Conceitos de Internet e TCP/IP

A Internet é uma rede pública de comunicação de dados, com controle descentralizado e que utiliza o conjunto de protocolos TCP/IP como base para a estrutura de comunicação e seus serviços de rede. Isto se deve ao fato de que a arquitetura TCP/IP fornece não somente os protocolos que habilitam a comunicação de dados entre redes, mas também define uma série de aplicações que contribuem para a eficiência e sucesso da arquitetura.

Entre os serviços mais conhecidos da Internet estão o correio-eletrônico (protocolos SMTP, POP3), a transferência de arquivos (FTP), o compartilhamento de arquivos (NFS), a emulação remota de terminal (Telnet), o acesso à informação hipermídia (HTTP), conhecido como WWW (World Wide Web).

A Internet é dita ser um sistema aberto uma vez que todos os seus serviços básicos assim como as aplicações são definidas publicamente, podendo ser implementadas e utilizadas sem pagamento de royalties ou licenças para outras instituições.

O conjunto de protocolos TCP/IP foi projetado especialmente para ser o protocolo utilizado na Internet. Sua característica principal é o suporte direto a comunicação entre redes de diversos tipos. Neste caso, a arquitetura TCP/IP é independente da infra-estrutura de rede física ou lógica empregada. De fato, qualquer tecnologia de rede pode ser empregada como meio de transporte dos protocolos TCP/IP, como será visto adiante.

Alguns termos utilizados frequentemente, são explicados de forma resumida adiante:

A Internet (nome próprio) é a denominação da rede mundial que interliga redes no mundo. É formada pela conexão complexa entre centenas de milhares de redes entre si. A Internet tem suas políticas controladas pelo IAB (Internet Architecture Board), um fórum patrocinado pela Internet Society, uma comunidade aberta formada por usuários, fabricantes, representantes governamentais e pesquisadores.

Um internet é um termo usado para definir uma rede genérica formada pela interligação de redes utilizando o protocolo TCP/IP.

Uma intranet é a aplicação da tecnologia criada na Internet e do conjunto de protocolos de transporte e de aplicação TCP/IP em uma rede privada, interna a uma empresa. Numa intranet, não somente a infra- estrutura de comunicação é baseada em TCP/IP, mas também grande quantidade de informações e aplicações são disponibilizadas por meio dos sistemas Web (protocolo HTTP) e correio-eletrônico.

Uma extranet, ou extended intranet é a extensão dos serviços da intranet de uma empresa para interligar e fornecer aplicações para outras empresas, como clientes, fornecedores, parceiros, etc… Desta forma a extranet é a utilização de tecnologias como Web e correio-eletrônico para simplificar a comunicação e a troca de informações entre empresas.

World Wide Web é a designação do conjunto de informações públicas disponibilizadas na Internet por meio do protocolo HTTP. É o somatório das informações que podem ser acessadas por um browser Web na Internet. As informações internas de uma empresa que são acessíveis via um browser Web são enquadradas no termo intranet.

Evolução de TCP/IP e Internet

Em 1966, o Departamento de Defesa do governo americano iniciou, através de sua agência DARPA (Defense Advanced Research Projects Agency) projetos para a interligação de computadores em centros militares e de pesquisa, com o objetivo de criar um sistema de comunicação e controle distribuído com fins militares. Esta iniciativa teve como um dos motivadores o surgimento de minicomputadores com grande poder de processamento, que poderiam ter seu emprego enriquecido com o acesso a uma grande rede de comunicação. Esta rede recebeu o nome de ARPANET.

O principal objetivo teórico da ARPANET era formar uma arquitetura de rede sólida e robusta que pudesse sobreviver a uma perda substancial de equipamento e ainda operar com os computadores e enlaces de comunicação restantes.

Para alcançar este objetivo, o sistema de comunicação deveria suportar diversos tipos de equipamentos distintos, ser dividido em diversos níveis de protocolos distintos para permitir a evolução independente de cada um deles e ser baseado em transferência de pacotes de informação.

Durantet a década de 70 até 1983, a ARPANET era baseada em IMPs (Interface Message Processors), rodando diversos protocolos, sendo o principal o NCP (Network Control Protocol). O TCP/IP ainda estava sendo projetado e a Internet era formada por máquinas de grande porte e minicomputadores ligados aos IMPs. O roteamento fora dos IMPS não existia, impedindo a conexão de máquinas em rede local que surgiam. Ou seja, para se ligar à ARPANET era necessária a ligação direta a um IMP.

Nesta época, os computadores com potencial para se ligar na rede eram de grande porte e em número reduzido. As diferenças de porte desta rede imaginada na época e o que se observa hoje é gigantesco. Um dos projetistas dos sistemas de comunicação da ARPANET, referindo-se ao tamanho de um byte para os identificadores das máquinas, afirmou que “256 máquinas é essencialmente infinito”.

No começo de 1980, a ARPANET foi dividida em ARPANET e MILNET, separando a porção acadêmica e militar. Nesta época, a ARPA decidiu adotar o Unix como sistema operacional prioritário para o suporte de seus projetos de pesquisa (dos quais a ARPANET era um deles), escolhendo a Universidade da Califórnia - Berkeley com centro de desenvolvimento. A ARPA incentivou a criação nativa do suporte de TCP/IP no Unix.

O protocolo TCP/IP começou a ser projetado em 1977 com o objetivo de ser o único protocolo de comunicação da ARPANET. Em 1/1/1983, todas as máquinas da ARPANET passaram a utilizar o TCP/IP como protocolo de comunicação. Isto permitiu o crescimento ordenado da rede, eliminando as restrições dos protocolos anteriores. Em 1986, a NSF (Network Science Foundation) passou a operar o backbone (espinha dorsal) de comunicações com o nome de NSFNet e iniciou a formação de redes regionais interligando os institutos acadêmicos e de pesquisa.

Desde 1983 começaram a surgir diversas redes paralelas nos Estados Unidos financiadas por órgãos de fomento a pesquisa como a CSNET (Computer Science Net), HEPNet (High Energy Physics Net), SPAN (Nasa Space Physics Network) e outras. Estas redes foram integradas ao NSFNet e adicionadas a redes de outros países, caracterizando o início de uso do termo Internet em 1988.

Em 1993, foram criados os protocolos HTTP e o browser Mosaic, dando início ao World Wide Web (WWW). O World Wide Web foi o grande responsável pela crescimento exponencial da Internet, pois permitiu o acesso a informações com conteúdo rico em gráficos e imagens e de forma estruturada. O WWW foi também o grande motivador do uso comercial da Internet, permitindo às empresas disponibilizar informações e vender produtos via Internet.

A NSFNet foi privatizada em 1995, e o backbone passou a ser distribuído e complexo, formado por múltiplas redes de prestadoras de serviços de telecomunicações como AT&T, MCI, Sprint e outros. Hoje a Internet não é mais formada por um único backbone central, mas por um conjunto de grandes provedores de acesso. Em 1995 foi permitido também o tráfego de informações comerciais na Internet.

No Brasil, o acesso à Internet foi iniciado com a conexão de instituições acadêmicas como a Fapesp, USP, Unicamp, PUC-Rio, UFRJ e outras em 1989. Foram formados dois backbones regionais, a RedeRio e a ANSP (An Academic Network at São Paulo) interligando as principais instituições destes estados. Posteriormente foi criada a RNP (Rede Nacional de Pesquisa) com o objetivo de formar um backbone nacional de acesso à Internet e de estimular a formação de redes regionais como a Rede Minas, Rede Tchê e outras.

Em 1995, foi liberado o tráfego comercial, com a Embratel montando e operando o backbone comercial no Brasil. O fornecimento de serviços IP não foi considerado monopólio da Telebrás, permitindo o surgimento de provedores de acesso à Internet.

Hoje o backbone da Internet no Brasil é formado por diversos backbones nacionais interligados entre si, como a RNP, a Embratel e de outras empresas como IBM, Unisys, GlobalOne e outros provedores. O Comitê Gestor da Internet Brasil é o responsável pela determinação de regras e políticas para a porção brasileira da Internet e a Fapesp é responsável pelo registro de nomes de domínio .br.

Protocolos TCP/IP

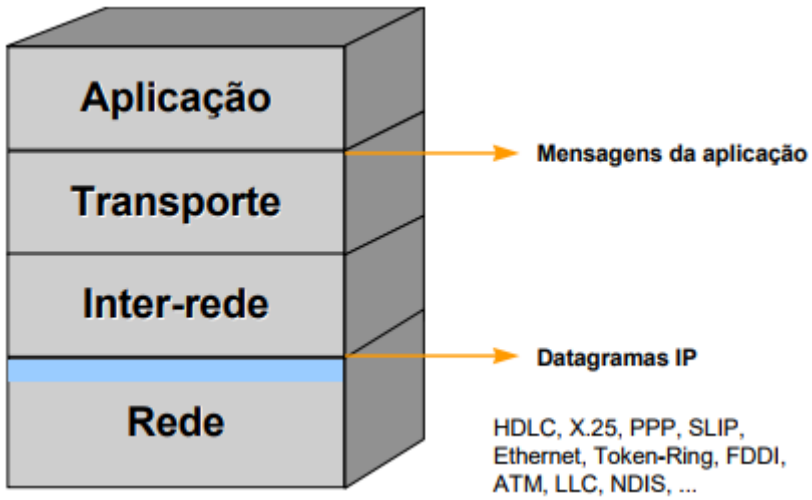
TCP/IP é um acrônimo para o termo Transmission Control Protocol/Internet Protocol Suite, ou seja é um conjunto de protocolos, onde dois dos mais importantes (o IP e o TCP) deram seus nomes à arquitetura. O protocolo IP, base da estrutura de comunicação da Internet é um protocolo baseado no paradigma de chaveamento de pacotes (packet-switching).

Os protocolos TCP/IP podem ser utilizados sobre qualquer estrutura de rede, seja ela simples como uma ligação ponto-a-ponto ou uma rede de pacotes complexa. Como exemplo, pode-se empregar estruturas de rede como Ethernet, Token-Ring, FDDI, PPP, ATM, X.25, Frame-Relay, barramentos SCSI, enlaces de satélite, ligações telefônicas discadas e várias outras como meio de comunicação do protocolo TCP/IP.

A arquitetura TCP/IP, assim como OSI realiza a divisão de funções do sistema de comunicação em estruturas de camadas. Em TCP/IP as camadas são:

Aplicação Tranporte Inter-Rede Rede

A figura 1 ilustra a divisão em camadas da arquitetura TCP/IP:



Camada de Rede

A camada de rede é responsável pelo envio de datagramas construídos pela camada Inter-Rede.

Esta camada realiza também o mapeamento entre um endereço de identificação de nível Inter-rede para um endereço físico ou lógico do nível de Rede. A camada Inter-Rede é independente do nível de Rede.

Alguns protocolos existentes nesta camada são:

Protocolos com estrutura de rede própria (X.25, Frame-Relay, ATM)

Protocolos de Enlace OSI (PPP, Ethernet, Token-Ring, FDDI, HDLC, SLIP, …)

Protocolos de Nível Físico (V.24, X.21)

Protocolos de barramento de alta-velocidade (SCSI, HIPPI, …)

Protocolos de mapeamento de endereços (ARP - Address Resolution Protocol) - Este protocolo pode ser considerado também como parte da camada Inter-Rede.

Os protocolos deste nível possuem um esquema de identificação das máquinas interligadas por este protocolo. Por exemplo, cada máquina situada em uma rede Ethernet, Token-Ring ou FDDI possui

um identificador único chamado endereço MAC ou endereço físico que permite distinguir uma máquina de outra, possibilitando o envio de mensagens específicas para cada uma delas. Tais rede são chamadas redes locais de computadores.

Da mesma forma, estações em redes X.25, Frame-Relay ou ATM também possuem endereços que as distinguem uma das outras.

As redes ponto-a-ponto, formadas pela interligação entre duas máquinas não possuem, geralmente, um endereçamento de nível de rede (modelo TCP/IP), uma vez que não há necessidade de identificar várias estações.

Camada Inter-Rede

Esta camada realiza a comunicação entre máquinas vizinhas através do protocolo IP. Para identificar cada máquina e a própria rede onde estas estão situadas, é definido um identificador, chamado endereço IP, que é independente de outras formas de endereçamento que possam existir nos níveis inferiores. No caso de existir endereçamento nos níveis inferiores é realizado um mapeamento para possibilitar a conversão de um endereço IP em um endereço deste nível.

Os protocolos existentes nesta camada são:

Protocolo de transporte de dados: IP - Internet Protocol

Protocolo de controle e erro: ICMP - Internet Control Message Protocol

Protocolo de controle de grupo de endereços: IGMP - Internet Group Management Protocol

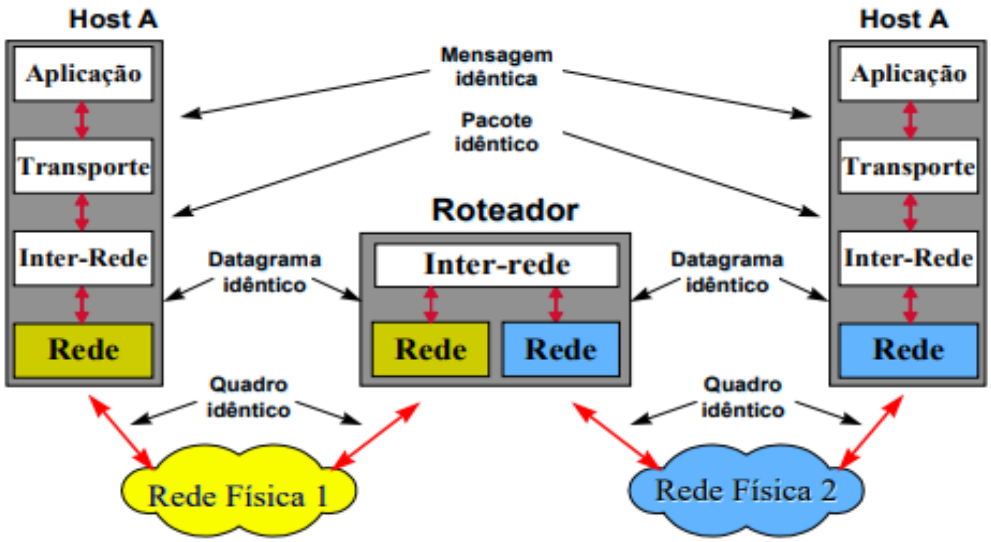
Protocolos de controle de informações de roteamento

O protocolo IP realiza a função mais importante desta camada que é a própria comunicação inter-redes.

Para isto ele realiza a função de roteamento que consiste no transporte de mensagens entre redes e na decisão de qual rota uma mensagem deve seguir através da estrutura de rede para chegar ao destino.

O protocolo IP utiliza a própria estrutura de rede dos níveis inferiores para entregar uma mensagem destinada a uma máquina que está situada na mesma rede que a máquina origem.

Por outro lado, para enviar mensagem para máquinas situadas em redes distintas, ele utiliza a função de roteamento IP. Isto ocorre através do envio da mensagem para uma máquina que executa a função de roteador. Esta, por sua vez, repassa a mensagem para o destino ou a repassa para outros roteadores até chegar no destino.



Camada de Transporte

Esta camada reúne os protocolos que realizam as funções de transporte de dados fim-a-fim, ou seja, considerando apenas a origem e o destino da comunicação, sem se preocupar com os elementos intermediários. A camada de transporte possui dois protocolos que são o UDP (User Datagram Protocol) e TCP (Transmission Control Protocol).

O protocolo UDP realiza apenas a multiplexação para que várias aplicações possam acessar o sistema de comunicação de forma coerente.

O protocolo TCP realiza, além da multiplexação, uma série de funções para tornar a comunicação entre origem e destino mais confiável. São responsabilidades do protocolo TCP: o controle de fluxo, o controle de erro, a sequenciação e a multiplexação de mensagens.

A camada de transporte oferece para o nível de aplicação um conjunto de funções e procedimentos para acesso ao sistema de comunicação de modo a permitir a criação e a utilização de aplicações de forma independente da implementação.

Desta forma, as interfaces socket ou TLI (ambiente Unix) e Winsock (ambiente Windows) fornecem um conjunto de funções-padrão para permitir que as aplicações possam ser desenvolvidas independentemente do sistema operacional no qual rodarão.

Camada de Aplicação

A camada de aplicação reúne os protocolos que fornecem serviços de comunicação ao sistema ou ao usuário. Pode-se separar os protocolos de aplicação em protocolos de serviços básicos ou protocolos de serviços para o usuário:

Protocolos de serviços básicos, que fornecem serviços para atender as próprias necessidades do sistema de comunicação TCP/IP: DNS, BOOTP, DHCP

Protocolos de serviços para o usuário: FTP, HTTP, Telnet, SMTP, POP3, IMAP, TFTP, NFS, NIS, LPR, LPD, ICQ, RealAudio, Gopher, Archie, Finger, SNMP e outros

Posicionamento do Nível OSI

A arquitetura TCP/IP possui uma série de diferenças em relação à arquitetura OSI. Elas se resumem principalmente nos níveis de aplicação e Inter-rede da arquitetura TCP/IP.

Como principais diferenças pode-se citar:

OSI trata todos os níveis, enquanto TCP/IP só trata a partir do nível de Rede OSI

OSI tem opções de modelos incompatíveis. TCP/IP é sempre compatível entre as várias implementações

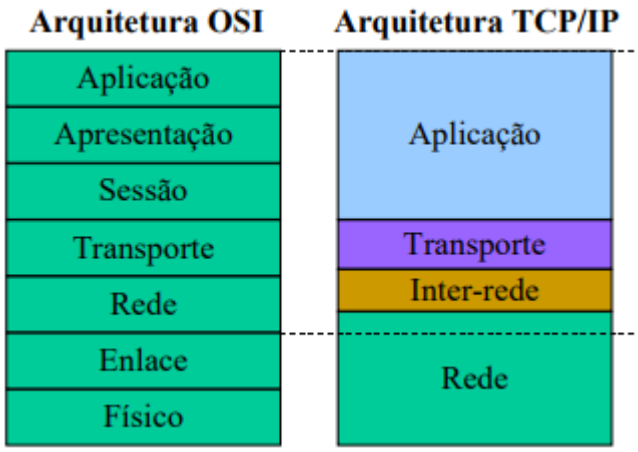
OSI oferece serviços orientados a conexão no nível de rede, o que necessita de inteligência adicional em cada equipamento componente da estrutura de rede. Em TCP/IP a função de roteamento é bem simples e não necessita de manutenção de informações complexas

TCP/IP tem função mínima (roteamento IP) nos nós intermediários (roteadores)

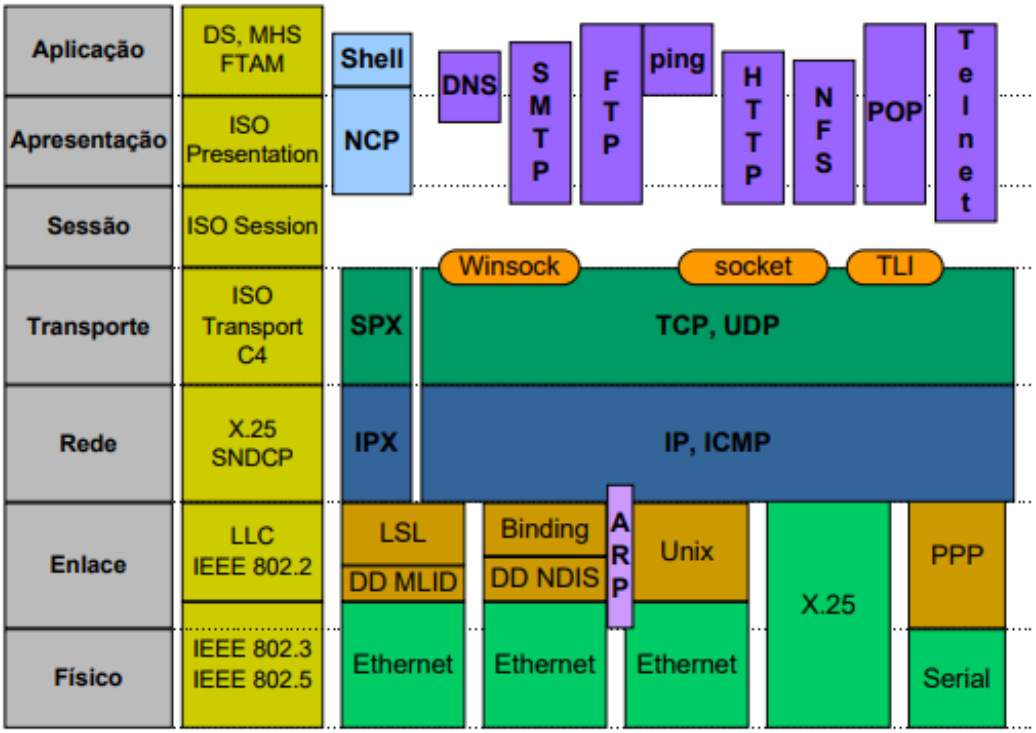
Aplicações TCP/IP tratam os níveis superiores de forma monolítica, Desta forma OSI é mais eficiente pois permite reaproveitar funções comuns a diversos tipos de aplicações. Em TCP/IP, cada aplicação tem que implementar suas necessidades de forma completa.

A figura 3 ilustra a comparação entre TCP/IP e OSI. Note que a camada Inter-rede de TCP/IP apresenta uma altura menor que o correspondente nível de Rede OSI. Isto representa o fato de que uma das funções do nível de Rede OSI é realizada pelo nível de Rede TCP/IP.

