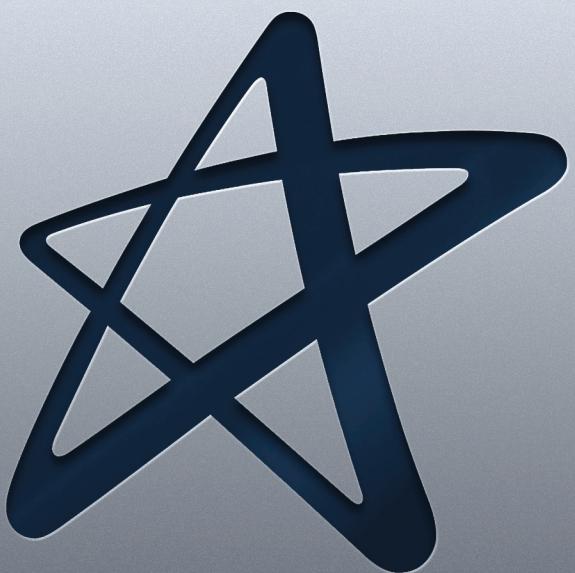


Tecnologias de Redes



Cruzeiro do Sul Virtual
Educação a distância

Material Teórico



Sub-Redes e Camada de Transporte

Responsável pelo Conteúdo:

Prof. Esp. Antoni Eduardo Marques da Silva

Revisão Textual:

Prof. Esp. Claudio Pereira do Nascimento



- Segmentação de Rede;
- Comunicação Entre Sub-Redes;
- VLSM e Seus Benefícios;
- A Camada de Transporte;
- TCP – Transmission Control Protocol;
- UDP – User Datagram Protocol;
- Endereçamento TCP e UDP (Portas).



OBJETIVO DE APRENDIZADO

- Compreender e apresentar como segmentar redes locais utilizando a técnica de endereçamento de sub-nets, entender a camada de transporte;
- Entender o funcionamento de protocolos orientados a conexão e sem orientação a conexão.



Orientações de estudo

Para que o conteúdo desta Disciplina seja bem aproveitado e haja maior aplicabilidade na sua formação acadêmica e atuação profissional, siga algumas recomendações básicas:

Determine um horário fixo para estudar.

Mantenha o foco! Evite se distrair com as redes sociais.

Procure manter contato com seus colegas e tutores para trocar ideias! Isso amplia a aprendizagem.

Seja original! Nunca plágie trabalhos.

Aproveite as indicações de Material Complementar.

Conserve seu material e local de estudos sempre organizados.

Não se esqueça de se alimentar e de se manter hidratado.

Assim:

- ✓ Organize seus estudos de maneira que passem a fazer parte da sua rotina. Por exemplo, você poderá determinar um dia e horário fixos como seu “momento do estudo”;
- ✓ Procure se alimentar e se hidratar quando for estudar; lembre-se de que uma alimentação saudável pode proporcionar melhor aproveitamento do estudo;
- ✓ No material de cada Unidade, há leituras indicadas e, entre elas, artigos científicos, livros, vídeos e sites para aprofundar os conhecimentos adquiridos ao longo da Unidade. Além disso, você também encontrará sugestões de conteúdo extra no item **Material Complementar**, que ampliarão sua interpretação e auxiliarão no pleno entendimento dos temas abordados;
- ✓ Após o contato com o conteúdo proposto, participe dos debates mediados em fóruns de discussão, pois irão auxiliar a verificar o quanto você absorveu de conhecimento, além de propiciar o contato com seus colegas e tutores, o que se apresenta como rico espaço de troca de ideias e de aprendizagem.

Segmentação de Rede

No passado, na implementação de redes IPv4 era muito comum que todos os dispositivos fossem incluídos em uma única rede, o que acarretava alguns problemas de performance dependendo da quantidade de hosts instalados. Esse tipo de implementação era conhecido como um projeto de rede linear. Em redes pequenas esse tipo de implementação não trazia maiores problemas, mas conforme as redes iam crescendo, tal modelo poderia trazer problemas mais graves.

Como já sabemos, as redes LAN baseadas no protocolo Ethernet, são redes onde os dispositivos utilizam broadcasts para identificar os serviços de dispositivos necessários para a comunicação. Em uma rede baseada em broadcast um único endereço de broadcast é enviado para todos os dispositivos contidos em uma mesma rede. Um protocolo exemplo que transmite dados em broadcast seria o DHCP - Protocolo de Configuração Dinâmica de Host, ou seja, para que os dispositivos cliente possam localizar o servidor DHCP eles precisam enviar um broadcast de descoberta, afim de encontrá-lo.

Em grandes redes, esse tipo de solicitação via broadcast pode acarretar uma quantidade significativa de tráfego de rede, que por ventura deixariam lentas as comunicações e operações de rede entre os dispositivos dessa rede, pois todos os dispositivos que aceitam esse broadcast devem processá-lo para identificar que tipo de requisição havia recebido. Se isso acontecer ele poderá até mesmo retardar operações de aplicações desses dispositivos. Em função disso, é muito importante segmentarmos essas grandes redes em redes menores (sub-redes), para que possam retardar tais problemas.

A ideia da segmentação de redes é a divisão de uma rede grande em redes menores. Essas redes por sua vez são mapeadas usando a técnica de divisão de sub-redes. Com isso, os administradores de rede podem agrupar os hosts dentro das sub-redes e defini-las por exemplo como sendo uma localização, um departamento, ou até mesmo uma área de dispositivos, como por exemplo uma rede específica de impressoras ou de pontos de acesso a redes sem fio. Essas divisões com certeza melhoram o desempenho da rede em função da redução de tráfego total entre os dispositivos em uma grande rede.

Comunicação Entre Sub-Redes

Da mesma forma que uma rede tradicional, para que as sub-redes possam se comunicar entre si, é necessário a inclusão de uns dispositivos de roteamento, ou seja, um roteador ou até mesmo um firewall com as funções de roteamento habilitadas. Esse rotador interliga uma rede LAN através de sua interface de gateway padrão. Isso ocorre porque o tráfego que é destinado a um host que tem como intui se comunicar fora da rede local será processado pelo roteador e é enviado para o determinado destino.

