Gerenciamento de Serviços de Redes de Computadores e Virtualização

Prof. Me. Wallace Rodrigues de Santana

www.neutronica.com.br

Atribuição-NãoComercial-CompartilhaIgual 3.0 Brasil (CC BY-NC-SA 3.0)

Você tem a liberdade de:

Compartilhar — copiar, distribuir e transmitir a obra.

Remixar — criar obras derivadas.

Sob as seguintes condições:

Atribuição — Você deve creditar a obra da forma especificada pelo autor ou licenciante (mas não de maneira que sugira que estes concedem qualquer aval a você ou ao seu uso da obra).

Uso não comercial — Você não pode usar esta obra para fins comerciais.

Compartilhamento pela mesma licença — Se você alterar, transformar ou criar em cima desta obra, você poderá distribuir a obra resultante apenas sob a mesma licença, ou sob uma licença similar à presente.

Ficando claro que:

Renúncia — Qualquer das condições acima pode ser **renunciada** se você obtiver permissão do titular dos direitos autorais.

Domínio Público — Onde a obra ou qualquer de seus elementos estiver em **domínio público** sob o direito aplicável, esta condição não é, de maneira alguma, afetada pela licença.

Outros Direitos — Os seguintes direitos não são, de maneira alguma, afetados pela licença:

- Limitações e exceções aos direitos autorais ou quaisquer **usos livres** aplicáveis;
- Os **direitos morais** do autor;
- Direitos que outras pessoas podem ter sobre a obra ou sobre a utilização da obra, tais como **direitos de imagem** ou privacidade.

Aviso — Para qualquer reutilização ou distribuição, você deve deixar claro a terceiros os termos da licença a que se encontra submetida esta obra. A melhor maneira de fazer isso é com um link para esta página.

Módulo Zero

Apresentação da Disciplina

Objetivo geral

- Capacitar os alunos na compreensão dos principais serviços de redes e apresentar os conceitos básicos sobre virtualização de servidores.

Objetivos específicos

- Identificar e compreender a funcionalidade dos elementos lógicos e físicos de redes de computadores;
- Instalar e configurar serviços nos principais sistemas operacionais;
- Compreender e implementar o endereçamento de redes;
- Manipular e configurar serviços ligados as redes de computadores e Internet, tais como DNS, Web, E-mail, FTP, AntiSPAM, Proxy e SMB;
- Identificar as ferramentas de virtualização mais adequadas para instalação de servidores.

Módulos

PARTE I

1. Modelos de Referência OSI e TCP/IP
2. Camada de Rede e Protocolo IP
3. Dynamic Host Configuration Protocol
4. Domain Name System
5. File Transfer Protocol
6. Hypertext Transfer Protocol
7. Correio Eletrônico
8. Antispam
9. Server Message Block
10. Network File System

Módulos

PARTE II

11. Cluster e Grid Computing
12. Consolidação e Virtualização de Servidores
13. Computação em Nuvem
14. Serviços de Computação em Nuvem

Ementa

- Histórico e evolução da Internet;
- Serviços e conceitos ligados a Internet;
- Modelo de Referência OSI e TCP/IP;
- IPv4 e IPv6;
- DHCP e DNS;
- Serviços Web;
- Correio Eletrônico;
- FTP;
- AntiSPAM e Proxy;
- Samba;
- Cluster e Grid Computing;
- Virtualização e Computação em Nuvem.

Referências

BÁSICAS

KUROSE, J. F.; ROSS, K. W. **Redes de Computadores e a Internet: uma abordagem Top-Down.** 6ª ed. São Paulo: Pearson, 2013. E-book.

TANENBAUM, Andrew S.; WETHERALL David. **Redes de Computadores.** 5ª ed. São Paulo: Pearson Brasil, 2011. E-book.

VERAS, Manoel.; **Computação em Nuvem.** São Paulo: Editora Brasport, 2015.

COMPLEMENTARES

LEMOS, Ronaldo; FELICE, Massimo di. **A vida em rede.** São Paulo: Pearson Brasil, 2015. E-book

MORIMOTO, C.E. **Redes - Guia Prático.** Porto Alegre: Sul Editores, 2010.

PINOCHET, Luis Hernan Contreras. **Tecnologia da Informação e Comunicação.** Editora Campus, 2014.

THOMPSON, M. A. **Windows Server 2012 - Instalação, Configuração e Administração de Redes.** São Paulo: Editora Érica, 2012.

TIBET, C.V. **Linux - Administração e Suporte.** 1ª ed. São Paulo: Novatec, 2001.

Módulo 1

Modelos de Referência OSI e TCP/IP

Antecedentes

No início as redes eram proprietárias e a implementação de um fabricante era incompatível com a implementação de outro fabricante. Exemplos desta época são as redes SNA (Systems Network Architecture) da IBM, XNS (Xerox Network Services) da Xerox e DECnet da Digital.

Modelos de referência

No inicio da década de 1980 a *International Organization for Standardization* (ISO) criou um modelo de referência para conexão de redes denominado *Open Systems Interconnection* (norma ISO 7498:1984), que ficou conhecido como modelo ISO/OSI ou simplesmente modelo OSI.

O modelo OSI aproveitou as boas práticas presentes nas implementações SNA e XNS.

No início da década de 1990, a *International Electrotechnical Commission* (IEC) juntou-se à ISO para reescrever a norma, que em 1994 foi publicada como norma ISO/IEC 7498-1 Segunda Edição.

Implementações pós OSI

As primeiras implementações pós OSI baseavam-se nas redes XNS na Xerox. Entre elas destacam-se as redes NetWare da Novell, VINES (*Virtual Integrated NEtwork Service*) da Banyan e AppleTalk da Apple.

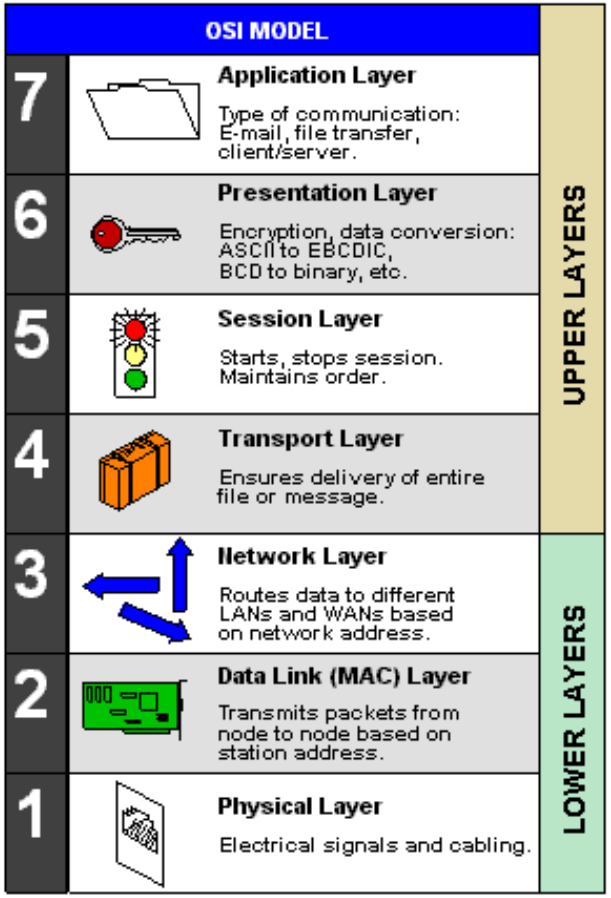
Modelo de referência OSI

O modelo de referência OSI é composto por sete camadas e representa um modelo base para a implementação da pilha de protocolos da rede, sem no entanto especificar exatamente os serviços e protocolos de cada camada.

A transmissão de dados entre uma origem e um destino deve seguir uma sequência lógica de operações, desde a captura dos dados, passando por sua transformação até a transmissão dos mesmos.

A ideia básica por trás do modelo OSI é que cada camada deve implementar apenas as operações e serviços necessários para abstrair cada etapa da transmissão de dados.

Cada camada deve se comunicar apenas com as camadas adjacentes, ou seja, uma camada sempre recebe dados da camada anterior e depois repassa para a camada posterior.

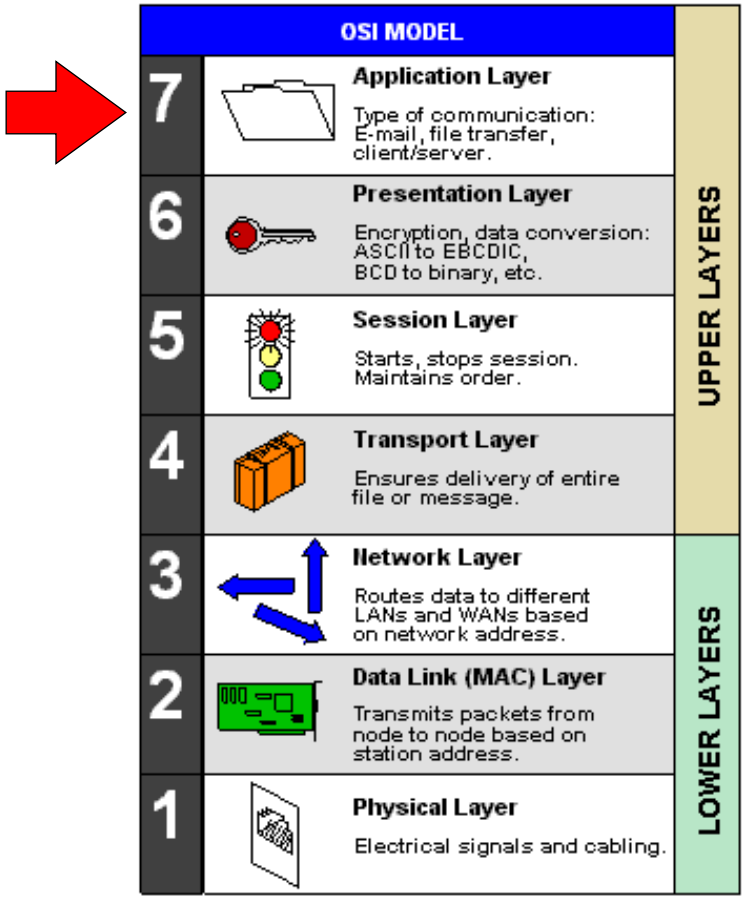


Fonte: Computer Desktop Encyclopedia

Camada de aplicação

É nesta camada que residem as aplicações, tais como o navegador de Internet, cliente de correio eletrônico, transferência de arquivos, entre outros.

Esta camada funciona como uma interface entre as aplicações do usuário e a pilha de protocolos das camadas mais baixas.

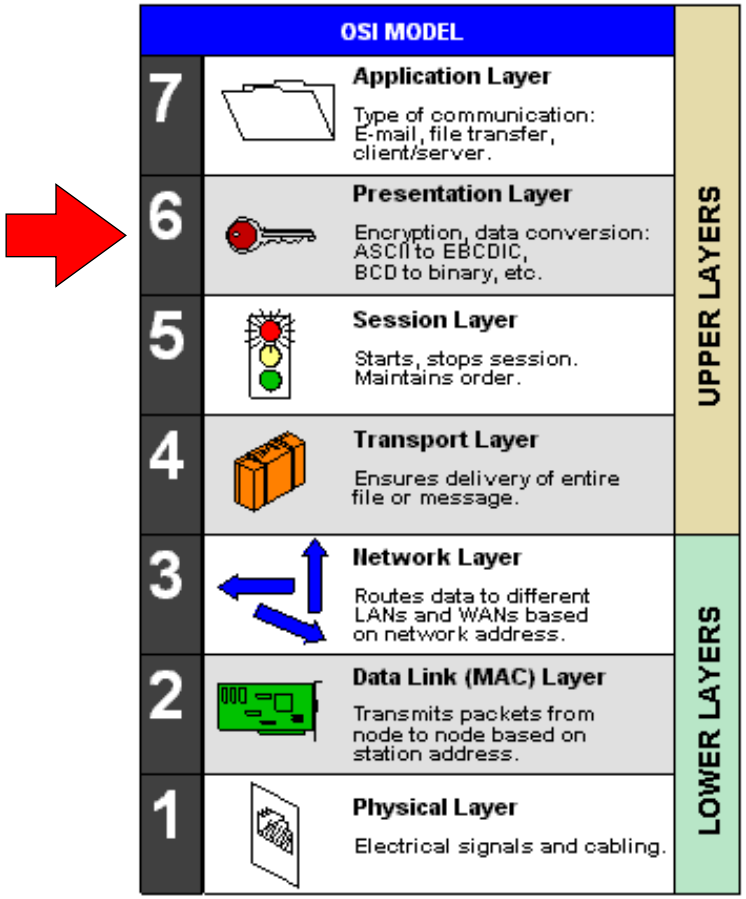


Fonte: Computer Desktop Encyclopedia

Camada de apresentação

Esta camada é responsável por converter os dados em um formato universal que possa ser interpretado por sistemas de plataformas diferentes.

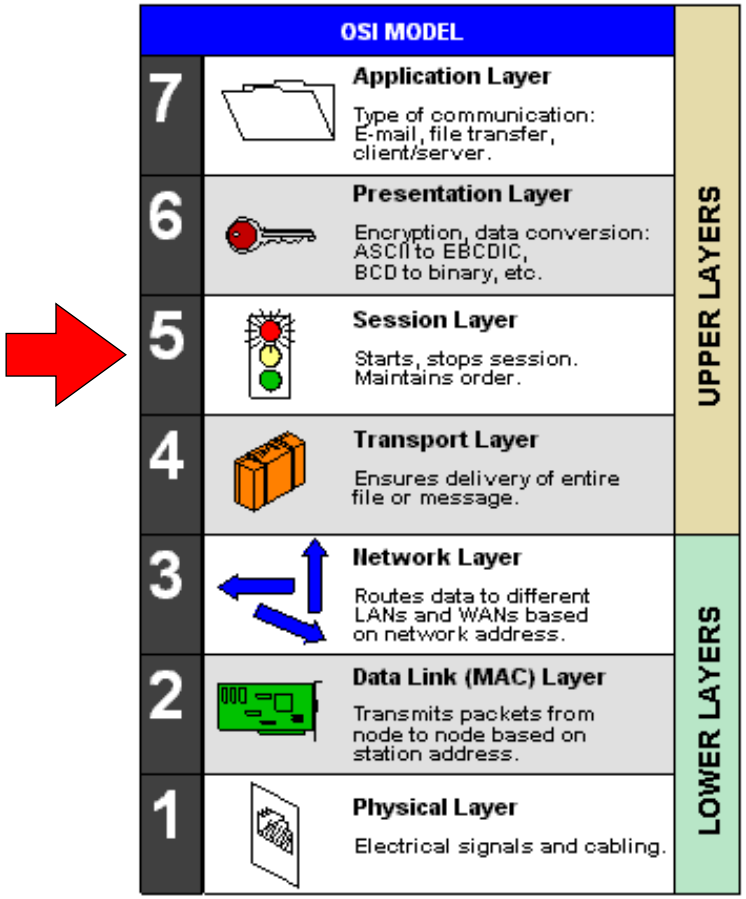
É nesta camada que as operações de criptografia e compactação de dados são executadas.



Fonte: Computer Desktop Encyclopedia

Camada de sessão

A camada de sessão controla o estabelecimento da comunicação entre um par origem e destino. É responsável por iniciar e encerrar as sessões de comunicação.



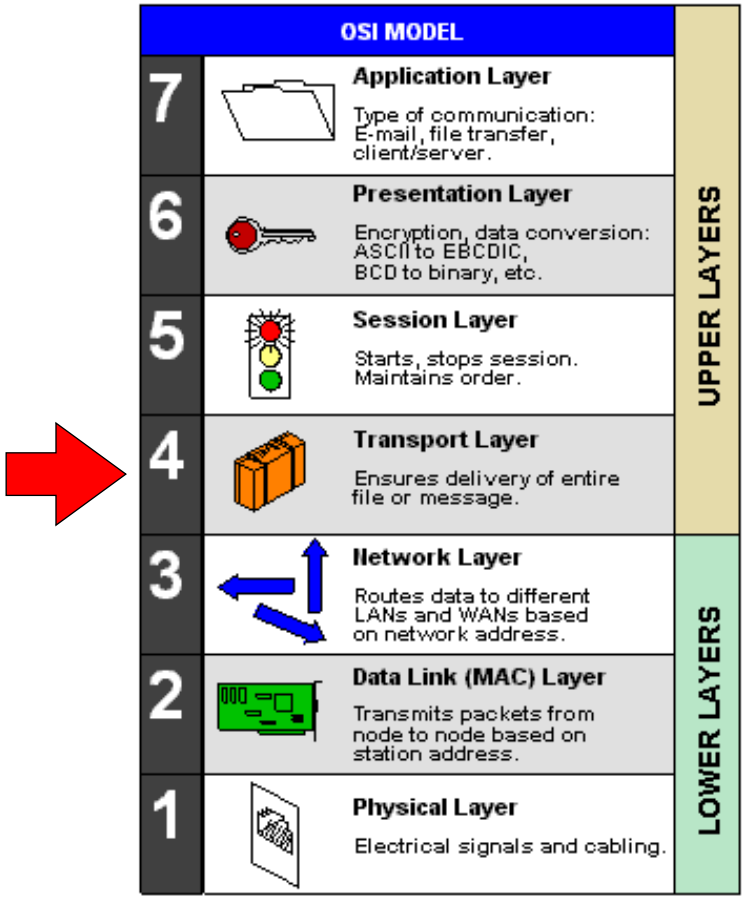
Fonte: Computer Desktop Encyclopedia

Camada de transporte

Esta camada é responsável por segmentar os dados provenientes das camadas superiores e entregá-las da melhor maneira possível ao destinatário.

Uma vez que os dados podem ser segmentados, a camada de transporte numera sequencialmente cada segmento, e estes deverão ser novamente juntados no destino.

A entrega pode ser do tipo confiável (com confirmação de entrega) ou do tipo não confiável (sem confirmação).

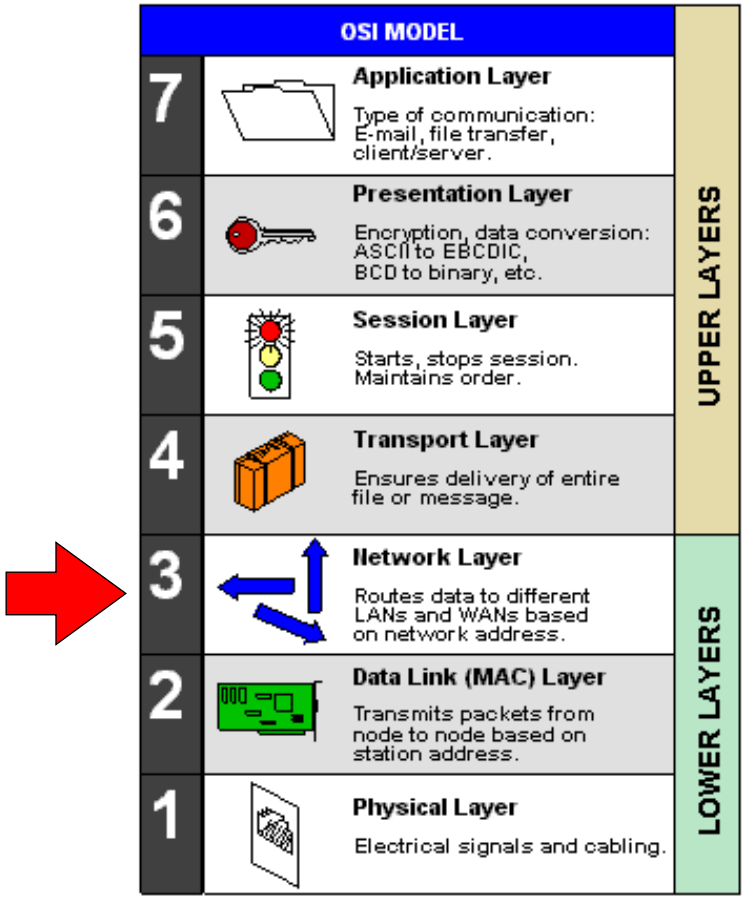


Fonte: Computer Desktop Encyclopedia

Camada de rede

A camada de rede é responsável por fazer a entrega dos dados em redes distintas.

Os protocolos da camada de rede usam o endereço de rede para identificar qual o melhor caminho para entregar dados entre a origem e o destino.

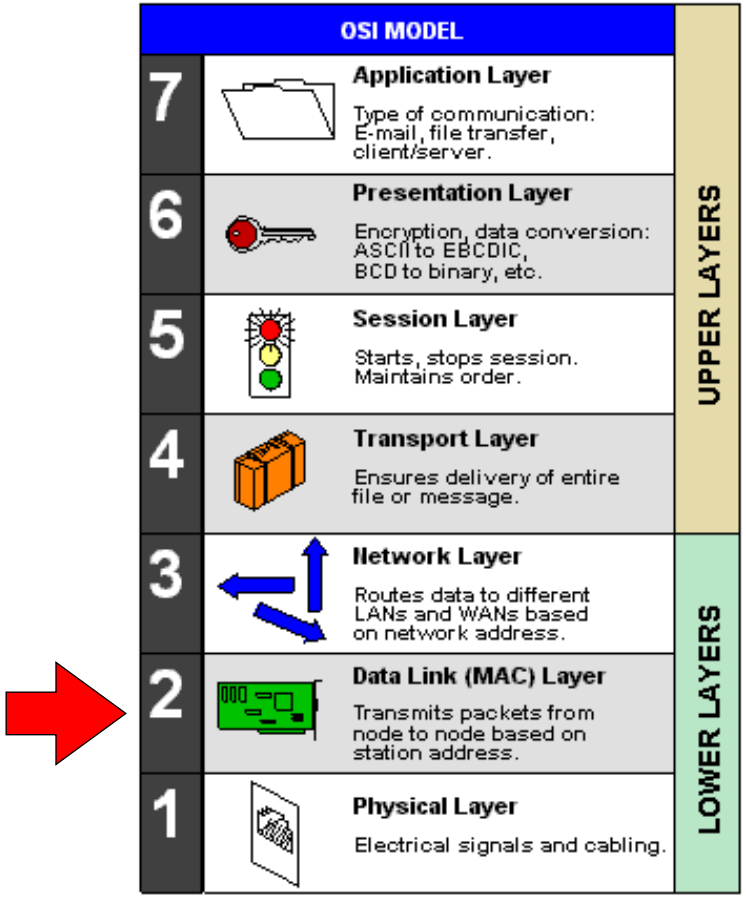


Fonte: Computer Desktop Encyclopedia

Camada de enlace

A camada de enlace é responsável por fazer a entrega de dados em redes locais, ou ainda, entre máquinas que estejam no mesmo segmento de rede.

Os protocolos da camada de enlace usam apenas o endereço local de cada estação, sem levar em conta o endereço de rede.

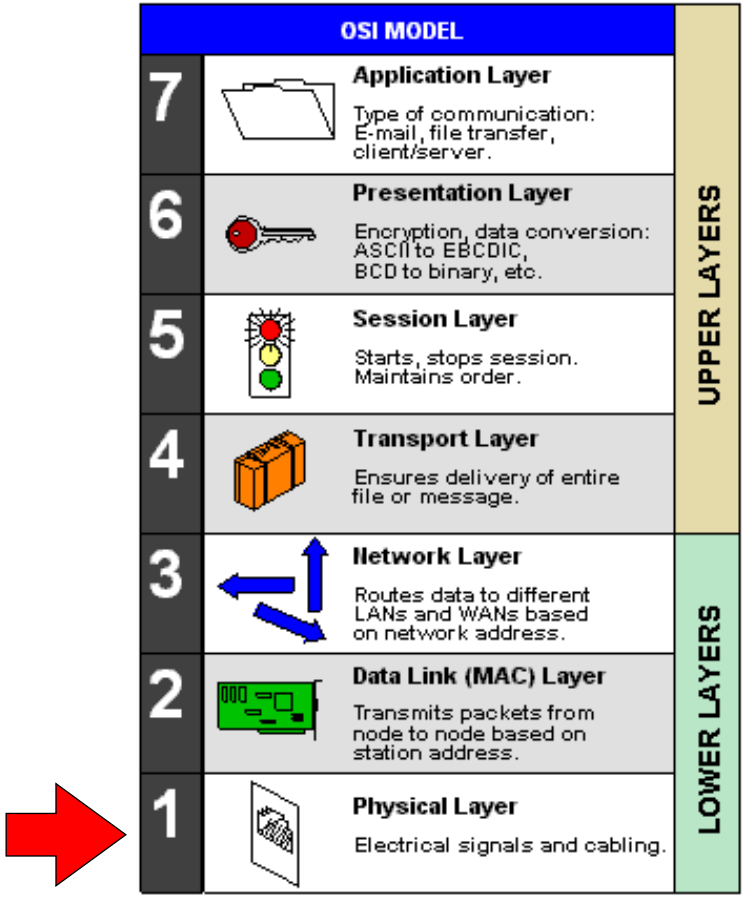


Fonte: Computer Desktop Encyclopedia

Camada física

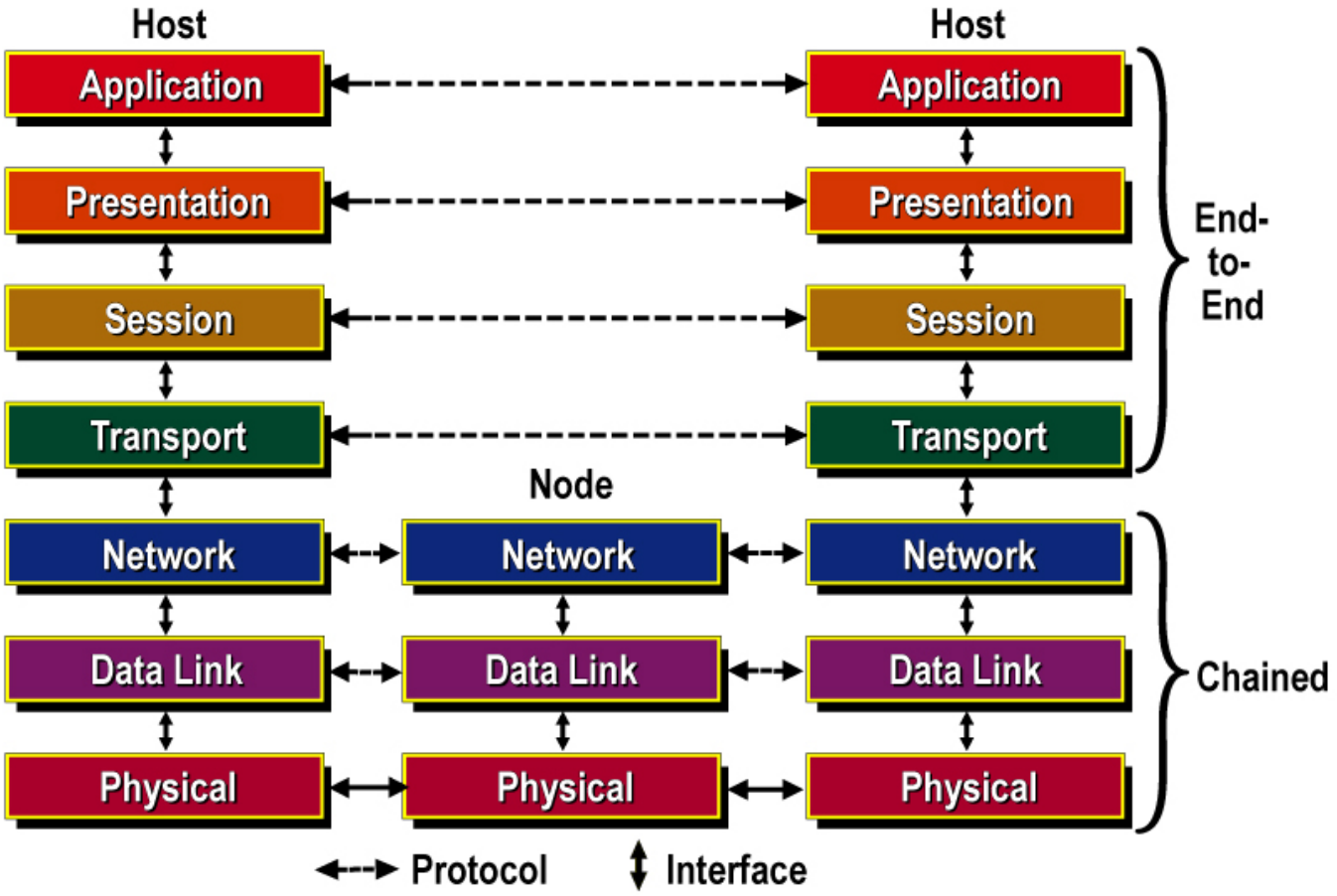
A camada física define as especificações elétricas, físicas e mecânicas dos meios físicos de transmissão.

Esta camada é responsável por enviar uma sequencia de bits entre a origem e o destino.

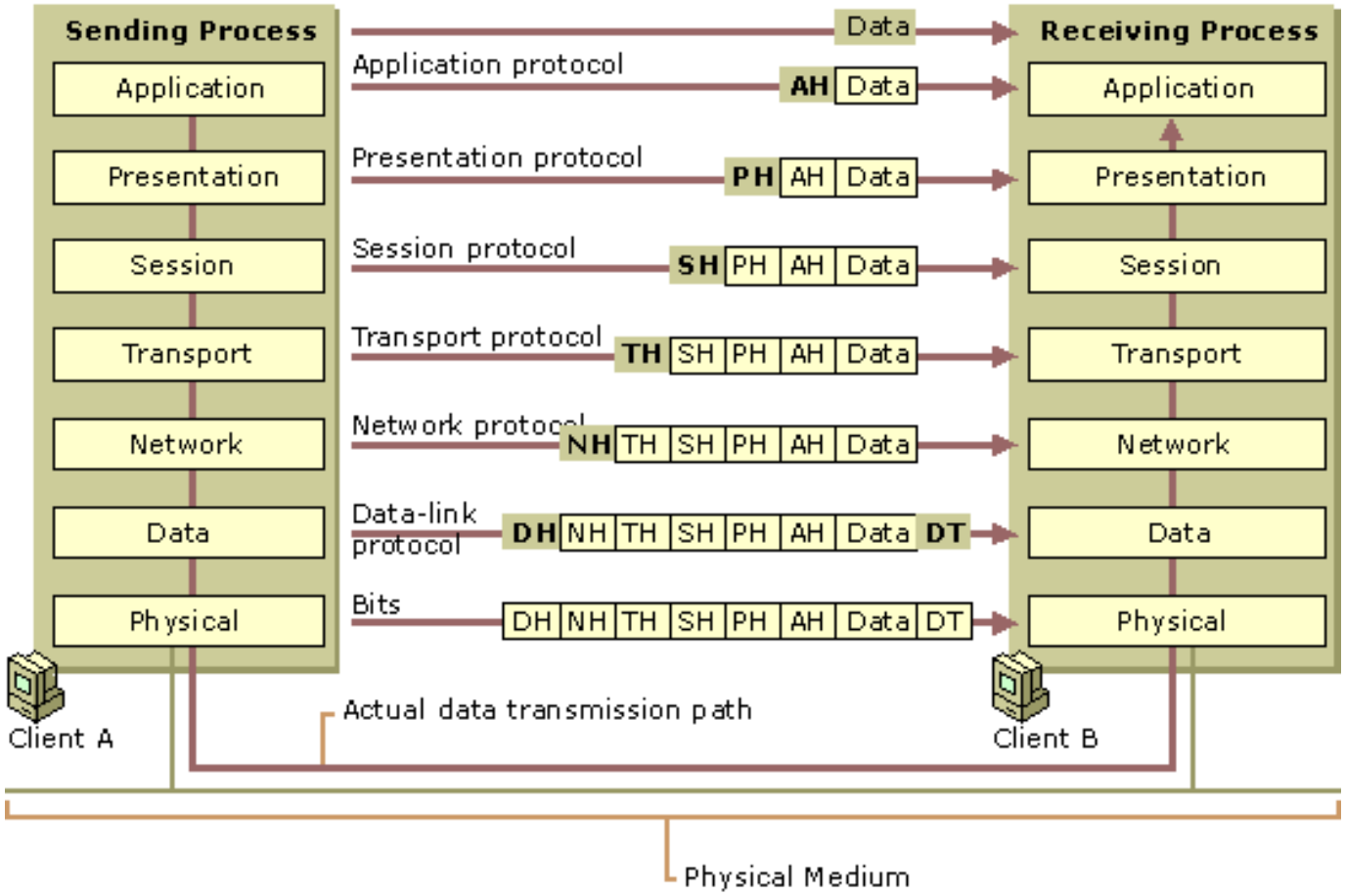


Fonte: Computer Desktop Encyclopedia

Relação entre as camadas



Fluxo de dados no modelo OSI



Modelo de referência TCP/IP

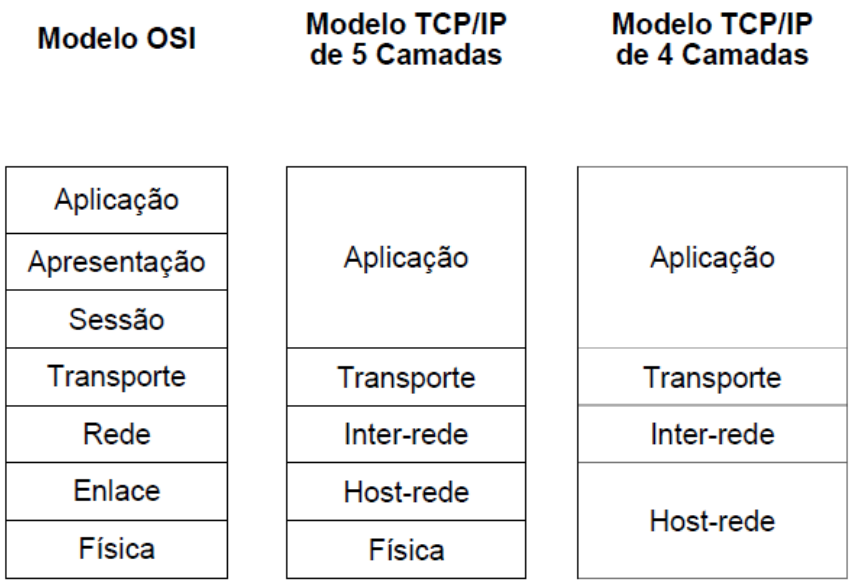
O modelo de referência TCP/IP surgiu de um projeto do exército dos Estados Unidos com o objetivo de criar uma rede que fosse tolerante à falhas.

Houve a participação intensa de universidades e órgãos de pesquisa, e com o fim da Guerra Fria, a rede começou a aceitar que outras organizações pudessem se conectar à rede.

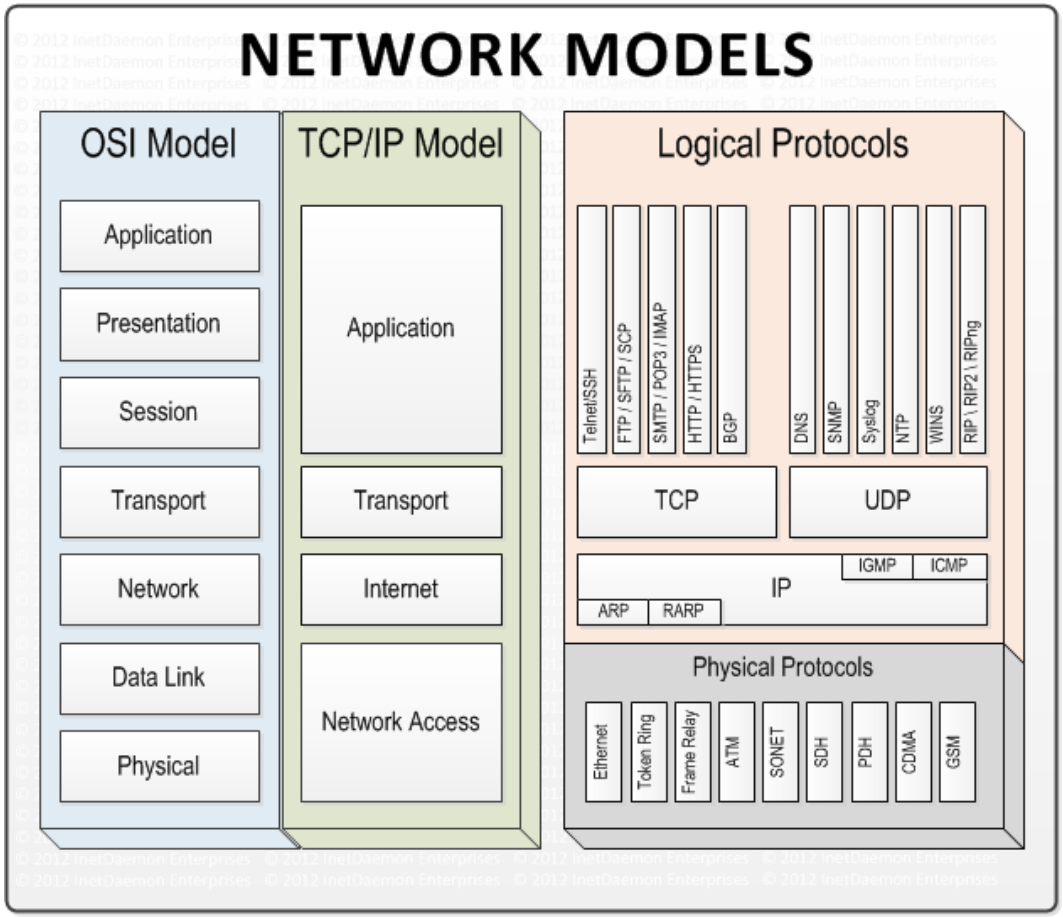
O Modelo de Referência não seguiu a mesma padronização do Modelo OSI, e por isso alguns autores adotam um modelo de 4 camadas, enquanto outros adotam o modelo de 5 camadas.

O modelo TCP/IP não é baseado no modelo OSI. A sua comparação destina-se apenas a facilitar o entendimento do modelo.

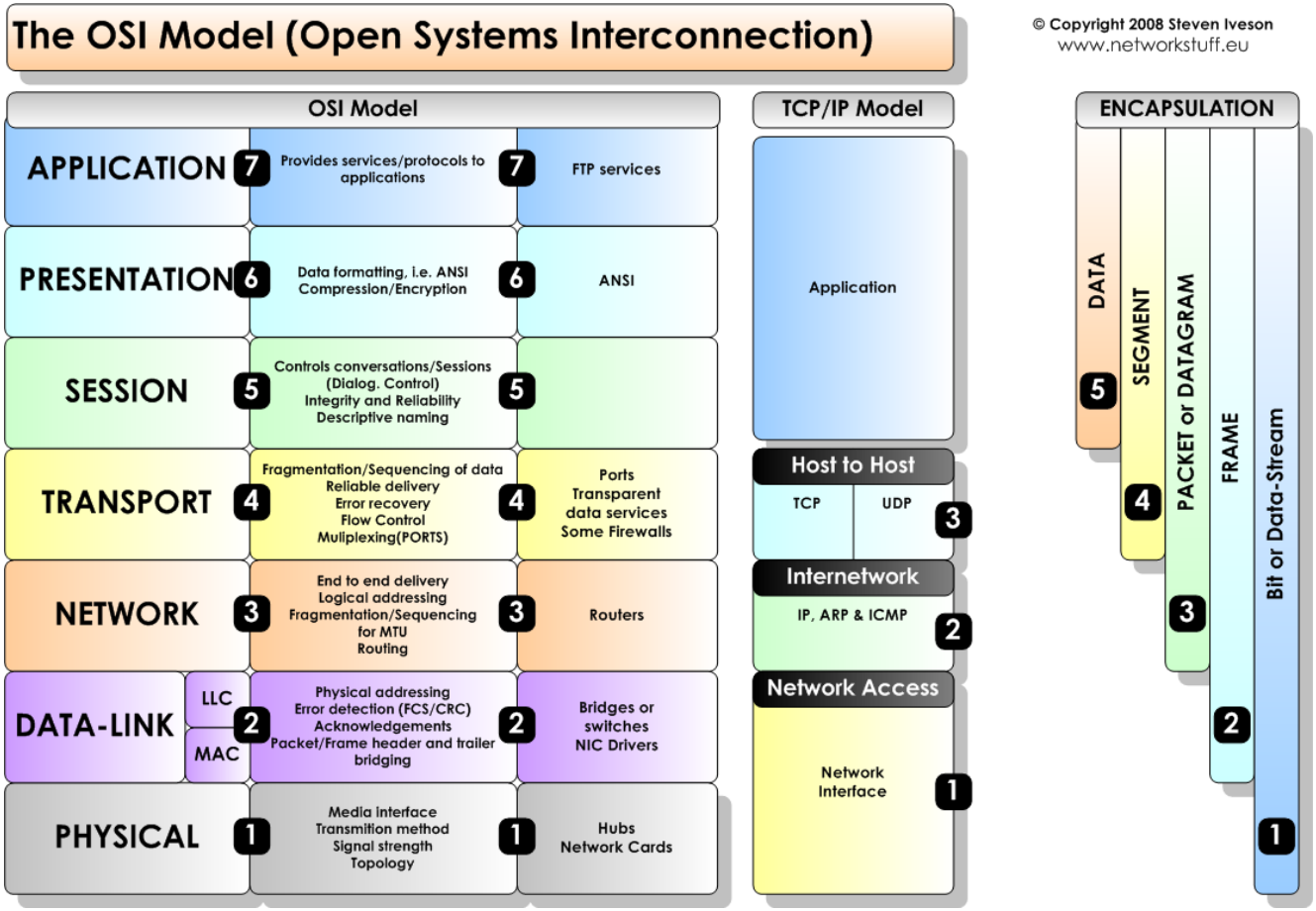
O Modelo de Referência TCP/IP recebe este nome porque seus dois principais protocolos são o de transporte (TCP) e o de rede (IP).



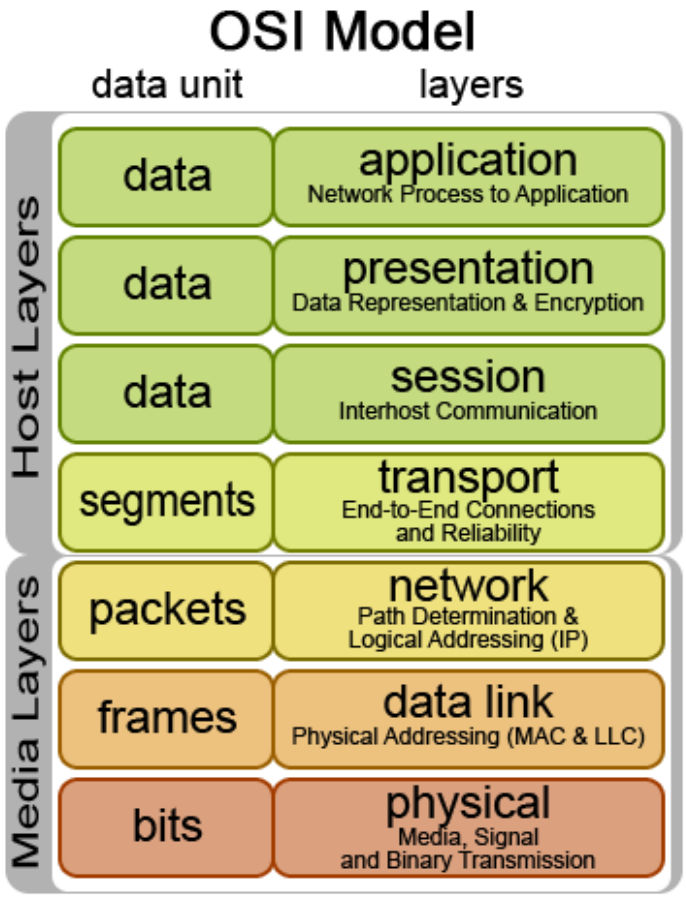
Relação entre os modelos OSI e TCP/IP



Relação entre os modelos OSI e TCP/IP

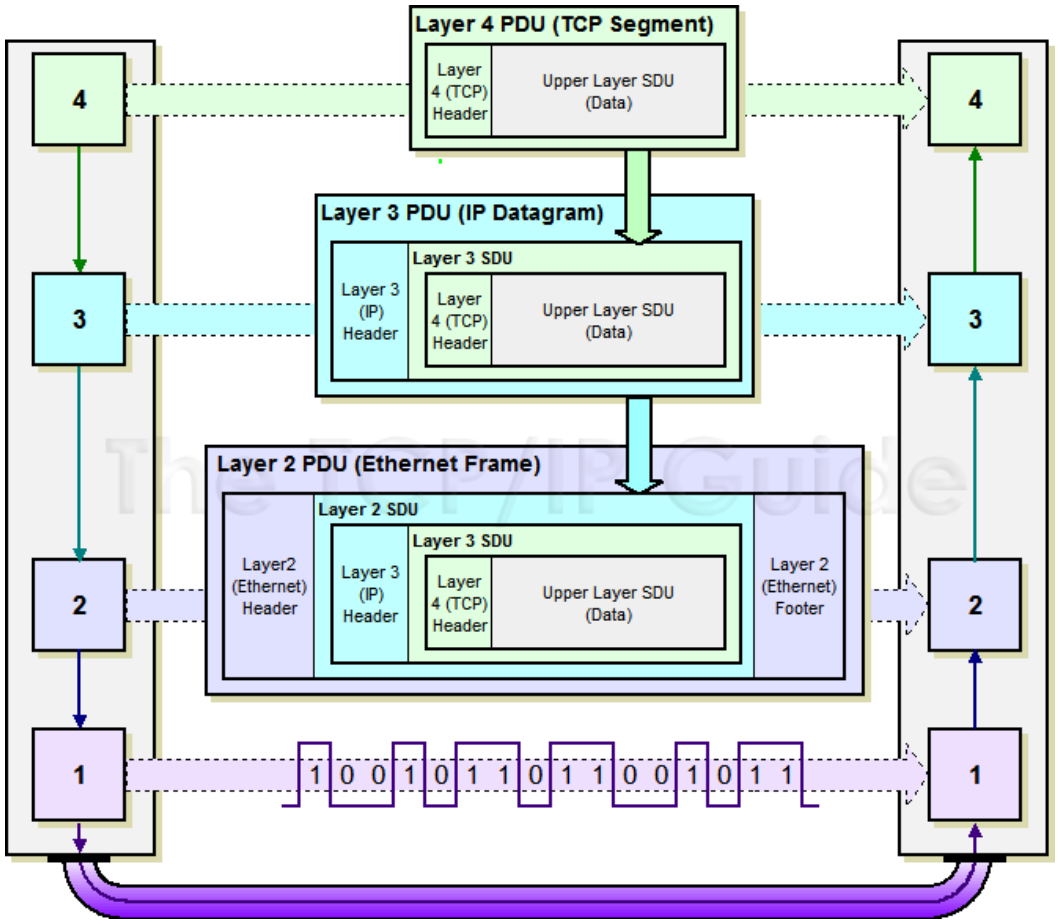


Unidades de informação



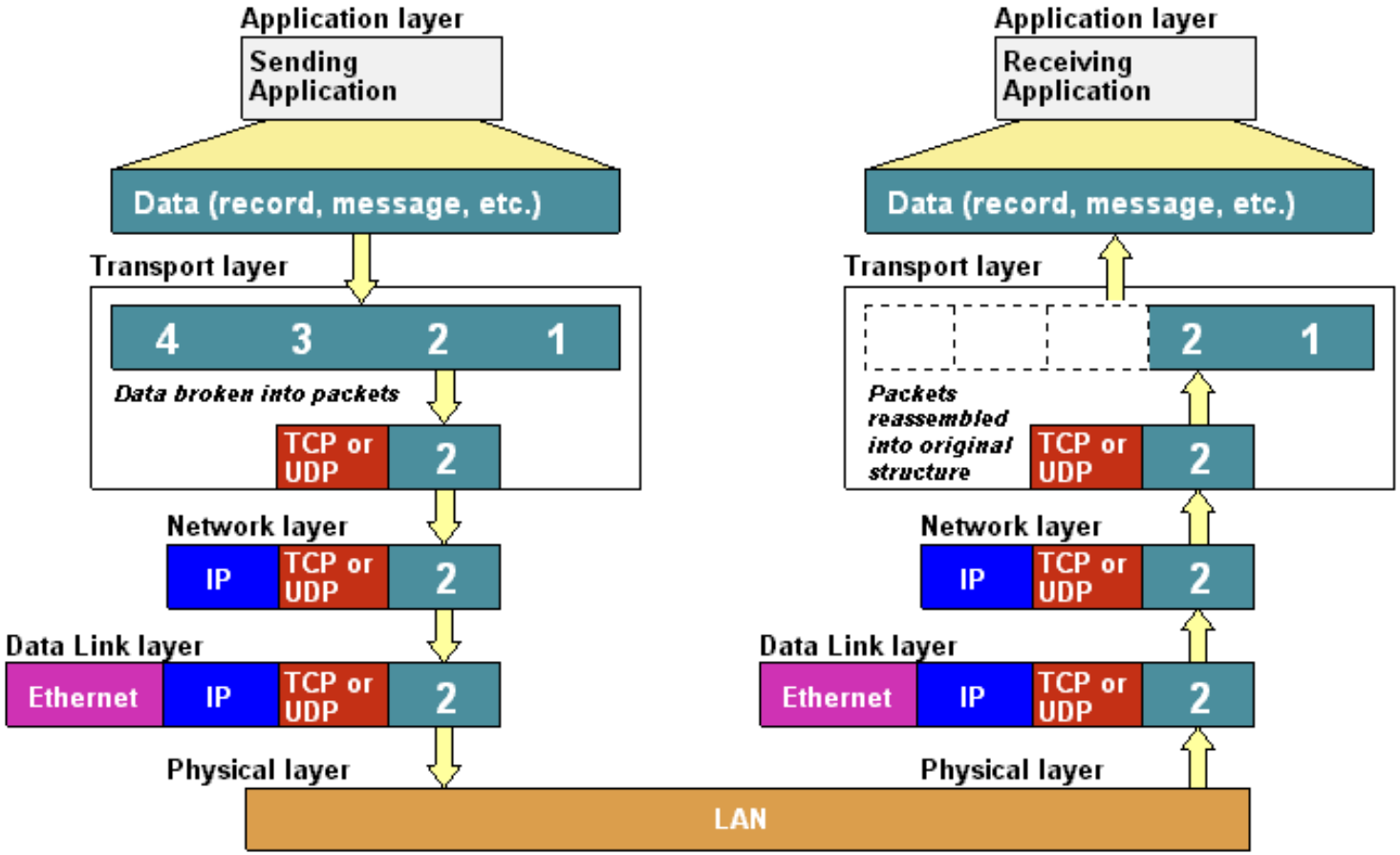
CAMADA	UNIDADE DE INFORMAÇÃO
Aplicação, Apresentação e Sessão	Mensagem ou Dados
Transporte	Segmento
Rede	Pacote ou Datagrama
Enlace	Quadro ou Frame
Física	Bits

Fluxo de dados no modelo TCP/IP

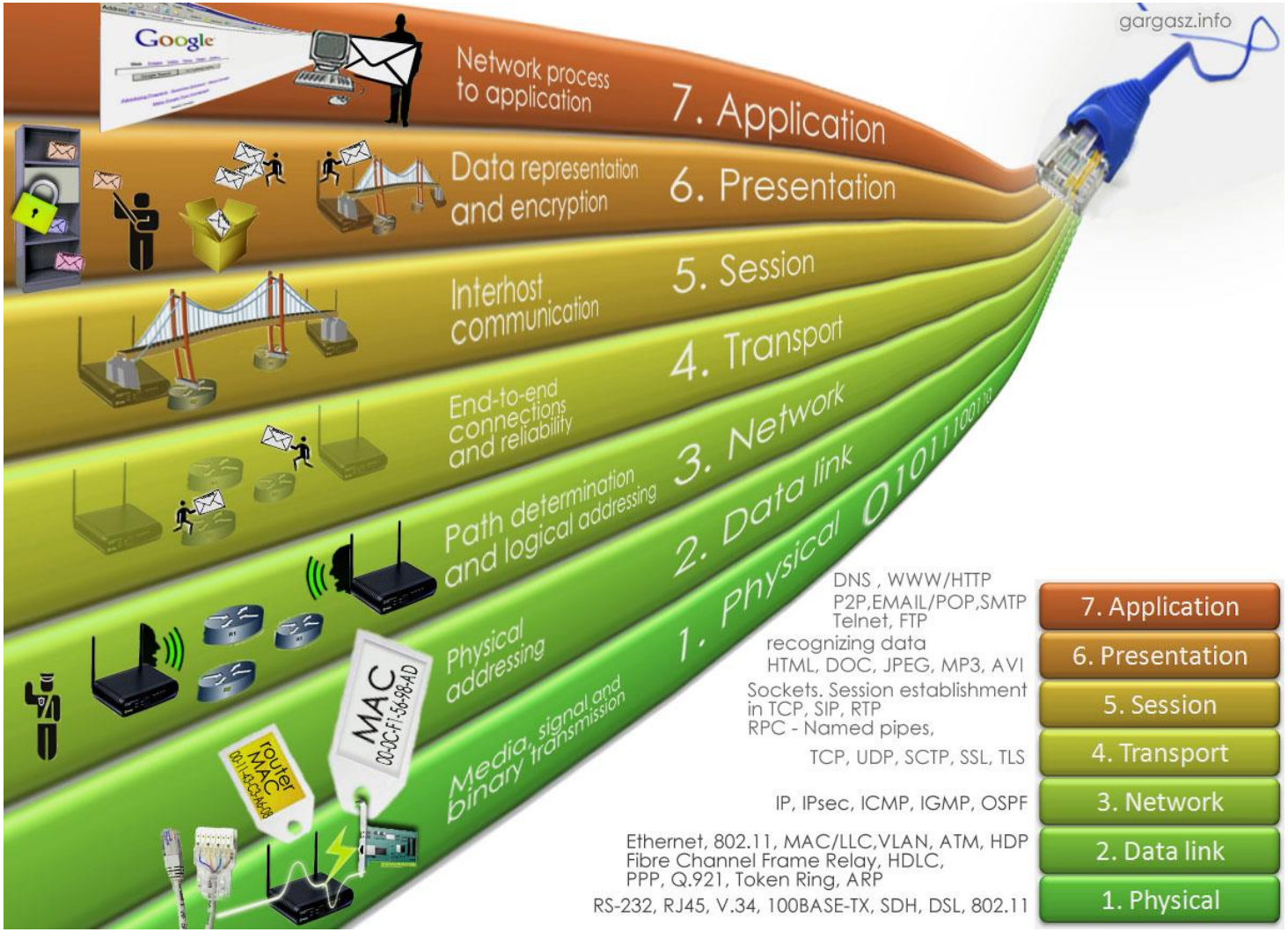


PDU – Protocol Data Unit
SDU – Service Data Unit

Fluxo de dados no modelo TCP/IP



Protocolos TCP/IP



Para saber mais...