Esta função é a entrega local de mensagens dentro da mesma rede. O IP só trata a entrega e a decisão de roteamento quando o origem e o destino da mensagem estão situados em redes distintas.



A figura abaixo ilustra um posicionamento geral de diversos protocolos nas arquiteturas OSI, TCP/IP e Novell Netware:



Internet e Padronização de Protocolos e Funções

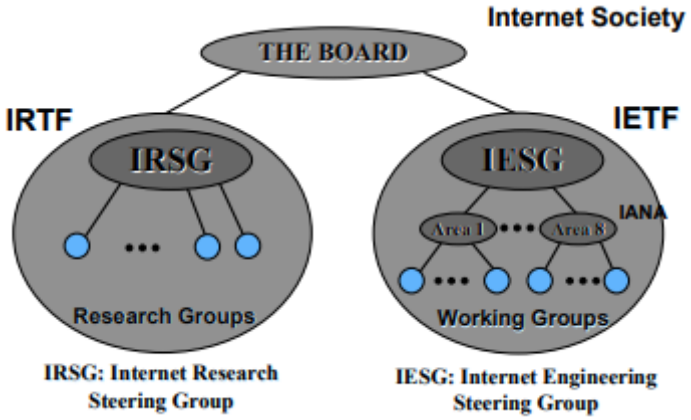
A Internet é controlada pelo IAB (Internet Architecture Board) em termos de padronizações e recomendações. Este gerencia as funções de definição de padrões de protocolos, criação de novos protocolos, evolução, etc.. O IAB é um forum suportado pela Internet Society (ISOC), cujos membros organizam as reuniões e o funcionamento do IAB, além de votarem os seus representantes.

O controle da Internet em relação a sua operação normal é dividida em diversos órgãos, alguns centrais e outros por países. Por exemplo, o órgão que gerencia toda a política de fornecimento de endereços IP e outros códigos utilizados nos protocolos é o IANA - Internet Assigned Numbers Authority.

Por sua vez, a distribuição de endereços IP, assim como nomes de domínio (DNS), assim como a manutenção da documentação de padronização da Internet é realizada pelo InterNIC (Internet Network Information Center) que atualmente é operado por um conjunto de empresas, principalmente AT&T e Network Solutions Inc. Outro órgão relevante é

o GTLD-Mou, um comitê criado em 1997 para decidir sobre a padronização de novos nomes básicos da Internet (como .com, .org, .gov, .arts, .web e outros).

A figura 4 ilustra o diagrama da IAB. Este consiste de um órgão executivo, o IETF (Internet Engineering Task Force), que é responsável pela definição e padronização de protocolos utilizados na Internet. O IRTF (Internet Research Task Force) é responsável por criar, projetar e propor novas aplicações, em nome do IAB. Além das contribuições iniciadas pelo IRTF, qualquer instituição ou pessoa pode submeter propostas de novos protocolos ou aplicações ao IRTF.



O processo de padronização é baseado em um documento chamado RFC (Request for Comments) que contém a definição ou proposição de algum elemento (prática, protocolo, sistema, evolução, aplicação, histórico, etc…) para a Internet. Quando uma nova proposta é submetida ela recebe o nome de Draft Proposal.

Esta proposta será analisada pelo Working Group especializado na área que se refere e se aprovada por votação, recebe um número e se torna uma RFC. Cada RFC passa por fases, onde recebe classificações Como Proposed Standard, Draft Standard, até chegar a um Internet Standard. Um protocolo não precisa se tornar um Internet Standard para ser empregado na Internet. De fato são poucos os que tem esta classificação.

As RFCs podem ter os seguintes status:

S = Internet Standard PS = Proposed Standard DS = Draft Standard

BCP = Best Current Practices E = Experimental

I = Informational H = Historic

Hoje existem aproximadamente 2400 RFCs publicadas. Cerca de 500 reúnem as informações mais importantes para implementação e operação da Internet

Abaixo enumera-se algumas RFCs importantes e a classificação por STANDARD. O STANDARD é o agrupamento das RFC que se referem a um determinado padrão:

Classif.	STD	RFC	Descrição
Padrões	STD-1	2200	INTERNET OFFICIAL PROTOCOL STANDARDS
	STD-2	1700	ASSIGNED NUMBERS
	STD-3	1122	Requirements for Internet hosts - communications layers
	=	1123	Requirements for Internet hosts - application and support
	STD-4	1009	Requirements for Internet Gateways
		1812	Requirements for IP Routers
		1918	Address Allocation for Private Internets
Internet		2135	Internet Society By-Laws
		2134	Articles of Incorporation of Internet Society
		2008	Implication of Various Address Allocation Policies for Internet Routing

		2026	The Internet Standards Process - Rev.3
		2050	The Internet Registry IP Allocation Guidelines
IP	STD-5	791	IP - Internet Protocol
	=	792	ICMP - Internet Control Message Protocol
	=	919	Broadcasting Internet Datagrams
	=	922	Broadcasting Internet datagrams in the presence of subnets
	=	950	Internet standard subnetting procedure
	=	1112	Host extensions for IP multicasting - IGMP
		2101	IPv4 address Behaviour Today
		1256	ICMP Router Discovery Protocol
		2236	Internet Group Management Protocol, v.2
		1788	ICMP Domain Name Messages
		1191	Path MTU Discovery Protocol
UDP	STD-6	768	User Datagram Protocol - UDP
TCP	STD-7	793	Transmission Control Protocol
		1144	Compressing TCP headers for low speed serial links
		1323	TCP Extensions for High Performance
Telnet	STD-8	854	Telnet Protocol specification
	=	855	Telnet Option Specification
FTP	STD-9	959	File Transfer Protocol - FTP
SMTP	STD-10	821	Simple Mail Transfer Protocol - SMTP
	=	1869	SMTP Service Extensions
	=	1870	SMTP Service Extension for Message Size Declaration
		1652	SMTP Service Extensions for 8-bit MIME transport
		1891	SMTP Service Extensions for Delivery Status Notification
		2142	Mailbox Names for Common Services, Roles and Functions
Mail-Content	STD-11	822	Standard Format for ARPANET Messages
	=	1049	Content-type header field for Internet messages
NTP	STD-12	1119	Network Time Protocol v.2 - NTP
DNS	STD-13	1034	Domain names - concepts and facilities
	=	1035	Domain names - implementation and specification
	STD-14	974	Mail Routing and the Domain Name System
	STD-15	1137	A Simple Network Management Protocol - SNMP
		1034,	Domain Names, concepts, facilities, implementation and speci- fication
		1035	
		2100	The Naming of Hosts
		2136	Dynamic Updates in the Somain Name System
		2181	Clarifications to DNS Specification
		2182	Selection and Operation of Secondary DNS Servers
SNMP-MIB	STD-16	1155	Structure and Identification of Management Information for TCP/IP-
			based Internets
	=	1212	Concise MIB Definitions
	STD-17	1213	Management Information Base for Network Management of TCP/IP-
			based internets - MIB II

Netbios/IP	STD-19	1001	Protocol standard for a NetBIOS service on a TCP/UDP transport:
			Cocenpts and Methods
	=	1002	Protocol standard for a NetBIOS service on a TCP/UDP transport:
			Detailed specifications
TFTP	STD-33	1350	Tiny FTP Protocol Rev.2
IP-SLIP	STD-47	1055	IP datagrams over serial lines: SLIP
PPP	STD-51	1661	Point-to-Point Protocol - PPP
		1662	PPP in HDLC-like Framing
		1332	IPCP - PPP IP Control Protocol
		1570	PPP LCP Extensions
		1662	PPP in HDLC Framing
		2153	PPP Vendor Extensions
POP3	STD-53	1939	Post Office Protocol v.3 - POP3
RIP		1722,	RIP - Routing Information Protocol version 2
		1723	
OSPF	STD-54	2328	Open Shortest Path Protocol - OSPF v.3
		2154	OSPF with Digital Signatures
ARP		866	ARP - Address Resolution Protocol
		903	RARP - Reverse Address Resolution Protocol
		1027	Proxy ARP
ATM		1483	Multiprotocol Encapsulation Over ATM
		1577	Classic IP over ATM
BOOTP		951	BOOTP - Bootstrap Protocol
		1497	BOOTP Vendor Extensions
		1533	DHCP Options and BOOTP Vendor Extensions
BGP		1771	Border Gateway Protocol 4
		1517,	CIDR - ClassLess Interdomain Router
		1518,	
		1519	
		1930	Guidelines for creation, slection and registration of an Autono-mous
			System (AS)
DHCP		2131	DHCP - Dynamic Host Configuration Protocol
		2132	DHCP Options and BOOTP Vendor Extensions
		1534	Interoperation Between DHCP and BOOTP
		2241	DHCP Options for Novell Directory Services
		2242	Netware/IP Domain Name and Information
RADIUS		2138	Remote Authentication Dial-in User Service (RADIUS)
		2139	RADIUS Accounting
HTML		1866	HTML - HyperText Markup Language
		2110	MIME E-mail Encapsulation of Aggregate Documents such as HTML
HTTP		2068	HTTP/1.1 - HyperText Transfer Protocol
		2109	HTTP State Management Mechanism
		2168	Resolution of Uniform Resource Identifiers using the Domain Name
			System

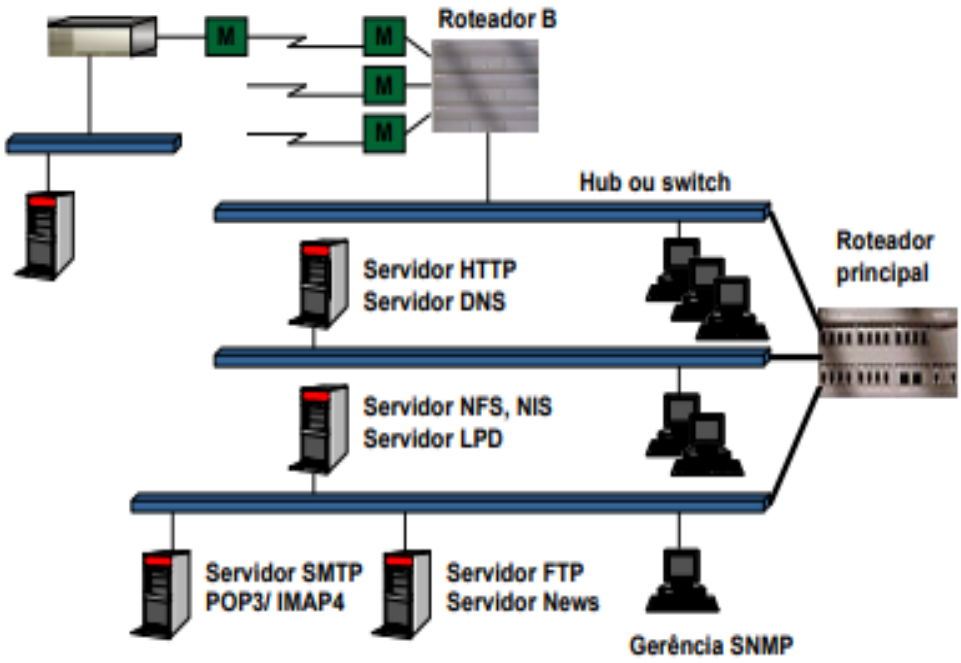
		2145	Use and Interpretation of HTTP Version Numbers
LDAP		2251	LDAP (Lightweight Directory Access Protocol) v.3
IRC		1459	IRC - Internet Relay Chat
MIME		1521	MIME - Multipurpose Internet Mail Extension
NFS		1813	NFS Version 3 - Network File System
NNTP		977	NNTP - Network News Tranport Protocol
IPV6		2147	TCP and UDP over IPv6 Jumbograms
		2185	Routing Aspects of IPv6 Transition
ICP		2186	Internet Cache Protocol (ICP), v2
		2187	Application of ICP, v2
Segurança		2196	Site Security Handbook
Histórico		2235	Hobbe's Internet Timeline
Resumos		2151	A Primer on Internet and TCP/IP Tools and Utilities

No Brasil, assim como nos outros países, existem órgãos específicos para o controle local. No Brasil o Comitê Gestor da Internet é responsável pela definição de políticas de utilização, e a FAPESP é responsável pela distribuição de endereços e atribuição de nomes de domínio.

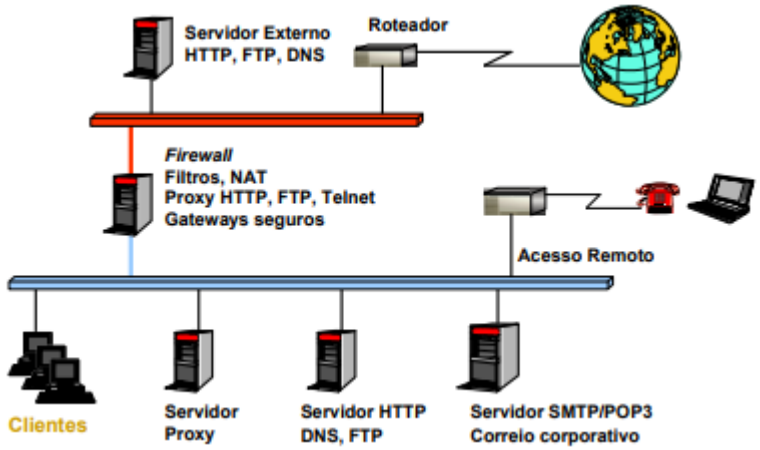
Exemplos de Aplicação de Redes com Arquitetura TCP/IP

Seguem abaixo, alguns exemplos de aplicações da arquiteturas distintas de rede baseadas em TCP/IP, como por exemplo, redes internas de empresas baseadas em transporte TCP/IP, serviços de redes de empresas conectados à Internet, provedores de acesso à Internet.

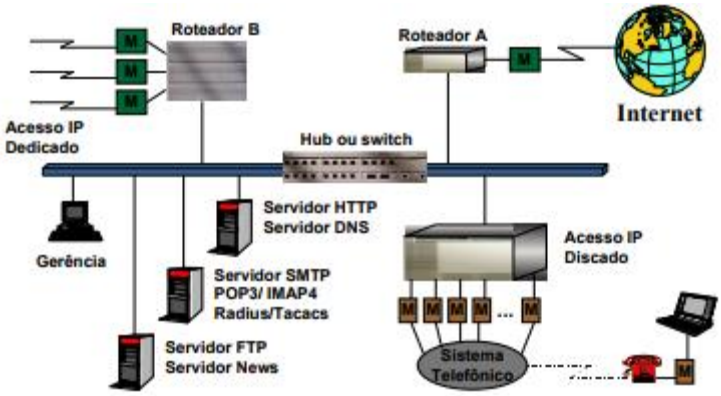
Exemplo 1: Redes internas à empresa utilizando protocolos TCP/IP para formar a estrutura de comunicação e a base das aplicações de rede (correio-eletrônico), compartilhamento de arquivos, distribuição de informação via hipertexto, etc… e chamadas de intranet:



Exemplo 2: Uma estrutura de rede TCP/IP conectada à Internet de forma segura, através da utilização de um firewall, que realiza o filtro de pacotes IP e o transporte de protocolo de aplicações por meio de um gateway (proxy):

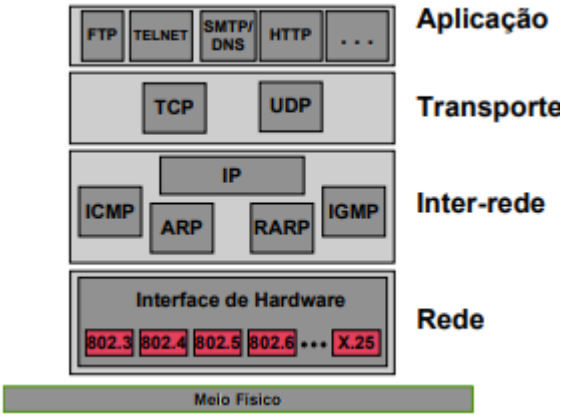


Exemplo 3: Um provedor de acesso à Internet, fornecendo serviços de conexão a usuários discados e empresas por meio de ligação dedicada, além de oferecer os serviços básicos de Internet como HTTP, SMTP, POP3, FTP, etc…



Protocolos da Camada Inter-Rede

A figura 8 ilustra o posicionamento de diversos protocolos da arquitetura TCP/IP:



O Nível Inter-rede compreende principalmente os protocolos IP e ICMP e IGMP (Internet Group Management Protocol). Os protocolos ARP e RARP são pertencentes na verdade aos dois níveis, Inter-rede e Rede pois realizam funções com informações de ambos.

Protocolo IP

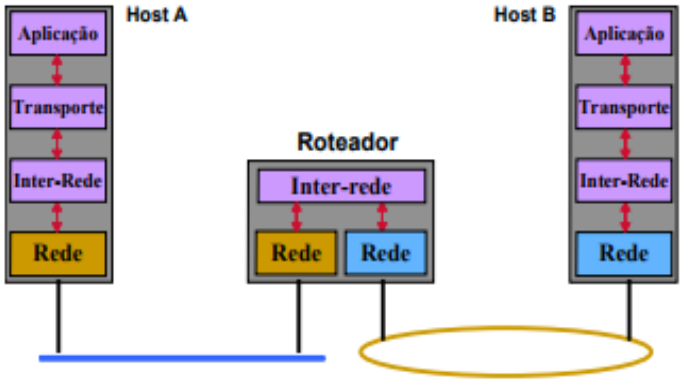
O Protocolo IP é responsável pela comunicação entre máquinas em uma estrutura de rede TCP/IP. Ele provê a capacidade de comunicação entre cada elemento componente da rede para permitir o transporte de uma mensagem de uma origem até o destino.

O protocolo IP provê um serviço sem conexão e não-confiável entre máquinas em uma estrutura de rede. Qualquer tipo de serviço com estas características deve ser fornecido pelos protocolos de níveis superiores. As funções mais importantes realizadas pelo protocolo IP são a atribuição de um esquema de endereçamento independente do endereçamento da rede utilizada abaixo e independente da própria topologia da rede utilizada, além da capacidade de rotear e tomar decisões de roteamento para o transporte das mensagens entre os elementos que interligam as redes.

Na arquitetura TCP/IP, os elementos responsáveis por interligar duas ou mais redes distintas são chamados de roteadores. As redes interligadas podem ser tanto redes locais, redes geograficamente distribuídas, redes de longa distância com chaveamento de pacotes ou ligações ponto-a-ponto seriais. Um roteador tem como característica principal a existência de mais de uma interface de rede, cada uma com seu próprio endereço específico. Um roteador pode ser um equipamento específico ou um computador de uso geral com mais de uma interface de rede.

Por outro lado, um componente da arquitetura TCP/IP que é apenas a origem ou destino de um datagrama IP (não realiza a função de roteamento) é chamado de host.

As funções de host e roteador podem ser visualizadas na figura 9:



Endereços IP

Um endereço IP é um identificador único para certa interface de rede de uma máquina. Este endereço é formado por 32 bits (4 bytes) e possui uma porção de identificação da rede na qual a interface está conectada e outra para a identificação da máquina dentro daquela rede. O endereço IP é representado pelos 4 bytes separados por . e representados por números decimais. Desta forma o endereço IP: 11010000 11110101 0011100 10100011 é representado por 208.245.28.63.

Como o endereço IP identifica tanto uma rede quanto a estação a que se refere, fica claro que o endereço possui uma parte para rede e outra para a estação. Desta forma, uma porção do endereço IP designa a rede na qual a estação está conectada, e outra porção identifica a estação dentro daquela rede.

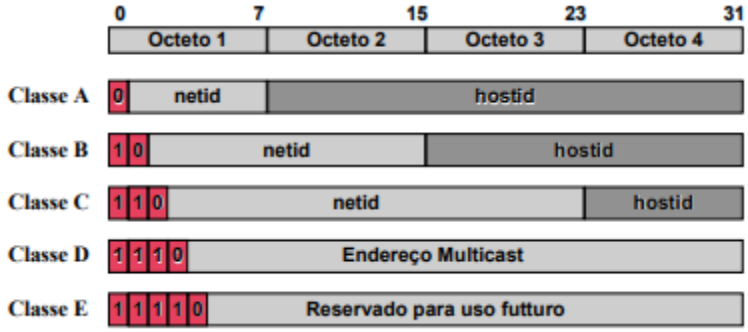
Uma vez que o endereço IP tem tamanho fico, uma das opções dos projetistas seria dividir o endereço IP em duas metades, dois bytes para identificar a rede e dois bytes para a estação. Entretanto isto traria inflexibilidade pois só poderiam ser endereçados 65536 redes, cada uma com 65536 estações. Uma rede que possuísse apenas 100 estações estaria utilizando um endereçamento de rede com capacidade de 65536 estações, o que também seria um desperdício.

A forma original de dividir o endereçamento IP em rede e estação, foi feita por meio de classes. Um endereçamento de classe A consiste em endereços que tem uma porção de identificação de rede de 1 byte e uma porção de identificação de máquina de 3 bytes. Desta forma, é possível endereçar até 256 redes com 2 elevado a 32 estações. Um endereçamento de classe B utiliza 2 bytes para rede e 2 bytes para estação, enquanto um endereço de classe C utiliza 3 bytes para rede e 1 byte para esta-

ção. Para permitir a distinção de uma classe de endereço para outra, utilizou-se os primeiros bits do primeiro byte para estabelecer a distinção (veja figura abaixo).