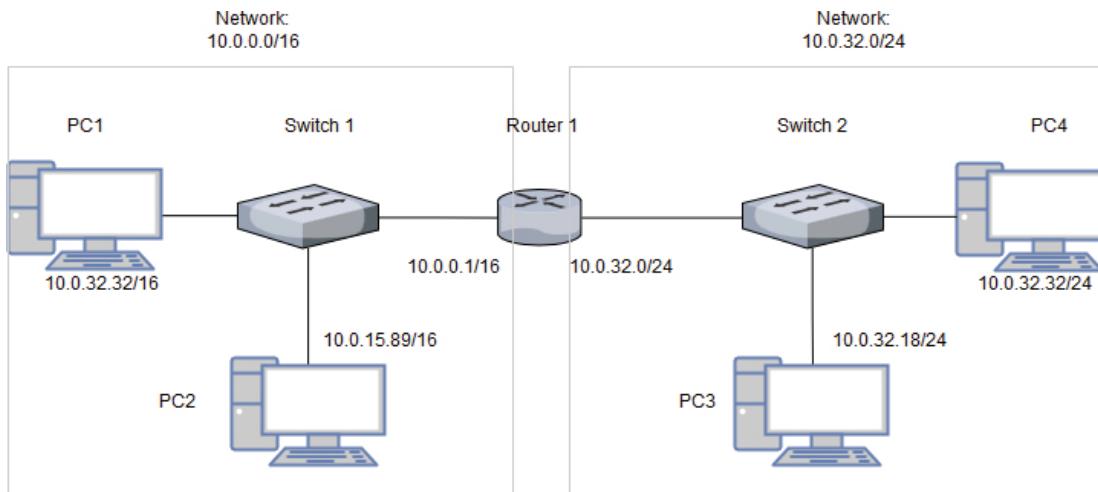


Figura 1 – Comunicação entre Sub-Redes

Quando dividimos espaços em sub-redes, o funcionamento é exatamente o mesmo se comparado com a segmentação em redes genéricas, ou seja, temos endereços dos hosts e do gateway padrão pertencentes a essa sub-rede calculada.

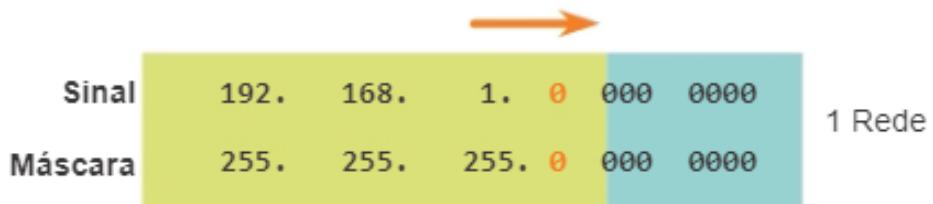
É importante lembrar que o tráfego de rede não pode ser encaminhado entre duas ou mais sub-redes sem o uso de um roteador, sendo que cada interface do roteador deve ter um endereço IPv4 que pertença à rede ou sub-rede na qual a interface está conectada.

Divisão de Sub-Redes

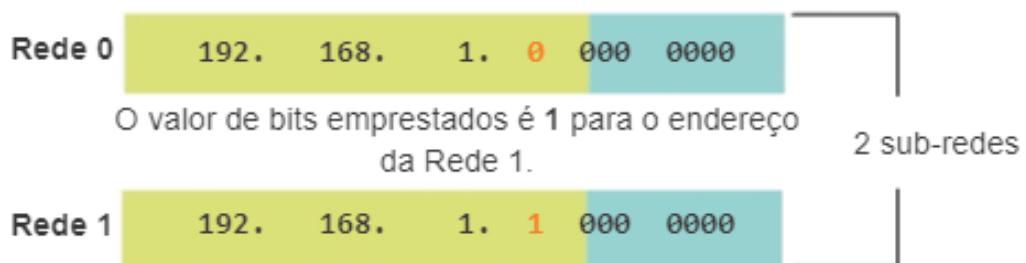
Quando é realizada uma divisão em sub-redes, temos o endereço que identifica a rede como um todo, o endereço que identifica o broadcast dessa rede e o intervalo de endereços que tem como função identificar os dispositivos de redes pertencentes a rede ou sub-rede calculada. O prefixo e a máscara de sub-rede possuem modos diferentes para representar a porção de rede e a porção host de um determinado endereço.

Para que possamos criar sub-redes de uma rede mestra, é necessário primeiro entender a classe padrão da rede a ser subnetada, para que possamos identificar os octetos reservados para o HOST.ID, ou seja, os intervalos que definem os hosts de uma rede. Sabendo isso, pegaremos emprestado os bits do intervalo de host do endereço, seguindo uma ordem da esquerda para a direita e sem pular bits.

Quanto mais bits de host foram emprestados, mais sub-redes poderão ser criadas, ou seja., para cada bit emprestado, dobramos o número de sub-redes disponíveis. Por exemplo, se 1 bit é emprestado, 2 sub-redes podem ser criadas. Se 2 bits, 4 sub-redes são criadas, se 3 bits forem emprestados, 8 sub-redes são criadas, e assim por diante. Contudo, com cada bit que pegamos emprestado, menos endereços de host ficam disponíveis por sub-rede. (CISCO NETACAD, 2017)



O valor de bits emprestados é **0** para o endereço da Rede 0.



As novas sub-redes têm a **MESMA** máscara de sub-rede.



Figura 2 – Criação de Sub-Redes

Como verificamos, apenas os bits porção de host do endereço podem ser emprestados para a criação de sub-redes. A porção de rede do endereço é atribuída exclusivamente pelo provedor de serviços e não pode ser alterada. A rede 192.168.1.0/24 tem 24 bits na porção de rede e 8 bits na porção de host, que é indicado com a máscara de sub-rede 255.255.255.0 ou a notação de /24. Sem a divisão de sub-rede, essa rede oferece suporte a uma única rede local, que pode ser composta por 254 dispositivos ativos, um endereço que identifica a rede e outro que identifica os endereços de broadcast, que tem como função alcançar todos os dispositivos dessa rede.

Na figura é emprestado do bit mais significativo (bit mais à esquerda) na porção de host, estendendo, assim, a porção de rede para 25 bits. Isso cria 2 sub-redes identificadas usando um 0 no bit emprestado da primeira rede e o 1 no bit emprestado da segunda rede. A máscara de sub-rede de ambas as redes usa um 1 na posição de bit emprestado para indicar que esse bit é agora porção de rede. Quando convertermos o octeto binário para decimal vemos que o primeiro endereço de sub-rede é 192.168.1.0 e o segundo endereço de sub-rede é 192.168.1.128. Como um bit foi emprestado, a máscara de sub-rede de cada sub-rede é 255.255.255.128 ou /25. (CISCO NETACAD, 2017)

Sinal	192.	168.	1.	0	000	0000	Rede: 192.168.1.0/24
Máscara	255.	255.	255.	0	000	0000	Máscara: 255.255.255.0
O empréstimo de 1 bit cria 2 sub-redes com a mesma máscara.							
Rede 0	192.	168.	1.	0	000	0000	Rede: 192.168.1.0/25
Máscara	255.	255.	255.	1	000	0000	Máscara: 255.255.255.128
Rede 1	192.	168.	1.	1	000	0000	Rede: 192.168.1.128/25
Máscara	255.	255.	255.	1	000	0000	Máscara: 255.255.255.128

Figura 3 – Criação de Sub-Redes

Sub-Redes em Uso

No exemplo anterior, a rede 192.168.1.0/24 foi dividida em sub-redes para criar duas sub-redes:

- 192.168.1.0/25;
- 192.168.1.128/25.

