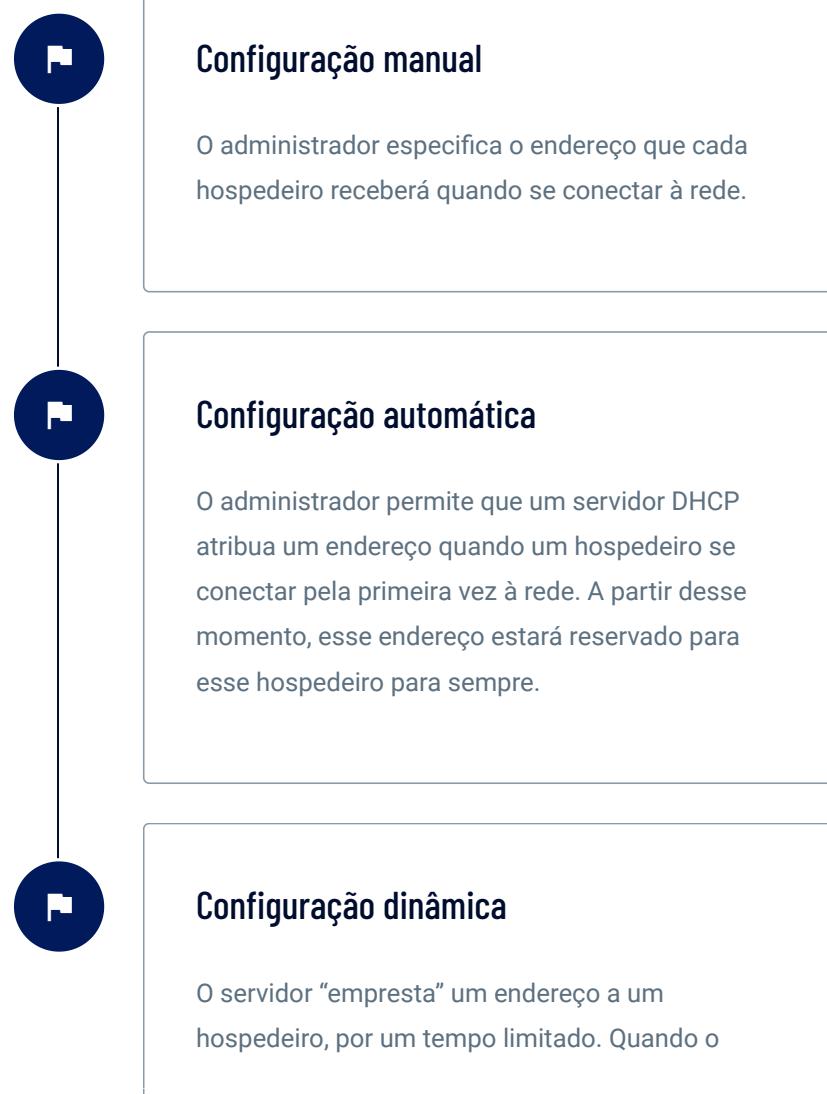
Para facilitar a distribuição de endereços IP e demais parâmetros de rede entre os vários hospedeiros que podem estar presentes em uma rede, e administrar a atribuição desses parâmetros de forma automatizada, a [IETF](#) desenvolveu o DHCP, cuja especificação encontra-se na RFC 2131.

Para utilizar o mecanismo dinâmico de alocação de endereços do DHCP, o administrador do sistema deve configurar um servidor fornecendo um grupo de endereços IP. Sempre que um novo computador se conecta à rede, entra em contato com o servidor e solicita um endereço. O servidor opta por um dos endereços especificados pelo administrador e aloca tal endereço para o computador.

De modo geral, o **DHCP permite três tipos de atribuição de endereços:**

## IETF

Internet Engineering Task Force (IETF) é um grupo internacional aberto que tem como objetivo identificar e propor soluções a questões relacionadas à utilização da Internet, além de propor padronização das tecnologias e dos protocolos envolvidos.



Camada de rede  
hospedeiro não estiver mais utilizando esse endereço, ele poderá ser alocado a outro hospedeiro.

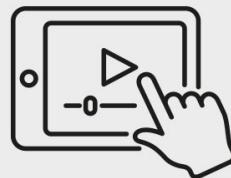
Um servidor DHCP empresta um endereço para o hospedeiro por um período de tempo determinado que ele especifica ao alocar o endereço. Durante esse período, o servidor não alocará o mesmo endereço a outro cliente, entretanto, quando a alocação terminar, o cliente deverá renová-la ou cancelar o uso do endereço.

O período ideal de alocação depende da rede e das necessidades de um hospedeiro em particular. O DHCP não especifica uma quantidade de tempo para o período de alocação. O protocolo permite que um cliente solicite um tempo de alocação específico e que um servidor informe ao cliente quanto tempo de alocação lhe será concedido. Fica a cargo do administrador determinar esse tempo.