… acesse a norma ISO/IEC 7498-1 OSI – Basic Reference Model, da International Organization for Standardization (ISO) e da International Electrotechnical Commission (IEC).

… acesse o material online sobre o Modelo de Referência ISO/OSI, do Prof. Dr. Nilton Alves Jr., do Centro de Pesquisas Físicas, Brasil.

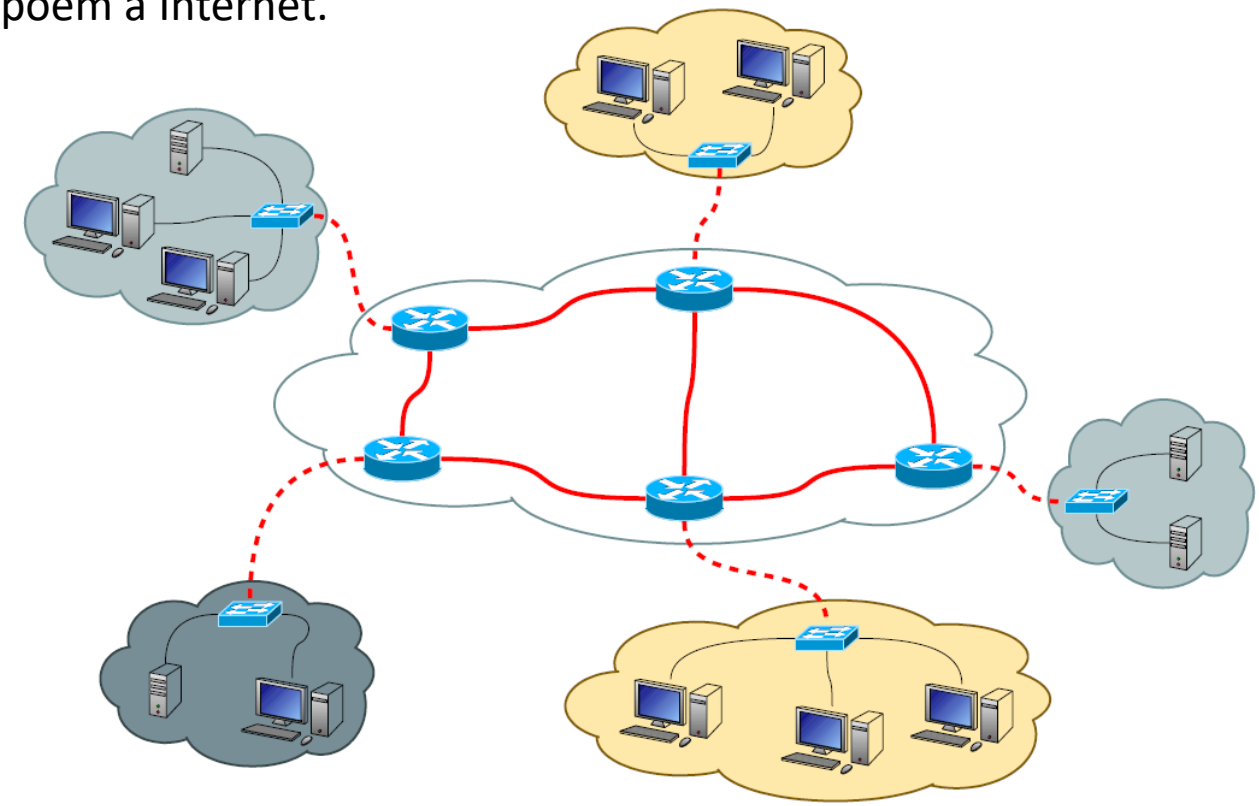
… acesse o artigo OSI Reference Model – The ISO Model of Architecture for Open Systems Interconnection, de Hubert Zimmermann.

Módulo 2

Camada de Rede e Protocolo IP

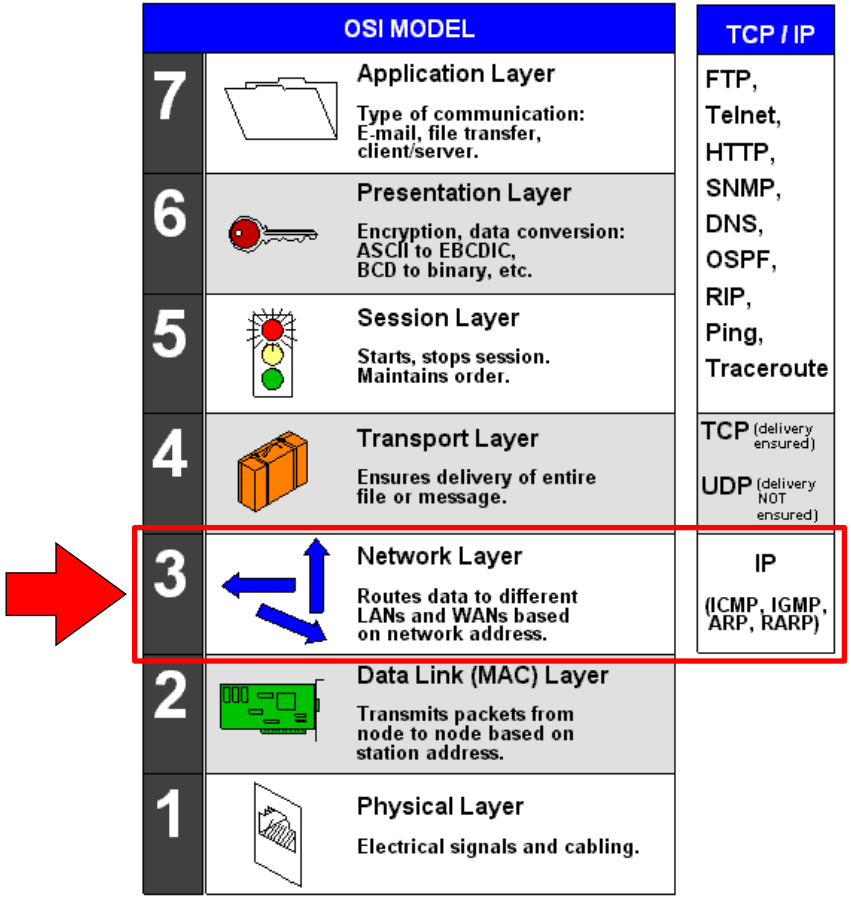
Introdução

A camada de rede é responsável por enviar informações entre a origem e o destino da transmissão de dados pelas diferentes redes e caminhos alternativos que compõem a Internet.



O protocolo IP

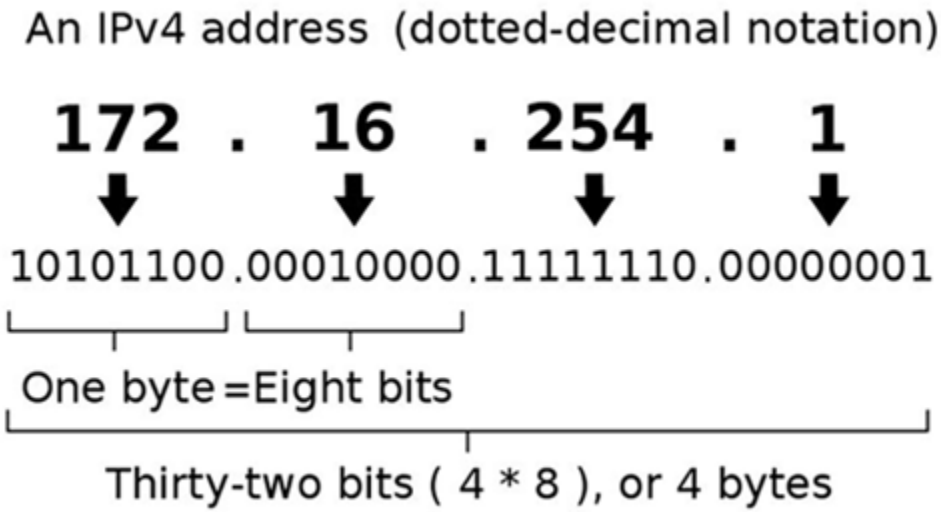
O Internet Protocol, ou simplesmente IP é um protocolo da camada de rede que tem por objetivo identificar unicamente um *host* na rede mundial de computadores e transmitir os datagramas (pacotes) da origem ao destino.



Fonte: Computer Desktop Encyclopedia

Endereço IPv4

O endereço IP é um número binário composto por 32 bits. Cada grupo de 8 bits é conhecido como octeto, de modo que um endereço possui 4 octetos. O endereço IP pode ser escrito na notação binária ou decimal, conforme exemplo abaixo:



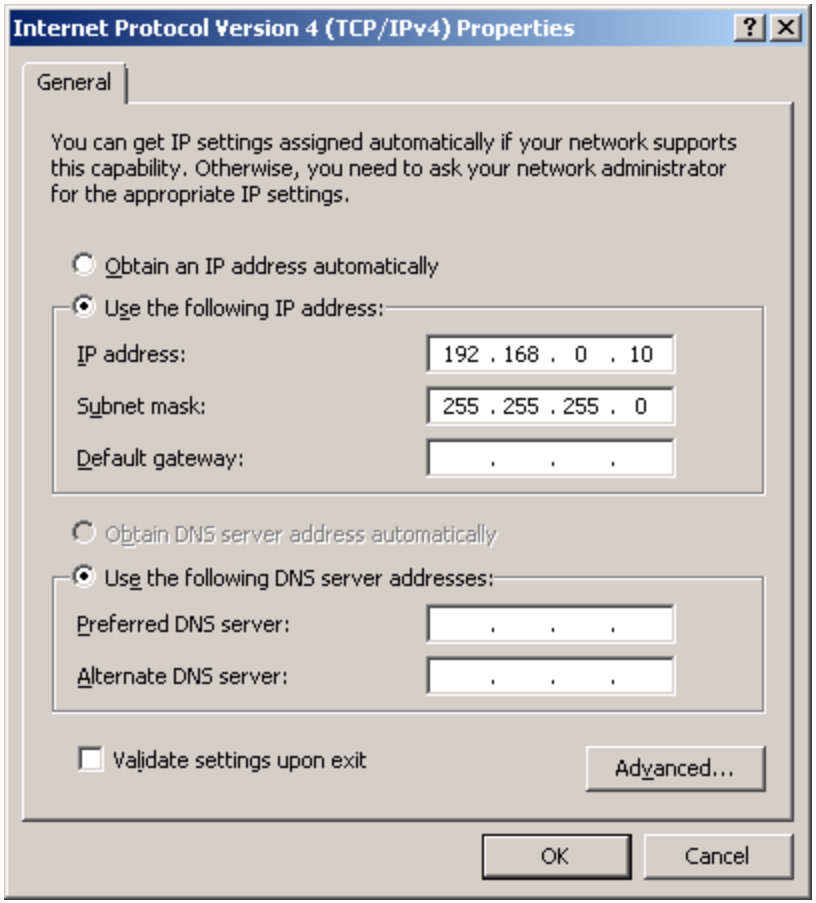
Máscara de rede

Todo *host* para funcionar na rede deve possuir um endereço IP, que o identifica unicamente na rede.

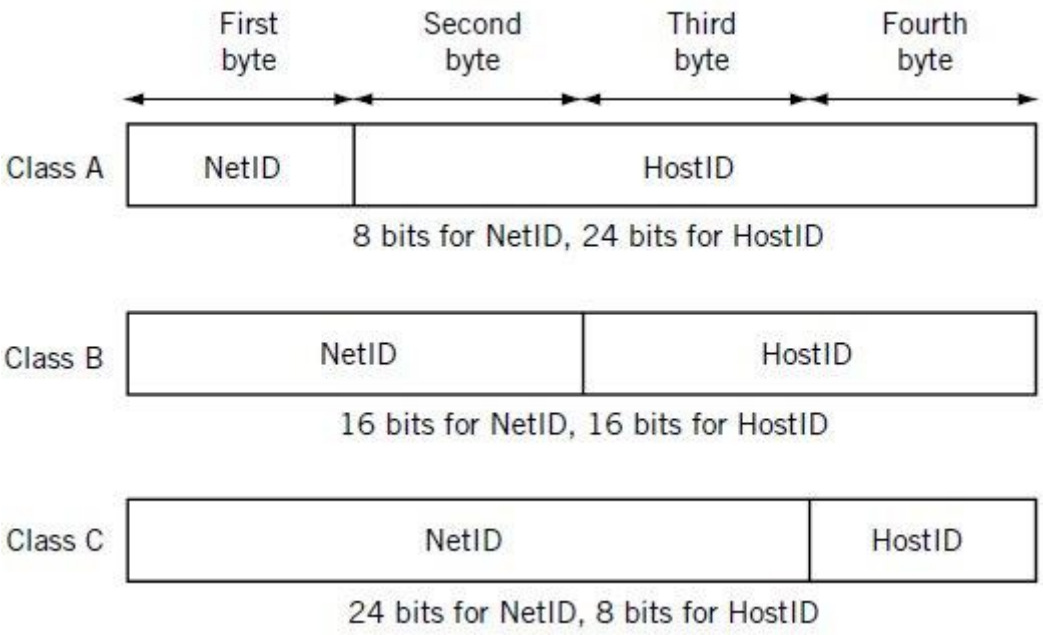
No entanto, o IP carrega duas informações: a rede onde o *host* está conectado e o próprio *host*.

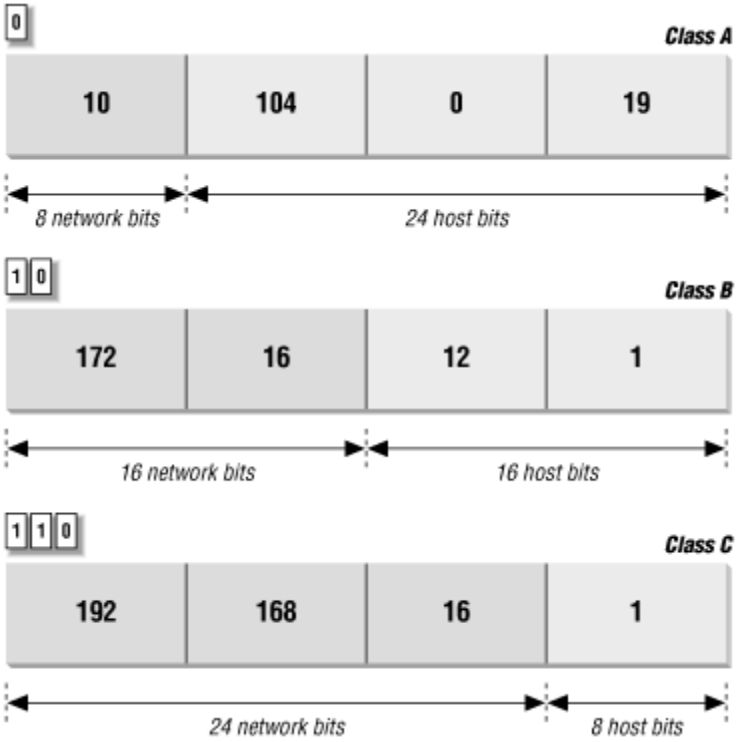
Estas duas informações são obtidas por meio da máscara de rede.

Class A	11111111.00000000.00000000.00000000	255.0.0.0
Class B	11111111.11111111.00000000.00000000	255.255.0.0
Class C	11111111.11111111.11111111.00000000	255.255.255.0

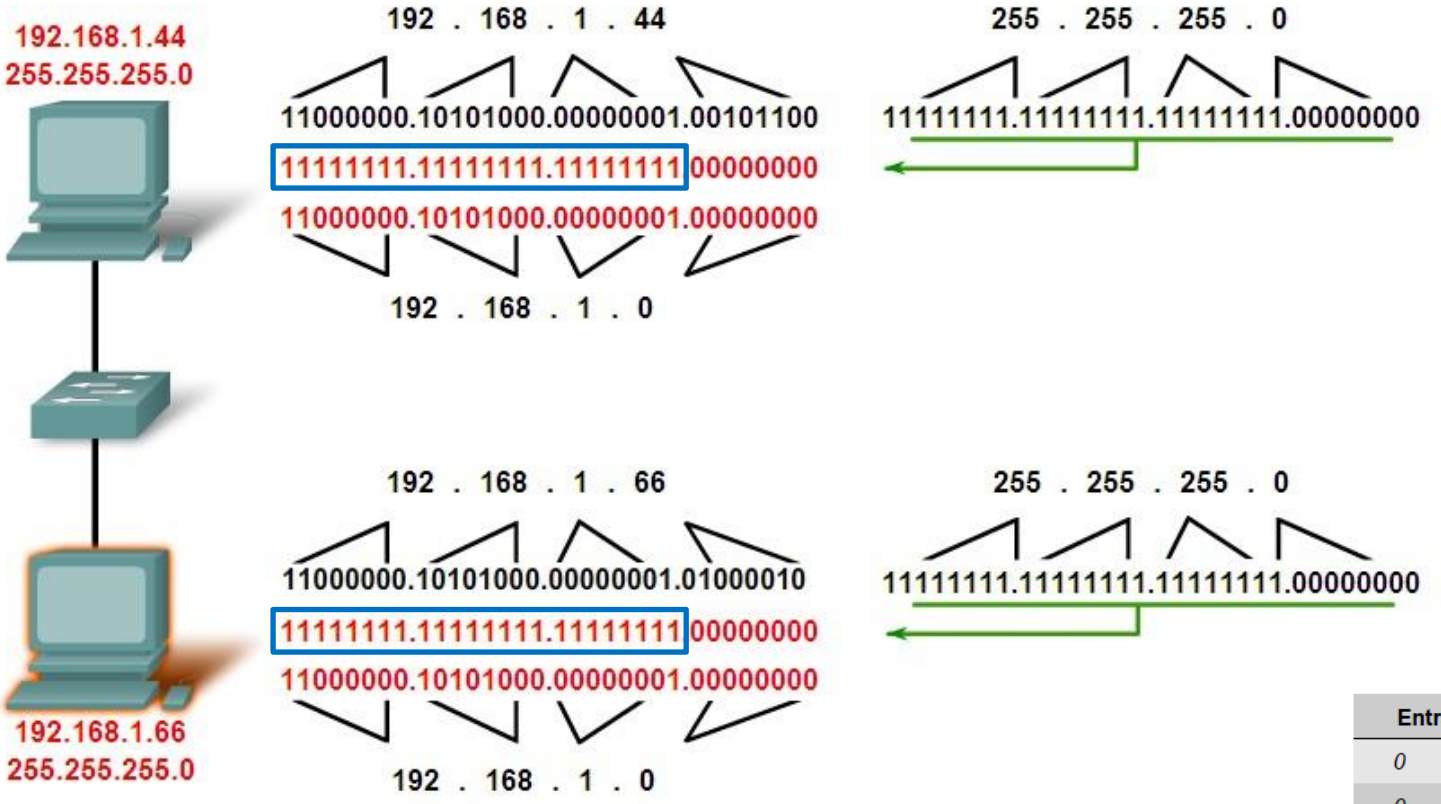


Máscara de rede



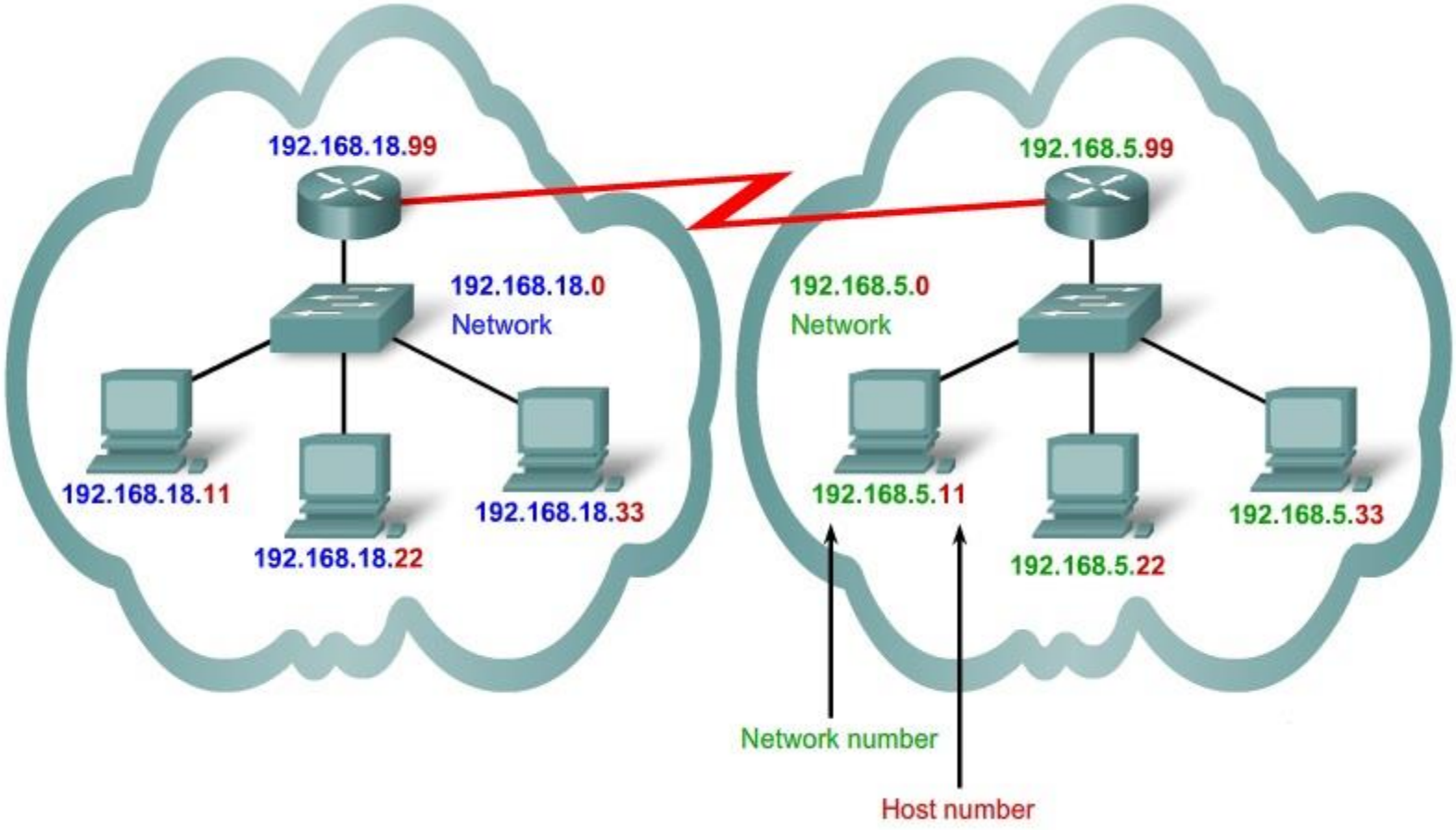


Operação “E” lógico

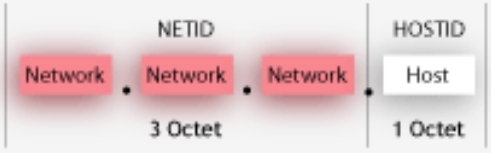


Entradas		Saída
0	0	0
0	1	0
1	0	0
1	1	1

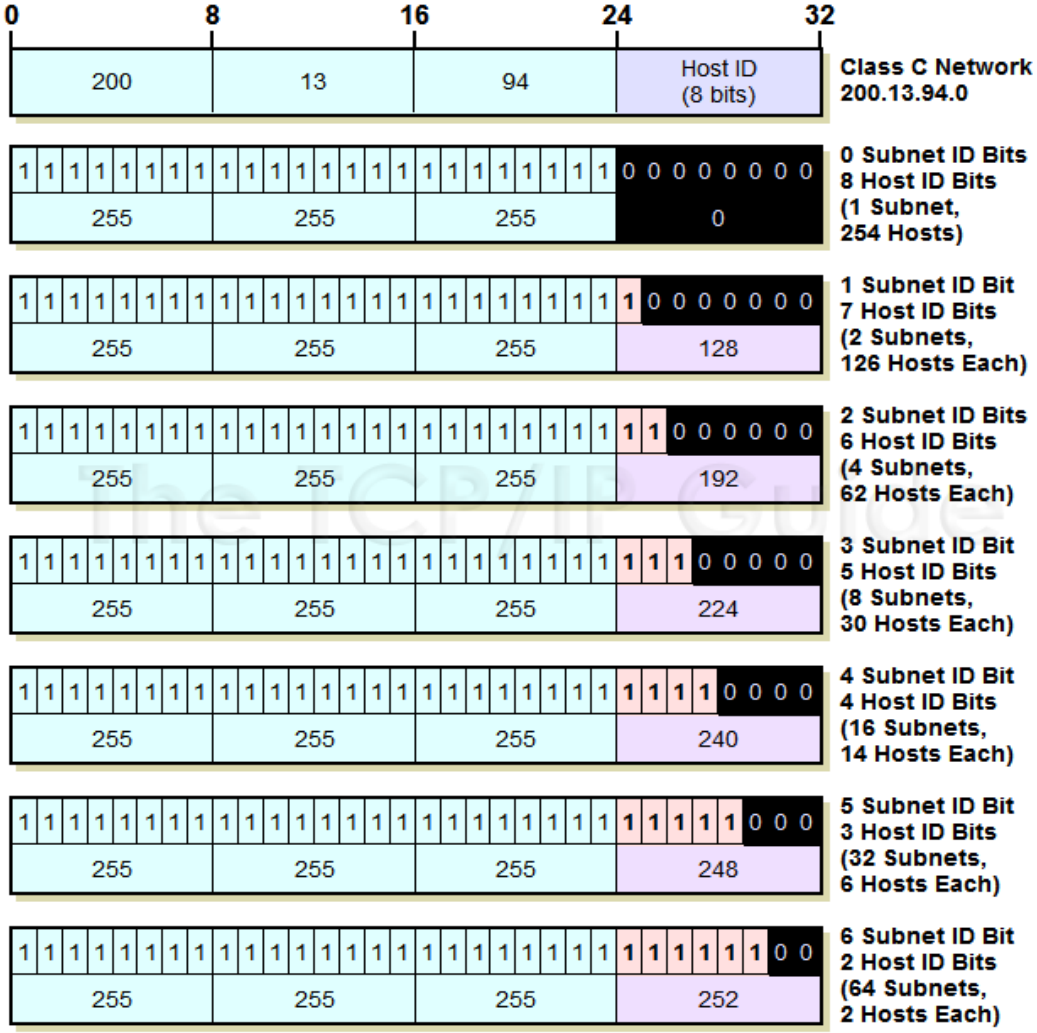
Redes distintas



Máximo de hosts por classe

CLass	First Octet Range	Default Subnet Mask	Max Hosts	Format
A	1-126	255.0.0.0	16M	
B	128-191	255.255.0.0	64K	
C	192-223	255.255.255.0	254	
D	224-239	N/A	N/A	
E	240-255	N/A	N/A	

Criando subredes



Máscara de rede – notação

Máscara	CIDR
0.0.0.0	/0
128.0.0.0	/1
192.0.0.0	/2
224.0.0.0	/3
240.0.0.0	/4
248.0.0.0	/5
252.0.0.0	/6
254.0.0.0	/7

Máscara	CIDR
255.0.0.0	/8
255.128.0.0	/9
255.192.0.0	/10
255.224.0.0	/11
255.240.0.0	/12
255.248.0.0	/13
255.252.0.0	/14
255.254.0.0	/15

Classe A

Máscara	CIDR
255.255.0.0	/16
255.255.128.0	/17
255.255.192.0	/18
255.255.224.0	/19
255.255.240.0	/20
255.255.248.0	/21
255.255.252.0	/22
255.255.254.0	/23

Classe B

Máscara	CIDR
255.255.255.0	/24
255.255.255.128	/25
255.255.255.192	/26
255.255.255.224	/27
255.255.255.240	/28
255.255.255.248	/29
255.255.255.252	/30
255.255.255.254	/31

Classe C

IP público e privado

IP público é todo aquele que pode ser usado na Internet e é visível em toda a rede mundial de computadores.

Já o IP privado é visível apenas dentro de uma rede particular, e não pode ser acessado por outros computadores da Internet.

Além destes, existem ainda endereços IP reservados para fins específicos.

CIDR address block	Description	Reference
0.0.0.0/8	Current network (only valid as source address)	RFC 1700
10.0.0.0/8	Private network	RFC 1918
14.0.0.0/8	Public data networks	RFC 1700
127.0.0.0/8	Loopback	RFC 3330
128.0.0.0/16	Reserved (IANA)	RFC 3330
169.254.0.0/16	Link-Local	RFC 3927
172.16.0.0/12	Private network	RFC 1918
191.255.0.0/16	Reserved (IANA)	RFC 3330
192.0.0.0/24	Reserved (IANA)	RFC 3330
192.0.2.0/24	Documentation and example code	RFC 3330
192.88.99.0/24	IPv6 to IPv4 relay	RFC 3068
192.168.0.0/16	Private network	RFC 1918
198.18.0.0/15	Network benchmark tests	RFC 2544
223.255.255.0/24	Reserved (IANA)	RFC 3330
224.0.0.0/4	Multicasts (former Class D network)	RFC 3171
240.0.0.0/4	Reserved (former Class E network)	RFC 1700
255.255.255.255	Broadcast	

Faixa de endereços IP privados.

IP público e privado

A IANA (Internet Assigned Numbers Authority) reservou três blocos do espaço de endereço IP para redes privadas:

- 10.0.0.0 - 10.255.255.255 (prefixo 10/8);
- 172.16.0.0 - 172.31.255.255 (prefixo 172.16/12);
- 192.168.0.0 - 192.168.255.255 (prefixo 192.168/16).

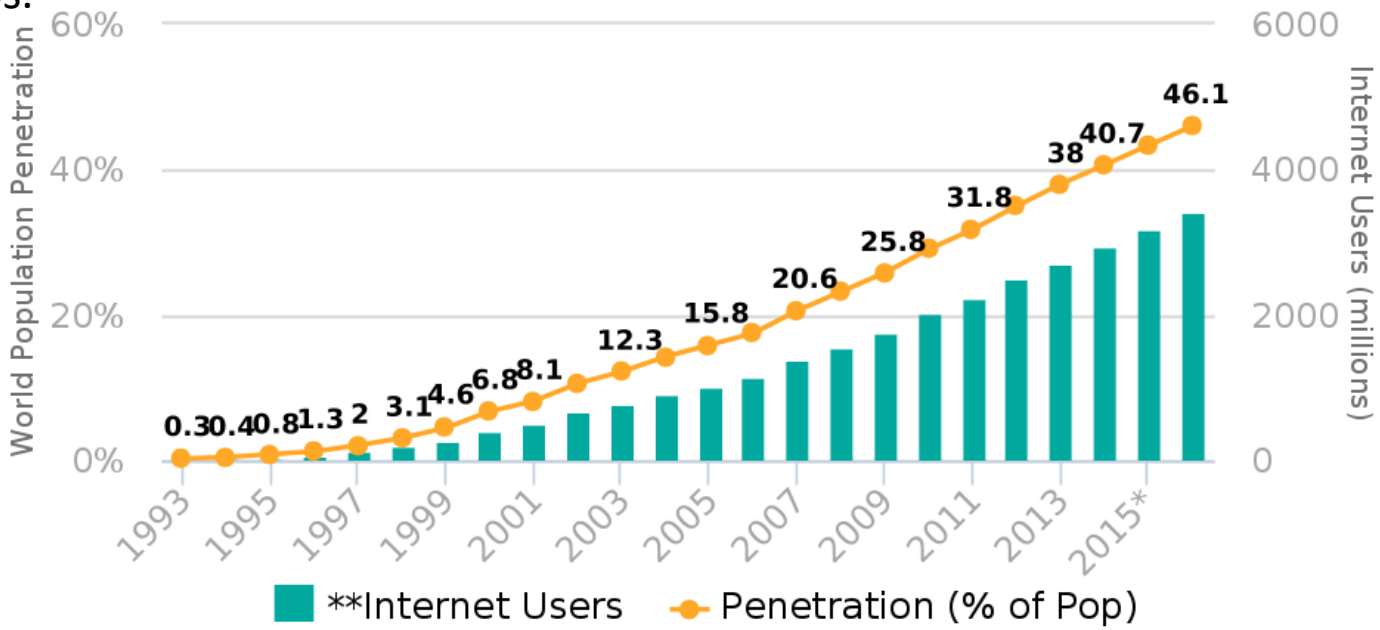
CIDR address block	Description	Reference
0.0.0.0/8	Current network (only valid as source address)	RFC 1700
10.0.0.0/8	Private network	RFC 1918
14.0.0.0/8	Public data networks	RFC 1700
127.0.0.0/8	Loopback	RFC 3330
128.0.0.0/16	Reserved (IANA)	RFC 3330
169.254.0.0/16	Link-Local	RFC 3927
172.16.0.0/12	Private network	RFC 1918
191.255.0.0/16	Reserved (IANA)	RFC 3330
192.0.0.0/24	Reserved (IANA)	RFC 3330
192.0.2.0/24	Documentation and example code	RFC 3330
192.88.99.0/24	IPv6 to IPv4 relay	RFC 3068
192.168.0.0/16	Private network	RFC 1918
198.18.0.0/15	Network benchmark tests	RFC 2544
223.255.255.0/24	Reserved (IANA)	RFC 3330
224.0.0.0/4	Multicasts (former Class D network)	RFC 3171
240.0.0.0/4	Reserved (former Class E network)	RFC 1700
255.255.255.255	Broadcast	

Faixa de endereços IP privados.

Esgotamento do IPv4

Como o IPv4 possui de 32 bits de tamanho, isso dá um total de aproximadamente 4 bilhões de endereços distintos possíveis.

Quando a pilha TCP/IP entrou em operação em 1983, acreditava-se que este número seria mais do que suficiente para endereçar os dispositivos existentes e futuros.

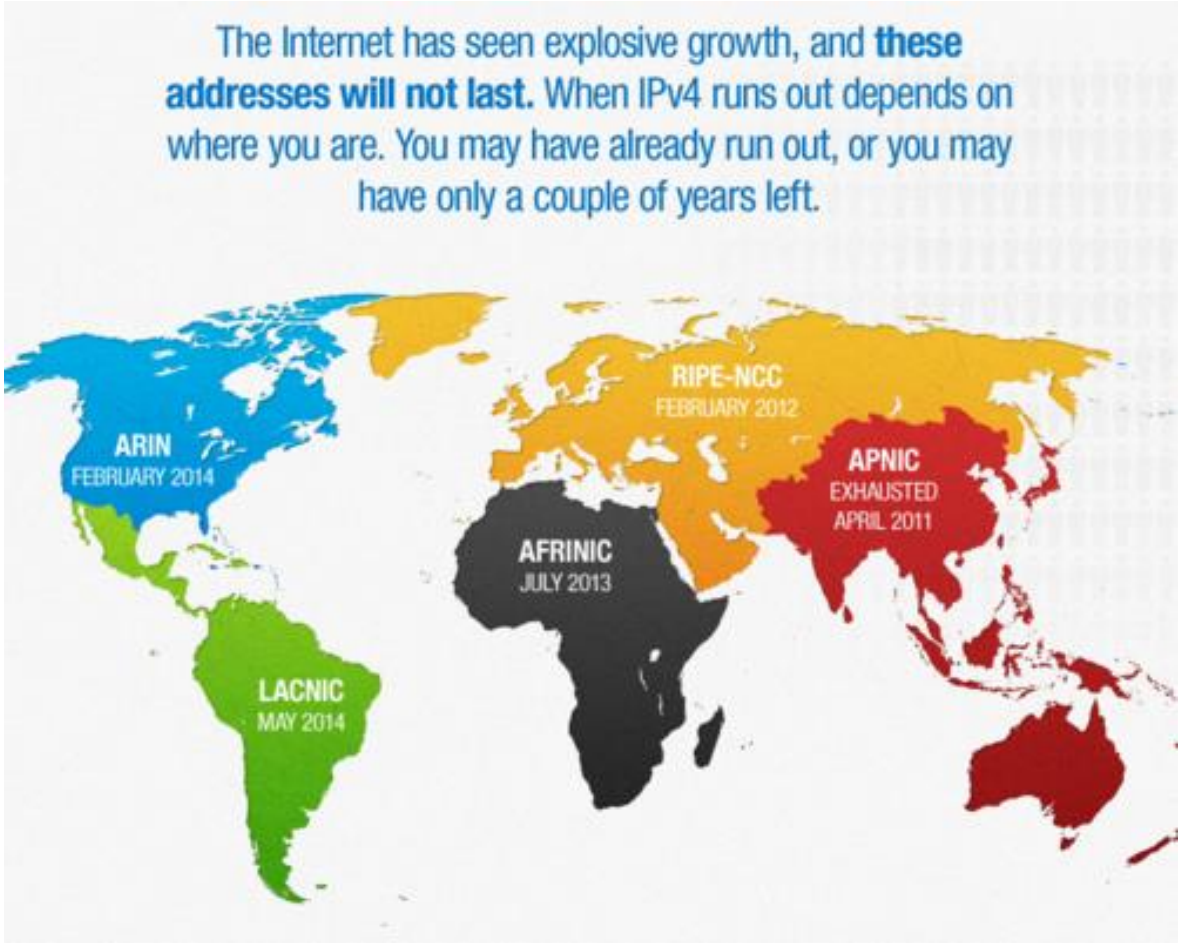


Fonte: statpedia.com com dados extraídos de internetlivestats.com

Esgotamento do IPv4

No entanto, passado mais três décadas, dada a popularização da Internet, o número de usuários tem crescido cada vez mais e mais, e se cada um dos 7 bilhões de habitantes da Terra precisa-se de um endereço IPv4, não haveria como atender a tal demanda.

Esgotamento do IPv4

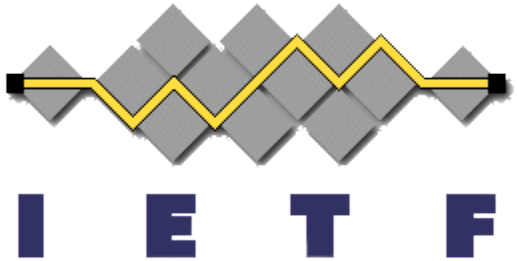


Fonte: blog.cloudflare.com

Esgotamento do IPv4 – Alternativas CIDR

Diante do cenário de esgotamento dos endereços IPv4, a IETF (Internet Engineering Task Force) passou a discutir estratégias para solucionar a questão do esgotamento dos endereços IP e o problema do aumento da tabela de roteamento.

Para isso, em novembro de 1991, é formado um grupo de trabalho denominado ROAD (ROuting and Addressing), que apresentou como solução a estes problemas a utilização do CIDR (Classless Interdomain Routing).

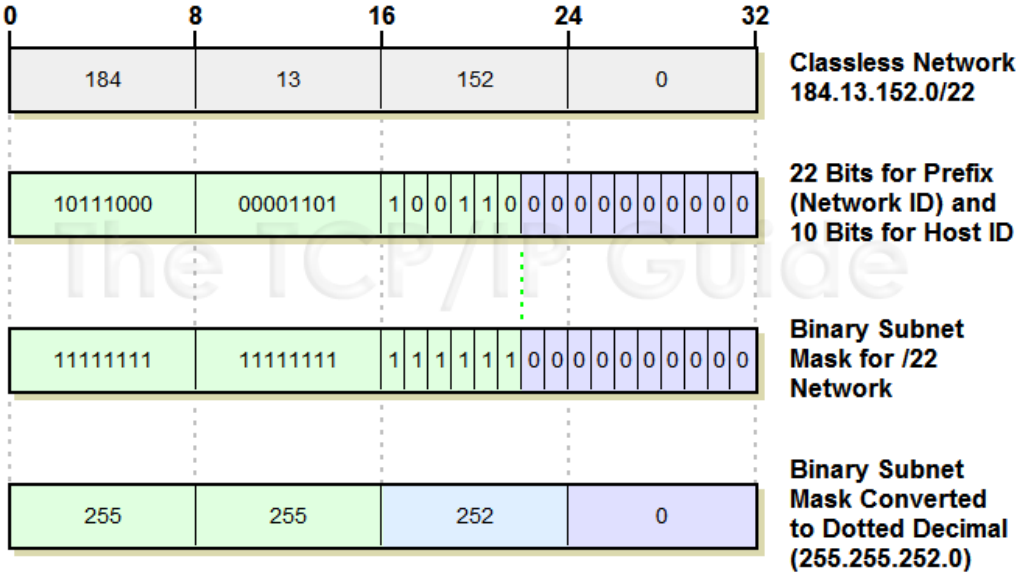


Fonte: ipv6.br

Esgotamento do IPv4 – Alternativas CIDR

Definido na RFC 4632 (tornou obsoleta a RFC 1519), o CIDR tem como ideia básica o fim do uso de classes de endereços, permitindo a alocação de blocos de tamanho apropriado a real necessidade de cada rede; e a agregação de rotas, reduzindo o tamanho da tabela de roteamento.

Com o CIDR os blocos são referenciados como prefixo de redes.

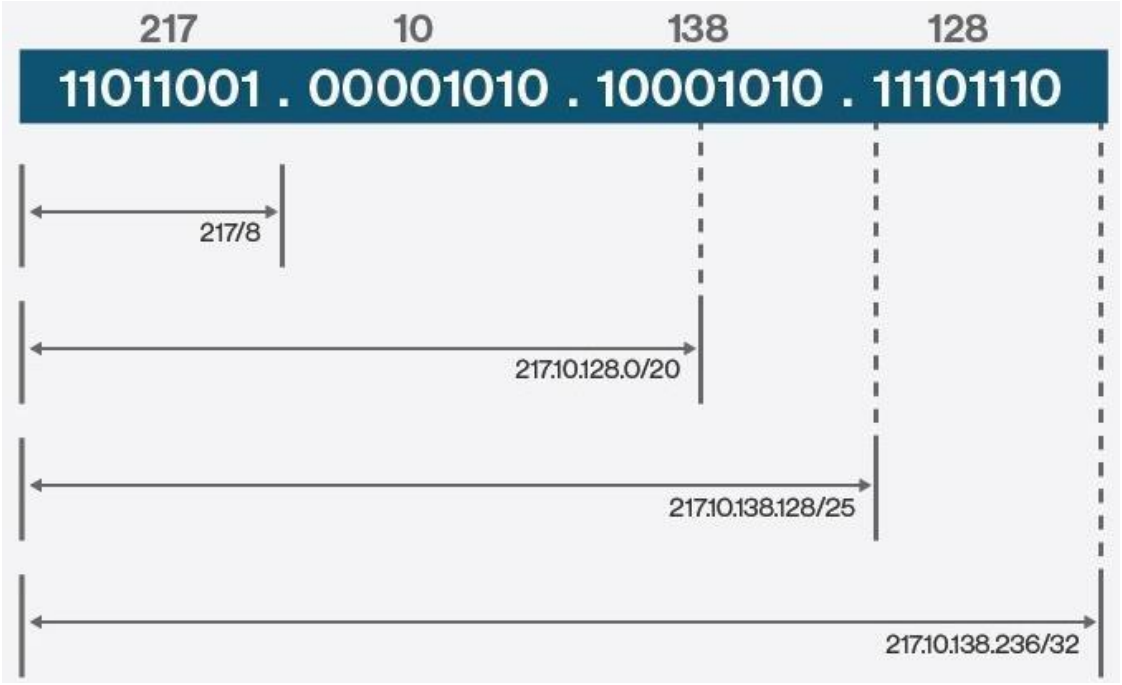


Fonte: ipv6.br

Esgotamento do IPv4 – Alternativas CIDR

Por exemplo, no endereço A.B.C.D/X, os X bits mais significativos indicam o prefixo da rede.

Outra forma de indicar o prefixo é através de máscaras, onde a máscara 255.0.0.0 indica um prefixo /8, 255.255.0.0 indica um /16, e assim sucessivamente.



Fonte: ipv6.br

Esgotamento do IPv4 – Alternativas DHCP

Outra solução, apresentada na RFC 2131 (tornou obsoleta a RFC 1541), foi o protocolo DHCP (Dynamic Host Configuration Protocol).

Através do DHCP um host é capaz de obter um endereço IP automaticamente e adquirir informações adicionais como máscara de rede, endereço do gateway (roteador padrão) e o endereço do servidor DNS local.

O DHCP tem sido muito utilizado por parte dos ISPs (Internet Service Provider) por permitir a atribuição de endereços IP temporários a seus clientes conectados.

Desta forma, torna-se desnecessário obter um endereço para cada cliente, devendo-se apenas designar endereços dinamicamente, através de seu servidor DHCP.

Este servidor terá uma lista de endereços IP disponíveis, e toda vez que um novo cliente se conectar a rede, lhe será designado um desses endereços de forma arbitrária, e no momento que o cliente se desconecta, o endereço é devolvido.

Fonte: ipv6.br

Esgotamento do IPv4 – Alternativas
NAT

O NAT (Network Address Translation) foi outra técnica paliativa desenvolvida para resolver o problema do esgotamento dos endereços IPv4.

Definida na RFC 3022 (tornou obsoleta a RFC 1631), tem como ideia básica permitir que, com um único endereço IP, ou um pequeno número deles, vários hosts possam trafegar na Internet.

Dentro de uma rede, cada computador recebe um endereço IP privado único, que é utilizado para o roteamento do tráfego interno.

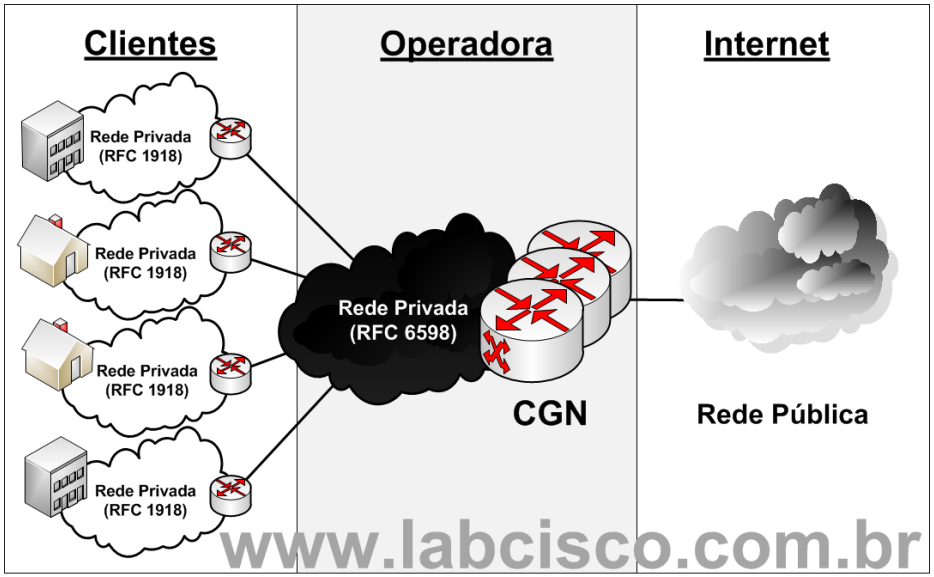
No entanto, quando um pacote precisa ser roteado para fora da rede, uma tradução de endereço é realizada, convertendo endereços IP privados em endereços IP públicos globalmente únicos.

Para tornar possível este esquema, utiliza-se os três intervalos de endereços IP declarados como privados na RFC 1918, sendo que a única regra de utilização é que nenhum pacote contendo estes endereços pode trafegar na Internet pública.

Fonte: ipv6.br

Esgotamento do IPv4 – Alternativas CGN/LSN

O Carrier Grade NAT (CGN) ou Large Scale NAT (LSN), definido na RFC 6264, é uma técnica de tradução de grande porte que vem sendo praticada por algumas operadoras de telecomunicações que não possuem mais endereços IPv4 disponíveis e, portanto, se encontram em situação crítica.



Essa prática consiste em aplicar o NAT na própria rede da operadora, antes mesmo de chegar ao usuário, entregando para seu cliente um endereço privado.

Fonte: labcisco.blogspot.com

IPv6 – especificações

As especificações da IPv6 foram apresentadas inicialmente na RFC 1883 de dezembro de 1995, no entanto, em dezembro de 1998, esta RFC foi substituída pela RFC 2460.

Como principais mudanças em relação ao IPv4 destacam-se:

- **Maior capacidade para endereçamento:** no IPv6 o espaço para endereçamento aumentou de 32 bits para 128 bits, permitindo: níveis mais específicos de agregação de endereços; identificar uma quantidade muito maior de dispositivos na rede; e implementar mecanismos de autoconfiguração;
- **Simplificação do formato do cabeçalho:** alguns campos do cabeçalho IPv4 foram removidos ou tornaram-se opcionais, com o intuito de reduzir o custo do processamento dos pacotes nos roteadores;

Fonte: ipv6.br

IPv6 – especificações

• **Suporte a cabeçalhos de extensão:** as opções não fazem mais parte do cabeçalho base, permitindo um roteamento mais eficaz, limites menos rigorosos em relação ao tamanho e a quantidade de opções, e uma maior flexibilidade para a introdução de novas opções no futuro;

• **Capacidade de identificar fluxos de dados:** foi adicionado um novo recurso que permite identificar de pacotes que pertençam a determinados tráfegos de fluxos, para os quais podem ser requeridos tratamentos especiais;

• **Suporte a autenticação e privacidade:** foram especificados cabeçalhos de extensão capazes de fornecer mecanismos de autenticação e garantir a integridade e a confidencialidade dos dados transmitidos.

Fonte: ipv6.br

IPv6 – distribuição de endereços

Como não pode haver repetição de endereços, eles são um recurso que tem de ser gerenciado de forma centralizada na Internet.

O autoridade responsável por este controle é a IANA (Internet Assigned Numbers Authority).

Fonte: ipv6.br

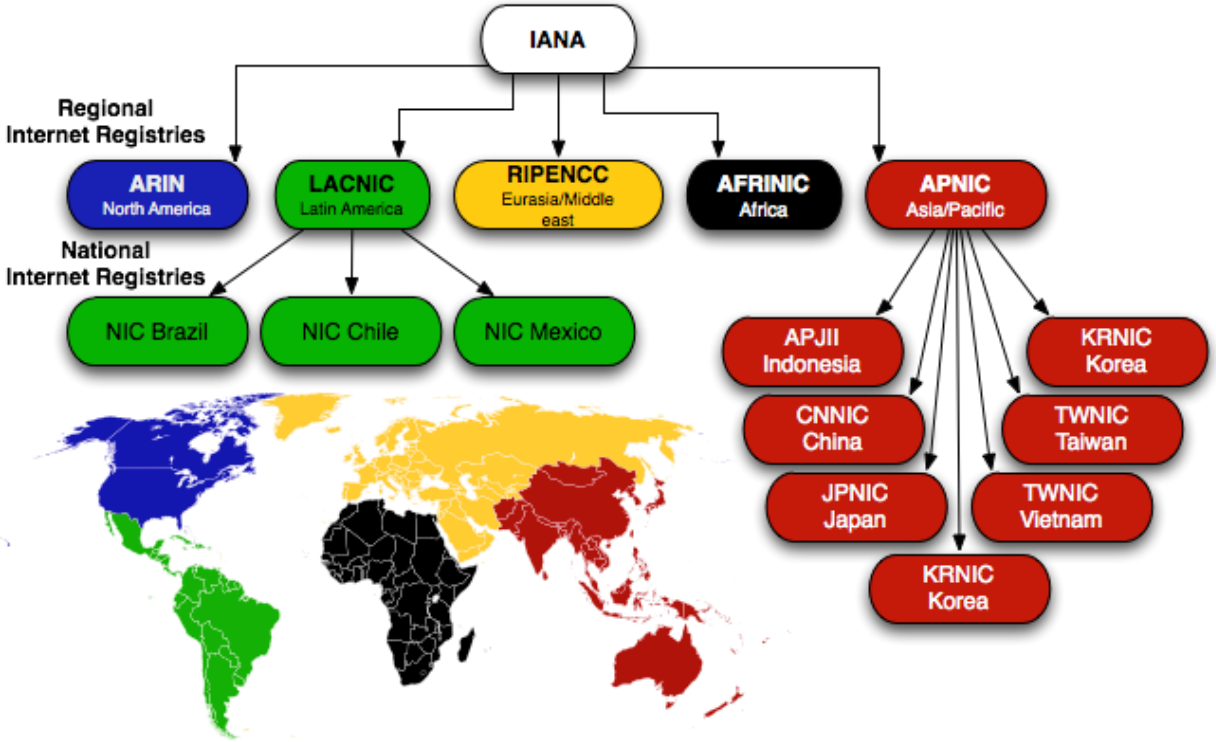
IPv6 – distribuição de endereços

Atualmente a função da IANA é realizada pela ICANN (Internet Corporation for Assigned Names and Numbers), e a estrutura de distribuição de IPs é hierárquica, contando também com organizações regionais, chamadas de RIR (Regional Internet Registries) e em alguns casos, estruturas nacionais, chamadas de NIR (National Internet Registries).