Baseando-se na próxima figura, observamos que o roteador R1 possui dois segmentos da LAN conectados às suas interfaces de GigabitEthernet. Essas sub-redes criadas serão utilizadas nos segmentos conectados a essas interfaces, afim de agirem como endereços de gateways dos dispositivos nas redes locais segmentadas. Nas interfaces do roteador deve-se ter atribuído um endereço IP dentro da faixa de endereços válidos da sub-rede calculada e determinada para o uso. Como melhor prática é muito comum utilizar o primeiro ou último endereço disponível (conhecido como primeiro ou último endereço válido) do intervalo de rede calculada. Podemos definir um endereço válido, como sendo um endereço que podemos aplicar em uma interface de rede física ou virtual, não sendo o endereço dedicado a rede e nem o endereço dedicado ao broadcast da rede ou sub-rede a ser utilizada.

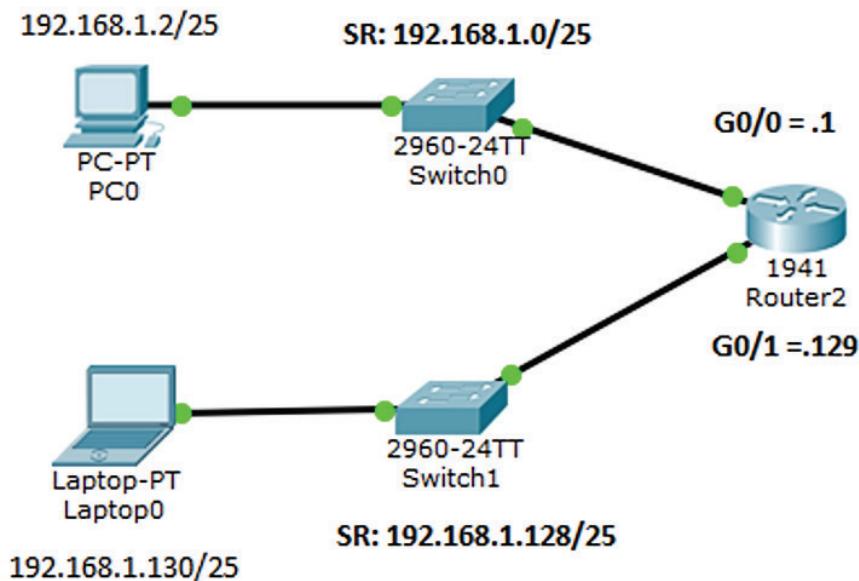


Figura 4 –Sub-Redes em uso

Fonte: Acervo do Conteudista

A primeira sub-rede, 192.168.1.0/25, é usada para a rede conectada à GigabitEthernet 0/0 e a segunda sub-rede, 192.168.1.128/25, é usada para a rede conectada à GigabitEthernet 0/1. Para atribuir um endereço IP em cada uma dessas interfaces, é necessário determinar o intervalo de endereços IP válidos para cada sub-rede. (CISCO NETACAD, 2017)

A seguir estão as diretrizes para cada uma das sub-redes:

- **Endereço de Rede** – Todos os bits em 0 na porção host do endereço.
- **Primeiro Endereço de Host** – Todos os bits em 0 mais um bit em 1 mais à direita na porção host do endereço.
- **Último Endereço de Host** – Todos os bits em 1 mais um bit em 0 mais à direita na porção host do endereço.
- **Endereço de Broadcast** – Todos os bits em 1 na porção host do endereço.

Fórmula de Divisão de Sub-Redes

Fórmula utilizada para a criação de sub-redes:

2^n (onde n = número de bits pegos emprestados)

$2^1 = 2$ sub-redes

Fórmula de Divisão de Hosts

Fórmula utilizada para a criação de hosts dentro de uma rede:

2^n (onde n = número de bits que restam no campo do host)

$2^7 = 128$

Como podemos verificar, após criamos uma sub-rede, dois desses endereços tornam-se inválidos, pois serão utilizados para a identificação da sub-rede criada e do endereço atribuído para o broadcast dessa sub-rede. Isso quer dizer que cada uma das sub-redes criadas conforme nossos cálculos possuem 126 (128-2) endereços de host válidos.

VLSM e Seus Benefícios

Quando criamos uma sub-rede tradicionalmente, as vezes podemos ter endereços de hosts que não serão devidamente utilizados para o respectivo endereçamento. Por exemplo: dada uma rede 192.168.10.0/27, ou seja, pegamos emprestado da rede genérica de classe C 192.168.10.0 a quantidade de 3 bits que estavam no campo de host, podemos ter cerca de 6 redes válidas, sendo que cada sub-rede calculada pode ter 30 hosts válidos em cada uma. Se usarmos essas sub-redes em uma conexão ponto-a-ponto, necessitamos apenas de dois hosts ativos, desperdiçando 28 endereços de rede devidamente calculados quando as sub-redes foram criadas.

Baseando-se na forma clássica de criação de sub-redes, podemos pegar uma sub-rede em específico e criarmos uma segmentação de segundo nível, ou seja, pegar uma sub-rede e segmenta-la novamente, afim de obtermos uma quantidade de redes maiores com menos hosts em cada uma delas.

Por exemplo, na topologia apresentada na próxima figura exige sete sub-redes, uma para cada uma das quatro LANs e uma para cada uma das três conexões WAN ponto-a-ponto entre os roteadores. Usando a divisão em sub-redes tradicionalmente com o dado endereço 192.168.20.0/24, 3 bits podem ser emprestados da porção de host no quarto octeto dedicado a host (último octeto) para atender aos requisitos de sub-rede das sete sub-redes.

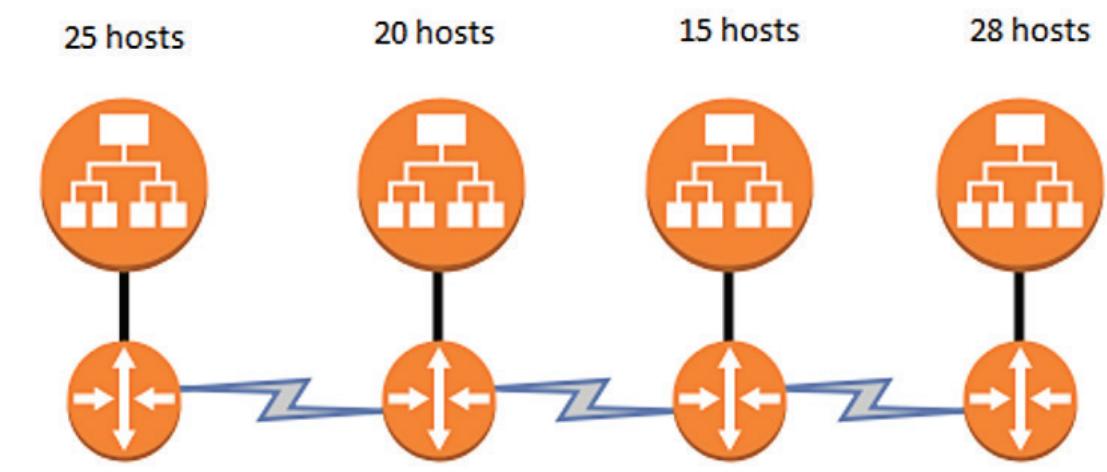


Figura 5 – Topologia de rede necessita dimensionamento

Fonte: Acervo do Conteudista

A aplicação de um esquema de endereçamento padrão de sub-rede ao cenário apresentado não é muito eficiente e resulta em desperdício de endereços.