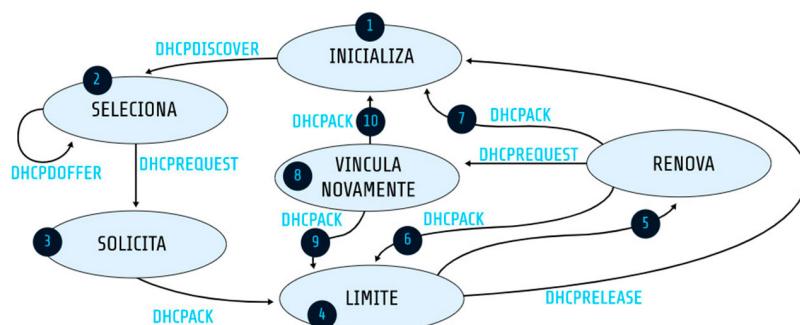


## Funcionamento do DHCP

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



A figura mostra os seis estados em que um cliente DHCP pode estar:



Entenda o que ocorre em cada etapa:



Inicializa

Quando um cliente faz a primeira inicialização, ele entra no estado INICIALIZA. Para dar início à aquisição de um endereço IP, o cliente primeiro entra em contato com todos os servidores DHCP da rede local difundindo uma mensagem DHCPDISCOVER, utilizando para isso a porta 67 do protocolo UDP, que é o protocolo de transporte da Internet sem conexões.



## Seleciona

Após, ele passa para o estado SELEICIONA. Os servidores da rede local recebem a mensagem e enviam, cada um, uma mensagem DHCPOFFER. As mensagens DHCPOFFER contêm informações de configuração e o endereço IP oferecido pelo servidor.

O cliente, assim, seleciona um dos servidores e lhe envia uma mensagem DHCPREQUEST, solicitando uma alocação.



## Solicita

Em seguida, vai para o estado SOLICITA. O servidor responde com a mensagem DHCPACK, dando início à alocação. O cliente segue então para o estado LIMITE, que é o estado normal de operação.



## Límite

Uma vez no estado LIMITE, o cliente configura três temporizadores que controlam a renovação, a revinculação e o fim da alocação. Um servidor DHCP pode especificar os valores explícitos para os temporizadores, caso contrário o cliente utiliza o padrão. O valor padrão para o primeiro

Camada de rede  
temporizador e a metade do valor do tempo total da alocação.



## Renova

Quando o temporizador ultrapassar o tempo determinado pela primeira vez, o cliente deverá tentar renovar a alocação. Para solicitar uma renovação, o cliente transmite uma mensagem DHCPREQUEST ao servidor que lhe fez a alocação. Em seguida, muda para o estado RENOVA para aguardar uma resposta.

O servidor pode responder de dois modos: instruindo o cliente a não utilizar mais o endereço ou autorizando-o a prolongar o uso.



## Limite

Caso aprove a renovação, o servidor transmite um DHCPACK, o que faz o cliente retornar ao estado LIMITE e continuar com o uso do endereço.



## Inicializa

Se um servidor não aprovar a renovação do endereço, ele envia um DHCPNACK, que faz com que o cliente interrompa imediatamente o uso do endereço e retorne ao estado INICIALIZA.



## Vincula novamente

Caso o cliente esteja no estado RENOVA e não chegue nenhuma resposta, provavelmente o servidor que concedeu a alocação estaria fora de

Camada de rede alcance. Para contornar essa situação, o DHCP conta com um segundo temporizador, que expira após 87,5% do período da alocação e faz com que o cliente mude o estado VINCULA NOVAMENTE.



## Limite

Ao fazer essa transição, o cliente passa a difundir a mensagem DHCPREQUEST para qualquer servidor da rede local. Caso receba uma resposta positiva, o cliente retorna ao estado LIMITE e reconfigura os temporizadores.



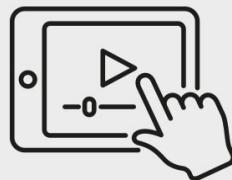
## Inicializa

Se a resposta for negativa, o cliente deve mudar para o estado INICIALIZA e interromper imediatamente o uso do endereço IP.

Quando um cliente não precisa mais da alocação, o DHCP permite que ele a encerre sem precisar esperar que o prazo da validade termine. Para isso, o cliente envia uma mensagem DHCPRELEASE ao servidor. Depois de enviar a mensagem, o cliente deixa o estado LIMITE e volta ao estado INICIALIZA.

## ICMP (*Internet Control Message Protocol*)

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



Durante o tráfego de informações pela rede, podem ocorrer **erros ou situações anormais** que precisam ser do conhecimento dos hospedeiros. Para permitir a troca desse tipo de informação, foi criado o **protocolo ICMP**. Com ele, hospedeiros conseguem enviar mensagens de

controle e erros para outros hospedeiros. O erro sempre é reportado ao sistema que gerou o datagrama.