Nesta forma de divisão é possível acomodar um pequeno número de redes muito grandes (classe A) e um grande número de redes pequenas (classe C). Esta forma de divisão é histórica e não é mais empregada na Internet devido ao uso de uma variação que é a sub-rede, como será visto em seção adiante. Entretanto sua compreensão é importante para fins didáticos.

As classes originalmente utilizadas na Internet são A, B, C, D, E., conforme mostrado abaixo. A classe D é uma classe especial para identificar endereços de grupo (multicast) e a classe E é reservada.



A Classe A possui endereços suficientes para endereçar 128 redes diferentes com até 16.777.216 hosts (estações) cada uma.

A Classe B possui endereços suficientes para endereçar 16.284 redes diferentes com até 65.536 hosts cada uma.

A Classe C possui endereços suficientes para endereçar 2.097.152 redes diferentes com até 256 hosts cada uma.

As máquinas com mais de uma interface de rede (caso dos roteadores ou máquinas interligadas à mais de uma rede, mas que não efetuam a função de roteamento) possuem um endereço IP para cada uma, e podem ser identificados por qualquer um dos dois de modo independente. Um endereço IP identifica não uma máquina, mas uma conexão à rede.

Alguns endereços são reservados para funções especiais:

Endereço de Rede: Identifica a própria rede e não uma interface de rede específica, representado por todos os bits de hostid com o valor ZERO.

Endereço de Broadcast: Identifica todas as máquinas na rede específica, representado por todos os bits de hostid com o valor UM.

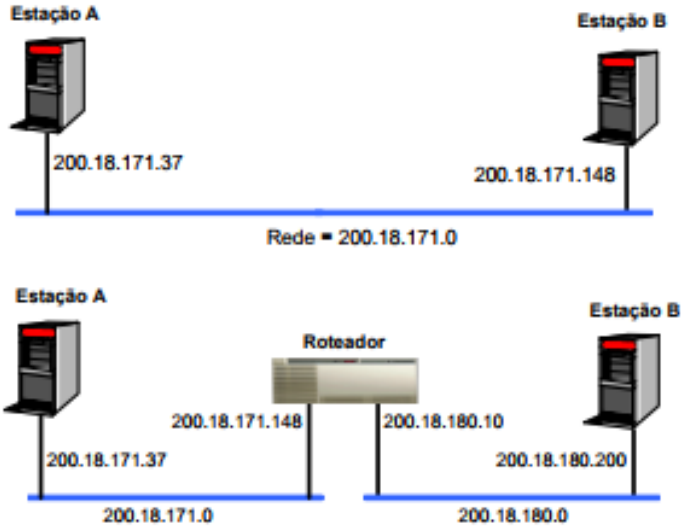
Desta forma, para cada rede A, B ou C, o primeiro endereço e o último são reservados e não podem ser usados por interfaces de rede.

Endereço de Broadcast Limitado: Identifica um broadcast na própria rede, sem especificar a que rede pertence. Representado por todos os bits do endereço iguais a UM = 255.255.255.255.

Endereço de Loopback: Identifica a própria máquina. Serve para enviar uma mensagem para a própria máquina rotear para ela mesma, ficando a mensagem no nível IP, sem ser enviada à rede. Este endereço é 127.0.0.1. Permite a comunicação inter-processos (entre aplicações) situados na mesma máquina.

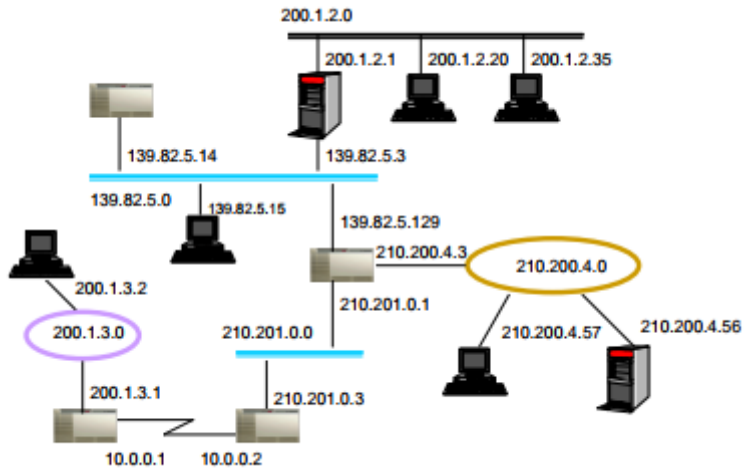
As figuras abaixo mostram exemplos de endereçamento de máquinas situadas na mesma rede e em redes diferentes. Pode ser observado que como o endereço começa por 200 (ou seja, os dois primeiros bits são 1 e o terceiro 0), eles são de classe C. Por isto, os três primeiros bytes do endereço identificam a rede.

Como na primeira figura, ambas as estações tem o endereço começando por 200.18.171, elas estão na mesma rede. Na segunda figura, as estações estão em redes distintas e uma possível topologia é mostrada, onde um roteador interliga diretamente as duas redes.



A figura abaixo ilustra um diagrama de rede com o endereçamento utilizado. Note que não há necessidade de correlação entre os endereços utilizados nas redes adjacentes. O mecanismo para que uma mensagem chegue na rede correta é o roteamento. Cada elemento conectando mais de uma rede realiza a função de roteamento IP, baseado em decisões de rotas. Note que mesmo os enlaces formados por ligações ponto-a- pontos são também redes distintas.

Neste diagrama existem 6 redes, identificadas por 200.1.2.0, 139.82.0.0, 210.200.4.0, 210.201.0.0, 10.0.0.0 e 200.1.3.0.



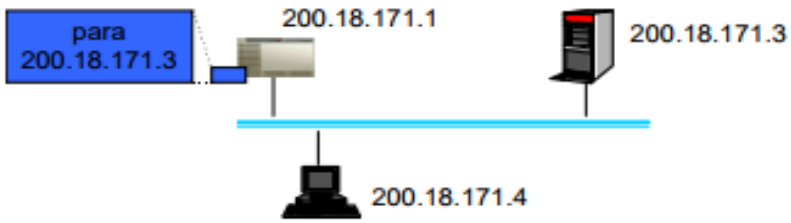
Mapeamento de Endereços IP em Endereços de Rede

Os protocolos de rede compartilhada como Ethernet, Token-Ring e FDDI possuem um endereço próprio para identificar as diversas máquinas situadas na rede. Em Ethernet e Token-Ring o endereçamento utilizado é chamado endereço físico ou endereço MAC - Medium Access Control , formado por 6 bytes, conforme a figura abaixo:

Este tipo de endereçamento só é útil para identificar diversas máquinas, não possuindo nenhuma informação capaz de distinguir redes distintas. Para que uma máquina com protocolo IP envie um pacote para outra máquina situada na mesma rede, ela deve se basear no protocolo de rede local, já que é necessário saber o endereço físico. Como o protocolo IP só identifica uma máquina pelo endereço IP, deve haver um mapeamento entre o endereço IP e o endereço de rede MAC. Este mapeamento é realizado pelo protocolo ARP.

O mapeamento via protocolo ARP só é necessário em uma rede do tipo compartilhada como Ethernet, Token- Ring, FDDI, etc. Em uma rede ponto-a-ponto como, por exemplo, um enlace serial, o protocolo ARP não é necessário, já que há somente um destino possível.

A figura abaixo mostra uma rede com 3 estações, onde uma máquina A com endereço IP 200.18.171.1 deseja enviar uma mensagem para a máquina B cujo endereço é 200.18.171.3. A mensagem a ser enviada é uma mensagem IP. No caso do exemplo abaixo, antes de efetivamente enviar a mensagem IP, a estação utilizará o protocolo ARP para determinar o endereço MAC da interface cujo endereço IP é o destino da mensagem.



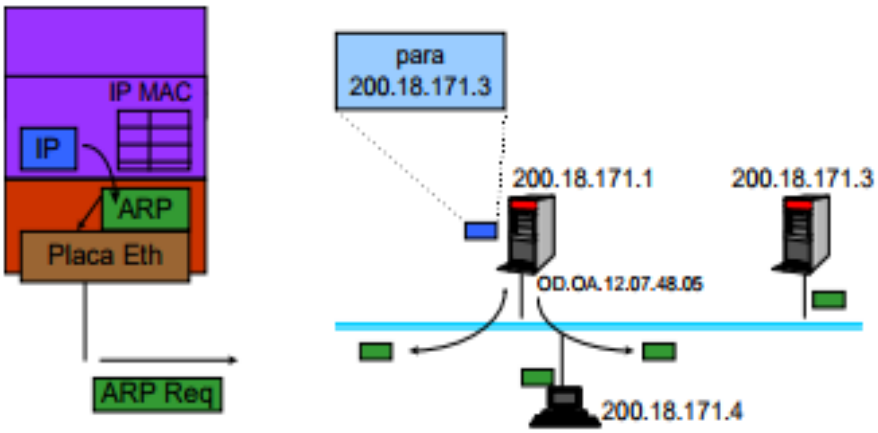
O funcionamento do protocolo ARP é descrito abaixo:

Estação A verifica que a máquina destino está na mesma rede local, determinado através dos endereços origem e destino e suas respectivas classes.

O protocolo IP da estação A verifica que ainda não possui um mapeamento do endereço MAC para o endereço IP da máquina destino.

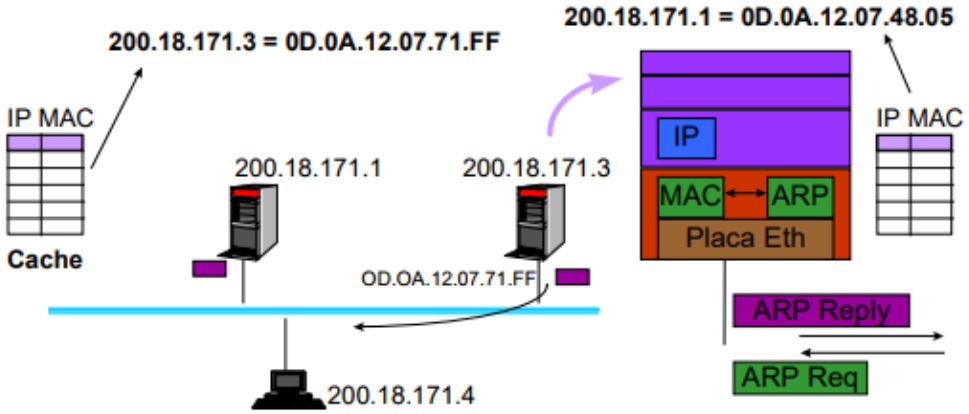
O protocolo IP solicita ao protocolo que o endereço MAC necessário

Protocolo ARP envia um pacote ARP (ARP Request) com o endereço MAC destino de broadcast (difusão para todas as máquinas)



A mensagem ARP enviada é encapsulada em um pacote Ethernet conforma mostrado abaixo:

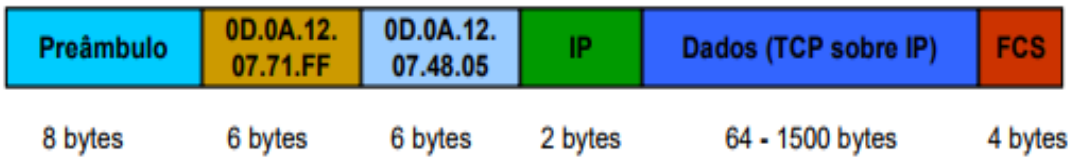
Todas as máquinas recebem o pacote ARP, mas somente aquela que possui o endereço IP especificado responde. A máquina B já instala na tabela ARP o mapeamento do endereço 200.18.171.1 para o endereço MAC de A.



A resposta é enviada no pacote Ethernet, encapsulado conforme mostrado abaixo, através de uma mensagem ARP Reply endereçado diretamente para a máquina origem

A máquina A recebe o pacote e coloca um mapeamento do endereço IP de B e seu endereço MAC respectivo. Esta informação residirá em uma tabela que persistirá durante um certo tempo.

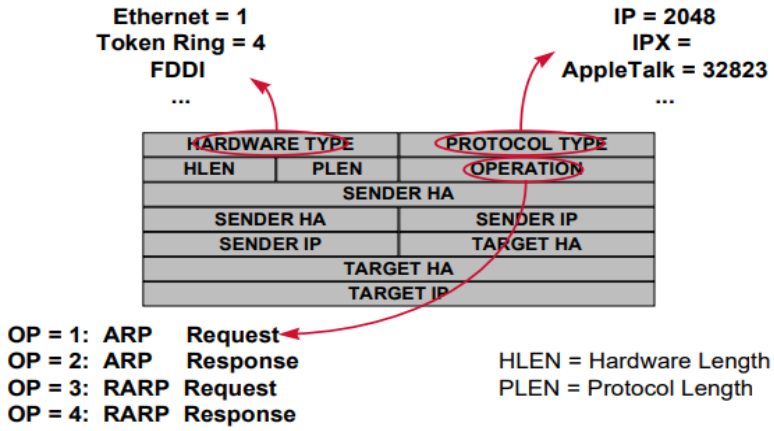
Finalmente a máquina A transmite o pacote IP inicial, após saber o endereço MAC da estação destino.



Os protocolos de nível de Rede como Ethernet possuem um identificador para determinar o tipo do protocolo que está sendo carregado no seu campo de dados. Um pacote Ethernet pode, por exemplo, carregar os protocolos ARP, IP, RARP, IPX, Netbios e outros. A figura abaixo mostra o formato do quadro Ethernet.

Note que o campo protocolo, de 2 bytes de tamanho identifica o protocolo sendo carregado no campo de dados. No caso de transporte de um pacote ARP, o valor é 0806h (hexadecimal), enquanto que no caso de IP este campo tem o valor 0800h.

O protocolo ARP possui dois pacotes, um REQUEST e um REPLY, com o formato abaixo. No REQUEST, são preenchidos todos os dados exceto o endereço MAC do TARGET. No REPLY este campo é completado.



HARDWARE TYPE identifica o hardware (Ethernet, Token-Ring , FDDI, etc) utilizado, que pode variar o tamanho do endereço MAC.

PROTOCOL TYPE identifica o protocolo sendo mapeado (IP, IPX, etc,) que pode variar o tipo do endereço usado.

OPERATION identifica o tipo da operação, sendo:

1 = ARP Request, 2 = ARP Reply, 3 = RARP Request, 4 = RARP Reply

Roteamento IP

O destino de um mensagem IP sendo enviado por uma máquina pode ser a própria estação, uma estação situada na mesma rede ou uma estação situada numa rede diferente. No primeiro caso, o pacote é enviado ao nível IP que o retorna para os níveis superiores. No segundo caso, é realizado o mapeamento por meio de ARP e a mensagem é enviada por meio do protocolo de rede.

Quando uma estação ou roteador deve enviar um pacote para outra rede, o protocolo IP deve enviá-lo para um roteador situado na mesma rede. O roteador por sua vez irá enviar o pacote para outro roteador, na mesma rede que este e assim sucessivamente até que o pacote chegue ao destino final. Este tipo de roteamento é chamado de Next-Hop Routing, já que um pacote é sempre enviado para o próximo roteador no caminho.

Neste tipo de roteamento, não há necessidade de que um roteador conheça a rota completa até o destino. Cada roteador deve conhecer apenas o próximo roteador para o qual deve enviar a mensagem. Esta decisão é chamada de decisão de roteamento. Uma máquina situado em uma rede que tenha mais de um roteador deve também tomar uma decisão de roteamento para decidir para qual roteador deve enviar o pacote IP.

Quando uma estação deve enviar uma mensagem IP para outra rede, ela deve seguir os seguintes passos:

Determinar que a estação destino está em outra rede e por isto deve-se enviar a mensagem para um roteador

Determinar, através da tabela de rotas da máquina origem, qual roteador é o correto para se enviar a mensagem

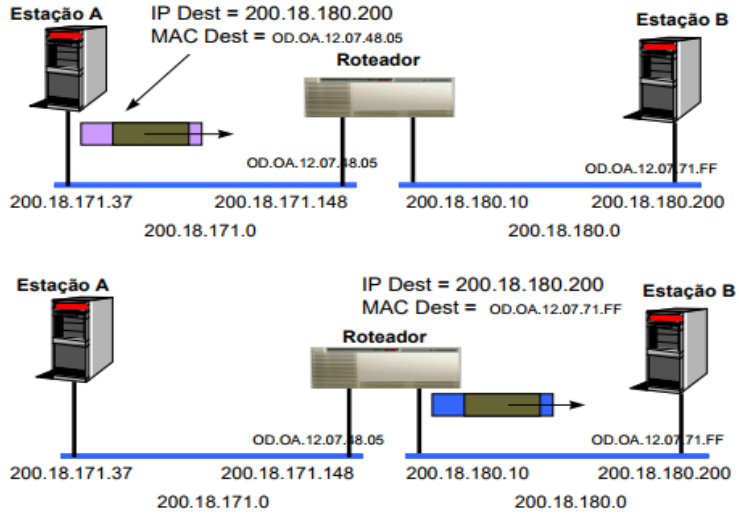
Descobrir, através do protocolo ARP, qual o endereço MAC do roteador

Enviar a mensagem IP com o endereço de nível de rede apontado para o roteador e o endereço IP (na mensagem IP) endereçado para a máquina destino.

Uma questão importante no pacote roteado consiste no fato de que o pacote a ser roteado é endereçado fisicamente ao roteador (endereço MAC), mas é endereçado logicamente (endereçamento IP) à máquina destino. Quando o roteador recebe um pacote que não é endereçado a ele, tenta roteá-lo.

A decisão de roteamento é baseada em uma tabela, chamada de tabela de rotas, que é parte integrante de qualquer protocolo IP. Esta tabela relaciona cada rede destino ao roteador para onde o pacote deve ser enviado para chegar a ela.

As figuras abaixo mostram o funcionamento do roteamento:



Nas figuras acima o roteamento é realizado somente por um roteador. Caso houvesse mais de um roteador a ser atravessado, o primeiro roteador procederia de forma idêntica à Estação A, ou seja determinaria a rota correta e enviaria a mensagem para o próximo roteador.

O Algoritmo de Transmissão de um pacote IP é descrito abaixo. A transmissão pode ser aplicada tanto a um host quanto a uma estação:

Datagrama pronto para ser transmitido

Caso:

Endereço Destino == Endereço Transmissor

Entrega datagrama pela interface loopback (127.0.0.1)

Fim

Endereço de rede do destino == endereço de rede local

Descobre o endereço físico do destino (ARP)

Transmite datagrama pela interface correta

Fim

Endereço de rede do destino != endereço de rede local

Verifica tabela de rotas

Descobre rota que se encaixa com a rede destino

Descobre o endereço físico do gateway (ARP)

Transmite o datagrama para o gateway

Fim

O Algoritmo de Recepção de um pacote IP é descrito abaixo:

Datagrama recebido da camada intra-rede, defragmentado e testado

Caso:

Endereço Destino = Endereço do Host, ou E.D. = outras interfaces do Host, ou E.D. = Broadcast

Passa datagrama para níveis superiores -> FIM

Caso:

Máquina que recebeu não é roteador

Descarta datagrama -> FIM2.2.2 Máquina é roteador (possui mais de uma interface IP)

Caso:

Endereço IP destino = Rede IPcom interface direta

Descobre o endereço físico do destino (ARP)

Transmite datagrama pela interface respectiva -> FIM

Caso Endereço de rede do destino endereço de rede local

Verifica tabela de rotas

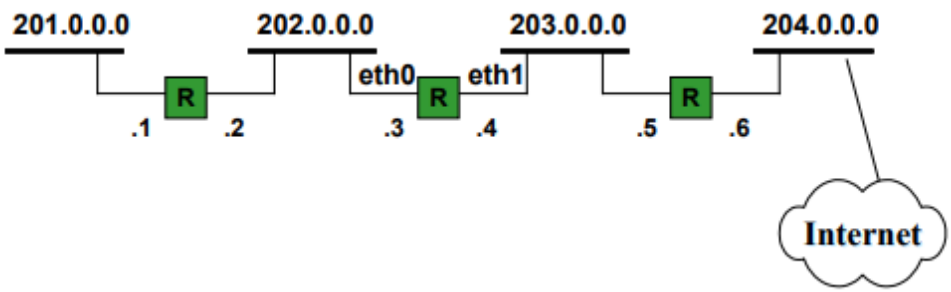
Descobre o endereço físico do gateway (ARP)

Transmite o datagrama para o gateway -> FIM

Fim

O exemplo abaixo ilustra uma estrutura de redes e a tabela de rotas dos roteadores. As tabelas de rotas de cada roteador são diferentes uma das outras. Note nestas tabela a existência de rotas diretas, que são informações redundantes para identificar a capacidade de acessar a própria rede na qual os roteadores estão conectados. Este tipo de rota apesar de parecer redundante é útil para mostrar de forma semelhante as rotas diretas para as redes conectadas diretamente no roteador.

Outra informação relevante é a existência de uma rota default. Esta rota é utilizada durante a decisão de roteamento no caso de não existir uma rota específica para a rede destino da mensagem IP. A rota default pode ser considerada como um resumo de diversas rotas encaminhadas pelo mesmo próximo roteador. Sem a utilização da rota default, a tabela de rotas deveria possuir uma linha para cada rede que pudesse ser endereçada. Em uma rede como a Internet isto seria completamente impossível.



