

1 ENDEREÇAMENTO TCP/IP - RESUMO

Cada máquina na Internet possui um ou mais endereços de rede que são únicos, ou seja, não podem haver dois endereços iguais. Este endereço é chamado de **número Internet**, **Endereço IP** ou ainda **número IP**. É um número de 32 bits (Ipv4), sendo comumente representado por quatro números separados por pontos, como **143.54.8.11**. Este endereço pode ser estruturado de maneiras diferentes, usando uma parte para designar uma rede e as demais para designar os computadores naquela rede.

1.1 Introdução

1.1.1 Quem gerencia a numeração IP no mundo

A gerência da numeração IP mundial é feita desde outubro de 1998 pelo ICANN (Internet Corporation for Assigned Names and Numbers – <http://www.icann.org>), que é um órgão privado responsável por entrega de nomes de domínio e números IP. Ele está gradativamente tomando as funções do IANA – *Internet Assigned Numbers Authority* (<http://www.iana.org>), que é subsidiado pelo governo. O IANA/ICANN possuem representantes em 3 regiões mundiais: a) ARIN (American Registry for Internet Numbers – <http://www.arin.net>, responsável pelas Américas, Caribe e África abaixo do Sahara); b) RIPE (Reséau IP Européens – <http://www.ripe.net>, responsável pela Europa, parte da África e “Middle East”); c) APNIC (Ásia-Pacific Network Information Center – <http://www.apnic.net>).

A empresa que necessita um número IP deve procurar seu provedor, que, por sua vez, deve procurar o representante da sua região (no nosso caso o ARIN).

1.1.2 Correspondência número - nome

Além de números IP, cada máquina na Internet está associada com um nome que a distingue das outras. Esse nome é composto de caracteres separados por ponto, como **polaris.inf.ufrgs.br**, formando o *Fully Qualified Domain Name* (FQDN) de cada máquina. Neste caso, **polaris** é o nome da máquina e **inf.ufrgs.br** é o **domínio** ao qual esta máquina pertence. Os caracteres após o último ponto (**br** no exemplo anterior), indicam o tipo da organização. Existem vários tipos que diferenciam as entidades entre si, entre eles pode-se citar:

- **com** - instituição comercial ou empresa (ex: apple.com - Apple Computers);
- **edu** - instituição educacional (ex: berkeley.edu - Universidade de Berkeley);
- **gov** - órgão do governo (ex: nasa.gov - NASA);
- **mil** - organização militar (ex: nic.ddn.mil - departamento de defesa dos EUA);
- **net** - *gateways* e *hosts* administrativos de uma rede (ex: uu.net);
- **org** - organizações privadas que não se enquadram nas outras categorias (ex: eff.org);

- **países** - cada país tem duas letras que o caracterizam (ex: **br** - Brasil, **us** - EUA, **fr** - França, **de** - Alemanha, **au** – Austrália, e assim por diante). Baseados na norma ISO 3166 – <http://www.iana.org/country-codes.txt> mar 99. Consultar também <http://www.iana.org/cctld/cctld-whois.htm> (nov 2000).

No Brasil, temos os seguintes, entre outros (<http://www.seunome.com/dv1.shtml> – mar 99):

- gov.br entidades governamentais
- org.br entidades não-governamentais sem fins lucrativos
- com.br entidades comerciais
- mil.br entidades militares
- net.br empresas de telecomunicações
- g12.br entidades de ensino de primeiro e segundo grau
- art.br artes: músicas, pintura, folclore, entre outros
- esp.br clubes, esportes em geral
- ind.br organizações industriais
- inf.br provedores de informações (rádios, TVs, Jornais, Revistas, Bibliotecas)
- psi.br provedores de serviço Internet
- rec.br entretenimento, diversão, jogos, etc.
- tmp.br eventos temporários, como feiras e exposições
- etc.br os que não se enquadram nas categorias citadas

Recentemente no Brasil foram criados novos domínios, como .nom, .firm, e assim por diante. Esses novos “*generic top levels domains*” podem ser vistos na recomendação final do IAHD (*International Ad Hoc Committee*), no endereço <http://www.iahc.org/draft-iahc-recommend-00.html> – mar 99. Ver também na lista de domínios de primeiro nível na página do registro br – <http://registro.br>.

A Internet permite que o usuário utilize tanto números IP como nomes de domínios para referir-se a uma determinada localidade. Para tanto, é necessário um método de conversão entre as duas formas de representação da informação, que é executado pelo DNS (*Domain Naming System*), que faz a conversão e mantém a informação distribuída em cada domínio da Internet, com o objetivo de facilitar a administração do domínio e melhor organizar as informações entre os *hosts* conectados.

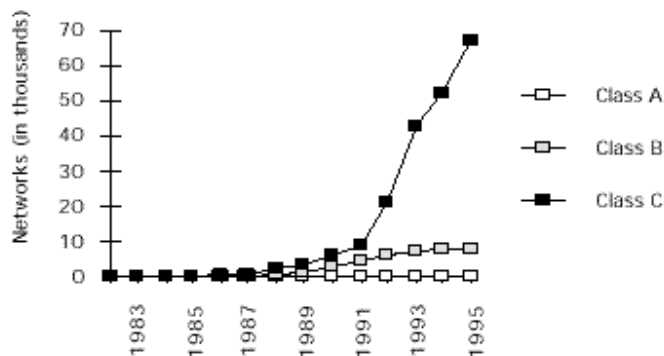
No Brasil, o banco de dados do registro de DNS fica na FAPESP (Federação de Auxílio a Pesquisa do Estado de São Paulo – <http://www.fapesp.br>).