Fonte: ipv6.br

IPv6 – distribuição de endereços

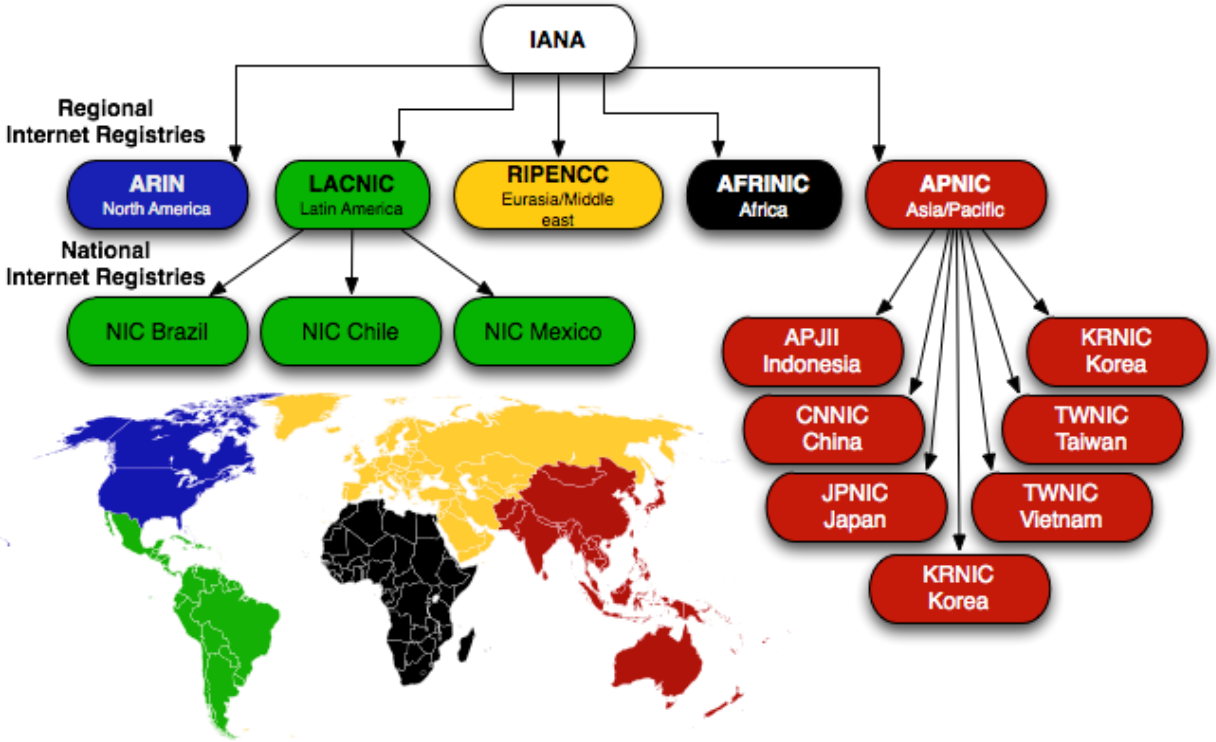
Há cinco RIRs: o ARIN, na América do Norte, o LACNIC, na América Latina e Caribe, o RIPENCC, abrangendo a Europa e parte da Ásia, o AFRINIC, na África e o APNIC, na região da Ásia e Oceania.



Fonte: ipv6.br

IPv6 – distribuição de endereços

Cada uma dessas organizações é responsável por definir as regras de distribuição dos endereços em sua respectiva área de atuação, bem como implementa-las.



Fonte: ipv6.br

IPv6 – distribuição de endereços

Em alguns países há entidades nacionais para a distribuição dos IPs.

É o caso do Brasil, por exemplo, onde o NIC.br é quem gerencia esse recurso.

lacnic

nic.br

NIC CHILE
somos el punto CL

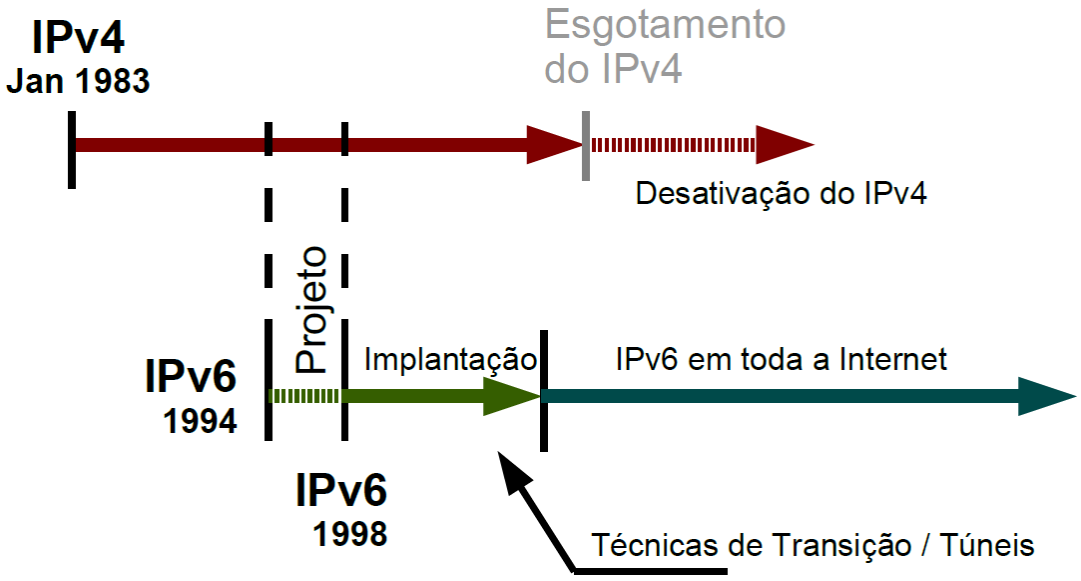
NIC MÉXICO

Fonte: ipv6.br

IPv6 – transição

O IPv6 foi projetado de tal forma que não é compatível com o IPv4. Eles não podem interoperar diretamente.

Assim, o plano inicial era fazer uma transição gradual, mantendo o IPv4 e adicionando o IPv6 em todos os dispositivos da Internet ao longo do tempo, de forma que, antes dos endereços livres IPv4 esgotaram-se, o IPv6 estivesse instalado em toda a Internet.

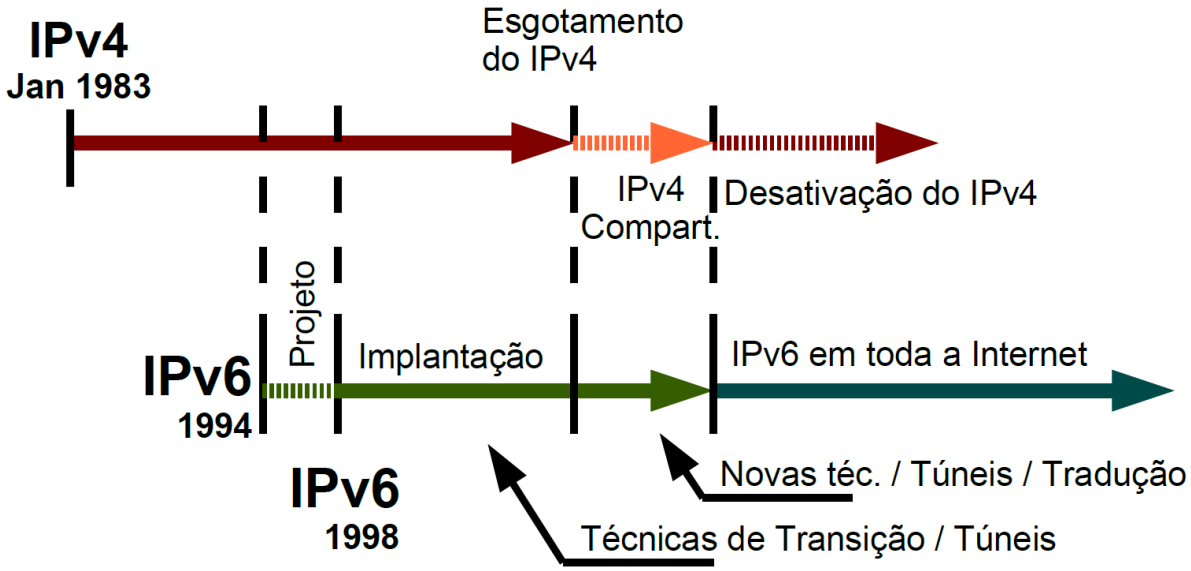


Fonte: ipv6.br

IPv6 – transição

No Brasil, construiu-se o cronograma abaixo como referência para a implantação do protocolo no país.

Ele foi construído com base no diálogo com diversos provedores de acesso, de serviços e operadoras de telecomunicações, em diversas reuniões de coordenação ao longo dos anos de 2011 e 2012.

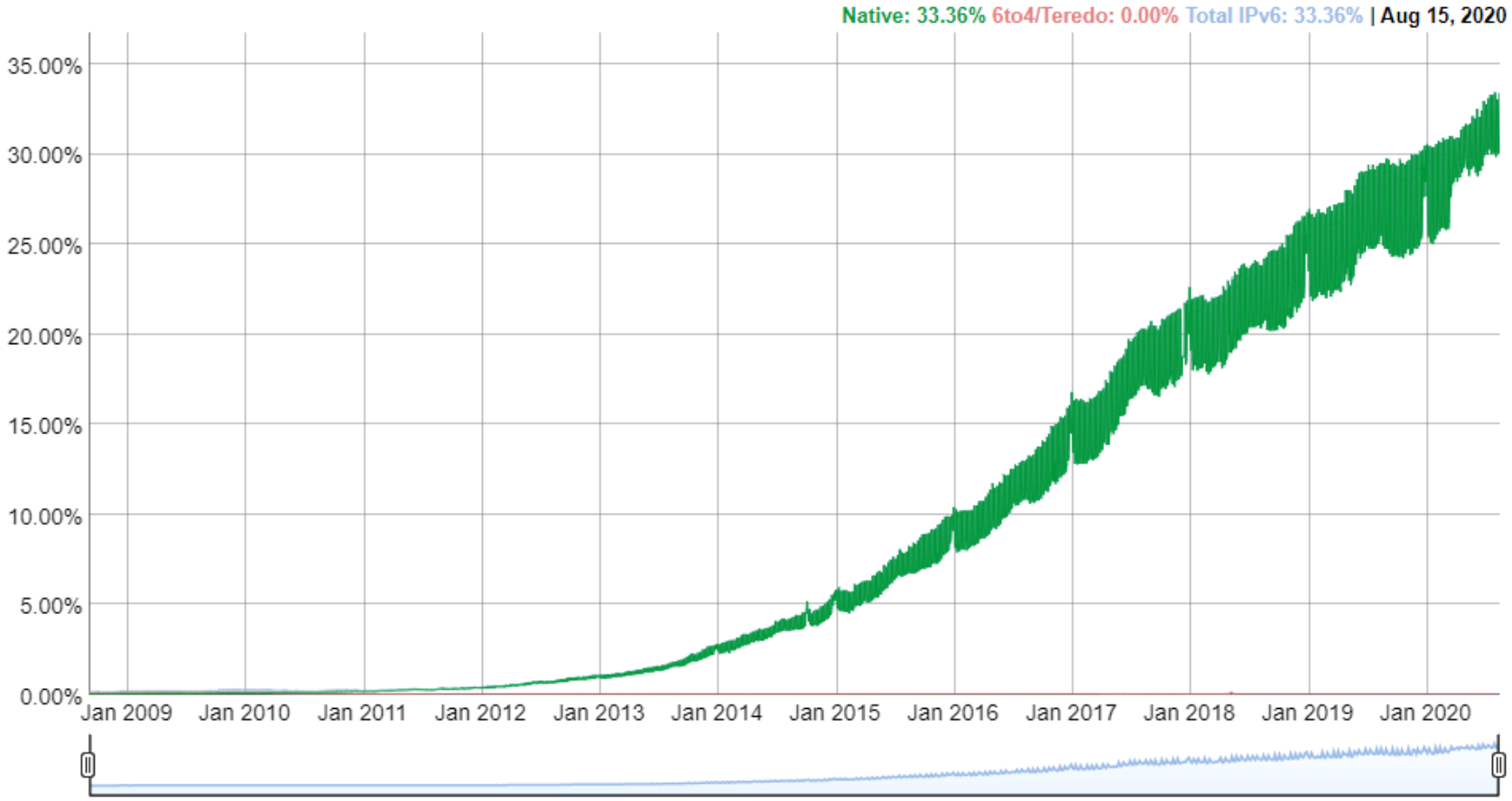


Fonte: ipv6.br

IPv6 – adoção

IPv6 Adoption

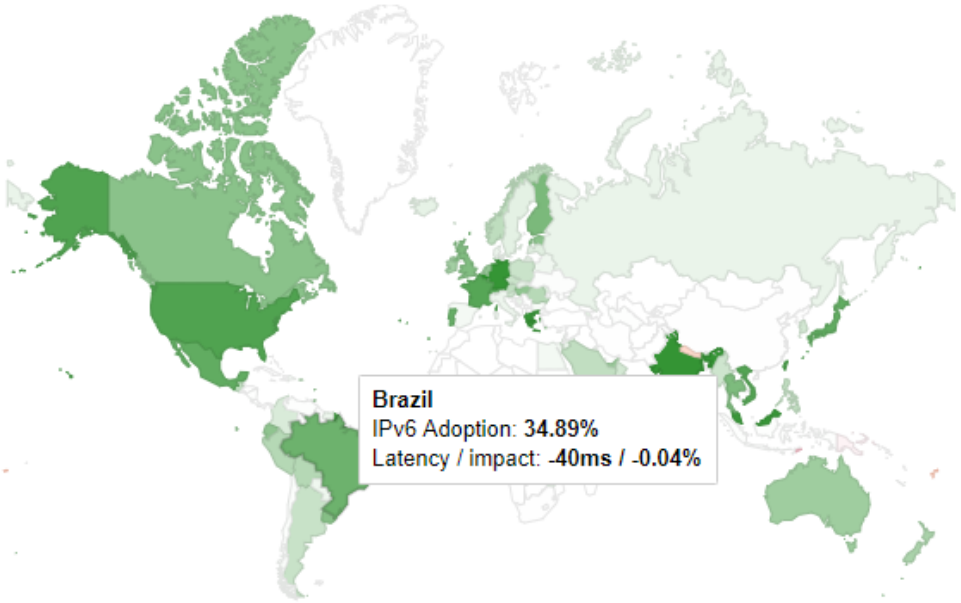
We are continuously measuring the availability of IPv6 connectivity among Google users. The graph shows the percentage of users that access Google over IPv6.



Fonte: www.google.com/intl/en/ipv6/statistics.html

IPv6 – adoção

Per-Country IPv6 adoption



World | Africa | Asia | Europe | Oceania | North America | Central America | Caribbean | South America

The chart above shows the availability of IPv6 connectivity around the world.

- Regions where IPv6 is more widely deployed (the darker the green, the greater the deployment) and users experience infrequent issues connecting to IPv6-enabled websites.
- Regions where IPv6 is more widely deployed but users still experience significant reliability or latency issues connecting to IPv6-enabled websites.
- Regions where IPv6 is not widely deployed and users experience significant reliability or latency issues connecting to IPv6-enabled websites.

Fonte: www.google.com/intl/en/ipv6/statistics.html

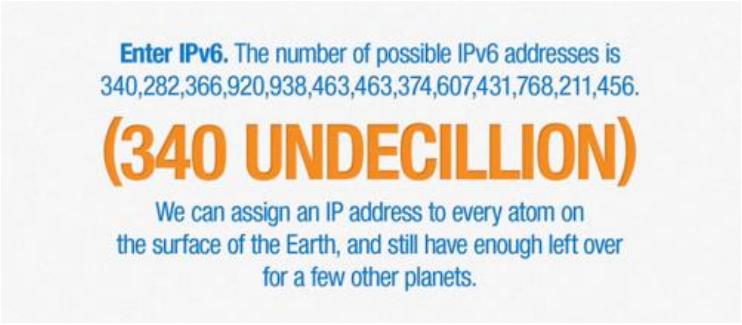
IPv4 e IPv6

O IPv4 é um protocolo de rede com tamanho de 32 bits, o que significa que ele possui 2^{32} combinações possíveis, ou seja:

$$2^{32} = 4.294.967.296\ endereços$$

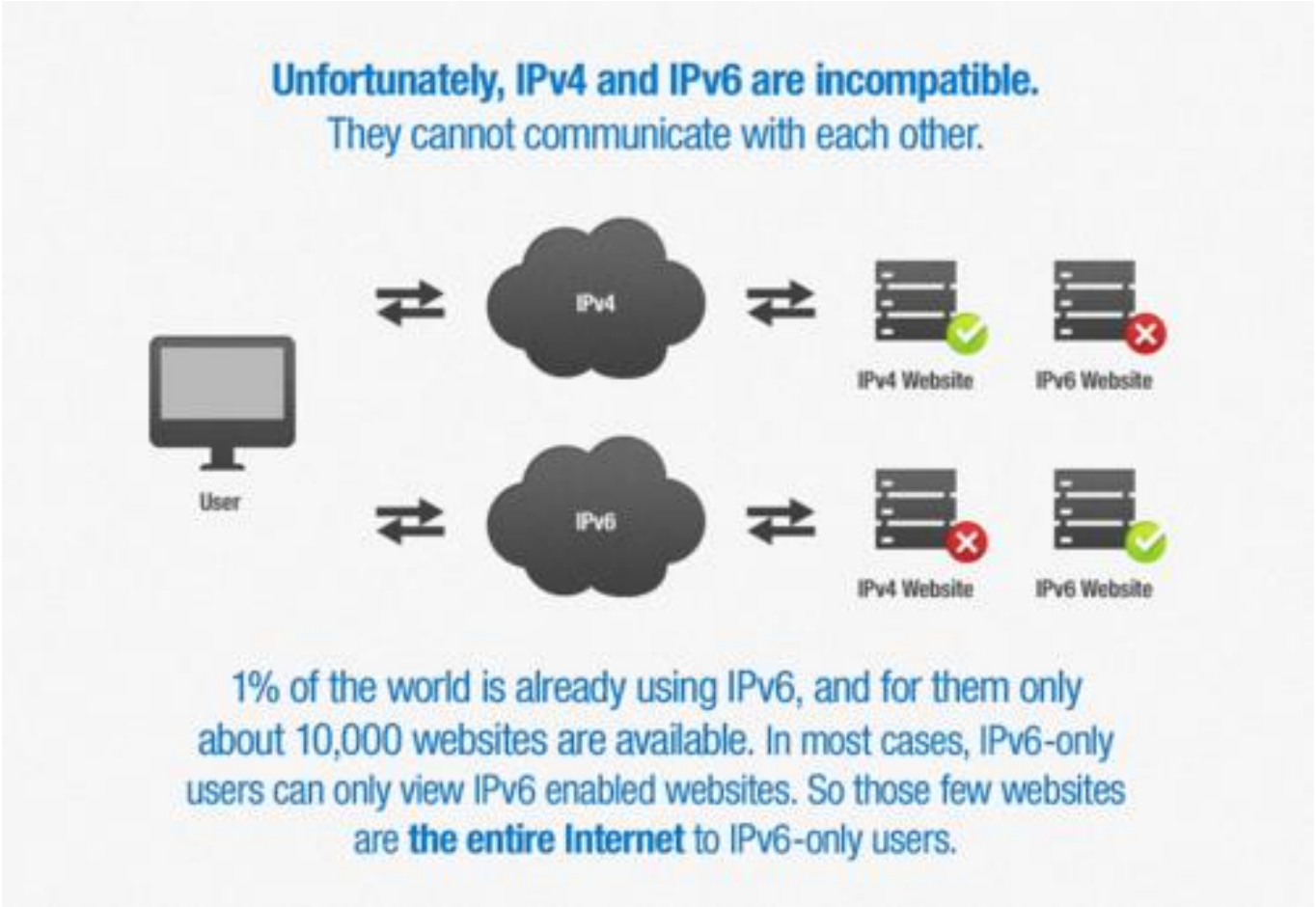
Já o protocolo IPv6 tem o tamanho de 128 bits, o significa que ele possui 2^{128} combinações possíveis, ou seja:

$$2^{128} = 340.282.366.920.938.463.463.374.607.431.768.211.456\ endereços$$



Fonte: ipv6.br e blog.cloudfare.com

IPv4 e IPv6



Fonte: blog.cloudfare.com

Endereço IPv6

Os 32 bits dos endereços IPv4 são divididos em quatro grupos de 8 bits cada, separados por ".", escritos com dígitos **decimais**. Por exemplo:

192.168.0.10

A representação dos endereços IPv6 divide o endereço em oito grupos de 16 bits, separando-os por ":", escritos com dígitos **hexadecimais** (0-F). Por exemplo:

2001:0DB8:AD1F:25E2:CADE:CAFE:F0CA:84C1

Na representação de um endereço IPv6, é permitido utilizar tanto caracteres maiúsculos quanto minúsculos.

Fonte: ipv6.br

Endereço IPv6 – regras de abreviação

Além disso, regras de abreviação podem ser aplicadas para facilitar a escrita de alguns endereços muito extensos. É permitido omitir os zeros a esquerda de cada bloco de 16 bits, além de substituir uma sequência longa de zeros por "::".

Por exemplo, o endereço:

2001:0DB8:0000:0000:130F:0000:0000:140B

pode ser escrito como:

2001:DB8::130F:0:0:140B

--ou--

2001:DB8:0:0:130F::140B

Neste caso a abreviação do grupo de zeros só pode ser realizada uma única vez, caso contrário poderá haver ambiguidade na representação do endereço.

Fonte: ipv6.br

Endereço IPv6 – regras de abreviação

Se o endereço:

2001:0DB8:0000:0000:130F:0000:0000:140B

fosse escrito como:

2001:DB8::130F::140B

não seria possível determinar se ele corresponde a:

2001:DB8:0:0:130F:0:0:140B

--ou--

2001:DB8:0:0:0:130F:0:140B

--ou--

2001:DB8:0:130F:0:0:0:140B

Fonte: ipv6.br

Endereço IPv6 – regras de abreviação

Esta abreviação também pode ser feita no final ou no início do endereço, como ocorre em:

2001:DB8:0:54:0:0:0:0

que pode ser escrito da forma:

2001:DB8:0:54::

Fonte: ipv6.br

Endereço IPv6 – prefixos de rede

Outra representação importante é a dos prefixos de rede. Em endereços IPv6 ela continua sendo escrita do mesmo modo que no IPv4, utilizando a notação CIDR.

Esta notação é representada da forma “endereço IPv6/tamanho do prefixo”, onde “tamanho do prefixo” é um valor decimal que especifica a quantidade de bits contíguos à esquerda do endereço que compreendem o prefixo.

O exemplo de prefixo de subrede apresentado a seguir indica que dos 128 bits do endereço, 64 bits são utilizados para identificar a subrede:

- Prefixo **2001:db8:3003:2::/64**
- Prefixo global **2001:db8::/32**
- ID da subrede **3003:2**

Esta representação também possibilita a agregação dos endereços de forma hierárquica, identificando a topologia da rede através de parâmetros como posição geográfica, provedor de acesso, identificação da rede, divisão da subrede, etc. Com isso, é possível diminuir o tamanho da tabela de roteamento e agilizar o encaminhamento dos pacotes.

Fonte: ipv6.br

Endereço IPv6 – URLs

Com relação a representação dos endereços IPv6 em URLs (Uniform Resource Locators), estes agora passam a ser representados entre colchetes.

Deste modo, não haverá ambiguidades caso seja necessário indicar o número de uma porta juntamente com a URL.

Exemplos:

http://[**2001:12ff:0:4::22**]/index.html

http://[**2001:12ff:0:4::22**]:8080

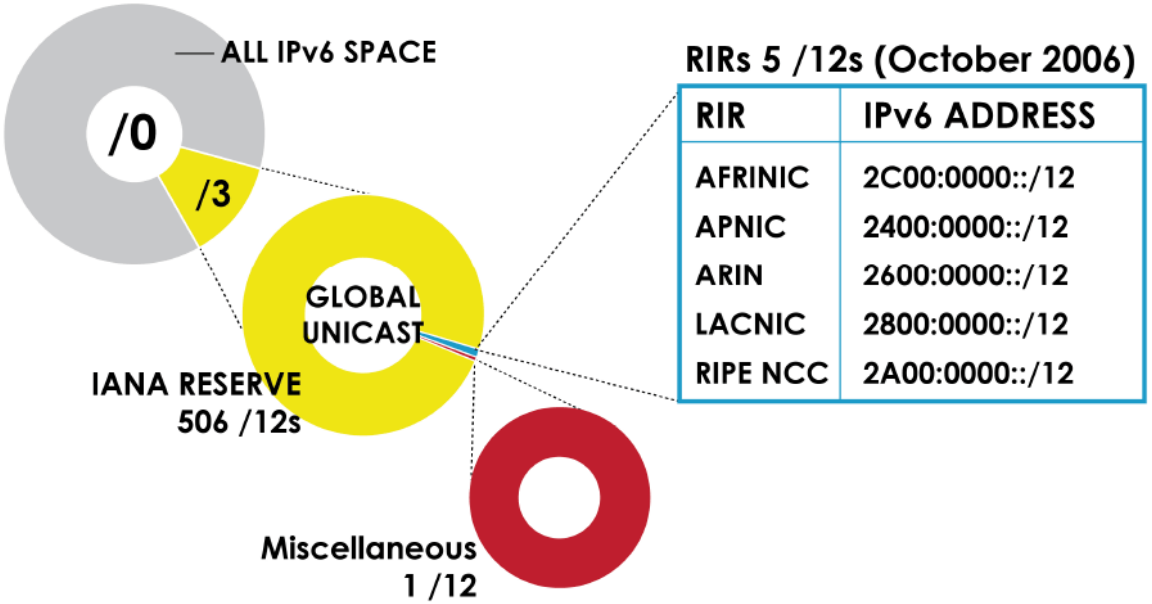
Fonte: ipv6.br

Endereço IPv6 – alocação e designação

Na hierarquia das políticas de atribuição, alocação e designação de endereços, cada RIR recebe da IANA um bloco /12 IPv6.

O bloco **2800::/12** corresponde ao espaço reservado para o LACNIC alocar na América Latina.

O NIC.br por sua vez, trabalha com a subrede **2801::/16** que faz parte deste /12.

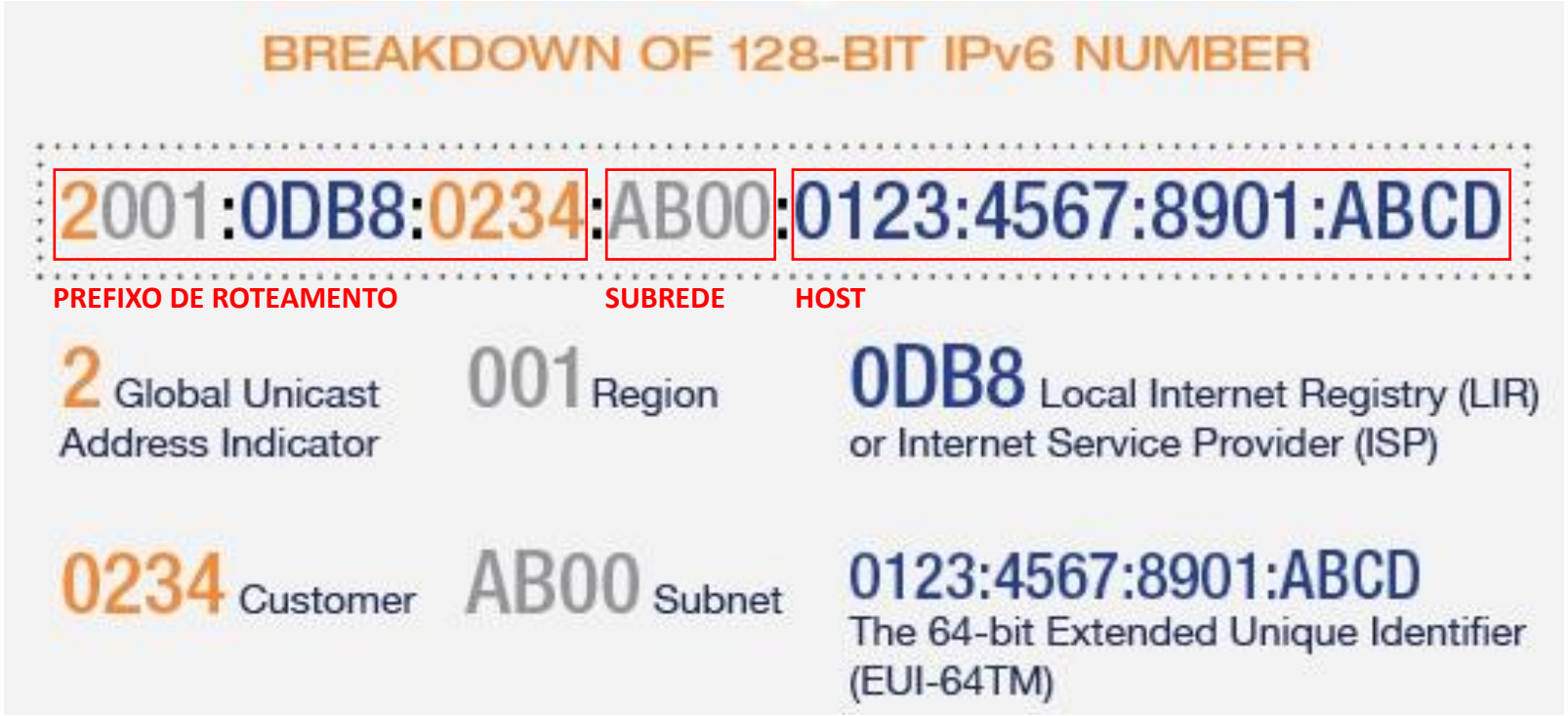


RIRs 5 /12s (October 2006)

RIR	IPv6 ADDRESS
AFRINIC	2C00:0000::/12
APNIC	2400:0000::/12
ARIN	2600:0000::/12
LACNIC	2800:0000::/12
RIPE NCC	2A00:0000::/12

Fonte: ipv6.br

Endereço IPv6 – alocação e designação

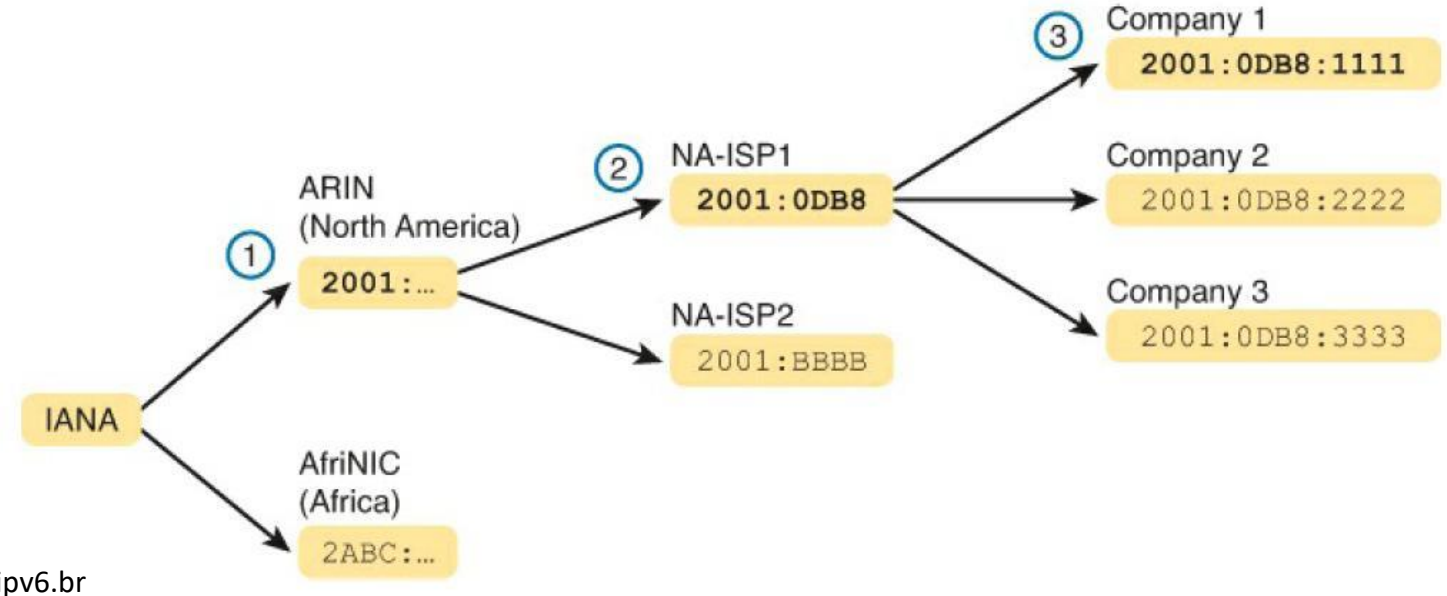


Fonte: blog.cloudfare.com e arin.net

Endereço IPv6 – alocação e designação

A alocação mínima para ISPs é um bloco /32, no entanto, alocações maiores podem ser feitas mediante apresentação de justificativa de utilização.

Um aspecto importante que merece destaque é que diferente do IPv4, com IPv6 a utilização é medida em relação ao número de designações de blocos de endereços para usuários finais, e não em relação ao número de endereços designados aos usuários finais.



Fonte: ipv6.br

Endereço IPv6 – recomendação do NIC.br

O NIC.br recomenda utilizar:

- **/64 a /56 para usuários domésticos:** Para usuários móveis pode-se utilizar /64, pois normalmente apenas uma rede é suficiente. Para usuários residenciais recomenda-se redes maiores. Se o provedor optar por, num primeiro momento, oferecer apenas /64 para usuários residenciais, ainda assim recomenda-se que no plano de numeração se reserve um /56.
- **/48 para usuários corporativos:** Empresas muito grandes podem receber mais de um bloco /48. Para planejar a rede é preciso considerar que para cada rede física ou VLAN com IPv6 é preciso reservar um /64. Esse é o tamanho padrão e algumas funcionalidades, como a autoconfiguração dependem dele. É preciso considerar também a necessidade de expansão futura, assim como a necessidade de agregação nos protocolos de roteamento.

Fonte: ipv6.br

Para saber mais...

… acesse o material online sobre Camada de Rede, do Prof. Dr. Romildo Martins da Silva Bezerra, do Instituto Federal de Educação, Ciência e Tecnologia da Bahia, Brasil

… acesse o material online sobre o Protocolo TCP/IP, do Prof. Dr. Nilton Alves Jr., do Centro de Pesquisas Físicas, Brasil.

… acesse o material online sobre TCP, UDP e Portas de Comunicação, de Júlio Battisti.

... acesse os diversos materiais disponíveis em www.ipv6.br.

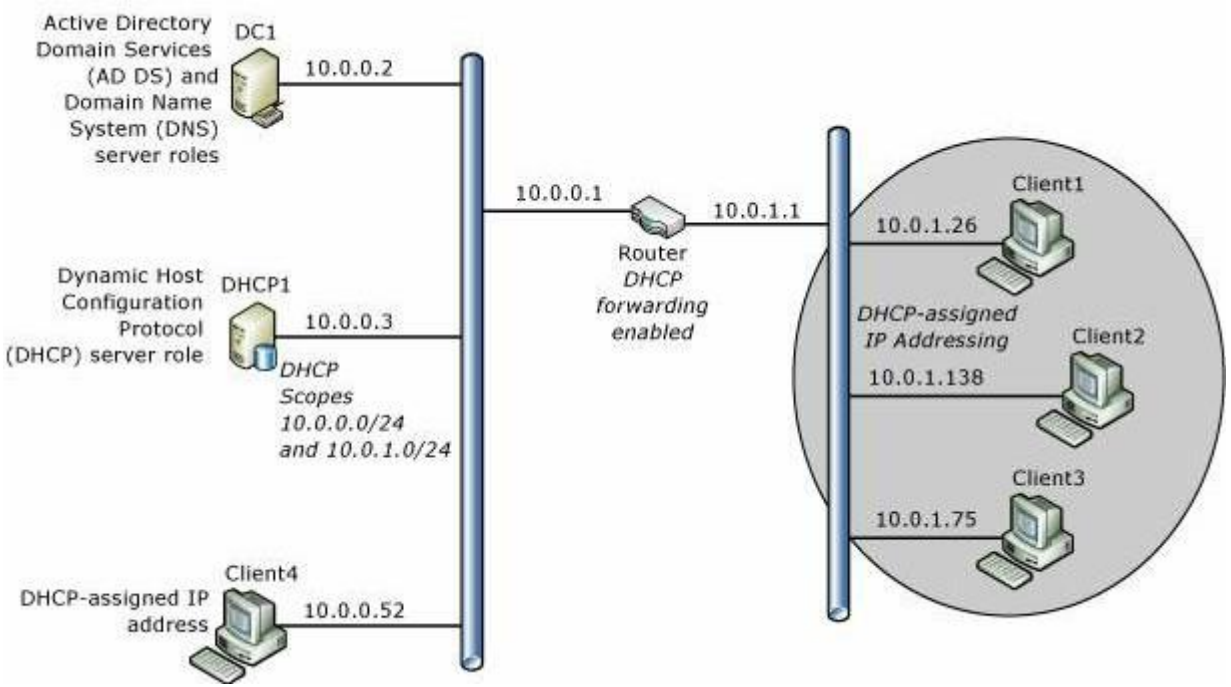
Módulo 3

Dynamic Host Configuration Protocol

Protocolo DHCP

O protocolo DHCP é usado para atribuir endereços IP e outras informações de conectividade de forma automática para os clientes de uma rede.

O DHCP é sucessor do protocolo BOOTP.



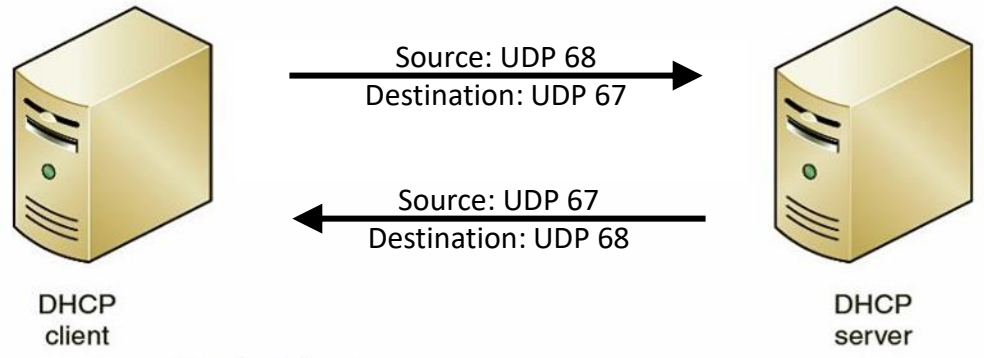
Requisitos para o servidor DHCP

O servidor DHCP, assim como qualquer outro servidor da rede, sempre deverá ter um IP fixo.

Para que os clientes possam obter configurações do servidor DHCP, é necessário que neste seja configurado o Escopo, que nada mais é que faixas de endereços IP´s previamente planejadas que serão distribuídos aos clientes da rede.

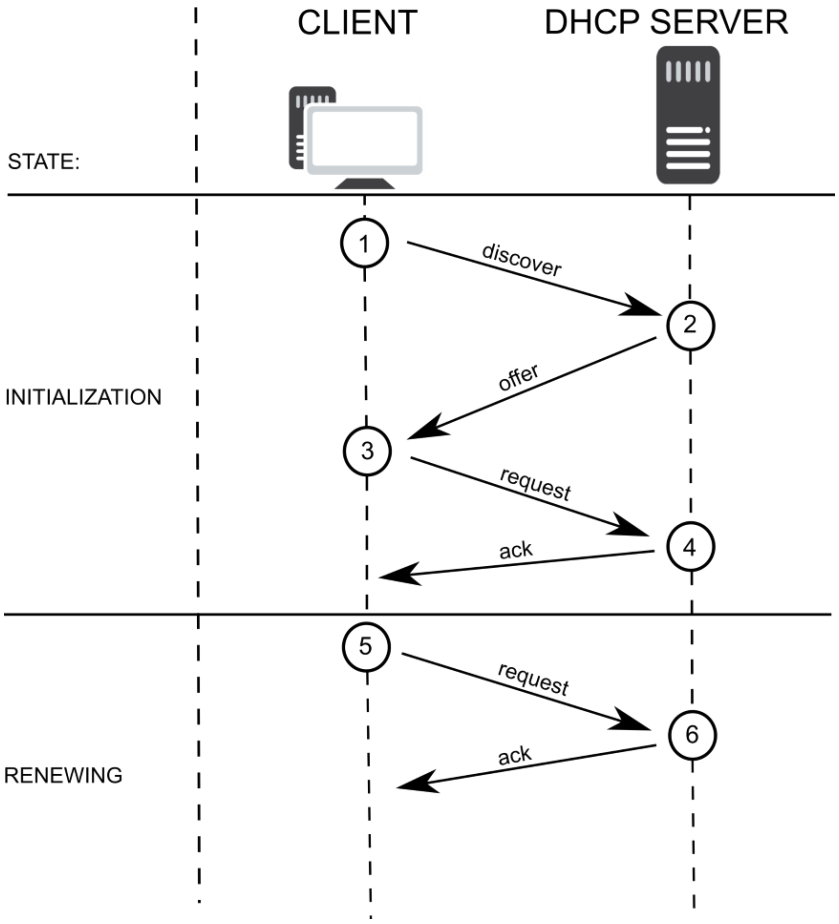
Dentro de cada escopo, além da faixa de endereços IP, pode-se configurar também as exclusões, as reservas e as opções de escopo, como por exemplo o endereço do *default gateway* e dos servidores de nome (DNS).

O cliente DHCP usa a porta UDP 68 e o servidor DHCP a porta UDP 67.



Passos na comunicação DHCP

1. O cliente envia para a rede uma mensagem DHCPDISCOVER por meio de difusão (broadcast) para descobrir possíveis servidores DHCP na rede;
2. O servidor DHCP da rede responde com a mensagem DHCPOFFER direcionada (unicast);
3. O cliente envia para a rede uma mensagem (broadcast) DHCPREQUEST;
4. O servidor DHCP responde com uma mensagem DHCPACK (unicast), que contém as configurações de IP, máscara, default gateway, etc. O cliente, ao receber esta mensagem, usa os parâmetros contidos nela para configurar a conexão de rede, mas se o cliente receber uma mensagem DHCPNAK, todo o processo é reiniciado;
5. Caso o cliente necessite renovar suas configurações de IP, ele deve enviar novamente uma mensagem DHCPREQUEST ao servidor DHCP solicitando a extensão do tempo de utilização (lease);
6. O servidor reponde com uma mensagem DHCPACK estendendo o tempo de utilização das configurações (lease);
7. Caso o cliente não necessite mais das configurações, ele envia uma mensagem DHCPRELEASE para o servidor DHCP.

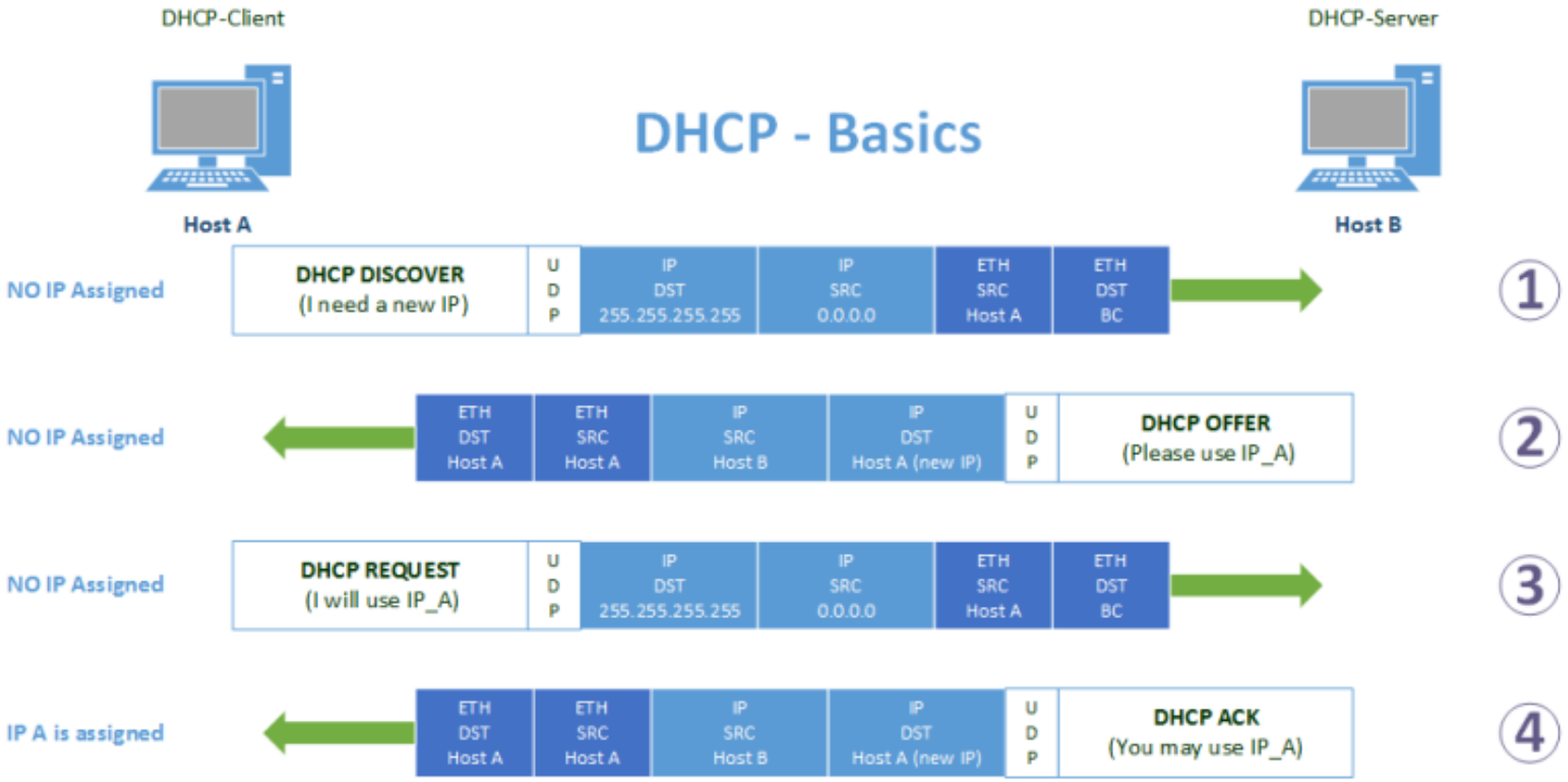


DHCP – lista de mensagens

VALOR	MENSAGEM	REFERÊNCIA
1	DHCPDISCOVER	RFC 2132
2	DHCPOFFER	RFC 2132
3	DHCPREQUEST	RFC 2132
4	DHCPDECLINE	RFC 2132
5	DHCPACK	RFC 2132
6	DHCPNAK	RFC 2132
7	DHCPRELEASE	RFC 2132
8	DHCPINFORM	RFC 2132
9	DHCPFORCERENEW	RFC 3203
10	DHCPLEASEQUERY	RFC 4388
11	DHCPLEASEUNASSIGNED	RFC 4388
12	DHCPLEASEUNKNOWN	RFC 4388
13	DHCPLEASEACTIVE	RFC 4388
14	DHCPBULKLEASEQUERY	RFC 6926
15	DHCPLEASEQUERYDONE	RFC 6926
16	DHCPACTIVELEASEQUERY	RFC 7724
17	DHCPLEASEQUERYSTATUS	RFC 7724
18	DHCPTLS	RFC 7724

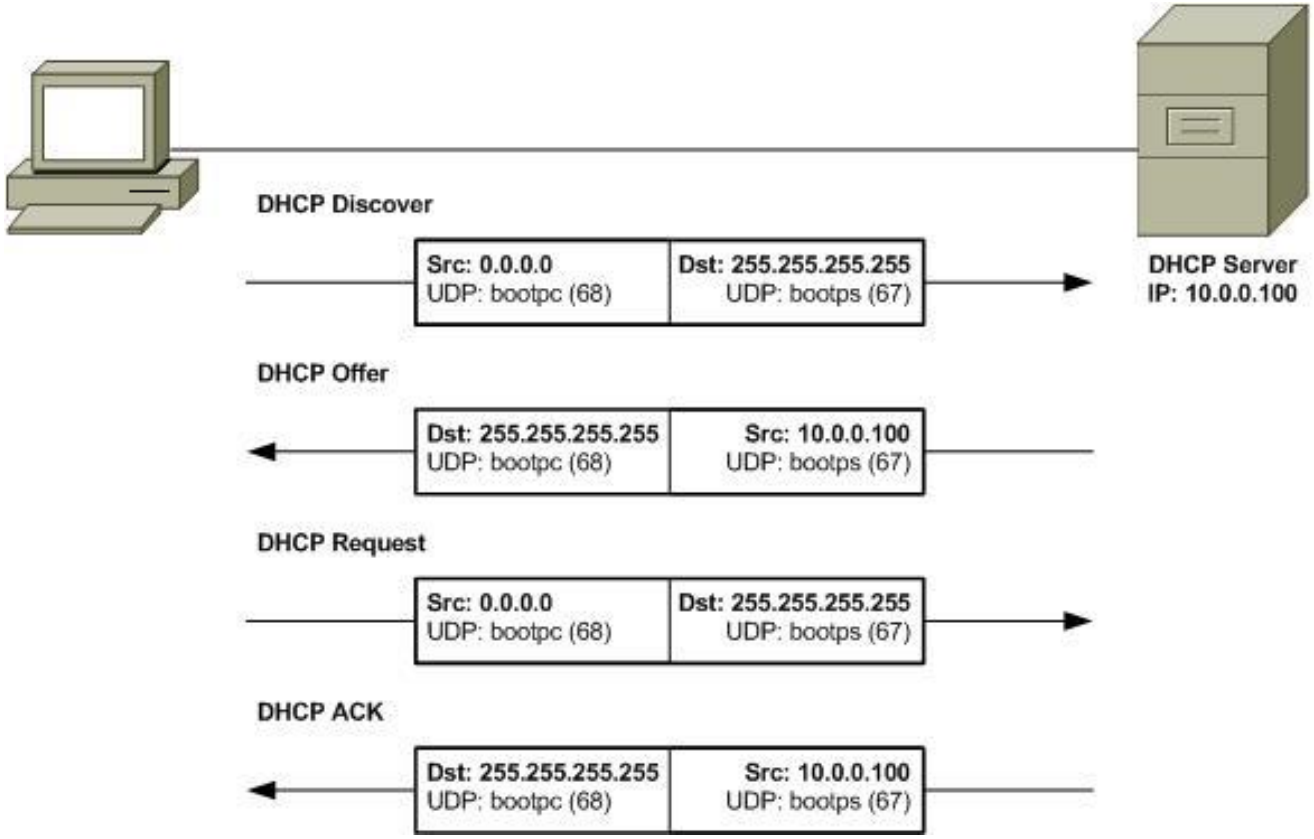
Fonte: www.iana.org/assignments/bootp-dhcp-parameters/bootp-dhcp-parameters.xhtml

Passos na comunicação DHCP – resumo I



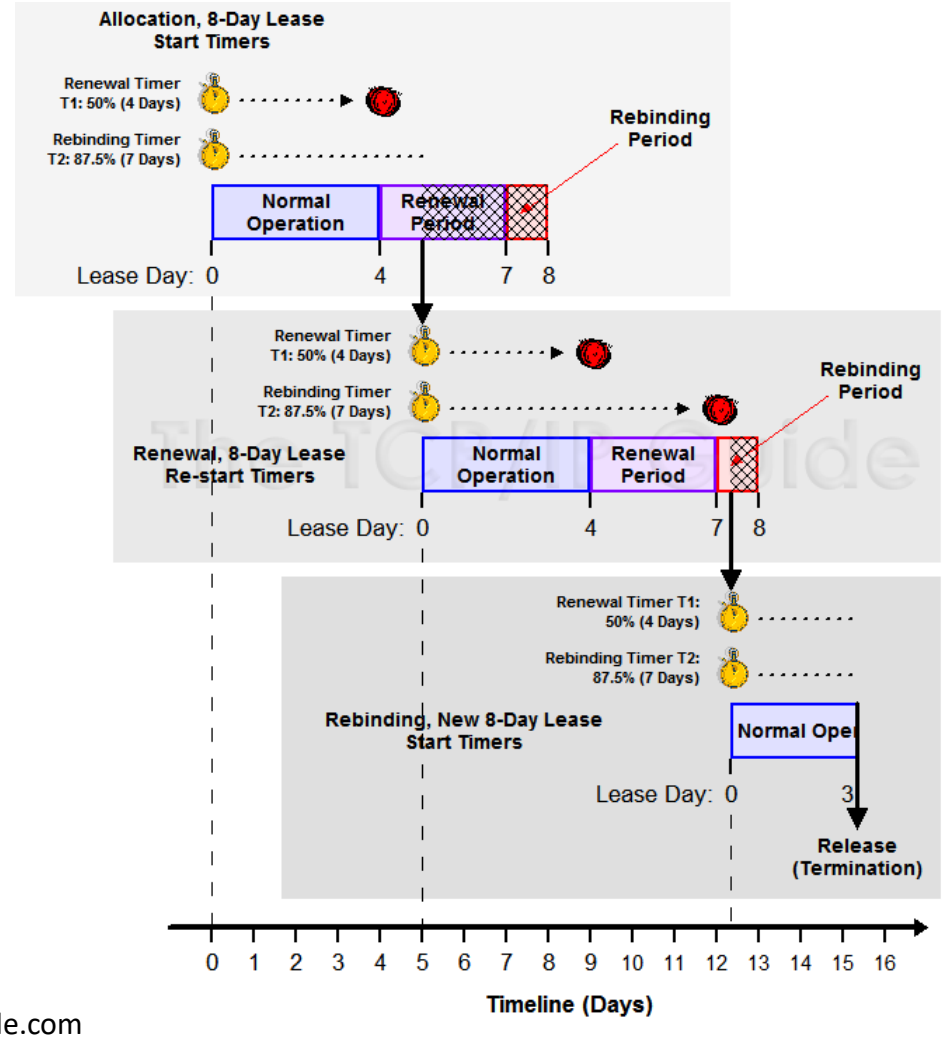
Fonte: crnetpackets.com

Passos na comunicação DHCP – resumo II



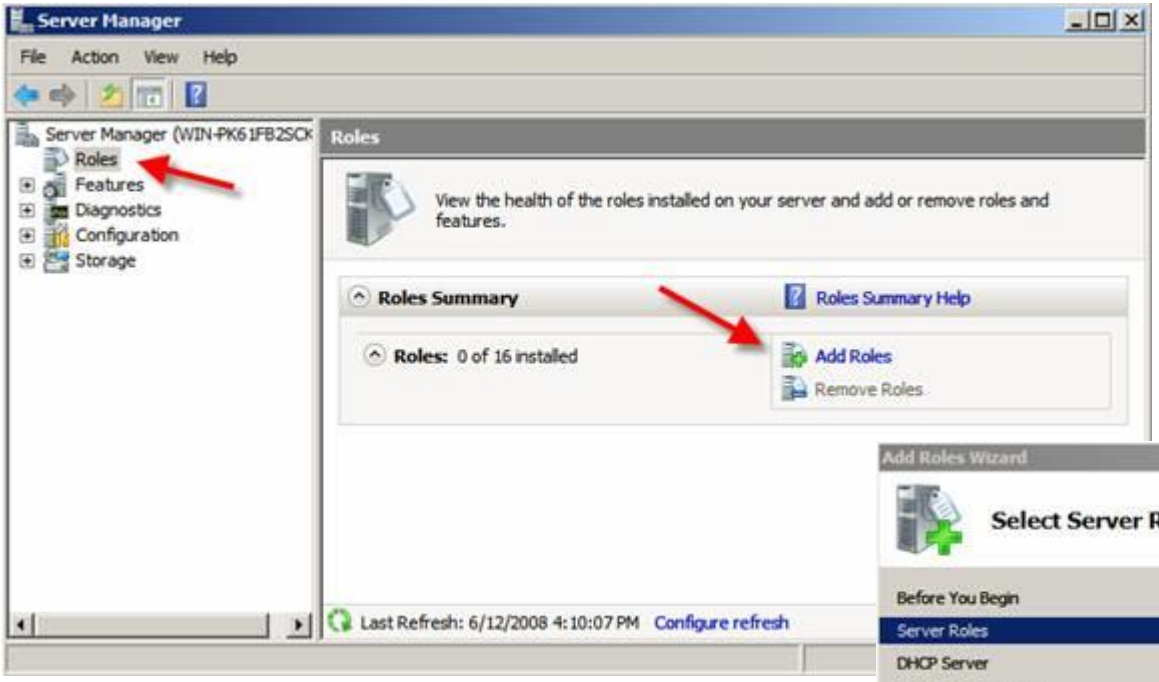
Fonte: kikobeats.github.io/server-sandbox/

Ciclo de vida da alocação do DHCP



Fonte: www.tcpipguide.com

Serviço DHCP – Windows

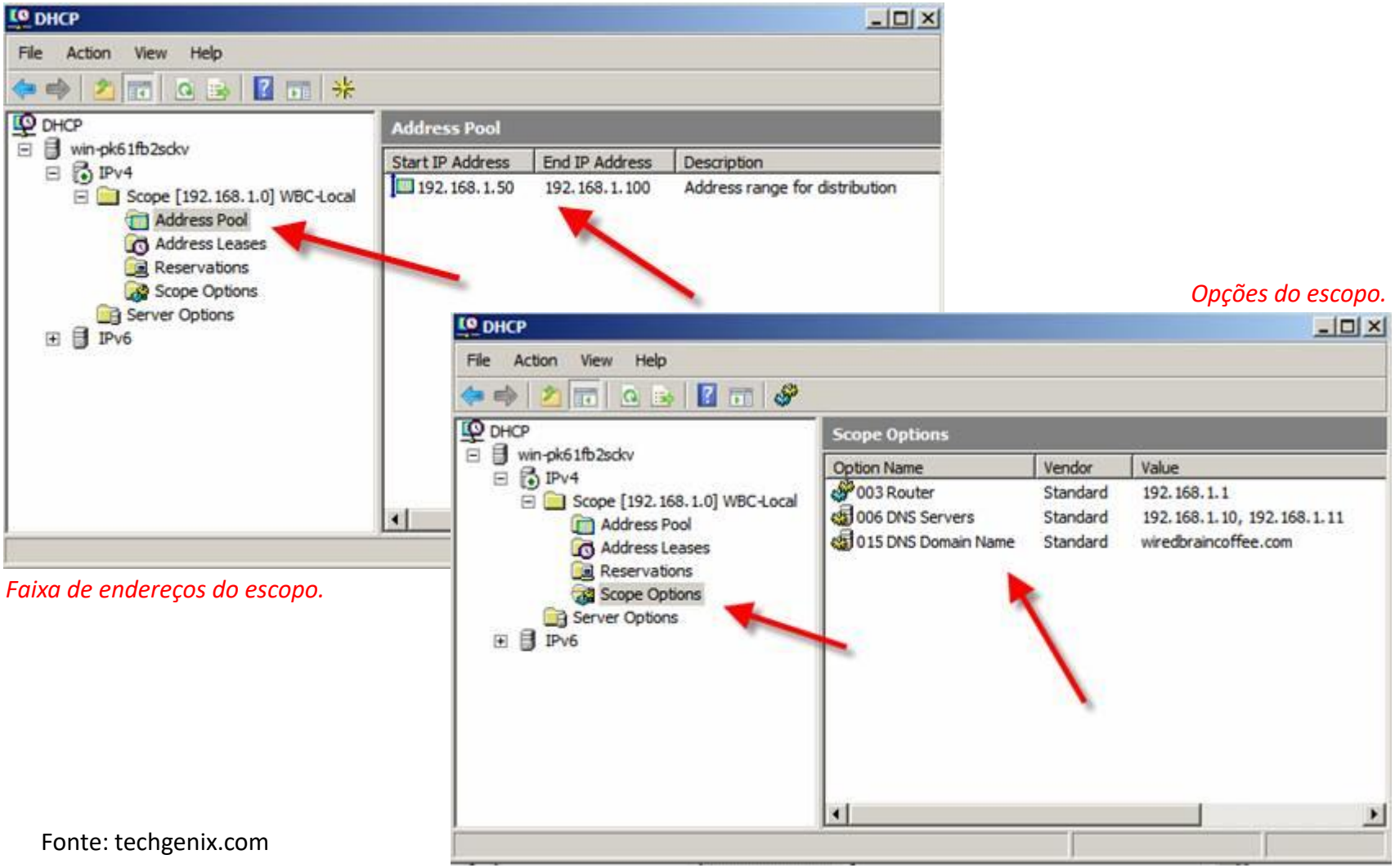


Adicionar um papel (role) de servidor.

Selecionando o papel (role) DHCP.

Fonte: techgenix.com

Serviço DHCP – Windows

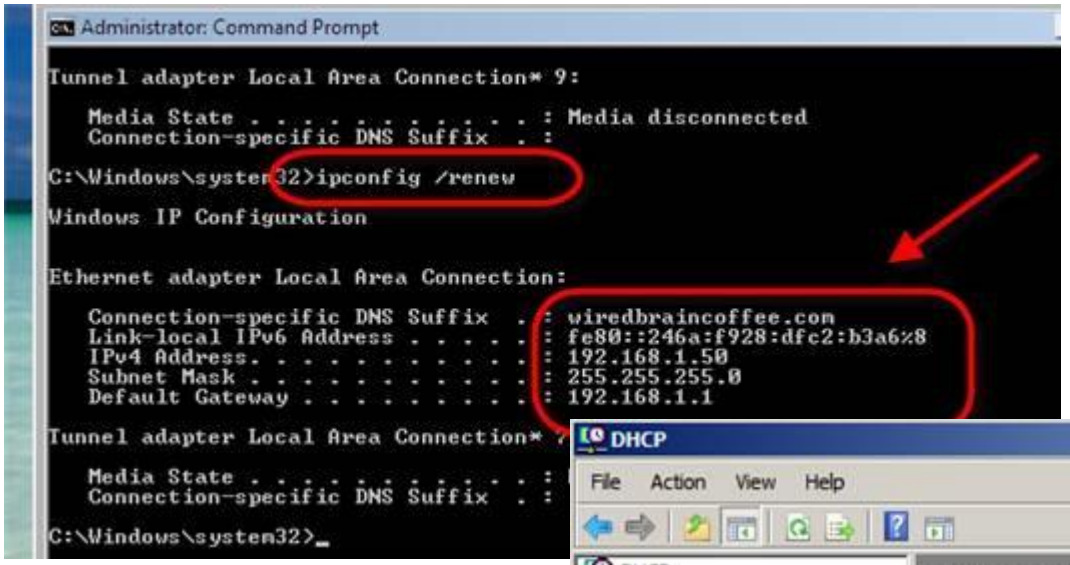


Faixa de endereços do escopo.

Opções do escopo.

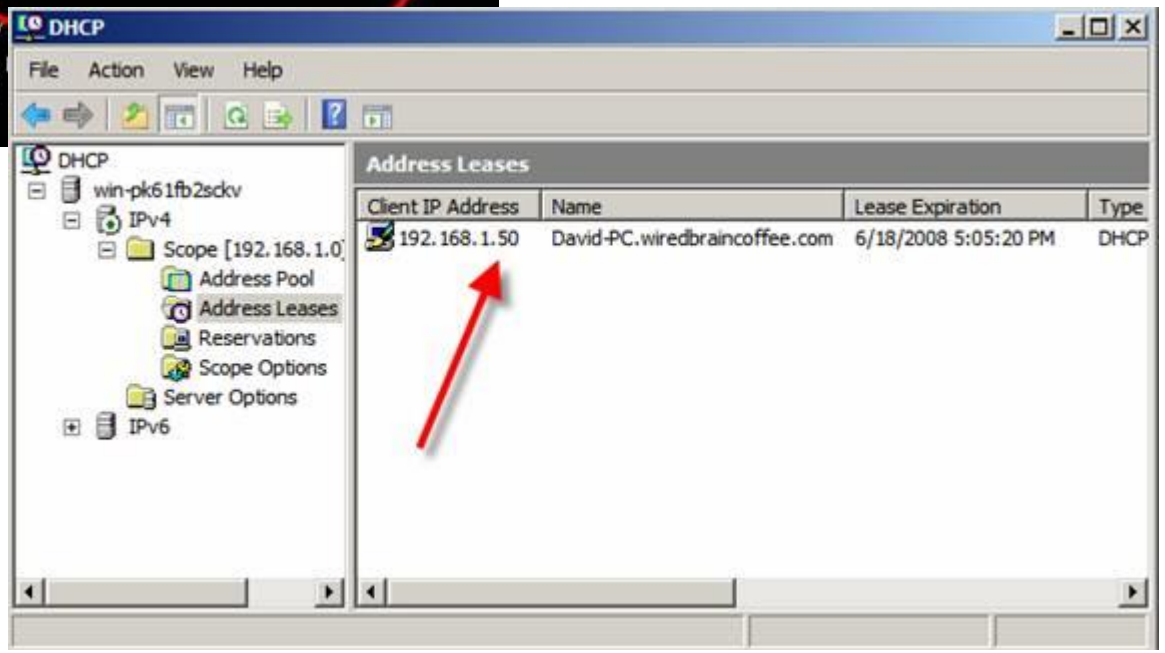
Fonte: techgenix.com

Serviço DHCP – Windows



Cliente solicita endereço IP.

Lista de endereços IP em uso.



Fonte: techgenix.com

Serviço DHCP – Linux

- Instalar o pacote DHCP*:

```
# yum install dhcp
```

- Editar o arquivo:

```
vi /etc/dhcp/dhcpd.conf
```

- Incluir as seguintes linhas referentes aos parâmetros globais:

```
option domain-name “acme.com”;
option domain-name-servers 192.168.0.10;
default-lease-time 3600;
max-lease-time 7200;
authoritative;
```

*Válido para RedHat e CentOS.

Serviço DHCP – Linux

- Incluir as seguintes linhas referentes aos parâmetros de uma subrede específica:

```
subnet 192.168.0.0 netmask 255.255.255.0 {
     option routers                  192.168.0.1;
     option subnet-mask              255.255.255.0;
     option domain-search            "acme.com";
     option domain-name-servers      192.168.0.10;
     range 192.168.0.100 192.168.0.150;
}
```

DHCP – lista de opções*

#	NOME	TAMANHO	DESCRIÇÃO	REFERÊNCIA
0	Pad	0	None	RFC 2132
1	Subnet Mask	4	Subnet Mask Value	RFC 2132
2	Time Offset	4	Time Offset in Seconds from UTC	RFC 2132
3	**Router**	**N**	**N/4 Router addresses**	**RFC 2132**
4	Time Server	N	N/4 Timeserver addresses	RFC 2132
5	Name Server	N	N/4 IEN-116 Server addresses	RFC 2132
6	**Domain Server**	**N**	**N/4 DNS Server addresses**	**RFC 2132**
7	Log Server	N	N/4 Logging Server addresses	RFC 2132
8	Quotes Server	N	N/4 Quotes Server addresses	RFC 2132
9	LPR Server	N	N/4 Printer Server addresses	RFC 2132
10	Impress Server	N	N/4 Impress Server addresses	RFC 2132
11	RLP Server	N	N/4 RLP Server addresses	RFC 2132
12	Hostname	N	Hostname string	RFC 2132
13	Boot File Size	2	Size of boot file in 512 byte chunks	RFC 2132
14	Merit Dump File	N	Client to dump and name the file to dump it to	RFC 2132
15	**Domain Name**	**N**	**The DNS domain name of the client**	**RFC 2132**
...	...	...	...	...

*Lista completa disponível em: www.iana.org/assignments/bootp-dhcp-parameters/bootp-dhcp-parameters.xhtml

APIPA

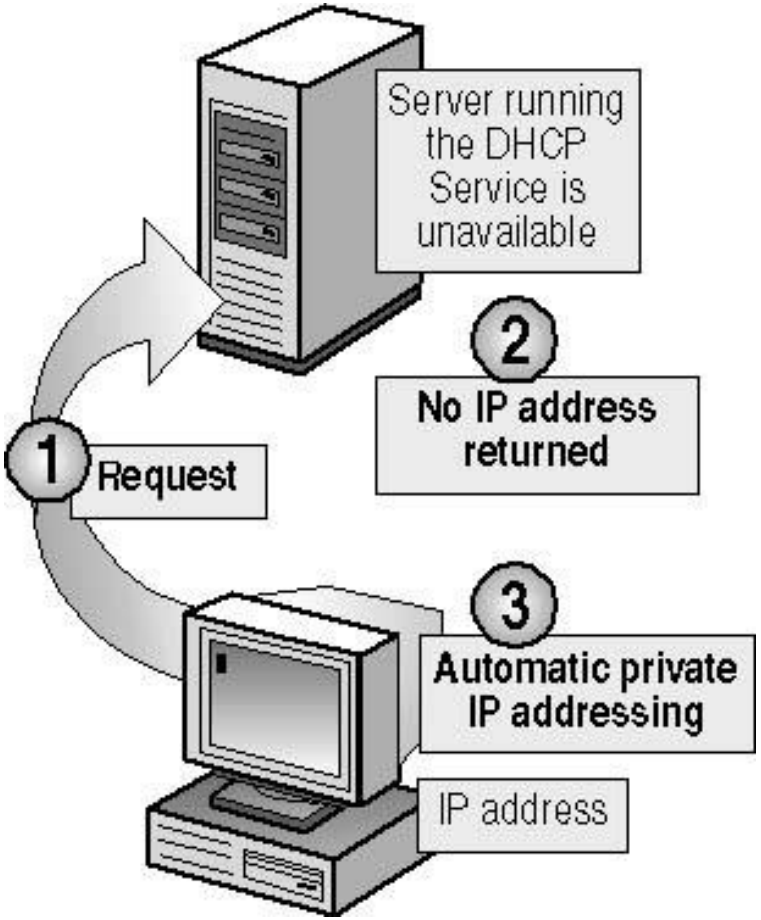
Automatic Private IP Addressing (Endereçamento Automático de IP Privado) é um recurso do Windows que permite a um computador atribuir um endereço IP automaticamente quando um servidor DHCP não estiver disponível na rede.

Se nenhum servidor DHCP estiver disponível no momento, o computador selecionará um endereço IP privado da faixa 169.254.0.0-169.254.255.255, que foi reservado pela IANA para este fim.

O cliente usa o protocolo de resolução de endereços (ARP) para garantir que o endereço escolhido não esteja sendo usado por outro computador da rede.

O equivalente em ambientes GNU/Linux é o Zeroconf (Zero Configuration Networking).

Fonte: whatis.techtarget.com



Para saber mais...

… acesse o material online sobre Dynamic Host Configuration Protocol, de Júlio Battisti.

… veja a animação online do funcionamento do protocolo DHCP, da RAD University.

Módulo 4

Domain Name System

DNS

O Domain Name System é um banco de dados hierárquico que oferece o serviço de resolução de nomes URL (Uniform Resource Locator) usados para identificar um domínio.

Toda comunicação na Internet é feita por meio dos endereços IP, mas é muito mais fácil memorizar URL´s do que endereços IP.

Assim, o que o serviço de DNS faz é converter as URL´s em endereços IP:

www.brasil.gov.br → 170.246.252.242

www.tj.sp.gov.br → 200.142.86.230

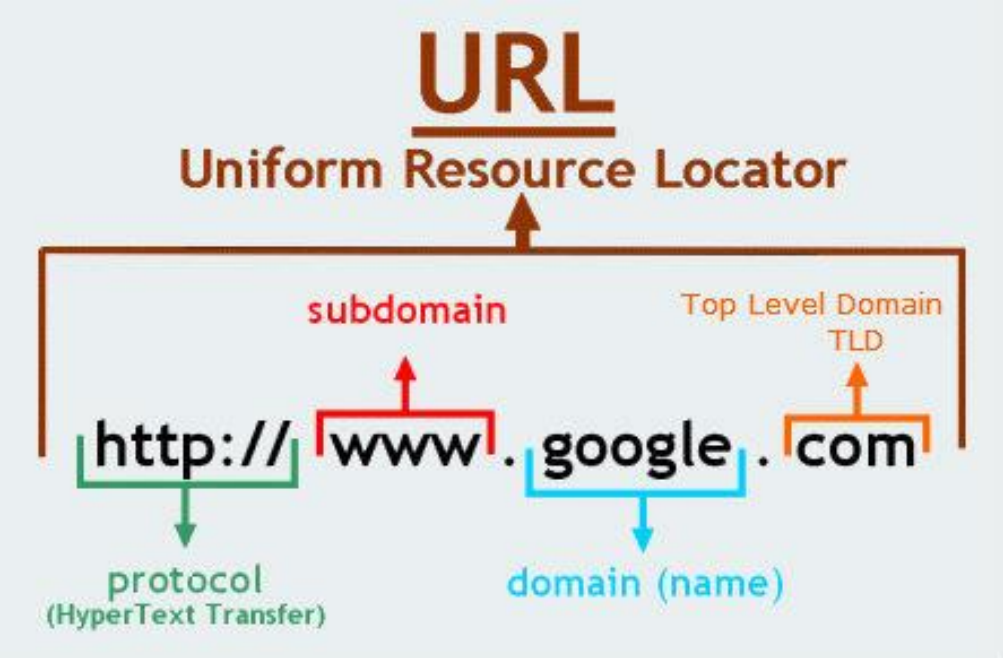
www.google.com → 172.217.172.132

www.bb.com.br → 170.66.11.10

URL – Uniform Resource Locator

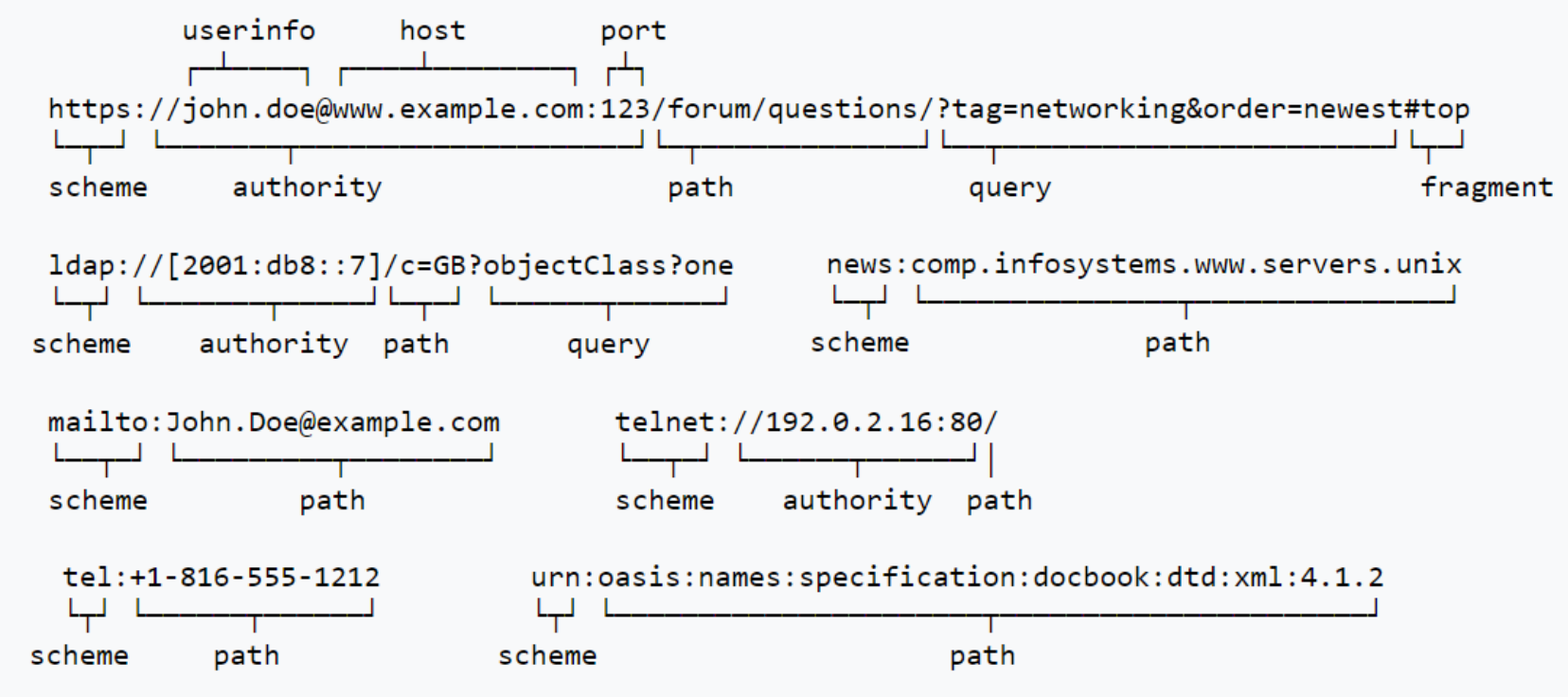
Um URL (Uniform Resource Locator), é uma referência a um recurso da web que especifica sua localização em uma rede de computadores e um mecanismo para recuperá-lo.

Um URL é um tipo específico de URI (Uniform Resource Identifier).



URI – Uniform Resource Identifier

Um URI (Uniform Resource Identifier) é uma cadeia de caracteres que identifica inequivocamente um recurso específico.



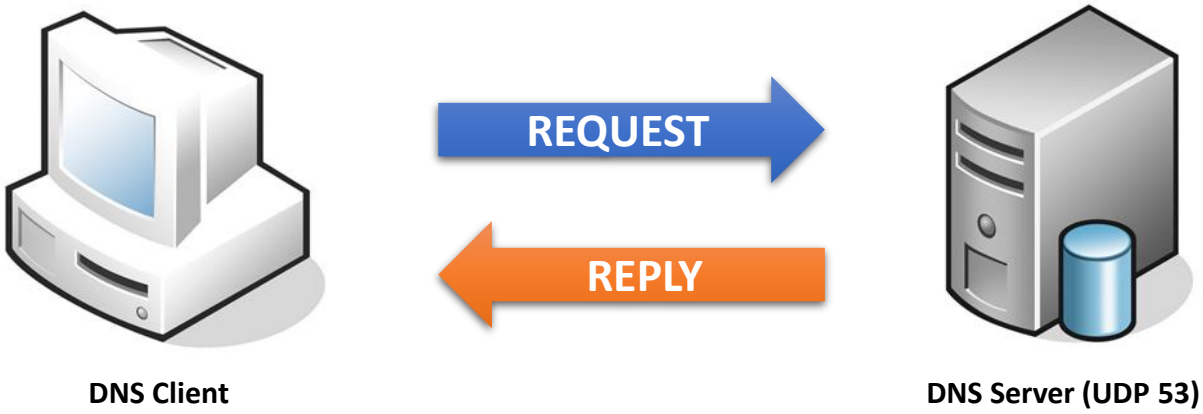
Fonte: wikipedia.org

DNS

Um cliente DNS é todo aquele que requisita respostas a uma determina consulta feita a um servidor DNS.

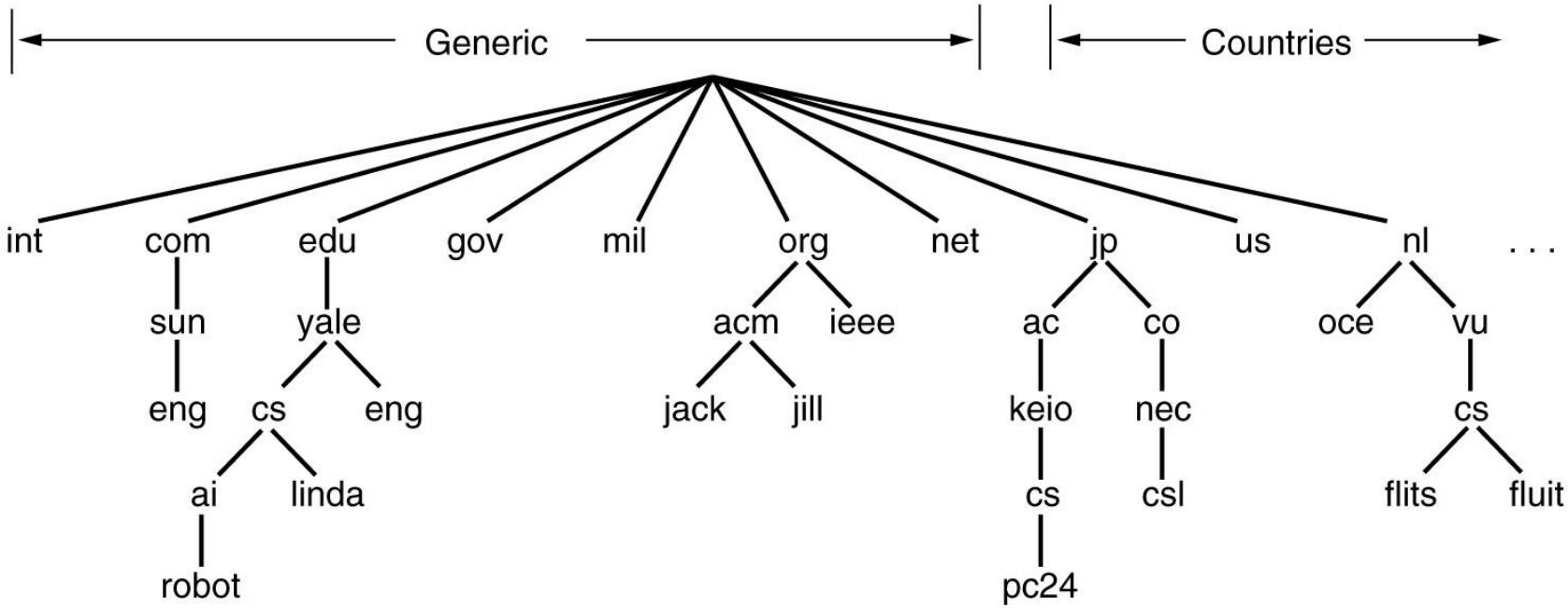
Um servidor DNS é todo aquele que responde às consultas feitas por um cliente.

O servidor DNS opera na porta UDP 53.



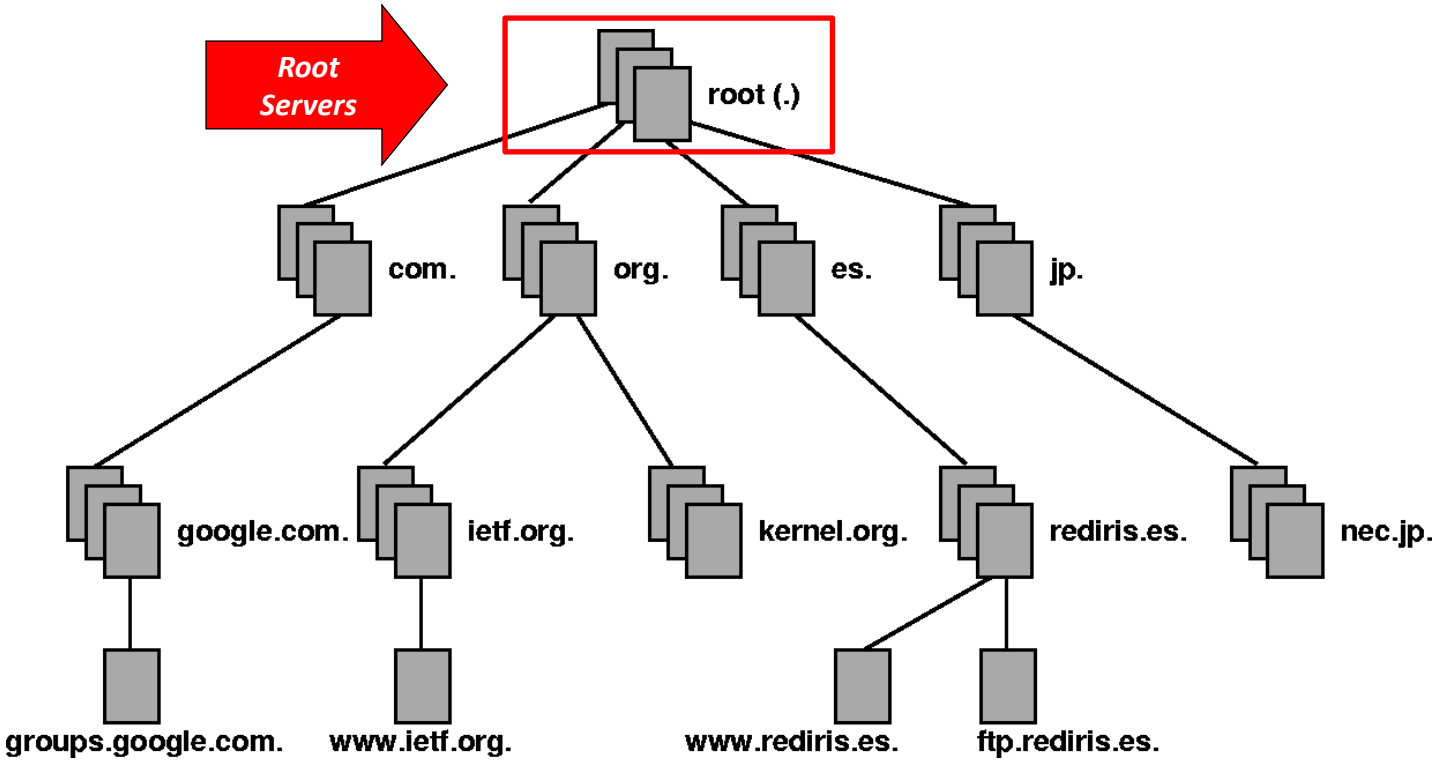
DNS

Os nomes de domínio servem para identificar uma rede ou grupo de computadores. Estão dispostos de forma hierárquica e geralmente possuem um ou mais servidores DNS responsáveis por mapear todos os nomes abaixo daquele domínio (ou subdomínio) em endereços IP.



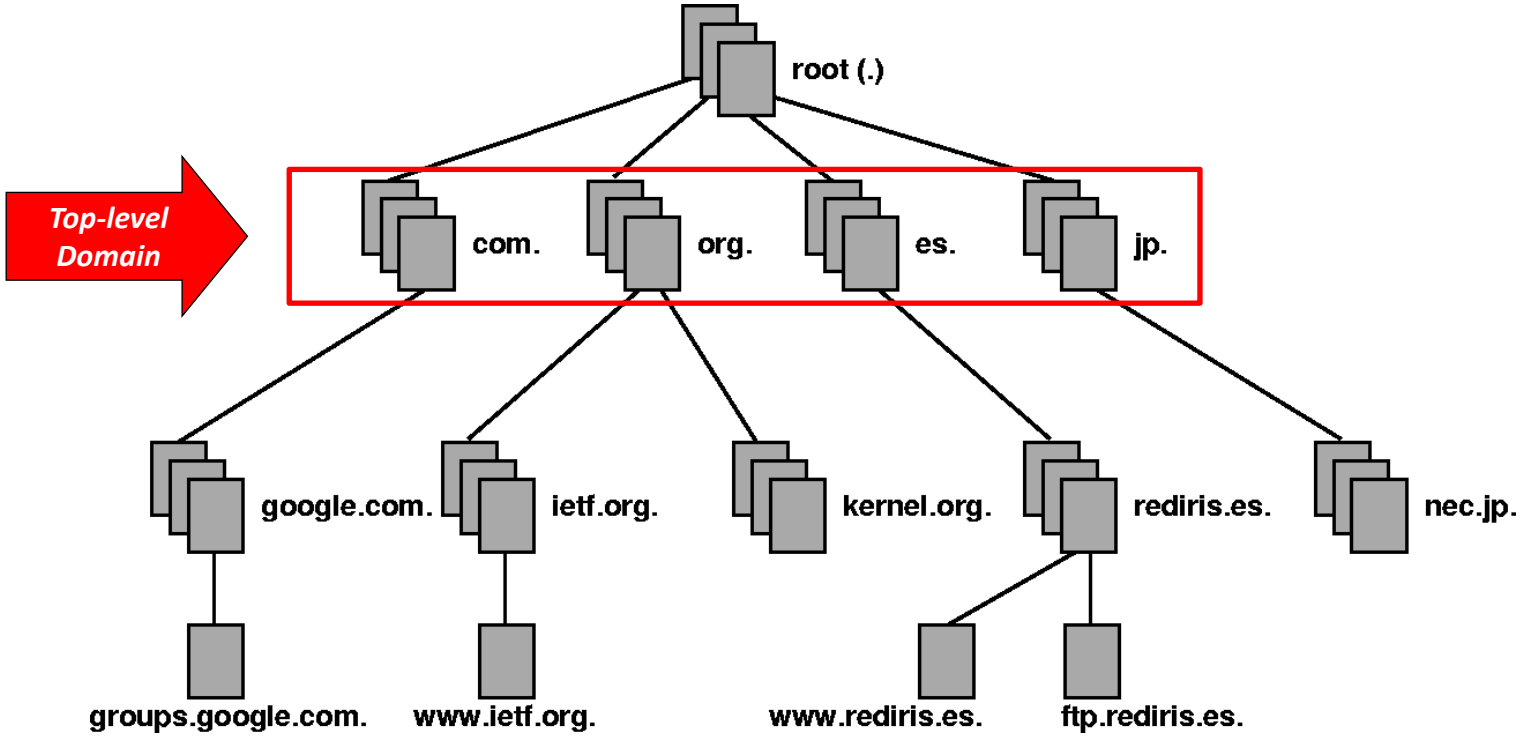
DNS

O ponto mais alto da cadeia é denominado *root*. O servidor DNS responsável por este ponto é o *root server*. Este servidor possui todas as entradas para os servidores imediatamente abaixo dele.



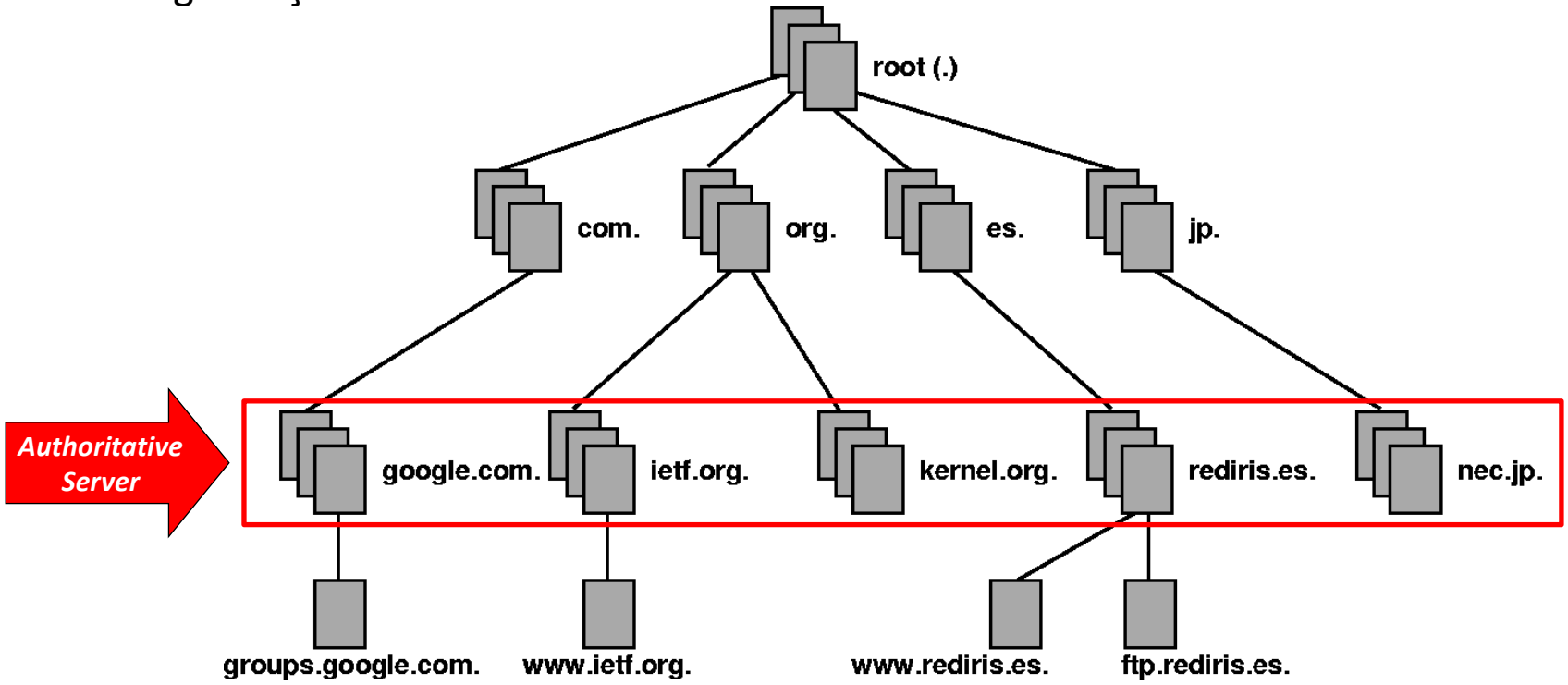
DNS

Os *Top-level Domain* identificam domínios genéricos, como .com ou .gov, e domínios de países, como .br, .jp, .it, etc. O servidor DNS responsável por este ponto é o *TLD server*. Este servidor possui todas as entradas para os servidores imediatamente abaixo dele.

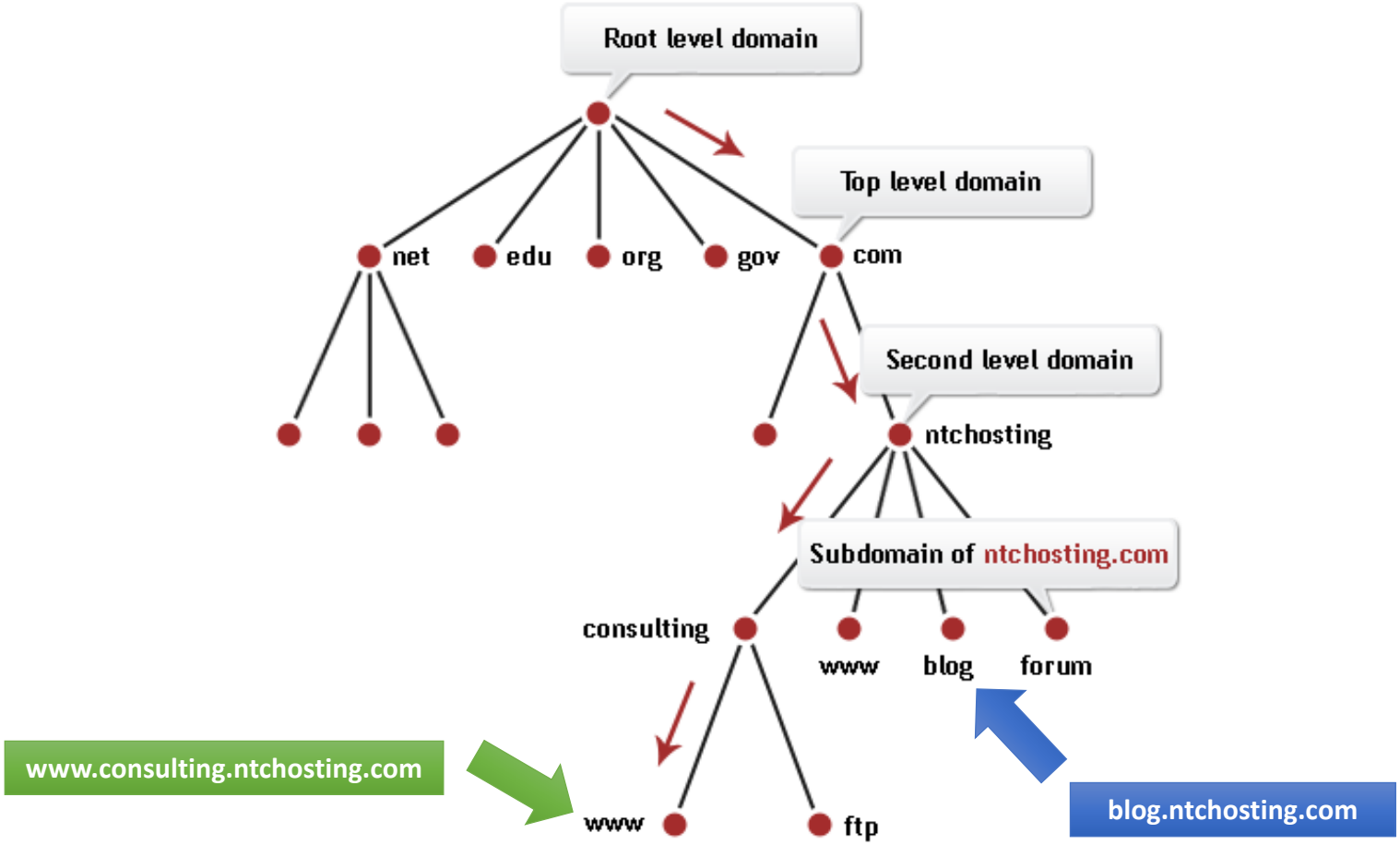


DNS

Os servidores autoritativos são responsáveis pelas empresas ou organizações que representam. O servidor DNS responsável por este ponto é o *authoritative server*. Este servidor possui todas as entradas para os servidores e demais *hosts* dentro da organização.



DNS – exemplo


Root level domain
Top level domain
net
edu
org
gov
com
Second level domain
ntchosting
Subdomain of ntchosting.com
consulting
www
blog
forum
www
ftp
www.consulting.ntchosting.com
blog.ntchosting.com

DNS – Root servers

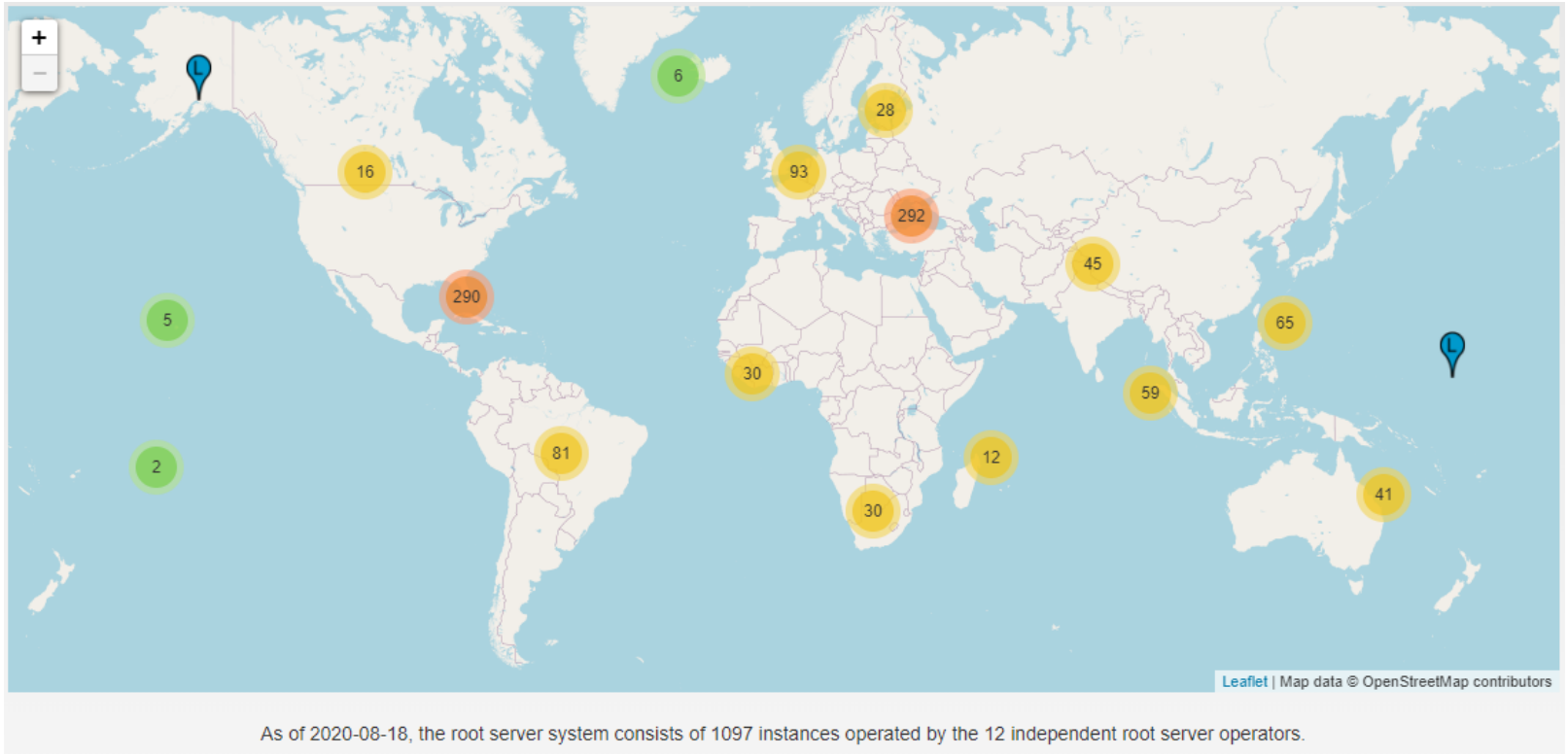
Os *root servers* são servidores DNS que possuem informações sobre os servidores *top-level domain* e são os primeiros a serem consultados. Ao todo são treze.

NOME	IP	OPERADOR
a.root-servers.net	198.41.0.4	Verisign, Inc.
b.root-servers.net	199.9.14.201	USC-ISI
c.root-servers.net	192.33.4.12	Cogent Communications
d.root-servers.net	199.7.91.13	University of Maryland
e.root-servers.net	192.203.230.10	NASA
f.root-servers.net	192.5.5.241	Internet Systems Consortium, Inc.
g.root-servers.net	192.112.36.4	US Department of Defense
h.root-servers.net	198.97.190.53	US Army Research Lab
i.root-servers.net	192.36.148.17	Netnod
j.root-servers.net	192.58.128.30	Verisign, Inc.
k.root-servers.net	193.0.14.129	RIPE NCC
l.root-servers.net	199.7.83.42	ICANN
m.root-servers.net	202.12.27.33	WIDE Project

Fonte: www.iana.org/domains/root/servers

DNS – Root servers

Os *root servers* são formados por 1097 instâncias mantidas por doze operadores diferentes, distribuídas geograficamente conforme o mapa.

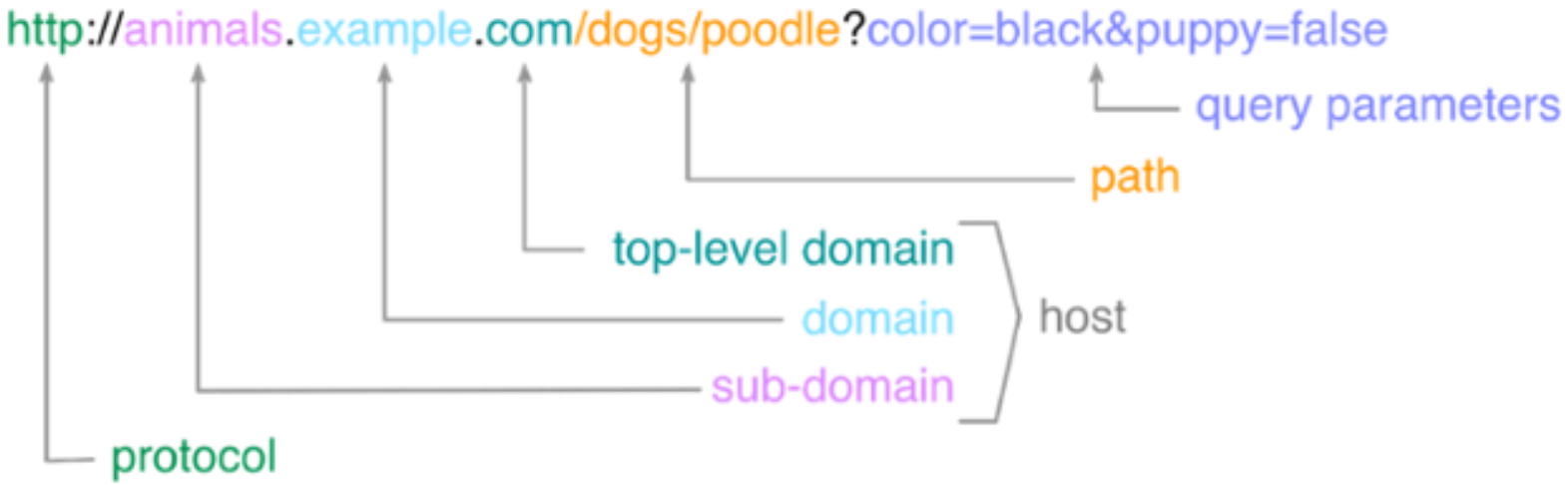


Fonte: root-servers.org

DNS – Top-level domain

Um Top-level domain (TLD) é um dos domínios no nível mais alto da hierarquia de nomes de domínio da Internet. Os top-level domain são instalados a partir dos root server.

Para todos os domínios em níveis inferiores, é a última parte do nome do domínio, ou seja, o último rótulo de um nome de domínio totalmente qualificado, também conhecido como FQDN (Fully Qualified Domain Name).



Fonte: wikipedia.org

DNS – Authoritative server

Um Authoritative server é um servidor que fornece respostas a perguntas sobre nomes em uma zona. Um authoritative server somente retorna respostas para consultas sobre nomes de domínio que foram especificamente configurados pelo administrador. Os servidores de nomes também podem ser configurados para fornecer respostas autoritativas a consultas em algumas zonas, enquanto atuam como um servidor de nomes em cache para todas as outras zonas.

Um authoritative server pode ser do tipo principal (master) ou um servidor secundário (slave). Um servidor primário para uma zona é o servidor que armazena as versões definitivas de todos os registros nessa zona.

Um authoritative server master é identificado pelo registro SOA (Start of Authority). Um servidor secundário para uma zona usa um mecanismo de atualização automática para manter uma cópia idêntica do banco de dados do servidor primário para uma zona.

Fonte: wikipedia.org

DNS – Authoritative server
Start of authority

Um registro SOA (Start of Authority) é um tipo de registro de recurso no Sistema de Nomes de Domínio (DNS) que contém informações administrativas sobre a zona, especialmente no que se refere a transferências de zona.

O formato de registro SOA é especificado no RFC 1035.

Fonte: wikipedia.org

DNS – Authoritative server
Start of authority

IN: Tipo de zona a que se refere o SOA. Geralmente usa-se IN para Internet;

SOA: Abreviação de Start of Authority;

MNAME: Nome do servidor principal da zona. Neste exemplo, master.example.com;

RNAME: E-mail do administrador responsável pela zona. Neste exemplo, hostmaster@example.com. Note que o símbolo “@” foi trocado por “.”;

SERIAL: Número de série para esta zona. Se um servidor de nomes secundário (slave) perceber que este número aumentou, ele irá assumir que os dados foram atualizados e iniciará uma transferência de zona;

Fonte: wikipedia.org

DNS – Authoritative server
Start of authority

REFRESH: Número de segundos após o qual os servidores de nomes secundários devem consultar o mestre para detectar alterações de zona;

RETRY: Número de segundos após o qual os servidores de nomes secundários devem tentar novamente contatar o servidor primário (master), caso o mesmo não responda. Este valor tem de ser menor que REFRESH;

EXPIRE: Número de segundos após o qual os servidores de nome secundários devem parar de responder à solicitação para esta zona se o mestre não responder. Este valor deve ser maior que a soma de REFRESH e RETRY;

TTL: Tempo de vida para fins de armazenamento em cache negativo. Originalmente, esse campo tinha o significado de um valor TTL mínimo para registros de recursos nessa zona; foi alterado para o seu significado atual pelo RFC 2308.

Fonte: wikipedia.org

DNS – Authoritative server
Registros de recurso

Registros de recursos DNS ou “resource records” ou simplesmente “RRs” são o conteúdo do arquivo de zona DNS. O arquivo de zona contém mapeamentos entre nomes de domínio e endereços IP na forma de registros de texto.

Existem muitos tipos de registros de recursos*. Os mais comuns são:

- SOA – start of authority;
- TXT – text;
- NS – name server;
- A – address;
- PTR – pointer;
- CNAME – canonical name;
- MX – mail exchange;
- SRV – server.

*Uma lista completa pode ser consultada em en.wikipedia.org/wiki/List_of_DNS_record_types

DNS – Authoritative server
Registros de recurso

SOA: Cada arquivo de zona terá um registro SOA e ele estará presente no início. Esse tipo de registro contém informações sobre a própria zona e sobre outros registros. Cada zona terá apenas um registro SOA.

```
IN   SOA      nameserver.place.dom.  postmaster.place.dom.
```

TXT: Este registro serve para adicionar comentários ou informações adicionais.

```
example.com   IN  TXT   "This domain name is an example"
```

Fonte: interserver.net

DNS – Authoritative server
Registros de recurso

NS: Este registro serve para mostrar quais são os servidores autoritativos da zona. Eles indicam servidores primários e secundários para a zona especificada no registro SOA. As zonas podem conter muitos registros NS, mas devem conter pelo menos um registro NS para uma zona DNS.

Por exemplo, quando o administrador do domínio abc.com delega autoridade para que noamdc1.noam.abc.com. administre o subdomínio noam.abc.com., a seguinte linha deve ser adicionada à zona abc.com e noam.abc.com:

```
noam.abc.com. IN NS noamdc1.noam.abc.com.
```

Ou seja, o servidor “noamdc1.noam.abc.com” passa a ser autoritativo para o domínio “noam.abc.com”.

Fonte: interserver.net

DNS – Authoritative server
Registros de recurso

A: Este registro mapeia um nome de domínio para um endereço IP. No exemplo abaixo, o seguinte registro de recurso, localizado na zona abc.com, mapeia o FQDN do servidor para seu endereço IP:

```
abc.com IN A 172.16.48.1
```

Para mapear o FQDN de endereços IPv6, usa-se o registro AAAA ao invés de A.

PTR: Este registro é um ponteiro que funciona como um reverso ao registro A. Ele mapeia um nome de domínio para um endereço IP de modo a se obter o DNS reverso, como no exemplo abaixo:

```
1.48.16.172.in-addr.arpa. IN PTR abc.com.
```

Fonte: interserver.net

DNS – Authoritative server
Registros de recurso

CNAME: Este registro serve para criar um alias para o nome do domínio. Um exemplo do registro CNAME é dado abaixo.

```
ftp.abc.com. IN CNAME ftp1.abc.com.
```

Depois que um cliente DNS consulta o registro de recurso para ftp.abc.com, o servidor DNS localiza o registro de recurso CNAME. Em seguida, ele resolve a consulta do registro de recurso A para ftp1.abc.com e retorna os registros de recurso A e CNAME para o cliente.

Fonte: interserver.net

DNS – Authoritative server
Registros de recurso

MX: Esse registro representa o servidor responsável por processar ou encaminhar mensagens de correio eletrônico em um domínio DNS.

Processar uma mensagem significa entregá-la ao destinatário ou passá-lo para um tipo diferente de transporte de correio. Encaminhar uma mensagem significa enviá-lo para seu servidor de destino final, ou seja, ele será o SMTP (Simple Mail Transfer Protocol) para outro servidor de troca de mensagens que esteja mais próximo do destino final ou o enfileire por um período de tempo especificado.

Somente servidores de troca de mensagens usam registros MX. O exemplo a seguir mostra registros de recursos MX para os servidores de e-mail para o domínio noam.abc.com.:

```
*. noam.abc.com. IN MX 0 mailserver1.noam.abc.com.
*. noam.abc.com. IN MX 10 mailserver2.noam.abc.com.
*. noam.abc.com. IN MX 10 mailserver3.noam.abc.com.
```

O número após IN MX indica a prioridade do servidor de mensagens.

Fonte: interserver.net

DNS – Authoritative server
Registros de recurso

SRV: Esse registro permite que sejam especificados servidores para os quais devem ser direcionados o tráfego de serviços específicos.

Abaixo segue o formato deste registro:

```
_serv._prot.example.com SRV 10 0 5060 serv.example.com
```

Onde:

- Service: o nome do serviço, que deve ser precedido de "_";
- Protocol: o nome do protocolo, que deve ser precedido de "_";
- Domain: o nome do domínio que receberá o tráfego original desse serviço;
- Priority: o primeiro número (10) indica a prioridade do servidor alvo;
- Weight: se dois registros tiverem a mesma prioridade, o peso (0) será usado;
- Port: a porta TCP ou UDP que será usada pelo serviço (5060);
- Target: o domínio ou subdomínio alvo, que deve ter um registro A ou AAAA para resolver para o endereço IP.

Fonte: interserver.net

DNS – Authoritative server
Registros de recurso

Exemplo de uso do registro SRV:

```
_ldap._tcp.example.com. IN SRV 10 50 389 ds1.example.com.
```

Onde:

- Service: LDAP;
- Protocol: TCP;
- Domain: example.com;
- Priority: 10;
- Weight: 50;
- Port: 389 (TCP);
- Target: ds1.example.com.

Fonte: interserver.net

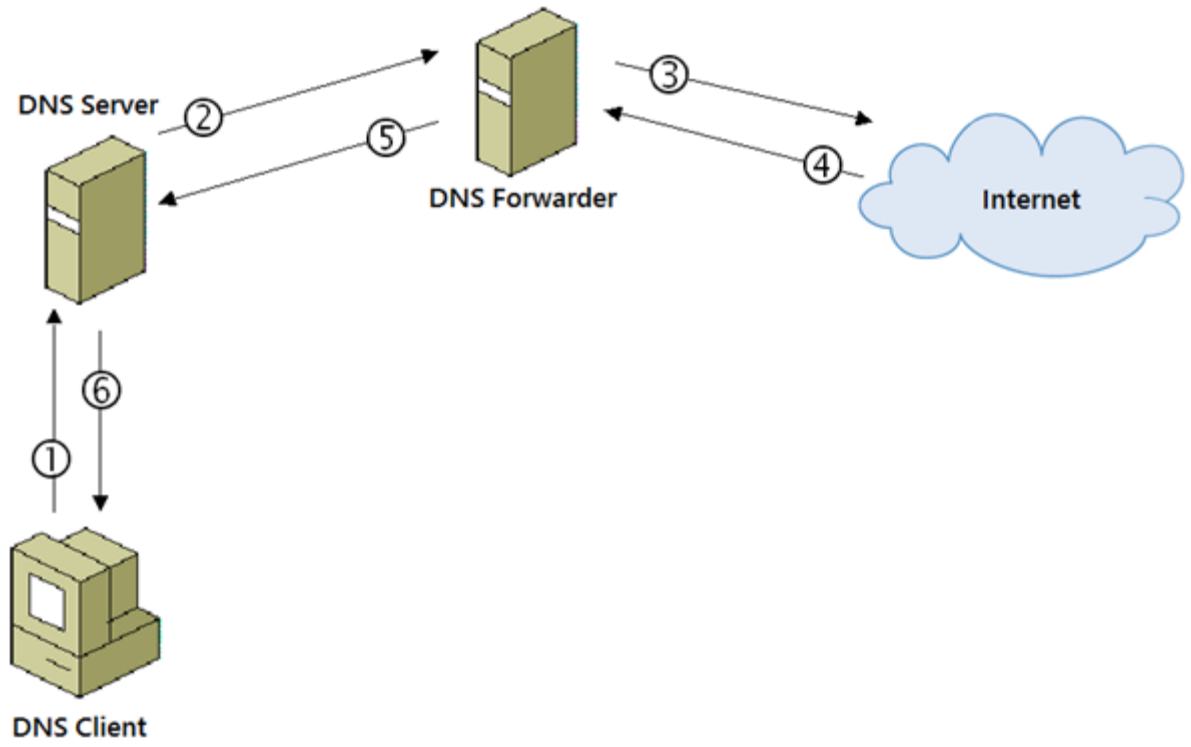
DNS – Authoritative server Exemplo

```
$ORIGIN example.com.     ; start of the zone file
$TTL 30m                 ; default cache expiration time for resource records
example.com. IN SOA ns.example.com. root.example.com. (
1999120701               ; serial number of this zone file
1d                       ; frequency to refresh secondary DNS (d=day)
1d                       ; frequency to refresh secondary DNS in case of problem
4w                       ; secondary DNS expiration time (w=week)
1h                       ; minimum caching time if resolution failed
)
example.com. NS dns1.dnsprovider.com.    ; name server
example.com. NS dns2.dnsprovider.com.    ; another name server
example.com. MX 10 mx1.dnsprovider.com   ; mail server
example.com. MX 10 mx2.dnsprovider.com   ; another mail server
example.com. A 192.168.100.1             ; IP address for root domain
www          A 192.168.100.1             ; IP address for www subdomain
```

Fonte: ns1.com

DNS - Forwarder

Um DNS forwarder ou encaminhador é um servidor DNS que encaminha consultas DNS para outros servidores e armazena localmente um cache de pesquisas já realizadas.



Arquivo hosts

O arquivo *hosts.txt* é um arquivo texto que nos primórdios da Internet era mantido manualmente e disponibilizado via compartilhamento de arquivos pelo Stanford Research Institute para a associação ARPANET, contendo os nomes de host e seus endereços.

Nos sistemas operacionais modernos, este arquivo permanece como um mecanismo de resolução de nome alternativo.

No GNU/Linux, este arquivo se encontra em:

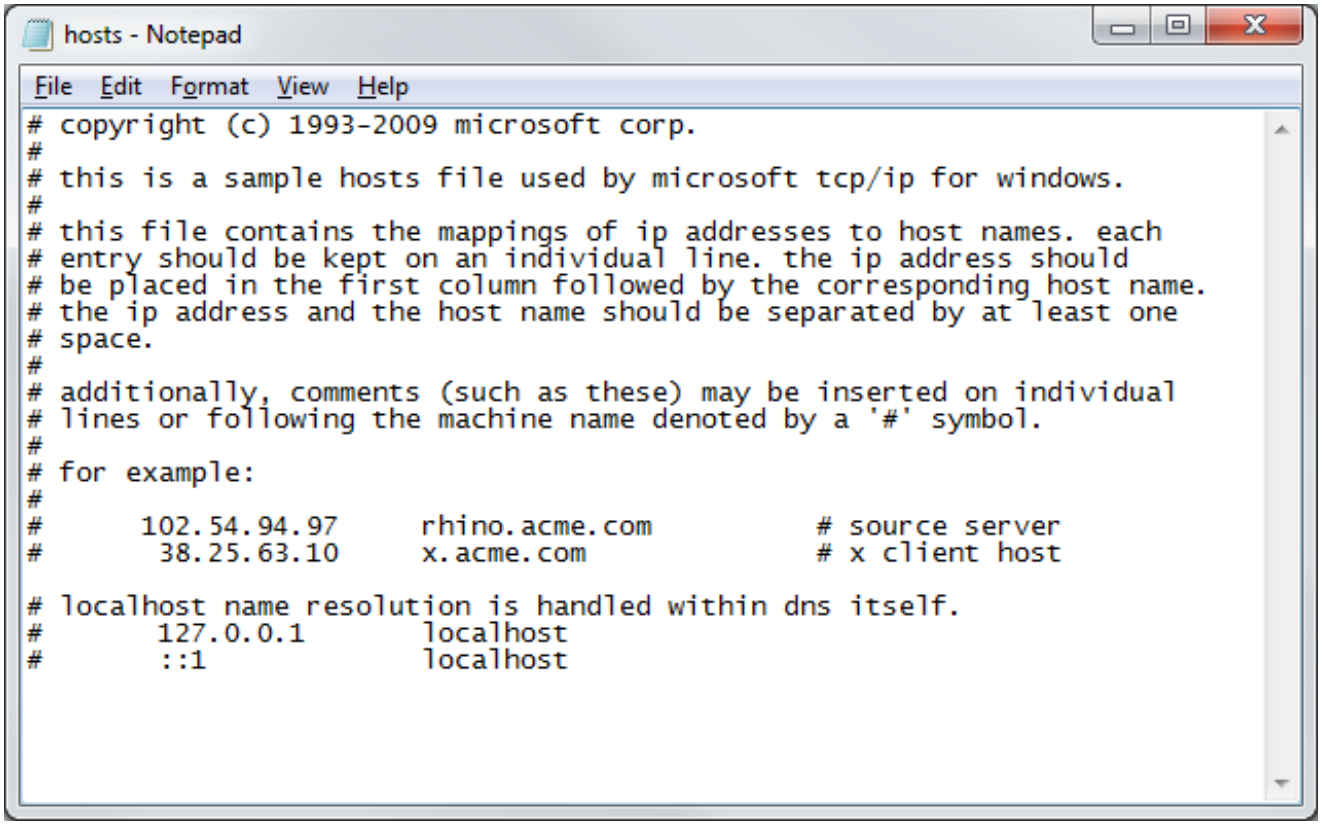
\etc\hosts

No Windows, este arquivo se encontra em:

c:\windows\system32\drivers\etc\hosts.txt

Arquivo hosts

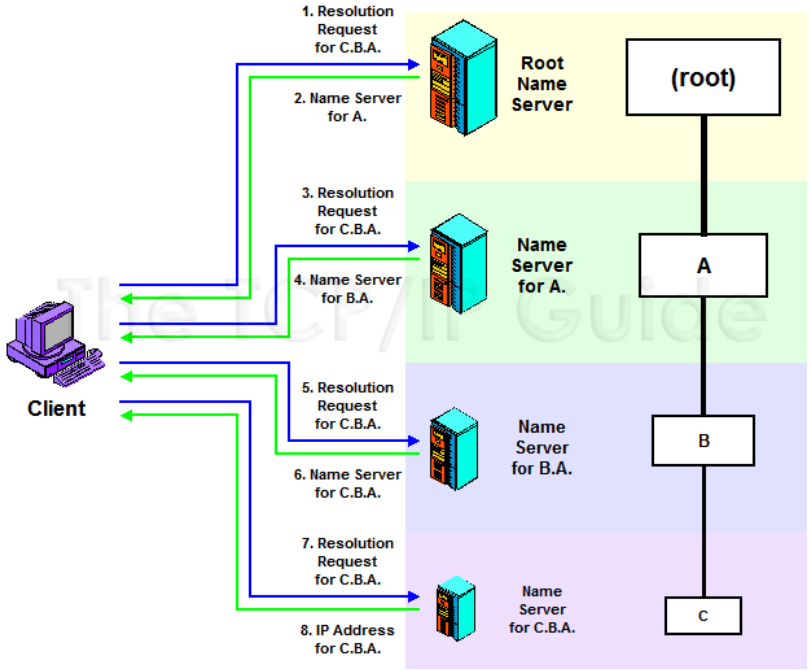
Arquivo *hosts.txt* no Windows:



DNS – resolução de nomes
Técnica iterativa

Quando um cliente envia uma solicitação iterativa para um servidor DNS, o servidor responde com a resposta à solicitação, ou seja, o endereço IP correspondente, ou então com o nome de outro servidor que tenha as informações.

O cliente original deve então iterar com o novo servidor, enviando uma nova solicitação para este possa responder ou fornecer outro nome de servidor.

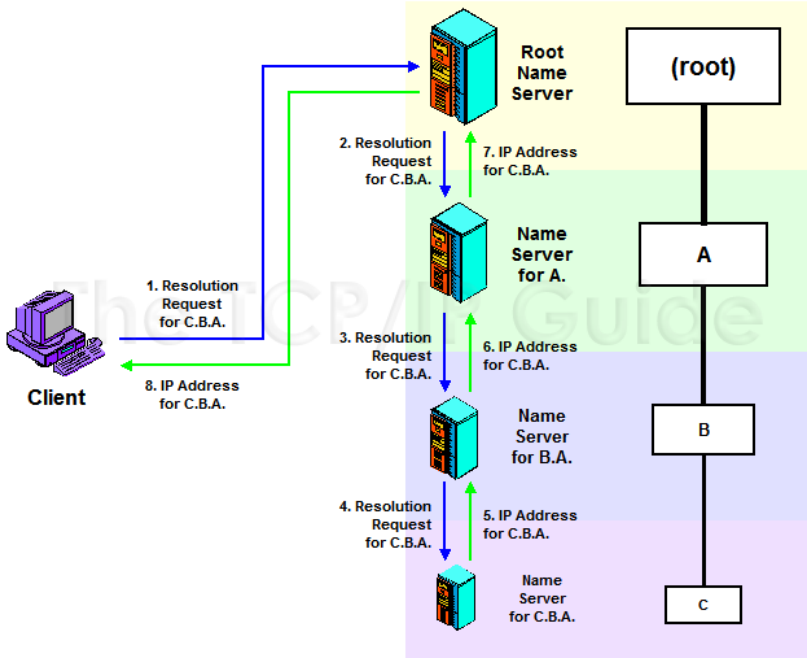


Fonte: tcpipguide.com

DNS – resolução de nomes
Técnica recursiva

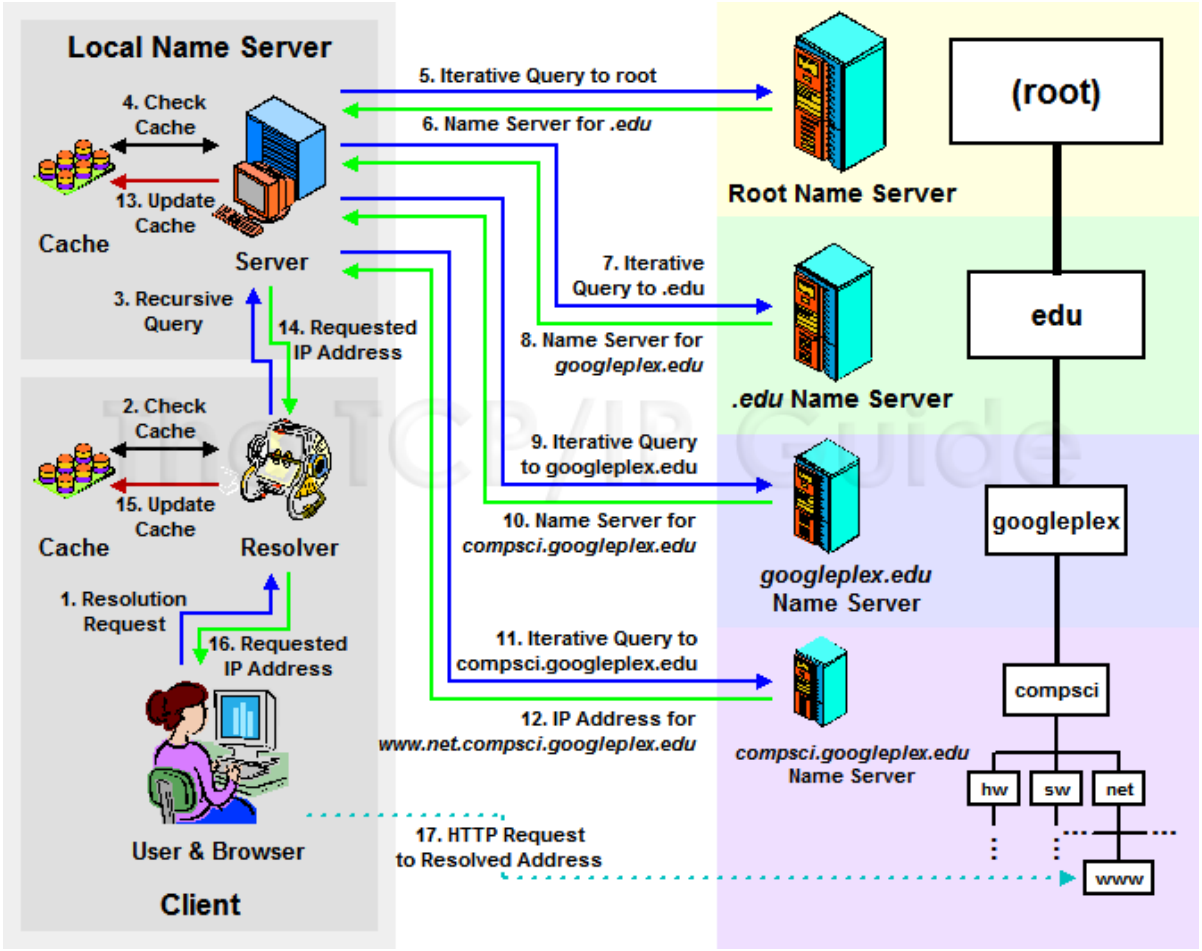
Quando um cliente envia uma solicitação recursiva para um servidor DNS, o servidor responde com a resposta se tiver a informação solicitada. Caso contrário, o servidor assumirá a responsabilidade de encontrar a resposta, tornando-se um cliente em nome do cliente original e enviando novas solicitações para outros servidores.

O cliente original envia apenas uma solicitação e, eventualmente, obtém as informações desejadas (ou uma mensagem de erro, se não estiver disponível).



Fonte: tcpipguide.com

DNS – resolução de nomes
Resumo



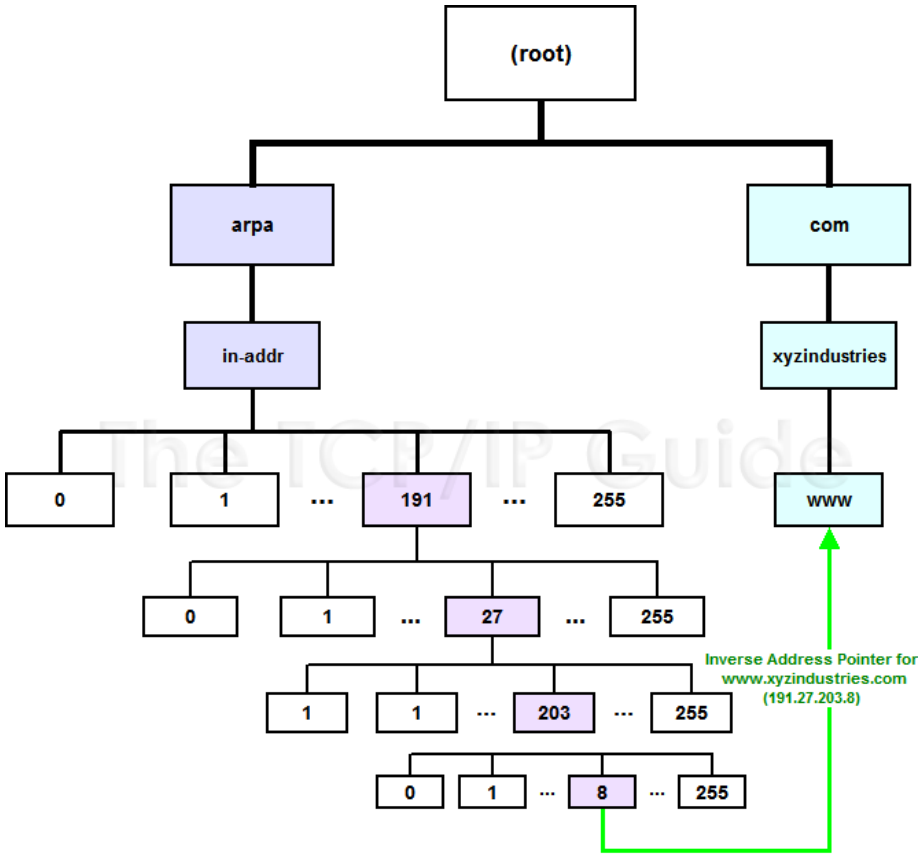
Fonte: tcpipguide.com

DNS – resolução de nomes
Pesquisa reversa

A hierarquia especial "IN-ADDR.ARPA" foi criada para permitir pesquisas reversas fáceis de nomes DNS.

"IN-ADDR.ARPA" contém 256 subdomínios numerados de 0 a 255, cada um dos quais tem 256 subdomínios numerados de 0 a 255 e assim por diante, abaixo de quatro níveis. Assim, cada endereço IP é representado na hierarquia.

No diagrama ao lado, o nome de domínio DNS “www.xyzindustries.com” tem um registro de recurso convencional apontando para seu endereço IP 191.27.203.8, bem como um registro de resolução reversa em 8.203.27.191.IN-ADDR.ARPA, apontando para o nome de domínio “www.xyzindustries.com”.



Fonte: tcpipguide.com

DNS – Ferramentas

NSLOOKUP é uma ferramenta disponível em ambiente Linux e Windows que permite pesquisar informações sobre registros de DNS de um determinado servidor DNS.

DNS – NSLOOKUP

Exemplo de pesquisa recursiva sem especificar o servidor DNS:

```
C:\>nslookup www.brasil.gov.br
Server:  UnKnown
Address:  192.168.0.1

Non-authoritative answer:
Name:    www.brasil.gov.br
Address:  170.246.255.242
```

DNS – NSLOOKUP

Exemplo de pesquisa recursiva especificando o servidor DNS:

```
C:\>nslookup www.brasil.gov.br dns.google
Server:  dns.google
Address:  8.8.8.8

Non-authoritative answer:
Name:    www.brasil.gov.br
Address:  170.246.255.242
```

DNS – NSLOOKUP

Exemplo de pesquisa do registro SOA:

```
C:\>nslookup -type=soa www.brasil.gov.br dns.google
Server:  dns.google
Address:  8.8.4.4

brasil.gov.br
        primary name server = alpha.planalto.gov.br
        responsible mail addr = postmaster.planalto.gov.br
        serial  = 2020080810
        refresh = 300 (5 mins)
        retry   = 300 (5 mins)
        expire  = 604800 (7 days)
        default TTL = 300 (5 mins)
```

DNS – NSLOOKUP

Exemplo de pesquisa recursiva especificando o servidor DNS autoritativo do domínio:

```
C:\>nslookup www.brasil.gov.br alpha.planalto.gov.br
Server:  alpha.planalto.gov.br
Address:  170.246.255.10

Name:    www.brasil.gov.br
Address:  170.246.255.242
```

DNS – NSLOOKUP

Exemplo de pesquisa não recursiva para “br”:

```
C:\>nslookup -norecurse -type=ns br a.root-servers.net
(…)
Server:  UnKnown
Address:  198.41.0.4

br      nameserver = a.dns.br
br      nameserver = b.dns.br
(…)
a.dns.br        internet address = 200.219.148.10
b.dns.br        internet address = 200.189.41.10
(…)
```

DNS – NSLOOKUP

Exemplo de pesquisa não recursiva para “gov.br”:

```
C:\>nslookup -norecurse -type=ns gov.br a.dns.br
Server:  a.dns.br
Address:  200.219.148.10

gov.br  nameserver = a.dns.br
gov.br  nameserver = b.dns.br
gov.br  nameserver = c.dns.br
gov.br  nameserver = d.dns.br
gov.br  nameserver = e.dns.br
gov.br  nameserver = f.dns.br
```

DNS – NSLOOKUP

Exemplo de pesquisa não recursiva para “brasil.gov.br”:

```
C:\>nslookup -norecurse -type=ns brasil.gov.br a.dns.br
Server:  a.dns.br
Address:  200.219.148.10

brasil.gov.br   nameserver = alpha.planalto.gov.br
brasil.gov.br   nameserver = alpha2.planalto.gov.br
alpha.planalto.gov.br   internet address =
170.246.255.10
alpha2.planalto.gov.br  internet address =
170.246.255.11
```

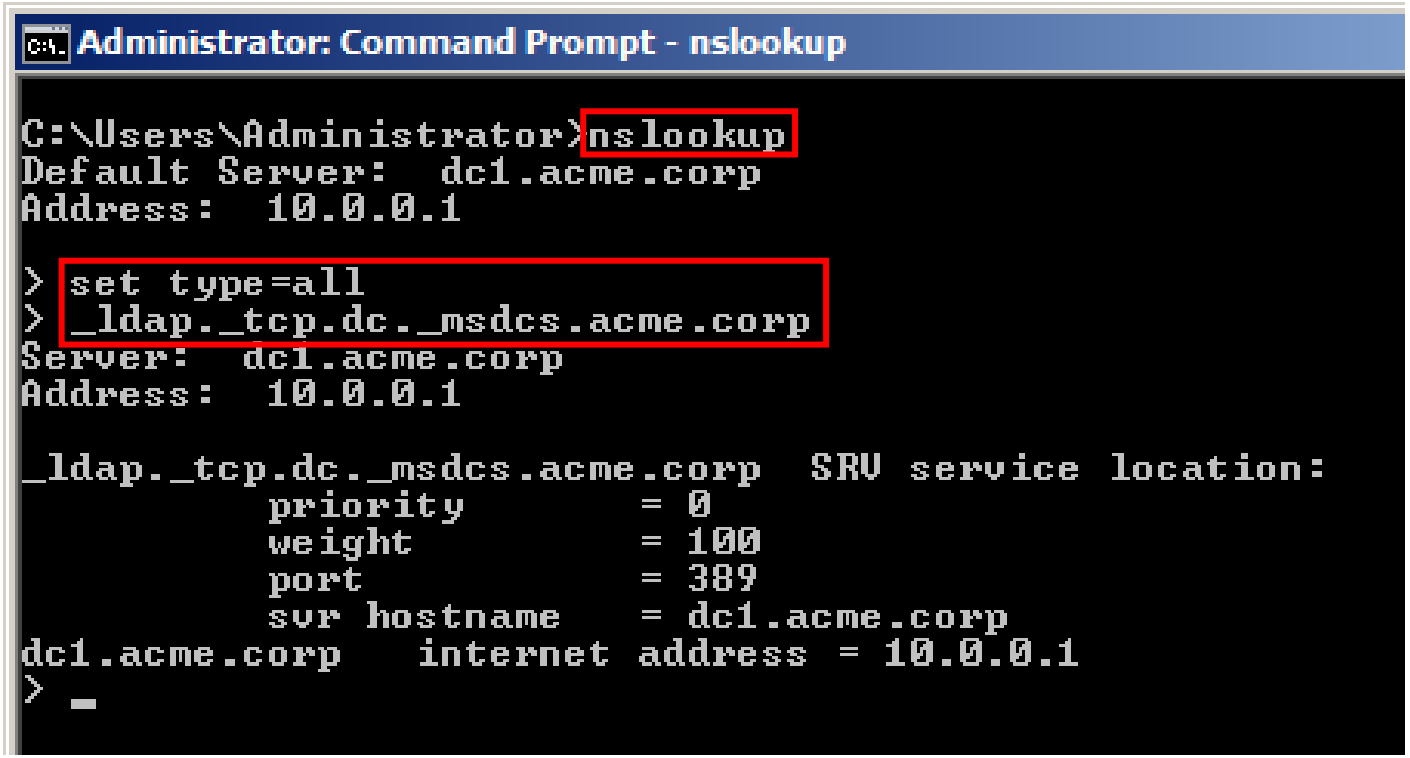
DNS – NSLOOKUP

Exemplo de pesquisa não recursiva para “www.brasil.gov.br”:

```
C:\>nslookup -norecurse -type=a www.brasil.gov.br
alpha.planalto.gov.br

Server:  alpha.planalto.gov.br
Address:  170.246.255.10

Name:    www.brasil.gov.br
Address:  170.246.255.242
```

DNS – NSLOOKUP

Exemplo de pesquisa para o controlador de domínio que atende uma determinada rede, neste exemplo a rede ACME.CORP:



Para saber mais...

… leia o material online sobre Domain Name System, de Júlio Battisti.

... leia a apostila Domain Name Service Configuração e Administração, de Rubens Queiroz de Almeida.

… veja a animação online do funcionamento do protocolo DNS, da RAD University.

… leia o tutorial DNS apresentado no 3º PTT Fórum, do registro.br.

… veja a lista de Top-Level Domains, da Internet Assigned Numbers Authority (IANA).

… veja a lista de Domínios de Segundo Nível do Brasil, do registro.br.

Módulo 5

File Transfer Protocol

FTP

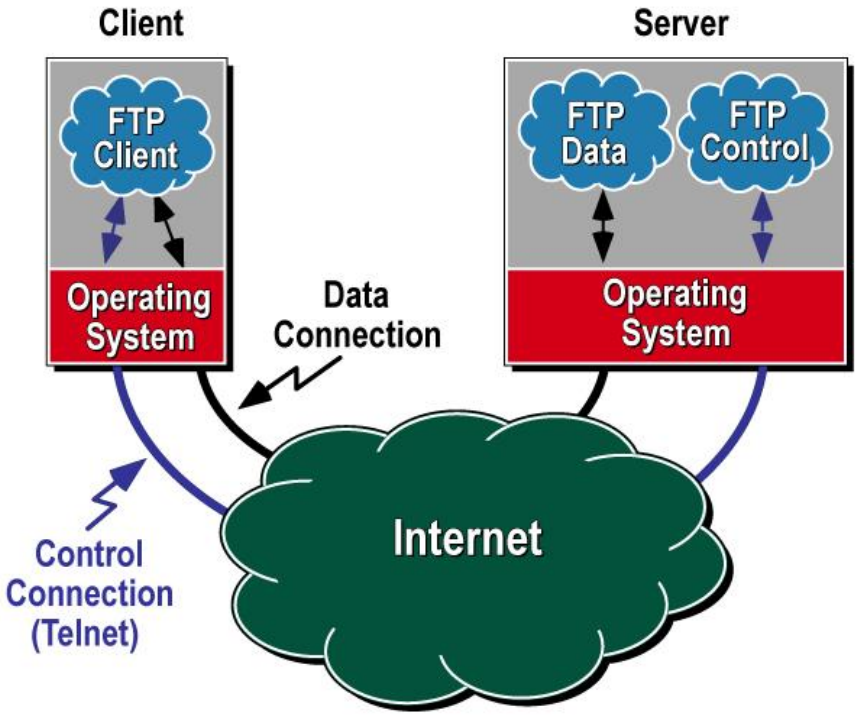
O protocolo FTP (File Transfer Protocol) é usado para transferir arquivos usando como método de transporte o protocolo TCP.

É baseado no modelo cliente/servidor e usa duas conexões, uma para dados e outra para controle.

FTP

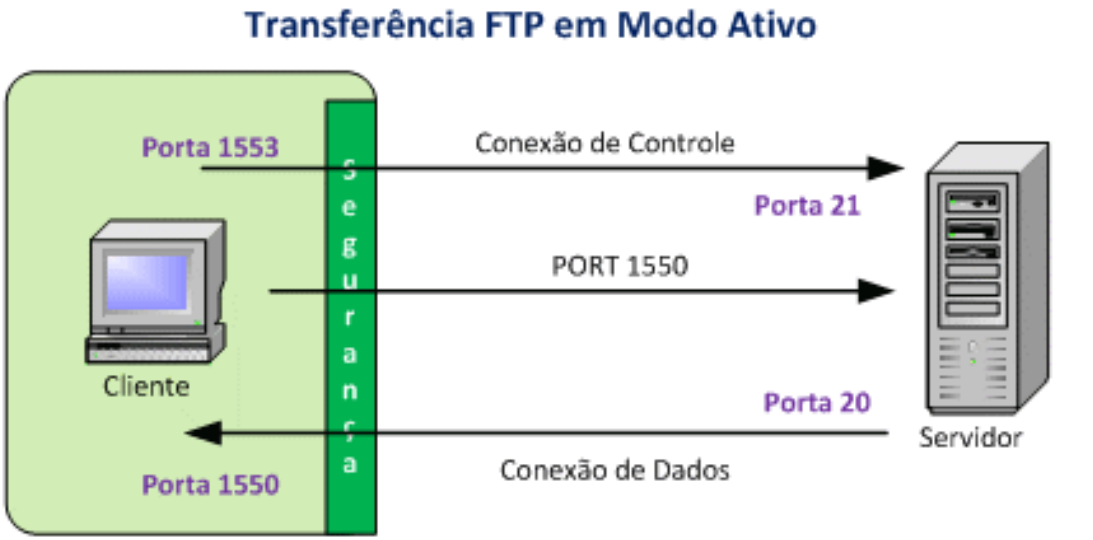
Quando o cliente FTP deseja conectar-se ao servidor FTP, é realizada uma conexão TCP na porta 21 do servidor, denominada conexão de controle. É por esta conexão que serão enviados e recebidos os comandos de listagem e cópias de arquivos, por exemplo.

Quando o cliente realiza uma cópia ou envio de arquivo, uma nova conexão TCP é aberta, desta vez na porta 20 do servidor, por onde irão trafegar os arquivos.



FTP – modo ativo

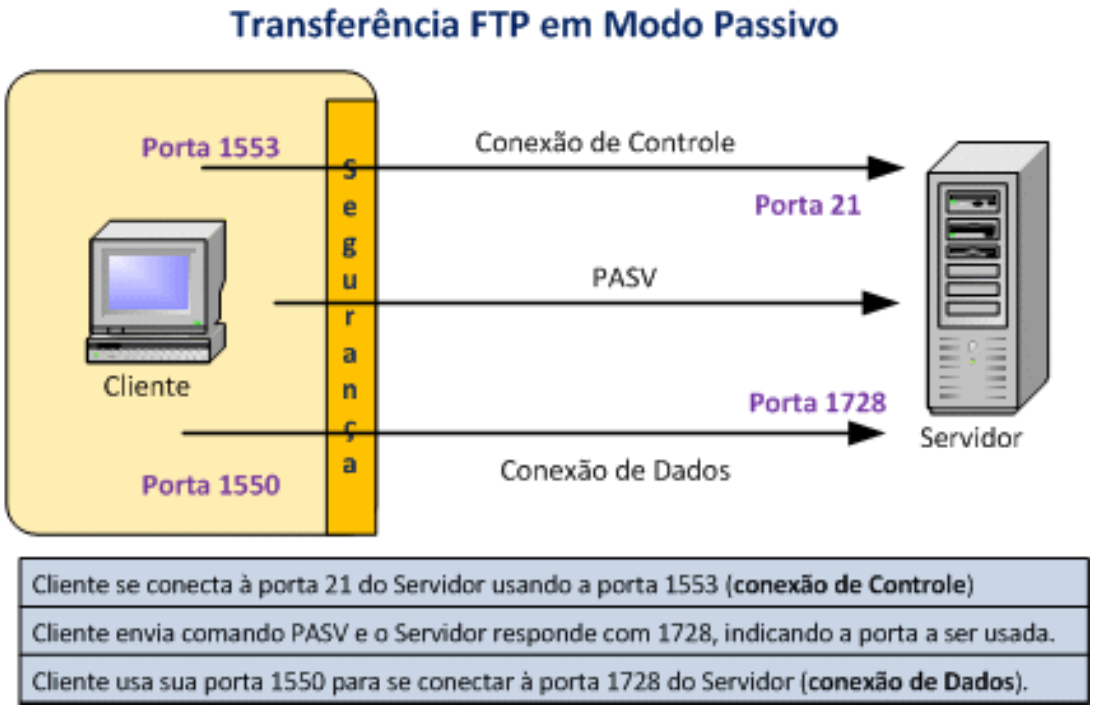
No modo ativo a conexão é gerenciada pelo cliente FTP. Neste caso, após estabelecer uma conexão TCP na porta 21 do servidor, o cliente envia um comando PORT seguido do número da porta onde o servidor deverá estabelecer a conexão de dados.



Cliente se conecta à porta 21 do Servidor usando a porta 1553 (**conexão de Controle**)
Cliente envia comando PORT 1550, notificando ao Servidor qual porta deve ser usada
Servidor usa sua porta 20 para se conectar à porta 1550 do Cliente (**conexão de Dados**).

FTP – modo passivo

No modo passivo a conexão é gerenciada pelo servidor FTP. Neste caso, após estabelecer uma conexão TCP na porta 21 do servidor, o cliente envia um comando PASV e espera uma resposta do servidor indicando qual porta deverá ser usada para transmissão de dados.



FTP – ferramentas

FTP é uma ferramenta disponível em ambiente Linux e Windows que permite listar, enviar e receber arquivos de um servidor FTP.

Para saber mais...

… leia o tutorial Serviço de FTP, de Gerson Konnus.

… leia o tutorial How to set up an FTP Server in Windows 2000, da Microsoft Corporation.

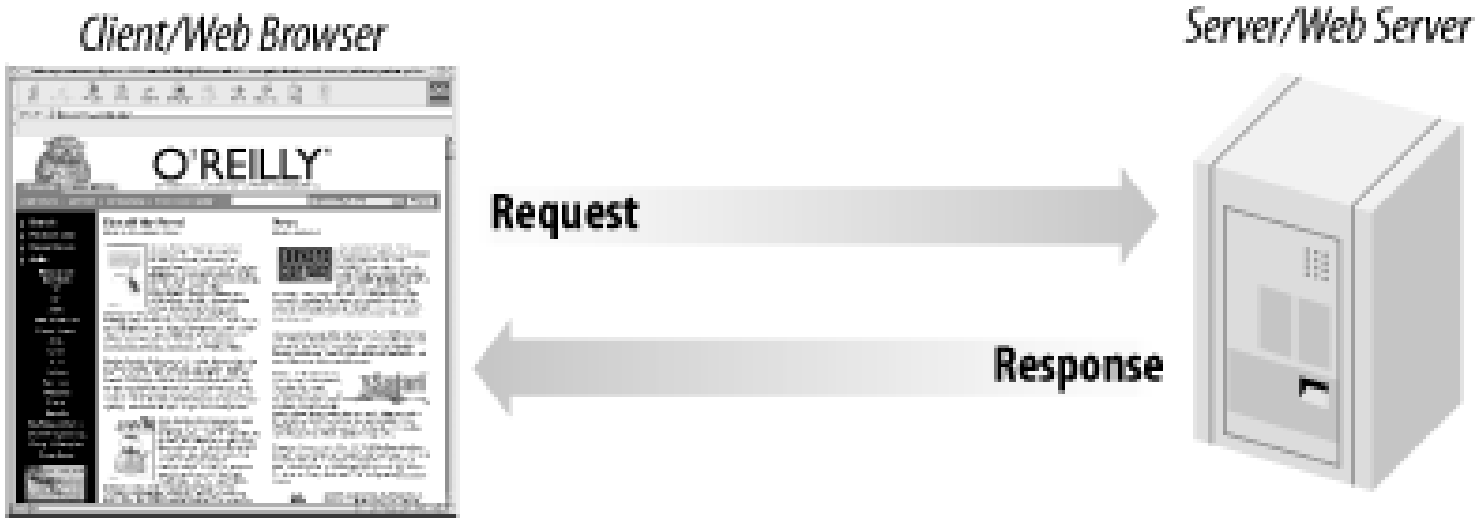
Módulo 6

Hypertext Transfer Protocol

HTTP

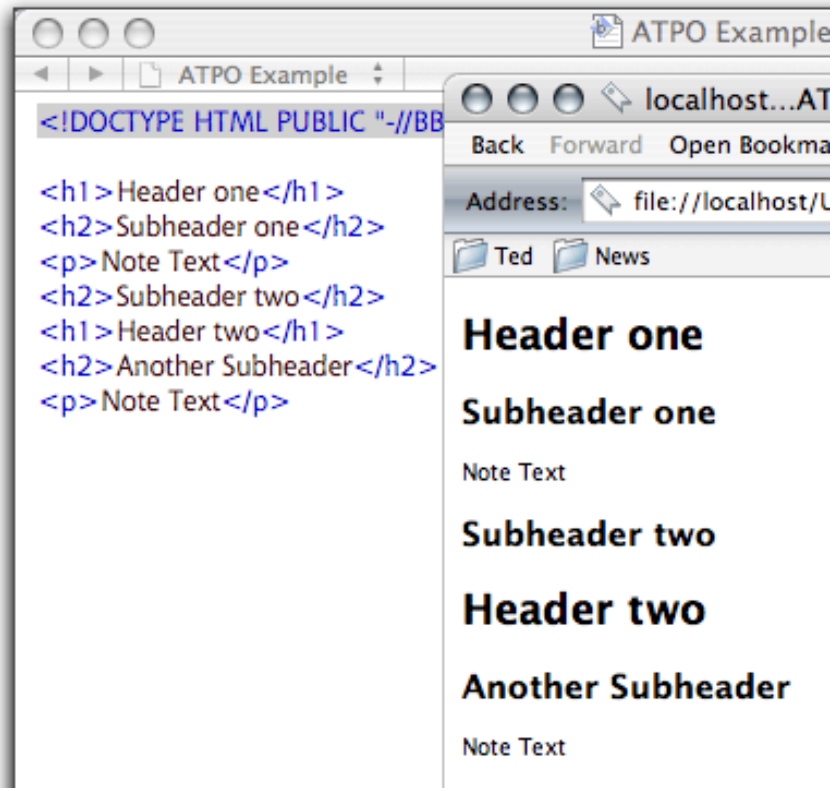
O Hypertext Transfer Protocol, ou Protocolo de Transferência de Hipertexto, é usado para transferência de dados do tipo hipertexto, que nada mais é que um texto estruturado que pode conter elementos de multimídia como som e imagem e que utiliza ligações lógicas para outros textos.

O protocolo HTTP trabalha no modelo cliente/servidor, e podem ser transferidos dados do tipo texto claro, HTML, som, imagens, entre outros.

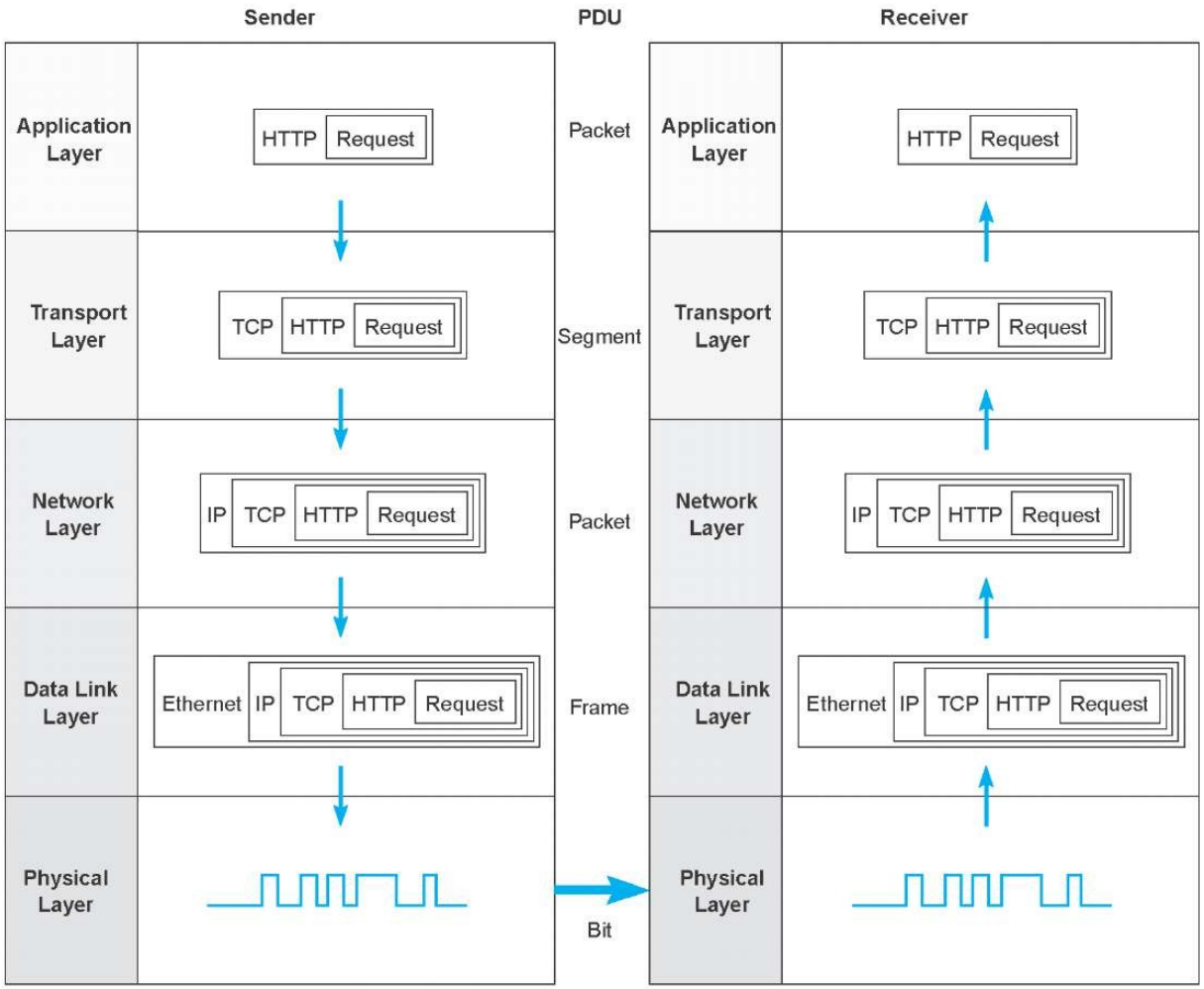


HTTP

O HyperText Markup Language, ou Linguagem de Marcação de Hipertexto é usado para formatar páginas Web. A linguagem HTML é interpretada pelos navegadores Web.

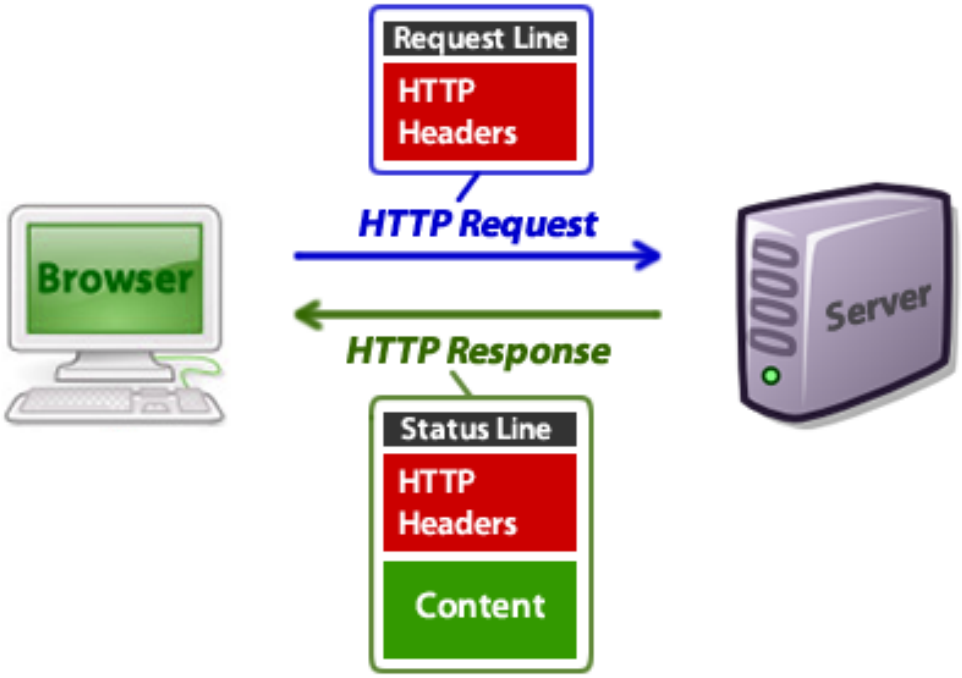


HTTP – PDU


Sender
PDU
Receiver
Application Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer
HTTP Request
TCP HTTP Request
IP TCP HTTP Request
Ethernet IP TCP HTTP Request
Packet
Segment
Packet
Frame
Bit
Application Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer
HTTP Request
TCP HTTP Request
IP TCP HTTP Request
Ethernet IP TCP HTTP Request

HTTP

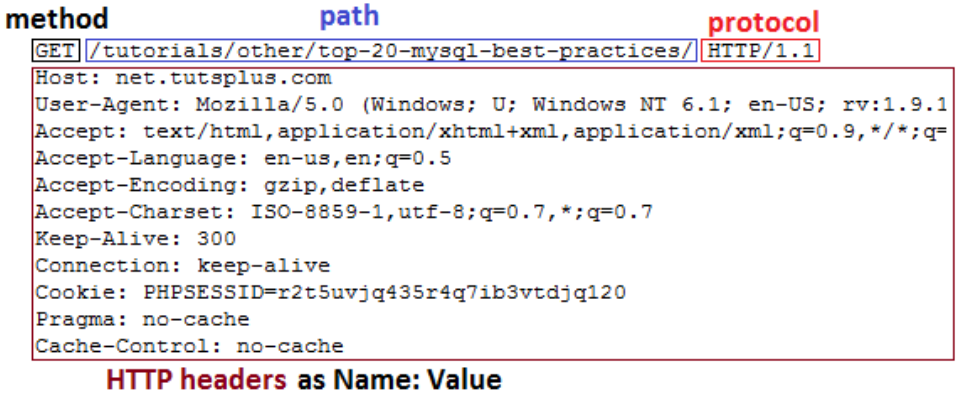
Uma sessão HTTP inicia com a requisição do cliente, que envia uma mensagem HTTP Request. O servidor Web configurado por padrão na porta 80 captura a requisição e envia uma mensagem HTTP Response, que contém o cabeçalho da resposta e os dados do recurso requisitado.



HTTP – Estrutura de requisição

A estrutura do pedido de requisição pode ser dividida em quatro partes:

- O método (method) indica o tipo de requisição. Os mais comuns são GET, HEAD e POST;
- O caminho (path) é a localização do recurso que se deseja recuperar. Pode ser uma página HTML, uma imagem, arquivo de áudio, etc;
- O protocolo (protocol) contém a versão do protocolo HTTP que o navegador está usando;
- O cabeçalho (header) HTTP contém várias informações sobre a requisição e o navegador Web.



HTTP – Métodos

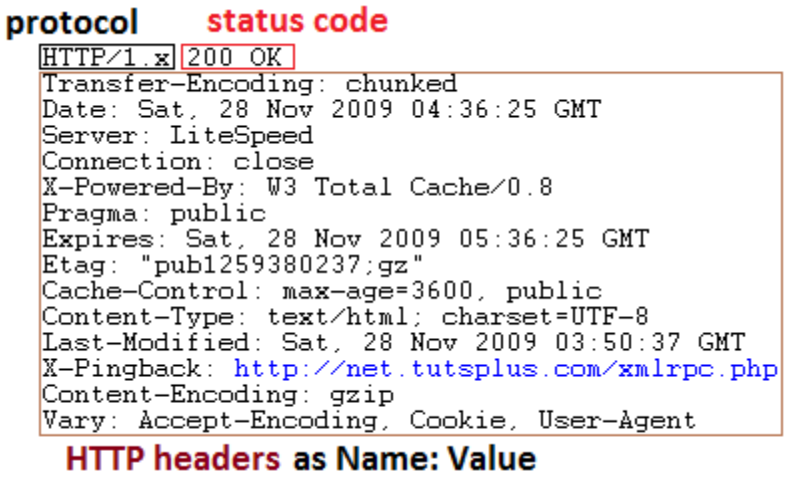
O método usado pelo protocolo HTTP para o pedido de requisição pode ser do tipo:

- GET – método usado para recuperar as informações sobre um determinado recurso e o próprio recurso;
- HEAD – método usado para recuperar apenas as informações sobre um determinado recurso;
- POST – método usado para enviar informações do cliente para o servidor. Usado em preenchimento de formulário, por exemplo.

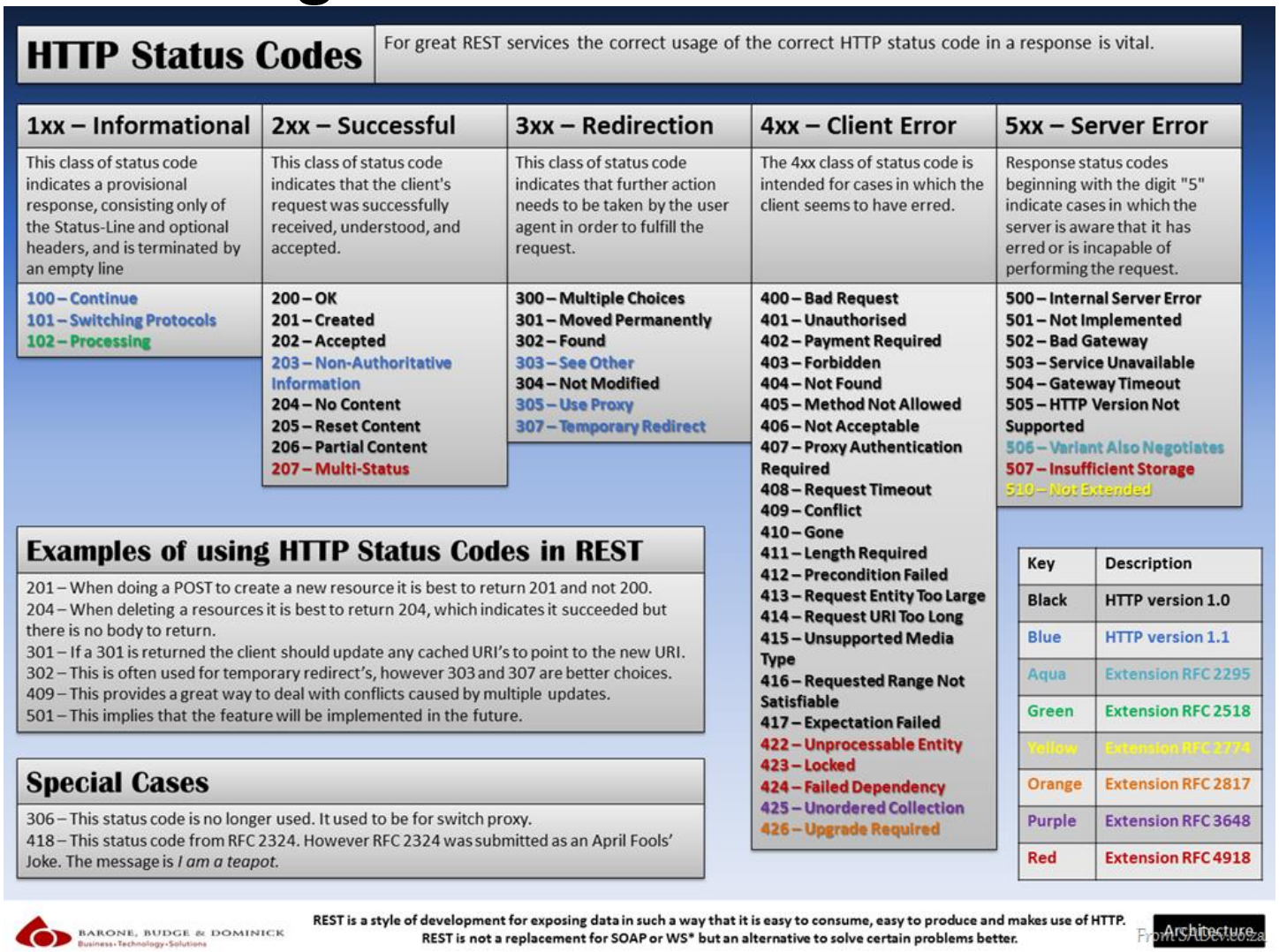
HTTP – Estrutura de Resposta

A estrutura do pedido de resposta pode ser dividida em três partes:

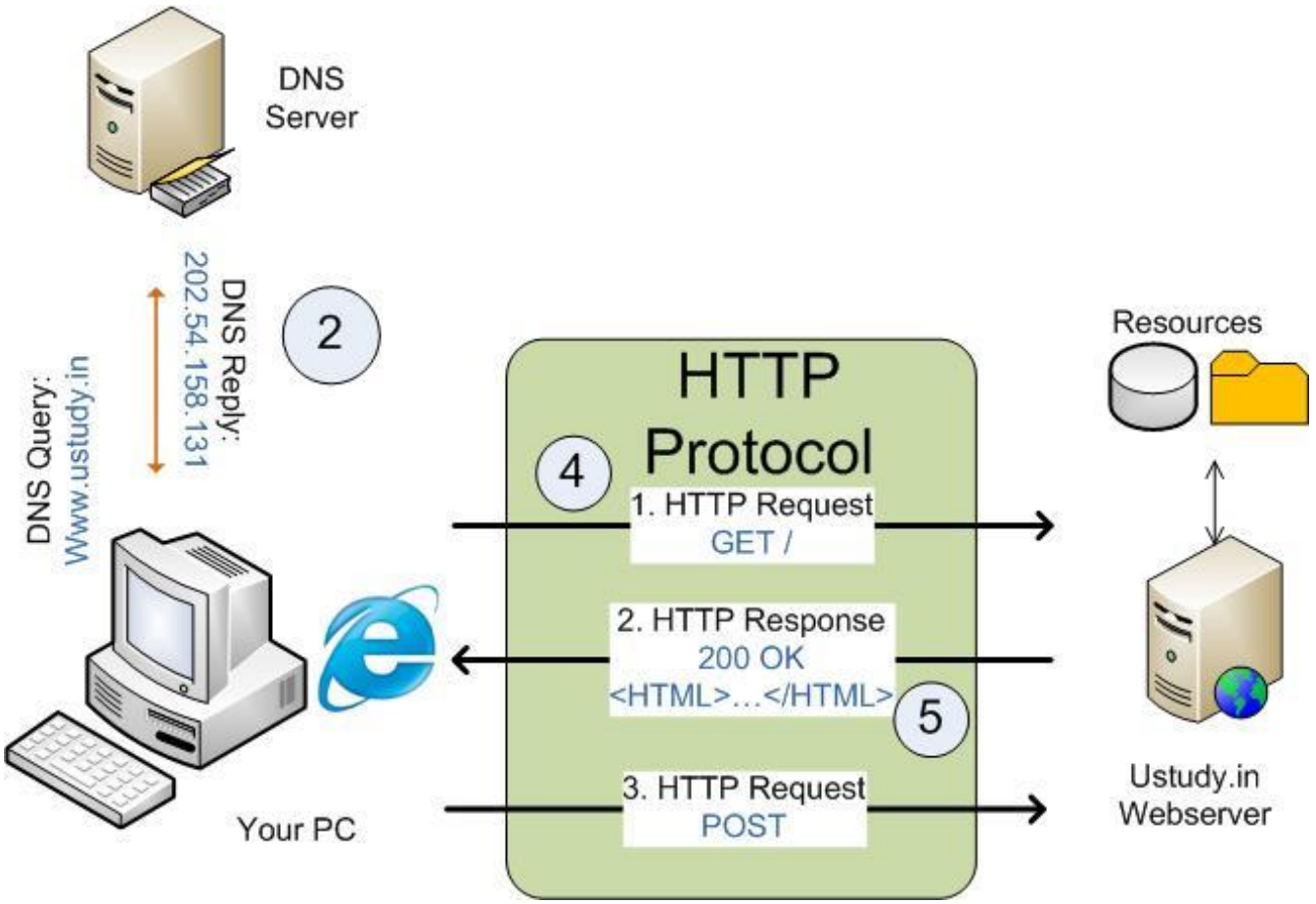
- O protocolo (protocol) contém a versão do protocolo HTTP que o servidor está usando;
- O código de estado (status code) indica, entre outras coisas, se a requisição foi ou não atendida com sucesso;
- O cabeçalho (header) HTTP contém várias informações sobre a resposta e o servidor Web.



HTTP – Códigos de estado



HTTP – Exemplo



HTTP – Exemplo

```
josh@blackbox:~$ telnet en.wikipedia.org 80
Trying 208.80.152.2...
Connected to rr.pmtpa.wikimedia.org.
Escape character is '^]'.
GET /wiki/Main_Page http/1.1                                                    Request
Host: en.wikipedia.org

HTTP/1.0 200 OK                                                                 Response headers
Date: Thu, 03 Jul 2008 11:12:06 GMT
Server: Apache
X-Powered-By: PHP/5.2.5
Cache-Control: private, s-maxage=0, max-age=0, must-revalidate
Content-Language: en
Vary: Accept-Encoding,Cookie
X-Vary-Options: Accept-Encoding;list-contains=gzip,Cookie;string-contains=enwikiToken;string-contains=enwikiLoggedOut;string-contains=enwiki_session;
string-contains=centralauth_Token;string-contains=centralauth_Session;string-contains=centralauth_LoggedOut
Last-Modified: Thu, 03 Jul 2008 10:44:34 GMT
Content-Length: 54218
Content-Type: text/html; charset=utf-8
X-Cache: HIT from sq39.wikimedia.org
X-Cache-Lookup: HIT from sq39.wikimedia.org:3128
Age: 3
X-Cache: HIT from sq38.wikimedia.org
X-Cache-Lookup: HIT from sq38.wikimedia.org:80
Via: 1.0 sq39.wikimedia.org:3128 (squid/2.6.STABLE18), 1.0 sq38.wikimedia.org:80 (squid/2.6.STABLE18)
Connection: close

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">      Response body
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en" dir="ltr">
        <head>
                <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
                        <meta name="keywords" content="Main Page,1778,1844,1863,1938,1980 Summer Olympics,2008,2008 Guizhou riot,2008 Jerusal
...
''' This content has been removed to save space
...
"Non-profit organization">nonprofit</a> <a href="http://en.wikipedia.org/wiki/Charitable_organization" title="Charitable organization">charity</a>.<b
r /></li>
                                <li id="privacy"><a href="http://wikimediafoundation.org/wiki/Privacy_policy" title="wikimedia:Privacy policy">Privac
y policy</a></li>
                                <li id="about"><a href="/wiki/Wikipedia:About" title="Wikipedia:About">About Wikipedia</a></li>
                                <li id="disclaimer"><a href="/wiki/Wikipedia:General_disclaimer" title="Wikipedia:General disclaimer">Disclaimers</a>
</li>
                        </ul>
                </div>
</div>

                <script type="text/javascript">if (window.runOnloadHook) runOnloadHook();</script>
<!-- Served by srv93 in 0.050 secs. --></body></html>
Connection closed by foreign host.
josh@blackbox:~$
```

HTTP – Exemplo

No exemplo a seguir é demonstrado como acessar um site usando o TELNET ao invés de um cliente Web comum.

O usuário conecta-se ao site Web usando o comando TELNET no console por meio do comando:

`telnet en.wikipedia.org 80`

```
josh@blackbox:~$ telnet en.wikipedia.org 80
Trying 208.80.152.2...
Connected to rr.pmtpa.wikimedia.org.
Escape character is '^]'.
```

HTTP – Exemplo

O usuário deseja recuperar o recurso **/wiki/Main_Page** por meio dos comandos:

```
GET /wiki/Main_Page http/1.1
Host: en.wikipedia.org
```

HTTP – Exemplo

O servidor responde com um código de estado **200 OK**, indicando que a página existe, bem como o cabeçalho de resposta.

```
HTTP/1.0 200 OK
Date: Thu, 03 Jul 2008 11:12:06 GMT
Server: Apache
X-Powered-By: PHP/5.2.5
Cache-Control: private, s-maxage=0, max-age=0, must-revalidate
Content-Language: en
Vary: Accept-Encoding,Cookie
X-Vary-Options: Accept-Encoding;list-contains=gzip,Cookie;string-contains=enwikiToken;string-contains=enwikiLoggedOut;string-contains=enwiki_session;
string-contains=centralauth_Token;string-contains=centralauth_Session;string-contains=centralauth_LoggedOut
Last-Modified: Thu, 03 Jul 2008 10:44:34 GMT
Content-Length: 54218
Content-Type: text/html; charset=utf-8
X-Cache: HIT from sq39.wikimedia.org
X-Cache-Lookup: HIT from sq39.wikimedia.org:3128
Age: 3
X-Cache: HIT from sq38.wikimedia.org
X-Cache-Lookup: HIT from sq38.wikimedia.org:80
Via: 1.0 sq39.wikimedia.org:3128 (squid/2.6.STABLE18), 1.0 sq38.wikimedia.org:80 (squid/2.6.STABLE18)
Connection: close
```

Response headers

HTTP – Exemplo

O servidor responde com o conteúdo do recurso solicitado no formato HTML.

Response body

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en" dir="ltr">
        <head>
                <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
                                <meta name="keywords" content="Main Page,1778,1844,1863,1938,1980 Summer Olympics,2008,2008 Guizhou riot,2008 Jerusal
...
''' This content has been removed to save space
'''
"Non-profit organization">nonprofit</a> <a href="http://en.wikipedia.org/wiki/Charitable_organization" title="Charitable organization">charity</a>.<b
r /></li>
                                <li id="privacy"><a href="http://wikimediafoundation.org/wiki/Privacy_policy" title="wikimedia:Privacy policy">Privac
y policy</a></li>
                                <li id="about"><a href="/wiki/Wikipedia:About" title="Wikipedia:About">About Wikipedia</a></li>
                                <li id="disclaimer"><a href="/wiki/Wikipedia:General_disclaimer" title="Wikipedia:General disclaimer">Disclaimers</a>
</li>
                        </ul>
                </div>
</div>

                <script type="text/javascript">if (window.runOnloadHook) runOnloadHook();</script>
<!-- Served by srv93 in 0.050 secs. --></body></html>
```

HTTP – Exemplo

O servidor encerra a conexão.

```
Connection closed by foreign host.
josh@blackbox:~$
```

Para saber mais...

… acesse o visualizador de Cabeçalho de Requisição e Resposta HTTP web-sniffer.net.

… acesse o visualizador de Cabeçalho de Requisição e Resposta HTTP web-sniffer.me.

Módulo 7

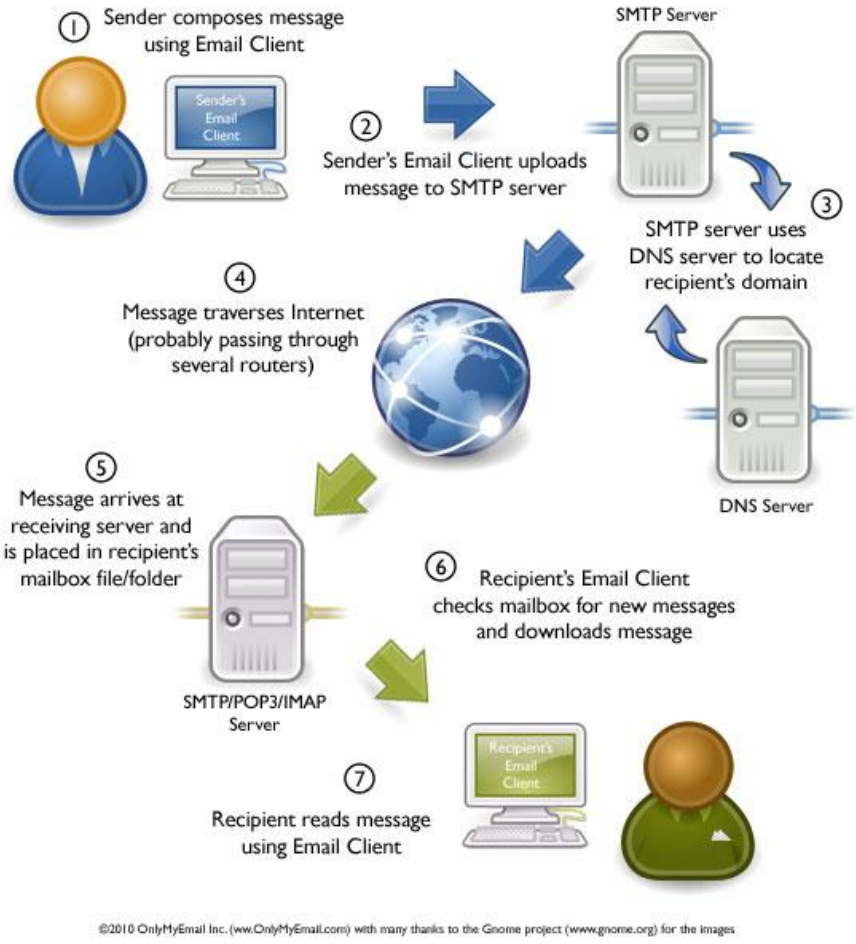
Correio Eletrônico

Introdução

Correio eletrônico é um método digital para troca de mensagens entre um remetente e um ou vários destinatários.

Correio eletrônico

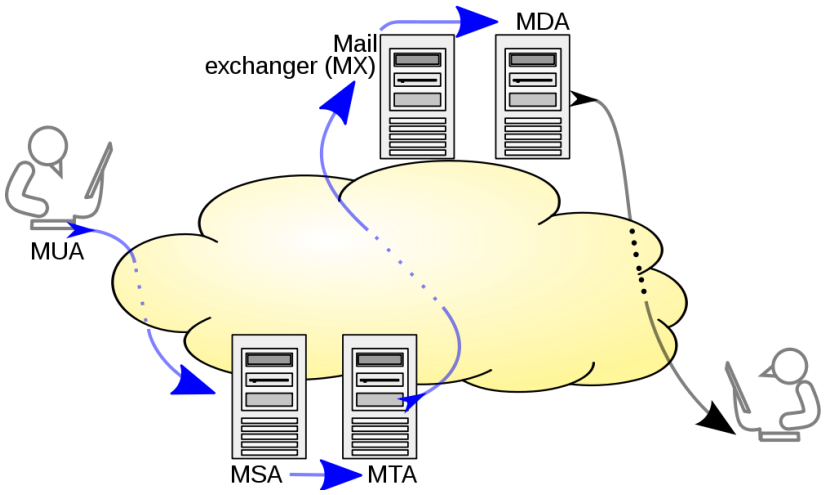
O sistema de correio eletrônico é composto por servidores de correio, que contêm as caixas postais dos usuários, e por clientes de correio, que permitem que os usuários possam interagir com o sistema, ou seja, lendo e postando mensagens.



Correio eletrônico
Componentes

O cliente de correio é também conhecido como MUA (*Mail User Agent*).

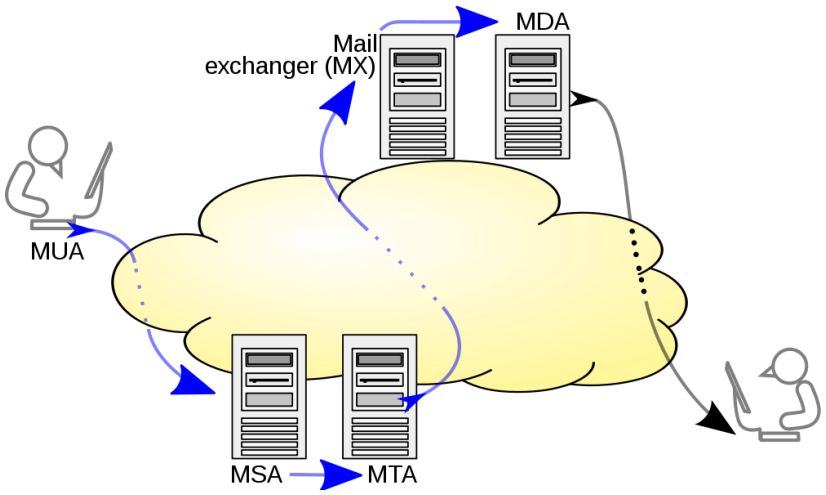
O MUA permite que o usuário possa criar, enviar e receber mensagens.



Correio eletrônico
Componentes

Quando o cliente cria uma mensagem, o próximo passo é enviá-la para o MSA (*Mail Submission Agent*), que é responsável por tratar e enviar a mensagem para o MTA (*Mail Transfer Agent*), responsável por enviar a mensagem pela Internet, por meio do protocolo SMTP (*Simple Mail Transfer Protocol*), para o MX (Mail Exchanger) do destinatário.

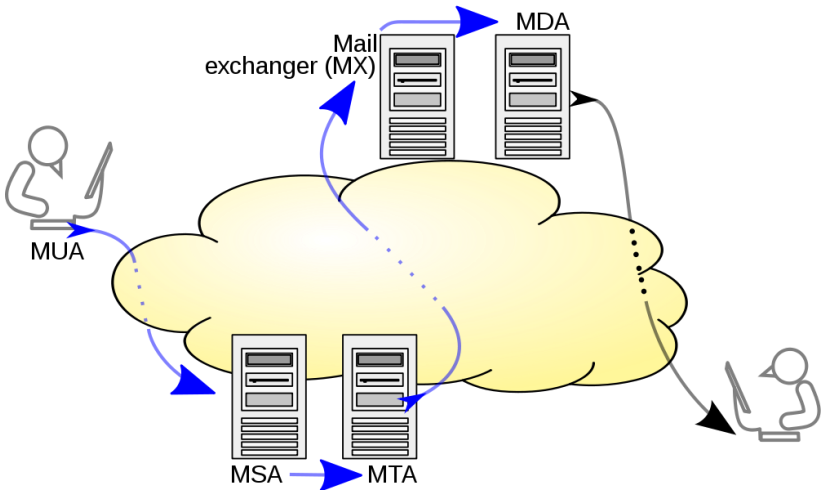
O MSA e o MTA trabalham em conjunto e geralmente estão instalados e configurados no mesmo servidor de correio.



Correio eletrônico
Componentes

Quando o MX do destinatário recebe a mensagem, este direciona para o MDA (*Mail Delivery Agent*), que é responsável por disponibilizar a mensagem para o destinatário por meio de serviços de Webmail, ou por meio dos protocolos POP (*Post Office Protocol*) ou IMAP (*Internet Message Access Protocol*).

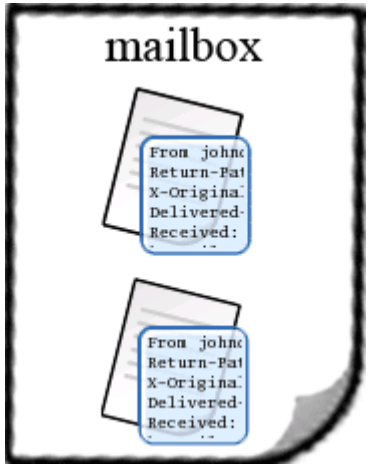
O MX e o MDA trabalham em conjunto e também podem estar instalados e configurados no mesmo servidor.



Correio eletrônico
Armazenamento de mensagens

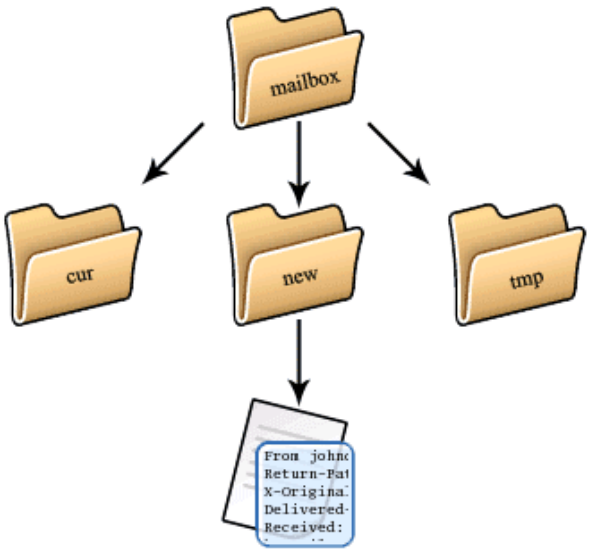
Em ambientes Unix há duas formas de se armazenar mensagens: o formato tradicional denominado Mailbox e o formato moderno Maildir.

O formato Mailbox é a maneira tradicional de armazenar mensagens de correio no mundo Unix. Nesse formato, um arquivo de texto normal é usado como arquivo de caixa de correio do usuário de e-mail.



Correio eletrônico
Armazenamento de mensagens

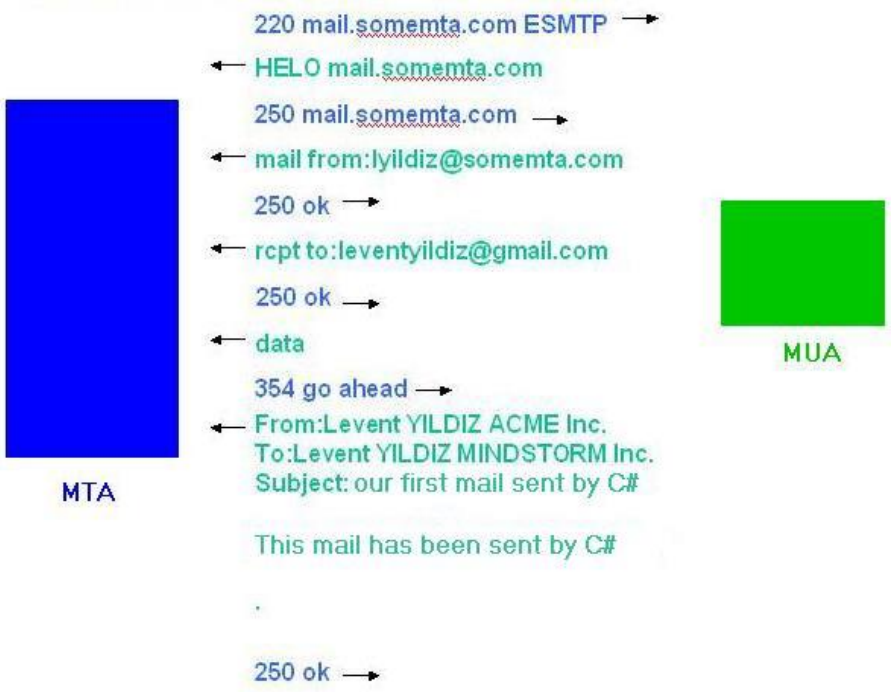
O formato Maildir é uma nova maneira de armazenar mensagens de email em ambientes Unix, e consiste de um diretório geralmente chamado Maildir e que é criado para cada usuário de email. Sob esse diretório, há mais três diretórios chamados new, cur e tmp.



Protocolo SMTP

O protocolo SMTP (*Simple Mail Transfer Protocol*) é usado para transferir mensagens de correio entre o cliente (MUA) e o servidor de correio ou ainda entre servidores de correio de diferentes organizações.

Opera na porta TCP 25 (envio de mensagens entre servidores de correio) ou na porta TCP 587 (envio de mensagens entre cliente e servidor de correio).



Protocolo SMTP
Exemplo

Usando o Telnet como MUA, o usuário Mickey deseja enviar uma mensagem para dois destinatários: Mônica e Magali.

O quadro abaixo mostra a sequencia de comandos. Linhas em negrito são mensagens do servidor.

```
helo disney.com
250 OK
mail from:mickey@disney.com
250 OK - mail from <mickey@disney.com>
rcpt to:monica@panini.com.br
250 OK – Recipient <monica@panini.com.br>
rcpt to:magali@panini.com.br
250 OK – Recipient <magali@panini.com.br>
data
354 Senda data. End with CRLF.CRLF
subject:Ferias
Viaje pra Disney nestas ferias!
.
250 OK
quit
221 closing connection
Connection close by foreign host.
```

Protocolo SMTP
Relay

MTAs configurados como *relay* aberto transmitem mensagens de qualquer domínio, ou mesmo só de domínios determinados, para qualquer outro, sem pedir autenticação, sem restringir (ou restringindo muito pouco) a faixa de endereços IP de origem.

Os *relays* abertos são utilizados por *spammers* pelo fato de proverem anonimato. Para o responsável pelo MTA com *relay* aberto sendo abusado, as consequências são o consumo de recursos e a possível inclusão do MTA em listas de bloqueio. Além disso, ele pode passar a receber mensagens de erro e reclamações sobre os spams enviados via seu MTA.

É importante, ao configurar um MTA, restringir ao máximo os endereços IP que tem permissão para usá-lo como *relay*, se possível limitando ao localhost.

Antes de tornar um serviço de correio eletrônico público é fundamental verificar se ele está se comportando como *relay* aberto. Uma maneira fácil de fazer isso é através de um telnet pela porta adequada, digitando os comandos SMTP diretamente.

Fonte: antispam.br

Protocolo SMTP
Relay – Exemplo

É possível verificar se um MTA é um relay aberto usando o Telnet.

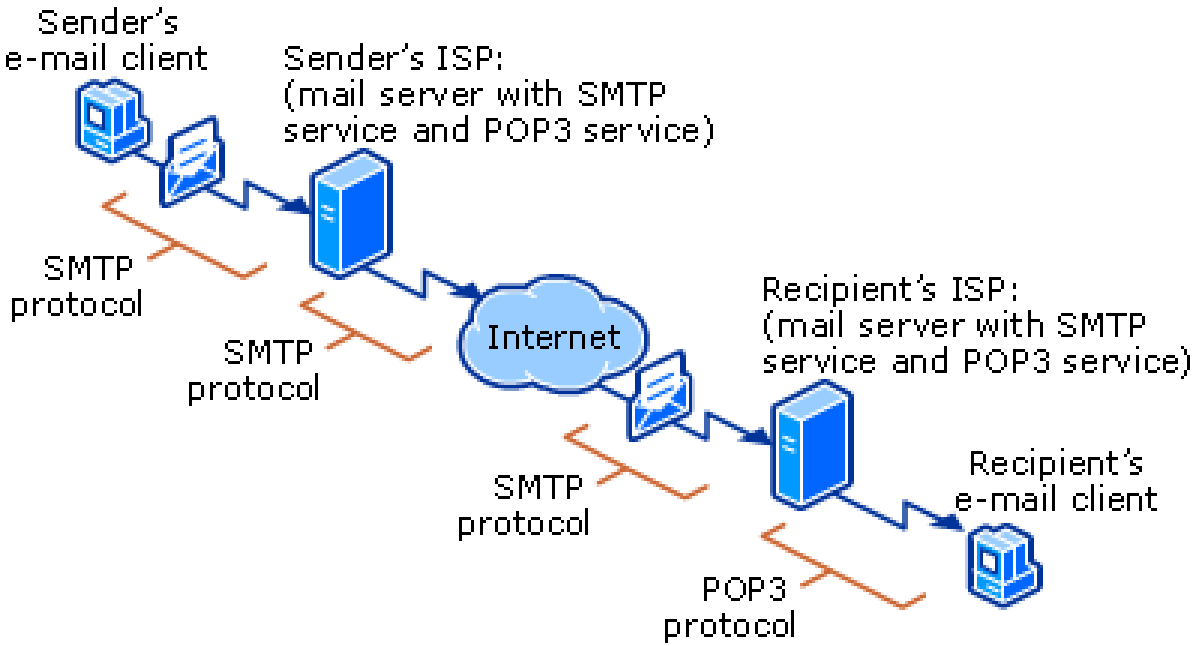
O quadro abaixo mostra a sequencia de comandos. Linhas em negrito são mensagens do servidor.

```
myhost:~$ telnet mailserver.example.com 25
Trying 192.0.2.82...
Connected to mailserver.example.com.
Escape character is '^]'.
220 babbo.example.com ESMTP Sendmail 8.13.4/8.13.4; Tue, 20 Sep
2005 16:31:04 -0300
helo myhost.example.net
250 babbo.example.org Hello IDENT:1008@myhost.example.net
192.0.2.44], pleased to meet you
mail from: <john.doe@example.net>
250 2.1.0 <john.doe@example.net>... Sender ok
rcpt to: <fulano@example.edu>
550 5.7.1 <fulano@example.edu>... Relaying denied. Proper
authentication required.
quit
221 2.0.0 babbo.example.com closing connection
Connection closed by foreign host.
```

Fonte: antispam.br

Protocolo POP

O protocolo POP (*Post Office Protocol*) é usado para baixar todas as mensagens da caixa postal do usuário, armazená-las localmente e em seguida apagá-las do servidor de correio, ainda que seja possível manter uma cópia da mensagem no servidor. O protocolo POP está na versão 3 (POP3) e é indicado para conexões *off-line*. Ele opera na porta TCP 110.



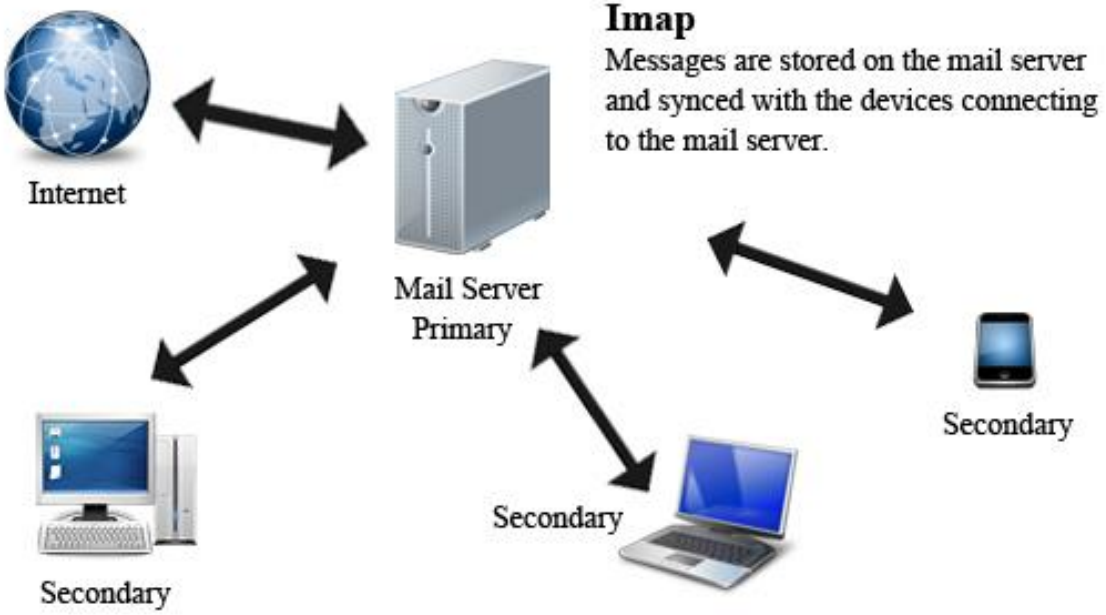
Protocolo POP
Exemplo

Usando o Telnet como MUA, a usuária Mônica visualiza uma mensagem de sua caixa postal e em seguida a apaga. O quadro abaixo mostra a sequencia de comandos. Linhas em negrito são mensagens do servidor.

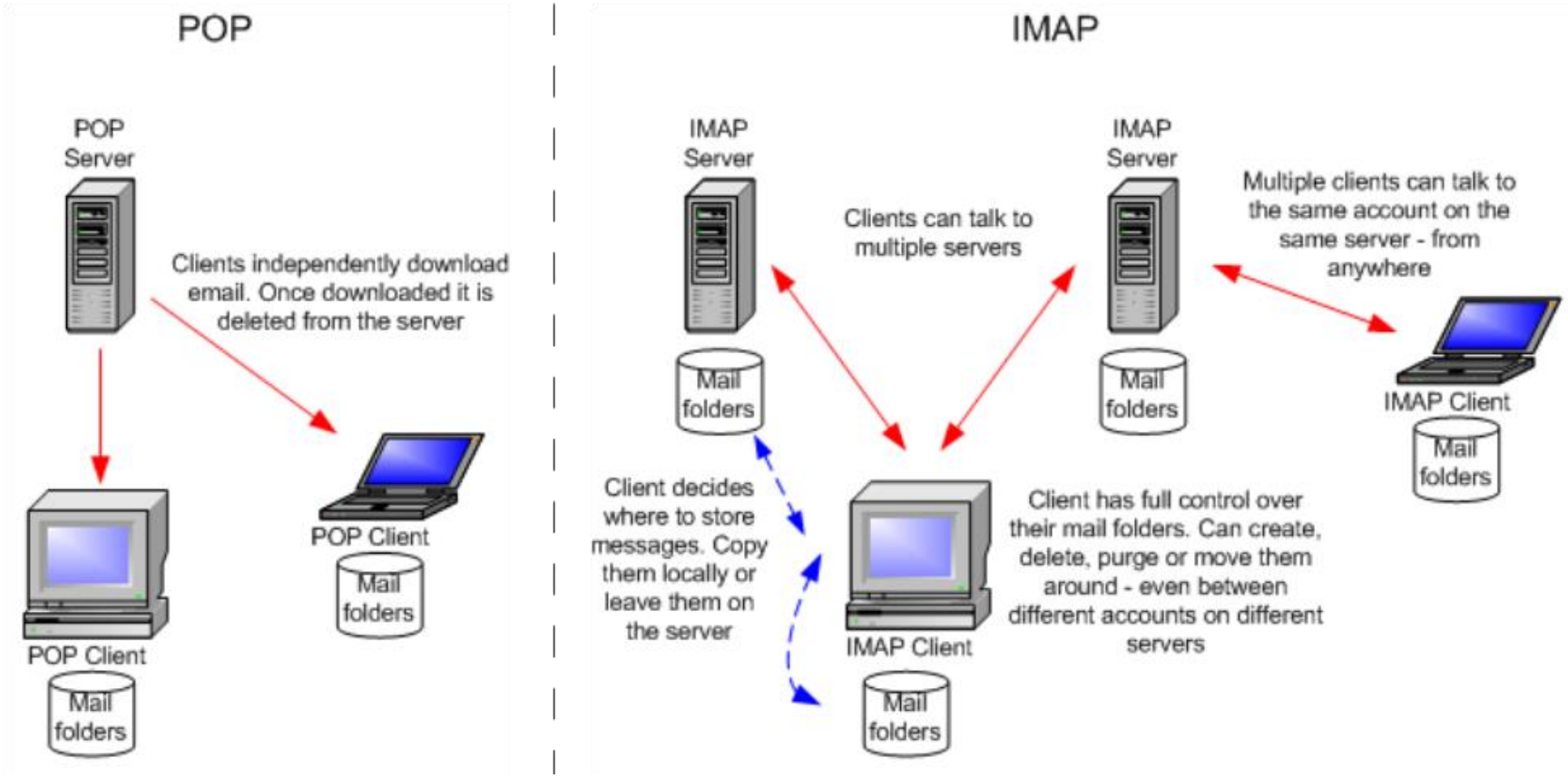
```
user monica
+OK
pass 1234
+OK User successfully logged on
list
1 264
.
retr 1
Received: from disney.com (192.168.0.10 [192.168.0.10]) by MAIL.panini.com.br
with SMTP (Microsoft Exchange Internet Mail Service Version 5.5.2653.13)
          id LN8FJSQ1: Tue, 22 May 2012 08:29:55 -0700
subject:Ferias
Viaje pra Disney nestas ferias!

.
dele 1
+OK
quit
+OK Microsoft Exchange POP3 server version 5.5.2653.23 signing off
Connection close by foreign host.
```

Protocolo IMAP

O protocolo IMAP (*Internet Message Access Protocol*) sincroniza o cliente com o servidor de correio, de modo que as mensagens não precisam ser copiadas do servidor para a máquina local. Permite ainda que vários clientes possam conectar-se a mesma caixa postal. O protocolo IMAP está na versão 4 (IMAP4) e é indicado para conexões *on-line*. Ele opera na porta TCP 143.



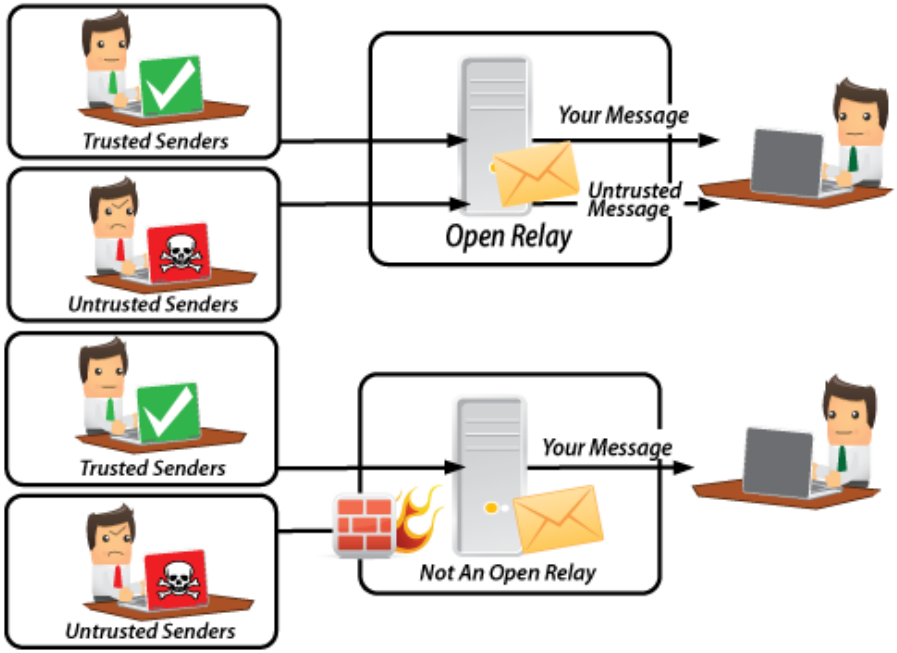
POP vs. IMAP


POP
POP Server
Clients independently download email. Once downloaded it is deleted from the server
POP Client
Mail folders
POP Client
Mail folders
IMAP
IMAP Server
Clients can talk to multiple servers
IMAP Server
Multiple clients can talk to the same account on the same server - from anywhere
Mail folders
Mail folders
IMAP Client
Mail folders
Client decides where to store messages. Copy them locally or leave them on the server
IMAP Client
Mail folders
Client has full control over their mail folders. Can create, delete, purge or move them around - even between different accounts on different servers

Relay

Um servidor de correio pode ser configurado de duas formas: com retransmissão ativada (*Open Relay*) ou desativada (*Not Open Relay*). No primeiro caso, qualquer usuário pode conectar-se ao servidor de correio e enviar mensagens, mesmo que ele não possua uma caixa postal ou autorização para tal. No segundo caso, somente usuários que possuem caixas postais ou autorização podem enviar mensagens.

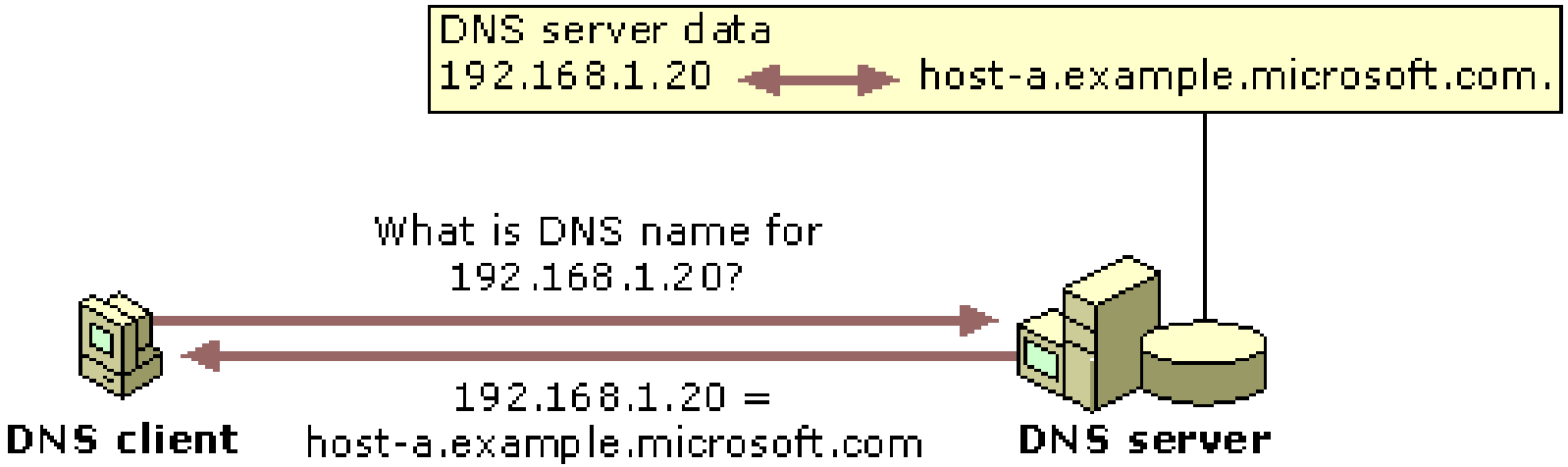
Vírus e *spammers* procuram usar servidores de correio que estejam operando no modo *Open Relay*.



DNS reverso

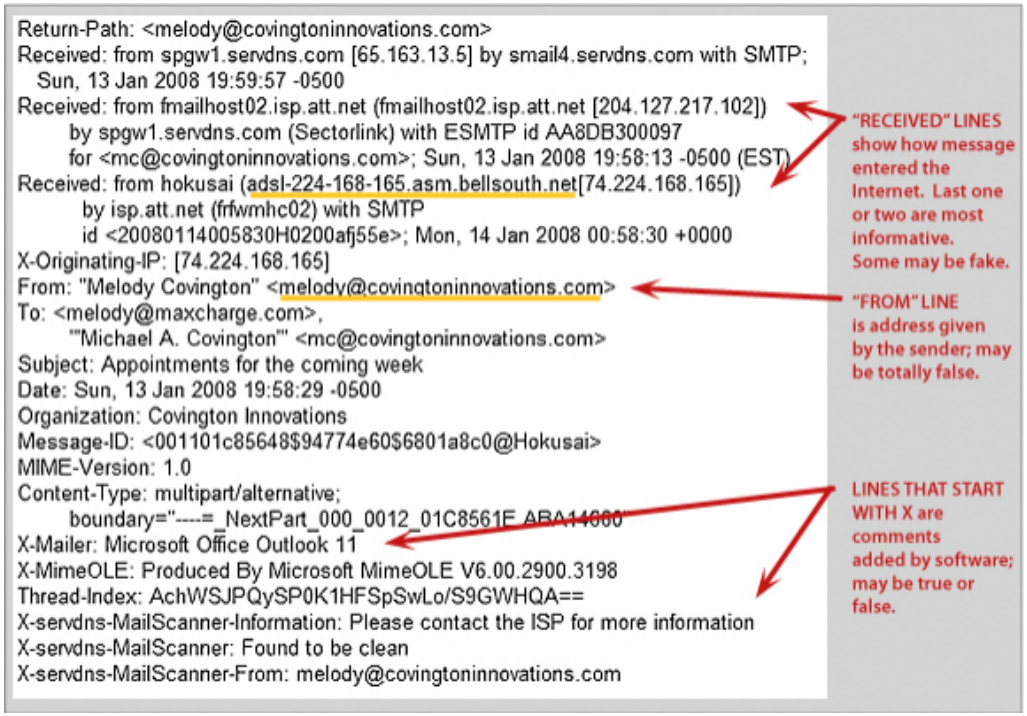
Ao contrário da pesquisa DNS direta, onde dado um nome de domínio retorna-se com um endereço IP, na pesquisa DNS reversa é dado um endereço IP e como resultado obtêm-se um nome de domínio.

Este recurso é bastante usado pelos servidores de correio para verificar se as mensagens recebidas provem de domínios reais ou forjados.



Cabeçalho da mensagem

Toda mensagem de correio possui um cabeçalho que contem informações importantes sobre remetente e destinatário, bem como por quais servidores a mensagem passou e por quais filtros (antivírus e *antispam*) foi submetida.



Para saber mais...

… acesse o documento sobre Noções básicas sobre a pesquisa inversa, da Microsoft.

… acesse o Analisador de Cabeçalho de e-mail, da MX Toolbox.

Módulo 8

Antispam

Introdução

O que é spam?

São e-mails não solicitados, que geralmente são enviados para um grande número de pessoas.

Quando o conteúdo é exclusivamente comercial, esse tipo de mensagem é chamada de UCE (Unsolicited Commercial E-mail).

Fonte: antispam.br

Introdução

O que são spam zombies?

São computadores de usuários finais que foram comprometidos por códigos maliciosos em geral, como *worms*, *bots*, vírus e cavalos de tróia.

Estes códigos maliciosos, uma vez instalados, permitem que *spammers* utilizem a máquina para o envio de spam, sem o conhecimento do usuário.

Enquanto utilizam máquinas comprometidas para executar suas atividades, dificultam a identificação da origem do spam e dos autores também.

Os spam *zombies* são muito explorados pelos *spammers*, por proporcionar o anonimato que tanto os protege.

Fonte: antispam.br

Introdução

Problemas causados pelo spam

O spam pode afetar os usuários do serviço de correio eletrônico de diversas formas, ameaçando assim a produtividade e a segurança, entre outros.

- **Não recebimento de e-mails:** Boa parte dos provedores de Internet limita o tamanho da caixa postal do usuário no seu servidor. Caso o número de spams recebidos seja grande, ele corre o risco de ter sua caixa postal lotada com mensagens não solicitadas. Se isto ocorrer, passará a não receber e-mails e, até que possa liberar espaço em sua caixa postal, todas as mensagens recebidas serão devolvidas ao remetente. Outro problema é quando o usuário deixa de receber e-mails nos casos em que regras antispam ineficientes são utilizadas, por exemplo, classificando como spam mensagens legítimas;
- **Gasto desnecessário de tempo:** Para cada spam recebido, o usuário necessita gastar um determinado tempo para ler, identificar o e-mail como spam e removê-lo da caixa postal;

Fonte: antispam.br

Introdução

- **Aumento de custos:** Independente do tipo de acesso à Internet utilizado, quem paga a conta pelo envio do spam é quem o recebe. Por exemplo, para um usuário que utiliza acesso discado à Internet, cada spam representa alguns segundos a mais de ligação que ele estará pagando;
- **Perda de produtividade:** Para quem usa o e-mail como ferramenta de trabalho, o recebimento de spams aumenta o tempo dedicado à tarefa de leitura de e-mails, além de existir a chance de mensagens importantes não serem lidas, serem apagadas por engano ou lidas com atraso;

Fonte: antispam.br

Introdução

- **Conteúdo impróprio ou ofensivo:** Como a maior parte dos spams é enviada para conjuntos aleatórios de endereços de e-mail, é bem provável que o usuário receba mensagens com conteúdo que julgue impróprio ou ofensivo;
- **Prejuízos financeiros causados por fraude:** O spam tem sido amplamente utilizado como veículo para disseminar esquemas fraudulentos, que tentam induzir o usuário a acessar páginas clonadas de instituições financeiras ou a instalar programas maliciosos, projetados para furtar dados pessoais e financeiros. Esse tipo de spam é conhecido como *phishing/scam*. O usuário pode sofrer grandes prejuízos financeiros, caso forneça as informações ou execute as instruções solicitadas nesse tipo de mensagem fraudulenta.

Fonte: antispam.br

Introdução

Como identificar?

Para identificar um spam, as seguintes características devem ser observadas:

- **Cabeçalhos suspeitos:** O cabeçalho do e-mail aparece incompleto, sem o remetente ou o destinatário. Ambos podem aparecer como apelidos ou nomes genéricos, tais como: amigo@, suporte@ etc. A omissão do destinatário é um dos casos mais comuns, pois, os *spammers* colocam listas enormes de e-mails no campo reservado para Cópias Carbono Ocultas ou Blind Carbon Copies (Cco: ou Bcc:), já que tais campos não são mostrados ao usuário que recebe a mensagem;

Fonte: antispam.br

Introdução

- **Campo Assunto (Subject) suspeito:** O campo reservado para o assunto do e-mail (subject) é uma armadilha para os usuários e um artifício poderoso para os *spammers*. A maioria dos filtros antispam está preparada para barrar e-mails com diversos assuntos considerados suspeitos. No entanto, os *spammers* adaptam-se e tentam enganar os filtros colocando no campo assunto conteúdos enganosos, tais como: vi@gra (em vez de viagra) etc. Como os *spammers* utilizam esses subterfúgios, alguns e-mails suspeitos podem não ser identificados e, nesse momento, os usuários devem estar atentos para não abrir e-mails de spam, executar arquivos em anexo e ter sua máquina contaminada por um código malicioso;

Fonte: antispam.br

Introdução

- **Opções para sair da lista de divulgação:** Existem spams que tentam justificar o abuso, alegando que é possível sair da lista de divulgação, “clicando” no endereço anexo ao e-mail. O usuário deve verificar se fez realmente o cadastro na lista em questão. Se não tiver certeza, melhor ignorar o e-mail, afinal, um dos artifícios usados pelos *spammers* para validar a existência dos endereços de e-mail é justamente solicitar a confirmação. Também é importante jamais clicar em um link enviado por e-mail. Sempre digite a URL no navegador;
- **E-mails enviados “uma única vez”:** Embora seja um dos recursos mais antigos, entre aqueles utilizados pelos *spammers*, ainda são encontrados e-mails de spam alegando que serão enviados “uma única vez”. Essa é uma característica de e-mail de spam;

Fonte: antispam.br

Introdução

- **Sugestão para apenas remover:** Uma das mais frequentes e piores desculpas usadas pelos *spammers* é alegar que se o usuário não tem interesse no e-mail não solicitado, basta “removê-lo”. Essa é uma característica de e-mail de spam;
- **Leis e regulamentações:** Não existem regulamentações brasileiras referentes à prática de spam. Portanto, citações que envolvam leis e regulamentações são características de e-mail de spam;

Fonte: antispam.br

Introdução

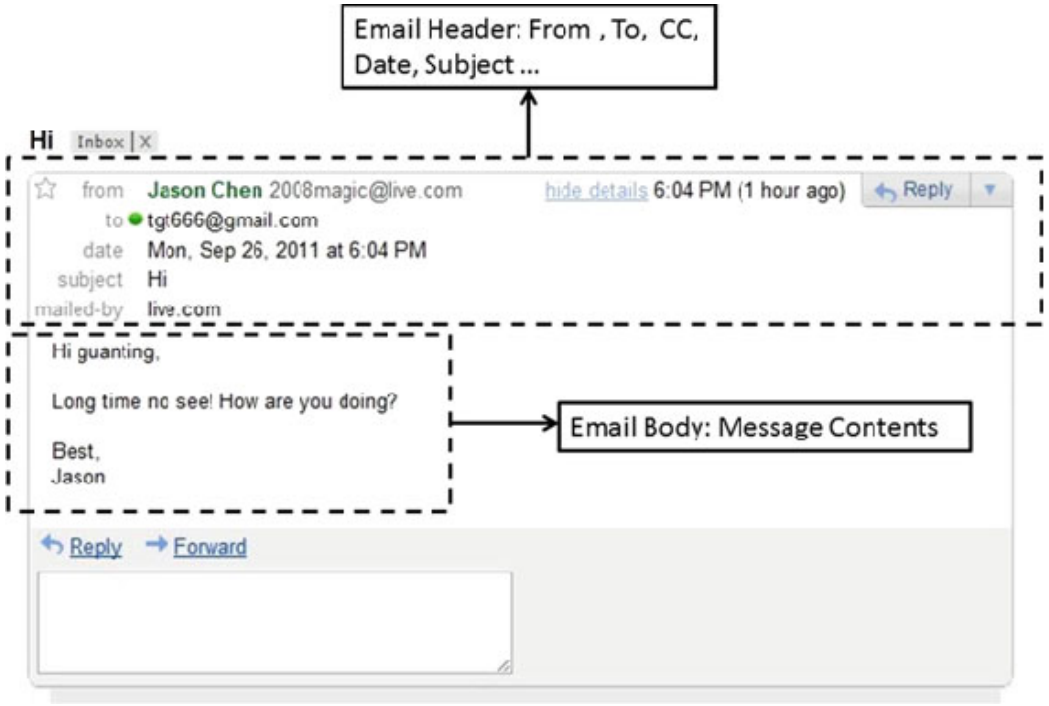
- **Correntes, boatos e lendas urbanas:** São características de spam os e-mails contendo: textos pedindo ajuda financeira, contando histórias assustadoras, pedindo para que sejam enviados “a todos que você ama” ou, ainda, ameaçando que algo acontecerá caso não seja repassado a um determinado número de pessoas;
- **Golpes e fraudes:** Com certa frequência, os e-mails de spam são portadores de fraudes e golpes disseminados na rede. Alguns exemplos são: e-mails de promoções e e-mails de instituições financeiras ou governamentais. Nesses casos, a melhor defesa é a informação. Conhecer os tipos de golpes e como eles podem chegar até a sua caixa postal é a melhor estratégia de defesa.

Fonte: antispam.br

Estrutura de uma mensagem

Uma mensagem de correio eletrônico pode ser dividida em três partes:

- o envelope;
- o cabeçalho;
- o corpo.



Fonte: antispam.br

Estrutura de uma mensagem
Envelope

O envelope contém as informações necessárias para que o MTA que recebe uma mensagem saiba o que fazer com ela e para quem retornar esta mensagem em caso de erro. No protocolo SMTP o envelope é construído a partir dos comandos **MAIL FROM** e **RCPT TO**.

O **MAIL FROM** indica um endereço para onde deve ser enviada uma mensagem de erro em caso de necessidade. Mensagens de erro possuem MAIL FROM vazio, indicando que não deve haver caminho de volta. Isto evita que sejam enviadas mensagens de erro sobre outras mensagens de erro.

O **RCPT TO** indica o destinatário da mensagem, podendo ser apenas um ou vários. Estes destinatários podem ser locais ou não. No caso de destinatários locais o MTA chama um MDA para efetivamente entregar a mensagem. No caso de destinatários remotos o MTA age como cliente, retransmitindo a mensagem para o MTA apropriado.

Fonte: antispam.br

Estrutura de uma mensagem
Cabeçalho

O cabeçalho de uma mensagem é composto de diversos campos. Estes campos contém informações tanto para o MTA quanto para o MUA, e podem ser inseridos na mensagem pelo MUA e pelos vários MTAs através dos quais a mensagem passou.

Os campos mais importantes no gerenciamento de problemas relativos a spam são:

From: designa o remetente nominal da mensagem, que não é necessariamente igual ao que aparece no envelope ou no campo Return-Path.

Exemplo:

