



Introdução a Redes de Computadores TCP/UDP/IP

Fernando Braghetto
Maio/2005



Ementa 1 - Básico

- ◆ Entender porque rede é importante
- ◆ Tecnologias de Rede
 - ◆ Motivação, Topologias, Tipos de Transmissão em meios físicos,
- ◆ Redes de Comutação de Pacotes
 - ◆ Segmentação, atraso (por processamento, fila (queuing), transmissão e propagação)
- ◆ Protocolos



Ementa 2 - Protocolos

- ◆ Modelo OSI
- ◆ Camada de Transporte (4)
 - ◆ TCP, UDP e ICMP
 - ◆ Introdução à Multiplexação
 - ◆ Portas e Serviços, Formato do Segmento, Características dos Protocolos (UDP vs. TCP)
 - ◆ TCP: Tráfego orientado à conexão
 - ◆ Número Seqüência, Número Reconhecimento
 - ◆ Estimativa de RTT
 - ◆ Controle de Fluxo e Controle de Congestionamento
- ◆ Camada de Rede (3)
 - ◆ IP
 - ◆ Endereços IPs e Classes (e porque não existem?), CIDR
 - ◆ Formato dos Pacotes
 - ◆ Resolução de Nomes para IP (DNS é da camada 7)
 - ◆ Roteamento Estático vs. Roteamento Dinâmico (RIP, OSPF, BGP4)
 - ◆ **QoS (Quality of Service)**



Ementa 3 – Futuro das Redes

- ◆ Entenderemos o futuro das redes
 - ◆ Estatísticas
 - ◆ Negócios
 - ◆ Tecnologia
 - ◆ Segurança

- ◆ Estimado em cerca de 60 minutos de apresentação (60 slides)



Introdução a Redes de Computadores



Introdução – O que é Rede?

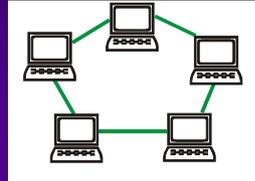
- ◆ Definição
 - ◆ Dois ou mais computadores conectados um ao outro por um meio de transmissão usando o mesmo protocolo de comunicação.
 - ◆ Divididas em Redes Locais (LAN) e redes de longa distância (WAN)
- ◆ Motivação
 - ◆ Facilitar o compartilhamento de informações
 - ◆ Compartilhamento de recursos caros (discos/impressoras)
 - ◆ Centralização Administração
 - ◆ Aumentar Eficiência



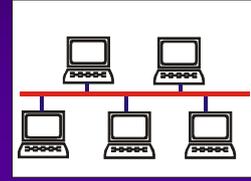
Tecnologias de Redes

◆ Topologias:

- ◆ Barramento
- ◆ Estrela
- ◆ Anel



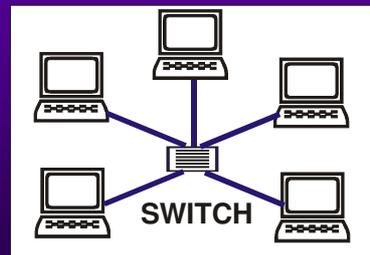
Anel



Barramento

◆ Tipos Redes

- ◆ Cliente – Servidor
 - ◆ Servidor compartilha recursos
 - ◆ Não precisa ser o maior da rede
- ◆ Par – a – par
 - ◆ Todos são clientes e servidores



Estrela



Topologias de Rede

◆ Estrela

- ◆ Ethernet com Switch

◆ Anel

- ◆ FDDI
- ◆ Token Ring

◆ Barramento

- ◆ Ethernet com coaxial

◆ Tecnologia Ethernet

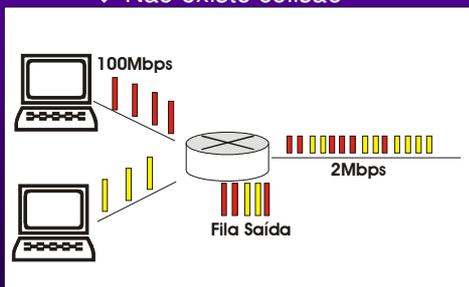
- ◆ Colisões em Ethernet
 - ◆ Quadros com 46bytes
- ◆ Hubs x Switches
 - ◆ Switches mais inteligentes
- ◆ Regra 5-4-3
 - ◆ 5 segmentos rede
 - ◆ 4 repetidores (hubs)
 - ◆ 3 podem conter estações



Tipos de Transmissões

◆ Rede de Circuitos

- ◆ Telefones
- ◆ Tecnologia FDM (ar)
- ◆ Circuitos são alocados
- ◆ Reserva antecipada de recursos (desperdício)
- ◆ Não existe colisão



◆ Redes por comutação de pacotes

- ◆ Internet
- ◆ Tecnologia TDMA/CDMA
- ◆ Sem reserva de recursos
- ◆ Estatisticamente comporta 3,5~4 vezes a quantidade de usuários
- ◆ Colisões são possíveis em Ethernet (ver trabalho CSMA/CD). Probabilidade calculada por:

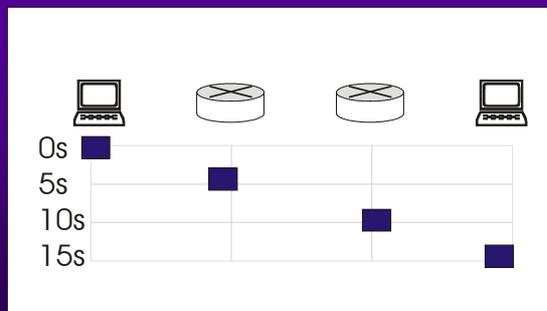
$$P(A) = \frac{n(A)}{n(S)} = \left(\frac{1}{10}\right) = 0,1$$

← Comutação de Pacotes



Redes de Comutação de Pacotes - Segmentação

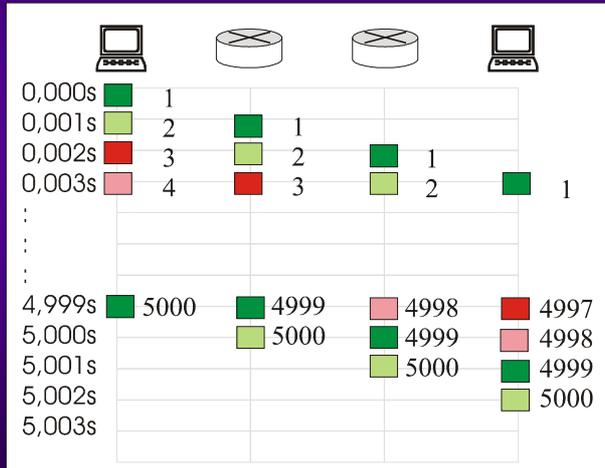
- ◆ Técnica store-and-forward
- ◆ Uso de Segmentação de Pacotes
- ◆ Quase todos usam segmentação de pacotes



Sem Segmentação



Redes de Comutação de Pacotes - Segmentação



Com Segmentação – Tarefa da Camada de Rede



Protocolos de Redes



Protocolos?