Além de oferecer a possibilidade de troca de informações sobre erros e de controle, o ICMP inclui um mecanismo de envio e resposta (echo) para testar quando um destino é alcançável ou não. O comando ping utiliza esse protocolo.

O ICMP é definido na RFC 792.

Uma mensagem ICMP é carregada dentro de um datagrama IP, ou seja, **as mensagens ICMP são transportadas pelo protocolo IP**. Existem diversas mensagens ICMP. **As mais importantes são:**

- *Destination unreachable* (destino inalcançável): datagrama não pode ser entregue. Usada quando a sub-rede ou um roteador não pode localizar o destino, ou um datagrama com o bit DF não pode ser entregue porque há uma rede de pacotes pequenos no caminho.
- *Time exceeded* (tempo excedido): campo tempo de vida do IP chegou a zero. Enviada quando um pacote é descartado porque seu contador chegou a zero. Esse evento é um sintoma de que os datagramas estão entrando em loop, de que há congestionamento ou de que estão sendo definidos valores muito baixos para o temporizador.
- *Parameter problem* (problema de parâmetro): campo de cabeçalho inválido. Essa mensagem indica existência de um problema no software IP do hospedeiro transmissor ou no software de um roteador pelo qual o pacote transitou.
- *Source quench* (redução de origem): conhecida como pacote regulador. Usada para ajustar os hospedeiros que estejam enviando pacotes demais a fim de controlar o congestionamento na rede.
- *Redirect* (redirecionar): usada quando um roteador percebe que o pacote pode ter sido incorretamente roteado. É usada pelo roteador para informar ao hospedeiro transmissor a respeito do provável erro.
- *Echo request* (solicitação de eco): perguntar a um hospedeiro se ele está ativo. Ao receber a mensagem echo request, o destino deve enviar de volta uma mensagem echo reply.
- *Echo reply* (resposta de eco): confirmação de que o hospedeiro está ativo.

- *Timestamp request* (solicitação de carimbo de data e hora): mesmo que *echo request*, mas solicita que a hora da chegada da mensagem e da partida da resposta sejam registradas na mensagem de resposta.
- *Timestamp reply* (resposta de carimbo de data e hora): resposta à mensagem *timestamp request*.

Falta pouco para atingir seus objetivos.

Vamos praticar alguns conceitos?

### Questão 1

O DHCP é um protocolo que foi desenvolvido para

A

fazer o mapeamento entre o endereço de rede e o endereço de enlace.

B

traduzir o endereço interno de um hospedeiro em um endereço externo válido.

C

facilitar a configuração de rede dos hospedeiros de uma sub-rede.

D

permitir a configuração manual dos parâmetros de rede.

E

realizar a sinalização de erros na configuração de rede.

Parabéns! A alternativa C está correta.

A tarefa de configurar os parâmetros de rede em todos os hospedeiros de uma rede pode ser uma tarefa demorada,

principalmente para grandes redes. Dessa forma, o DHCP foi desenvolvido para facilitar o trabalho do administrador, o qual configura os parâmetros em um servidor, sendo esses parâmetros recebidos pelos hospedeiros de forma automática durante sua inicialização.

## Questão 2

Quando um datagrama não pode ser entregue porque o roteador não consegue localizar o destino, o protocolo responsável por informar este problema é o

A ICMP.

B DHCP.

C NAT.

D ARP.

E HTTP.

Parabéns! A alternativa A está correta.

O ICMP foi desenvolvido como o protocolo responsável pelo envio das mensagens de controle e de erro de uma rede. Assim, quando o roteador não consegue localizar o destino, o software ICMP do equipamento envia de volta à origem do datagrama uma mensagem ICMP contendo essa informação.



## 4 - Algoritmos de roteamento

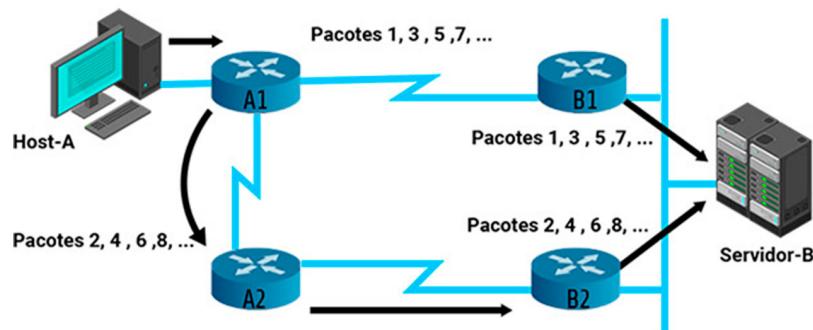
Ao final deste módulo, você será capaz de selecionar os algoritmos de roteamento.

# Encaminhando os pacotes na rede

## Roteamento

A principal função da camada de rede é **rotear pacotes do hospedeiro origem para o hospedeiro destino da melhor forma possível**. Na maioria dos casos, os pacotes necessitarão passar por vários roteadores para chegar ao destino.