Mundialmente, o controle é através do Internic – www.internic.net

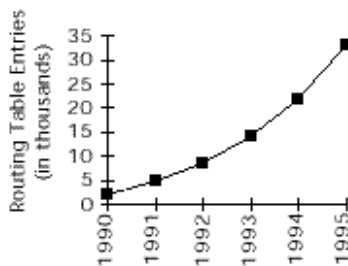
1.1.3 Problemas relacionados com o crescimento da Internet

- Eventual exaustão do endereçamento IPv4. Essa exaustão teve uma folga com a definição de números de intranet (item 1.1.4) e utilização de NAT (*Network Address Translator- RFC 1631*), bem como a utilização de proxy nas empresas.
- Habilidade para rotear tráfego em um número crescente de redes (tabelas de roteamento)

IPv4: endereços de 32 bits ($2^{32} = 4.294.967.296$) endereços disponíveis. Parece bastante, mas ele é mal distribuído na visão classful de endereços. O gráfico a seguir mostra o crescimento da alocação de endereços /SEM 96/.



Tabelas de roteamento: Continuando o crescimento de forma desorganizada, haveria um excesso de entradas nas tabelas de roteamento. O gráfico a seguir ilustra esse crescimento.



1.1.4 range de IPs livres para Intranet (RFC 1918)

10.0.0.0 ate 10.255.255.255 para Classe A

172.16.0.0 ate 172.31.255.255 para Classe B

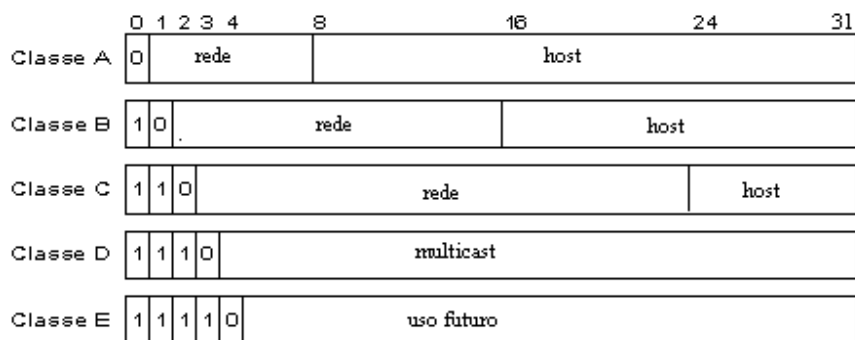
192.168.0.0 ate 192.168.255.255 para Classe C

1.1.5 Definições básicas de endereçamento

Rede, host, endereço broadcast, endereço subrede.

1.2 Endereçamento Classful

O endereçamento classful pode ser visto na tabela a seguir:



Observe que o endereço é auto-contido, ou seja, não precisa máscara. Por exemplo, se os primeiros dois bits de um endereço IP são 1-0, então trata-se de um classe B, e o ponto de divisão entre rede e host é entre o 15^o e o 16^o bit. Esse conceito simplificado de roteamento foi usado no princípio da Internet, pois os protocolos de roteamento originais não suportavam uma máscara.

1.2.1 Exercícios

- Fazer esquematicamente o espaço de endereçamento IPv4 para as classes A, B, C, D e E, desenhando o resultado.
 - Especificar número de hosts e redes máximo para cada classe.
 - Especificar quanto por cento do espaço de endereçamento é usado para cada classe.
 - Especificar o range de endereços na terminologia de ponto decimal para cada classe.
- Coloque os seguintes números em binário, dizendo a que classe de rede eles pertencem:
 - 9.3.158.1 _____
 - 100.8.5.4 _____
 - 143.54.8.11 _____
 - 200.248.3.1 _____
 - 224.8.1.8 _____
 - 247.5.4.3 _____

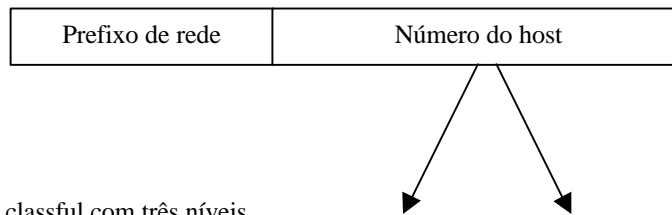
1.3 Subredes

RFC 950 (1985): definição de um processo padrão para dividir uma classe A, B ou C em pedaços menores, utilizando subredes.

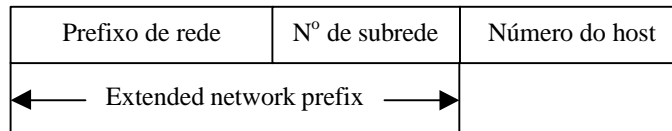
Melhorias: Diminui tabelas de roteamento na Internet; administradores podem ter autonomia na criação de subredes internas à empresa (antes necessitavam requisitar outro número de rede).