- ◆ Protocolos são regras de como mensagens (pacotes) são enviados pelas redes
 - ◆ Compatibilidade e Interoperabilidade
 - ◆ Protocolos são usados em camadas
- ◆ Protocolos de rede podem ser roteáveis ou não roteáveis
 - ◆ IPX e IP são roteáveis
 - ◆ NetBEUI não é roteável
 - ◆ Grande maioria das redes usam IP
- ◆ Nos anos 80, redes isoladas. Protocolos que permitem conectar duas LANs são roteáveis.

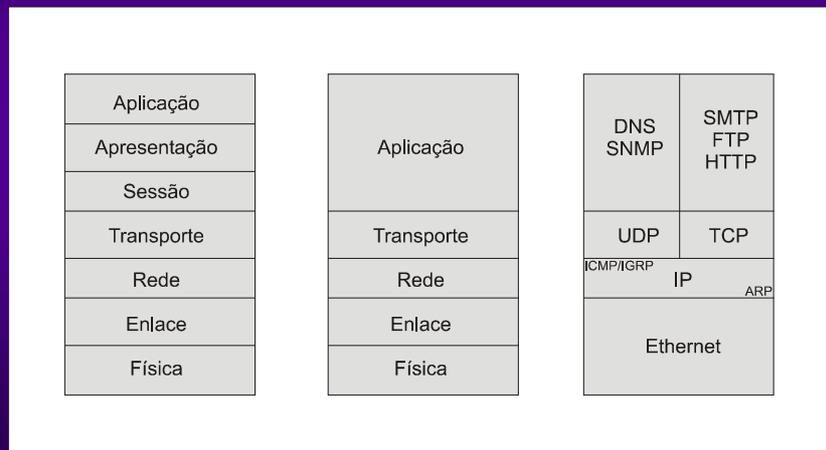


TCP/IP

- ◆ Atualmente versão 4
 - ◆ Internet 2 usa IPv6
- ◆ Roteável
- ◆ TCP/IP não é UM protocolo, é uma suíte de protocolos.
 - ◆ Camadas 3 à 7 do modelo OSI (em ATM usa AAL5)
 - ◆ Independente de meio físico
- ◆ Utiliza endereços IP (Internet Protocol)



Protocolos e Camadas



Camadas TCP/IP

- ◆ Pacote tem nome diferente em cada camada:
 - ◆ Quadro (enlace)
 - ◆ Datagramas (rede)
 - ◆ Segmentos (transporte)
 - ◆ Mensagem (aplicação)
- ◆ Hubs x Switches x Bridges x Routers
 - ◆ Hubs Nível 1 (elétrico/físico)
 - ◆ Switches Nível 2 (ethernet/MAC Address)
 - ◆ Bridges Nível 2
 - ◆ Roteadores Nível 3 (Rede/IP)
 - ◆ Casos especiais de Switches Nível 7 (aplicação)



Protocolos e Camadas

Aplicação	Aplicações, Gateways ou Switches Level 7 (Firewalls normalmente nível 4)
Apresentação	
Sessão	
Transporte	
Rede	Roteadores
Enlace	Switches
Física	Hubs e cabos



Cabeçalho Camada Rede

IP versão 4

Bits																															
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Versão		Tam. Cabeçalho		Tipo de Serviço												Tamanho do Datagrama															
Identificador												Flags				Fragment Offset															
Tempo de Vida				Protocolo								Checksum do Cabeçalho																			
Endereço IP de Origem																Endereço IP de Destino															
Opções																Complemento															
Dados																															

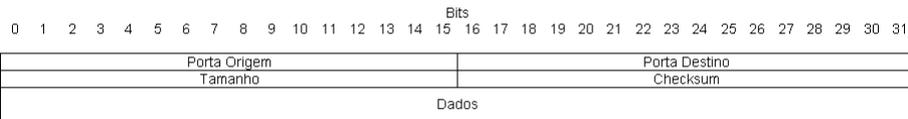
IP versão 6

Bits																															
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Versão		Classe de Tráfego						Identificador de Fluxo																							
Tamanho do Cabeçalho						Próximo Cabeçalho								Limite de Passos (hops)																	
Endereço IP Origem (128 bits)																															
Endereço IP Destino (128 bits)																															
Dados																															

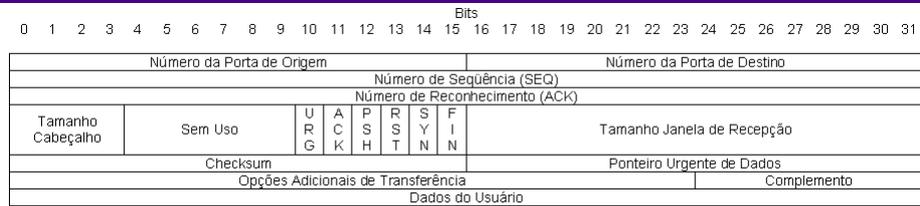


Cabeçalhos Protocolo Transporte

UDP



TCP

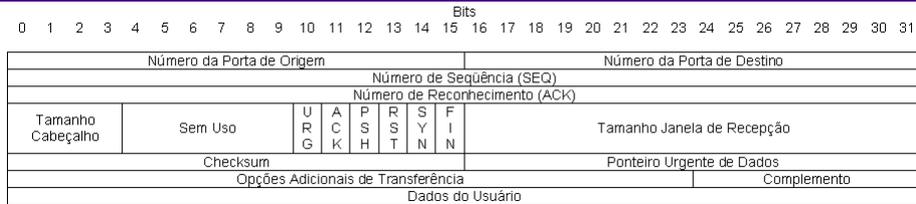


Camada Transporte

TCP, UDP, ICMP



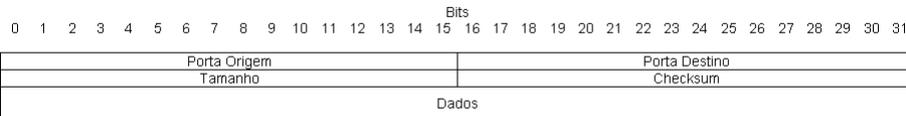
Cabeçalho TCP (Camada 4)



Ponteiro URG (Urgent Pointer Field Significant – 1 bit – Pouco Usado):
Ponteiro ACK (Acknowledgment Field Significant – 1 bit):
Ponteiro PSH (Push Function – 1 bit – Pouco Usado):
Ponteiro RST (Reset connection – 1 bit):
Ponteiro SYN (Synchronize sequence numbers – 1 bit):
Ponteiro FIN (No more data from sender – 1 bit):



Cabeçalho UDP (Camada 4)



- ◆ Porque UDP tem Checksum?
 - ◆ O que acontece quando há erro?
 - ◆ Descarte ou passa a aplicação com flag erro.