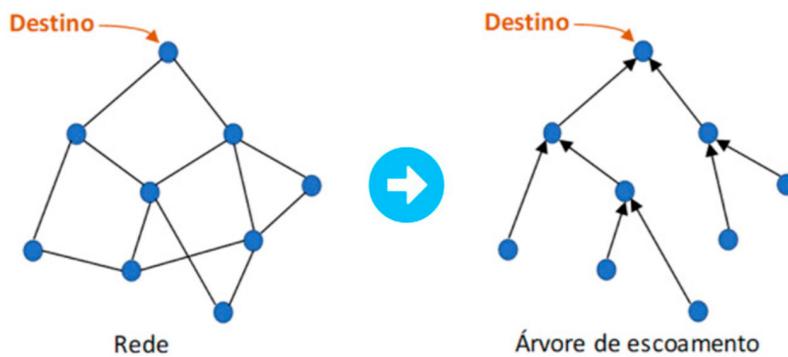
O algoritmo de roteamento é a parte do software da camada de rede responsável pela decisão sobre a linha de saída a ser usada na transmissão do pacote que entra. Caso a sub-rede utilize datagramas, essa decisão deverá ser tomada para todos os pacotes de dados recebidos e se utilizar circuitos virtuais internamente, as decisões de roteamento serão tomadas somente quando o circuito virtual estiver sendo estabelecido.



## Princípio da otimização

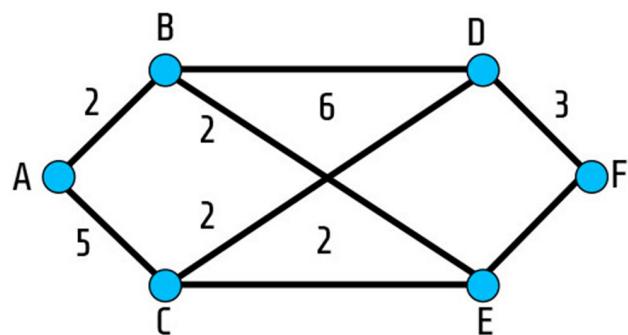
O princípio da otimização de Bellman (1957) estabelece que se o roteador B estiver no caminho ótimo entre o roteador A e o roteador C, o caminho ótimo de B a C também estará na mesma rota.

Esse princípio nos diz que o conjunto de rotas ótimas de todas as origens para um determinado destino é uma árvore conhecida como **árvore de escoamento** (*sink tree*, em inglês), cuja raiz é o hospedeiro destino. Os algoritmos de roteamento precisam descobrir a árvore de escoamento a partir da topologia da rede.



## Roteamento pelo caminho mais curto

O protocolo de roteamento pelo caminho mais curto é um protocolo **estático** cuja ideia é criar um grafo da sub-rede, com cada nó do grafo representando um roteador e cada arco indicando um enlace. Para escolher uma rota entre um determinado par de roteadores, o algoritmo simplesmente encontra o caminho mais curto entre eles no grafo.



Uma forma de medir o comprimento do caminho é em **número de saltos** (**quantidade de enlaces que devem ser utilizados**). Utilizando essa unidade métrica na figura anterior, os caminhos ABD e ABE são igualmente longos. Outra unidade métrica é a **distância geográfica**, caso em que ABD é mais longo que ABE.

Há também a unidade métrica baseada no tráfego entre os enlaces. Nesse grafo, o caminho mais curto é o caminho mais rápido, e não o caminho com menos arcos ou com menor distância.

Os valores dos arcos podem ser calculados como uma função da distância, da largura de banda, do tráfego médio, do custo de comunicação, do comprimento médio de fila, do retardo detectado ou de outros fatores. Alterando a função de atribuição de pesos, o algoritmo calcularia o caminho mais curto medido de acordo com determinados critérios que podem ser ou não combinados.

Existem diversos algoritmos para o cálculo do caminho mais curto. O mais conhecido deles foi desenvolvido por [Dijkstra](#), em 1959. Nele, cada roteador armazena sua menor distância até a origem e o caminho a ser seguido. Na inicialização do algoritmo não existe caminho conhecido, assim, a distância é marcada como infinito. Conforme o algoritmo progride, os caminhos e seus custos são encontrados.

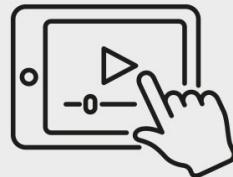
## Dijkstra

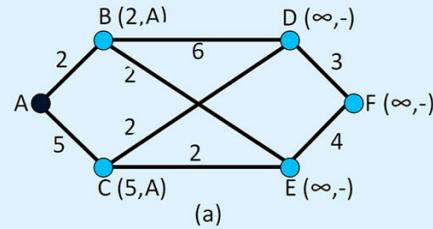
Edsger Wybe Dijkstra (Roterdã, 11 de maio de 1930 – Nuenen, 6 de agosto de 2002) foi um holandês, cientista da computação, conhecido por suas contribuições nas áreas de desenvolvimento de algoritmos e programas, de linguagens de programação (pelo qual recebeu o Prêmio Turing de 1972 por suas contribuições fundamentais), sistemas operacionais e processamento distribuído.