A figura a seguir mostra a idéia básica de subredes.

Hierarquia classful com dois níveis



Hierarquia classful com três níveis



O “extended network prefix” é identificado pela **máscara de subrede**, e forma a nova notação de endereçamento utilizado atualmente, o “/x”. Por exemplo, um classe A tradicional é um /8. Um classe B é um /16 e um classe C é um /24.

Por exemplo, um endereço de rede 143.54.8.11 com máscara 255.255.255.0 pode ser expresso como 143.54.8.11/24. Isso facilita o entendimento, como mostra a tabela a seguir.

143.54.8.11	10001111.00110110.00001000.00001011
255.255.255.0	11111111.11111111.11111111.00000000
143.54.8.11/24	10001111.00110110.00001000.00001011

1.3.1 Questões de Projeto

1. Qual o total de subredes da empresa hoje e no futuro?
2. Qual o número de hosts de cada subrede hoje e no futuro?

1.3.2 Exercícios

1. Uma organização recebeu o número de rede 156.1.1.0/24, e precisa definir 6 subredes. A maior subrede deve suportar 25 hosts. Defina o seguinte: a) o tamanho do extended network prefix b) máscara de subrede c) número de cada subrede d) endereço broadcast de cada subrede f) endereços de host para cada subrede; g) endereço do roteador e default gateway para cada subrede
2. Uma organização recebeu o número de rede 156.1.0.0/16, e precisa definir 8 subredes. Defina o seguinte: a) o tamanho do extended network prefix b) máscara de subrede c) número de cada subrede d) endereço broadcast de cada subrede f) endereços de host para cada subrede

1.4 VLSM – Variable Length Subnet Masks e CIDR (Classless Inter-Domain Routing)

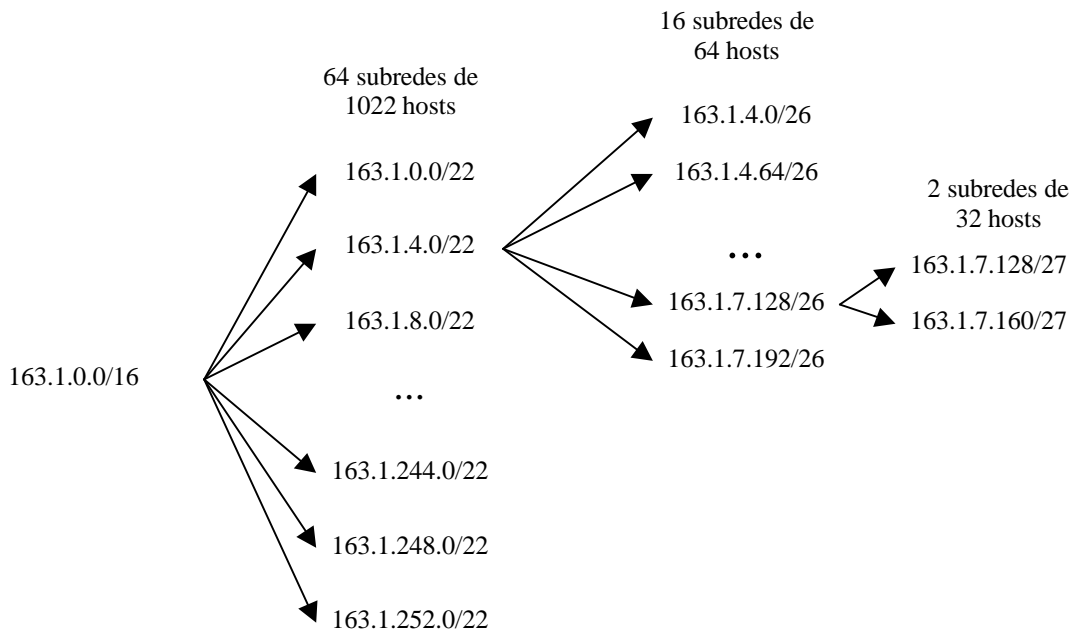
VLSM: RFC 1009 (1987) e RFC 1716.
 CIDR: RFC 1517, 1518, 1519 e 1520.

1.4.1 VLSM

Podendo dividir a rede em subredes de tamanho variável permite uma melhor utilização do espaço de endereços destinados à empresa. Antes a empresa tinha que ficar com um número fixo de subredes de tamanho fixo. Com VLSM, é possível ter redes com grande número de hosts e também com pequeno número de hosts.

Exemplo: Suponha que uma empresa razoavelmente grande tenha um classe B cheio (163.1.0.0/16), permitindo até 65.534 hosts. Entretanto, essa empresa precisa de algumas subredes com aproximadamente **1.000 máquinas**, e outras em setores com aproximadamente **30 máquinas**. Se dividir igualmente o espaço de endereçamento (um /22), terá somente **64 subredes de 1022 hosts**, o que provocará um desperdício em setores pequenos (aproximadamente 1.000 endereços desperdiçados). Qual a solução? VLSM.

A figura a seguir mostra uma alternativa.

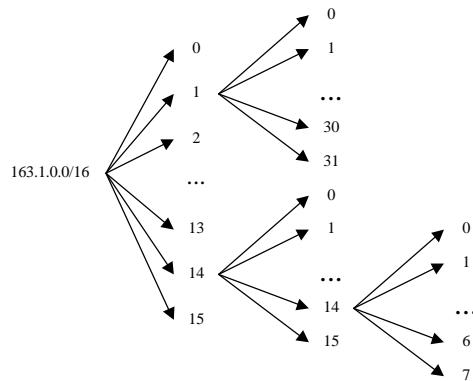


1.4.2 CIDR

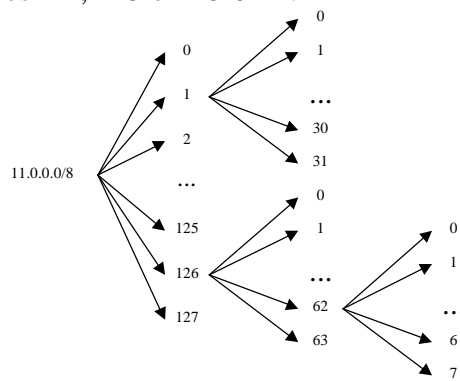
O CIDR é praticamente a mesma coisa que o VLSM, porém, envolve a Internet externa à empresa, a fim de facilitar o roteamento entre domínios.

1.4.3 Exercícios

1. Para a figura a seguir, definir a) todas as subredes envolvidas, com máscaras de subrede e extended network prefix para cada uma b) endereçamento de hosts, e endereço broadcast para as subredes 1-1, 13 e 14-14-1.



2. Para a figura a seguir, definir a) todas as subredes envolvidas, com máscaras de subrede e extended network prefix para cada uma b) endereçamento de hosts, e endereço broadcast para as subredes 1-1, 125 e 126-62-1.



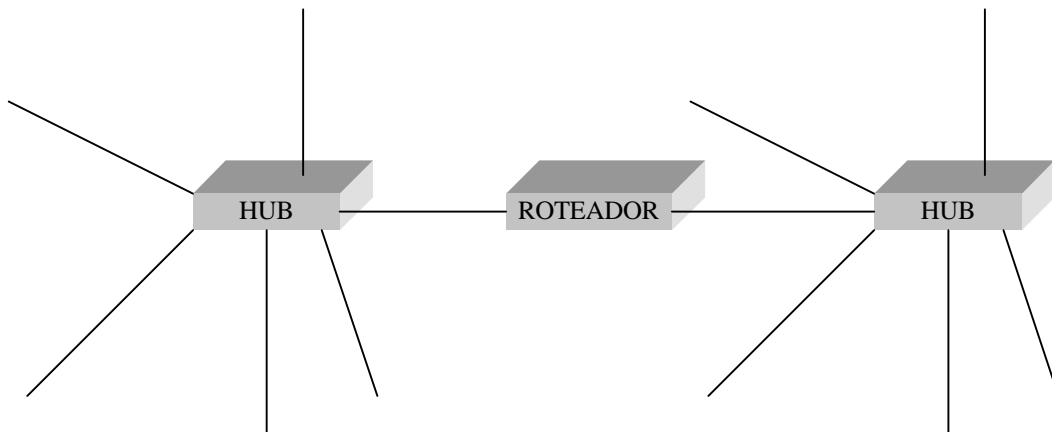
1.5 Referências

/SEM 96/ Semeria, Chuck. "Understanding IP addressing": everything you ever wanted to know. NSD Marketing: 3Com Corporation, 1996. Em <http://www.3com.com/nsc/501302.htm>, jan 99.

2 ALGUNS PROTOCOLOS BÁSICOS DA PILHA TCP/IP

2.1 Exemplo de subredes

Explicar em aula.



2.2 ICMP - Internet Control Message Protocol

ICMP é um protocolo que roda sobre IP e é usado para comunicar diversas informações de controle a outras estações, principalmente mensagens de erro. Algumas são listadas a seguir:

- **Destination Unreachable:** pacote não pôde ser entregue.
- **Redirect:** Esta mensagem é gerada por um gateway quando ele está sendo usado por outra estação, mas sabe que existe uma rota muito mais curta para chegar ao destino. Assim, redireciona o pacote ao gateway correto e gera uma mensagem à estação origem avisando da melhor rota.
- **Echo Request e Echo Reply:** Utilizado principalmente para saber se uma máquina está “viva”. Ao receber essa mensagem, a estação envia uma mensagem de Echo Reply de volta à origem. Esse recurso é usado pelo aplicativo ping e traceroute.

Outras mensagens ICMP, como as *Subnet Mask Request* e *Subnet Mask Reply* são semelhantes às *Echo Request* e *Echo Reply*, exceto que elas fornecem informações adicionais sobre a subrede à que pertence a máquina requerida.

2.3 ARP: Address Resolution Protocol

O protocolo de resolução de endereços ARP (Address Resolution Protocol) é utilizado para o mapeamento do endereço IP em números MAC. Quando inicializadas, as estações não possuem uma tabela de endereços IP<->físico armazenada. Em vez disso, para cada endereço IP solicitado que não esteja na tabela da estação, o protocolo ARP manda um pedido via broadcast de nível 2 para o endereço IP determinado. O destinatário que tiver tal

endereço IP responde à máquina solicitante seu endereço físico. Nessa ocasião, tanto a tabela da máquina origem quanto a da máquina destinatária são atualizadas com os endereços. O endereço de hardware e o endereço IP do computador então é armazenado no cache do ARP para uso futuro. Para ver a cache, pode-se utilizar o comando `arp -a`, como mostra a figura.

```
C:\> arp -a
```

```
Interface: 10.16.169.9 on Interface 0x1000002
  Endereço Internet           Endereço físico           Tipo
  10.16.169.1                 02-a0-c9-d0-d9-dc       dinâmico
  10.16.169.10                00-60-97-74-02-b9       dinâmico
  10.16.169.13                00-50-04-05-8a-88       dinâmico
```

A duração da tabela de arp quando não usada é aproximadamente 2 minutos. Quando usada é de aproximadamente 10 minutos, e quando configurada estaticamente não é retirada (TCP/IP implementation details - Technet).

2.4 RARP: Reverse Address Resolution Protocol

Serve para que uma estação descubra o endereço IP associado a um endereço Ethernet. Ele é necessário, por exemplo, quando uma estação *diskless* é inicializada e necessita descobrir qual o endereço de seu servidor, por exemplo. Para obter tal informação, ela envia uma mensagem *broadcast* solicitando a algum servidor enviar seu endereço IP. Uma estação *diskless*, como se sabe, conhece seu endereço Ethernet e pouca coisa mais.

2.5 O aplicativo ping (packet Internet Groper)

O utilitário ping envia uma seqüência de pacotes ICMP do tipo *Echo Request* para determinada localidade. O *host* que recebe essa mensagem deve enviar de volta pacotes do tipo *Echo Reply*, permitindo assim descobrir se o *host* destino está funcionando ou não, como mostra o exemplo abaixo.

```
roesler@polaris 2 % ping cs.colorado.edu
cs.colorado.edu is alive
```

Como pode-se ver, a resposta indica apenas que a máquina **cs.colorado.edu** está "viva", o que é bastante importante em determinadas situações, como por exemplo quando a comunicação não está funcionando e deseja-se saber onde está o problema.

Outra utilidade do ping é quando usa-se também a *flag -s*, que faz com que seja informado também o tempo total que a mensagem levou para ir até o destino e retornar, permitindo assim analisar o retardo da rede, e também a melhor alternativa na hora de instalar um servidor remoto. Um exemplo de utilização do **ping -s** é mostrado a seguir.

```
roesler@polaris 15 % ping -s archie.ans.net 300 20
PING forum.ans.net: 300 data bytes
308 bytes from forum.ans.net (147.225.1.10): icmp_seq=0. time=1046. ms
308 bytes from forum.ans.net (147.225.1.10): icmp_seq=1. time=904. ms
...
308 bytes from forum.ans.net (147.225.1.10): icmp_seq=18. time=932. ms
```

308 bytes from forum.ans.net (147.225.1.10): icmp_seq=19. time=1226. ms

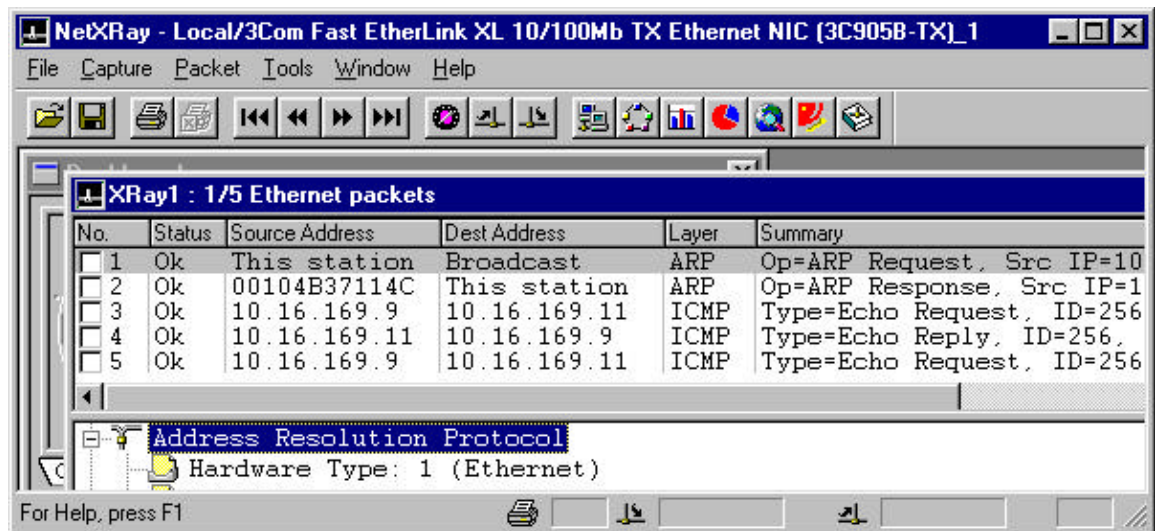
----forum.ans.net PING Statistics----

20 packets transmitted, 20 packets received, 0% packet loss
round-trip (ms) min/avg/max = 866/976/1291

No comando acima, ordenou-se que a mensagem de teste de eco contenha 300 bytes de dados e que seja repetida 20 vezes. A estatística mostra que as mensagens demoraram uma média de 976 ms para executar o trajeto de ida e volta ao *host* **archie.ans.net**, localizado em Nova Iorque.