A tabela de rotas para o roteador da esquerda é descrita abaixo:

Rede Destino	Roteador (Gateway)	Hops
201.0.0.0	eth0 (rota direta)	0
202.0.0.0	eth1 (rota direta)	0
203.0.0.0	202.0.0.3	1
204.0.0.0	203.0.0.3	2
default	203.0.0.3	--

A tabela de rotas para o roteador central é descrita abaixo:

Rede Destino	Roteador (Gateway)	Hops
202.0.0.0	eth0 (rota direta)	0
203.0.0.0	eth1 (rota direta)	0
201.0.0.0	202.0.0.2	1
204.0.0.0	203.0.0.5	1
default	203.0.0.5	--

A tabela de rotas para o roteador da direita é descrita abaixo:

Rede Destino	Roteador (Gateway)	Hops
203.0.0.0	eth0 (rota direta)	0
204.0.0.0	eth1 (rota direta)	0
202.0.0.0	203.0.0.4	1
201.0.0.0	203.0.0.4	1
default	204.0.0.7**	--

** Não mostrado na figura.

A rota default geralmente é representada nos sistemas operacionais como a rede 0.0.0.0

Roteamento Estático x Roteamento Dinâmico

A alimentação das informações na tabela de rotas pode ser de modo estático ou dinâmico ou ambos simultâneamente. Na alimentação estática, as rotas são preenchidas manualmente, geralmente pela configuração inicial da máquina. Na alimentação dinâmica, protocolos como RIP, RIP2, OSPF ou BGP4 são responsáveis pela aquisição de informações sobre a topologia da rede e a publicação de rotas na tabela de rotas dos roteadores envolvidos.

Como exemplos de rotas definidas estaticamente, pode-se citar:

· Uma rota default (ou roteador default) configurado manualmente nas estações (caso típico da maioria das estações-cliente em uma rede. P.ex., Janela de configuração básica de TCP/IP em Windows 3.1, Windows 95 e Windows NT

· Mais de uma rota default, com os roteadores configurados manualmente nas estações

· Rotas adicionais estáticas configuradas manualmente endereçando redes específicas. P.ex. Comando

route add dos sistemas operacionais Windows 95 e Windows NT

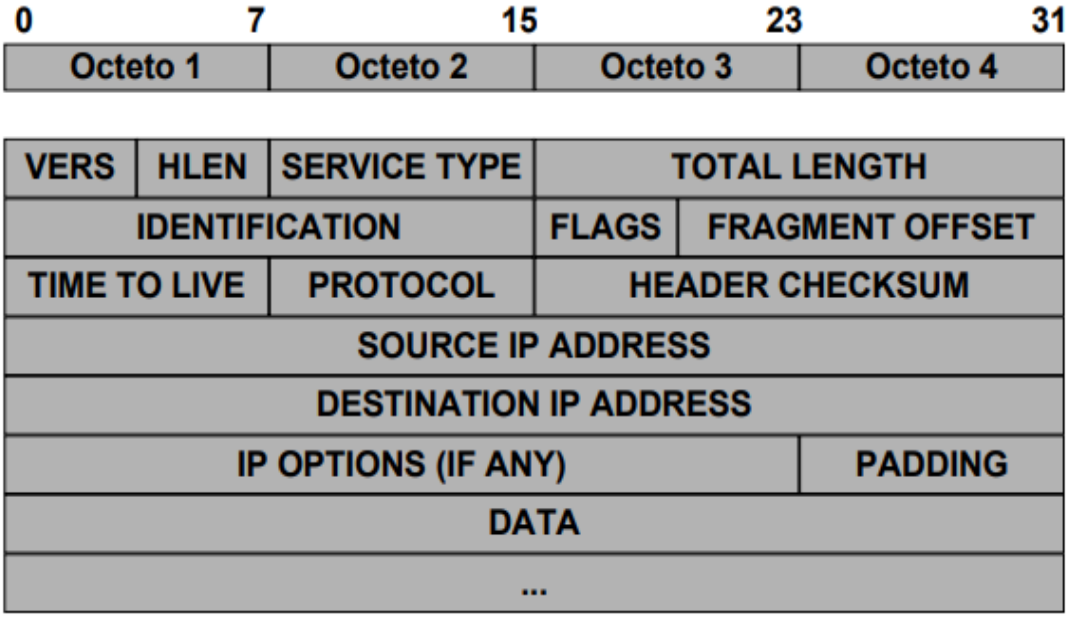
· Roteadores descobertos através do protocolo ICMP Router Advertisement

· Rotas informadas através do protocolo ICMP Redirect

Pacote IP

O protocolo IP define a unidade básica de transmissão, que é o pacote IP. Neste pacote são colocadas as informações relevantes para o envio deste pacote até o destino.

O pacote IP possui o formato descrito abaixo:



Os campos mais importantes são descritos abaixo:

VERSION - Informa a versão do protocolo IP sendo carregado. Atualmente a versão de IP é 4 HEADER LENGTH - Informa o tamanho do header IP em grupos de 4 bytes

TYPE OF SERVICE - Informa Como o pacote deve ser tratado, de acordo com sua prioridade e o tipo de serviço desejado como Baixo Retardo, Alta Capacidade de Banda ou Alta Confiablilidade. Normalmente este campo não é utilizado na Internet

IDENTIFICATION - Identifica o pacote IP unicamente entre os outros transmitidos pela máquina. Este campo é

usado para identificar o pacote IP no caso de haver fragmentação em múltiplos datagramas

FLAGS (3 bits) - um bit (MF - More Fragments) identifica se este datagrama é o último fragmento de um pacote IP ou se existem mais. Outro bit (DNF - Do Not Fragment) informa aos roteadores no caminho se a aplicação exige que os pacotes não sejam fragmentados.

FRAGMENT OFFSET - Informa o posicionamento do fragmento em relação ao pacote IP do qual faz parte.

TIME-TO-LIVE - Este valor é decrementado a cada 1 segundo que o pacote passa na rede e a cada roteador pelo quel ele passa. Serve para limitar a duração do pacote IP e evitar que um pacote seja roteador eternamente na Internet como resultado de um loop de roteamento.

PROTOCOL - Informa que protocolo de mais alto-nível está sendo carregado no campo de dados. O IP pode carregar mensagens UDP, TCP, ICMP, e várias outras.

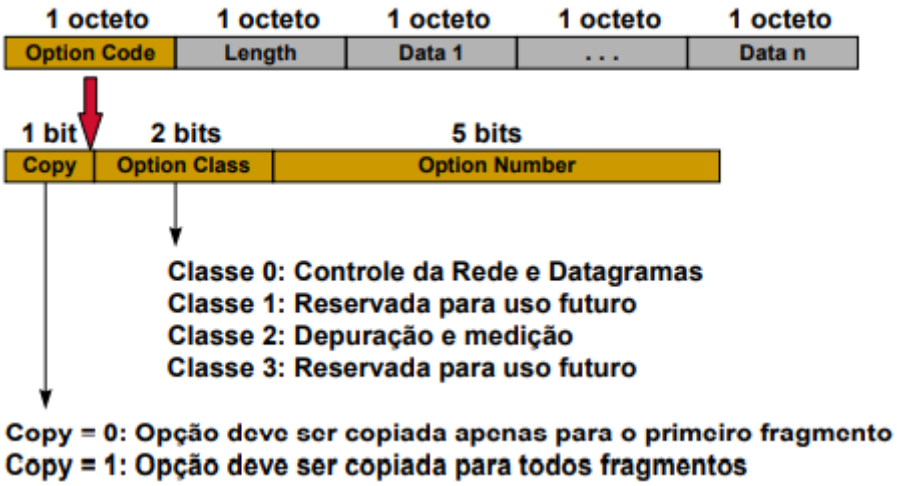
HEADER CHECKSUM - Valor que ajuda a garantir a integridade do cabeçalho do pacote IP

SOURCE ADDRESS - Endereço IP da máquina origem do pacote IP DESTINATION ADDRESS - Endereço IP da máquina destino do pacote IP

OPTIONS - Opções com informações adicionais para o protocolo IP. Consiste de um byte com a identificação da opção e uma quantidade de bytes variável com as informações específicas. Um pacote IP pode transportar várias opções simultaneamente.

Opções IP

O formato das opções IP é descrita no quadro abaixo:



As opções IP que podem ser utilizadas são:

Classe		Código	Composição	Descrição
0		0	--	Fim da Lista de Opções
0		1	--	Nenhuma Operação
0		3	variável	LOOSE SOURCE ROUTING (Especifica a rota aproximada que um datagrama deve seguir)
0		7	variável	RECORD ROUTE (Escreve os endereços dos roteadores por onde o pacote passou)
0		9	variável	STRICT SOURCE ROUTING (Especifica a rota exata que um datagrama deve seguir)
2		4	variável	INTERNET TIMESTAMP (A cada roteador grava a hora da passagem para outra rede)

As opções IP são utilizadas basicamente como forma de verificação e monitoração de uma rede IP. As opções que especificam a rota até o destino não são utilizadas normalmente pois o IP é baseado na técnica de Next- Hop routing. Ainda assim, estes mecanismos são pouco utilizados como ferramenta de testes e verificação, sendo raros os programas que os implementam.

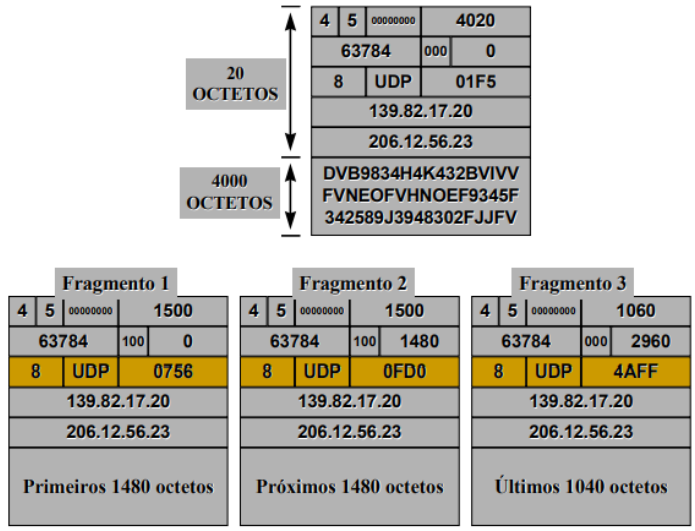
Fragmentação

Um pacote IP pode ter um tamanho de até 64 Kbytes. Entretanto o nível de rede geralmente tem um tamanho máximo menor que 64K. Por exemplo, uma rede Ethernet pode transmitir uma mensagem de até 1500 bytes.

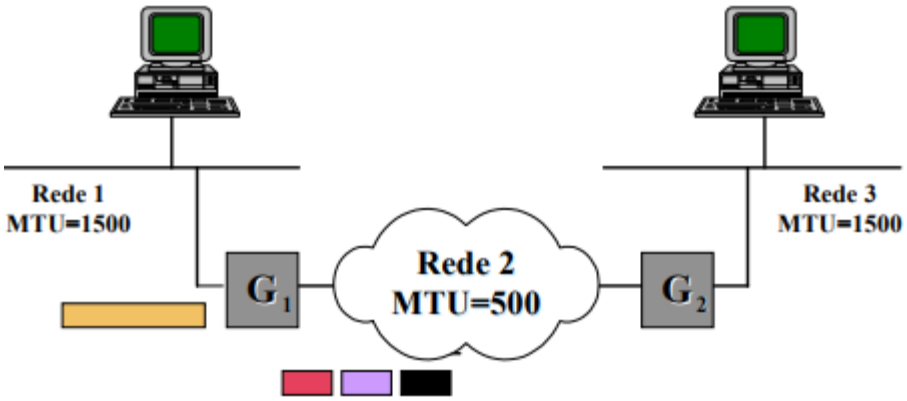
Este valor é chamado de MTU - Maximum Transmission Unit - para este tipo de rede. A camada IP deve então ser capaz de dividir um pacote IP maior que 1500 bytes em diversos fragmentos de até 1500 bytes cada um.

A fragmentação do pacote IP pode ocorrer na máquina origem ou em algum roteador que possua uma rede com MTU menor que o tamanho do pacote IP sendo roteado. Note que durante o percurso até o destino, um fragmento pode ser novamente fragmentado se o MTU da rede seguinte for ainda menor que o tamanho do fragmento. A remontagem do pacote só é realizada pela máquina destino, baseado nas informações de FRAGMENT OFFSET e bit MF. A perda de um fragmento inutiliza o datagrama inteiro.

O campo FRAGMENT OFFSET identifica a posição em Bytes do fragmento face ao pacote IP completo conforme pode ser visto nas figuras abaixo:



A figura abaixo mostra a fragmentação de um pacote quando este passa para uma rede com MTU menor que o tamanho do pacote IP.



Endereçamento em Subredes

A divisão de endereçamento tradicional da Internet em classes, causou sérios problemas de eficiência na distribuição de endereços. Cada rede na Internet, tenha ela 5, 200, 2000 ou 30 máquinas deveria ser compatível com uma das classes de endereços. Desta forma, uma rede com 10 estações receberia um endereço do tipo classe C, com capacidade de endereçar 256 estações.

Isto significa um desperdício de 246 endereços. Da mesma forma, uma rede com 2000 estações receberia uma rede do tipo classe B, e desta forma causaria um desperdício de 62000 endereços.

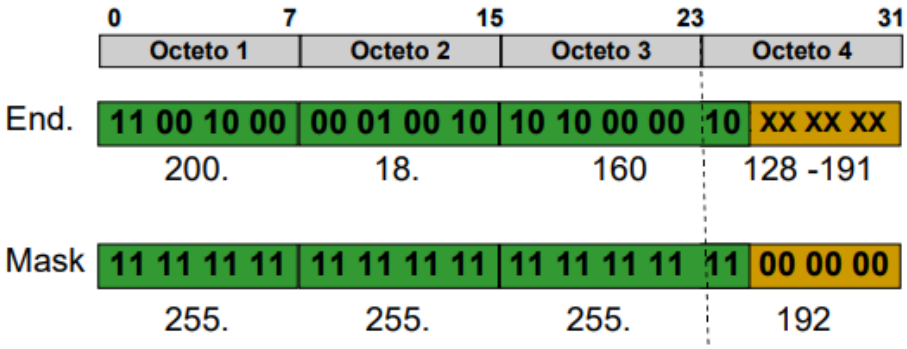
O número de redes interligando-se à Internet a partir de 1988 aumentou, causando o agravamento do problema de disponibilidade de endereços na Internet, especialmente o desperdício de endereços em classes C e B. Desta forma, buscou-se alternativas para aumentar o número de endereços de rede disponíveis sem afetar o funcionamento dos sistemas existentes. A melhor alternativa encontrada foi flexibilizar o conceito de classes - onde a divisão entre rede e host ocorre somente a cada 8 bits.

A solução encontrada foi utilizar a identificação de rede e host no endereçamento IP de forma variável, podendo utilizar qualquer quantidade de bits e não mais múltiplos de 8 bits conforme ocorria anteriormente. Um identificador adicional, a MÁSCARA, identifica em um endereço IP, que porção de bits é utilizada para identificar a rede e que porção de bits para host.

A máscara é formada por 4 bytes com uma sequência contínua de 1’s, seguida de uma sequência de 0’s. A porção de bits em 1 identifica quais bits são utilizados para identificar a rede no endereço e a porção de bits em 0, identifica que bits do endereço identificam a estação.

Obs. A máscara pode ser compreendida também como um número inteiro que diz a quantidade de bits um utilizados. Por exemplo uma máscara com valor 255.255.255.192, poderia ser representada como /26. Este tipo de notação é empregada em protocolos de roteamento mais recentes

Este mecanismo está representado na figura abaixo:



Neste endereço 200.18.160.X, a parte de rede possui 26 bits para identificar a rede e os 6 bits restantes para identificar os hosts. Desta forma, o endereço 200.18.160.0 da antiga classe C, fornecido a um conjunto de redes pode ser dividido em quatro redes com as identificações abaixo. Note que os 4 endereços de rede são independentes entre si. Elas podem ser empregadas em redes completamente separadas, e até mesmo serem utilizadas em instituições distintas.

200.18.160.[00XXXXXX]

200.18.160.[01XXXXXX]

200.18.160.[10XXXXXX] e

200.18.160.[11XXXXXX]

Em termos de identificação da rede, utiliza-se os mesmos critérios anteriores, ou seja, todos os bits de identificação da estação são 0. Quando os bits da estação são todos 1, isto identifica um broadcast naquela rede específica. Desta forma temos as seguintes identificações para endereço de rede:

200.18.160.0

200.18.160.64

200.18.160.128 e

200.18.160.192

Os endereços de broadcast nas redes são:

200.18.160.63

200.18.160.127

200.18.160.191 e

200.18.160.255

Os possíveis endereços de estação em cada rede são:

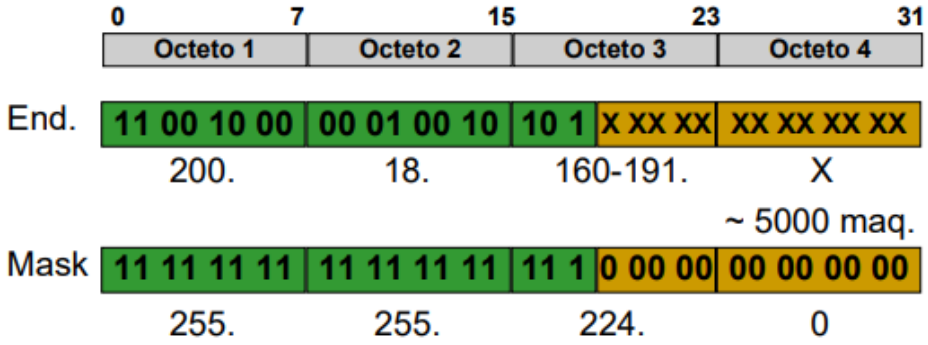
200.18.160.[1-62]

200.18.160.[65-126]

200.18.160.[129-190] e

200.18.160.[193-254]

O mesmo raciocínio de subrede pode ser usado para agrupar várias redes da antiga classe C em uma rede com capacidade de endereçamento de um maior número de hosts. A isto dá-se o nome de superrede. Hoje, já não há mais esta denominação pois não existe mais o conceito de classes. Um endereço da antiga classe A, como por exemplo 32.X.X.X pode ser dividido de qualquer forma por meio da máscara.



As máscaras das antigas classes A, B e C são um sub-conjunto das possibilidades do esquema utilizado atualmente, conforme mostrado abaixo:

Classe A: máscara equivalente = 255.0.0.0 Classe B: máscara equivalente = 255.255.0.0 Classe C: máscara equivalente = 255.255.255.0

Flexibilidade de Endereçamento

Uma conclusão que pode-se obter da análise acima é que uma identificação de uma rede, composta de um endereço de rede e uma máscara (p.ex. 200.18.171.64 e máscara 255.255.255.192) é, na verdade, um espaço de endereçamento, que pode ser usado da forma mais indicada. Por exemplo um espaço de endereçamento dado por

Rede = 32.10.20.128 com máscara 255.255.255.192

pode endereçar 64 endereços (62 endereços válidos) em uma rede só. Mas podemos subdividi-lo em subredes, de tal forma que poderemos ter:

1 rede de 64 endereços (usando o endereço e a máscara como estão)

2 redes de 32 endereços (aumentando em mais um bit a máscara)

Neste caso temos o endereço 32.10.20.128 dividido da seguinte forma:

```
Rede 1 = 32.10.20.128 com máscara 255.255.255.224 e
Rede 2 = 32.10.20.160 com máscara 255.255.255.224
```

Neste caso, cada rede formada pode ter até 30 endereços, pois deve-se sempre reservar os bits TODOS ZERO para o endereço de rede e os bits TODOS UM para o endereço de broadcast.

Desta forma, os endereços de máquina em cada rede são:

```
Rede 1: 32.10.20.[129-158]  e
Rede 2: 32.10.20.[161-190]
```

Note que deve-se sempre respeitar o espaço de endereçamento original. Um dos erros mais comuns é utilizar parte do endereçamento vizinho, o que está errado pois pertence a outro espaço de endereçamento.

Redes De 16 Endereços (Aumentando Em Dois Bits A Mascara Original)

Neste caso temos o endereço 32.10.20.128 dividido da seguinte forma:

```
Rede 1 = 32.10.20.128 com máscara 255.255.255.240
Rede 2 = 32.10.20.144 com máscara 255.255.255.240
Rede 3 = 32.10.20.160 com máscara 255.255.255.240
Rede 4 = 32.10.20.176 com máscara 255.255.255.240
```

Neste caso, cada rede formada pode ter até 14 estações

Então os endereços de máquina em cada rede são:

```
Rede 1: 32.10.20.[129-142]
Rede 2: 32.10.20.[145-158]
  Rede 3: 32.10.20.[161-174]
  Rede 4: 32.10.20.[177-190]
```

Note que o espaço de endereçamento original sempre se manteve, variando de 128 a 191

8 redes de 8 endereços

16 redes de 4 endereços (onde 4 endereços são na verdade duas estações, devido aos endereços reservados de rede e broadcast)

E só ! Pois 32 redes de 2 estações não existe pois seria uma rede sem nenhuma estação pois os dois endereços disponíveis já seriam utilizados para rede e broadcast.

Entretanto, as formas acima ainda não são as únicas formas de divisão do espaço de endereçamento. Pode-se dividir em mais uma dezena de forma, utilizando-se divisões do espaço de endereçamento de forma não homogênea. Um exemplo claro pode ser dado por exemplo observando redes reais, onde a quantidade de estações pode variar bastante entre cada uma.

Po exemplo, supondo que o espaço de endereçamento acima com capacidade de endereçar 64 estações deva ser utilizado em uma empresa com 50 estações. Estas 50 estações estão divididas em 3 redes, sendo uma com 30 estações e duas com 10 estações. Pode-se observar que nenhuma das formas de divisão acima são aceitáveis pois ou não comportam o número de redes necessárias (divisão em duas) ou não comportam o número de estações (divisão em 4).