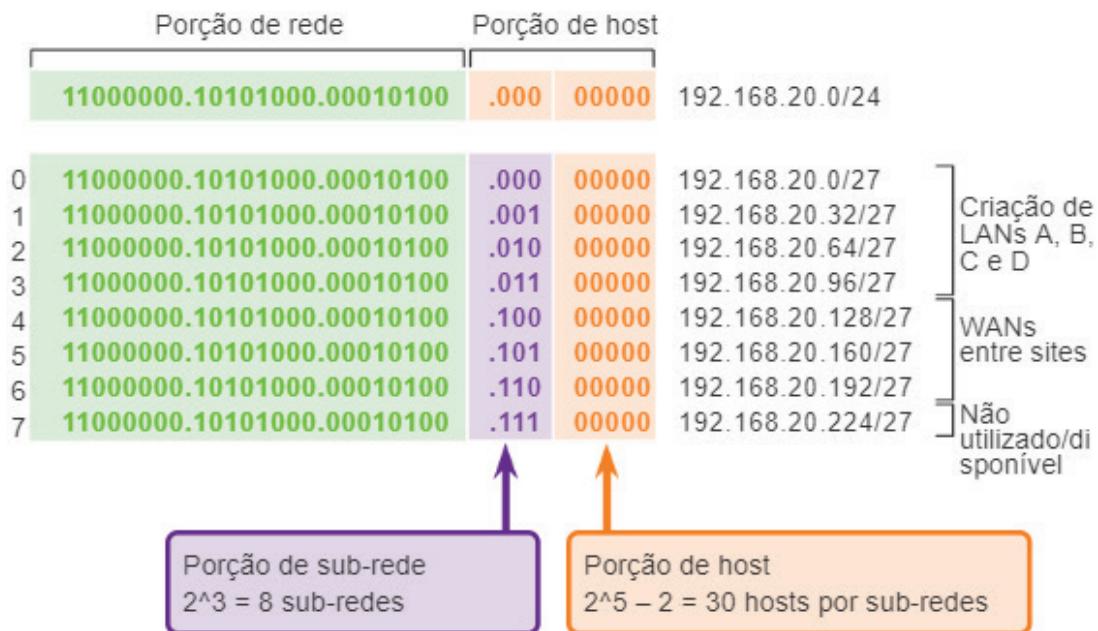


Figura 6 – Esquema clássico de endereçamento

A técnica de divisão de uma sub-rede clássica ou utilizar uma Máscara de Sub-Rede de Tamanho Variável ou apenas VLSM), foi desenvolvida para evitar desperdiçar endereços que as sub-redes tradicionais apresentavam.

Máscara de Rede de Tamanho Variável (VLSM)

Em todos os exemplos anteriores de divisão em sub-redes, observe que a mesma máscara de sub-rede foi aplicada em todas as sub-redes. Isso significa que cada sub-rede tem o mesmo número de endereços de host disponíveis. Por esse motivo a divisão em sub-redes cria sub-redes de mesmo tamanho e por esse motivo cada sub-rede criada utiliza uma mesma máscara de sub-rede. A técnica de VLSM pode permitir que um espaço de rede seja dividido em partes diferentes e menores ou maiores do que as criadas originalmente. Nesse caso a técnica de VLSM poderá ter a máscara de sub-rede com tamanhos variáveis dependendo da quantidade de bits que foram emprestados, afim de se ajustas a quantidade de endereços válidos para cada segmento criado, evitando um maior desperdício de endereços.

A forma como se divide sub-redes utilizando a técnica de VLSM é muito parecida com a original, porém é necessário a criação de vários passos para uma correta customização. Para que se possa aplicar o VLSM, primeiramente vamos segmentar as redes em sub-redes e estas sub-redes divididas novamente com tamanhos de máscaras variáveis, o que podem ser repetidos várias vezes e obtendo tamanhos de sub-redes diferentes entre si.

Como pudemos ver no exemplo anterior, a rede 192.168.20.0/24 foi dividida em oito sub-redes do mesmo tamanho; sete destas oito sub-redes foram atribuídas conforme seus segmentos físicos, sendo que quatro dessas sub-redes foram utilizadas para as redes locais e três sub-redes utilizadas para as conexões ponto-a-ponto de WAN entre os roteadores da topologia. É fácil notar que nas conexões ponto-a-ponto tivemos um grande desperdício de endereços, pois o que realmente precisamos seria dois endereços válidos do grupo de sub-redes calculadas para essa necessidade. Em função disso, podemos utilizar a técnica de VLSM afim de economizar tais endereços nas conexões de WAN.

Como exemplo, vamos utilizar a última sub-rede calculada 192.168.20.224/27 para a criação de redes menores que se adeque a configuração dos hosts ponto-a-ponto das conexões WAN. É importante lembrar que quando o número de endereços necessários para a identificação de dispositivos for conhecido, a fórmula $2^n - 2$ (onde n é igual ao número de bits de host restantes) pode ser aplicada. Para fornecermos dois endereços válidos, 2 bits emprestados do campo de host devem ser deixados na porção de host.

$$2^2 - 2 = 2$$

Como do cálculo original da sub-rede temos 5 bits dedicado ao host no espaço do endereço 192.168.20.224/27, 3 bits poderiam ser emprestados, sobrando 2 bits na porção de host. Conforme apresentado na próxima figura, esse esquema de divisão de endereços de sub-redes do VLSM reduz o número de endereços por sub-rede a um tamanho mais adequado para as conexões ponto-a-ponto, evitando desperdício. Dividir a sub-rede proposta 7 em sub-redes menores, permite que as sub-redes 4, 5 e 6 fiquem disponíveis para a criação de futuras redes.



Figura 7 – Endereçamento com VLSM
 Fonte: iStock/Getty Images

Como podemos verificar, definimos que as redes com prefixo /27 serão utilizadas para as redes locais (LANs), já as conexões ponto-a-ponto usarão as sub-redes segmentadas com a técnica de VLSM de prefixo /30, que podem acomodar dois endereços de IPv4 válidos, evitando desperdícios.

A Camada de Transporte

A camada de transporte tem como função estabelecer uma sessão de comunicação temporária entre dois hosts finais permitindo que os dados sejam transmitidos corretamente. Esta camada pode fornecer um método de distribuição de dados em toda a rede de uma maneira que possa assegurar que os dados possam ser devidamente colocados de uma forma ordenada para que o receptor possa receber os dados sem problemas, ela também proporciona a segmentação dos dados em pedaços menores e controla o fluxo desses segmentos para que possam ser reagrupados no destinatário. No TCP/IP, esses processos de segmentação dos dados e reagrupamento das informações no destinatário, podem ser obtidos através da utilização de dois protocolos da camada de transporte: TCP – *Transmission Control Protocol* e UDP – *User Datagram Protocol*.

As principais responsabilidades dos protocolos de camada de transporte são:

- Rastrear a comunicação de uma forma individual entre as aplicações nos dispositivos de origem e destino;
- Segmentar os dados para que esses segmentos possam ser gerenciados afim que sejam remontados nos dispositivos destino.
- Identificar a aplicação apropriada e controlar o fluxo da mesma o processo de comunicação.

Confiabilidade da Camada de Transporte

Como comentamos a camada de transporte também tem como atividade gerenciar os requisitos que a comunicação seja confiável, pois diferentes aplicações de rede possuem diferentes características e necessidades de confiabilidade para funcionarem corretamente.

A camada de rede que possui o protocolo IP está preocupada apenas com o endereçamento, estrutura de encapsulamento dos pacotes e com o processo de roteamento. O protocolo IP não especifica como os pacotes serão transportados ou entregues em um receptor.

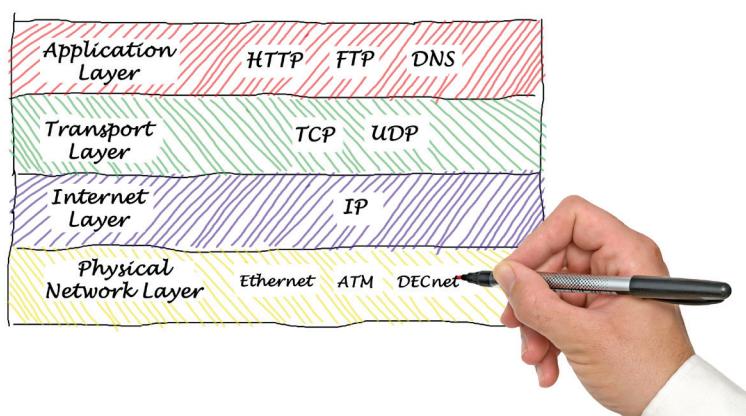


Figura 8 – Confiabilidade do transporte

Fonte: Wikimedia Commons

Nesse contexto, os protocolos de transporte que definem e especificam como as informações devem ser transmitidas entre dispositivos de redes. O IP então utiliza os protocolos de transporte TCP ou UDP para que os hosts possam ser ativados e que possam transferir os dados.

O protocolo de transporte TCP é considerado um protocolo de transporte confiável, muito completo e que pode garantir que todos os segmentos de dados possam chegar no destino. Ao contrário, o protocolo UDP tem um cabeçalho muito simples e que praticamente não fornece nenhuma confiabilidade da transmissão, porém por ser um protocolo enxuto tem maior flexibilidade e rapidez quando transportado na rede.

TCP – Transmission Control Protocol

No conjunto de protocolos TCP/IP, o protocolo de transporte TCP é a camada intermédia entre IP abaixo dela, e uma aplicação acima dela. Usando o TCP, as aplicações em hosts em rede podem estabelecer conexões confiáveis um para o outro. Este protocolo garante entrega em ordem de dados a partir do emissor para o receptor de forma confiável.

Operação Básica do Protocolo

O Transmission Control Protocol é orientado a conexão, ou seja, os dados do usuário não são trocados entre os pontos TCP até que uma conexão for estabelecida entre os dois pontos finais. Essa ligação é realizada durante toda a transmissão de dados entre os nós.

Coneções TCP tem três fases:

- Estabelecimento de conexão;
- Transferência de dados;
- Término de conexão.

Estabelecimento de Conexão

Para estabelecer uma conexão, o TCP usa um handshake de 3 vias (aperto de mão triplo). Antes de um cliente tentar se conectar com um servidor, o servidor deve ligar primeiro para uma porta afim de abrir uma conexão. Isso é chamado de abertura passiva. Uma vez que a abertura passiva é estabelecida, um cliente pode iniciar uma abertura ativa. O servidor, em seguida, envia uma confirmação para o cliente. Neste ponto, o cliente e o servidor devem ter recebido um reconhecimento da conexão.

Transferência de Dados

Algumas das principais características definidas TCP além de UDP:

- Transferência de dados sem erros;
- Transferência de dados ordenada;
- A retransmissão de pacotes perdidos;
- Descarte de pacotes duplicados;
- Estrangulamento congestionamento.

Erro de Transferência de Dados

Transferência de dados sem erros é garantida pelo TCP. Ele faz isso calculando uma soma de controle de 16 bits sobre o pacote de TCP (cabeçalho e de dados). Na extremidade receptora, se a soma de verificação não coincide com o conteúdo do pacote, ele é descartado. Quando o lado de envio não receber uma confirmação do pacote descartado, ele é retransmitido. Ou seja, não é o dispositivo de recepção que pede para que o dado seja retransmitido e sim o dispositivo que origem que não recebe uma confirmação do dado que ele transmitiu, nesse caso, ele retransmite o dado, pois acredita que esse dado não foi recebido pelo destinatário.

Transferência de Dados Ordenada

Fluxos de dados chamados segmentos são usados por pontos TCP para falar uns com os outros. Os segmentos podem ser muito grandes, de modo que o TCP necessite romper os segmentos em unidades menores de dados. Estas unidades são passadas e encapsuladas no pacote do protocolo IP. Para cada unidade de dados é atribuída um número de sequência, que se torna parte do segmento TCP. Na extremidade receptora, o módulo TCP utiliza os números de sequência no segmento para reconstituir os dados de utilizador na ordem correta.

Retransmissão de Pacotes Perdidos

Ao transmitir grandes quantidades de dados, não é incomum para algumas informações perderem dados ao longo do caminho. A fim de garantir a transferência segura de dados, o TCP requer um reconhecimento de cada pacote que envia. Esta confirmação é enviada pelo módulo TCP no host de recebimento. Se o reconhecimento não for recebido dentro de um período de tempo especificado (time to live ou tempo de vida), será retransmitido.

Descarte de Pacotes Duplicados

O cliente TCP retransmite pacotes que ele determina que sejam perdidos. O módulo TCP no lado de recebimento pode eventualmente receber pacotes que foram considerados perdidos após o envio do lado que retransmitiu os dados. Isto pode resultar na extremidade de recepção o recebimento de dois ou mais cópias

de um mesmo pacote. O módulo de fim TCP do receptor utiliza os números de sequência únicos no pacote para determinar se a duplicação de dados ocorreu e elimina quaisquer pacotes que são determinados duplicados.