Através deste utilitário, após testes com todos os *hosts* servidores possíveis e em diferentes horários, pode-se escolher o servidor que tem a rota mais rápida, para então instalar um determinado cliente (Archie, por exemplo) direcionado a ele.

A figura a seguir mostra o que acontece na rede quando se faz um ping.



2.6 Traceroute

O utilitário **traceroute**, de Van Jacobsen, envia a mensagem *Echo Request* do ICMP para determinada localidade, mas seqüencialmente incrementa o valor da variável *Time To Live* (TTL) a partir de 1. Isso faz com que o pacote retorne ao *host* origem com a variável "*TTL Expired*" ativada por cada *host* até onde a mensagem chega com cada TTL incrementalmente definido, até o destino final. Dessa forma, o *host* origem pode descobrir a rota feita pelas mensagens através da rede. Esse mecanismo permite determinar a estrutura da rede e é bastante padrão, não exigindo privilégios especiais para ser executado por um determinado *host*. O exemplo abaixo foi executado na estação **caracol** do Instituto de Informática da UFRGS, no dia 27 de julho de 1993, às 22:30, e mostra o caminho que percorre uma mensagem até chegar ao Japão.

```
caracol-gw% traceroute ftp.tohoku.ac.jp
traceroute to akiu.gw.tohoku.ac.jp (130.34.8.9), 30 hops max, 40 byte packets
 1 routcv (143.54.2.98) 16 ms 14 ms 15 ms
 2 routcc (143.54.1.10) 144 ms 193 ms 194 ms
```

- 3 vortex (143.54.1.7) 82 ms 13 ms 16 ms
- 4 cisco-poa (143.54.1.9) 21 ms 19 ms 18 ms
- 5 cisco-sao (192.111.229.9) 42 ms 47 ms 43 ms
- 6 fnal-brazil.es.net (192.74.212.5) 2195 ms 2023 ms 1714 ms
- 7 fnal-e-fnal2.es.net (134.55.12.129) 2194 ms 1842 ms 2364 ms
- 8 lbl-fnal.es.net (134.55.4.129) 2596 ms 2138 ms 2693 ms
- 9 lbl-lc2-1.es.net (134.55.12.98) 3002 ms * 832 ms
- 10 llnl-lbl-t3.es.net (134.55.12.65) 735 ms 874 ms 756 ms
- 11 llnl-e-llnl2.es.net (134.55.12.225) 1457 ms 1515 ms 947 ms
- 12 ames-llnl.es.net (134.55.4.161) 1496 ms 2106 ms 2415 ms
- 13 ARC2.NSN.NASA.GOV (192.52.195.11) 2879 ms 1988 ms 1900 ms
- 14 ARC5.NSN.NASA.GOV (192.100.12.5) 1875 ms 20684 1235 ms
- 15 132.160.251.2 (132.160.251.2) 2322 ms 1970 ms 2498 ms
- 16 jp-gate.wide.ad.jp (133.4.1.1) 3061 ms 2779 ms 2815 ms
- 17 wnoc-snd.wide.ad.jp (133.4.4.2) 2477 ms 2460 ms 2419 ms
- 18 nogu.gw.tohoku.ac.jp (130.34.10.10) 2774 ms 2126 ms 2963 ms
- 19 izumi.gw.tohoku.ac.jp (130.34.10.3) 2448 ms 3465 ms 2740 ms
- 20 akiu.gw.tohoku.ac.jp (130.34.8.9) 1990 ms 1073 ms 1530 ms

Como pode-se notar, este utilitário é bastante importante para descobrir a topologia de uma determinada rede. Pode-se descobrir inclusive problemas de roteamento, como mostra o exemplo acima, onde a mensagem sai do roteador routcv (campus do vale, em Viamão), vai para o routcc (campus central, na reitoria, Porto Alegre), sendo então redirecionada para o vortex (campus da saúde, na rua Ramiro Barcelos), voltando em seguida para o roteador cisco-poa (na reitoria novamente), para então seguir seu caminho até o cisco-sao (São Paulo) e então seguir viagem para o exterior (Chicago, etc...). Como ficou mostrado, a mensagem faz um vai e volta da reitoria para o campus da saúde, podendo simplesmente entrar diretamente do routcc para o cisco-poa, eliminando um hop. Esse problema aconteceu devido a uma modificação de roteadores, e foi temporário.

Além disso, pode-se constatar que o maior gargalo existente na comunicação é quando a mensagem sai do Brasil (cisco-sao) e chega em Chicago (fnal-brazil-es.net). Nesse ponto ocorre um salto de tempo (de 47 ms passa para 2023 ms), mostrando que existe um congestionamento nessa rota e que há a necessidade de uma conexão mais rápida do Brasil para o exterior.