A solução é realizar uma divisão do espaço de endereçamento de forma não homogênea. Isto é realizado de forma simples, utilizando metade do espaço original para a rede de 30 estações e dividindo o espaço restante em duas redes de 16 endereços.

De forma resumida, a divisão é da seguinte forma:

O espaço original; é dividido em dois, onde temos duas redes de 32 endereços:

```
Rede 1 = 32.10.20.128 com máscara 255.255.255.224
Rede 2 = 32.10.20.160 com máscara 255.255.255.224
```

Utiliza-se a rede 1 que possui os endereços de estação 32.10.20[129-158] para a rede com 30 estações. A rede 2 é na verdade um outro espaço de endereçamento (!) dado por 32.10.20.160 com máscara 255.255.255.224. Pode-se então dividir este espaço de endereçamento em duas rede, bastando aumentar um bit na máscara de rede:

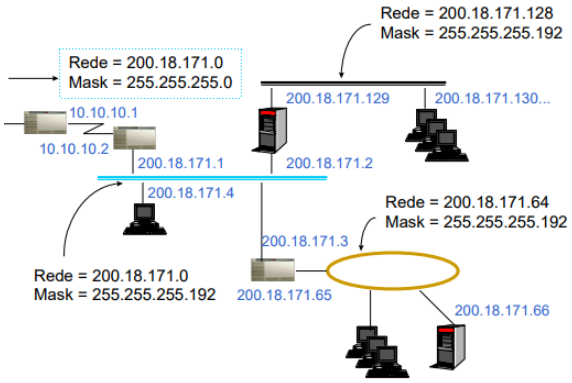
```
Rede 2 = 32.10.20.160 com máscara 255.255.255.240
Rede 3 = 32.10.20.176 com máscara 255.255.255.240
```

Então, o resultado final são três redes, onde a rede 2 possui os endereços de rede 32.10.20.[161-174] para estações e a rede 3 possui os endereços 32.10.20.[177-190] para as estações.

A figura abaixo mostra um exemplo de redes de uma empresa que ao se ligar à Internet, recebeu o espaço de endereçamento 200.18.171.0 com máscara 255.255.255.0 para ser utilizado em suas 3 redes internas. As rede possuem cada uma 50 estações, de modo que a divisão mais adequada é dividir o espaço em 4 redes de 64 endereços.

Neste caso o espaço de endereçamento 200.18.171.0 com máscara 255.255.255.0 foi dividido em três subredes, cada uma com capacidade de endereçar até 62 estações (64 endereços retirando o [000000] e o [111111]).

Note neste exemplo, que para a Internet, as três redes são vistas como uma só pois as rotas na Internet sempre se referem à rede 200.18.171.0 com máscara 255.255.255.0. Por isto os termos rede e espaço de endereçamento são utilizados de forma indistinta.



Roteamento Com Sub-Rede

Com a utilização de sub-rede, a tabela de rotas possui um campo adicional que é a mascara de rede, já que a identificação de uma rede possui uma máscara.

No caso do exemplo anterior, um roteador qualquer na Internet que conecte este conjunto de redes à Internet possui apenas uma rota para a rede 200.18.171.0, com máscara 255.255.255.0, endereçada para o roteador

10.0.0.1. Isto mostra que a informação roteamento das diversas sub-redes pode ser agregada em uma única linha na tabela de rotas.

Por exemplo apesar de possuir centenas de redes, os roteadores na Internet possuem uma única linha para a PUC, sendo a rede destino 139.82.0.0 e a máscara 255.255.0.0. somente dentro da PUC, os roteadores internos devem saber distinguir as diversas sub-redes formadas.

No exemplo anterior, o roteador interna da empresa não pode ter uma rota genérica para a rede 200.18.171.0, mas precisa saber endereçar as diversas sub-redes. Isto se dá pela utilização de rotas associadas a máscara. A tabela abaixo mostra a tabela de rotas deste roteador:

Rede Destino	Máscara	Roteador (Gateway)	Hops
200.18.171.0	255.255.255.192	200.18.171.1 (eth0)	0
10.0.0.0	255.0.0.0	10.0.0.1 (serial1)	0
200.18.171.64	255.255.255.192	200.18.171.3	1
200.18.171.128	255.255.255.192	200.18.171.2	1
default	0.0.0.0	10.0.0.2	--

A tabela de rotas do roteador inferior é dada pela tabela abaixo:

Rede Destino	Máscara	Roteador (Gateway)	Hops
200.18.171.0	255.255.255.192	200.18.171.3 (eth0)	0
200.18.171.64	255.255.255.192	200.18.171.65 (eth1)	0
200.18.171.128	255.255.255.192	200.18.171.2	1
default	0.0.0.0	200.18.171.1	--

A máscara de rede faz parte de toda tabela de rotas.

O algoritmo de Recepção de pacote IP e roteamento com a introdução da máscara de sub-rede fica alterado conforme abaixo:

Datagrama recebido da camada intra-rede, defragmentado e testado

Caso:

Endereço Destino = Endereço do Host, ou E.D. = outras interfaces do Host, ou E.D. = Broadcast

Passa datagrama para níveis superiores -> FIM

Caso:

Máquina que recebeu não é roteador

Descarta datagrama -> FIM

Máquina é roteador (possui mais de uma interface IP)

Caso:

Endereço de rede IP destino = Alguma das Redes IP com interface direta 2.2.2.1.1.1 Descobre o endereço físico do destino (ARP)

2.2.2.1.1.2 Transmite datagrama pela interface respectiva -> FIM

Faz um AND lógico bit-a-bit do endereço IP com as máscaras de cada rede da tabela de rotas e compara com o endereço de rede da rota respectiva

Se algum conferir, descobriu uma rota

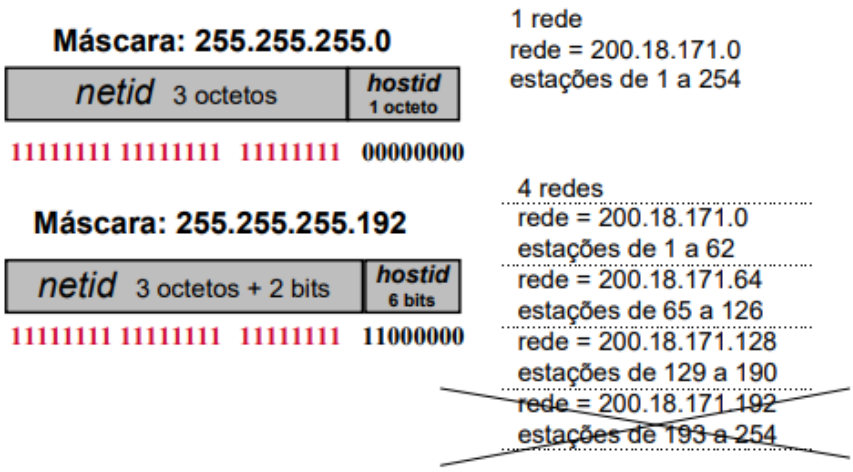
2.2.2.1.3.1 Verifica na tabela de rotas o endereço IP do roteador destino desta rota. 2.2.2.1.3.2 Descobre o endereço físico do gateway (ARP)

2.2.2.1.3.3 Transmite o datagrama para o gateway -> FIM

Sub-Redes Não Utilizáveis:

Devido a motivos históricos do desenvolvimento de TCP/IP, a divisão em sub-redes tem algumas restrições quanto a utilização de algumas sub-redes. Basicamente, não se pode utilizar o endereçamento que contêm todos os bits UM da porção da sub-rede. As implementações mais novas permitem que este endereçamento seja utilizado.

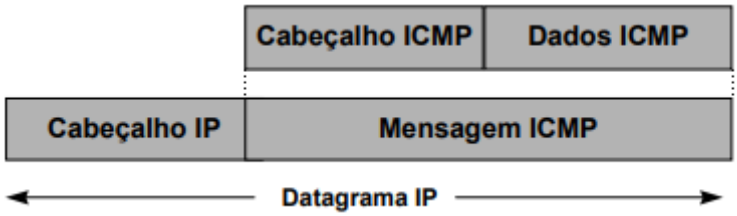
A figura abaixo ilustra esta restrição na utilização da sub-rede com os dois bits 11, para o caso da máscara 255.255.255.192. No caso da utilização da máscara 255.255.255.224, não se deve utilizar a sub-rede com bits 111.



Protocolo ICMP

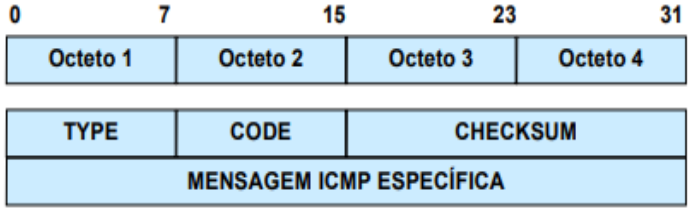
O protocolo ICMP é um protocolo auxiliar ao IP, que carrega informações de controle e diagnóstico, informando falhas como TTL do pacote IP expirou, erros de fragmentação, roteadores intermediários congestionados e outros.

Uma mensagem ICMP é encapsulada no protocolo IP, conforme ilustrado na figura abaixo. Apesar de encapsulado dentro do pacote IP, o protocolo ICMP não é considerado um protocolo de nível mais alto.



A mensagem ICMP é sempre destinada ao host origem da mensagem, não existindo nenhum mecanismo para informar erros aos roteadores no caminho ou ao host destino.

As mensagens ICMP possuem um identificar principal de tipo (TYPE) e um identificador de sub-tipo (CODE), conforme pode ser visto no formato de mensagem ilustrado abaixo:



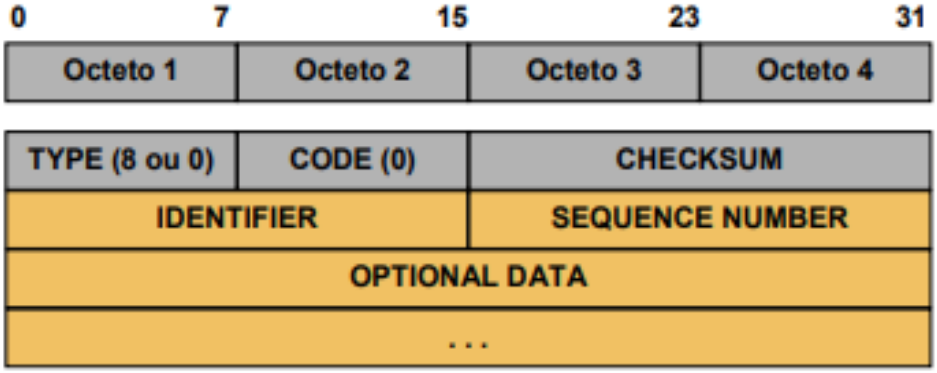
Os tipos de mensagem ICMP são listados na tabela abaixo:

Tipo	Mensagem ICMP	Categoria
0	Echo Reply	Controle
3	Destination Unreachable	Erro
4	Source Quench	Controle
5	Redirect	Controle
8	Echo Request	Controle
9	Router Advertisement (RFC 1256)	Controle
10	Router Solicitation (RFC 1256)	Controle
11	Time Exceeded for a Datagram	Erro
12	Parameter Problem on a Datagram	Erro
13	Timestamp Request	Controle
14	Timestamp Reply	Controle
15	Information Request (obsoleto)	Controle
16	Information Reply (obsoleto)	Controle
17	Address Mark Request	Controle
18	Address Mark Reply	Controle

As mensagens ICMP são listadas abaixo:

Echo Request e Echo Reply

Utilizada pelo comando ping, a mensagem Echo Request enviada para um host causa o retorno de uma mensagem Echo Reply. É utilizada principalmente para fins de testes de conectividade entre as duas máquinas.



Destination Unreacheable

Esta mensagem possui diversos sub-tipos para identificar o motivo da não alcançabilidade: os sub-tipos utilizados atualmente são:

: Network Unreachable - Rede destino inalcançável

: Host Unreachable (ou falha no roteamento para subnet) - Máquina destino inalcançável

: Protocol Unreachable - Protocolo destino desativado ou aplicação inexistente

: Port Unreachable - Porta destino sem aplicação associada

: Fragmentation Needed and DNF set - Fragmentação necessária mas bit DNF setado.

Alterado também pela RFC 1191 para suporta o protocolo Path MTU Discovery

: Source Route Failed - Roteamento por rota especificada em opção IP falhou

: Destination Network Unknown 7: Destination Host Unknown

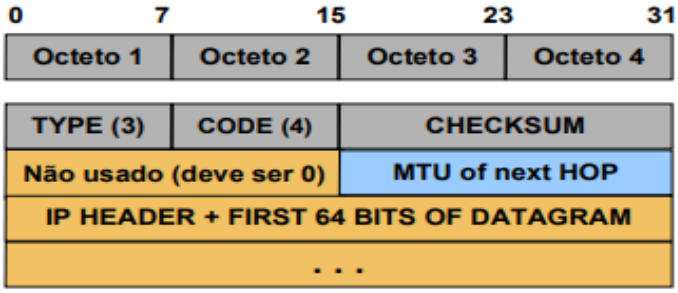
: Source Host Isolated

: Communication with destination network administratively prohibited 10 : Communication with destination host administratively prohibited

O sub-tipo Fragmentation Needed and DNF set é utilizado como forma de um host descobrir o menor MTU nas redes que serão percorridas entre a origem e o destino. Por meio desta mensagem, é possível enviar pacotes que não precisarão ser fragmentados, aumentando a eficiência da rede. Esta técnica, que forma um protocolo é denominado de ICMP MTU Discovery Protocol, definido na RFC 1191.

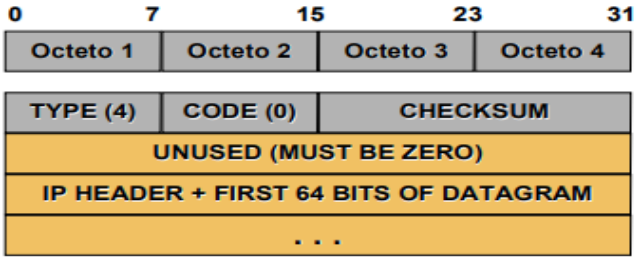
A operação é simples. Todo pacote IP enviado é marcado com o bit DNF (Do Not Fragment), que impede sua fragmentação nos roteadores. Desta forma, se uma pacote IP, ao passar por um roteador para chegar a outra rede com MTU menor, deva ser fragmentado, o protocolo IP não irá permitir e enviará uma mensagem ICMP Destination Unreacheable para o destino.

Para suportar esta técnica, a mensagem ICMP foi alterada para informar o MTU da rede que causou o ICMP. Desta forma, a máquina origem saberá qual o valor de MTU que causou a necessidade de fragmentação, podendo reduzir o MTU de acordo, nos próximos pacotes. Esta mensagem está ilustrada abaixo:



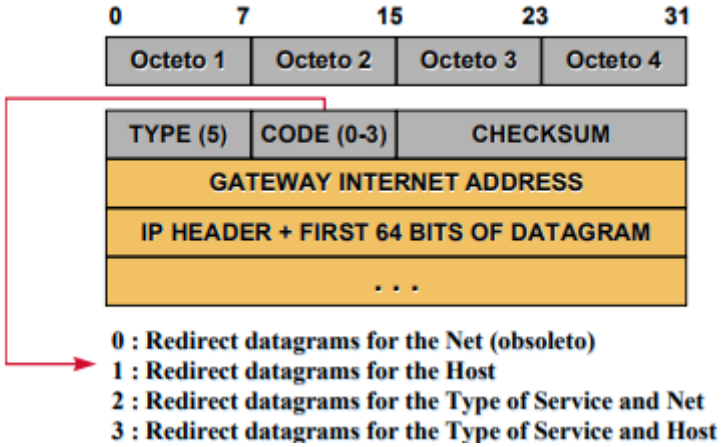
Source Quench

Esta mensagem é utilizada por um roteador para informar à origem, que foi obrigado a descartar o pacote devido a incapacidade de roteá-lo devido ao tráfego.

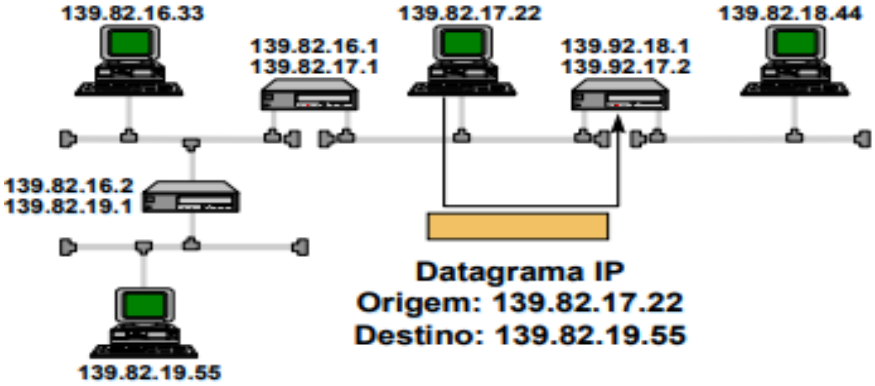


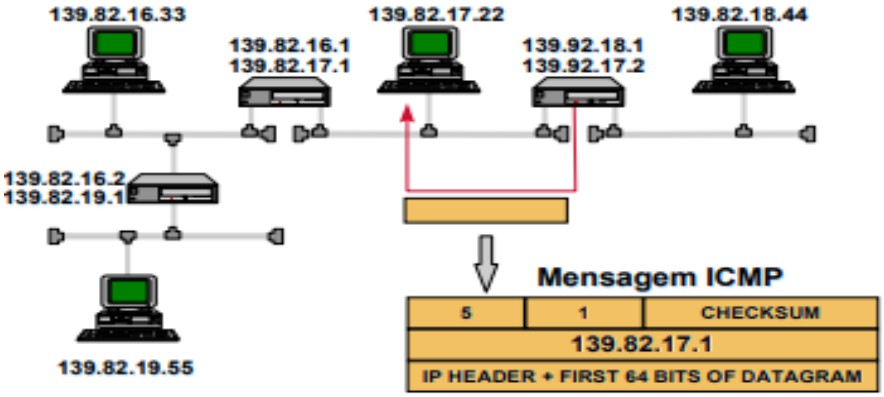
Redirect

Esta mensagem, uma das mais importantes do protocolo IP, é utilizada por um roteador para informar ao host origem de uma mensagem que existe uma rota direta mais adequada através de outro roteador. O host, após receber a mensagem ICMP, instalará uma rota específica para aquele host destino:



A operação do ICMP Redirect ocorre conforme os diagramas abaixo. Note que a rota instalada é uma rota específica para host, com máscara 255.255.255.255, não servindo para outras máquinas na mesma rede. Se uma máquina se comunica com 10 máquinas em outra rede e se basear em ICMP Redirect para aprender as rotas, ele instalará pelo menos 10 entradas na tabela de rede, uma para cada máquina



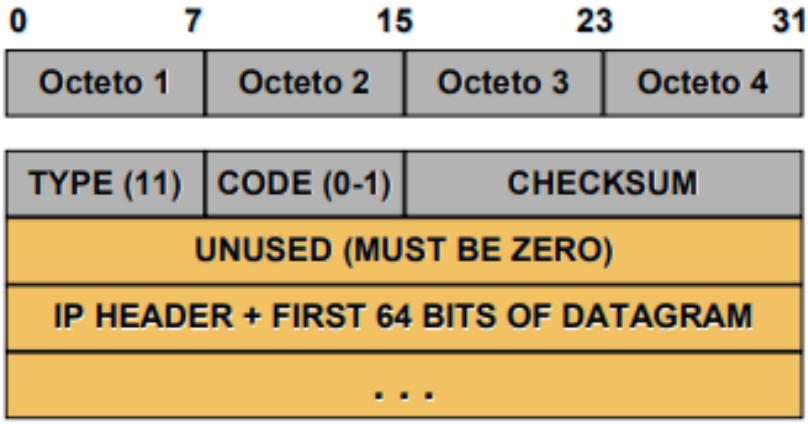


Na figura acima, a estação 139.82.17.22 instalou, após a mensagem ICMP, a seguinte rota na tabela de rotas:

Rede Destino	Máscara	Roteador (Gateway)	Hops
139.82.19.55	255.255.255.255	139.82.17.1	0

TTL Expired

Esta mensagem ICMP originada em um roteador informa ao host de origem que foi obrigado a descartar o pacote, uma vez que o TTL chegou a zero.



Esta mensagem é utilizada pelo programa traceroute (ou tracert no Windows) para testar o caminho percorrido por um pacote. O programa funciona da seguinte forma:

É enviada uma mensagem ICMP Echo Request para um endereço IP destino. Esta mensagem é enviada com TTL = 1.

Quando chega ao primeiro roteador, este decrementa o valor de TTL da mensagem IP e retorna uma mensagem ICMP TTL Expired. O programa armazena o endereço IP do roteador que enviou a mensagem TTL Expired.

O programa envia outra mensagem ICMP Echo Request para o endereço IP destino. Esta mensagem é enviada desta vez com TTL=2.

A mensagem atravessa o primeiro roteador e tem o TTL decrementado para 1. Quando chega ao segundo roteador, o TTL torna-se 0 e este roteador envia uma mensagem ICMP TTL Expired para a máquina origem. Esta armazena o endereço do segundo roteador.

Esta operação prossegue até que a máquina destino responda. Todos os roteadores no caminho são registrados.