Estrangulamento de Congestionamento

A propriedade final do TCP é o estrangulamento de congestionamento ou controle de fluxo. O objetivo é que o TCP para a ser capaz de enviar dados para o fim de recepção no ritmo mais rápido possível, sem sobrecarregá-lo.

Quando o primeiro TCP começa a transmissão de dados até o final, ele define um temporizador. O temporizador determina quanto tempo o remetente deve esperar por um pacote a ser reconhecido antes de retransmiti-lo. Se todos os pacotes são recebidos bem antes do tempo expirar, o TCP irá gradualmente aumentar a velocidade de transmissão, até que os pacotes comecem a se tornar não reconhecidos durante o período de tempo limite. Quando um número significativo de pacotes tem que ser retransmitidos, o TCP diminui a taxa na qual ele envia dados para a outra extremidade.

Terminação de Conexão

A fase de terminação de conexão utiliza, no mínimo, um aperto de mão de quatro vias, com cada um dos lados da ligação, que se encerra de forma independente. Quando um ponto final deseja parar a sua metade da conexão, ele transmite um pacote especial com um sinalizador que indica que ela seja concluída. A outra extremidade reconhece a bandeira. Uma terminação de conexão típica inclui este aperto de mão de duas fases a partir de ambas as extremidades da ligação.

Sockets TCP

O TCP - Transmission Control Protocol é orientado a conexão. Uma ligação virtual é criada pela primeira vez, em seguida, mantida através da duração da transferência de dados. Os pontos finais da conexão entre os pontos TCP são chamados soquetes (sockets). Um encaixe é identificado por uma combinação do endereço de origem e de acolhimento de porta em conjunto com o endereço de destino de acolhimento e de porta. A chegada de pacotes de dados TCP são identificados como pertencentes a uma ligação TCP específica pelo seu encaixe. Do ponto de vista lógico, pontos TCP podem se comunicar diretamente uns com os outros através da conexão socket. Na realidade, leitura e escrita de pacotes para um soquete é como as interfaces TCP com a camada IP abaixo dela.

É muito comum na literatura se confundirem com portas e soquetes, por acreditarem ser a mesma coisa. Na verdade, a porta TCP é um número de identificação da aplicação a ser transportada no segmento, já o soquete (socket) é uma combinação da porta mais o endereço de pacote.

Estrutura do Cabeçalho do TCP

O TCP assume uma sobrecarga adicional para ganhar essas funções. Como mostrado na figura, cada segmento TCP tem 20 bytes de sobrecarga no cabeçalho que encapsula os dados da camada do aplicativo. Isso é consideravelmente mais que um segmento UDP, que tem apenas 8 bytes de sobrecarga. A sobrecarga adicional inclui:

- **Porta de Origem (16 bits)** – Porta que identifica a aplicação que origina a transmissão;
- **Porta de Destino (16 bits)** – Porta que identifica a aplicação que deve receber a transmissão.
- **Número de sequência (32 bits)** – Usado somente para a remontagem dos dados.
- **Número de confirmação (32 bits)** – Indica os dados que foram recebidos.
- **Comprimento do cabeçalho (4 bits)** – Conhecidos como «deslocamento de dados. Mostra o comprimento do cabeçalho de segmento TCP.
- **Reservado (6 bits)** – Esse campo é reservado para o futuro.
- **Bits de controle (6 bits)** – Inclui códigos de bit, ou flags, que indicam a finalidade e a função do segmento TCP.
- **Tamanho da janela (16 bits)** – Indica o número de bytes que podem ser aceitos ao mesmo tempo.
- **Checksum (16 bits)** – Usado para a verificação de erros do cabeçalho e de dados do segmento.
- **Urgente (16 bits)** – Indica se os dados são urgentes.

Exemplos de aplicativos que usam o TCP são navegadores Web, e-mail e transferências de arquivos.

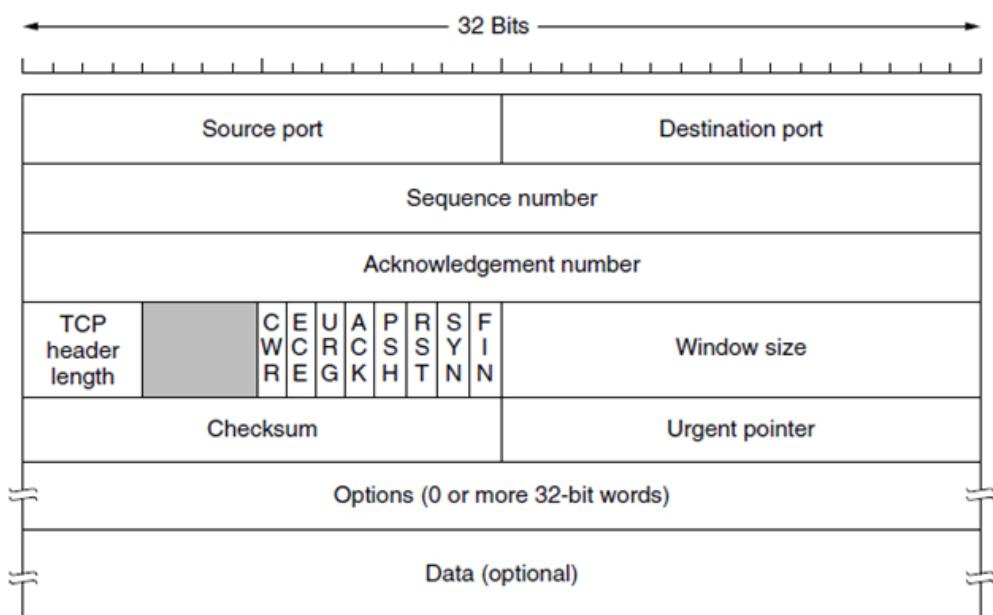


Figura 9 – Cabeçalho do Segmento TCP

Fonte: Wikimedia Commons

Os aplicativos que usam TCP

O seguinte é uma lista de aplicativos comuns que usam diretamente os serviços de transferência de dados confiáveis fornecidos pelo TCP:

- **File Transfer Protocol (FTP):** Fornece um mecanismo para mover arquivos de dados entre sistemas finais. Os programas cliente e servidor FTP, assim como a maioria dos navegadores da Web, contêm uma implementação do protocolo FTP.
- **HyperText Transfer Protocol (HTTP):** Protocolo usado para mover páginas da Web através de uma conexão de internet. O protocolo HTTP é construído em navegadores da Web e servidores Web.
- **Interactive Mail Access Protocol (IMAP):** Fornece aos clientes acesso a mensagens de e-mail e caixas de correio através de uma rede. Ele é incorporado em aplicações de e-mail
- **Post Office Protocol (POP):** Permite clientes lerem e removerem e-mail residente num servidor remoto. Ele é incorporado em aplicações de e-mail.
- **Remote Login (Rlogin):** Fornece a rede capacidade de login remoto.
- **Simple Mail Transfer Protocol (SMTP):** Usado para entregar e-mail de um sistema para outro. Ele é incorporado em aplicações de e-mail.
- **Secure Shell (SSH):** Fornece acesso remoto a computadores, proporcionando algoritmos de criptografia dos dados
- **Telnet:** Fornece terminal de rede ou capacidade de login remoto sem algoritmos de segurança.

UDP – User Datagram Protocol

O UDP - User Datagram Protocol descrito no RFC 768 é muito simples e seu PDU é chamado datagrama. Datagramas são considerados não confiáveis, pois não garantem que o dado será reconstruído se não forem recebidos em ordem correta. Se for necessário termos confiabilidade nas informações transmitidas, o protocolo de transporte UDP e aplicações que o utilizam por natureza não deve ser utilizado.

Enquanto UDP não é confiável, a falta de verificação e correção de erros fazem com que este protocolo seja muito rápido e eficiente para diversas aplicações de dados intensivos ou menos sensíveis a tempo, como por exemplo o Domain Name Service (DNS), o Simple Network Management Protocol (SNMP), o Dynamic Host Configuration Protocol (DHCP) e o Routing Information Protocol (RIP) este último considerado um protocolo de roteamento. A utilização do protocolo de transporte UDP também é adequada para streaming de vídeo.

Alguns recursos que descrevem o datagrama UDP:

- **Sem Orientação a Conexão** – O UDP não possui um mecanismo de estabelecimento de conexão entre os dispositivos na comunicação, antes que os dados sejam transmitidos.
- **Entrega Não Confiável** – O UDP não fornece mecanismos de confiabilidade para que os dados possam ser enviados de uma forma segura, além disso não possui processos de retransmissão de dados perdidos ou corrompidos.
- **Sem Reconstrução Ordenada** – Os dados são transmitidos em sequência e devem ser recebidos nessa mesma ordem, pois o UDP não possui técnicas de remontagem e reagrupamento dos datagramas enviados.
- **Sem Controle de Fluxo** – O UDP não possui um mecanismo de controle e gerenciamento da qualidade na transmissão dos dados. Se a origem transmitir os dados e os recursos de rede ficarem sobrecarregados, o host destino provavelmente irá descartar os dados até que os recursos possam se tornar disponíveis novamente, além disso ele não possui um mecanismo reenvio de dados descartados ou corrompidos.

Estrutura do Cabeçalho do UDP

Embora o UDP não inclua os mecanismos de confiabilidade e de controle de fluxo do TCP, a entrega de dados de baixa carga do UDP faz com que ele se torne um protocolo de transporte muito bom para aplicações que podem tolerar alguma perda de dados. Os segmentos dos dados do UDP são chamados de datagramas. Esses datagramas por sua vez são enviados como o “melhor esforço” pelo protocolo de camada de transporte.

Os campos do cabeçalho UDP são:

- **Porta de Origem (16 bits)** – Porta que identifica a aplicação que origina a transmissão.
- **Porta de Destino (16 bits)** – Porta que identifica a aplicação que deve receber a transmissão.
- **Comprimento do cabeçalho (16 bits)** – Conhecidos como «deslocamento de dados». Mostra o comprimento do cabeçalho de segmento TCP.
- **Checksum (16 bits)** – Usado para a verificação de erros do cabeçalho e de dados do segmento.

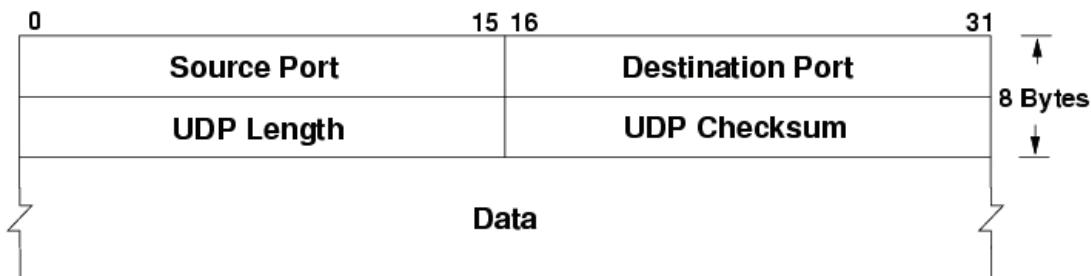


Figura 10 – Cabeçalho do Datagrama UDP

Os aplicativos que usam UDP

O seguinte é uma lista de aplicativos comuns que usam diretamente os serviços de transferência de dados não confiáveis fornecidos pelo UDP.

- **Domain Name Service (DNS):** Fornece um mecanismo de tradução de Domínios de rede para endereços IP correspondentes ou vice-e-versa.
- **Simple Network Management Protocol (SNMP):** É um protocolo utilizado para o gerenciamento de dispositivos de rede.
- **Dynamic Host Configuration Protocol (DHCP):** Fornece serviços e mapeamento de endereços para dispositivos de redes solicitantes.
- **Trivial File Transfer Protocol (TFTP):** Protocolo de transferência de arquivos trivial, muito usado em transmissão de arquivos pequenos, como por exemplo o arquivo de configuração de um roteador.
- **Routing Information Protocol (RIP):** Protocolo de roteamento do tipo vetor distância muito usado para rede pequenas.

Endereçamento TCP e UDP (Portas)

Como vimos anteriormente, nos cabeçalhos de cada segmento TCP ou datagrama UDP, há uma porta origem e destino. O número da porta origem é o número que associa a aplicação original do dispositivo que irá participar da transmissão e a porta destino é o número que identificará qual a aplicação ou serviço que o dispositivo de origem necessita utilizar na transmissão. Uma porta então é um identificador numérico definido e marcado dentro de cada segmento e que é utilizado para rastrear conversas específicas e serviços destino solicitados:

- **Porta Origem** – O número da porta origem é gerada aleatoriamente pelo host de envio para identificar uma conversação entre dois dispositivos que fazem parte da transmissão. Isso permite que várias conversações possam ocorrer simultaneamente. Em outras palavras, um dispositivo pode enviar várias solicitações de serviço HTTP para um servidor Web ao mesmo tempo. As conversas separadas são rastreadas com base em portas origem. (CISCO NETACAD, 2017)
- **Porta Destino** – O host de origem preenche um número de porta destino no segmento para informar qual o servidor destino que será solicitado. Por exemplo, a porta 80 se refere ao HTTP ou ao serviço Web. Quando um cliente especifica a porta 80 na porta destino, o servidor que receber a mensagem sabe que os serviços Web são solicitados. (CISCO NETACAD, 2017)