2.7 Telnet

O telnet permite a qualquer usuário pertencente à Internet se transformar em um terminal de outro computador ou equipamento (roteador, por exemplo), mesmo que este esteja do outro lado do mundo. Para tanto, esse usuário deverá possuir um *login* e respectiva senha do equipamento remoto. A partir da conexão, o usuário vai possuir todos os poderes de acesso conferidos pela sua senha de *login*.

Para abrir uma conexão com a máquina remota, o usuário deve digitar: **telnet local.domínio**, onde **local.domínio** é o endereço da máquina que ele deseja acessar. Quando a conexão é aberta, é solicitado um nome de *login* e uma senha. Ao responder corretamente estas informações, o usuário torna-se um terminal da máquina remota.

Alguns locais tem senha pública, como a biblioteca da UFRGS - telnet asterix.ufrgs.br, login SABI ou sabibib, sem senha. O resultado pode ser visto a seguir.

```

Telnet - asterix.ufrgs.br
Conectar  Editar  Terminal  Ajuda
UAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
3
3      UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
3      SISTEMA DE AUTOMACAO DE BIBLIOTECAS - Versao 4.4
3
3
3      SSSSSS   AAAAAA   BBBBBBBB   ii
3      SSSSSSS  AAAAAAAA  BBBBBBBB
3      SS       AA   AA   BB   BB   ii
3      SSSSS   AAAAAAAA  BBBBBBBB   ii
3      SSSSS   AAAAAAAA  BBBBBBBB   ii
3      SS       AA   AA   BB   BB   ii
3      SSSSSSS  AA   AA   BBBBBBBB   iiii
3      SSSSS   AA   AA   BBBBBBBB   iiii
3
3
3      UAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
3      3  Voce esta entrando no SABI - Sistema de Automacao de Bibliotecas.
3      3  Este sistema contem informacoes sobre o acervo disponivel nas
3      3  unidades que integram o Sistema de Bibliotecas da UFRGS (SBU).
3      3  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAU
3
3      Teclle <?> para informacoes gerais ou <ENTER> para continuar
3
3      AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAU
  
```

Outro exemplo é para configurar remotamente um equipamento de rede, como o switch 1000 da 3Com, como mostra a figura abaixo:

```

Telnet - 10.16.169.4
Conectar  Editar  Terminal  Ajuda
lqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqk
x
x      SuperStack II Switch
x
x      tqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
x
x
x
x      SSSS U  U PPPP EEEEE RRRR  SSSS TTTT AAA  CCCC K  K  III III
x      S  U  U P  P E   R  R S  T  A  A C  K  K  I  I
x      SSS U  U PPPP EEEE RRRR SSS  T  AAAAA C  KKK  I  I
x      S  U  U P  E   R  R  S  T  A  A C  K  K  I  I
x      SSSS  UUU P  EEEEE R  R SSS  T  A  A  CCCC K  K  III III
x
x
x
x
x      switch1000
x
x
x      Press Enter to Continue ...
x      [Enter]
x
x      tqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
x
x      mqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqj
  
```

Muitas aplicações podem ser simuladas quando se sabe o protocolo e o socket correspondente. Um exemplo é usar o telnet para enviar mail, como mostra a tabela a seguir.

Comando	Argumento	Descrição
\$ telnet <ip>	25	Faz telnet na porta 25 (porta de mail)
HELO	Domínio origem	
Mail from:	Nomeorig@dominioorig	Configura originador da mensagem
RCPT to:	Nomedest@dominiodest	Configura destinatário da mensagem
Data		Avisa que é mail de dados

Subject:	Assunto	Configura assunto da mensagem
From:	Nomeorig@dominioorig	Avisa novamente originador
		Executa mensagem
.		Linha "." para finalizar mensagem
Quit		Sai do telnet

Ou para recebendo mail, utilizando telnet para a porta 110 (POP3). Comandos disponíveis:

USER - login as a user
 PASS - specify a password
 APOP - perform secure login
 STAT - show mailbox statistics
 RETR - send a message
 LIST - show message numbers and sizes
 DELE - delete a message
 RSET - 'undo' all mailbox changes
 TOP - show lines from a message
 QUIT - close the connection
 NOOP, RPOP, LAST are also supported.

Comando	Argumento	Descrição
\$ telnet <ip>	110	Faz telnet na porta 110 (porta de mail)
User	username	Configura usuário
Pass	*****	A senha ecoa na tela
List		Mostra mensagens

2.8 FTP - File Transfer Protocol

O FTP (*File Transfer Protocol*) é um método bastante comum de transferir arquivos através de redes de computadores, pois permite a entrada de um usuário em qualquer máquina que implemente este serviço (localmente ou no mundo inteiro), mesmo que o usuário não possua senha naquela máquina (*anonymous ftp*).

Para permitir este acesso, a máquina que disponibiliza seu banco de dados deve tomar certas precauções para evitar danos e acessos indevidos, assim, os diretórios e arquivos normalmente são bloqueados para escrita, e somente um certo número de comandos são disponibilizados para os usuários, como será visto adiante.