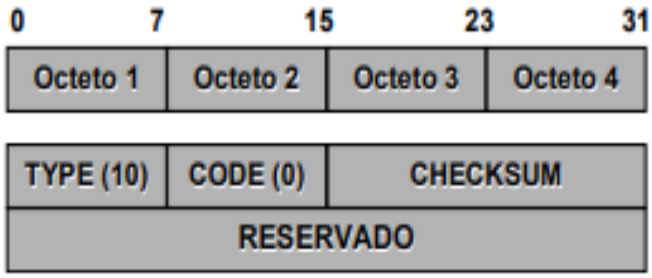
Note, entretanto, que devido à diferenças de rotas seguidas pelos diversos pacotes, o caminho obtido não necessariamente é único. A execução do programa traceroute mais de uma vez pode revelar rotas diferentes seguidas pelos pacotes.

ICMP Router Solicitation/Advertisement

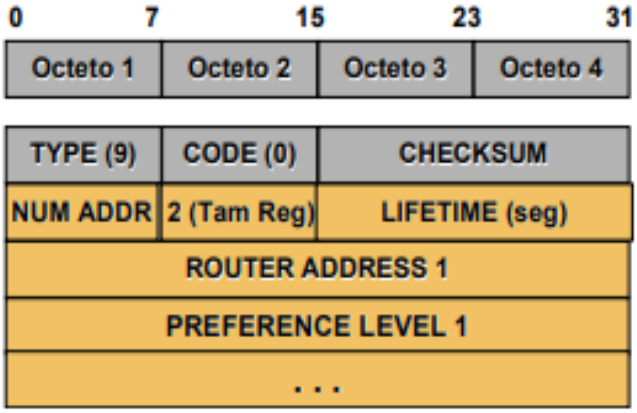
Esta variação de ICMP, definido na RFC 1256 foi projetada para permitir que um roteador possa divulgar sua existência para as máquinas existentes na rede. O objetivo desta função é evitar a necessidade de se configurar manualmente todas as estações da rede com a rota default e permitir que uma estação conheça outros roteadores além do default que possam rotear no caso de falha do principal.

A mensagem é composta de duas formas: a solicitação de divulgação de uma roteador e o anúncio de um roteador. O roteador pode ser configurado para enviar automaticamente as mensagens de anúncio ou fazê-lo apenas comandado por uma mensagem de solicitação.

A mensagem ICMP Router Solicitation é mostrada abaixo:



A mensagem ICMP Router Solicitation é mostrada abaixo:



Esta mensagem pode conter a divulgação de diversos roteadores iniciada a partir de um que seja configurado para divulgá-los. O número de preferência é a ordem de preferência que estes roteadores podem ser utilizados pelas estações.

Aquisição de informações de roteamento

Em uma estação e em um roteador, as informações constantes na tabela de rotas podem ser obtidas de diversas formas.

As rotas podem ser obtidas por uma estação ou em um roteador de diversas formas, com limitações dependendo da implementação do TCP/IP em cada sistema operacional:

Estação sem nenhuma rota. Neste caso, a estação vai precisar de pelo menor um roteador default. A estação pode obter um roteador default através de:

protocolo ICMP Router Advertisement

Protocolo BOOTP ou DHCP durante a etapa de boot ou após ela.

Escuta dos protocolos de roteamento como RIP e outras para descobrir roteadores

outras, sempre não respeitando a divisão em camadas

Estação com somente um roteador default. Com um roteador, a estação já pode operar corretamente. No caso de existir rotas melhores através de outros roteadores, o roteador default informará rotas específicas através de ICMP Redirect, sempre específica para uma estação destino.

Estação com mais de um roteador default, poderá utilizar os diversos roteadores default, no caso de falha do primeiro.

Estação com rotas específicas para outras redes configuradas de forma manual.

Estação executando algum protocolo de roteamento, geralmente na forma SOMENTE ESCUTA. Desta forma, a estação pode aprender informações de rotas trocadas entre os roteadores sem divulgar rotas.

É possível inclusive ocorrer o recebimento de informações conflitantes ou não idênticas de rotas para determinadas redes. O roteador resolve estes conflitos com a adoção de prioridades para rotas aprendidas por meios diferentes. Geralmente, a ordem de prioridade da forma de aprendizagem das rotas é da seguinte forma:

Rotas configuradas estaticamente tem maior prioridade, exceto se houver outra rota mais específica (com máscara mais longa). P. exemplo, um roteador possui uma rota para a rede 200.0.0.0 mas aprende uma rotas específica para 200.0.0.123. Esta última terá maior prioridade

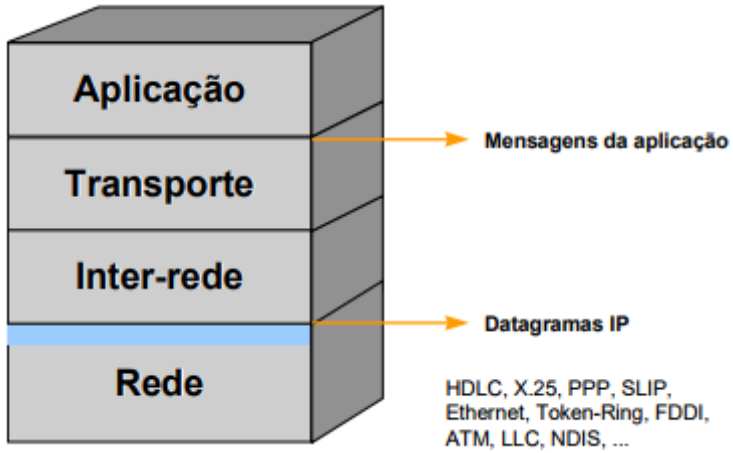
Rotas específicas aprendidas por meio de ICMP Redirect e rotas default aprendidas por meio de ICMP Router Advertisement

Rotas aprendidas por meio dos protocolos OSPF e BGP

Rotas aprendidas por meio do protocolo RIP

Protocolos da Camada de Transporte

A figura 1 ilustra a divisão em camadas da arquitetura TCP/IP:



Camada de Transporte

Esta camada reúne os protocolos que realizam as funções de transporte de dados fim-a-fim, ou seja, considerando apenas a origem e o destino da comunicação, sem se preocupar com os elementos intermediários. A camada de transporte possui dois protocolos que são o UDP (User Datagram Protocol) e TCP (Transmission Control Protocol).

O protocolo UDP realiza apenas a multiplexação para que várias aplicações possam acessar o sistema de comunicação de forma coerente.

O protocolo TCP realiza além da multiplexação, uma série de funções para tornar a comunicação entre origem e destino mais confiável. São responsabilidades do protocolo TCP o controle de fluco, o controle de erro, a sequenciação e a multiplexação de mensagens.

A camada de transporte oferece para o nível de aplicação um conjunto de funções e procedimentos para acesso ao sistema de comunicação de modo a permitir a criação e a utilização de aplicações de forma independente da implementação. Desta forma, as interfaces socket (ambiente Unix) e Winsock (ambiente Windows) fornecem um conjunto de funções-padrão para permitir que as aplicações possam ser desenvolvidas independentes do sistema operacional no qual rodarão.

Protocolo UDP

O protocolo UDP fornece uma forma simples de acesso ao sistema de comunicação, provendo um serviço sem conexão, sem confiabilidade e sem correção de erros. A principal função do nível de transporte implementada em UDP é a capacidade de multiplexação de acesso ao sistema de comunicação. Esta função permite que vários processos ou programas executando em um computador possam acessar o sistema de comunicação e o tráfego de dados respectivo a cada um deles seja corretamente identificado, separado e utilize buffers individuais.

Um processo é o programa que implementa uma aplicação do sistema operacional, e que pode ser uma aplicação do nível de aplicação TCP/IP.

A forma de identificação de um ponto de acesso de serviço (SAP) do modelo OSI é a porta de protocolo em TCP/IP. A porta é a unidade que permite identificar o tráfego de dados destinado a diversas aplicações. A identificação única de um processo acessando os serviços TCP/IP é, então, o endereço IP da máquina e a porta (ou portas) usadas pela aplicação.

Cada processo pode utilizar mais de uma porta simultâneamente, mas uma porta só pode ser utilizada por uma aplicação em um dado momento. Uma aplicação que deseje utilizar os serviços de comunicação deverá requisitar uma ou mais portas para realizar a comunicação. A mesma porta usada por uma aplicação pode ser usada por outra, desde que a primeira tenha terminado de utilizá-la.

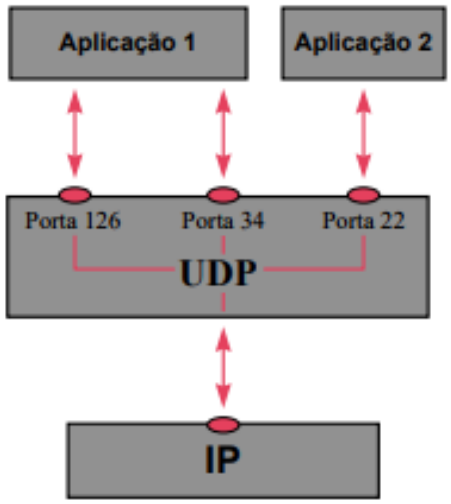
A forma de utilização de portas mostra uma distinção entre a parte cliente e a parte servidora de uma aplicação TCP/IP. O programa cliente pode utilizar um número de porta qualquer, já que nenhum programa na rede terá necessidade de enviar uma mensagem para ele.

Já uma aplicação servidora deve utilizar uma número de porta bem-conhecido (Well-known ports) de modo que um cliente qualquer, querendo utilizar os serviços do servidor, tenha que saber apenas o endereço IP da máquina onde este está executando.

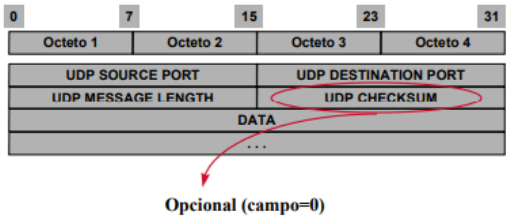
Se não houvesse a utilização de um número de porta bem conhecido, a arquitetura TCP/IP deveria possuir um mecanismo de diretório para que um cliente pudesse descobrir o número da porta associado ao servidor. Para evitar este passo intermediário, utiliza-se números de porta bem conhecidos e o cliente já possui pré programado em seu código o número de porta a ser utilizado.

Os números de porta de 1 a 1023 são números bem-conhecidos para serviços (aplicações) atribuídos pela IANA (Internet Assigned Numbers Authority). Os números de 1024 a 65535 podem ser atribuídos para outros serviços e são geralmente utilizados pelas programas-cliente de um protocolo (que podem utilizar um número de porta qualquer). Este conjunto de números tem ainda a atribuição de alguns serviços de forma não oficial, já que os primeiros 1024 números não conseguem comportar todos os protocolos TCP/IP existentes.

A figura abaixo ilustra a multiplexação/demultiplexação realizada pelo protocolo UDP, camada de transporte:



Formato Da Mensagem UDP



Protocolo TCP

O protocolo TCP trabalha no mesmo nível que o protocolo UDP, mas oferece serviços mais complexos, que incluem controle de erros e fluxo, serviço com conexão e envio de fluxo de dados. TCP utiliza o mesmo conceito de porta de UDP. Para TCP, uma conexão é formada pelo par (End. IP. Origem, Porta Origem) e (End. IP Destino, Porta Destino).

O protocolo TCP oferece as seguintes características:

Controle de Fluxo e Erro fim-a-fim

Serviço confiável de transferência de dados

Comunicação full-duplex fim-a-fim

A aplicação basta enviar um fluxo de bytes

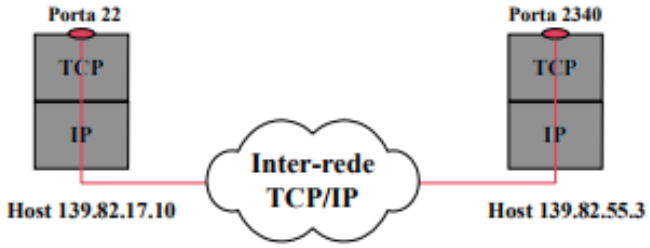
Desassociação entre qtd. de dados enviados pela aplicação e pela camada TCP

Ordenação de mensagens

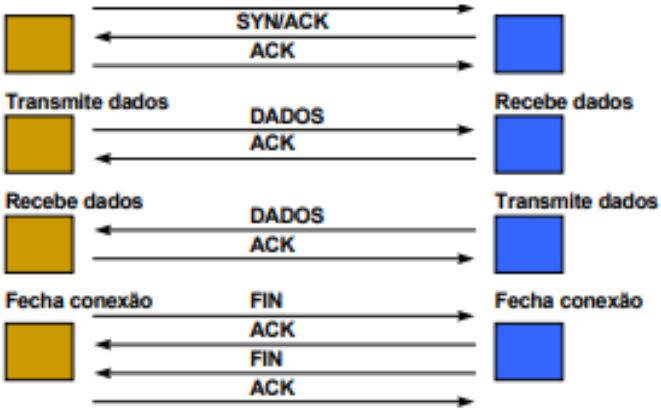
Multiplexação de IP, através de várias portas

Opção de envio de dados urgentes

A conexão TCP é ilustrada na figura abaixo:

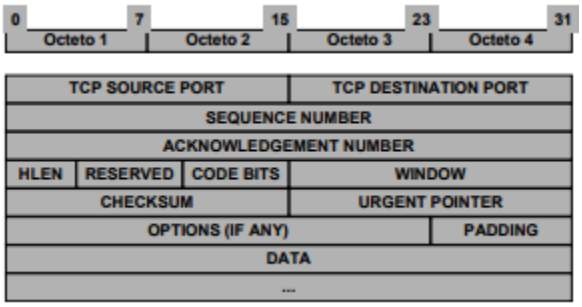


Uma conexão TCP é formada por três fases: o estabelecimento de conexão, a troca de dados e o finalização da conexão, conforme ilustrado na figura abaixo:



A fase inicial de estabelecimento de conexão é formada de três mensagens, formando o three-way-hanshaking, conforme a figura abaixo:

O pacote TCP é formado pela mensagem mostrada abaixo:



Estes campos são definidos da seguinte forma:

TCP SOURCE PORT: Porta origem da mensagem

TCP DESTINATION PORT: Porta destino da mensagem

SEQUENCE NUMBER: número de sequência dos dados sendo transmitidos face ao conjunto total de dados já transmitidos. Este número indica a posição do primeiro byte de dados sendo transmitido em relação ao total de bytes já transmitidos nesta conexão. O primeiro número de sequência utilizado não é zero ou um, mas começa de um valor aleatório.

Logo se um pacote está trasmitindo do 1234o. byte até o 2000o. byte de uma conexão e o SEQUENCE NUMBER inicial utilizado nesta conexão foi 10000, o campo SEQUENCE NUMBER conterá o valor 11234. O sequence number em um sentido da conexão (máquina A para B) é diferente do seuqnece number do sentido inverso, já que os dados transmitidos por um e outro lado são completamente distintos.

ACKNOWLEDGE NUMBER: número que significa o reconhecimento dos dados recebidos até então no sentido inverso. O ACK de um sentido é transmitido em piggy-backing no outro sentido. O ACK contém o número do próximo byte do fluxo de dados recebido, que a origem deste pacote espera receber da outra máquina. Este valor leva em consideração o número de SEQUENCE NUMBER inicial praticado pela outra máquina. O valor de ACK informa sempre o próximo byte ainda não recebido do conjunto contíguo de bytes recebidos do transmissor.

CODE BITS: São formados por seis bits, URG, ACK, PSH, RST, SYN e FIN, cuja utilização é mostrada abaixo:

URG: bit de Urgência: significa que o segmento sendo carregado contém dados urgentes que devem ser lidos com prioridade pela aplicação. A aplicação origem é responsável por acionar este bit e fornecer o valor do URGENT POINTER que indica o fim dos dados urgentes. Um exemplo da utilização desta facilidade é o aborto de uma conexão (por exemplo por Control-C), que faz com que a aplicação destino examine logo o pacote até o fim da área de urgência, descubra que houve um Control-C e termine a conexão.

ACK: bit de Reconhecimento: indica que o valor do campo de reconhecimento está carregando um reconhecimento válido.

PSH: bit de PUSH: Este mecanismo que pode ser acionado pela aplicação informa ao TCP origem e destino que a aplicação solicita a transmissão rápida dos dados enviados, mesmo que ela contenha um número baixo de bytes, não preeenchendo o tamanho mínimo do buffer de transmissão.

RST: bit de RESET: Informa o destino que a conexão foi abortada neste sentido pela origem

SYN: bit de Sincronismo: ë o bit que informa que este é um dos dois primeiros segmentos de estabelecimento da conexão.

FIN: bit de Terminação: indica que este pacote é um dos dos pacotes de finalização da conexão WINDOW: Este campo informa o tamanho disponível em bytes na janela de recepção da origem deste pacote. Por meio deste valor, o TCP pode realizar um controle adequando de fluxo para evitar a sobrecarga do receptor.

Quando este valor é igual a zero, o transmissor não envia dados, esperando receber um pacote com WINDOW maior que zero. O transmissor sempre vai tentar transmitir a quantidade de dados disponíveis na janela de recepção sem aguardar um ACK. Enquanto não for recebido um reconhecimento dos dados transmitidos e o correspondente valor de WINDOW > 0, o transmissor não enviará dados.

OPTIONS: O campo de opções só possui uma única opção válida que é a negociação do MSS (Maximum Segment Size) que o TCP pode transmitir. O MSS é calculado através do MTU ou através do protocolo ICMP Path MTU Discovery.

Protocolos Da Camada De Rede E Protocolos Auxiliares De TCP/IP

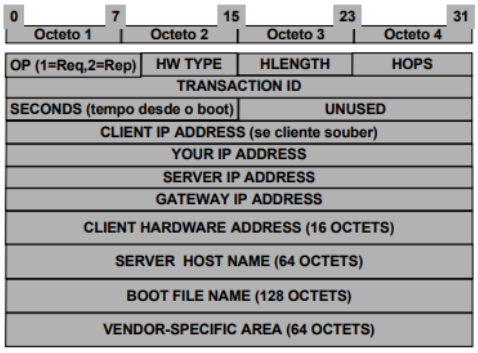
Este protocolos são agrupados neste capítulo, por fornecerem serviços auxiliares para TCP/IP, tanto a nível de enlace OSI quanto a nível a de aplicação.

BOOTP e DHCP

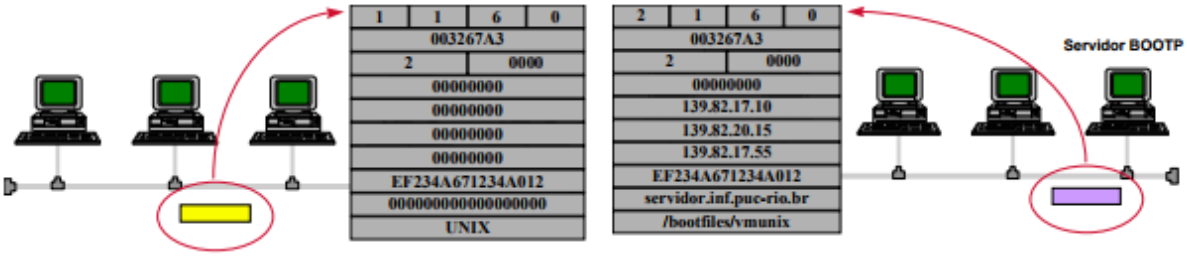
Este protocolos fornecem aos protocolos TCP/IP, as informações iniciais de configuração da máquina tais como endereço IP, máscara de sub-rede, roteadores default, rotas, servidores de Boot, servidores de nome e diversas outras informações. Eles são utilizados principalmente para realizar a administração centralizada de máquinas TCP/IP e possibilitar o BOOT de máquinas sem rígido e sem informações iniciais de configuração. O BOOTP (Bootstrap Protocol) é o protocolo mais antigo e o DHCP (Dynamic Host Control Protocol) está aos poucos o substituindo. O BOOTP é bastante utilizado para o boot inicial de dispositivos de rede, como roteadores, switches, hubs gerenciáveis, além de estações Unix diskless (sem disco e cada vez mais raras hoje). O DHCP é um pouco mais complexo e mais versátil e vem sendo utilizado principalmente para simplificar a administração de endereços e outros parâmetros de configuração de grandes instalações de máquinas TCP/IP.

Protocolo BOOTP

A mensagem BOOTP é encapsulada em UDP e possui o seguinte formato:



As mensagens BOOTP Request e BOOTP Reply tem o mesmo formato mas no Request alguns campos não são preeenchidos. Uma estação que deseja obter informações de configuração pode enviar uma mensagem BOOTP Request por broadcast. Um servidor de BOOTP pré configurado na rede com os parâmetros de cada cliente, receberá a mensagem e enviará os dados previamente armazenados para o cliente. Este procedimento é mostrado nas duas figuras abaixo:



Na área Vendor-Specific da mensagem BOOTP podem ser colocadas uma série de variáveis possíveis adicionais para configuração da estação cliente de BOOTP. Estas opções são definidas em RFCs adicionais e servem tanto para BOOTP quanto para DHCP.

Protocolo DHCP

O DHCP tem como principal vantagem em relação ao BOOTP a sua capacidade de configuração automática de estações, sem necessidade de criação de uma tabela de configuração para cada máquina (com seus parâmetros e endereços MAC respectivos, como é o caso de BOOTP). Desta forma, um administrador de rede pode configurar as diversas estações IP existentes na rede de modo genérico, sem especificar uma tabela para cada uma.