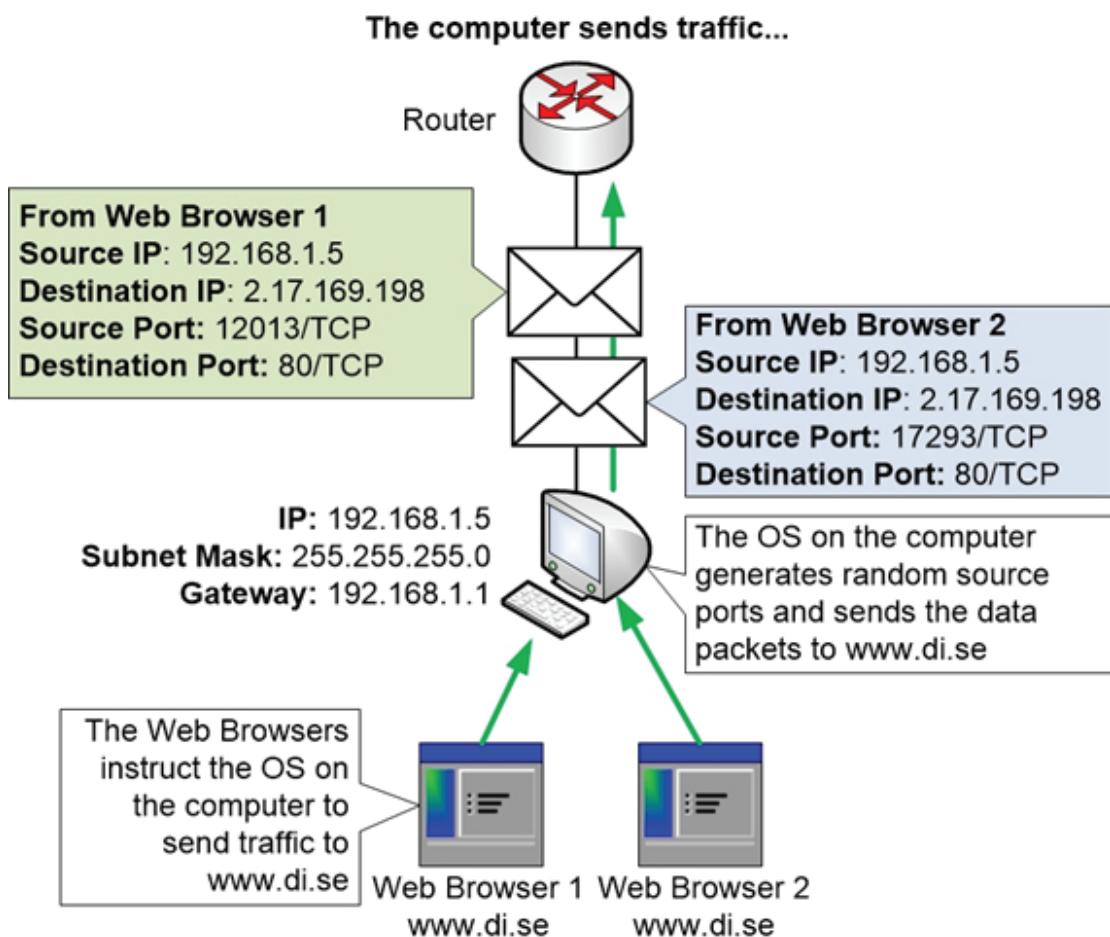


Figura 11 – Portas de conexão TCP/UDP

Padronização de Número de Portas

O IANA – Internet Assigned Numbers Authority é um dos órgãos mais importantes da Internet e define a designação de endereços IPs e identificação de número de portas TCP e UDP, como podemos verificar:

Existem diferentes tipos de números de portas:

- **Portas Muito Conhecidas (Números de 0 a 1023)** – Esses números estão reservados para serviços e aplicativos conhecidos. Elas são comumente utilizadas em aplicativos como o HTTP, IMAP, SMTP, telnet e muitas outras.
- **Portas Registradas (Números de 1024 a 49151)** – Estes números de portas são designados para processos ou aplicativos de clientes. Quando não usadas para um recurso de servidor, estas portas podem ser dinamicamente selecionadas por um usuário como sua porta origem.
- **Portas Dinâmicas ou Privadas (Números de 49152 a 65535)** – Esses números de porta também são conhecidos como portas efêmeras, elas são geralmente designadas de forma dinâmica aos aplicativos quando o cliente inicia uma conexão a um serviço de rede. A porta dinâmica é mais frequentemente utilizada para identificar o aplicativo do cliente durante a comunicação, enquanto o cliente utiliza a porta conhecida para identificar e estabelecer conexão com o serviço solicitado no servidor. Não é muito comum um cliente se conectar a um serviço usando uma porta dinâmica ou privada (embora alguns programas de compartilhamento de arquivos peer-to-peer o façam). (CISCO NETACAD, 2017)

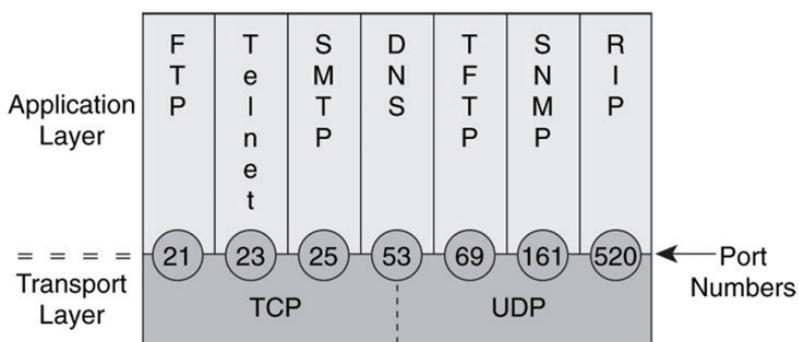


Figura 12 – Algumas portas TCP/UDP com suas aplicações correspondentes

Material Complementar

Indicações para saber mais sobre os assuntos abordados nesta Unidade:



Sites

Curso WEB: CISCO NETACAD

Curso WEB: CISCO NETACAD – Módulo de Introdução a Redes – Capítulo 7: Camada de Transporte.

<https://goo.gl/kSQz1K>

Curso WEB: CISCO NETACAD

Curso WEB: CISCO NETACAD – Módulo de Introdução a Redes – Capítulo 9: Divisão de Redes IP em Sub-Redes.

<https://goo.gl/kSQz1K>



Livros

Redes de Computadores

TANENBAUM, A. S. **Redes de Computadores**. 5^a Ed., Pearson, 2011.

Redes de computadores e internet

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

Referências

CISCO NETACAD – **Módulo de Introdução a Redes (CCNA1)** – 6^a Versão, Cisco Systems, 2017. (material on-line). Disponível em: <<https://www.netacad.com/pt-br>>

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

TANENBAUM, A. S. – **Redes de Computadores** – 5^a Ed., Pearson, 2011.



Cruzeiro do Sul
Educacional