Para criar a conexão com a máquina remota, o usuário deve digitar: **ftp local.domínio**, onde **local.domínio** é o endereço da máquina que ele deseja acessar. Quando a conexão for aberta, será solicitado o nome do usuário, e ele pode digitar seu nome (caso tenha senha na máquina) ou *anonymous* (em servidores disponíveis para todos). Será solicitado então uma senha, e nesta senha é sugerido que se coloque o endereço de correio eletrônico da pessoa (**nome@local.domínio**).

Neste instante a máquina libera o usuário para acessar seu banco de dados. Os principais comandos que estão disponíveis são os seguintes:

dir - lista o conteúdo do diretório da máquina remota;
cd - troca o diretório corrente;
bin - busca arquivo em modo binário (necessário para arquivos comprimidos);
get - transfere arquivo remoto para máquina local;

mput e mget – transferência de múltiplos arquivos

...

Existem vários outros comandos, que podem ser obtidos a partir do próprio programa, que normalmente possui um *help on-line* associado.

Vale lembrar que no FTP, no POP3 e no telnet a senha vai aberta, portanto, cuidado no usar.

3 PROTOCOLO DE TRANSPORTE - [TAN96], PG. 552 - 600

Serviço de transporte	Orientado à conexão (confiável) – para o usuário os erros são transparentes, e ele simplesmente cria um ponto de conexão (soquete) com o destino, sem se preocupar em corrigir os erros, pois é função da camada 4.
	Orientado a datagrama (não confiável) – o usuário utiliza esse serviço quando não tem compromisso com a confiabilidade, ou ela é menos importante, como por exemplo transmissão de voz.

Primitivas para um serviço simples de transporte

Primitiva	TPDU enviada	Significado
LISTEN	Nenhuma	Bloquear até que algum processo tente se conectar
CONNECT	CONNECTION REQ.	Tentar ativamente estabelecer uma conexão
SEND	DATA	Enviar informações
RECEIVE	Nenhuma	Bloquear até que uma TPDU DATA chegue
DISCONNECT	DISCONNECTION REQ.	Este lado quer encerrar a conexão

3.1 API de Sockets

Um *socket* pode ser considerado como um ponto de referência para o qual as mensagens são enviadas e a partir da qual as mensagens podem ser recebidas. Dessa forma, elas estabelecem um canal de comunicação e podem começar a enviar e receber mensagens uma para a outra. Uma estação de trabalho A que deseje se comunicar com uma estação de trabalho B deve saber o número do *socket* aberto pela aplicação que roda na estação B e vice-versa.

A seqüência típica de utilização de *sockets* para comunicação entre um processo cliente e um servidor está ilustrada na figura a seguir. O procedimento é assimétrico, porque um dos processos (o servidor) espera que o outro requisiute a conexão (o cliente).

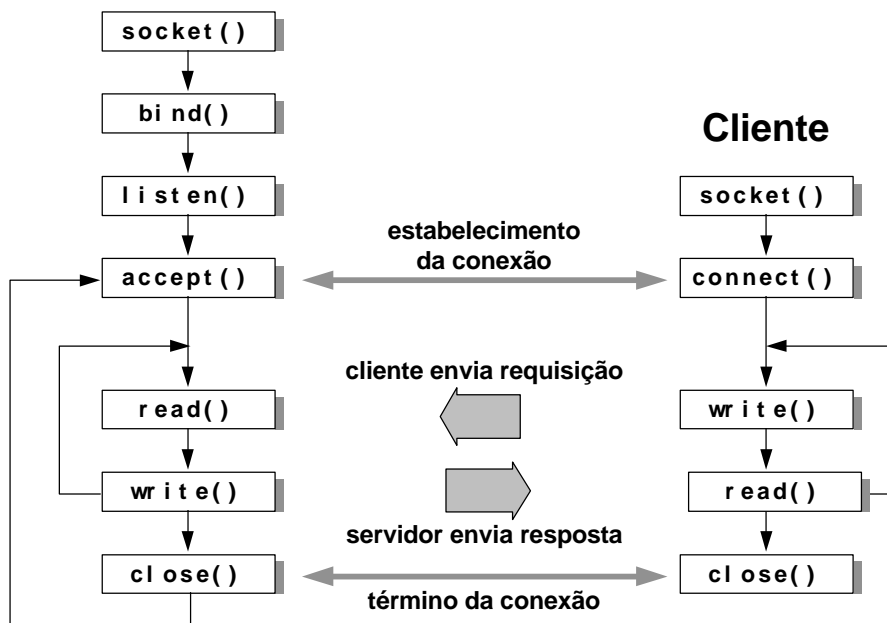


Figura: seqüência de chamadas para comunicação orientada a conexão

Na seqüência da figura acima, o servidor atende apenas um cliente, pois suporta uma única conexão aberta de cada vez. Tal interação é, entretanto, bem mais simples do que a de um servidor que precisa atender múltiplas chamadas concorrentemente. Neste caso, o servidor precisa continuamente verificar novos pedidos de requisição, ler das conexões já estabelecidas e tratar conexões falhas.

3.2 Exemplo: aplicativo utilizando *sockets*

`/**/ mostrar programa java`