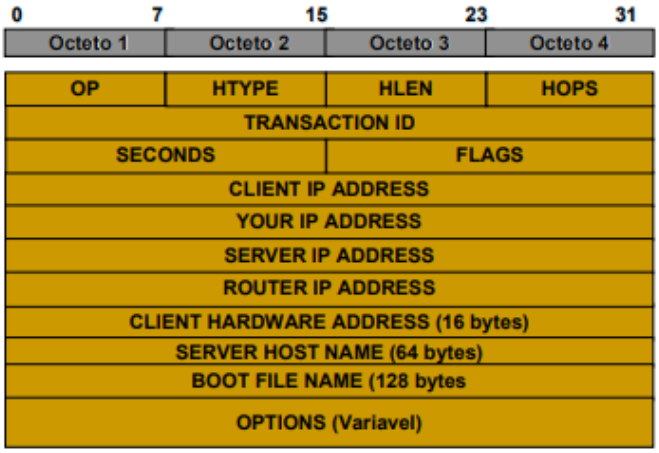
O DHCP tem a capacidade de distribuir endereços de forma dinâmica para as estações, usando três métodos de fornecimento distintos:

Empréstimo (leasing) de endereço aleatório por tempo limitado: Neste tipo de fornecimento de endereço IP, o servidor fornece ao cliente um endereço IP obtido de um conjunto pré-definido de endereços (p.ex. 192.168.0.10 a 192.168.0.90) por um tempo pré determinado.

Empréstimo de endereço aleatório por tempo infinito: Neste tipo, o servidor associa um endereço obtido do conjunto de endereços a um cliente na primeira vez que este cliente contactar o servidor. Nas demais vezes, será fornecido o mesmo endereço a este cliente (associado através do endereço MAC), mesmo que as duas máquinas sejam desligadas e ligadas. Este método simplifica a atribuição de endereços para uma quantidade grande de máquinas.

Empréstimo de endereço fixo: Neste tipo de fornecimento, o DHCP opera como o BOOTP, onde há a associação explícita entre o endereço IP e o endereço MAC da máquina origem, estipulado em uma tabela de configuração

A mensagem DHCP é compatível com BOOTP e possui o formato abaixo:



Ao contrário da mensagem BOOTP que possui apenas dois tipos de comandos (REQUEST e REPLY), a mensagem DHCP possui 8 tipos de comandos. Este comandos não são colocados no campo OP, como em BOOTP, mas para manter a compatibilidade, são colocados como uma opção especial no campo OPTIONS, a de código 53, associado a um dos comandos abaixo:

DHCP DISCOVER - Enviado pelo cliente para solicitar uma resposta de algum servidor DHCP

DHCP OFFER - Oferta de endereço IP de um servidor para um cliente. Um cliente pode receber várias ofertas de diferentes servidores DHCP

DHCP REQUEST - Requisição de um endereço específico daqueles oferecidos pelos servidores. É enviado por broadcast apesar de ser endereçado a um único servidor para que os demais tomem conhecimento da escolha.

DHCP DECLINE - Informa que a oferta contêm parâmetros incorretos (Erro)

DHCP ACK - Confirmação do servidor sobre a atribuição do endereço para a requisição do cliente. DHCP NAK - Servidor nega o fornecimento do endereço previamente oferecido, geralmente causado por um erro ou pelo fato do cliente ter demorado muito a requisitar o endereço solicitado.

DHCP RELEASE - Cliente libera o endereço IP utilizado. Ë raramente utilizado na prática, pois geralmente o cliente é desligado sem liberar o endereço. Ele retorna ao conjunto de endereços disponíveis no servidor devido ao estouro do tempo de leasing.

DHCP INFORM - Cliente que já possui endereço IP pode requisitar outras informações de configuração respectivas àquele endereço.

A operação de DHCP define diversos estados de funcionamento, quando o cliente está executando alguma ação e enviando uma das mensagens acima:

INITIALIZE

Configura interface com valor zero pois não tem endereço disponível - 0.0.0.0

Envia DHCPDISCOVER (UDP 67) como broadcast e muda para estado SELECT. Nesta mensagem, pode colocar opções de configurações desejadas

SELECT

Pode receber uma ou várias mensagens DHCPOFFER, cada uma com seus parâmetros distintos

Escolhe uma, envia DHCPREQUEST como broadcast e vai para estado REQUEST

REQUEST

Aguarda até receber DHCPACK do servidor escolhido. Se não receber, escolhe outra oferta e a solicita

Vai para o estado BOUND.

BOUND

É o estado normal de funcionamento.

Passa a utilizar o endereço, durante o tempo especificado pelo servidor

Quando o tempo atingir 50%, envia novo DHCPRequest para o servidor e passa para estado RENEW

Para cancelar o uso da endereço envia DHCPRelease

RENEW

Servidor pode enviar DHCPNAK, DHCPACK ou nenhuma resposta à solicitação de Request

Se receber ACK, volta para o estado BOUND

Se não receber resposta nenhuma, o cliente envia DHCPREQUEST em broadcast para que outros servidores possam enviar ofertas.

Se receber DHCPNAK, libera IP e vai para estado INITIALIZE

Opções DHCP

As opções DHCP tem o formato abaixo:

O código indica o tipo da opção. Os comandos DHCP tem sempre o código 53 e tamanho 1, sendo o próximo byte o código específico do comando:

1 = DHCPDISCOVER

2 = DHCPOFFER

3 = DHCPREQUEST

4 = DHCPDECLINE

5 = DHCPPACK

6 = DHCPNACK

7 = DHCPRELEASE

8 = DHCPINFORM

As opções de DHCP e BOOTP informam dados úteis para as diversas camadas TCP/IP, desde o nível de Reda ao Nível de Aplicação. Enumera-se algumas abaixo:

Opções Básicas:		
Code	**Param**	**Descrição**
0		Pad - alinhamento
255		Fim das opções
1	MASK	Máscara a ser utilizada pela estação
3	IP1, IP2, ...	Lista de roteadores default para a estação
6	IP1, IP2, …	Lista de servidores de DNS
9	IP1, IP2, …	Lista de servidores de impressão LPR
12	nome	Nome da máquina
13	número	Tamanho do arquivo de boot
15	nome	Nome do domínio
16	IP	Endereço do servidor de swap
17	nome	Path do diretório / da máquina
Opções de DHCP		
Code	**Param**	**Descrição**
50	IP	Endereço IP requerido preferencialmente
51	tempo (s)	Tempo de empréstimo de endereço
53	**mensagem**	**Mensagem DHCP**
54	IP	Identificação do servidor DHCP remetente
55	COD1, …	Cliente requisita opções ao servidor
56	texto	Mensagem de erro
57	número	Tamanho máximo da mensagem DHCP
58	tempo	T1 - Tempo de espera para estado RENEWING
59	tempo	T2 - Tempo de espera para estado REBINDING

Opções de IP

Code Param Descrição

1/0 Habilita IP Forwarding na estação

1/0 Habilita Source Routing na estação

Número Tamanho máximo do datagrama que cliente deve receber

Número Tamanho do TTL default da máquina

Número MTU da interface

1/0 Todas as interfaces tem o mesmo MTU ?

IP Endereço de broadacst da rede

1/0 Realizar ICMP Mask Discovery ?

31 1/0 Realizar ICMP Router Discovery ?

33 IP1/DEST1, IP2/DEST2, Rotas estáticas

Protocolo PPP

O protocolo PPP (Point-to-Point Protocol) é o principal protocolo para o transporte de IP sobre ligações ponto a ponto, criando um nível de enlace em um meio que não o possua. O PPP é empregado como protocolo de enlace nos seguintes tipos de meio: ligações seriais discadas, ligações seriais dedicadas (enlaces telefônicos, satélite, rádio), ligações ISDN e outras.

Pode-se diferenciar o funcionamento de PPP em dois grupos principais: quando empregado em ligações discadas ele provê os mecanismos de autenticação, com a correspondente interação com os dispositivos para verificar a autenticidade do originador da chamada, além de que as mensagens trocadas diferenciam o originador da chamada do receptor da chamada. Quando empregado em ligações dedicadas, geralmente não são trocadas mensagens de autenticação e o funcionamento do protocolo é praticamente simétrico em relação às mensagens trocadas.

PPP é genérico podendo carregar diversos protocolos de nível de rede OSI, além de possuir uma série de opções que podem ser negociadas pelos dois lados da conexão. PPP provê três tipos de funcionalidade:

Encapsulamento

Protocolos de Controle do Enlace PPP (protocolo LCP, PAP, CHAP, LQM)

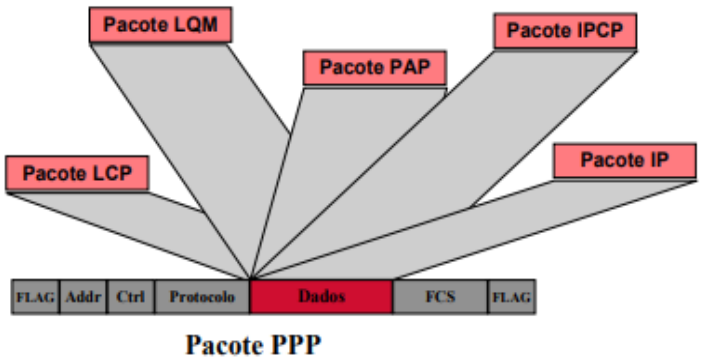
Protocolos de Controle do Protocolo de Nível 3 sendo carregado (protocolos IPCP, IPXCP, …)

O Encapsulamento de PPP na verdade não faz parte do protocolo, permitindo que ele se encaixe em outros protocolos de nível de enlace. O PPP pode utilizar diversos tipos de encapsulamento compatíveis com HDLC, ISDN e outros. Na sua forma default, o encapsulamento de PPP é similar ao início de um pacote HDLC, conforma a figura abaixo:

Os campos FLAG, ADDR e CTRL são similares a HDLC. Os campos Protocolo, Dados e FCS são comuns a todo pacote PPP. Protocolo contém o protocolo sendo carregado no campo de dados, sendo por exemplo os valores: LCP = C021, IPCP = 8021, IPXCP = 802B, PAP = C023, CHAP = C223, LQR = C025, IP = 0021, IPX =

002B, Bridging NCP = 8031, Netbios = 803F, ...

O encapsulamento dos diversos protocolos sobre PPP é mostrado na figura abaixo:



Protocolo LCP - Link Control Protocol

Este protocolo controla o enlace PPP. O formato de sua mensagem é dado abaixo:

COMANDO	ID	Length	Dados Variáveis

O Comando pode ser um dos seguintes tipos:

Configure-Request: Solicita o aceite para as opções especificadas no campo de dados

Configure-Ack: Concorda com as opções, para serem utilizadas pelo outro lado

Configure-Nack: Rejeita as opções, enumerando-as no campo de dados

Configure-Reject: Rejeita as opções que não possuem um campo de valor

Terminate-Request: Informa o fim da conexão PPP

Terminate-Ack: Concorda com o fim da conexão

Code-Reject: Informa erro no código do comando LCP

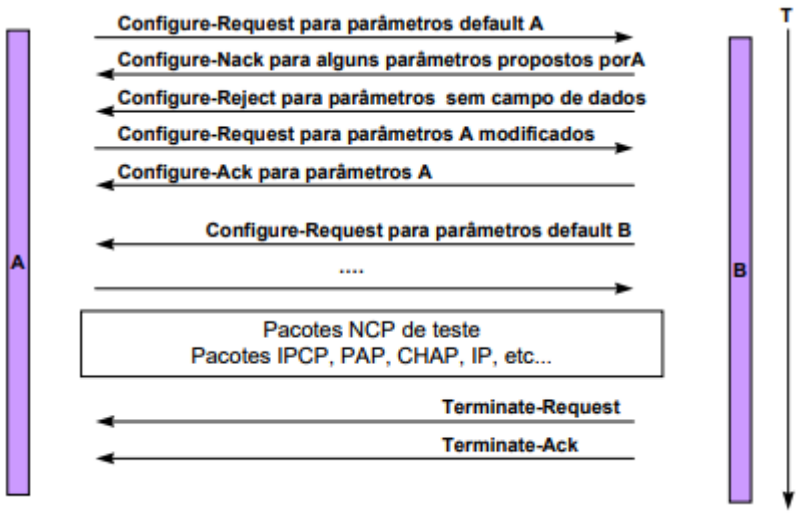
Protocol-Reject: Informa erro no protocolo da mensagem PPP

Echo-Request

Echo-Reply

Discard-Request

A troca de dados em uma conexão PPP é realizada conforme a figura abaixo. Os comandos de configuração do link PPP (LCP) são trocados com o objetivo de estabelecer os parâmetros de operação da ligação. Após o acordo dos comandos de configuração, são passados os comandos de configuração do protocolo de dados (IPCP) e, apóes estes, são finalmente passados os pacotes do protocolo IP.



As opções de configuração LCP mais utilizadas são:

Maximum Receive Unit

Authentication Protocol

Quality Protocol

Magic Number

Protocol Field Compression

Address Control Field Compression

Em ligações discadas é comum os servidores de acesso remoto possuírem a opção de detecção automática de PPP. Neste caso, como, geralmente os primeiros pacotes PPP trocados são os Configure-Request, basta que o receptor verifique se os dados correspondem aos códigos deste comando e, então, iniciem automaticamente o PPP.

Protocolo IPCP - Network Control Protocol

Os comandos possíveis no protocolo IPCP são:

Configure-Request: Solicita o aceite para as opções especificadas no campo de dados

Configure-Ack: Concorda com as opções, para serem utilizadas pelo outro lado

Configure-Nack: Rejeita as opções, enumerando-as no campo de dados

Configure-Reject: Rejeita as opções que não possuem um campo de valor

Terminate-Request: Informa o fim da troca de dados IP

Terminate-Ack: Concorda com o fim da troca de dados

Code-Reject: Informa erro no código do comando IPCP

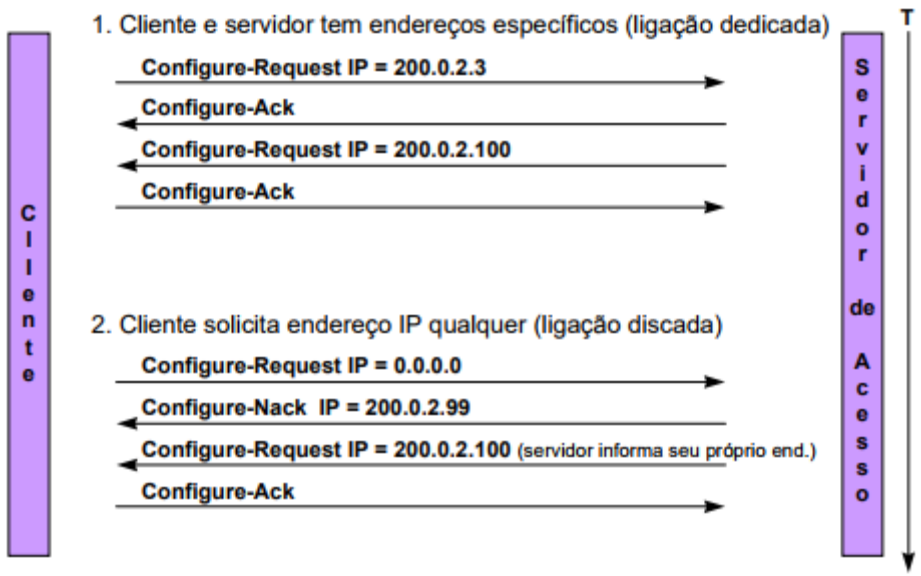
Este comandos são trocados de forma semelhante ao LCP, sendo que ao término da fase de acordo do IPCP, passam os dados do protocolo IP.

As principais opções de configuração de IPCP são:

IP Compression Protocol: Informa se será utilizado algum protocolo de compressão (e qual) para o cabeçalho IP

IP Address: origem informa ao destino o endereço IP a ser utilizado pela origem. No caso de conter 0.0.0.0 (que ocorre tipicamente na estação que realiza uma ligação serial discada), o outro lado (neste caso o servidor de acesso remoto) fornece o endereço IP a ser utilizado pela origem, através do comando Configure Nack.

As possíveis formas de negociação de endereço IP são dadas pela figura abaixo:



Protocolo SLIP

SLIP fornece apenas o encapsulamento para um enlace serial. Sua mensagem é dado na forma abaixo:

O funcionamento de SLIP ocorre da seguinte forma:

Transmite ESC

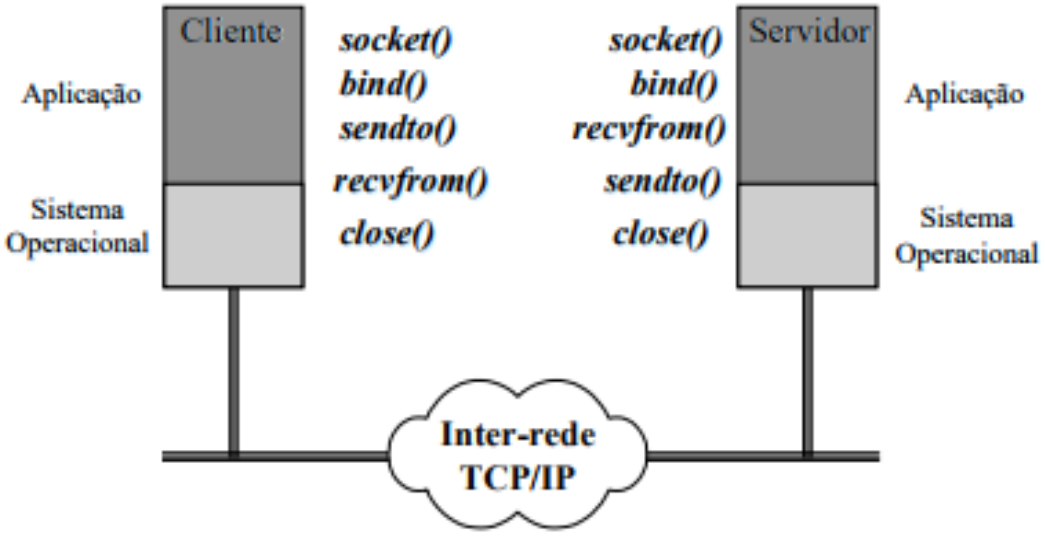
Transmite datagrama, caracter por caracter, substituindo um ESC nos dados por ESC ESC

Transmite END

Interfaces Do Nível De Transporte (Socket, Winsock)

A interface de socket do Unix é um conjunto de funções para permitir a utilização do sistema de comunicação por processos (programas) neste sistema operacional. A interface Winsock é composta de funções semelhantes a socket, para o ambiente Windows.

A interface socket possui funções distintas para a comunicação com e sem conexão. A utilização das funções de socket para a comunicação sem conexão é dada abaixo:



A utilização destas funções é dada abaixo:

socket: Inicializa a estrutura de dados do socket (equivalente ao SAP - Ponto de acesso de serviço), determinando qual o protocolo (PF_INET = TCP/IP) e o tipo do serviço (DGRAM = UDP e STREAM = TCP)

bind: associa o socket a uma porta USP ou TCP - pode-se dizer que para o programador, a porta do protocolo TCP ou UDP é efetivamente o socket.

sendto: solicita ao sistema de comunicação o envio de dados, especificando o endereço IP destino e a porta destino, além dos próprios dados.

recvfrom: informa ao sistema de comunicação que o programa está aguardando dados. O programa será congelado enquanto não houverem dados para receber, sendo reativado quando chegarem dados.

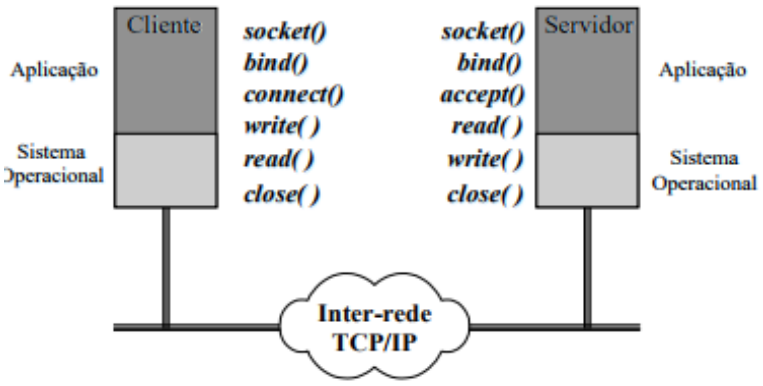
close: desassocia a porta do socket e desativa o socket.

Deve-se observar que nem todas as funções geram mensagens de rede. De fato, apenas a função sendto gera uma mensagem.