## Roteamento pelo caminho mais curto (Dijkstra)

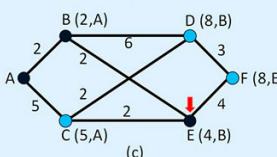
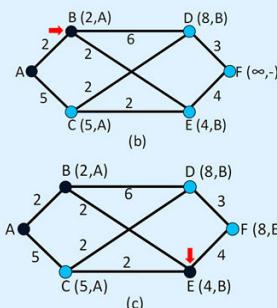
Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



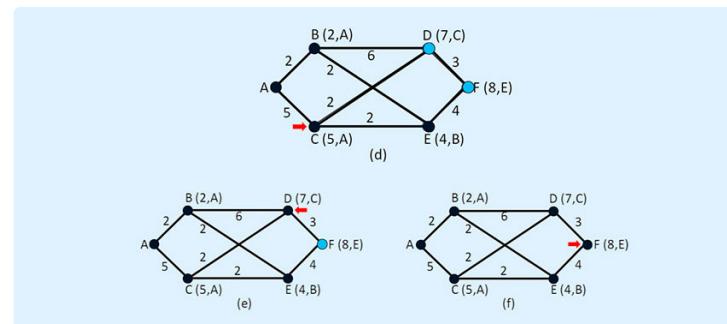


Para encontrar a árvore de escoamento de A, na figura, marca-se o nó A como permanente, o que é indicado por um círculo preenchido (a). Depois, é examinado, um a um, cada nó adjacente a A alterando o rótulo de cada um deles para a distância até A.

Sempre que um nó é rotulado novamente, ele também é rotulado com o nó a partir do qual o teste foi feito, assim, pode-se reconstruir o caminho final posteriormente. Após examinar cada nó adjacente a A, verifica-se todos os nós provisoriamente rotulados no grafo tornando permanente o de menor rótulo, que passa a ser o novo nó ativo.

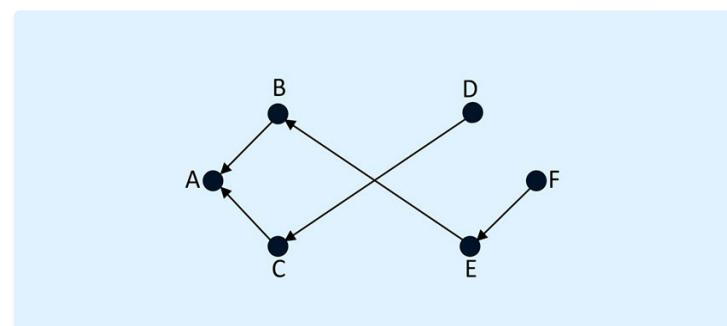


No segundo passo (b), verifica-se que o nó B é o que possui o menor valor entre os nós rotulados não permanentes. Logo, ele é marcado como permanente e os nós adjacentes são rotulados. Como B tem custo 2 e o enlace entre B e D possui custo 6, o custo de D passando por B será  $6 + 2 = 8$ . Da mesma forma, o custo de E passando por B será rotulado como 4. E assim o algoritmo prossegue, até que todos os nós sejam rotulados como permanentes.



Uma situação interessante acontece no quarto passo (d).

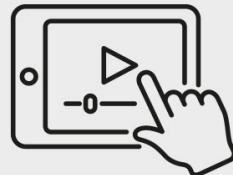
Quando o nó C é marcado como permanente, verifica-se que D anteriormente foi rotulado com valor 8 passando por B, porém, como vizinho de C, seu custo será 7. Então, o nó D é rotulado novamente com o valor 7.



Ao final, para obter a árvore de escoamento até A, basta seguir o caminho marcado por cada roteador até seu vizinho de menor custo.

## Algoritmos de roteamento

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



Conforme estudamos anteriormente, os **algoritmos de roteamento são executados pelos roteadores de uma sub-rede** para que sejam criadas as tabelas de repasse dos roteadores. Para isso, eles precisam trocar informações sobre o estado da rede e concordarem com a árvore de escoamento a ser montada para cada destino.

Os algoritmos de roteamento podem ser agrupados em:

**Não adaptativos**

Não baseiam suas decisões de roteamento em medidas ou estimativas do tráfego e da topologia atuais. A escolha da rota a ser utilizada é previamente calculada e transferida para os roteadores quando a rede é inicializada. Tal procedimento também é conhecido como roteamento estático.

**Adaptativos**

Mudam suas decisões de roteamento para refletir mudanças na topologia e/ou no tráfego. Tal procedimento também é conhecido como roteamento dinâmico.



Ainda, os algoritmos de roteamento podem ser classificados como algoritmos de roteamento globais ou descentralizados.

**Global**

Calcula o melhor caminho com base no conhecimento completo da rede. Para esse fim, o roteador deve obter informações sobre o estado de todos os roteadores e enlaces que compõem a sub-rede. Um exemplo desse tipo de roteamento é o roteamento de estado de enlace.

**Descentralizado**

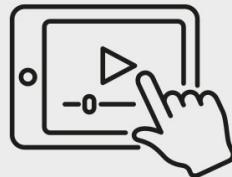
Neste algoritmo, nenhum roteador possui informação completa sobre o estado da rede. As rotas são calculadas com base em informações obtidas com roteadores vizinhos. Um exemplo desse tipo de roteamento é o roteamento de vetor de distância.



# Diferenças entre roteamento interno e externo

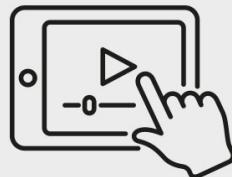
Para saber mais sobre a diferença entre roteamento interno e externo, assista ao vídeo.

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



## Roteamento de vetor de distâncias

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



O **roteamento de vetor de distância** é um algoritmo dinâmico que opera fazendo com que cada roteador mantenha uma tabela que fornece a melhor distância conhecida a cada destino e determina qual linha deve ser utilizada para se chegar lá. Essas tabelas são atualizadas por meio da troca de informações com os vizinhos.

Trata-se do algoritmo de roteamento original da **ARPANET** que também foi utilizado na internet com o nome rip. alguns roteadores utilizam protocolos de vetor de distância mais aperfeiçoados.

## ARPANET

Precursora da Internet: primeira rede a implementar o conjunto de protocolos TCP/IP.

### Comentário

Cada roteador mantém uma tabela de roteamento indexada por cada roteador da sub-rede. Cada entrada contém duas partes: a linha de saída preferencial a ser utilizada para esse destino e uma estimativa do tempo ou distância até o destino. A unidade métrica utilizada pode ser o número de hops, o retardo de tempo, o número total de pacotes enfileirados no caminho ou algo semelhante.

Presume-se que o roteador conheça a distância até cada um de seus vizinhos. Se a unidade métrica for contagem de saltos, a distância será de apenas um salto. Se a unidade métrica for o comprimento da fila, o roteador examinará cada uma das filas. Se a unidade métrica for o retardo, o roteador poderá medi-lo com pacotes “ICMP ECHO”.

### E como saber qual o caminho que possui melhor distância apenas obtendo informações dos vizinhos?

Seja  $dv(y)$  o custo do caminho de menor custo do roteador  $v$  para o roteador  $y$ , e seja  $c(x,v)$  o custo da ligação entre os roteadores  $x$  e  $v$ . Seja ainda  $v$  um vizinho do roteador  $x$ .

Então, o custo do caminho para ir de  $x$  a  $y$  passando pelo vizinho  $v$  será  $\{c(x,v) + dv(y)\}$ .

Logo, pelo princípio da otimização, o menor custo para ir de  $x$  a  $y$  será:

$$dx(y) = \min\{c(x,y) + dv(y)\},$$

onde  $\min\{ \cdot \}$  retornará o menor valor calculado para todos os vizinhos.

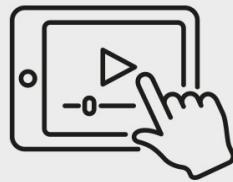
Conforme os vizinhos vão trocando informações sobre o estado da rede, o algoritmo converge para um **estado de equilíbrio** onde são obtidas as árvores de escoamento para cada destino.

Sempre que houver alguma modificação na rede, o roteador comunicará essa mudança a seus vizinhos, que atualizarão suas tabelas e repassarão essas alterações a seus próprios vizinhos. Assim, a alteração se propagará entre todos os roteadores e rapidamente a rede convergirá para um novo estado de equilíbrio.

Na Internet, o roteamento de vetor de distância é implementado pelo **RIP (Routing Information Protocol)**. A versão 1 do RIP é definida pela RFC 1058, enquanto sua versão 2 é definida pela RFC 2453.

### Roteamento de estado de enlace

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.



O roteamento de vetor de distância era utilizado na ARPANET até 1979, quando então foi substituído pelo **roteamento de estado de enlace**. Essa substituição foi basicamente motivada por dois problemas:

- A unidade métrica de retardo era o comprimento de fila, não levando em conta a largura de banda.
- O algoritmo geralmente levava muito tempo para convergir.

A ideia do roteamento de estado de enlace é simples. Cada roteador deve:

#### Descobrir seus vizinhos e aprender seus endereços de rede

Determinação dos vizinhos: quando um roteador é inicializado, ele envia um pacote HELLO em cada linha. O roteador da outra extremidade envia de volta uma resposta identificando-se.

#### Medir o retardo ou o custo para cada um de seus vizinhos

Medição do custo da linha: a forma mais simples de determinar o retardo é enviar um pacote ECHO pela linha. Medindo o tempo de ida e volta, o roteador pode obter uma estimativa razoável do retardo. Para obter resultados melhores, pode-se fazer o teste mais de uma vez e usar a média.

#### Criar um pacote que diga tudo o que acaba de ser aprendido

Criação de pacotes por estado de enlace: o pacote começa com a identidade do transmissor, seguida do número de sequência, da idade, e de uma lista de vizinhos. É fornecido o custo referente a cada vizinho. Um exemplo de sub-rede é ilustrado na figura a seguir, sendo os custos mostrados por linhas. Os pacotes de estado de enlace correspondentes a todos os seis roteadores são mostrados na figura seguinte.

#### Enviar esse pacote a todos os outros roteadores

Distribuição dos pacotes de estado de enlace: se estamos querendo criar as rotas para que os datagramas possam ser enviados pela rede, como distribuir as informações? A ideia é usar a inundação de pacotes, na qual os pacotes que são

recebidos por uma interface são replicados para as outras interfaces, fazendo, assim, com que a informação se espalhe por toda a rede.

Para controlar a inundação, cada pacote contém um número de sequência que é incrementado para cada pacote enviado.

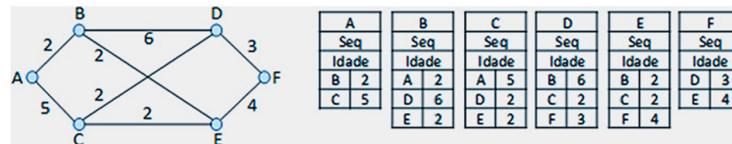
Quando é recebido, o novo pacote de estado de enlace é conferido na lista de pacotes já verificados. Se for novo, o pacote será encaminhado a todas as linhas, menos para aquela em que chegou. Se for uma cópia, o pacote será descartado. Se for recebido um pacote com número de sequência inferior ao mais alto detectado até o momento, ele será rejeitado.

Mas se um roteador apresentar falha, ele perderá o controle de seu número de sequência. A solução é incluir a idade de cada pacote após o número de sequência e decrementá-la uma vez por segundo. Quando a idade atingir zero, as informações desse roteador serão descartadas. O campo de idade é também decrementado por cada roteador durante o processo inicial de inundação para garantir que nenhum pacote viverá por um período indefinido.

#### Calcular o melhor caminho para cada um dos outros roteadores

Cálculo das novas rotas: uma vez que um roteador tenha acumulado um conjunto completo de pacotes de estado de enlace, poderá criar o grafo completo da sub-rede. Nesse momento, o algoritmo de Dijkstra pode ser executado localmente para calcular o melhor caminho.

Na Internet, o roteamento de estado de enlace é implementado pelo protocolo OSPF (*Open Shortest Path First*), definido pela RFC 2328.



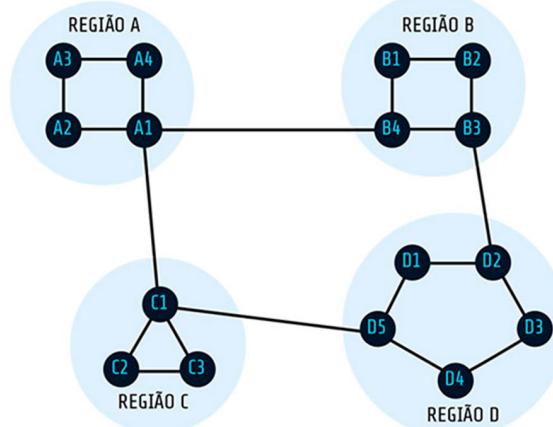
Assim, a topologia completa e todos os retardos são medidos e distribuídos para cada roteador. Em seguida, o algoritmo de Dijkstra pode ser usado para encontrar o caminho mais curto.

## Roteamento hierárquico

Conforme as redes aumentam, as tabelas de roteamento crescem proporcionalmente até que chega um momento em que **não é mais possível que os roteadores conheçam todos os caminhos da rede**. Assim, é necessário pensar em uma nova forma de realizar o roteamento. Foi pensando nessa situação que foi criado o **roteamento hierárquico**.

Quando o roteamento hierárquico é utilizado, **os roteadores são divididos em regiões**, com cada roteador conhecendo todos os detalhes sobre como rotear pacotes para destinos dentro de sua própria região, mas sem conhecer a estrutura interna de outras. No caso de redes muito grandes, uma hierarquia de dois níveis pode ser insuficiente, sendo necessário agrupar as regiões em clusters, os clusters em zonas, as zonas em grupos etc.

A imagem abaixo mostra um **exemplo quantitativo do roteamento em uma hierarquia de dois níveis com quatro regiões**. A tabela de roteamento completa do roteador A1 tem 16 entradas, mas quando o roteamento é feito hierarquicamente são necessárias apenas 7 entradas. Porém, esse tipo de roteamento pode implicar um provável aumento do caminho para alguns hospedeiros.



Entenda cada região nas tabelas abaixo:

Tabela completa

Destino	Linha	Saltos
A1	--	--
A2	A2	1
A3	A2	2
A4	A4	1

**Tabela completa**

B1	B4	2
B2	B4	3
B3	B4	2
B4	B4	1
C1	C1	1
C2	C1	2
C3	C1	2
D1	C1	3
D2	B4	3
D3	B4	4
D4	C1	3
D5	C1	2

Tabela completa.

Fábio Contarini Carneiro.

**Tabela hierárquica**

Destino	Linha	Saltos
A1	--	--
A2	A2	1
A3	A2	2
A4	A4	1
B	B4	1
C	C1	1

### Tabela hierárquica

D	C1	2
---	----	---

Tabela hierárquica

Fábio Contarini Carneiro.

Na Internet, o roteamento hierárquico é implementado por intermédio do protocolo BGP (*Border Gateway Protocol*), definido pela RFC 4271.

Falta pouco para atingir seus objetivos.

Vamos praticar alguns conceitos?

#### Questão 1

Um algoritmo de roteamento adaptativo possui como característica

A

adaptar suas interfaces de acordo com a vazão dos enlaces.

B

calcular as rotas a serem utilizadas com base no conhecimento completo da rede.

C

basear suas decisões de roteamento de acordo com o tráfego e a topologia recentes.

D

distribuir as informações de roteamento por intermédio de inundação de pacotes.

E

Garantir que as tabelas de repasse não sejam modificadas.

Parabéns! A alternativa C está correta.

Um algoritmo de roteamento adaptativo, conforme seu nome indica, adapta as rotas ao estado atual de enlaces e topologia, promovendo alterações na rota sempre que houver alguma mudança significativa no estado da rede.

## Questão 2

Quando a rede cresce a ponto de ser necessário alterar o esquema de roteamento para diminuir o tamanho das tabelas, deve-se utilizar o roteamento

A de vetor de distância.

B hierárquico.

C não adaptativo.

D ARP.

E estático.

Parabéns! A alternativa B está correta.

O roteamento hierárquico foi criado para resolver os problemas de armazenamento das tabelas de roteamento quando as redes crescem muito. Com sua utilização, não é necessário que os roteadores conheçam a topologia de toda a rede para tomar suas decisões de roteamento, bastando saber como rotear dentro de sua região e como chegar às outras regiões.

# Considerações finais

Acabamos de estudar a camada de rede do Modelo OSI e sua implementação mais importante, a camada de rede da arquitetura TCP/IP.

Vimos que o protocolo IP é o responsável por manter a Internet unida, promovendo um endereçamento universal aceito por todos os equipamentos que possuem acesso à Internet. Vimos ainda que é na camada de rede que é executado o algoritmo de roteamento, responsável pela montagem das tabelas de repasse dos roteadores, permitindo que esses equipamentos possam encaminhar os datagramas até seu destino, não importando o quanto distante esteja.

Podemos então concluir que essa camada é uma das mais importantes, responsável por toda a conectividade da Grande Rede, a Internet.

## Podcast

Neste podcast, o especialista responde perguntas relacionadas ao tema Camada de rede.

Para ouvir o *áudio*, acesse a versão online deste conteúdo.



## Explore +

Para saber mais sobre os assuntos explorados neste tema, leia:

- IP versão 6, a nova implementação do protocolo IP, KUROSE, J. F.; ROSS, K. W. **Redes de computadores e a internet**: uma abordagem top-down. 6. ed. São Paulo: Pearson, 2013. cap. 4.4.4.

- Implementação do OSPF, o algoritmo de roteamento interno oficial da Internet, KUROSE, J. F.; ROSS, K. W. **Redes de computadores e a Internet**: uma abordagem top-down. 6. ed. São Paulo: Pearson, 2013. cap. 4.6.2.

## Referências

BELLMAN, R.E. **Dynamic Programming**. Princeton, NJ: Princeton University Press, 1957.

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

COULOURIS, G. et al. **Sistemas distribuídos – conceitos e projeto**. 5. ed. Porto Alegre: Bookman, 2013.

FOROUZAN, B. A. **Comunicação de dados e redes de computadores**. 4. ed. Porto Alegre: AMGH, 2010.

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

SILBERSCHATZ, A.; GALVIN, P. B. **Sistemas operacionais – conceitos**. 5. ed. São Paulo: Prentice Hall, 2000.

SOARES, L. F. G. et al. **Redes de computadores – das LANs, MANs e WANS às redes ATM**. 2. ed. Rio de Janeiro: Campus, 1995.

STANEK, W. R., **Windows Server 2008**: guia completo. Porto Alegre: Bookman, 2009.

TANENBAUM, A. S.; WETHERALL, D. **Redes de computadores**. 5. ed. São Paulo: Pearson, 2011.



## Material para download

Clique no botão abaixo para fazer o download do conteúdo completo em formato PDF.

Download material

O que você achou do conteúdo?



Relatar problema