Protocolo de Transporte TCP vs. UDP

◆ UDP

- ◆ Cabeçalho 8 bytes
- ◆ Orientado à Datagrama
- ◆ Sem Garantia Entrega Pacotes
- ◆ Rápido

◆ Mas porque devemos usar UDP?

◆ TCP

- ◆ Cabeçalho 20 bytes
- ◆ Orientado a Conexão
- ◆ Garantia de Entrega de Pacotes
 - ◆ Reconhecimentos (ACKs)
 - ◆ NACK inferido por perda de pacote ou três "ACKs duplicados"
- ◆ Controle Fluxo e Congestionamento
 - ◆ Janelas Deslizantes
- ◆ Precisa estabelecer conexão (Three-way-handshaking)



Protocolo de Transporte TCP vs. UDP

◆ Aplicações que usam UDP:

- ◆ DNS
- ◆ SNMP
- ◆ RADIUS (Autenticação)
- ◆ RIP
- ◆ NTP

◆ Aplicações que usam TCP:

- ◆ HTTP
- ◆ POP3
- ◆ SMTP
- ◆ SMB
- ◆ Telnet / SSH
- ◆ FTP



Protocolo de Transporte TCP vs. UDP

- ◆ Para uma aplicação “falar” com outra usa-se cinco parâmetros
 - ◆ Protocolo Transporte
 - ◆ IP Origem
 - ◆ IP Destino
 - ◆ Porta Origem
 - ◆ Porta Destino
- ◆ Portas identificam qual aplicativo fala com um aplicativo específico.
 - ◆ Exemplos:
 - ◆ HTTP = TCP/80.
 - ◆ FTP = FTP/21.
 - ◆ SNMP = UDP/161.
 - ◆ POP3 = TCP/110.
 - ◆ SMTP = TCP/25.
 - ◆ DNS = UDP/53.

Lista mais completa pode ser encontrada

Windows XP/2000/NT -> C:\windows\system32\drivers\etc\services

Linux -> /etc/services



Flags do TCP – Bits de Controle

- ◆ **Ponteiro URG (Urgent Pointer Field Significant – 1 bit – Pouco Usado):** URG é usado em conjunto com PSH e também indica ao receptor que o campo de ponteiro urgente de dados contém um apontador para o último byte que deve ser passado para a aplicação.
- ◆ **Ponteiro ACK (Acknowledgment Field Significant – 1 bit):** Se habilitado (valor 1) significa que o este pacote possui um número de reconhecimento (ACK) válido. Isto permite que o reconhecimento venha junto com um pacote de dados.
- ◆ **Ponteiro PSH (Push Function – 1 bit – Pouco Usado):** Se habilitado (valor 1), significa que o receptor deverá passar os dados indicados para a aplicação mesmo que esteja fora de ordem (em situações normais, o destino poderá ter que aguardar que um pacote anterior chegue para que os dados sejam entregues em ordem para a aplicação).
- ◆ **Ponteiro RST (Reset connection – 1 bit):** Usado para reinicializar a conexão
- ◆ **Ponteiro SYN (Synchronize sequence numbers – 1 bit):** Serve para abrir uma nova conexão. Quando SYN é enviado, o host que receber significa que o número de sequência indica qual é o número do primeiro byte a ser enviado. Durante o estabelecimento de uma conexão a origem enviará SYN=1 com de um número de sequência X (primeiro byte). O host de destino receberá o pacote e responderá com o flag SYN=1 e o número de sequência Y (aleatório) e o número de reconhecimento (ACK Number) como X + 1, onde X é o número de sequência de quem pediu a conexão. Por fim a origem deve enviar um pacote com SYN=0, Número Sequência=Y+1 e Número de Reconhecimento X+2.
- ◆ **Ponteiro FIN (No more data from sender – 1 bit):** Serve para indicar que o host quer encerrar a conexão.



Outros Campos do TCP

- ◆ **Número da Porta de Origem (Source Port Number – 16 bits):** Contém um número inteiro entre 0 e 65535 que indica qual a porta de origem do cliente. Este número lógico identifica a aplicação que requisitou dados (origem). Isto serve para que o TCP saiba qual a aplicação que criou o pacote. Este número é único em um momento, ou seja, a porta de origem só pode ser usado por uma aplicação em um momento.
- ◆ **Número da Porta de Destino (Destination Port Number – 16 bits):** Contém um número inteiro entre 0 e 65535 que indica qual a porta de destino no servidor. Este número lógico indica qual aplicação deve responder a essa requisição. Quando o TCP recebe este pacote no servidor (por exemplo), é através deste número que sabe-se à qual aplicação deve ser repassado este segmento (data).
- ◆ **Número de Sequência (Sequence Number – 32 bits):** O TCP é um protocolo orientado à conexão, e prove entrega garantida, sem erros e em ordem. Para que o TCP consiga saber a ordem do pacote em relação ao stream original de dados, usa-se o Número de Sequência. No momento da conexão, o Número de Sequência é criado com um número aleatório. Os pacotes sucessores são o número de sequência anterior mais a quantidade de dados transferidos no último pacote.
- ◆ **Número de Reconhecimento (Acknowledgement Number – 32 bits):** Para que o TCP consiga confirmar o recebimento de um pacote, o destinatário envia neste campo o número do último byte, adicionado de um, recebido pelo servidor nesta conexão. Desta maneira se o remetente não receber o número de reconhecimento correto, ele poderá reenviar o pacote novamente para o servidor.
- ◆ **Tamanho do Cabeçalho (Header Length – 4 bits):** Indica qual o tamanho do cabeçalho do TCP (incluindo as opções adicionais). Desta maneira se um host receber este pacote, pode saber exatamente onde termina o cabeçalho e onde começam os dados do usuário.
- ◆ **Sem Uso (Reserved – 6 bits):** Um espaço que não está em uso e poderá ser usado no futuro. Todos os bits devem ser 0.
- ◆ **Tamanho da Janela de Recepção (Receive Window – 16 bits):** Utilizado para o controle de fluxo. Isto indica o número máximo da janela do receptor, ou seja, a quantidade máxima de bytes disponíveis para serem enviados ao receptor pelo transmissor sem receber um reconhecimento.
- ◆ **Checksum (Checksum – 16 bits):** O checksum é um valor binário de 16 bits que é um complemento de um de todas as palavras do cabeçalho e do texto.
- ◆ **Opções Adicionais de Transferência (Options – Tamanho variável, múltiplo 8 bits):** Opções podem ocupar o espaço no fim do cabeçalho do TCP. Todas as opções devem ser usados para cálculo do Checksum. A opção é dividida em tipo (8 bits), tamanho do campo desta opção somente e os dados. O tipo mais usado é o tipo 2 (binário 00000010) que indica o tamanho máximo do segmento suportado pelo host de origem. Por exemplo:

tipo	Tamanho	Valor
2	2	1024 bytes
00000010	00000010	0000010000000000

- ◆ **Padding (Tamanho variável):** Se o Tamanho das opções não for múltiplo de 32 bits, o padding pode ser completado por bits 0 para que o cabeçalho do TCP fique com tamanho correto.
- ◆ **Dados (Tamanho variável, máximo de bits definido pelo MSS):** Dados do Usuário.



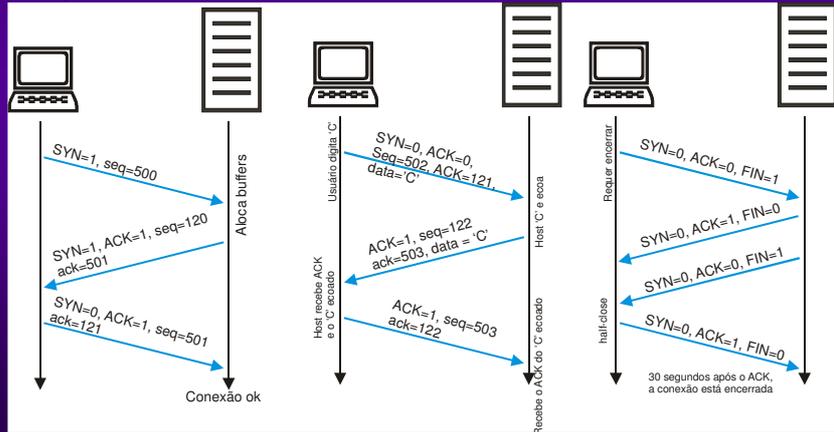
Seq Number e ACK Number (somente TCP)

- ◆ São campos para identificação para saber se o pacote chegou ou não ao destino
- ◆ Detecta perda de pacotes
- ◆ Detecta pacote duplicado
- ◆ Cada lado da comunicação usa um deles (TCP é um protocolo full-duplex).
- ◆ Somente fazem sentidos se o flag “ACK” estiver ativo.
- ◆ TCP/IP v4 usa ACK Cumulativo e SACK (TCPOptions)

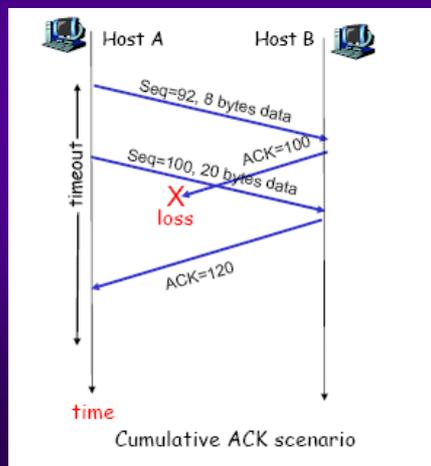


Seq Number e ACK Number (somente TCP)

- ◆ Vejamos um Three-way-handshaking e um telnet com o cliente digitando a letra "c":



ACK Cumulativo





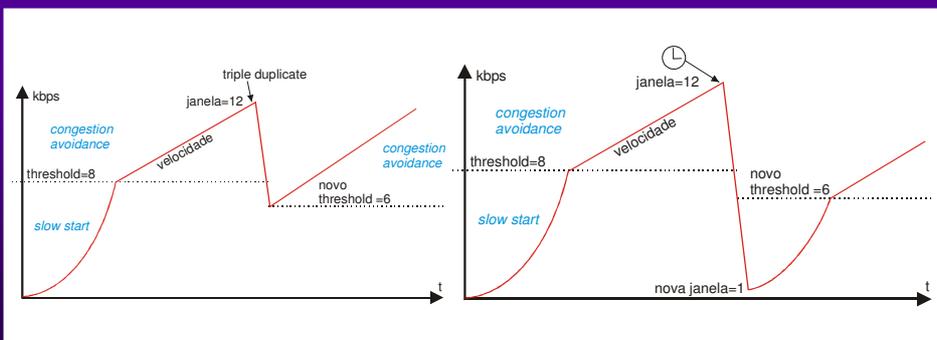
Quando enviar ACKs

Evento	Ação
Se chegarem segmentos em ordem, sem lacunas, e todos os anteriores já foram reconhecidos	Receptor dos dados pode esperar até 500ms para chegar pacotes, retardando o ACK. Se não chegar mais segmentos, envia um ACK.
Se chegarem segmentos em ordem sem lacunas e existe um ACK retardado pendente	Receptor deve enviar imediatamente um único ACK cumulativo.
Se chegou segmento fora de ordem, com número de seqüência maior do que o esperado, causando uma lacuna	Envia ACK triplicado, indicando o número de seqüência do próximo byte esperado
Se chegarem segmentos que preenche uma lacuna parcialmente ou completamente	Envia imediatamente um ACK cumulativo



Controle de Fluxo e Congestionamento (TCP only)

- ◆ Triple Duplicate (3 ACKs) é informativo de NACK
- ◆ Exemplos simples dos algoritmos TCP Reno e Tahoe, quando há timeout ou quando há triple duplicate





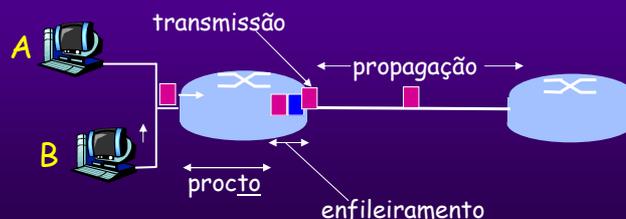
Controle de Fluxo e Congestionamento (TCP only)

- ◆ Campo “Tamanho da Janela de Recepção” é o $\min(\text{Controle Fluxo e Tamanho Buffer computador})$
- ◆ Ele usa os valores de RTT (round trip time), perda de pacote, capacidade do computador para calcular o tamanho da janela.
 - ◆ Controle Fluxo: Impede que a origem “inunde” o destino
 - ◆ Controle de Congestionamento: Impede que a origem “inunde” a rede e cause packet loss.



Entendendo atrasos

- ◆ Atrasos em 4 locais:
 - ◆ Processamento: Examinar header e fazer decisão (routing ou switching) - fixo
 - ◆ Fila (queuing): Espera para que seja a vez adequada para transmissão - variável
 - ◆ Transmissão: Tempo desde o primeiro bit até o último bit para colocar no cabo ou gerar os pulsos elétricos - fixo
 - ◆ Propagação: Tempo que o sinal leva para chegar no destino (entre $2 \cdot 10^8$ m/s e $3 \cdot 10^8$ m/s)





Estimativa de RTT

- ◆ É o tempo entre o envio do pacote (pelo transporte) para a camada IP, até o recebimento do ACK
- ◆ Necessita acompanhamento no decorrer do tempo (flutuação)



Estimativa RTT

- ◆ RFC2988 define $\alpha = 1/8$, $\beta = 1/4$
 $EstRTT = (1 - \alpha) \cdot EstRTT + \alpha \cdot RTTLstPck$
 $DevRTT = (1 - \beta) \cdot DevRTT + \beta \cdot |RTTLstPkt - EstRTT|$

Logo:

$$EstRTT = 0,875 \cdot EstRTT + 0,125 \cdot RTTLstPkt$$
$$DevRTT = 0,75 \cdot DevRTT + 0,25 \cdot |RTTLstPkt - EstRTT|$$

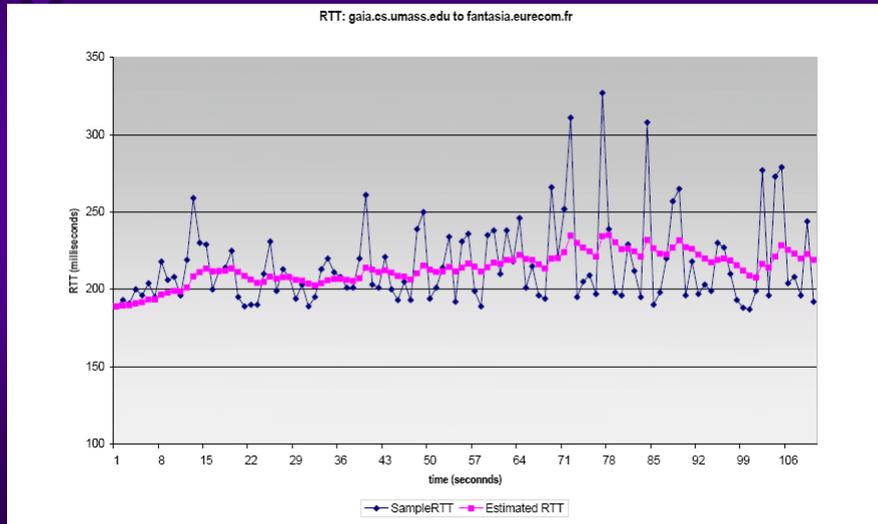
- ◆ É uma média ponderada (últimos pacotes pesam mais que os mais velhos). Foi chamada de EWMA (Exponential weighted moving average).

$$\text{Timeout} = EstRTT + 4 \cdot DevRTT$$

É o tempo que o TCP aguarda para “estourar” o relógio



Estimativa RTT



ICMP

- ◆ Acrônimo de Internet Control Message Protocol
- ◆ Criado para trocar mensagens de erro nas redes
- ◆ Roda em nível de rede (3,5 ☺), sobre o IP diretamente
- ◆ Mensagens comuns:
 - ◆ Network Unreachable
 - ◆ Port Unreachable
 - ◆ Echo Request/Reply
 - ◆ Ping
 - ◆ Traceroute



ICMP - Mensagens

O protocolo ICMP (Internet Control Message Protocol) é usado por hosts, roteadores, gateways para transmitir mensagens de um para outro. A tarefa mais comum é o envio das mensagens de erro, tais como "Host Não Alcançável", "Porta não disponível", etc. Suas mensagens são transmitidas sobre o protocolo IP.

Tipo	Código	Descrição
0	0	resposta de eco (ping)
3	0	rede destino inalcançável
3	1	estação destino inalcançável
3	2	protocolo destino inalcançável
3	3	porta destino inalcançável
3	6	rede destino desconhecida
3	7	estação destino desconhecida
4	0	abaixar fonte (controle de congestionamento - ã usado)
8	0	pedido eco (ping)
9	0	anúncio de rota
10	0	descobrir roteador
11	0	TTL (sobrevida) expirada
12	0	Erro de cabeçalho IP



Protocolo de Rede

IP

Internet Protocol version 4



Camada de rede

- ◆ Tarefas:
 - ◆ Identificar os hosts em uma rede
 - ◆ Prover roteamento entre redes
 - ◆ No Protocolo IP:
 - ◆ Estático: Definido pelo admin
 - ◆ Dinâmico: RIP, OSPF, BGP4
 - ◆ Segmentar (ou unir) pacotes – MTU/MSS
 - ◆ Estabelecimento de conexões: ATM requer call-setup
 - ◆ Camada 3: Independe do meio físico (IP roda em Ethernet, ATM, Wireless, Celular, etc)



Cabeçalho Camada Rede

IP versão 4

Bits																															
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Versão				Tam. Cabeçalho				Tipo de Serviço								Tamanho do Datagrama															
Identificador								Flags								Fragment Offset															
Tempo de Vida								Protocolo								Checksum do Cabeçalho															
Endereço IP de Origem																Endereço IP de Destino															
Opções																Complemento															
Dados																															



Campos do IP

- ◆ **Versão (Version - 4 bits):** A versão do protocolo IP. Atualmente a Internet usa a versão 4 e a Internet2 usa o IP versão 6.
- ◆ **Tamanho Cabeçalho (Header Length - 4 bits):** Tamanho em bytes do cabeçalho (incluindo as opções).
- ◆ **Tipo de Serviço (Type of Service - 8 bits):** O tipo de serviço (TOS) foi incluído para comportar diferentes tipos de datagramas IP (datagramas e as particularidades como TTL controlado, banda, etc). Roteadores podem observar este campo para definir políticas definidas por um administrador.
- ◆ **Tamanho do Datagrama (Datagram Length - 16 bits):** Indica o tamanho total do datagrama em bytes. Este valor pode ser até 65535 (216-1), entretanto este valor normalmente nunca é maior que 1500 bytes.
- ◆ **Flags, Identificação e Dados de Fragmentação (Ident - 16 bits, Flags - 3 bits, Fragment Offset - 13 bits):** Estes três campos servem para o controle da fragmentação de um pacote IP.

Flag com Bit 0:	Reservado	1 = Fragmentação Não Permitida
Flag com Bit 1:	0 = Possível Fragmentar	1 = Existem mais fragmentos deste datagrama.
Flag com Bit 2:	0 = Último Fragmento	
- ◆ **Tempo de Vida (Time to Live - 8 bits):** Este campo impede que um pacote seja transmitido em loop para sempre na internet. Toda vez que este datagrama passa por um roteador, este valor é decrementado. Quando este valor é 0, o datagrama é descartado.
- ◆ **Protocolo (Protocol - 8 bits):** Este campo é usado para que o sistema-fim possa identificar qual é o conteúdo do pacote (por exemplo um pacote TCP, UDP, ICMP, etc).
- ◆ **Checksum do Cabeçalho (Header Checksum - 16 bits):** Este campo é usado para que os roteadores possam saber se o cabeçalho está íntegro e se os dados não foram corrompidos durante a transferência. Se o checksum falhar, o roteador descartará o pacote.
- ◆ **Endereço IP de Origem (Source Address - 32 bits):** Este campo diz o endereço IP da origem do pacote (o sistema que criou o datagrama).
- ◆ **Endereço IP de Destino (Destination Address - 32 bits):** Este campo diz o endereço IP de destino (o sistema que deverá receber o datagrama).
- ◆ **Opções e Complemento (Options - Tamanho Variável, Padding - Tamanho Variável):** Alguns datagramas especiais podem usar as opções para os roteadores. Pouco Usado. O Complemento é um campo que serve para preencher com 0 até que o tamanho do cabeçalho tenha um tamanho múltiplo de 32 bits.
- ◆ **Dados (Data - Tamanho Variável):** Os dados propriamente dito. Normalmente neste campo é carregado um pacote de outro protocolo, como o TCP ou o UDP.



Protocolo IP

- ◆ **Endereços IP parecem com**
200.150.138.1
 - ◆ 4 octetos de 1 byte separados por ponto
 - ◆ Aproximadamente 3,7 bilhões endereços
- ◆ **Identificam unicamente um host na rede (como telefones no mundo)**
- ◆ **Divididos em 3 partes:**
 - ◆ Rede
 - ◆ SubRede
 - ◆ Host

Todos bits 0 ou todos bits 1 são proibidos



Classes de IPs

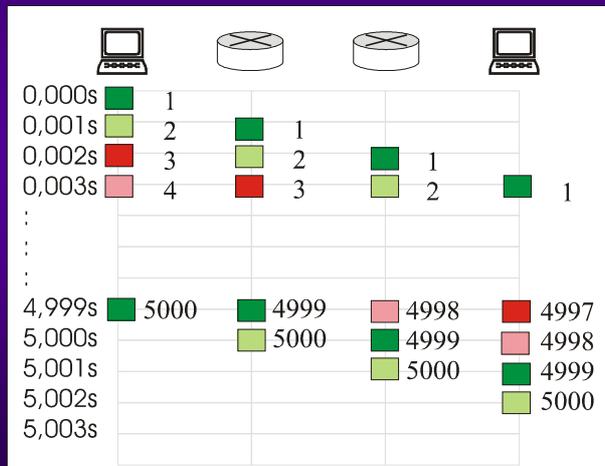
- ◆ Endereços IP possuem classes
 - ◆ Classe A (1.0.0.0 à 126.255.255.255)
 - ◆ Classe B (128.0.0.0 à 191.255.255.255)
 - ◆ Classe C (192.0.0.0 à 223.255.255.255)
 - ◆ Classe D (224.0.0.0 à 239.255.255.255)
- ◆ Notação CIDR (Classless InterDomain Routing) provê qualquer quantidade de hosts e redes

Exemplo CIDR: 200.174.0.0/16

1 2 3 4 5 6 7 8 9	1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 3 3 3			
0	0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2	rede	host	Classe A = 8 bits para rede / 24 bits para host 1.0.0.0 até 127.255.255.255
1 0		rede	host	Classe B = 16 bits para rede / 16 bits para host 128.0.0.0 até 191.255.255.255
1 1 0		rede	host	Classe C = 24 bits para rede / 8 bits para host 192.0.0.0 até 223.255.255.255
1 1 1 0		Multicast		Classe D = Multicast 32 bits 224.0.0.0 até 239.255.255.255



Redes de Comutação de Pacotes - Segmentação



Segmentação – Tarefa da Camada de Rede (se a aplicação não enviar do tamanho certo)



Segmentação de pacotes

Cabeçalho Original (4000 bytes)

Tamanho	ID	Flag fragmento	Início	Dados
4000	N	0	0

3 novos pacotes de 1500Bytes de MTU

Tamanho	ID	Flag fragmento	Início	Dados
1500	N	1	0

Tamanho	ID	Flag fragmento	Início	Dados
1500	N	1	1480

Tamanho	ID	Flag fragmento	Início	Dados
20(IP)+ 1020(Data)	N	0	2960

$N = (16 \text{ bits unsig int})$ identifica de qual comunicação é esse segmento

Nota: O início contém 20 bytes a menos do que o datagrama devido que o mesmo possui 20 bytes de cabeçalho IP



Segmentação

- ◆ Cada pacote pode escolher uma rota diferente
- ◆ Não é tarefa da camada de rede enviar “em ordem”
- ◆ Comutação de pacotes pode gerar problemas de colisão e congestionamento



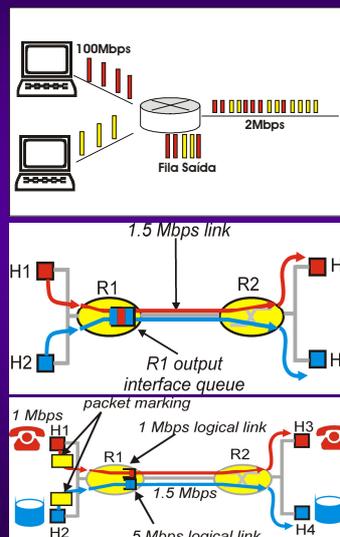
“QoS” em IP - Reflexão

- ◆ Internet é BestEffort
- ◆ Aplicações atuais são elásticas (suportam atrasos, perdas, jitter, etc)
- ◆ “Devemos construir aplicações mais flexíveis e adaptáveis ou devemos mudar a Internet para suportar um comportamento ‘inelástico’?”
- ◆ Grupo IETF trabalham para propor QoS em redes IPs, para prover algum tipo de QoS.
- ◆ Atualmente em estudos estão RSVP, Differentiated Services(DiffServ), and Integrated Services (IntServ)
- ◆ Não esquecer de fazer engenharia de tráfego.



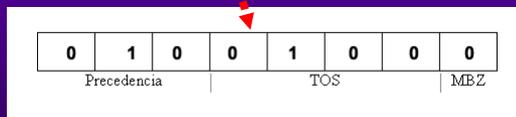
“QoS” em IP

- ◆ Depende da “boa vontade” do Roteador
- ◆ Necessita políticas de Admissão e políticas de permissão
- ◆ É statefull (Não escalável) em IntServ, ou usa ToS em DiffServ
- ◆ É mais funcional em ATM
- ◆ Normalmente necessita agregação de fluxo ou algo como um “Virtual Path”
- ◆ Devemos pensar em Prioridade, Isolamento de tráfego, Controle de admissão, classificação de pacotes.





“QoS” em IP (Flag ToS) Normalmente usado em Diffserv



Precedência = 2, TOS = 4, MBZ (*Must Be Zero*) = Não usado



“QoS” em IP (RFC1349)

- ◆ Precedência é a prioridade do pacote

- ◆ **Precedência:**

111	Controle da Rede	011	Flash
110	Controle Inter-rede	010	Imediato
101	CRITIC/ECP	001	Prioritário
100	Flash Override	000	Normal

- ◆ TOS (Tipo de Serviço)

1000	8	Minimizar o delay (RTT)
0100	4	Maximizar Throughput
0010	2	Maximizar Reability (Packet Loss)
0001	1	Minimizar Custo (Hops)
0000	0	Serviço Best-Effort Padrão

- ◆ MBZ = Não utilizado, reservado para futuro



Conclusão

- ◆ Não existe no TCP nenhuma maneira de prover qualidade de serviço já que o mesmo não prove nenhum campo para o uso por um algoritmo QoS
- ◆ Não confundir “garantia de entrega ordenada” com QoS
- ◆ O IP possui algumas poucas maneiras inefetivas para QoS
- ◆ Roteadores só “enxergam” IP e QoS é feito por routers, logo TCP não provê QoS.
- ◆ IP possui arquiteturas de QoS (diffserv, intserv, rsvp (setup protocolo), etc)
- ◆ QoS “de verdade” requer apoio da rede (callsetup, virtualpaths, aviso de congestionamento, reserva de recursos)

Bits																															
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Versão				Tam. Cabeçalho				Tipo de Serviço								Tamanho do Datagrama															
Identificador								Flags								Fragment Offset															
Tempo de Vida								Protocolo								Checksum do Cabeçalho															
Endereço IP de Origem																Endereço IP de Destino															
Opções																Complemento															
Dados																															



Revisão Protocolos Redes

- ◆ Dividido em camadas
- ◆ Camadas tem tarefas diferentes
 - ◆ Modelo Internet tem camadas física, enlace, rede, transporte e aplicação
- ◆ Protocolos são “padrões” nos formatos dos pacotes.
- ◆ HTTP, POP3, SMTP, DNS camada de aplicação
- ◆ TCP/UDP são camadas de transporte (identificam aplicação através de número de porta)
- ◆ IP é camada de rede (identifica o host através de endereços IP)



Entendendo o futuro das redes



O que é a Internet?

- ◆ Redes conectadas em outras redes (redes de redes)
- ◆ Computadores compartilham formas de envio (protocolos em comum)

ARPANET - 1969-1990
AlohaNet - 1970
Packet Radio, SATNET - 1973
Ethernet (Xerox PARC) 1973
Internet Dev't - 1973-1983
NSFNET - 1986-1995



Internet – Histórico (marcos)

- ◆ 1969 - 1985 - Basic Packet Net Research
- ◆ 1974 – Publicado primeiro design Internet
- ◆ 1983 – primeiro desenvolvimento (ArpaNet com TCP/IP)
- ◆ 1986 – primeira companhia de roteadores (Cisco)
- ◆ 1989 - WWW; MCI Mail/Internet link
- ◆ 1989 – Serviços Comerciais (UUNet, PSINet, CERFNet and CIX)
- ◆ 1990 - ARPANET for do ar;
- ◆ 1994 - WWW para Público (Browser Netscape)
- ◆ 1995 - NSFNet retired, competitive backbone
- ◆ 2002 – 8 milhões de usuários no Brasil
- ◆ 2005 – 3 milhões de hosts,



Estatísticas

22.5 Milhões Hosts
(Bellcore Junho 1997)

317 Milhões Hosts
(ISC Maio 2005)

50 Milhões Usuários
(NUA Julho 1997)

605 Milhões Usuários
(NUA Setembro 2002)

190 países IP
(NW Julho 1997)

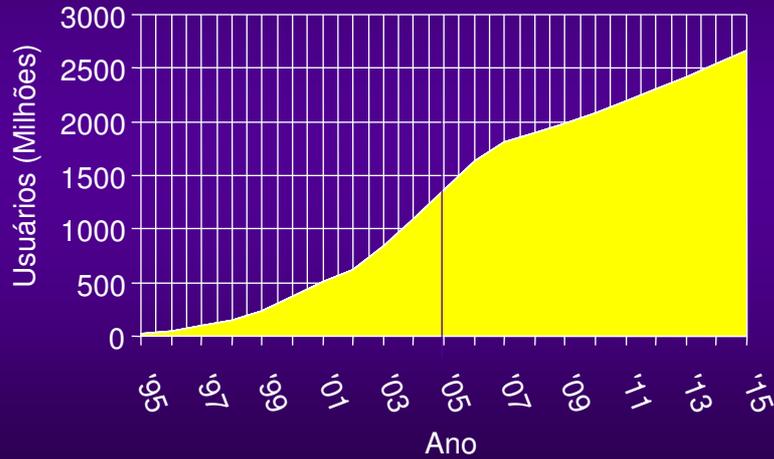
218/246 países IP
(NW Janeiro 2001)

1,3 bilhões telefones

Est: 1,2G users em Dez/2005



Estatísticas

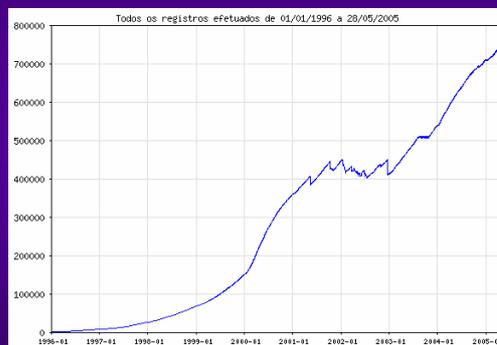


Source: Nua Internet Surveys + vgc projections



Estatísticas (2004)

Posição	País	Hosts (milhões)
1	Estados Unidos	162,1
2	Japão	12,9
3	Itália	5,4
4	UK	3,7
5	Alemanha	3,4
6	Holanda	3,2
7	Canadá	3,2
8	Brasil	3,163
9	Austrália	2,8
10	Taiwan	2,7



*Registros de domínio
No Brasil durante os últimos
8 anos*

Em 2003 eram apenas 2,2M e Brasil era o #9



Tecnologia

- ◆ Internet
 - ◆ Rede Comutação Pacotes
 - ◆ Sem Garantia (BestEffort)
 - ◆ Ninguém é dono
 - ◆ RFC, IETF, IANA, ARIN, ICANN, IPv6 Forum
- ◆ Endereços IP
 - ◆ Consumo IP alto
 - ◆ IPs são escassos (57% em uso)
 - ◆ Solução: IPv6 (Internet2)
 - ◆ Atuais IPv6 tunelados em IPv4



Tecnologia

- ◆ Redes no núcleo de alta velocidade
 - ◆ ISPs locais à Mbps
 - ◆ ISPs nacionais à Gbps
 - ◆ Provedores de backbone com vários Gbps
 - ◆ Cogita-se transferências à Tbps (terabits/s)
- ◆ Custos para usuários:
 - ◆ 1997 = R\$ 90,00
 - ◆ 2003 = R\$ 15 (0 talvez)
 - ◆ Banda Larga
 - ◆ 2003 = R\$ 100,00
 - ◆ 2009 = R\$???

Bits/s x Bytes/s



Segurança

- ◆ Ataques
 - ◆ Escuta (eavesdropping)
 - ◆ Switches Ajudam
 - ◆ Dificuldades nos ataques
 - ◆ Rotas de Retorno
 - ◆ ICMP Redirects
 - ◆ Firewalls
 - ◆ Criptografia Forte
- ◆ IP Spoofing
 - ◆ Falsificação IP
- ◆ SYN Attacks
- ◆ DoS / DDoS (Distributed Denial of Service)
- ◆ Firewalls
 - ◆ Ajudam?
- ◆ Ataques de Aplicação
 - ◆ Possível Root, buffer overflow
- ◆ LOGs
 - ◆ Horários Sincronizados
 - ◆ Armazenamento 5 Anos



Incidentes de Segurança no BR





Segurança

Segurança leva os administradores a loucura ☺

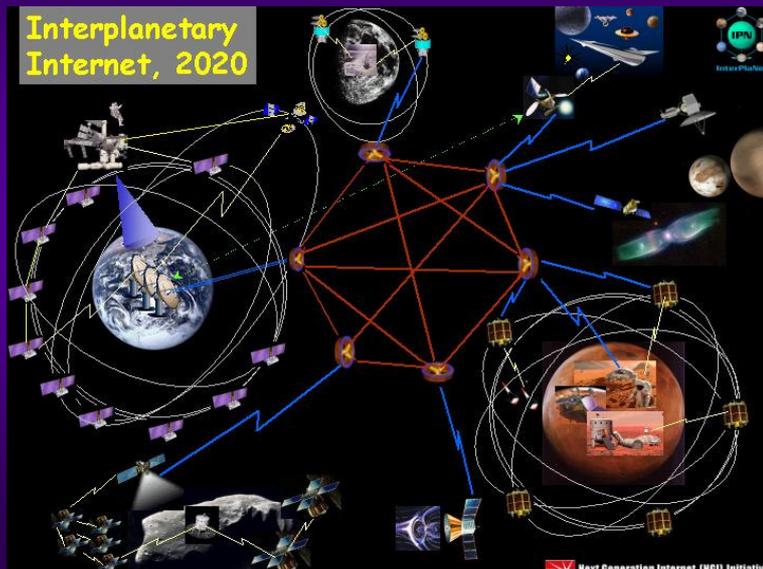
Mais estudo, mais conhecimento, mais neurótico é o administrador!

Fernando Braghetto



Futuro da Internet

Interplanetary Internet, 2020





Futuro da Internet

- ◆ InterPlaNet (Interplanetary Internet)
 - ◆ Projeto NASA
 - ◆ Testes em 2020
 - ◆ IP com altas latências
 - ◆ Vicent Cerf é um dos projetistas (MCI/Worldcom)



Futuro da Internet

- ◆ Redundância e Disponibilidade
- ◆ Segurança
 - ◆ Proteção Roteadores, Servidores DNS e Clientes
 - ◆ Criptografia Forte Fim-a-fim (VPN)
 - ◆ Firewalls
 - ◆ Privacidade / Autenticação (SmartCards)
 - ◆ Terceira Parte Confiável (?)
 - ◆ Spam acabará e-mail (53% spam)
- ◆ Wireless LAN (WLANs)
 - ◆ Acesso Alta Velocidade para usuários finais a baixo custo.



Futuro da Internet

- ◆ Internet 2 = IP versão 6
 - ◆ WebTVs, Palms, Câmeras Vigilância
 - ◆ Geladeira, Microondas terão IPs (celulares 4G??)
 - ◆ Telefones IPs
 - ◆ Mais Velocidade, maior gama serviços



Fontes

- ◆ Kurose e Ross – Computer Networking 2th Edition – Top down approach featuring the Internet. Editora Addison Wesley
- ◆ Microsoft System Engineer – Apostila TCP/IP
- ◆ Nua Internet Surveys + vgc projections - http://www.nua.ie/surveys/how_many_online/index.html
- ◆ Registro.br (Estatísticas)
- ◆ Apresentação de Vicent Cerf (MCI/Embratel Brasil, 2001)
- ◆ http://www.fccn.pt/crc2000/CRC2000_6.1.PDF
- ◆ <http://www.mci.com/cerfsup>
- ◆ <http://www.cse.ohio-state.edu/~jain/talks/ftp/ipqos/>
- ◆ <http://www.ietf.org/rfc/rfc1349.txt>
- ◆ <http://www.isc.org/index.pl?/ops/ds/new-survey.php>
- ◆ <http://www.cg.org.br/indicadores/brasil-mundo.htm#mundo>
- ◆ <http://www.nbso.nic.br/stats/incidentes/>
- ◆ <http://www.cs.umd.edu/~shankar/417-F01/Slides/chapter3b.ppt>
- ◆ [http://www.cs.huji.ac.il/course/2004/com1/Presentations/2004-5/Exercises/5_TCP/XLesson5-TCP-\(4spp\).pdf](http://www.cs.huji.ac.il/course/2004/com1/Presentations/2004-5/Exercises/5_TCP/XLesson5-TCP-(4spp).pdf)
- ◆ **QoS:** <http://www-2.cs.cmu.edu/afs/cs/academic/class/15441-f01/www/lectures/lecture22.ppt>



Introdução a Redes de Computadores

Fernando Braghetto

Hiway Internet Provider

Maio/2005

duvidas@braghetto.eti.br