```
From: "Fulano de Tal" fulano@example.com
```

Cuidado: Os campos **From** e **To** são nominais, o que significa que podem não refletir a real origem ou destino da mensagem. Como os MUAs costumam não exibir todos os campos do cabeçalho, o usuário pode ser levado a crer que eles realmente refletem o remetente e destinatário da mensagem.

Fonte: antispam.br

Estrutura de uma mensagem
Cabeçalho

To: / Cc: / Bcc: designam os destinatários que não necessariamente coincidem com os declarados no envelope.

Exemplo:

```
To: "Sicrano dos Anzóis" <sicrano@another.example.com>
Cc: Fidélis Teles de Meireles <fidelis@example.com>
```

Cuidado: Os campos **From** e **To** são nominais, o que significa que podem não refletir a real origem ou destino da mensagem. Como os MUAs costumam não exibir todos os campos do cabeçalho, o usuário pode ser levado a crer que eles realmente refletem o remetente e destinatário da mensagem.

Fonte: antispam.br

Estrutura de uma mensagem
Cabeçalho

Return-Path: geralmente copiado do envelope (MAIL FROM), é o endereço para onde retornar recibos de devolução (bounces).

Exemplo:

```
Return-Path: <fulano@example.com>mensagem
```

Este campo é muito útil para campanhas de marketing, por exemplo, pois o remetente da mensagem pode não desejar ter de lidar com dezenas, centenas ou milhares de respostas de erro a um e-mail enviado em massa.

Fonte: antispam.br

Estrutura de uma mensagem
Cabeçalho

Received: indica a procedência (pelo endereço IP), a data e a hora em que a mensagem foi recebida e, eventualmente, a auto identificação do transmissor (HELO/EHLO). Pelos vários campos Received presentes em um cabeçalho é possível verificar o caminho que a mensagem percorreu, porém só é realmente confiável o Received mais recente, pois ele foi inserido pelo MTA que está sob o controle do administrador.

Exemplo:

```
Received: from pulsar.example.com (192.0.2.38) by
quasar.example.com with SMTP; 5 Sep 2005 15:55:29 -0000
```

Neste caso a mensagem foi recebida de 192.0.2.38 às 18:55:29 no horário local de São Paulo, pois -0000 indica que a hora é UTC*.

*A sigla UTC é uma combinação do inglês Coordinated Universal Time (CUT) e do francês Temps Universel Coordonné (TUC) e significa Tempo Universal Coordenado.

Fonte: antispam.br

Estrutura de uma mensagem
Corpo da mensagem

O corpo da mensagem contém seu texto e anexos, se houverem. O formato e a codificação do corpo são descritos pelos campos:

- Content-type;
- Content-Transfer-Encoding;
- MIME-Type.

O corpo da mensagem é separado do cabeçalho por uma linha em branco.

MTAs não lidam com o corpo da mensagem. Alguns programas auxiliares de MTAs, entretanto, podem processar o corpo das mensagens, tais como antivírus e antispam baseados em análise de conteúdo.

O campo **MIME-Type** é usado para o envio de mensagens complexas.

Fonte: antispam.br

Funcionamento do correio eletrônico Conceitos

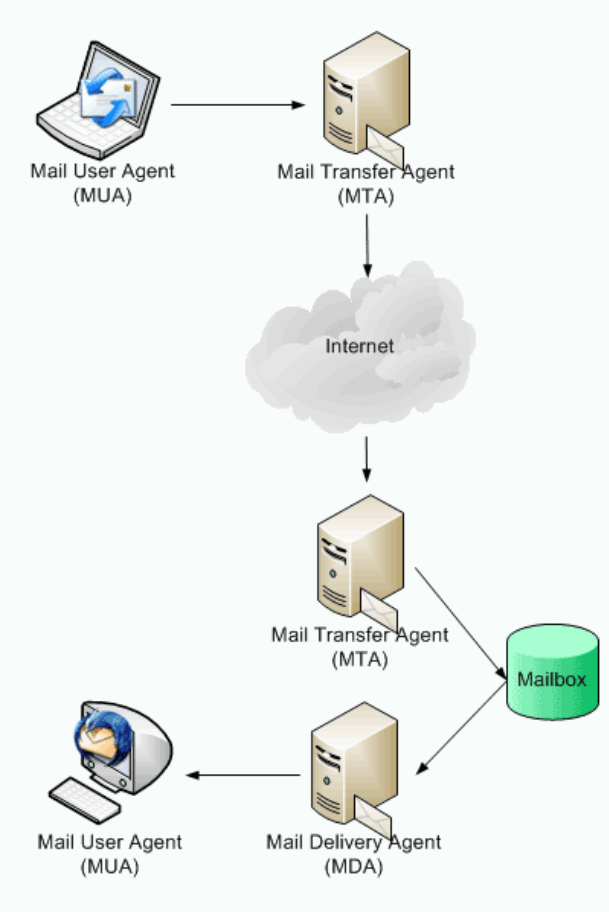
Caixa postal: é um arquivo ou diretório onde as mensagens são recebidas;

MUA (Mail User Agent): é uma aplicação ou programa utilizado diretamente pelo usuário para compor, enviar e ler mensagens. Exemplos de MUAs são: Outlook, Pine, Mutt, Mozilla Thunderbird, etc;

MTA (Mail Tranfer Agent): é uma aplicação responsável por passar mensagens para outros MTAs ou para um MDA, se o destino da mensagem for respectivamente remoto ou local. Há vários MTAs, por exemplo: Sendmail, Qmail, Exim e Postfix;

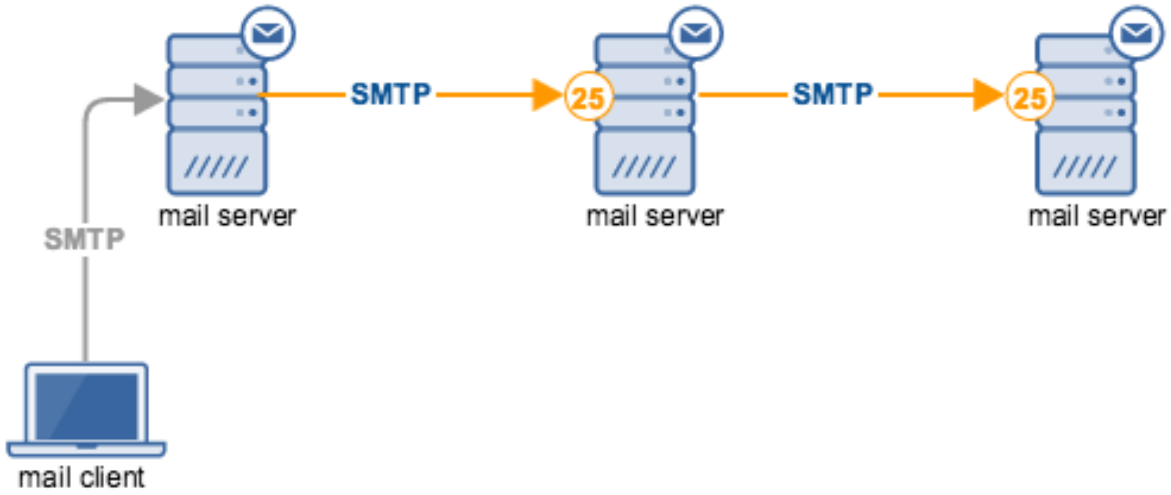
MDA (Mail Delivery Agent): é uma aplicação responsável por entregar mensagens em caixas postais. Um exemplo de MDA é o Procmail.

Fonte: antispam.br



Funcionamento do correio eletrônico
O caminho de uma mensagem

Normalmente a mensagem é composta pelo remetente em seu MUA, e submetida ao MTA corporativo ou do provedor de acesso usando o protocolo SMTP, na porta 25/TCP, que então transmite a mensagem para o MTA de destino.

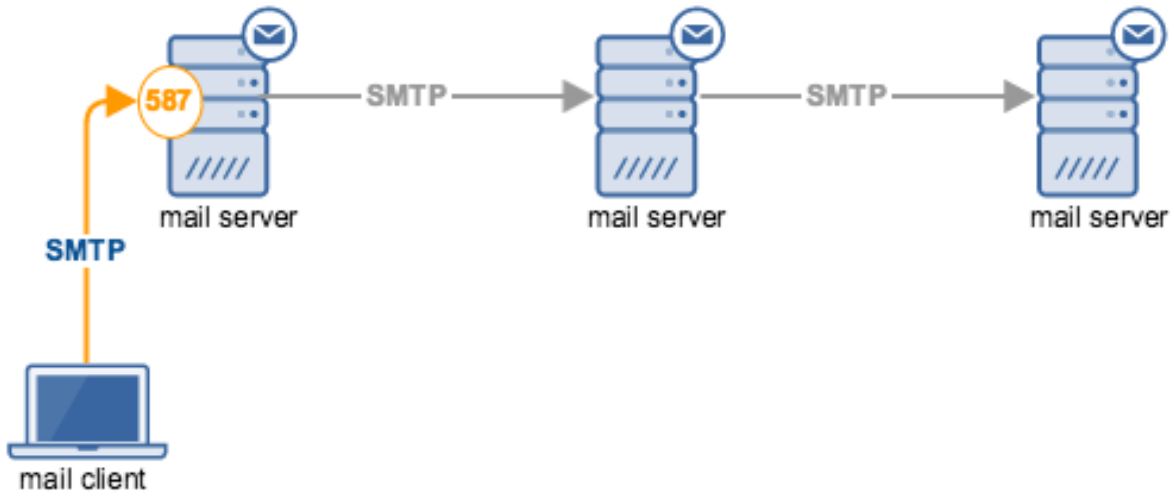


Fonte: antispam.br

Funcionamento do correio eletrônico
O caminho de uma mensagem

Há uma tendência de mudar esse processo de submissão para a porta 587/TCP, com SMTP autenticado.

Alternativamente, para proteger as credenciais de autenticação, pode-se usar a porta 465/TCP sob sessão segura (TLS).



Fonte: antispam.br

Funcionamento do correio eletrônico
O caminho de uma mensagem

Teoricamente a mensagem poderia ser enviada diretamente ao MTA de destino, mas é aconselhável fazê-lo por meio de um transmissor.

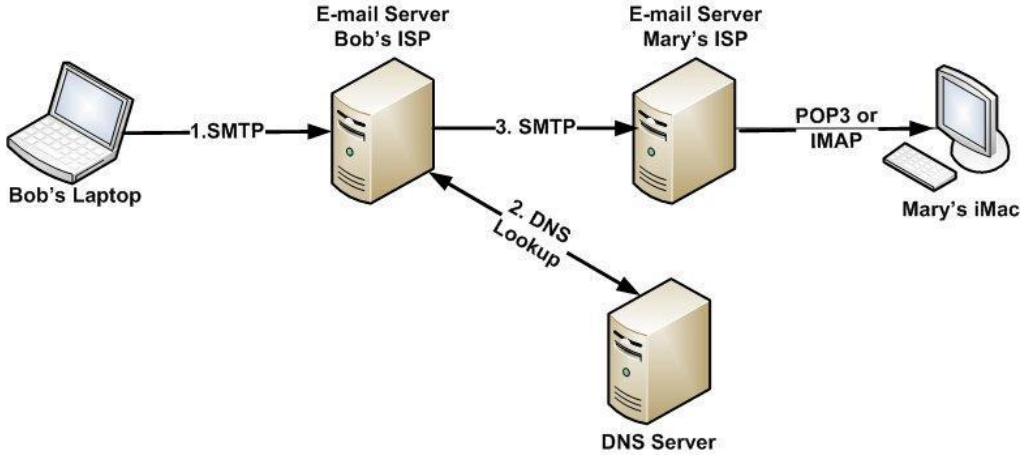
Isto é importante, pois caso ocorra um erro temporário ou o MTA de destino não esteja disponível, este MTA transmissor pode colocar a mensagem em uma fila e tentar retransmiti-la mais tarde.

Com isso também é possível desligar ou desconectar a estação do usuário da rede sem que a mensagem se perca.

Fonte: antispam.br

Funcionamento do correio eletrônico
O caminho de uma mensagem

O MTA transmissor descobre o endereço IP do MTA de destino pelo DNS, consultando o registro MX para o domínio do destinatário. Na falta do MX um registro A ou A6 pode ser utilizado. Os registros MX são ordenados por preferência e as tentativas de conexão via SMTP pela porta 25/TCP ocorrem de acordo com esta ordem. Caso não seja possível enviar a mensagem para os MTAs designados pelo DNS a mensagem é mantida em uma fila e uma nova tentativa de transmissão é programada para ser feita posteriormente. Caso o tempo limite para novas tentativas seja atingido a mensagem é devolvida ao remetente. Estes tempos são configurados pelo administrador.

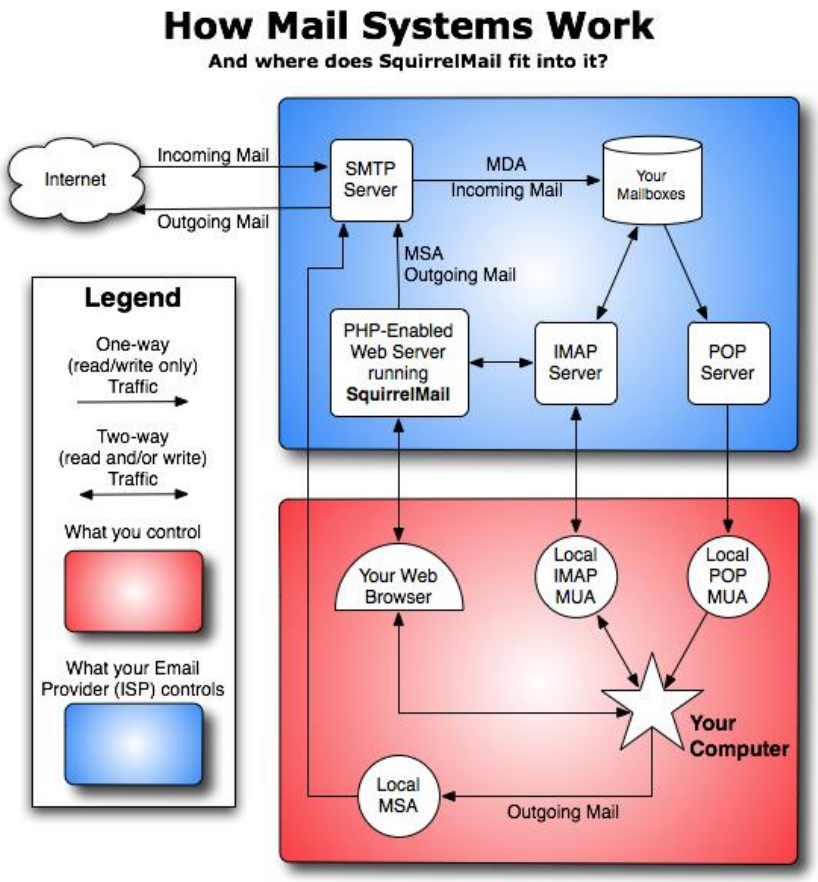


Fonte: antispam.br

Funcionamento do correio eletrônico
O caminho de uma mensagem

O MTA do destino recebe uma conexão via SMTP e verifica se a mensagem se destina a um usuário local ou não. Se o destinatário não for local e a transmissão for válida o MTA se comporta exatamente da mesma forma como o transmissor descrito acima. Se o destinatário for local o MTA repassa a mensagem para o MDA, que realiza a entrega.

O MDA entrega a mensagem na caixa postal do destinatário, normalmente um arquivo ou um diretório. O MDA também pode executar algumas ações pré-programadas em função do cabeçalho ou do corpo da mensagem.



Fonte: antispam.br

Funcionamento do correio eletrônico
O caminho de uma mensagem

Finalmente o destinatário da mensagem a recebe, podendo visualizá-la em seu MUA. Esse acesso pode ser feito de vários modos:

- a uma caixa postal local;
- através do protocolo POP (Post Office Protocol), em texto claro (110/TCP) ou cifrado (995/TCP);
- através do protocolo IMAP (Interactive Mail Access Protocol), em texto claro (143/TCP) ou cifrado (993/TCP).

Fonte: antispam.br

Técnicas de envio de spam

Os *spammers* utilizam diversas técnicas que procuram subverter o caminho da mensagem em algum ponto, assim como existem códigos que também enviam mensagens em grandes quantidades. Alguns dos mais importantes mecanismos de envio destes e-mails são:

- **Programas de envio de e-mail em massa:** também conhecidos como programas de *bulk mailing* ou *mass mailing*, são programas especialmente concebidos para entrega de e-mails em massa. Estes programas são fáceis de obter e podem ser configurados para enviar e-mails através de máquinas com proxies abertos. Estes proxies, em geral, são máquinas mal configuradas ou estão instalados em máquinas que se tornaram *spam zombies*. Os próprios criadores dos programas de envio em massa comumente vendem algum serviço de fornecimento de endereços de máquinas com proxies abertos;
- **Spam zombies:** são computadores de usuários finais que foram comprometidos por códigos maliciosos em geral, como *worms*, *bots*, vírus e cavalos de tróia. Estes códigos maliciosos, uma vez instalados, permitem que *spammers* utilizem a máquina para o envio de spam, sem o conhecimento do usuário;

Fonte: antispam.br

Técnicas de envio de spam

- **Vírus propagados por e-mail:** normalmente são recebidos como um arquivo anexado à uma mensagem de correio eletrônico. O conteúdo dessa mensagem procura induzir o usuário a clicar sobre o arquivo anexado, fazendo com que o vírus seja executado. Quando este tipo de vírus entra em ação, ele infecta arquivos e programas e envia cópias de si mesmo para os contatos encontrados nas listas de endereços de e-mail armazenadas no computador do usuário – estes endereços podem ser utilizados tanto como remetentes quanto como destinatários;
- **Uso de sites comprometidos:** alguns *spammers* utilizam servidores comprometidos para enviar spam. Uma prática comum é a inclusão de alguma página Web especial para o envio de spam, contendo scripts ou formulários para envio de e-mail;

Fonte: antispam.br

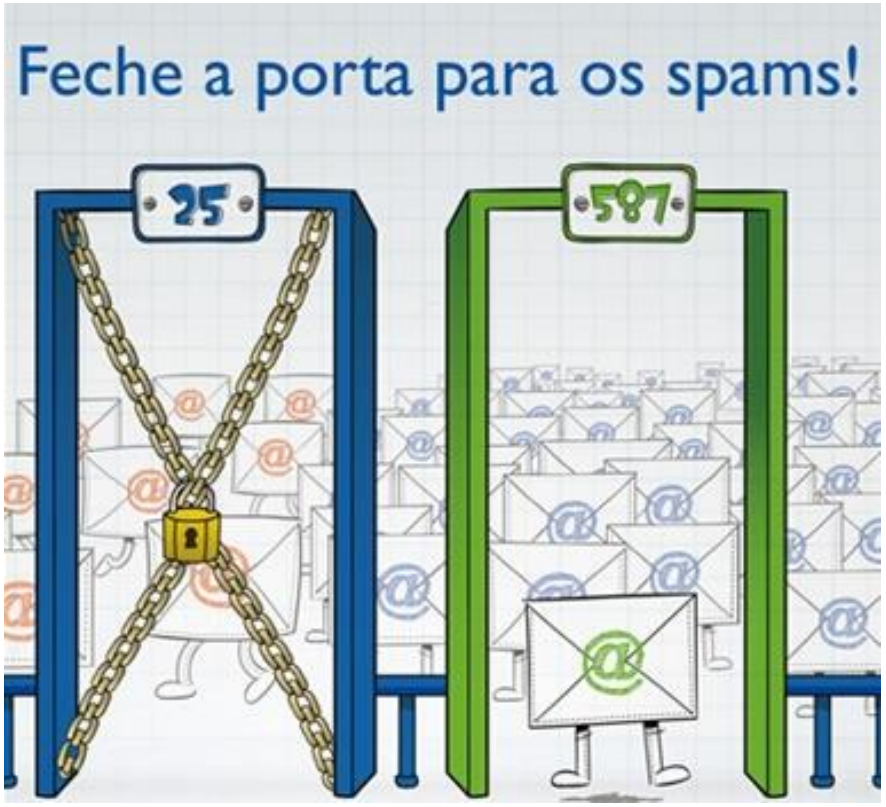
Técnicas de envio de spam

- **Abuso de formulários e scripts na Web:** muitos serviços Web tem algum tipo de transmissão do conteúdo de formulários por e-mail, sendo que tal funcionalidade pode ser abusada para a transmissão de spam. Um caso bem conhecido é o script CGI FormMail.pl, que envia o e-mail utilizando-se de informações providas no formulário e que pode ser abusado para prover anonimato para o *spammer*. Outro exemplo de funcionalidade que pode ser abusada é a função mail() da linguagem PHP e seus similares em outras linguagens. Spams enviados a partir de servidores Web mal configurados são dificilmente contidos pelas práticas atuais de contenção de spam, sendo muito importante que administradores de serviços Web evitem que seus servidores sejam abusados por *spammers*.

Fonte: antispam.br

Gerência da porta 25

O termo “Gerência da Porta 25” é utilizado para denominar o conjunto de políticas e padrões que podem ser utilizados em redes de usuários finais ou de caráter residencial, e que procura separar as funcionalidades de submissão de mensagens daquelas de transporte de mensagens entre servidores.



Os principais objetivos são:

- a mitigação do abuso de proxies abertos e máquinas infectadas para o envio de spam;
- aumentar a rastreabilidade de fraudadores e *spammers*.

Fonte: antispam.br

Gerência da porta 25

A definição do padrão para o protocolo de submissão é de 1998, sendo sua última revisão de 2011, no documento “RFC 6409: Message Submission for Mail”.

Este protocolo, chamado de “Message Submission”, fornece um meio para distinguir uma submissão do transporte de mensagens, permitindo assim:

- a aplicação de políticas diferentes para cada tipo de conexão, impedindo *relays* não autorizados ou introdução de e-mails não solicitados;
- a implementação de autenticação na submissão, incluindo aquela realizada remotamente por usuários autorizados;
- a possibilidade de implementar, futuramente, melhorias no serviço de submissão.

Fonte: antispam.br

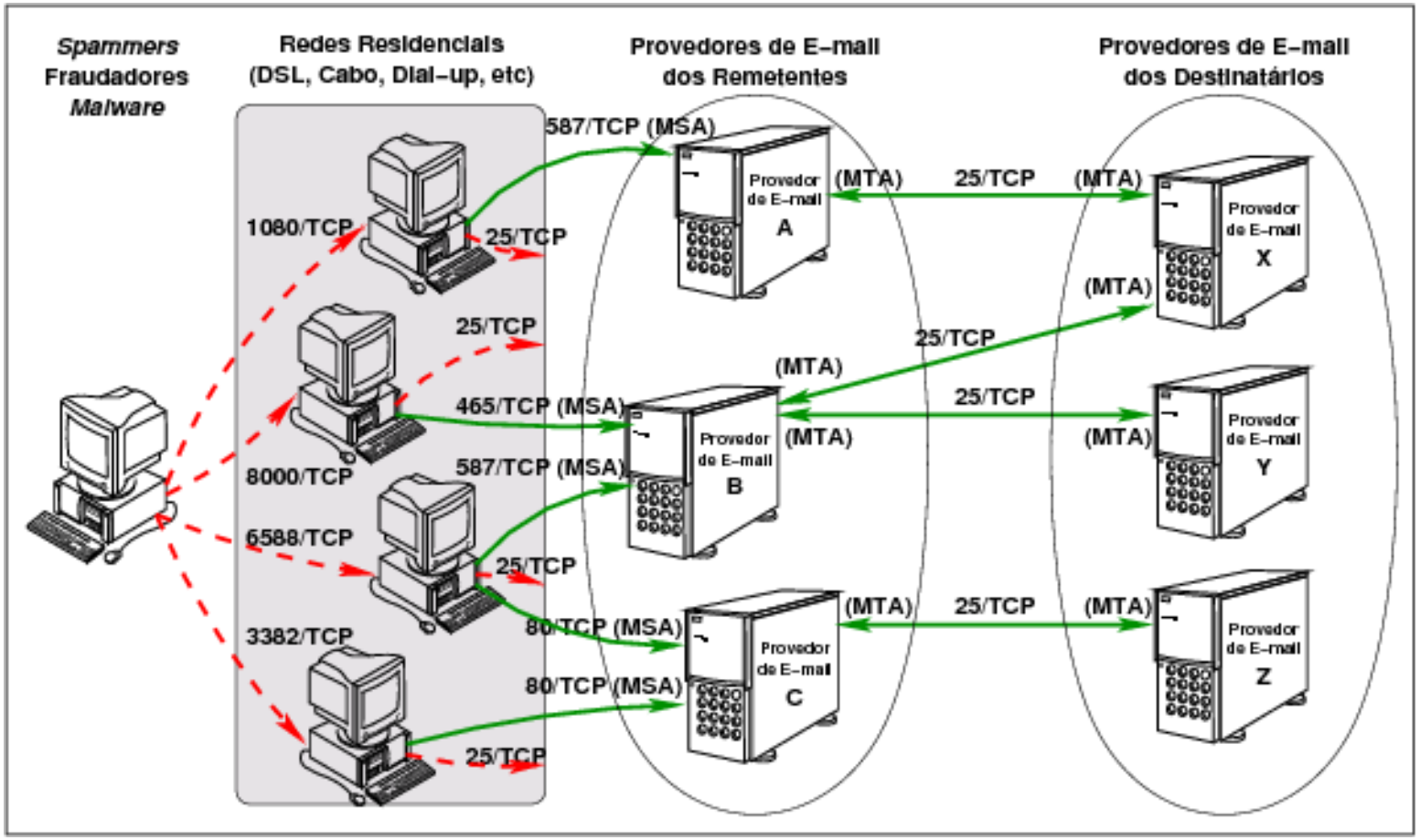
Gerência da porta 25

O Messaging Anti-Abuse Working Group (MAAWG) recomenda para redes de caráter residencial, além da adoção de Message Submission, as seguintes medidas:

- requerer autenticação para a submissão de mensagens, como recomendado na RFC 4954;
- configurar o software cliente de e-mail para usar a porta 587/TCP e autenticação;
- bloquear acesso de saída para porta 25/TCP a partir de todas as máquinas que não sejam MTAs ou explicitamente autorizadas.
- não interferir no tráfego para a porta 587/TCP;

Fonte: antispam.br

Gerência da porta 25



Fonte: antispam.br

Técnicas de bloqueio de spam
Listas de bloqueio

Listas de bloqueio são, talvez, o mais antigo mecanismo de combate ao spam. Estas listas são bases de dados de endereços IP que tenham sido identificados como possível fonte de spam, segundo os critérios da entidade que mantém a lista. As listas normalmente funcionam através de consultas DNS às bases de dados.

Também existem outros critérios de bloqueio, geralmente envolvendo DNS, como:

- bloqueio pela inexistência de reverso;
- bloqueio pela inconsistência do reverso;
- bloqueio pela presença do reverso em uma lista de bloqueio de domínios.

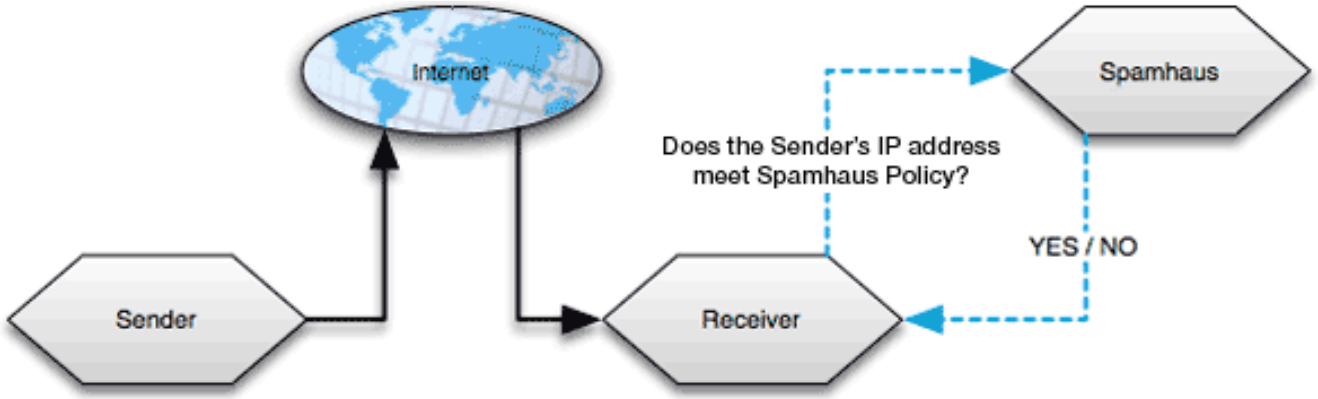
Fonte: antispam.br

Técnicas de bloqueio de spam
Listas de bloqueio

Listas de bloqueio

As listas de bloqueio possuem endereços IP de máquinas que, segundo o critério do mantenedor da lista, estão envolvidos em envio de spam.

Estas listas são implementadas através de zonas de DNS, semelhantes às de tradução reversa. Dado um endereço IP w.x.y.z a ser bloqueado, na lista este IP será incluído com o nome de domínio z.y.x.w.nome.da.lista.



Fonte: antispam.br

Técnicas de bloqueio de spam
Listas de bloqueio

Listas de linhas discadas

São listas que enumeram os domínios reversos de redes dedicadas somente ao acesso doméstico, seja propriamente discado ou de banda larga.

O funcionamento é semelhante às listas de bloqueio, mas neste caso o domínio consultado é o reverso clássico (in-addr.arpa). Se o domínio resultante for subdomínio de qualquer um listado, a mensagem é rejeitada.

Vários operadores de ADSL não fazem separação de seus usuários domésticos dos corporativos.

Fonte: antispam.br

Técnicas de bloqueio de spam
Listas de bloqueio

Listas de relays e proxies abertos

Relays abertos são MTAs que transmitem mensagens de qualquer domínio, ou mesmo só de domínios determinados, para qualquer outro, sem pedir autenticação, sem restringir (ou restringindo muito pouco) a faixa de endereços IP de origem. Relays abertos podem ser MTAs mal configurados ou programas instalados clandestinamente em máquinas comprometidas.

Proxies abertos usam um mecanismo diferente, mas com o mesmo efeito. Em vez de um MTA ser abusado, é um serviço de proxy que é abusado para retransmitir mensagens, por exemplo, através do comando CONNECT.

As listas de relays e proxies abertos funcionam exatamente como as listas de bloqueio convencionais. Algumas listas de bloqueio mais gerais retornam valores indicativos de que o IP foi incluído lá por ser um relay ou proxy aberto.

Muitas vezes o responsável por um MTA corrigiu o problema de relay aberto mas o endereço permanece listado por um longo tempo.

Fonte: antispam.br

Técnicas de bloqueio de spam
Listas de bloqueio

Lista de permissão

As listas de permissão consistem em uma lista de exceções às regras de bloqueio definidas por listas de bloqueio ou outros critérios. Normalmente a lista de permissão é mantida pelo próprio administrador do serviço de e-mail, e pode ser implementada através de DNS, listas de domínios, IPs ou blocos CIDR, ou através de regras de SPF que devem ser avaliadas antes de qualquer outra.

Por exemplo, usando SPF, se quiséssemos que qualquer IP da rede 192.0.2.0/24 pudesse enviar mensagens independentemente de registros SPF (ou a falta deles), ou mesmo que constem de alguma lista de bloqueio, bastaria incluir um registro SPF contendo ip4:192.0.2.0/24 para colocá-lo na lista de liberação.

Fonte: antispam.br

Técnicas de bloqueio de spam
Listas de bloqueio

Checagem de informações de DNS

Um método de bloqueio utilizado por alguns administradores de redes é impedir o recebimento de mensagens partindo de máquinas cujo endereço IP não possui um registro DNS do tipo PTR (endereço reverso). Adicionalmente também é possível verificar se o nome da máquina, retornado pela consulta PTR, possui um registro do tipo A que seja igual ao endereço IP originalmente consultado.

Há duas considerações que devem ser feitas sobre o bloqueio em função do endereço reverso de máquinas de usuários domésticos:

- este depende de convenções adotadas pelo provedor de serviços da rede de origem. Estas convenções podem mudar sem aviso prévio e não são uniformes entre diversos provedores;
- pode fazer com que MTAs válidos sejam bloqueados.

Possuir endereço reverso não é obrigatório, além disso, nem sempre os administradores de redes possuem controle direto sobre a configuração DNS de seus IPs. Desse modo, bloqueio em função do endereço reverso deve ser usado com cautela.

Fonte: antispam.br

Técnicas de bloqueio de spam
Listas de bloqueio

Checagem de aderência a RFCs

Também é possível barrar mensagens em função de alguns testes adicionais de DNS e de sua conformidade com padrões, tais como:

- **HELO/EHLO:** se a identificação fornecida nesse comando não estiver de acordo com a RFC 5321 ou não for possível resolver o nome do domínio, a mensagem é bloqueada;
- **MAIL FROM:** se o domínio do e-mail fornecido não existir, a mensagem é bloqueada.

É importante não bloquear mensagens em que o MAIL FROM é vazio, pois podem ser mensagens de erro, que não devem ser bloqueadas. No entanto, uma situação em que o bloqueio por MAIL FROM vazio é aplicável é quando a mensagem tiver mais de um destinatário, pois mensagens de erro possuem apenas um destinatário.

Fonte: antispam.br

Técnicas de bloqueio de spam
Filtros de conteúdo

Existem algumas técnicas de bloqueio de spam que se baseiam na análise do conteúdo da mensagem. Em geral, são filtros baseados no reconhecimento de padrões do conteúdo que buscam identificar se o e-mail pode conter um vírus ou se tem características comuns aos spams.

Tais filtros podem ser usados em conjunto com o MTA, com o MDA ou ainda com o MUA, de acordo com as preferências pessoais do usuário.

Considerando apenas os casos em que os filtros são usados em conjunto com o MTA, uma vez que uma mensagem foi identificada como contaminada ou como provável spam, há três possibilidades:

- ser recusada pelo MTA, com um código de erro “550 5.7.1 Message content rejected”;
- ser aceita pelo MTA, porém desviada para uma área especial de quarentena;
- ser aceita pelo MTA e encaminhada para o destinatário, porém com um campo do cabeçalho marcando a mensagem como suspeita.

Os filtros de conteúdo mais comuns são os antivírus e os identificadores Bayesianos de spam.

Fonte: antispam.br

Técnicas de bloqueio de spam
Filtros de conteúdo

Antivírus

Existem no mercado diversas opções de antivírus que podem ser utilizados em conjunto com MTAs, sendo que algumas destas opções são gratuitas. A maioria possui mecanismos de atualização automática, já que a criação de novos vírus é bastante intensa e exige atualizações diárias, ou até mesmo mais frequentes, das assinaturas dos antivírus.

Os programas antivírus não lidam diretamente com arquivos comprimidos ou no formato usual dos *e-mails*. Deste modo, antes do conteúdo da mensagem ser analisado pelo antivírus é necessário desmontar a mensagem e possivelmente descomprimir os anexos.

Devido ao trabalho de desmontagem da mensagem e depois o de reconhecimento de padrões, o uso de antivírus em conjunto com MTAs costuma implicar em altos consumos de CPU e memória do servidor. Deste modo, aconselha-se submeter as mensagens ao antivírus somente depois de terem sido avaliadas por outras técnicas.

Fonte: antispam.br

Técnicas de bloqueio de spam
Filtros de conteúdo

Filtros Bayesianos Antispam (heurística)

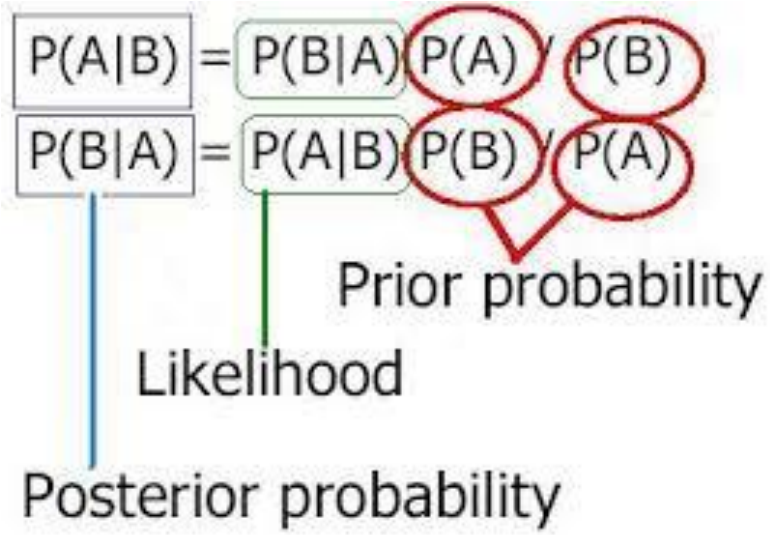
Os filtros Bayesianos implementam um algoritmo de probabilidade baseado na Teoria de Bayes. Os programas que utilizam filtros Bayesianos devem passar por um período inicial de treinamento, no qual tratam conjuntos de mensagens legítimas e também mensagens que conhecidamente são spam, criando uma base de dados inicial com informações sobre as ocorrências de palavras em cada um dos casos. Após este período, o programa passa a avaliar as mensagens considerando as ocorrências de palavras e então classifica cada e-mail de acordo com a probabilidade de ser spam ou não, tomando como base o treinamento inicial. Também é possível que este processo de treinamento seja continuado com novas mensagens.

Para que o filtro se adapte ao caráter mutável do spam é necessário que o treinamento do filtro seja contínuo, com a identificação dos spams que não foram classificados e das mensagens que não são spam e que foram rotuladas como tal.

Fonte: antispam.br

Técnicas de bloqueio de spam
Filtros de conteúdo

Como os filtros Bayesianos podem acarretar falsos positivos, é aconselhável não descartar uma mensagem marcada como spam, mas sim optar por colocá-la em quarentena. Esse problema pode ser agravado caso a base de dados com que ele toma decisão for desatualizada ou baseada em outro idioma.



Fonte: antispam.br

Técnicas de bloqueio de spam
Filtros de conteúdo

Bloqueio de anexos

Como muitos cavalos de tróia e vírus que afetam sistemas Windows são enviados, por exemplo, em arquivos executáveis (.exe) ou associados a certos aplicativos, como *screen savers* (.scr), alguns administradores procuram bloquear mensagens com determinados arquivos anexados.

O bloqueio pode ser feito com base no tipo ou no nome do arquivo, informações que podem ser obtidas no cabeçalho MIME. Os tipos dos anexos são dados pelo campo **Content-Type** e os nomes dos arquivos pelo atributo **name** deste campo.

Na prática, no entanto, esta técnica pode bloquear anexos que não são maliciosos, mas que estão entre os tipos proibidos, e pode deixar passar anexos que aparentemente não são hostis, como é o caso de imagens que exploram falhas no software usado para exibi-las.

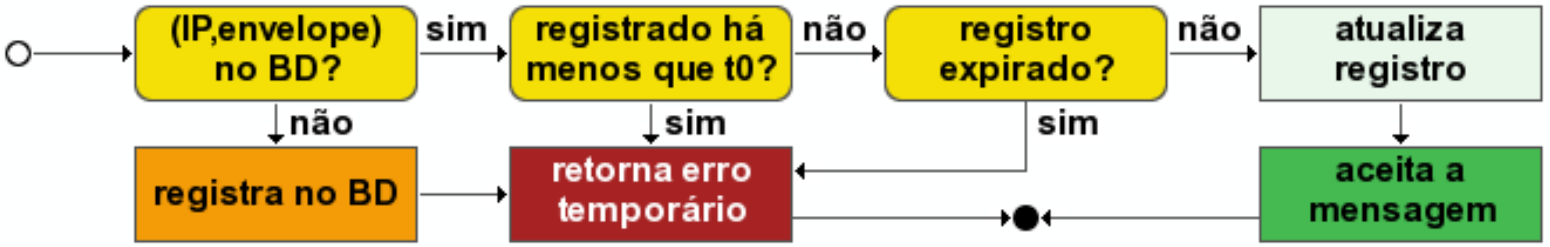
Fonte: antispam.br

Técnicas de bloqueio de spam
Greylist

O conceito de *greylisting* consiste em recusar temporariamente uma mensagem e esperar por sua retransmissão, e parte dos seguintes princípios:

- que e-mails válidos são enviados a partir de MTAs legítimos, que mantém filas e possuem políticas de retransmissão em caso de erros temporários;
- *spammers* e códigos maliciosos raramente usam MTAs legítimos.

Contudo, existem *spammers* que utilizam MTAs legítimos ou mesmo reenviam as mensagens a fim de contornar esta técnica. Ainda assim, o *greylisting* tem se mostrado eficiente para barrar mensagens enviadas por vírus, *worms* e *spam zombies*.



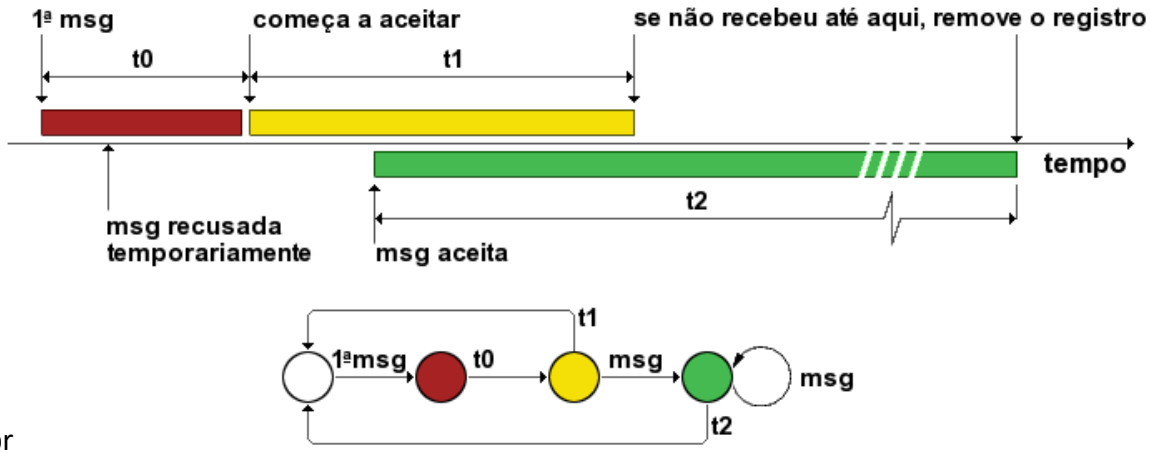
Fonte: antispam.br

Técnicas de bloqueio de spam
Greylist

Em geral as implementações de *greylisting* mantém um banco de dados com registros indexados por:

- endereço IP da origem;
- endereço do remetente no envelope;
- endereço do destinatário no envelope.

Estes registros contém uma variável de estado e alguns parâmetros de tempo. Os estados de um autômato de *greylisting* são assim representados:



Fonte: antispam.br

Técnicas de bloqueio de spam
Greylist

Registro de *greylisting*

- **inicial (branco):** (IP, envelope) não estão registrados no banco de dados, a primeira mensagem recebida faz passar para o estado de bloqueio temporário e é rejeitada com erro temporário;
- **bloqueio temporário (vermelho):** rejeita mensagens com erro temporário e sai deste estado para o de aceitação temporária somente depois de decorrido o tempo t0;
- **aceitação temporária (amarelo):** aceita uma mensagem, passando para o estado transparente ou, se não receber mensagem até que decorra t1, volta para o estado inicial;
- **transparente (verde):** aceita mensagens e retorna para este estado ou volta para o estado inicial se decorrer o tempo t2 sem que qualquer mensagem seja recebida. A cada nova mensagem a contagem de t2 é reiniciada.

Assim, IPs e envelopes com os quais se mantém correspondência regular atingirão o estado transparente rapidamente. Para estes será como se não estivesse sendo usado *greylisting*.

É importante também manter em uma lista de liberação, endereços IP que tem passagem livre pelo *greylisting*, ou porque são máquinas confiáveis (da própria rede, de redes conhecidas, etc.) ou por que seus MTAs não conseguem tratar corretamente erros temporários.

Fonte: antispam.br

Técnicas de bloqueio de spam
Sender Policy Framework (SPF)

SPF é uma tecnologia para combater a falsificação de endereços de retorno dos emails (*return-path*). O mecanismo permite:

- **ao administrador de um domínio:** definir e publicar uma política SPF, onde são designados os endereços das máquinas autorizadas a enviar mensagens em nome deste domínio;
- **ao administrador de um serviço de *e-mail*:** estabelecer critérios de aceitação de mensagens em função da checagem das políticas SPF publicadas para cada domínio.

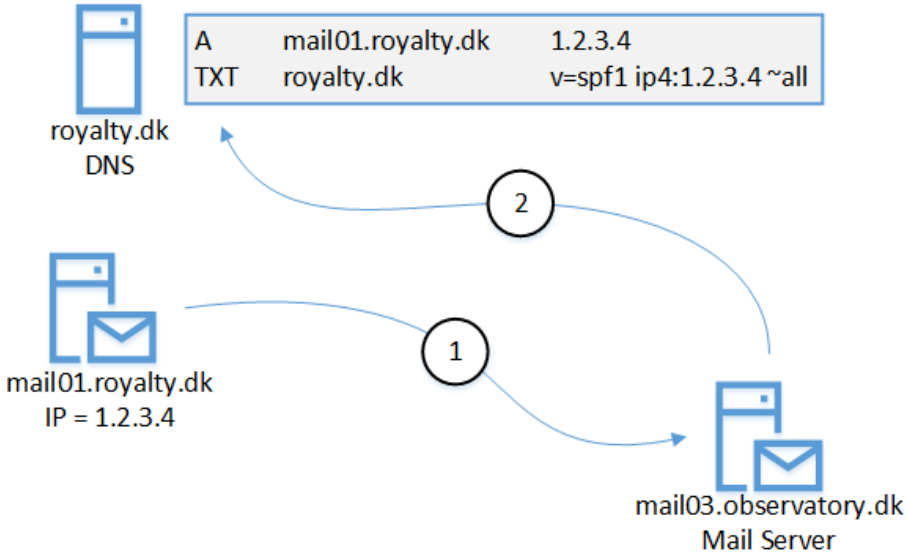
O processo de publicação de uma política SPF é independente da implantação de checagem de SPF por parte do MTA, estes podem ou não ser feitos em conjunto.

Fonte: antispam.br

Técnicas de bloqueio de spam
Sender Policy Framework (SPF)

Publicando a política SPF

Ao publicar uma política de SPF, o administrador de um domínio está autorizando determinados MTAs a enviar e-mails em nome deste domínio. O objetivo é evitar que terceiros enviem mensagem indevidamente em nome de seu domínio, e que mensagens de erro (*bounces*) causadas por spam com envelope falso sejam enviadas para o seu servidor.



Fonte: antispam.br

Técnicas de bloqueio de spam
Sender Policy Framework (SPF)

Publicando a política SPF - continuação

Estas políticas são publicadas através de registros TXT do DNS, em formato ASCII. Um exemplo desse registro é:

Exemplo:

```
example.com.      IN    TXT    "v=spf1 a mx ip4:192.0.2.32/27 -all"
```

Neste caso a política estabelece que pode enviar mensagens em nome do domínio example.com uma máquina que satisfaça um dos seguintes critérios:

- seu endereço IP deve ser um RR tipo A do domínio example.com (a);
- seja designada como MX do domínio example.com (mx); ou
- pertença ao bloco de endereços IP 192.0.2.32/27 (ip4).

A cláusula "-all" diz que devem ser recusados ("-", prefixo Fail) e-mails partindo de qualquer outro endereço IP (all).

Fonte: antispam.br

Técnicas de bloqueio de spam
Sender Policy Framework (SPF)

Prefixos SPF

Todas as opções de prefixos são:

- “+” Pass
- “-” Fail
- “~” SoftFail
- “?” Neutral

O prefixo é opcional, e se omitido o valor utilizado é o “+” (Pass).

A cláusula “all” deve ser sempre a cláusula mais à direita. Ela define qual resposta será retornada em uma consulta SPF, caso nenhuma das outras cláusulas se aplique.

O administrador de um MTA que consulte a política SPF do domínio do remetente de um e-mail, como definido no envelope, poderá rejeitar ou marcar como suspeita uma mensagem que não satisfaça à política SPF daquele domínio.

Fonte: antispam.br

Técnicas de bloqueio de spam
Sender Policy Framework (SPF)

Configurando o MTA

A maioria dos MTAs atuais possui suporte a SPF, seja através de filtros externos (Milters), patches ou suporte nativo. É necessário estabelecer quais serão as ações tomadas dependendo da resposta obtida à consulta SPF. O Internet-Draft "Sender Policy Framework (SPF) for Authorizing Use of Domains in E-MAIL" define algumas possíveis interpretações para os resultados obtidos:

- **não há registro SPF publicado:** neste caso, não é possível determinar se o endereço IP está ou não autorizado a enviar e-mails em nome do domínio sendo consultado.
- **neutral ("?"):** o dono do domínio não tem como ou não quer definir se um determinado endereço IP está ou não autorizado a enviar mensagens em nome do domínio. Este resultado deve ser tratado exatamente como se não existisse um registro SPF, não devendo ser avaliado de forma mais rigorosa devido a isto;
- **pass ("+"):** significa que o IP está autorizado a enviar mensagens em nome do domínio, sendo que o domínio consultado pode, então, ser considerado responsável pelo envio da mensagem;

Fonte: antispam.br

Técnicas de bloqueio de spam
Sender Policy Framework (SPF)

- **fail (“-”):** significa explicitamente que o IP não está autorizado a enviar mensagens em nome do domínio consultado. Este resultado pode ser utilizado para rejeitar a mensagem ou para marcá-la para ser avaliada mais rigorosamente;
- **softfail (“~”):** deve ser tratado como um resultado intermediário entre os níveis fail e neutral. Neste caso, o domínio consultado informa que acha que o IP não está autorizado, mas que não pode fazer uma afirmação taxativa. A mensagem não deve ser rejeitada apenas com base neste resultado, mas é recomendável submetê-la a outros testes. Softfail também tem sido usado para indicar uma situação transitória, em que o SPF está sendo adotado por um domínio.

Em geral, os MTAs que consultam registros SPF podem processar um conjunto de regras pré-definidas pelo administrador antes de processar o registro recebido por DNS.

Fonte: antispam.br

Para saber mais...

… consulte o site antispam.br.

Módulo 9

Server Message Block

Introdução

O Protocolo de SMB (Server Message Block) é um protocolo de comunicação cliente-servidor usado para compartilhar acesso a arquivos, impressoras, portas seriais e outros recursos em uma rede. Também pode transportar protocolos de transação para comunicação entre processos.

Criado pela IBM na década de 1980, o protocolo SMB já gerou várias variantes ou implementações, também conhecidas como dialetos, para atender aos requisitos de rede em evolução ao longo dos anos.

Fonte: techtarget.com

Introdução

Como o protocolo SMB funciona?

O protocolo SMB permite que um aplicativo – ou o usuário de um aplicativo – acesse arquivos em um servidor remoto, bem como outros recursos, incluindo impressoras e *named pipes*. Assim, um aplicativo cliente pode abrir, ler, mover, criar e atualizar arquivos no servidor remoto. Ele também pode se comunicar com qualquer programa do servidor configurado para receber uma solicitação do cliente SMB.

O protocolo SMB é conhecido como um protocolo de solicitação de resposta, o que significa que ele transmite várias mensagens entre o cliente e o servidor para estabelecer uma conexão.

Fonte: techtarget.com

Introdução

O protocolo SMB opera na camada 7, e os seus primeiros dialetos usavam API NetBIOS sobre TCP/IP ou os protocolos herdados, como o Novell IPX ou o NetBEUI.

<table>
<tr><th>OSI</th><th colspan="4"></th><th>TCP/IP</th></tr>
<tr><td>Application
Presentation</td><td colspan="4">SMB</td><td rowspan="2">Application</td></tr>
<tr><td>Session</td><td>NetBIOS</td><td rowspan="3">NetBEUI</td><td>NetBIOS</td><td>NetBIOS</td></tr>
<tr><td>Transport</td><td rowspan="2">IPX[1]</td><td rowspan="2">DECnet</td><td>TCP&UDP</td><td>TCP/UDP</td></tr>
<tr><td>Network</td><td>IP</td><td>IP</td></tr>
<tr><td rowspan="2">Link</td><td rowspan="2">802.2,
802.3,802.5</td><td>802.2</td><td rowspan="2">Ethernet V2</td><td rowspan="2">Ethernet V2</td><td rowspan="2">Ethernet or
others</td></tr>
<tr><td>802.3,802.5</td></tr>
<tr><td>Physical</td><td colspan="4"></td></tr>
</table>

Fonte: techtarget.com

Introdução

Portas de comunicação

Hoje, a comunicação com dispositivos que não suportam SMB diretamente sobre TCP/IP requer o uso do NetBIOS em um protocolo de transporte como o TCP.

O SMB originalmente era executado em cima do NetBIOS usando a porta 139. O NetBIOS é uma camada de transporte mais antiga que permite que os computadores Windows conversem entre si na mesma rede.

Versões posteriores do SMB (após o Windows 2000) começaram a usar a porta 445 em cima de uma pilha TCP. O uso do TCP permite que o SMB funcione pela Internet.

Fonte: techtarget.com

Introdução

Dialetos de protocolo SMB

As variantes do protocolo SMB melhoraram os recursos, a escalabilidade, a segurança e a eficiência da implementação original.

SMB 1.0 (1984): criado pela IBM para compartilhamento de arquivos no DOS. Introduziu o bloqueio oportunista (OpLock) como um mecanismo de cache do lado do cliente projetado para reduzir o tráfego de rede. A Microsoft posteriormente incluiria o protocolo SMB em seu produto LAN Manager;

CIFS (1996): Dialeto SMB desenvolvido pela Microsoft que foi introduzido no Windows 95. Adicionado suporte para tamanhos maiores de arquivo, transporte diretamente sobre TCP/IP e links simbólicos e hard links;

Fonte: techtarget.com

Introdução

Dialetos de protocolo SMB – continuação...

SMB 2.0 (2006): lançado com o Windows Vista e o Windows Server 2008. Redução do número de acessos para melhorar o desempenho, melhor escalabilidade e resiliência e suporte adicional para aceleração de WAN;

SMB 2.1 (2010): introduzido no Windows Server 2008 R2 e no Windows 7. O modelo de leasing de oplock do cliente substituiu o OpLock para melhorar o armazenamento em cache e melhorar o desempenho. Outras atualizações incluíram suporte a unidade de transmissão máxima (MTU) grande e eficiência de energia aprimorada, o que permitiu que os clientes com arquivos abertos de um servidor SMB entrassem no modo de suspensão;

Fonte: techtarget.com

Introdução

Dialetos de protocolo SMB – continuação...

SMB 3.0 (2012): foi lançado no Windows 8 e no Windows Server 2012. Adicionadas várias atualizações significativas para melhorar a disponibilidade, o desempenho, o backup, a segurança e o gerenciamento. Entre os novos recursos destacados estão o SMB Multichannel, o SMB Direct, o failover transparente de acesso ao cliente, o suporte remoto ao VSS, a criptografia SMB e muito mais;

SMB 3.02 (2014): introduzido no Windows 8.1 e no Windows Server 2012 R2. Atualizações de desempenho incluídas e a capacidade de desabilitar completamente o suporte ao CIFS/SMB 1.0, incluindo a remoção dos binários relacionados;

SMB 3.1.1 (2015): Lançado com o Windows 10 e Windows Server 2016. Adicionado suporte para criptografia avançada, integridade de pré-autenticação para evitar ataques man-in-the-middle e fence de dialeto de cluster, entre outras atualizações.

Fonte: techtarget.com

Introdução

Em 2017, os ataques de ransomware WannaCry e Petya exploraram uma vulnerabilidade no **SMB 1.0** para carregar malware em clientes vulneráveis e propagá-lo pelas redes.

A Microsoft divulgou posteriormente um patch, mas especialistas aconselharam usuários e administradores a darem o passo adicional de desabilitar o **SMB 1.0** e/ou **CIFS** em todos os sistemas.

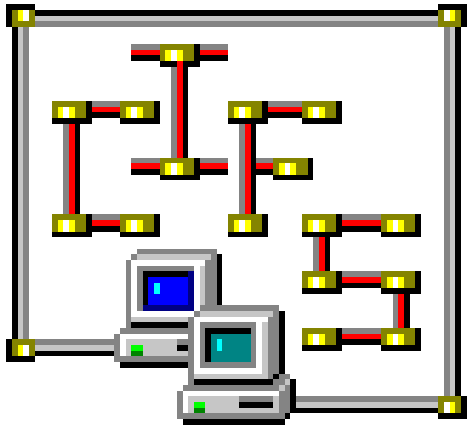
Fonte: techtarget.com

CIFS versus SMB

O CIFS é um dos primeiros dialetos do protocolo SMB desenvolvido pela Microsoft.

Embora os termos sejam às vezes usados de forma intercambiável, o CIFS refere-se apenas a uma única implementação de SMB.

A maioria dos sistemas modernos usa dialetos mais recentes do protocolo SMB.

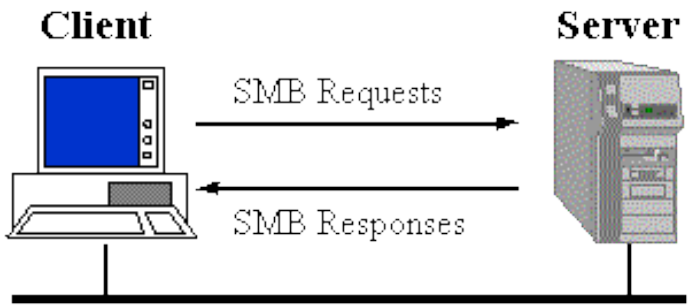


Fonte: techtarget.com

SMB – funcionamento

Os elementos de protocolo (solicitações e respostas) que clientes e servidores trocam são chamados de SMBs. Eles têm um formato específico que é muito semelhante para solicitações e respostas. Cada um consiste em uma porção de cabeçalho de tamanho fixo, seguida por um parâmetro de tamanho variável e uma porção de dados.

Depois de se conectar no nível NetBIOS, via NBF, NetBT, etc, o cliente está pronto para solicitar serviços do servidor. No entanto, o cliente e o servidor devem primeiro identificar qual variante de protocolo cada um deles entende.

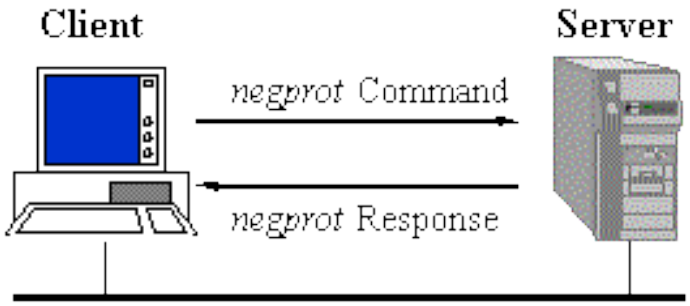


Fonte: samba.org

SMB – funcionamento

O cliente envia um SMB **negprot** ao servidor, listando os dialetos de protocolo que ele entende. O servidor responde com o índice do dialeto que deseja usar ou 0xFFFF, se nenhum dos dialetos for aceitável.

Dialetos mais recentes fornecem informações na resposta do **negprot** para indicar suas capacidades (tamanho máximo do buffer, nomes de arquivos canônicos, etc).



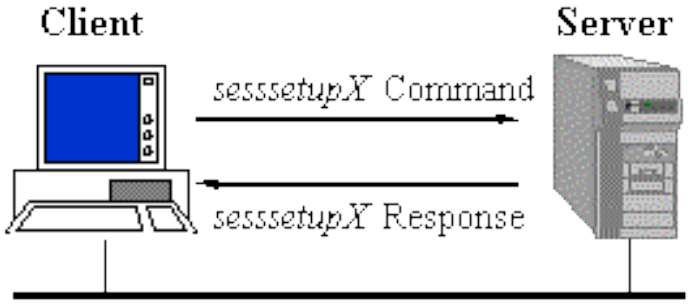
Fonte: samba.org

SMB – funcionamento

Uma vez que um protocolo tenha sido estabelecido, o cliente pode proceder ao logon no servidor, se necessário. Eles fazem isso com um SMB **sesssetupX**.

A resposta indica se eles forneceram ou não um par de senhas de nome de usuário válido e, em caso afirmativo, podem fornecer informações adicionais.

Um dos aspectos mais importantes da resposta é o UID do usuário conectado. Esse UID deve ser enviado com todas as SMBs subsequentes nessa conexão ao servidor.



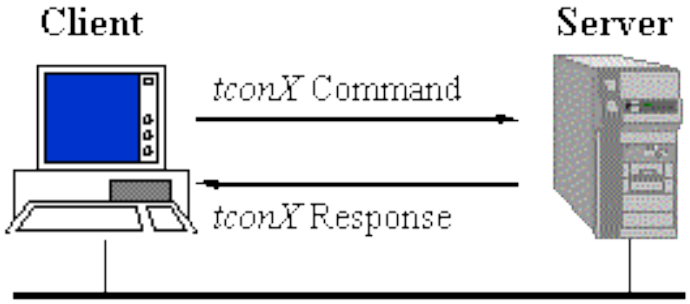
Fonte: samba.org

SMB – funcionamento

Depois que o cliente tiver feito logon, o cliente poderá continuar a se conectar a uma árvore.

O cliente envia um **tcon** ou **tconX** SMB especificando o nome da rede do compartilhamento ao qual deseja se conectar e, se tudo estiver correto, o servidor responderá com um TID que o cliente usará em todas as futuras SMBs relacionadas a esse compartilhamento.

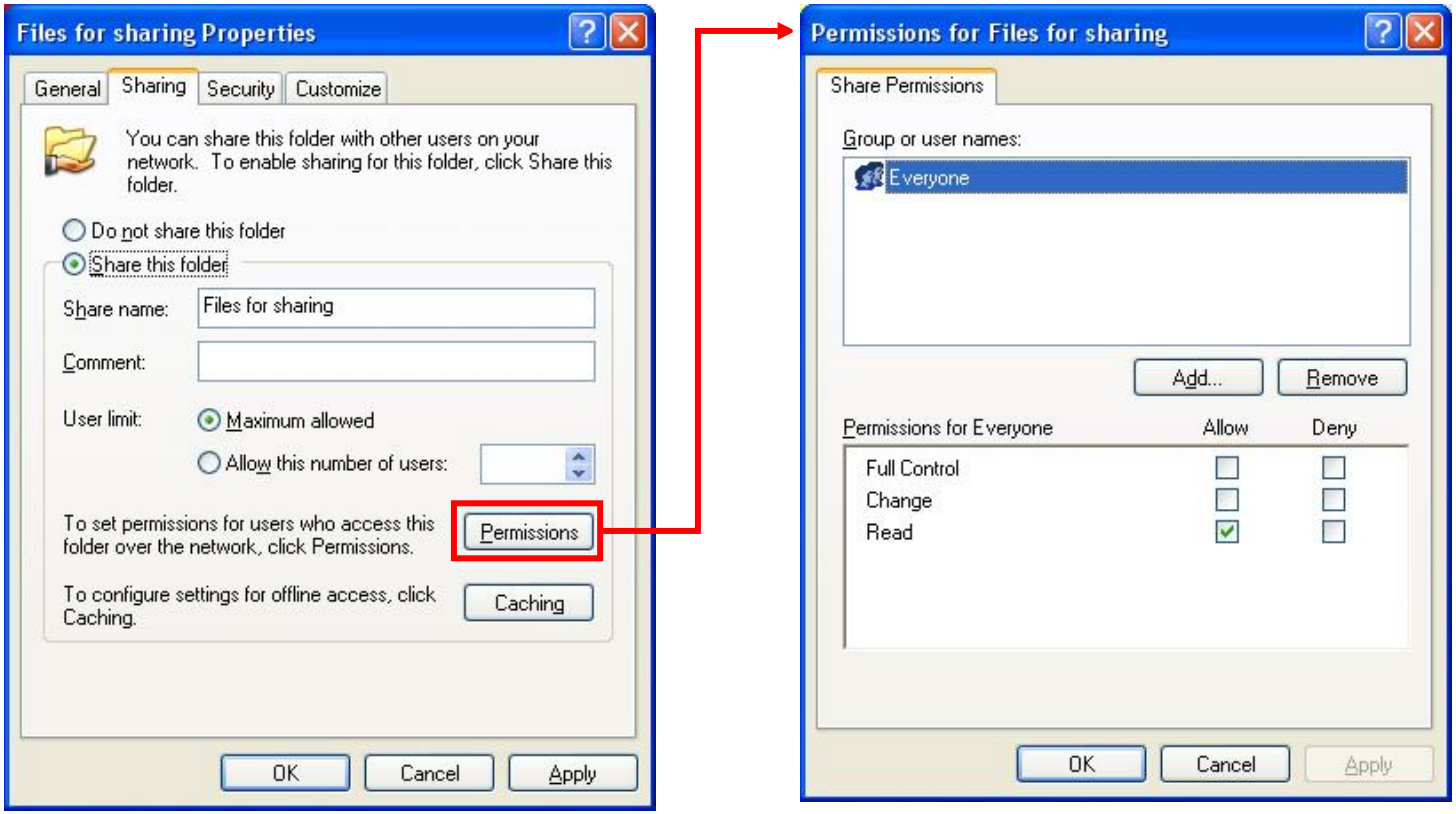
Tendo conectado a uma árvore, o cliente agora pode abrir um arquivo com um SMB aberto, seguido de leitura com SMBs lidos, gravação com SMBs de gravação e fechamento com SMBs próximos.



Fonte: samba.org

SMB – exemplo

Exemplo de configuração de uma pasta compartilhada em uma máquina Windows:



Samba

O Samba é uma implementação de código aberto do protocolo SMB para sistemas Unix e distribuições Linux.

Ele foi lançado em 1992 e suporta serviços de compartilhamento de arquivos e impressão, autenticação e autorização, resolução de nomes e anúncios de serviços entre servidores Linux/Unix e clientes Windows.

Fonte: techtarget.com

Samba

O Samba consiste de dois programas principais, o smbd e nmbd. O seu trabalho é implementar os quatro serviços CIFS básicos modernos, que são:

- Serviços de arquivo e impressão;
- Autenticação e autorização;
- Resolução de nomes;
- Anúncio de serviço (browsing).

Os serviços de arquivo e impressão são a base da suíte CIFS. Estes são fornecidos pelo smbd (daemon SMB). O smbd também lida com autenticação e autorização nos modos “share mode” e “user mode”. No modo share uma senha pode ser atribuída a um diretório compartilhado ou impressora e então concedida a todos que têm permissão para usar o compartilhamento. Com a autenticação no modo user, cada usuário tem seu próprio nome de usuário e senha, e o administrador do sistema pode conceder ou negar o acesso individualmente.

Fonte: samba.org

Samba

Daemon smbd

Os serviços de arquivo e impressão são a base da suíte CIFS. Estes são fornecidos pelo smbd.

O smbd também lida com autenticação e autorização nos modos “share mode” e “user mode”.

No modo share uma senha pode ser atribuída a um diretório compartilhado ou impressora e então concedida a todos que têm permissão para usar o compartilhamento.

Com a autenticação no modo user, cada usuário tem seu próprio nome de usuário e senha, e o administrador do sistema pode conceder ou negar o acesso individualmente.

Fonte: samba.org

Samba

Controlador de domínio

O sistema de domínio do Windows NT fornece um nível adicional de refinamento de autenticação para o CIFS. A ideia básica é que o usuário só precisa efetuar login uma vez para ter acesso a todos os serviços autorizados na rede. O sistema de domínio NT trata disso com um servidor de autenticação, chamado de controlador de domínio. Um Domínio NT é basicamente um grupo de máquinas que compartilham o mesmo Controlador de Domínio.

O sistema de domínio NT merece menção especial porque, até o lançamento do Samba versão 2, somente a Microsoft possuía código para implementar os protocolos de autenticação do domínio NT. Com a versão 2, o Samba introduziu o primeiro código de autenticação do domínio NT não derivado da Microsoft. O objetivo final é imitar completamente um controlador de domínio do Windows NT.

Fonte: samba.org

Samba

Daemon nmbd

As outras duas partes do CIFS, resolução de nomes e navegação (browsing), são tratadas pelo nmbd. Esses dois serviços envolvem basicamente o gerenciamento e a distribuição de listas de nomes NetBIOS.

A resolução de nomes assume duas formas: broadcast e ponto-a-ponto. Uma máquina pode usar um ou ambos os métodos, dependendo de sua configuração. A resolução por broadcast é a mais próxima do mecanismo NetBIOS original. Basicamente, um cliente que procura um serviço qualquer envia um broadcast na rede local e espera que a máquina que prove este serviço responda com um endereço IP.

O outro tipo de resolução de nomes envolve o uso de um servidor NBNS (NetBIOS Name Service). A Microsoft criou o WINS (Windows Internet Name Service), que é uma implementação do NBNS.

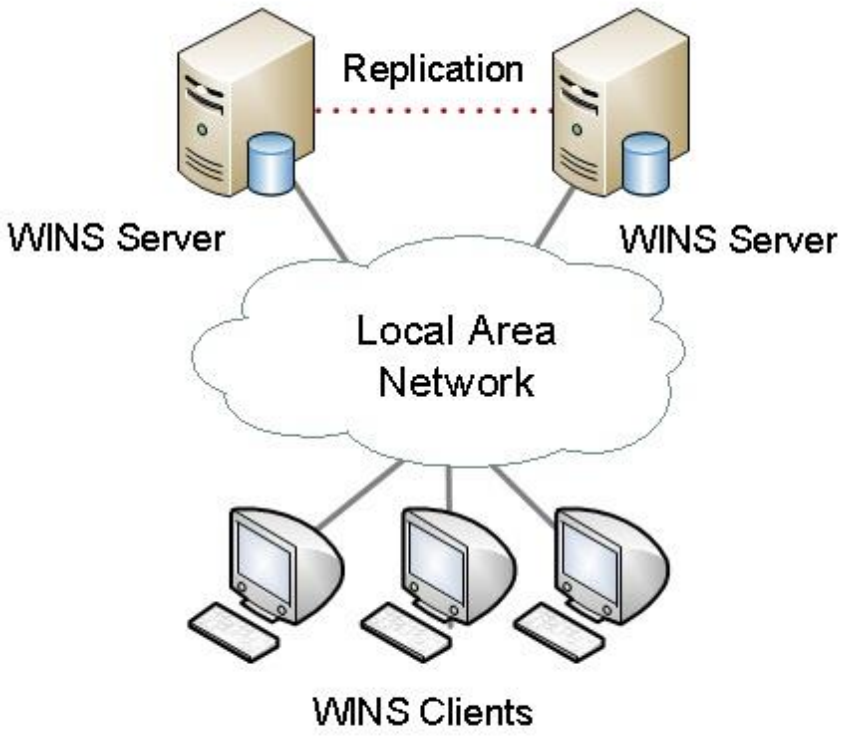
Fonte: samba.org

Samba

Servidor NBNS – resolução de nomes

Os clientes enviam seus nomes NetBIOS e endereços IP para o servidor NBNS, que mantém as informações em um banco de dados simples. Quando um cliente quer falar com outro cliente, ele envia o nome do outro cliente para o servidor NBNS. Se o nome estiver na lista, o NBNS devolve um endereço IP.

Os clientes em diferentes subredes podem compartilhar o mesmo servidor NBNS, portanto, diferentemente da transmissão, o mecanismo ponto-a-ponto não está limitado à LAN local. De muitas maneiras, o NBNS é semelhante ao DNS, mas a lista de nomes NBNS é quase totalmente dinâmica e há poucos controles para garantir que apenas clientes autorizados possam registrar nomes. Conflitos podem acontecer e ocorrem com bastante facilidade.



Fonte: samba.org

Samba

Servidor NBNS – browsing

Em uma LAN, os computadores participantes realizam uma eleição para decidir qual deles se tornará o Local Master Browser (LMB). O “vencedor” em seguida identifica-se, reivindicando um nome NetBIOS especial (além de qualquer outro nome que possa ter). O trabalho de LMBs é manter uma lista de serviços disponíveis, como pastas e impressoras compartilhadas, por exemplo.

Além dos LMBs, existem os Domain Master Browsers (DMBs). Os DMBs coordenam as listas de navegação nos domínios do NT, mesmo em redes roteadas. Usando o NBNS, um LMB localizará seu DMB para trocar e combinar listas de navegação. Assim, a lista de pesquisa é propagada para todos os hosts no domínio do NT.

Infelizmente, os tempos de sincronização são um pouco longos, e pode levar mais de uma hora para que uma alteração em uma subrede remota seja exibida no ambiente de rede.

Fonte: samba.org

Samba – exemplo

Exemplo de configuração de uma pasta compartilhada com acesso público em uma máquina Linux que pode ser acessada por clientes Windows/Linux:

```
[global]
   workgroup = WORKGROUP
   netbios name = Servidor1
   security = none
[NOME_PASTA]
   comment = Pasta Compartilhada
   path = /srv/samba/nome_pasta
   browsable = yes
   writable = yes
   read only = no
```

Fonte: samba.org

Para saber mais...

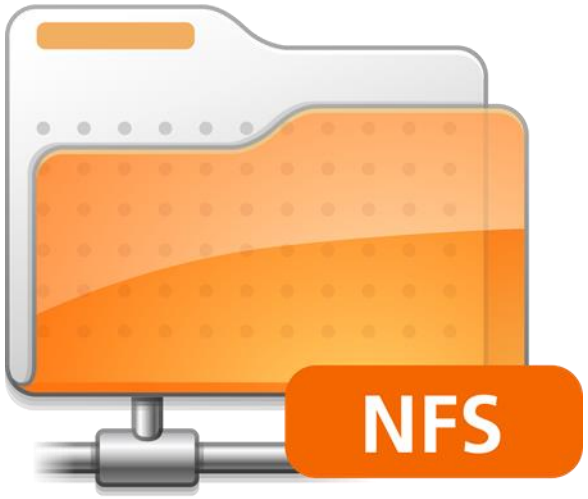
… consulte o site samba.org.

Módulo 10

Network File System

Introdução

O NFS (Network File System) é um protocolo de rede desenvolvido inicialmente pela Sun Microsystems em 1984 para prover um sistema de arquivos distribuídos, que tem por objetivo compartilhar arquivos e diretórios e torna-los acessíveis a qualquer computador conectado em uma rede.



Fonte: NFS Illustrated, de Brent Callaghan

Introdução

A habilidade de poder transferir arquivos entre diversos computadores conectados em uma rede foi uma das primeiras necessidades no uso da ARPANET em 1971.

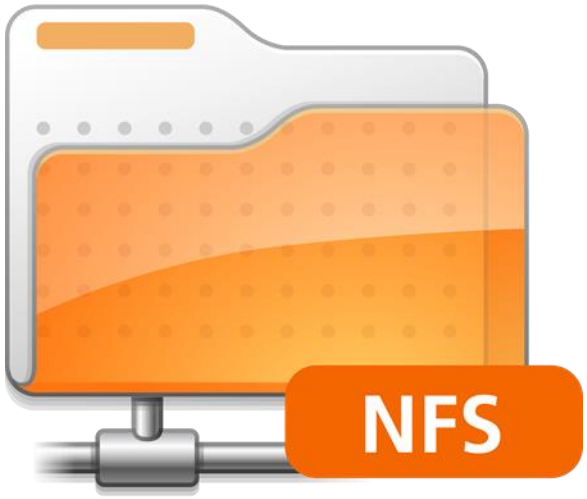
O objetivo da transferência de arquivos é mover um arquivo inteiro em uma rede de um computador para outro, o que é mais conveniente do que transportar o arquivo em um disquete ou fita magnética.

Fonte: NFS Illustrated, de Brent Callaghan

Introdução

Existem atualmente três versões do protocolo NFS:

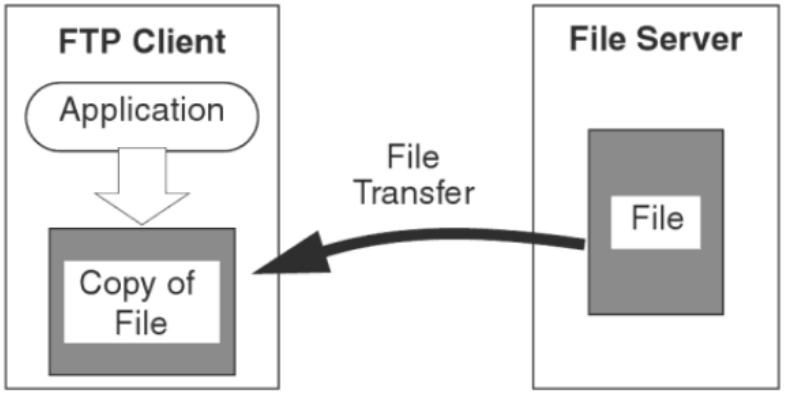
- NFS Version 2 Protocol Specification, de acordo com a RFC 1094;
- NFS Version 3 Protocol Specification, de acordo com a RFC 1813; e
- NFS Version 4 Protocol Specification, de acordo com a RFC 7931.



Protocolos de Transferência de Arquivos

O protocolo FTP, por exemplo, oferece suporte a vários tipos de arquivos diferentes, exibe listas de diretórios e permite alguma manipulação de diretórios no servidor. Com o FTP, os arquivos podem ser criados, removidos e renomeados.

No entanto, o FTP não permite que o conteúdo do arquivo seja manipulado diretamente pelos aplicativos.



É necessário transferir um arquivo em sua totalidade para um disco local antes de poder visualizá-lo ou alterá-lo, e isso torna o acesso a dados remotos menos atraente.

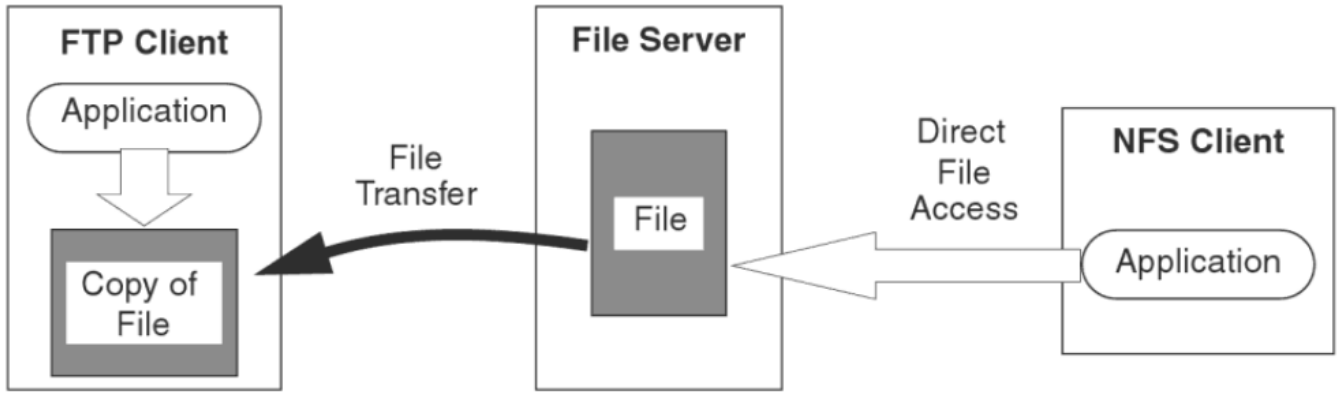
Um arquivo remoto que precisa de transferência de arquivo não pode ser aberto diretamente por um programa, pois ele deve ser transferido para um disco local antes de poder ser visualizado e transferido de volta se tiver sido modificado. Desta forma, o gerenciamento de arquivos fica complicado e custoso.

Fonte: NFS Illustrated, de Brent Callaghan

Protocolos de Acesso a Arquivos

Os protocolos de acesso a arquivos, como o NFS, são projetados para eliminar a necessidade de transferir o arquivo.

O arquivo permanece onde está, no servidor, e é manipulado no local. A manipulação no local tem benefícios óbvios se as alterações forem mínimas, ou seja, é fácil anexar um novo registro a um arquivo grande apenas enviando os dados do novo registro.



Fonte: NFS Illustrated, de Brent Callaghan

Protocolos de Acesso a Arquivos

A alternativa de transferência de arquivo requer que o arquivo seja transferido em sua totalidade em ambas as direções.

Os protocolos de acesso a arquivos têm vantagens significativas sobre os protocolos de transferência de arquivos:

- **Obter apenas o necessário:** se o aplicativo cliente desejar apenas uma pequena parte do arquivo, apenas essa parte precisará ser transferida. Por exemplo, um documento com várias páginas pode consistir em um índice inicial que descreve a localização dos dados dentro do arquivo. O cliente pode obter o índice analítico e, em seguida, obter os dados de interesse de um local dentro do arquivo;
- **Os arquivos remotos parecem ser locais:** um protocolo de acesso a arquivos remotos faz com que os arquivos de um servidor remoto apareçam como se estivessem em um disco local. O usuário de um aplicativo não precisa mais transferir conscientemente um arquivo antes de acessá-lo;

Fonte: NFS Illustrated, de Brent Callaghan

Protocolos de Acesso a Arquivos

- **Dados íntegros:** como os protocolos de acesso a arquivos acessam o arquivo do servidor diretamente, os dados do arquivo estão sempre atualizados (assumindo que não há inconsistência causada pelo cache);
- **Clientes sem disco:** se o cliente não tiver nenhum disco ou tiver menos espaço em disco suficiente para armazenar um arquivo grande, o arquivo não poderá ser transferido. Um protocolo de acesso a arquivos não possui requisitos de armazenamento local;
- **Sem espera:** a transferência de arquivos geralmente requer que todo o arquivo seja transferido antes que os dados possam ser acessados por um aplicativo. Um protocolo de acesso a arquivos pode fornecer dados a um aplicativo assim que eles chegam do servidor de arquivos;
- **Bloqueio de arquivo:** usando um protocolo de acesso a arquivos, um aplicativo cliente pode bloquear um arquivo no servidor para evitar que outros clientes obtenham ou alterem os dados. O bloqueio evita problemas causados por clientes sobrescrevendo as alterações uns dos outros em um arquivo.

Fonte: NFS Illustrated, de Brent Callaghan

Protocolos de Acesso a Arquivos

No entanto, o protocolo NFS também apresenta algumas desvantagens.

Uma delas é que para fazer um sistema de arquivos remoto parecer local, o cliente e o servidor de arquivos precisam de uma conexão de rede que seja aproximadamente tão rápida quanto a conexão da unidade de disco local, caso contrário, a ilusão de um disco “local” não pode ser mantida.

O FTP foi escrito numa época em que se transmitia dados em um enlace de longa distância a uma velocidade de até 56 kbps, enquanto a velocidade de uma conexão de disco SCSI local era de aproximadamente 12 Mbps – cerca de 200 vezes mais rápida.

Foram necessárias redes Ethernet e Token Ring que pudessem mover dados a 1 Mbps ou mais para trazer a velocidade de acesso à rede perto o suficiente daquela das conexões SCSI locais para que o acesso remoto se tornasse prático.

Em alta velocidade, os **protocolos de acesso a arquivos** de redes locais são os mais populares, mas para usuários de enlaces de longa distância lentos, os **protocolos de transferência de arquivos** são mais fáceis de usar.

Fonte: NFS Illustrated, de Brent Callaghan

Primeiros Protocolos de Acesso a Arquivos

As redes de alta velocidade que se tornaram disponíveis no início dos anos 1980 despertaram o interesse de muitos pesquisadores na construção de protocolos de acesso a arquivos.

Nessa época, crescia o interesse por protocolos baseados em Chamadas de Procedimento Remoto ou RPC (Remote Procedure Calls).

O sistema de arquivos RFS da AT&T foi contemporâneo do NFS, mas era complexo e tinha baixo desempenho.

O sistema operacional Apollo DOMAIN dava suporte ao acesso remoto a arquivos, mas sua forte integração ao sistema operacional e ao hardware da Apollo tornavam impraticável implementá-lo em outros sistemas operacionais.

O sistema operacional distribuído LOCUS, desenvolvido na UCLA no início dos anos 1980, fornecia muitos recursos avançados de acesso remoto a arquivos, como cache consistente de alto desempenho, independência de localização, migração e replicação de arquivos e recuperação de falhas. No entanto, assim como o RFS e o DOMAIN, seu sistema de arquivos distribuído era fortemente integrado ao sistema operacional LOCUS, o que também tornava impraticável sua implementação em outros sistemas operacionais.

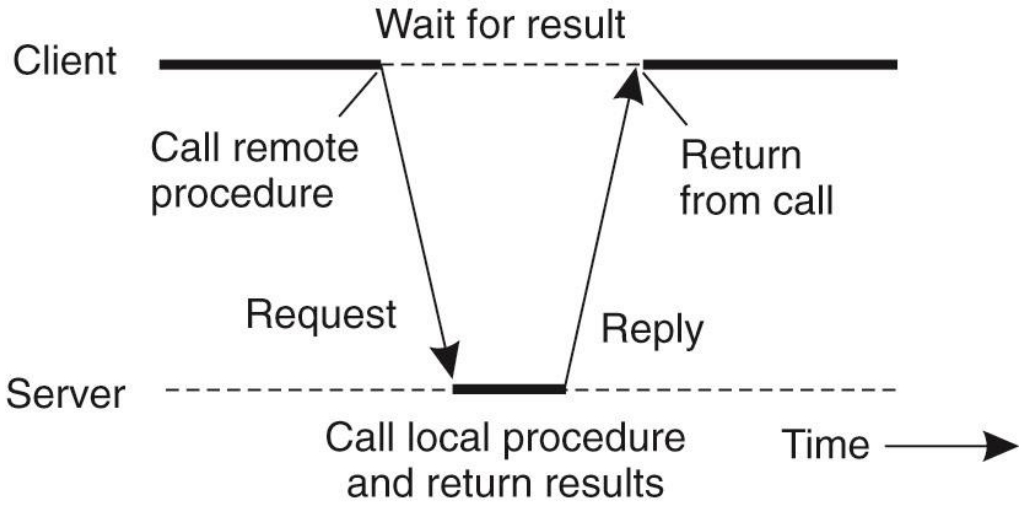
A Newcastle Connection foi outra implementação de acesso remoto a arquivos que precedeu o NFS por vários anos e teve sucesso na criação de expectativas de um sistema de arquivos distribuído que poderia ser portado para outros sistemas operacionais semelhantes ao UNIX.

Fonte: NFS Illustrated, de Brent Callaghan

Chamada de procedimento remoto

A Chamada de Procedimento Remoto ou RPC (Remote Procedure Call) ocorre quando um processo na máquina A (cliente) chama um procedimento na máquina B (servidor).

O processo chamador em A é suspenso e a execução do procedimento chamado ocorre em B. Informações podem ser transportadas do chamador para quem foi chamado nos parâmetros e podem voltar no resultado do procedimento.



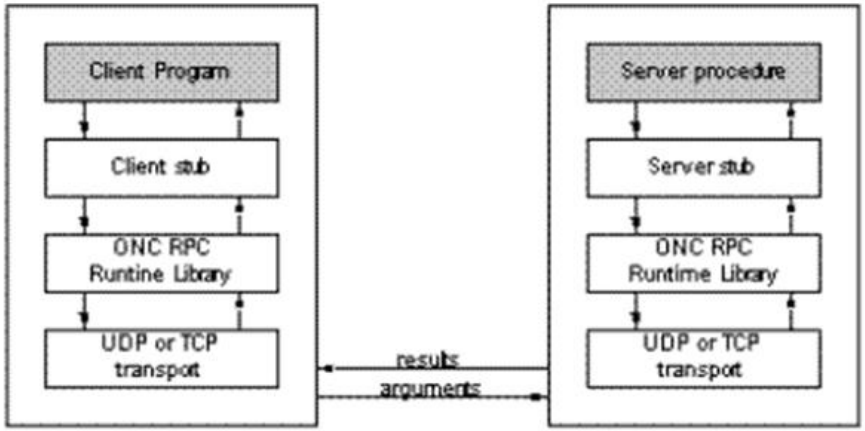
Fonte: Sistemas Distribuídos: Princípios e Paradigmas, de Andrew S. Tanenbaum e Maarten Van Steen.

Camadas de sessão e apresentação

As camadas de sessão e apresentação do modelo ISO/OSI definem a criação e o tempo de vida das conexões de rede e o formato dos dados enviados por essas conexões.

As sessões podem ser construídas sobre qualquer protocolo de transporte compatível. Assim, as sessões de login usam TCP, enquanto os serviços que transmitem informações sobre o host local usam UDP.

O protocolo de sessão usado pelo NFS é o RPC, ou mais precisamente, o protocolo ONC RPC (Open Network Computing Remote Procedure Call), que é um sistema de chamada de procedimento remoto desenvolvido pela Sun Microsystems nos anos 1980 como parte do projeto NFS.



Fonte: Managing NFS and NIS, de Hal Stern, Mike Eisler e Ricardo Labiaga

Modelo Cliente-Servidor

O RPC fornece um mecanismo para que um host faça uma chamada de procedimento que parece ser parte do processo local, mas é realmente executada em outra máquina da rede.

Normalmente, o host no qual a chamada de procedimento é executada possui recursos que não estão disponíveis no host de chamada.

Essa distribuição de serviços de computação impõe uma relação cliente/servidor nos dois hosts: o host que possui o recurso é um servidor para esse recurso e o host que faz a chamada se torna um cliente do servidor quando precisa acessar o recurso.

O recurso pode ser um sistema de arquivos compartilhado, como o NFS.

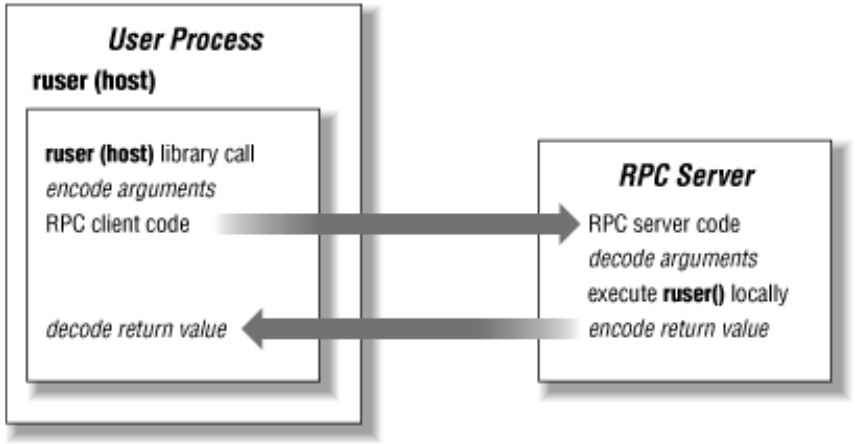
Fonte: Managing NFS and NIS, de Hal Stern, Mike Eisler e Ricardo Labiaga

Modelo Cliente-Servidor

Ao invés de executar o procedimento no host local, o sistema RPC agrupa os argumentos transmitidos ao procedimento em um datagrama de rede.

O método de empacotamento exato é determinado pela camada de apresentação, usando o protocolo XDR (eXternal Data Representation).

O cliente RPC cria uma sessão localizando o servidor apropriado e enviando o datagrama a um processo no servidor que pode executar o RPC.



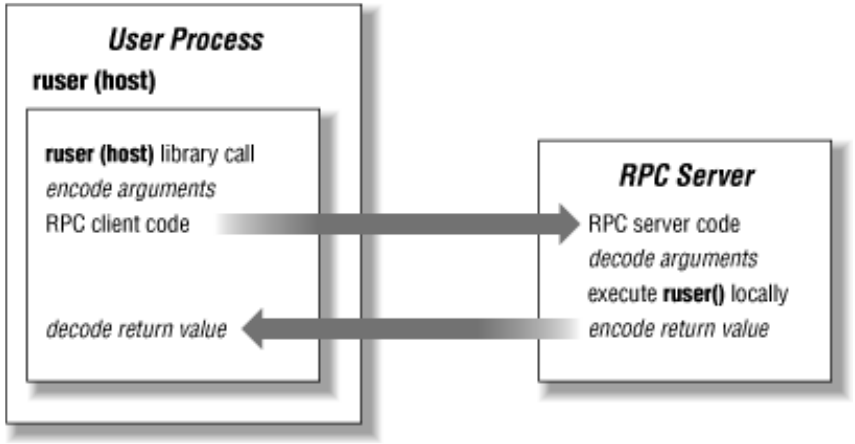
Fonte: Managing NFS and NIS, de Hal Stern, Mike Eisler e Ricardo Labiaga

Modelo Cliente-Servidor

No servidor, os argumentos são descompactados, o servidor executa o resultado, empacota o resultado (se houver) e o envia de volta ao cliente.

De volta ao lado do cliente, a resposta é convertida em um valor de retorno para a chamada de procedimento e o aplicativo do usuário é inserido novamente como se uma chamada de procedimento local tivesse sido concluída.

Este é o final da "sessão", conforme definido no modelo ISO.



Fonte: Managing NFS and NIS, de Hal Stern, Mike Eisler e Ricardo Labiaga

External Data Representation

Os dados do protocolo NFS transmitidos por meio de mensagens RPC devem ser representados em um formato que possa ser compreendido tanto pelo remetente quanto pelo computador destinatário.

O protocolo de representação de dados externos XDR (eXternal Data Representation) foi desenvolvido pela Sun Microsystems e é usado pelo NFS na camada de apresentação.

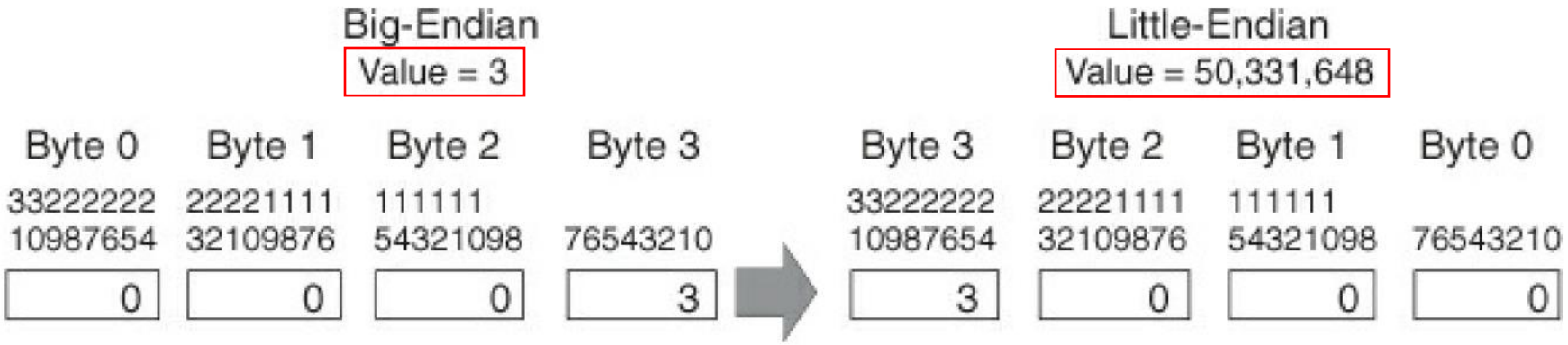
O XDR é construído com base na noção de uma ordem de bytes de rede imutável, chamada de forma canônica.

Fonte: Managing NFS and NIS, de Hal Stern, Mike Eisler e Ricardo Labiaga

External Data Representation

Não é realmente importante qual é a forma canônica – seu sistema pode ou não usar a mesma ordem de bytes e convenções de embalagem de estrutura.

A forma canônica simplesmente permite que os hosts da rede troquem dados estruturados (em oposição a fluxos de bytes) independentemente de quaisquer peculiaridades de uma máquina específica.



Todas as estruturas de dados são convertidas na ordem de bytes da rede e preenchidas de forma adequada.

Fonte: Managing NFS and NIS, de Hal Stern, Mike Eisler e Ricardo Labiaga; NFS Illustrated, de Brent Callaghan

NFS – Implementação

O NFS consiste de, pelo menos, duas partes principais: um servidor e um ou mais clientes.

O cliente acessa remotamente os dados armazenados na máquina servidora. Para tal, o servidor precisa estar rodando os seguintes daemons:

- **nfsd:** o daemon NFS, que atende requisições dos clientes NFS;
- **mountd:** o daemon de montagem NFS, que executa as solicitações que o nfsd lhe passa;
- **portmap:** o daemon portmapper permite que clientes NFS descubram qual porta o servidor NFS está utilizando.

O cliente também pode rodar um daemon, conhecido como **nfsiod**.

O daemon **nfsiod** atende às solicitações do servidor NFS. Isto é opcional e aumenta o desempenho, mas não é requerido para a operação normal e correta.

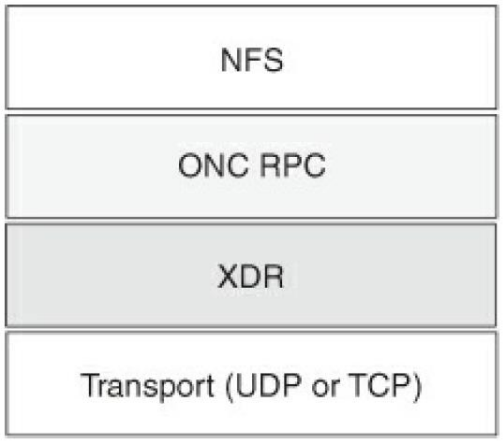
Fonte: FreeBSD Handbook

NFS – Resumo

O protocolo NFS, usado para acesso a arquivos em um sistema distribuído, reside na camada de aplicação e utiliza o protocolo ONC RPC (Open Network Computing Remote Procedure Call).

Por sua vez, na camada de apresentação, o NFS utiliza o protocolo XDR para padronizar as mensagens trocadas entre o cliente o servidor.

Por fim, na camada de transporte, o NFS utiliza o protocolo TCP para estabelecer uma sessão, enquanto utiliza o protocolo UDP para transferir dados.



Fonte: Managing NFS and NIS, de Hal Stern, Mike Eisler e Ricardo Labiaga

FIM