A sintaxe destas funções é mostrada abaixo:

```
sock1 = socket (pf, type, protocol)
      pf = PF_INET | PF_APPLETALK | PF_NETW | PF_UNIX
      type = SOCK_STREAM | SOCK_DGRAM | SOCK_RAW | SOCK_RDGRAM

close (sock1)

bind (sock1, localaddr, addrlen)
      localaddr = struct {ADDR_FMLY, PROTO_PORT, IP_ADDR}

sendto (sock1, message, length, flags, destaddr, addrlen)


recvfrom (sock1, buffer. length, flags, fromaddr, addrlen)

nptr = gethostbyname (name)
      nptr = struct {name, aliases, address_type, address}

nptr = gethostbyaddr (addr, len, type)

sptr = getservbyname (servname, proto)
      sptr = struct {name, protocol, port)
```

No caso de comunicação utilizando conexão, a utilização das funções é dada na figura abaixo:




```
connect (sock1, destaddr, addrlen)
      destaddr = struct {ADDR_FMLY, PROTO_PORT, IP_ADDR}

write (sock1, data, length)

read (sock1, buffer, length)

listen (sock1, qlength)

newsocket = accept (sock1, addr, addrlen)

ready = select (ndesc, indesc, outdesc, excdesc, timeout)
```

ndesc = numero de descritores a serem examinados indesc = descritores examinados

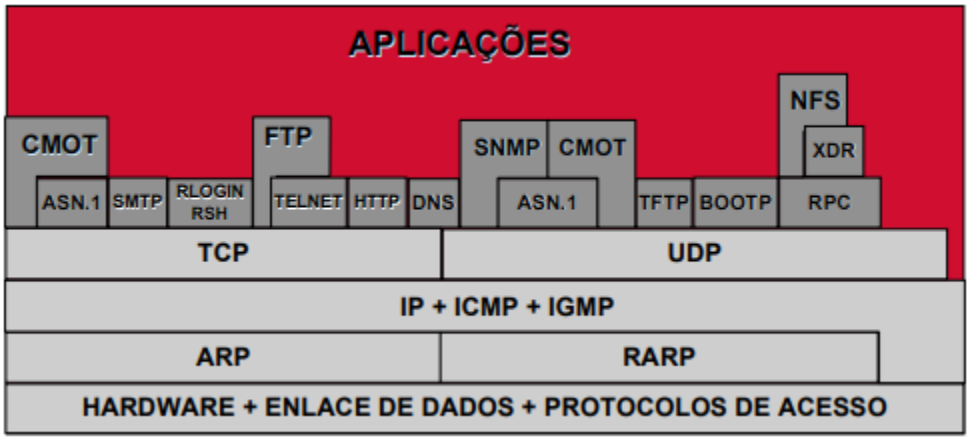
excdesc = descritores examinados para exceção timeout = tempo máximo de espera

Protocolos De Nível De Aplicação

Os protocolos de aplicação TCP/IP são aqueles que realizam as funções de alto-nível e que utilizam os serviços da camada de transporte UDP ou TCP para a comunicação.

Os protocolos de aplicação podem realizar funções diretamente acessíveis pelo usuário como FTP, HTTP, SMTP, POP3, IMAP4, Finger, Telnet, Chat, NFS, TFTP, NNTP e outros. Além disto, podem também realizar funções mais próximas do sistema de comunicação, tais como os protocolos DNS, BOOTP, DHCP, SNMP, BGP4, e outros.

As aplicações são ilustradas na figuira abaixo:



Protocolo DNS

O protocolo DNS (Domain Name System) especifica duas partes principais: regras de sintaxe para a definição de domínios e o protocolo utilizado para a consulta de nomes.

O DNS é basicamente um mapeamento entre endereços IP e nomes. A abordagem inicial para este mapeamento era a utilização de nomes planos, ou seja, sem hierarquia. Esta abordagem possui limitações intrínsecas quanto a escalabilidade e a manutenção. O sistema de nomes utilizado na Internet tem o objetivo de ser escalável, suportando a deinição de nomes únicos para todas as redes e máquinas na Internet e permitir que a administração seja descentralizada.

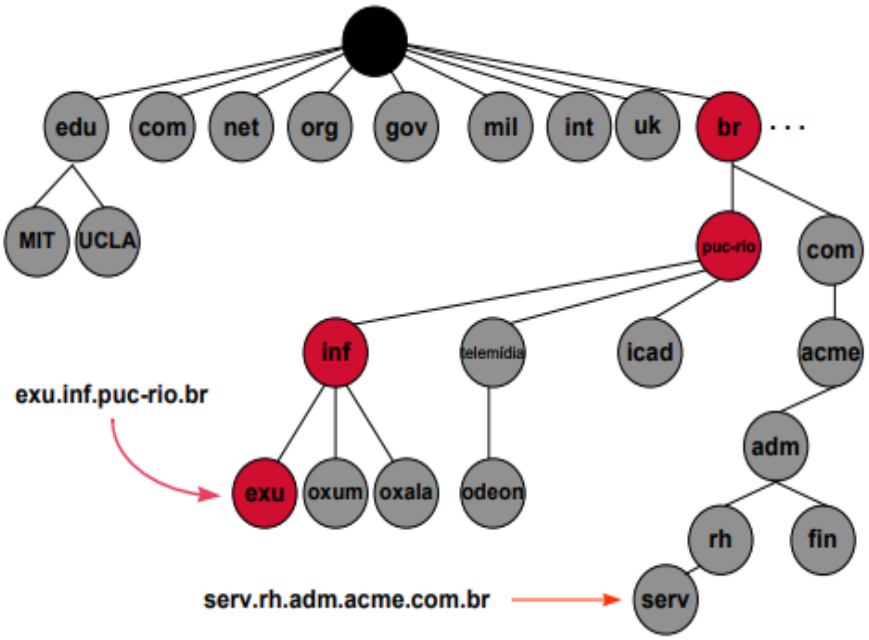
A estrutura de nomes na Internet tem o formato de uma árvore invertida onde a raiz não possui nome. Os ramos imediatamante inferiores à raiz são chamados de TLDs (Top-Level Domain Names) e são por exemplo:

.com, .edu., .org, .gov, .net, .mil, .br, .fr, .us, uk, etc… Os TLDs que não designam países são utilizados nos EUA. Os diversos países utilizam a sua própria designação para as classificações internas. No Brasil, por exemplo, temos os nomes .com.br., .gov.br, .net.br, .org.br e outros.

Cada ramo completo até a raiz como, por exemplo, puc-rio.br, acme.com.br, nasa.gov, e outros são chamados de domínios. Um domínio é a área administrativa englobando ele próprio e os subdomínios abaixo dele. Por exemplo o domínio .br engloba todos os subdomínios do Brasil. O domínio acme.com.br tem a responsabilidade por todos os domínios abaixo dele.

A delegação de responsabilidade de um domínio é a capacidade do DNS de simplificar a administração. Ao invés do domínio .br ser responsável diretamente por todos os seus sub-domínios e os que vierem abaixo deles, há na verdade uma delegação na atribuição de nomes para os diversos subdomínios. No exemplo acima, a empresa Acme possui a responsabilidade de administração do domínio acme.com.br.

A hierarquia de domínios pode ser observada na figura abaixo:



Os domínios principais genéricos, chamados de GTLDs (Generic Top Level Domain Names) que são .net, .com e .org são administrados pelo Internic (Internet Network Information Center) que também é responsável pela administração do espaço de endereçamento IP. Recentemente foram criados novos nomes de domínio genéricos que serão utilizado a partir de 98. São eles: .firm, .store, .web, .arts, .rec, .infor, .nom.

Os domínios são completamente independentes da estrutura de rede utilizada. Não existe necessariamente algum relacionamento entre eles. O DNS possui uma estrutura inversa para poder representar o endereçamento de rede, ou permitir que seja feito o mapemento do endereço IP correspondente a um nome. Esta estrutura possui como raiz principal a notação .arpa e possui como único ramo o .in-addr. Abaixo deste são colocados em ordem os bytes do endereço IP.

Implementação do DNS

O DNS é implementado por meio de uma aplicação cliente-servidor. O cliente é o resolver (conjunto de rotinas em uma implementação de TCP/IP que permite a aconsulta a um servidor) e um servidor geralmente é o programa bind ou uma implementação específica de um servidor de DNS (Windows NT).

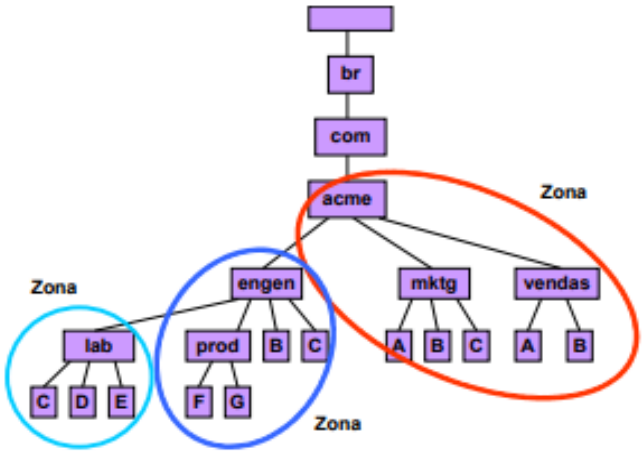
Um servidor de DNS pode ser responsável pela resolução de uma ou mais nomes de domínios (ex. acme.com.br, presid.acme.com.br). Seu escopo de atuação define a Zona de atuação de um servidor DNS.

Por exemplo, para resolver o domínio acme.com.br e seus sub-domínios existem três zonas: a primeira resolve o próprio domínio principal e os subdomínios mktg.acme e vendas.acme; a segunda resolve os domínios engen.acme e prod.engen.acme; e a terceira resolve o domínio lab.engen.acme.

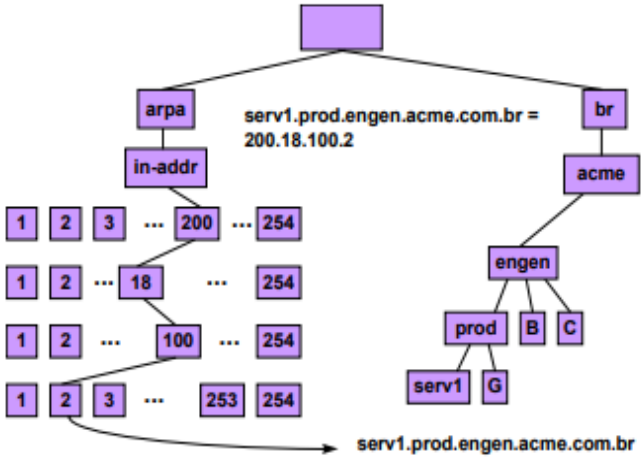
Cada zona possui um servidor de nomes principal ou primário, que mantêm em tabelas o mapeamento dos nomes em endereços IP daquele domínio.

Uma zona pode ter servidores secundários que possam substituir os primários em caso de falha. Os secundários, entretanto não possuem fisicamente as tabelas de mapeamento mas carregam regularmente as informações do primário.

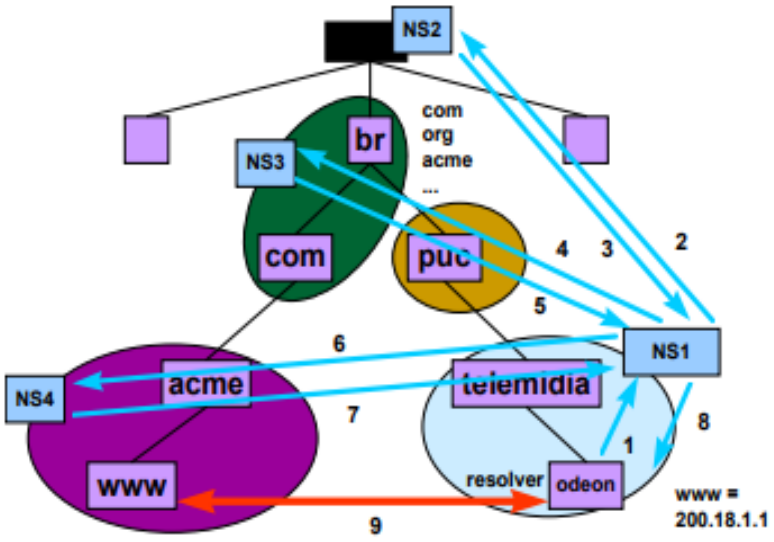
Veja figura abaixo:



Por outro lado, a representação do domínio reverso .in-addr.arpa para uma das máquinas de prod.engen.acme.com.br é visto abaixo:



A resolução de um nome é realizada de forma recursiva, consultando diversos servidores de nome até chegar àquele responsável pelo domínio consultado. Por exemplo a resolução do endereço www.lab.acme.com.br, será realizado pelo servidor da zona responsável por lab.acme.com.br. A figura abaixo ilustra o processo de consulta:



Protocolos de Roteamento

Protocolo RIP

Conforme citado em capítulos anteriores, o IP possui vários mecanismos para obter informações para sua tabela de rotas (específicas de cada máquina). A tabela de rotas de IP pode ser preeenchida por meio de:

Rotas default por meio de configuração estática (manual)

Rotas específicas por meio de configuração estática (manual)

Rotas default por meio do protocolo ICMP Router Advertisement

Rotas específicas para estação por meio de ICMP Redirect

Rotas aprendidas dinâmicamente por meio de protocolos de roteamento (ex. RIP, OSPF, BGP-4)

A última forma de aprendizado se aplica normalmente aos próprios roteadores, quando situados em redes complexas, já que suas tabelas de rota devem conter os detalhes de roteamento da rede (Uma estação por outro lado, pode ter rotas para um único roteador default e aprender rotas melhores por meio de ICMP Redirect).

O protocolo RIP é do tipo Vetor de Distância, já que baseia a escolha de rotas por meio da distância em número de roteadores. O funcionamento do protocolo RIP é bem simples, consistindo na divulgação de rotas de cada roteador para seus vizinhos (situados na mesma rede).

Cada roteador divulga sua tabela de rotas através de um broadcast na rede. Os demais roteadores situados na mesma rede recebem a divulgação e verificam se possuem todas as rotas divulgadas, com pelo menos o mesmo custo (custo é a quantidade de roteadores até o destino).

Se não possuírem rota para determinada rede divulgada, incluem mais uma entrada na sua tabela de rotas e colocam o roteador que a divulgou como o gateway para aquela rede. Em seguida, sua própria divulgação de rotas já conterá a rota nova aprendida.

Este processo se repete para todos os roteadores em um conjunto de redes, de modo que, após várias interações, todos já possuem rotas para todas as redes. Uma rota aprendida é mantida enquanto o roteador que a originou continuar divulgando. Caso o roteador pare de divulgar a rota ou nenhuma mensagem de divulgação seja recebida dele, o roteador que havia aprendido a rota a mantêm por 160 segundos, findos os quais rota é retirada da tabela de rotas. Neste caso, se outro roteador divulgar uma rota para aquela rede específica, esta será utilizada.

No caso em que um roteador, recebe rotas para uma mesma rede divulgadas por roteadores diferentes, a com menor custo é usada, sendo as demais descartadas.

O protocolo RIP não possui suporte para sub-rede (máscara de rede), o que só vem a ser suportado no protocolo RIPv2.

O custo de uma rota é a quantidade de roteadores que uma mensagem terá que atravessar desde o roteador que possui a rota até a rede destino. O custo máximo em RIP tem o valor de 16, que significa infinito. Por isto, o diâmetro máximo de uma rede com protocolo RIP é de 14 roteadores.

A mensagem RIP tem o seguinte formato:

0	7	15	23	31
Octeto 1	Octeto 1	Octeto 1	Octeto 1	
COMMAND	VERSION	MUST BE ZERO		
FAMILY OF NET 1		MUST BE ZERO		
IP ADDRESS OF NET 1				
MUST BE ZERO				
MUST BE ZERO				
DISTANCE TO NET 1				
FAMILY OF NET 2		MUST BE ZERO		
IP ADDRESS OF NET 2				
MUST BE ZERO				
MUST BE ZERO				
DISTANCE TO NET 2				
...				

Nesta mensagem, as rotas divulgadas por cada roteador são incluídas na parte IP ADDRESS OF NET X ….

As figuras abaixo mostram a divulgação de rotas por meio do protocolo RIP. Os roteadores divulgam e recebem informações de rotas via RIP, enquanto as estações apenas aprendem as rotas (RIP passivo).

Roteador G1 divulga sua tabela de rotas, que inicialmente contêm apenas as rotas diretas, para as redes ligadas diretamente.

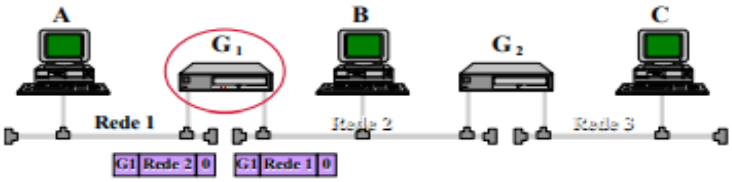


Tabela de Rotas

Rede	GW	M
Rede 1	-	0

Rede	GW	M
Rede 1	-	0
Rede 2	-	0

Rede	GW	M
Rede 2	-	0

Rede	GW	M

Rede	GW	M
Rede 3	-	0

O roteador G2, possui rotas para as redes ligadas diretamente, mas recebe um pacote de divulgação de rotas de R1, com uma rede nova (Rede 1). O roteador G2 instala a rota nova na sua tabela de rotas.

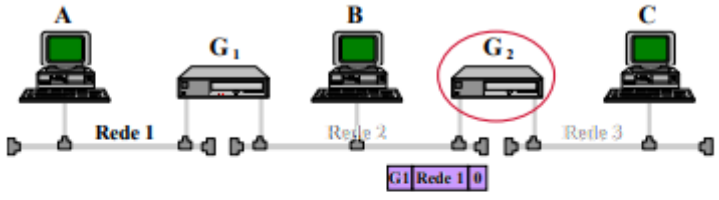


Tabela de Rotas

Rede	GW	M
Rede 1	-	0
Rede 2	G1	1

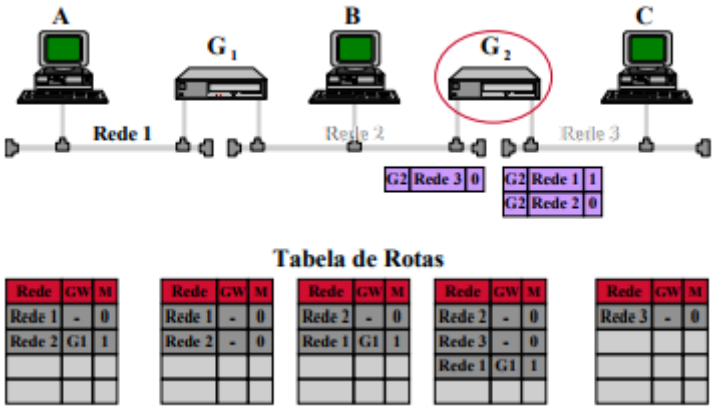
Rede	GW	M
Rede 1	-	0
Rede 2	-	0

Rede	GW	M
Rede 2	-	0
Rede 1	G1	1

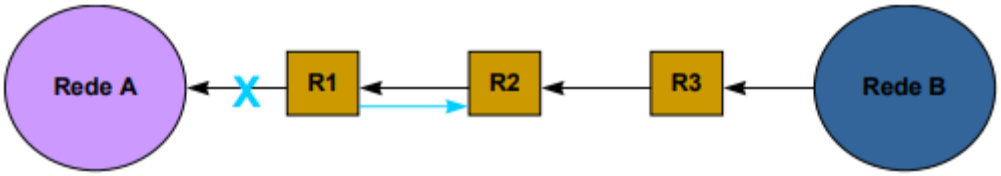
Rede	GW	M
Rede 2	-	0
Rede 3	-	0
Rede 1	G1	1

Rede	GW	M
Rede 3	-	0

O Roteador G2 divulga suas rotas para as redes ligadas diretamente, incluindo a rota nova aprendida de G1. G1, recebendo esta divulgação, instala uma rota nova para a Rede 3.



O protocolo RIP possui problemas intrínsecos de loop e convergência. O problema de convergência ocorre no seguinte caso



O roteador R2 havia aprendido uma rota para a Rede A, através de R1. Tanto R1 quanto R2 divulgam de 30 em 30 segundos a sua tabela de rotas por meio de RIP. No funcionamento normal, se R1 perder a rota para a Rede A, o roteador R1 divulgará uma mensagem RIP contendo uma rota para a Rede A com custo infinito (=16). O roteador R2, ao receber esta rota, verificará que ela veio de R1, de onde havia aprendido a rota para a rede A. Ele então procederá como determina o protocolo RIP e colocar a rota também com custo = 16.

Entretanto se, quando R1 perder a rota para a Rede A, R2 enviar sua tabela de rotas por RIP antes que R1 o tenha feito, R1 verificará que R2 possui uma rota melhor que ele para a rede A, com custo = 2 (já que R2 enviaria por meio de R1). R1 então instala uma rota para a rede A com custo = 3, sendo R2 o gateway da rota. Na próxima divulgação de R1, R2 constatará uma rota para a rede A com custo = 3. Ele então atualizará sua própria rota (já que a havia aprendido de R1), com custo = 4. A próxima divulgação de R2, causará a respectiva alteração do custo da rota em R1 para 5. Isto ocorre até que o custo desta rota atinja o valor 16.

O problema de convergência pode ser reduzido adotando-se as seguintes técnicas:

split -horizon update: não divulga rotas de volta para a interface de onde recebeu a informação de rota

hold-down: não aceita por 60s informações sobre uma rede após ela ser dada como não -alcançável

poison-reverse: divulga rotas de volta para a interface de onde recebeu a rota, mas com métrica 16 (não - alcançável e mantém este estado durante um tempo mínimo, mesmo recebendo rota para a rede

riggered-updates: força um roteador a divulgar imediatamente as rotas quando recebe rede não-alcançável

Protocolo RIP2

O protocolo RIP2 é bastante semelhante ao RIP, com as seguintes adições:

As rotas contêm a máscara da rede destino, permitindo divulgar rotas para subredes

O protocolo pode ser autenticado, adicionando segurança

RIP2 pode carregar informações de outros roteadores adjacentes, que funcionam com outros protocolos (como OSPF e BGP-4)

A mensagem RIP é mostrada abaixo: