CURSO IPv6 BÁSICO

Rodrigo Regis dos Santos
Antônio M. Moreiras
Eduardo Ascenço Reis
Ailton Soares da Rocha

Núcleo de Informação e Coordenação do ponto BR

São Paulo
2010

Núcleo de Informação e Coordenação do Ponto BR

Autores / Design / Diagramação
Rodrigo Regis dos Santos
Antônio Marcos Moreiras
Eduardo Ascenço Reis
Ailton Soares da Rocha

Sobre o Projeto IPv6.br

O IPv6 é a nova geração do Protocolo Internet.

Ele já vem sendo utilizado há algum tempo. Mas agora sua implantação deve ser acelerada. Ela é imprescindível para a continuidade do crescimento e da evolução da Internet!

O objetivo do projeto IPv6.br do NIC.br é estimular a utilização do novo protocolo na Internet e nas redes brasileiras. Para saber mais acesse o sítio Internet **www.ipv6.br** ou entre em contato pelo e-mail **ipv6@nic.br**.

O **CEPTRO.br**- Centro de Estudos e Pesquisas em Tecnologia de Redes e Operações do NIC.br – é responsável por projetos que visam melhorar a qualidade da Internet no Brasil e disseminar seu uso, com especial atenção para seus aspectos técnicos e de infraestrutura. Mais informações podem ser obtidas no sítio Internet **www.ceptro.br**.

Sobre os autores

Rodrigo Regis dos Santos é Bacharel em Ciência da Computação pela Universidade Presbiteriana Mackenzie e atualmente está especializando-se em Gestão e Infraestrutura de Telecomunicações pela mesma Universidade. Especialista em IPv6, trabalha no NIC.br atuando como um dos responsáveis pelo projeto IPv6.br, que tem por objetivo incentivar o uso do protocolo no país.

Antonio M. Moreiras é engenheiro eletricista e mestre em engenharia pela POLI/USP, MBA pela UFRJ, além de ter estudado Governança da Internet na Diplo Foundation e na SSIG 2010. Trabalha atualmente no NIC.br, onde está envolvido em projetos para o desenvolvimento da Internet no Brasil, como a disseminação do IPv6, a disponibilização da Hora Legal Brasileira via NTP, estudos da Web, ENUM, entre outros.

Eduardo Ascenço Reis é especialista em redes IP, sistemas Unix e serviços Internet. Como formação, possui Curso de Especialização em Redes de Computadores pelo LARC/USP e Bacharelado em Ciências Biológicas pela USP. Sua atuação profissional em TI ocorre desde 1995, com experiências nas empresas: Universidade de São Paulo (USP), Ericsson Brazil, comDominio (IDC - AS16397), CTBC Multimídia (NSP, ISP - AS27664). Atualmente atua como supervisor de projetos no Centro de Estudos e Pesquisas em Tecnologia de Redes e Operações (CEPTRO.br) do Núcleo de Informação e Coordenação do Ponto BR (NIC.br) e é um dos responsáveis pelo PTTmetro (PTT.br).

Ailton Soares da Rocha é Analista de Projetos no NIC.br, onde está envolvido com pesquisas e projetos relacionados à infraestrutura da Internet no país, com importante atuação nos projetos IPv6.br e NTP.br. Engenheiro eletricista e de telecomunicações graduado pelo INATEL, atuou por mais de 10 anos na coordenação da área de redes e Internet da instituição.

IPv6.br

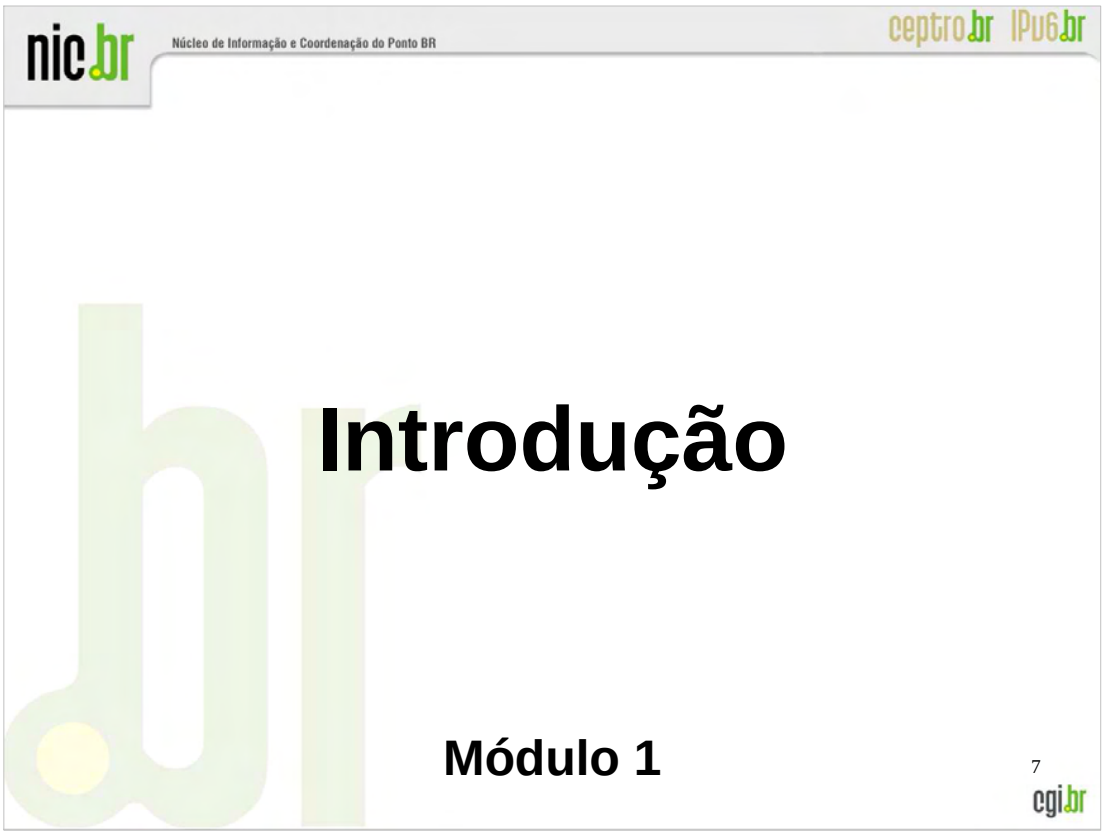
IPv6.br

Sobre a licença

- Para cada novo uso ou distribuição, você deve deixar claro para outros os termos da licença desta obra.
- No caso de criação de obras derivadas, os logotipos do CGI.br, NIC.br, IPv6.br e CEPTRO.br não devem ser utilizados.
- Na atribuição de autoria, essa obra deve ser citada da seguinte forma:
    - Apostila “Curso IPv6 básico” do NIC.br, disponível no sítio http://curso.ipv6.br ou através do e-mail ipv6@nic.br.
- Qualquer uma destas condições podem ser renunciadas, desde que você obtenha permissão do autor.
  Se necessário, o NIC.br pode ser consultado através do email ipv6@nic.br.
- Nada nesta licença prejudica ou restringe os direitos morais do autor.

IPv6.br

A Nova Geração do Protocolo Internet



Neste módulo de introdução, iniciaremos conhecendo um pouco da história da Internet e do desenvolvimento do protocolo IP.

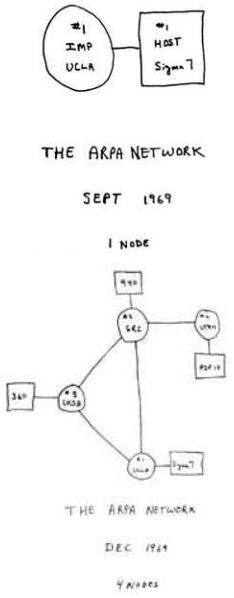
Entenderemos quais os problemas causados pela forma adotada inicialmente para distribuição dos endereços IP e pelo rápido crescimento da Internet, e quais as soluções adotadas para resolver esses problemas. Seguindo este histórico, veremos como algumas dessas soluções evoluíram até se chegar a definição da versão 6 do protocolo IP, o IPv6.

Também veremos neste módulo, através de dados estatísticos, a real necessidade da implantação do IPv6 nas redes de computadores, confrontando dados sobre o ritmo de crescimento da Internet e sobre a adoção e utilização do IPv6. Também serão discutidas as consequências da não implantação do novo protocolo IP e da larga utilização de técnicas tidas como paliativas, como por exemplo as NATs.

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

A Internet e o TCP/IP

- 1969 – Início da ARPANET
- 1981 – Definição do IPv4 na RFC 791
- 1983 – ARPANET adota o TCP/IP
- 1990 – Primeiros estudos sobre o esgotamento dos endereços
- 1993 – Internet passa a ser explorada comercialmente
  - Intensifica-se a discussão sobre o possível esgotamento dos endereços livres e do aumento da tabela de roteamento.



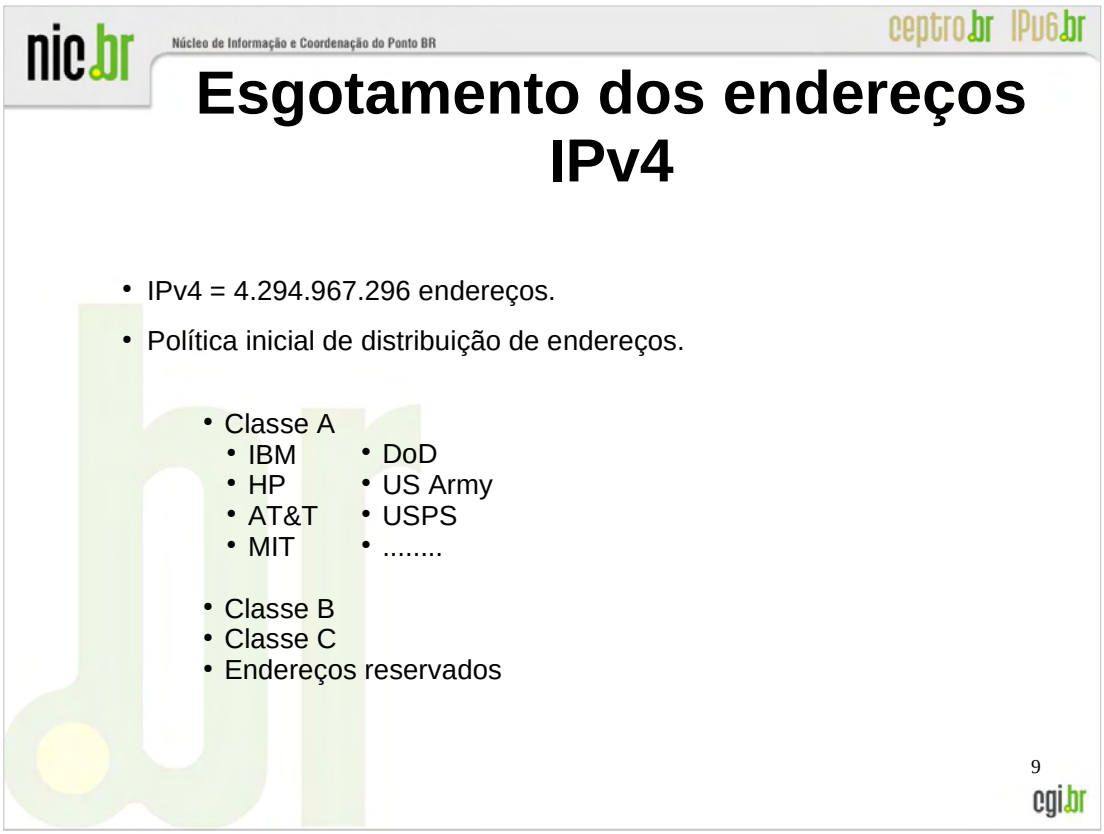
cgi.br

O Departamento de Defesa (DoD - *Department of Defense*) do governo estadunidense iniciou em 1966, através de sua Agencia de Pesquisas e de Projetos Avançados (ARPA - *Advanced Research Projects Agency*), um projeto para a interligação de computadores em centros militares e de pesquisa. Este sistema de comunicação e controle distribuído com fins militares recebeu o nome de ARPANET, tendo como principal objetivo teórico formar uma arquitetura de rede sólida e robusta que pudesse, mesmo com a queda de alguma estação, trabalhar com os computadores e ligações de comunicação restantes. Em 1969, são instalados os primeiros quatro nós da rede, na Universidade de Los Angeles (UCLA), na Universidade da Califórnia em Santa Bárbara (UCSB), no Instituto de Pesquisas de Standford (SRI) e na Universidade de Utah.

No início, a ARPANET trabalhava com diversos protocolos de comunicação, sendo o principal o NCP (*Network Control Protocol*). No entanto, em 1/1/1983, quando a rede já possuía 562 hosts, todas as máquinas da ARPANET passaram a adotar como padrão os protocolos TCP/IP, permitindo o crescimento ordenado da rede e eliminando restrições dos protocolos anteriores.

Definido na RFC 791, o protocolo IP possui duas funções básicas: a fragmentação, que permite o envio de pacotes maiores que o limite de tráfego estabelecido de um enlace, dividindo-os em partes menores; e o endereçamento, que permite identificar o destino e a origem dos pacotes a partir do endereço armazenado no cabeçalho do protocolo. A versão do protocolo IP utilizada na época e atualmente é a versão 4 ou IPv4. Ela mostrou-se muito robusta, e de fácil implantação e interoperabilidade, entretanto, seu projeto original não previu alguns aspectos como:

- O crescimento das redes e um possível esgotamento dos endereços IP;
- O aumento da tabela de roteamento;
- Problemas relacionados a segurança dos dados transmitidos;
- Prioridade na entrega de determinados tipos de pacotes.



As especificações do IPv4 reservam 32 bits para endereçamento, possibilitando gerar mais de 4 bilhões de endereços distintos. Inicialmente, estes endereços foram divididos em três classes de tamanhos fixos da seguinte forma:

- **Classe A**: definia o bit mais significativo como 0, utilizava os 7 bits restantes do primeiro octeto para identificar a rede, e os 24 bits restantes para identificar o *host*. Esses endereços utilizavam a faixa de **1.0.0.0** até **126.0.0.0**;
- **Classe B**: definia os 2 bits mais significativo como 10, utilizava os 14 bits seguintes para identificar a rede, e os 16 bits restantes para identificar o *host*. Esses endereços utilizavam a faixa de **128.1.0.0** até **191.254.0.0**;
- **Classe C**: definia os 3 bits mais significativo como 110, utilizava os 21 bits seguintes para identificar a rede, e os 8 bits restantes para identificar o *host*. Esses endereços utilizavam a faixa de **192.0.1.0** até **223.255.254.0**;

Classe	Formato	Redes	Hosts
A	7 Bits Rede, 24 Bits Host	128	16.777.216
B	14 Bits Rede, 16 Bits Host	16.384	65.536
C	21 Bits Rede, 8 Bits Host	2.097.152	256

Embora o intuito dessa divisão tenha sido tornar a distribuição de endereços mais flexível, abrangendo redes de tamanhos variados, esse tipo de classificação mostrou-se ineficiente. Desta forma, a classe A atenderia um número muito pequeno de redes, mas ocupava metade de todos os endereços disponíveis; para endereçar 300 dispositivos em uma rede, seria necessário obter um bloco de endereços da classe B, desperdiçando assim quase o total dos 65 mil endereços; e os 256 endereços da classe C não supriam as necessidades da grande maioria das redes.

Outro fator que colaborava com o desperdício de endereços, era o fato de que dezenas de faixas classe A foram atribuídas integralmente a grandes instituições como IBM, AT&T, Xerox, HP, Apple, MIT, Ford, Departamento de Defesa Americano, entre muitas outras, disponibilizando para cada uma 16.777.216 milhões de endereços. Além disso, 35 faixas de endereços classe A foram reservadas para usos específicos como *multicast, loopback* e uso futuro.

Em 1990, já existiam 313.000 *hosts* conectados a rede e estudos já apontavam para um colapso devido a falta de endereços. Outros problemas também tornavam-se mais efetivos conforme a Internet evoluía, como o aumento da tabela de roteamento.

Devido ao ritmo de crescimento da Internet e da política de distribuição de endereços, em maio de 1992, 38% das faixas de endereços classe A, 43% da classe B e 2% da classe C, já estavam alocados. Nesta época, a rede já possuía 1.136.000 *hosts* conectados.

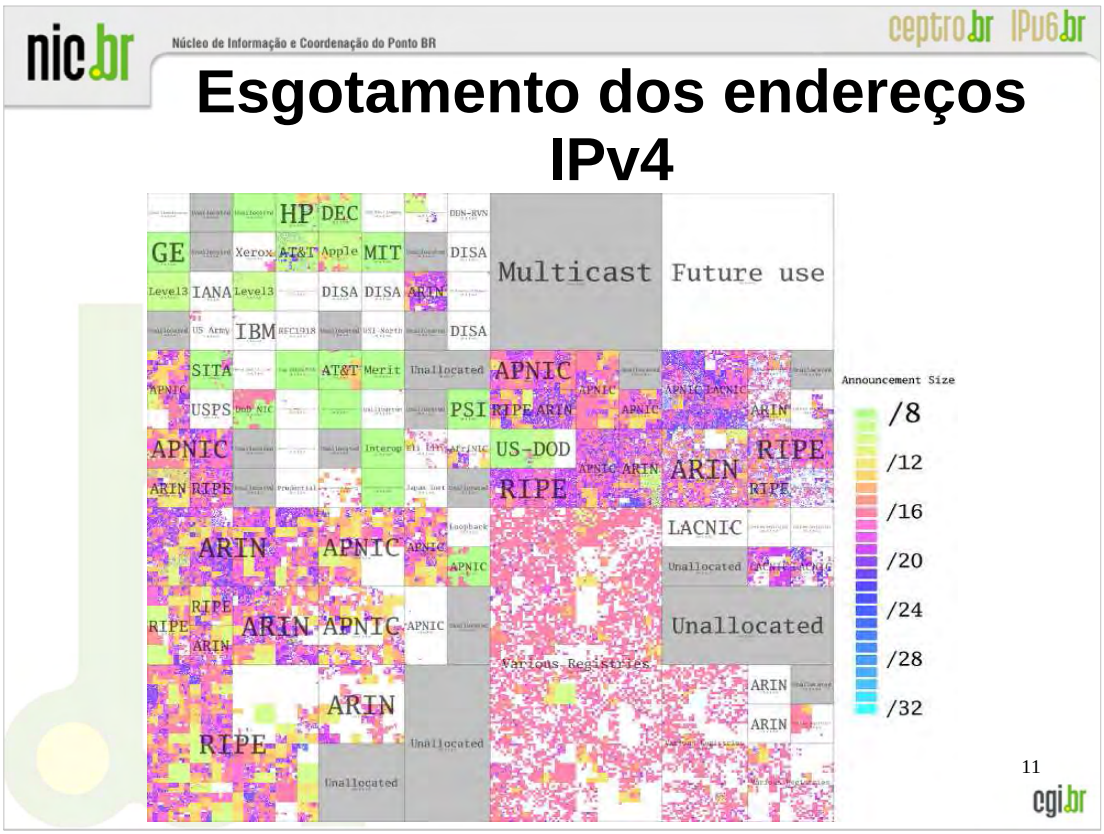
Em 1993, com a criação do protocolo HTTP e a liberação por parte do Governo estadunidense para a utilização comercial da Internet, houve um salto ainda maior na taxa de crescimento da rede, que passou de 2.056.000 de *hosts* em 1993 para mais de 26.000.000 de *hosts* em 1997.

Data	***Hosts***	***Domínios***
1981	213	-
1982	235	-
1983	562	-
1984	1.024	-
1985	1.961	-
1986	5.089	-
1987	28.174	-
1988	56.000	1.280
1989	159.000	4.800
1990	313.000	9.300
1991	617.000	18.000
1992	1.136.000	17.000
1993	2.056.000	26.000
1994	3.212.000	46.000
1995	8.200.000	120.000
1996	16.729.000	488.000
1997	26.053.000	1.301.000

Tabela de crescimento da Internet.

Mais informações:

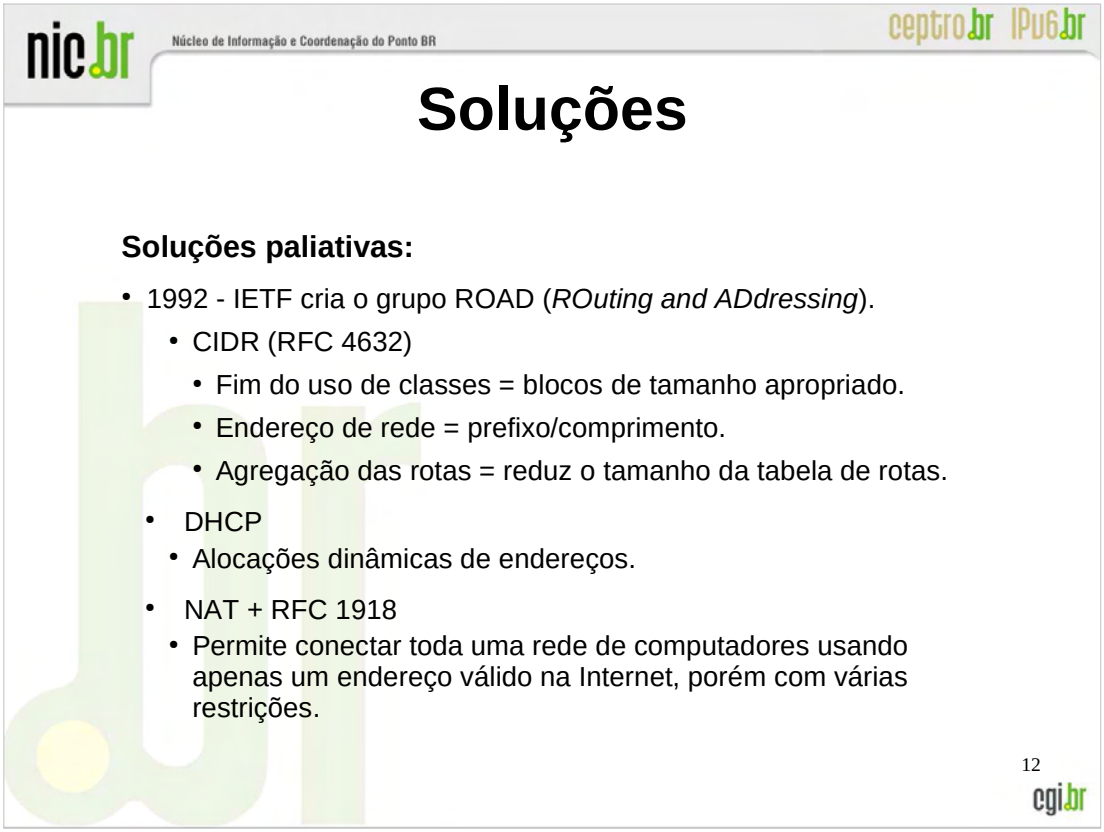
- RFC 1287 - *Towards the Future Internet Architecture*.
- RFC 1296 - *Internet Growth* (1981-1991)
- Solensky F., '*Continued Internet Growth*', *Proceedings of the 18th Internet Engineering Task Force*, Agosto 1990, http://www.ietf.org/proceedings/prior29/IETF18.pdf
- *IANA IPv4 Address Space Registry* - http://www.iana.org/assignments/ipv4-address-space/ipv4-address-space.xml
- RFC 3330 - *Special-Use IPv4 Addresses*.



Esta imagem mostra uma visualização das informações da tabela de roteamento BGP extraída do projeto Routeviews. Nela, o espaço de endereço IPv4 unidimensional é mapeado em uma imagem de bidimensional, onde blocos CIDR sempre aparecem como quadrados ou retângulos.

Mais informações:

• http://maps.measurement-factory.com/

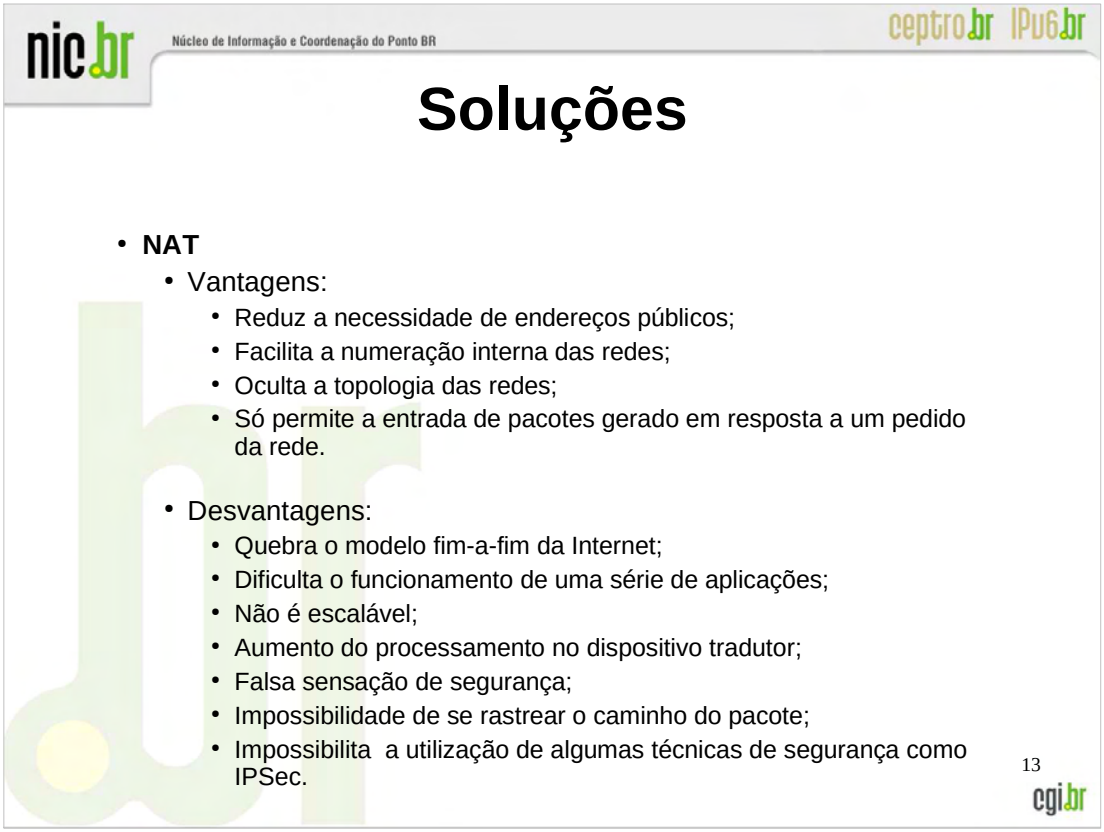


Diante desse cenário, a IETF (*Internet Engineering Task Force*) passa a discutir estratégias para solucionar a questão do esgotamento dos endereços IP e o problema do aumento da tabela de roteamento. Para isso, em novembro de 1991, é formado o grupo de trabalho ROAD (*ROuting and Addressing*), que apresenta como solução a estes problemas a utilização do CIDR (*Classless Inter-domain Routing*).

Definido na RFC 4632 (tornou obsoleta a RFC 1519), o CIDR tem como idéia básica o fim do uso de classes de endereços, permitindo a alocação de blocos de tamanho apropriado a real necessidade de cada rede; e a agregação de rotas, reduzindo o tamanho da tabela de roteamento. Com o CIDR os blocos são referenciados como prefixo de redes. Por exemplo, no endereço **a.b.c.d/x**, os x bits mais significativos indicam o prefixo da rede. Outra forma de indicar o prefixo é através de máscaras, onde a máscara **255.0.0.0** indica um prefixo **/8**, **255.255.0.0** indica um **/16**, e assim sucessivamente.

Outra solução, apresentada na RFC 2131 (tornou obsoleta a RFC 1541), foi o protocolo DHCP (*Dynamic Host Configuration Protocol*). Através do DHCP um *host* é capaz de obter um endereço IP automaticamente e adquirir informações adicionais como máscara de sub-rede, endereço do roteador padrão e o endereço do servidor DNS local.

O DHCP tem sido muito utilizado por parte dos ISPs por permitir a atribuição de endereços IP temporários a seus clientes conectados. Desta forma, torna-se desnecessário obter um endereço para cada cliente, devendo-se apenas designar endereços dinamicamente, através de seu servidor DHCP. Este servidor terá uma lista de endereços IP disponíveis, e toda vez que um novo cliente se conectar à rede, lhe será designado um desses endereço de forma arbitrária, e no momento que o cliente se desconecta, o endereço é devolvido.



A NAT (*Network Address Translation*), foi outra técnica paliativa desenvolvida para resolver o problema do esgotamento dos endereços IPv4. Definida na RFC 3022 (tornou obsoleta a RFC 1631), tem como idéia básica permitir que, com um único endereço IP, ou um pequeno número deles, vários *hosts* possam trafegar na Internet. Dentro de uma rede, cada computador recebe um endereço IP privado único, que é utilizado para o roteamento do tráfego interno. No entanto, quando um pacote precisa ser roteado para fora da rede, uma tradução de endereço é realizada, convertendo endereços IP privados em endereços IP públicos globalmente únicos.

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

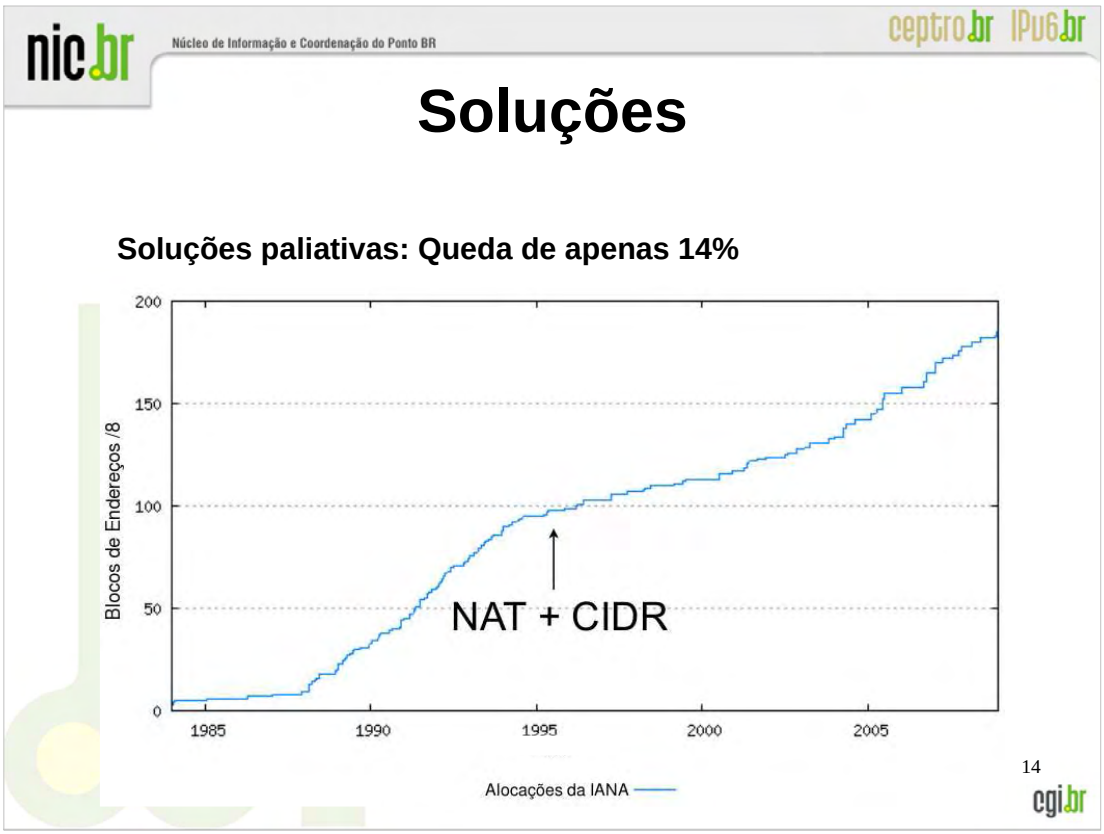
10.0.0.0 a **10.255.255.255 /8** (16.777.216 *hosts*)

172.16.0.0 a **172.31.255.255 /12** (1.048.576 *hosts*)

192.168.0.0 a **192.168.255.255 /16** (65.536 *hosts*)

A utilização da NAT mostrou-se eficiente no que diz respeito a economia de endereços IP, além de apresentar alguns outros aspectos positivos, como facilitar a numeração interna das redes, ocultar a topologia das redes e só permitir a entrada de pacotes gerados em resposta a um pedido da rede. No entanto, o uso da NAT apresenta inconvenientes que não compensam as vantagens oferecidas.

A NAT quebra o modelo fim-a-fim da Internet, não permitindo conexões diretas entre dois *hosts*, o que dificulta o funcionamento de uma série de aplicações, como P2P, VoIP e VPNs. Outro problema é a baixa escalabilidade, pois o número de conexões simultâneas é limitado, além de exigir um grande poder de processamento do dispositivo tradutor. O uso da NAT também impossibilita rastrear o caminho de pacote, através de ferramentas como *traceroute*, por exemplo, e dificulta a utilização de algumas técnicas de segurança como IPSec. Além disso, seu uso passa uma falsa sensação de segurança, pois, apesar de não permitir a entrada de pacotes não autorizados, a NAT não realiza nenhum tipo de filtragem ou verificação nos pacotes que passa por ela.

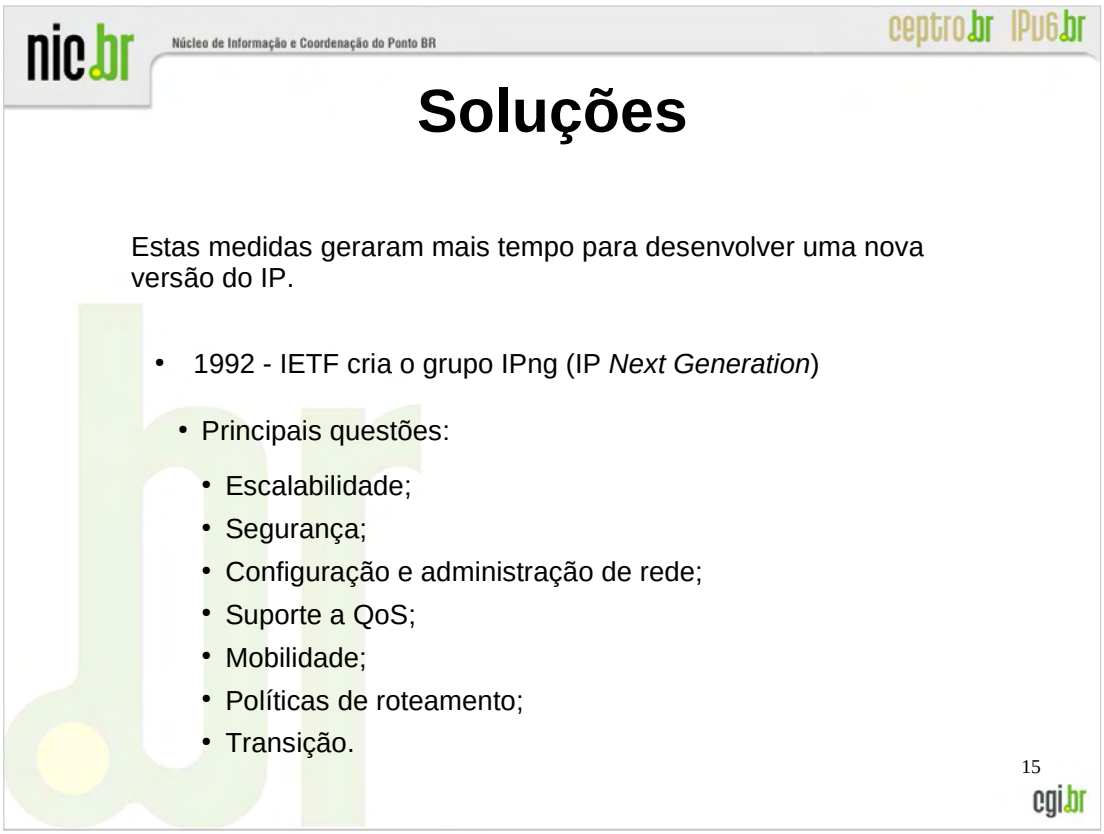


Embora estas soluções tenham diminuído a demanda por IPs, elas não foram suficientes para resolver os problemas decorrentes do crescimento da Internet. A adoção dessas técnicas reduziu em apenas 14% a quantidade de blocos de endereços solicitados à IANA e a curva de crescimento da Internet continuava apresentando um aumento exponencial.

Essas medidas, na verdade, serviram para que houvesse mais tempo para se desenvolver uma nova versão do IP, que fosse baseada nos princípios que fizeram o sucesso do IPv4, porém, que fosse capaz de suprir as falhas apresentadas por ele.

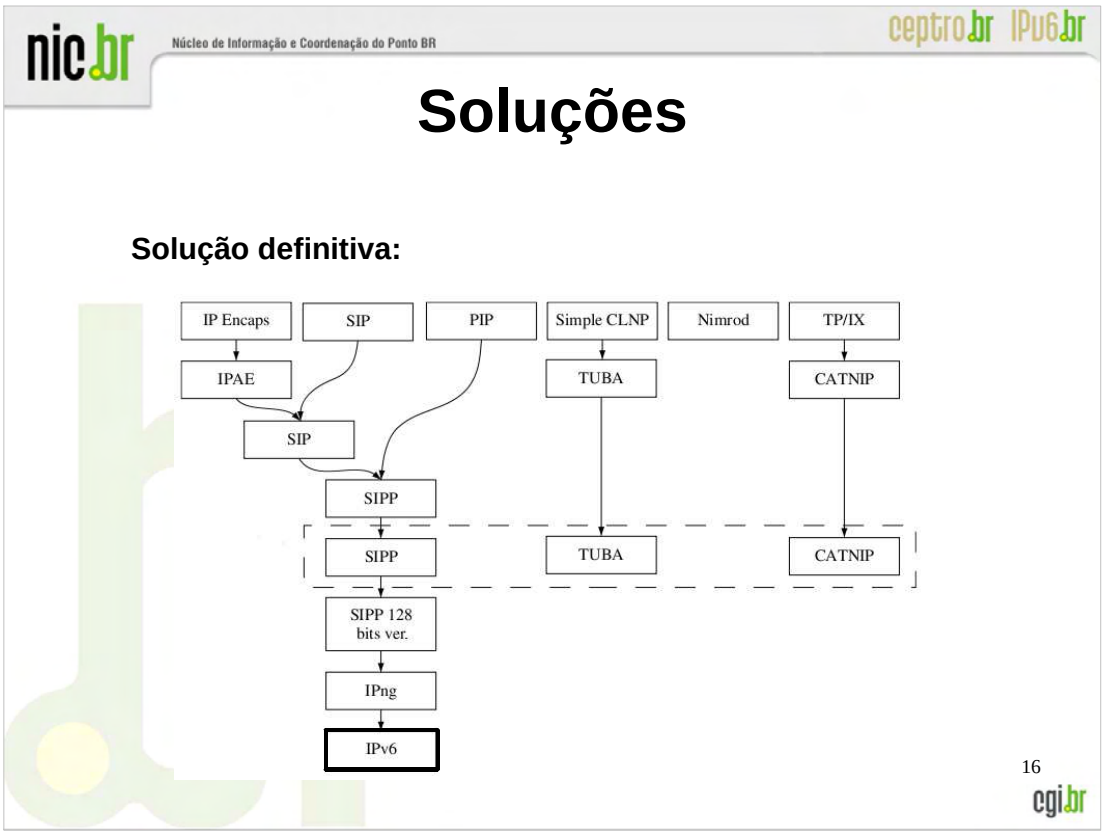
Mais informações:

- RFC 1380 - *IESG Deliberations on Routing and Addressing*
- RFC 1918 - *Address Allocation for Private Internets*
- RFC 2131 - *Dynamic Host Configuration Protocol*
- RFC 2775 - *Internet Transparency*
- RFC 2993 - *Architectural Implications of NAT*
- RFC 3022 - *Traditional IP Network Address Translator (Traditional NAT)*
- RFC 3027 - *Protocol Complications with the IP Network Address Translator*
- RFC 4632 - *Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan.*



Deste modo, em dezembro de 1993 a IETF formalizou, através da RFC 1550, as pesquisas a respeito da nova versão do protocolo IP, solicitando o envio de projetos e propostas para o novo protocolo. Esta foi umas das primeiras ações do grupo de trabalho da IETF denominado *Internet Protocol next generation* (IPng). As principais questões que deveriam ser abordadas na elaboração da próxima versão do protocolo IP foram:

- Escalabilidade;
- Segurança;
- Configuração e administração de rede;
- Suporte a QoS;
- Mobilidade;
- Políticas de roteamento;
- Transição.



Diversos projetos começaram a estudar os efeitos do crescimento da Internet, sendo os principais o CNAT, o IP *Encaps*, o *Nimrod* e o *Simple* CLNP. Destas propostas surgiram o TCP and UDP *with Bigger Addresses* (TUBA), que foi uma evolução do *Simple* CLNP, e o IP *Address Encapsulation* (IPAE), uma evolução do IP *Encaps*. Alguns meses depois foram apresentados os projetos *Paul's Internet Protocol* (PIP), o *Simple Internet Protocol* (SIP) e o TP/IX. Uma nova versão do SIP, que englobava algumas funcionalidades do IPAE, foi apresentada pouco antes de agregar-se ao PIP, resultando no *Simple Internet Protocol Plus* (SIPP). No mesmo período, o TP/IX mudou seu nome para *Common Architecture for the Internet* (CATNIP).

Em janeiro de 1995, na RFC 1752 o IPng apresentou um resumo das avaliações das três principais propostas:

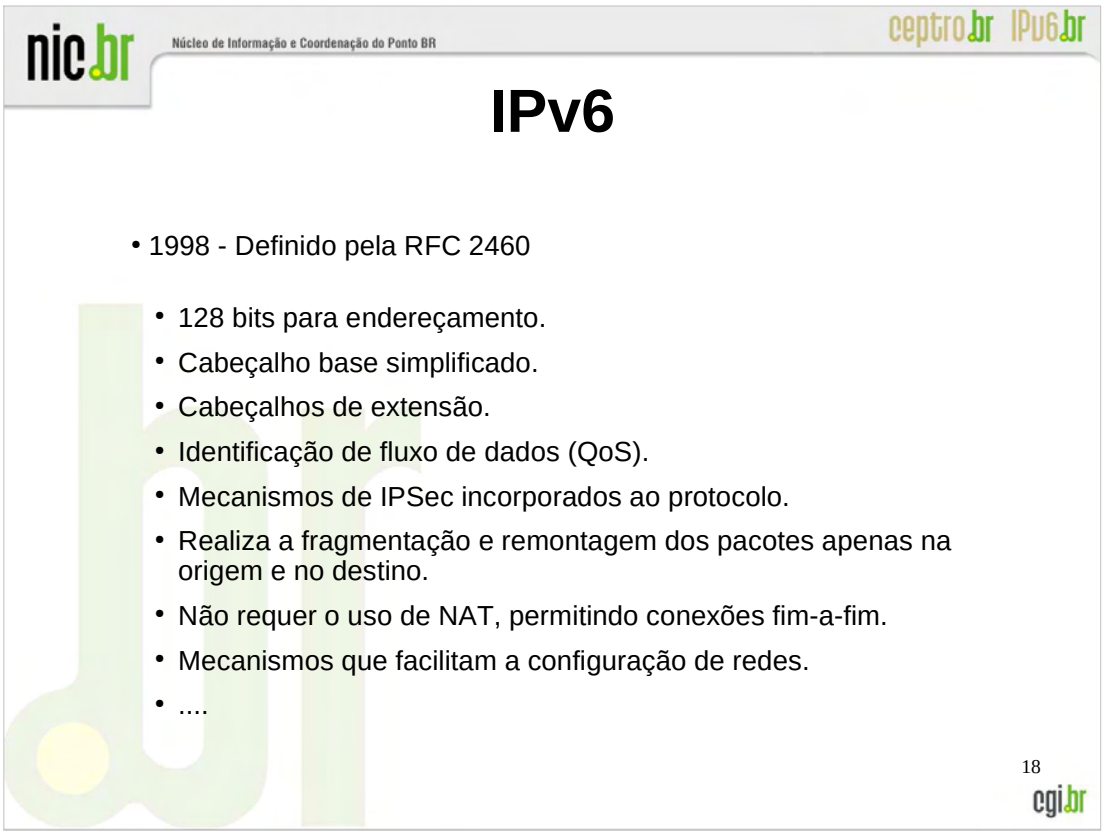
- **CANTIP** - foi concebido como um protocolo de convergência, para permitir a qualquer protocolo da camada de transporte ser executado sobre qualquer protocolo de camada de rede, criando um ambiente comum entre os protocolos da Internet, OSI e Novell;
- **TUBA** - sua proposta era de aumentar o espaço para endereçamento do IPv4 e torná-lo mais hierárquico, buscando evitar a necessidade de se alterar os protocolos da camada de transporte e aplicação. Pretendia uma migração simples e em longo prazo, baseada na atualização dos *host* e servidores DNS, entretanto, sem a necessidade de encapsulamento ou tradução de pacotes, ou mapeamento de endereços;
- **SIPP** - concebido para ser uma etapa evolutiva do IPv4, sem mudanças radicais e mantendo a interoperabilidade com a versão 4 do protocolo IP, fornecia uma plataforma para novas funcionalidades da Internet, aumentava o espaço para endereçamento de 32 bits para 64 bits, apresentava um nível maior de hierarquia e era composto por um mecanismo que permitia "alargar o endereço" chamado *cluster addresses*. Já possuía cabeçalhos de extensão e um campo *flow* para identificar o tipo de fluxo de cada pacote.

Entretanto, conforme relatado também na RFC 1752, todas as três propostas apresentavam problemas significativos. Deste modo, a recomendação final para o novo Protocolo Internet baseou-se em uma versão revisada do SIPP, que passou a incorporar endereços de 128 bits, juntamente com os elementos de transição e autoconfiguração do TUBA, o endereçamento baseado no CIDR e os cabeçalhos de extensão. O CATNIP, por ser considerado muito incompleto, foi descartado.

Após esta definição, a nova versão do Protocolo Internet passou a ser chamado oficialmente de IPv6.

Mais informações:

- RFC 1550 - *IP: Next Generation (IPng) White Paper Solicitation*
- RFC 1752 - *The Recommendation for the IP Next Generation Protocol*



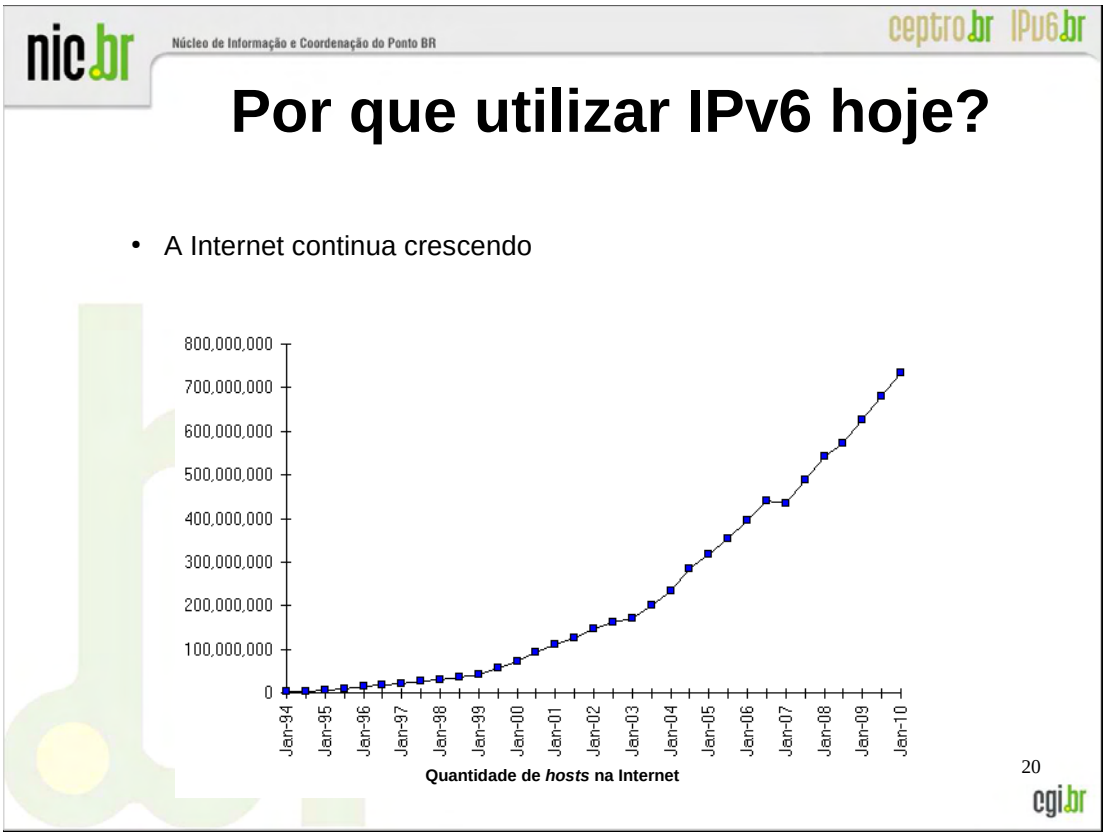
As especificações da IPv6 foram apresentadas inicialmente na RFC 1883 de dezembro de 1995, no entanto, em em dezembro de 1998, está RFC foi substituída pela RFC 2460. Como principais mudanças em relação ao IPv4 destacam-se:

- **Maior capacidade para endereçamento**: no IPv6 o espaço para endereçamento aumentou de 32 bits para 128 bits, permitindo: níveis mais específicos de agregação de endereços; identificar uma quantidade muito maior de dispositivos na rede; e implementar mecanismos de autoconfiguração. A escalabilidade do roteamento *multicast* também foi melhorada através da adição do campo "escopo" no endereço *multicast*. E um novo tipo de endereço, o *anycast*, foi definido;
- **Simplificação do formato do cabeçalho**: alguns campos do cabeçalho IPv4 foram removidos ou tornaram-se opcionais, com o intuito de reduzir o custo do processamento dos pacotes nos roteadores;
- **Suporte a cabeçalhos de extensão**: as opções não fazem mais parte do cabeçalho base, permitindo um roteamento mais eficaz, limites menos rigorosos em relação ao tamanho e a quantidade de opções, e uma maior flexibilidade para a introdução de novas opções no futuro;
- **Capacidade de identificar fluxos de dados**: foi adicionado um novo recurso que permite identificar de pacotes que pertençam a determinados tráfegos de fluxos, para os quais podem ser requeridos tratamentos especiais;
- **Suporte a autenticação e privacidade**: foram especificados cabeçalhos de extensão capazes de fornecer mecanismos de autenticação e garantir a integridade e a confidencialidade dos dados transmitidos.

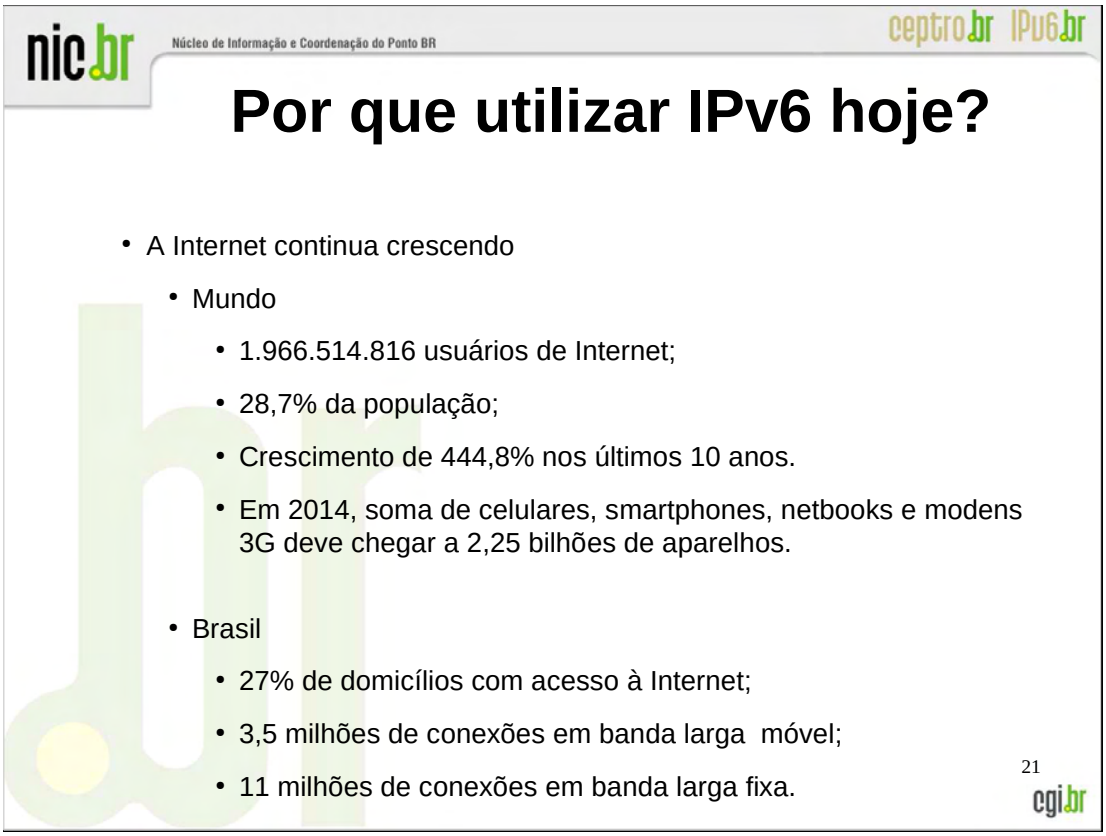
Além disso, o IPv6 também apresentou mudanças no tratamento da fragmentação dos pacotes, que passou a ser realizada apenas na origem; permite o uso de conexões fim-a-fim, princípio que havia sido quebrado com o IPv4 devido a grande utilização de NAT; trouxe recursos que facilitam a configuração de redes, além de outros aspectos que foram melhorados em relação ao IPv4.

Mais informações:

- RFC 2460 - *Internet Protocol, Version 6 (IPv6) Specification*



Durante os últimos dez anos, durante o desenvolvimento do IPv6, a Internet continuou apresentando um ritmo de crescimento cada vez mais acelerado. O número de *hosts* conectados à Internet saltou de 30.000.000 para aproximadamente 732.000.000 nos dias de hoje, com um número cada vez maior de usuários e dispositivos conectados à Rede.



Esta expansão da Internet pode ser medida por diversos fatores, e inúmeras pesquisas têm mostrado que este crescimento não ocorre de forma isolada.

Estima-se que existam no mundo 1.966.514.816 usuários de Internet, ou 28,7% da população da Terra, o que representa, considerando os últimos nove anos, um crescimento de 444,8%. Mantendo este ritmo de crescimento, dentro de dois anos serão dois bilhões de usuários, superando a previsão de que este número só seria alcançado em 2015. A tabela a seguir detalha esses números, apresentando a penetração e o crescimento da Internet em cada região do mundo.

Segundo dados da ABI Research a quantidade de equipamentos móveis capazes de acessarem a Internet, como celulares, smartphones, netbooks e modens 3G, deve chegar a 2,25 bilhões de aparelhos.

Regiões	***População (em 2010)***	***Usuários de Internet (em 2000)***	***Usuários de Internet (atualmente)***	***% por Região***	***Crescimento 2000-2010***
África	1.013.779.050	4.514.400	110.931.700	10,9 %	2.357,3 %
Ásia	3.834.792.852	114.304.000	825.094.396	21,5 %	621,8 %
Europa	813.319.511	105.096.093	475.069.448	58,4 %	352,0 %
Oriente Médio	212.336.924	3.284.800	63.240.946	29,8 %	1.825,3 %
América Norte	344.124.450	108.096.800	266.224.500	77,4 %	146,3 %
America Latina /Caribe	592.556.972	18.068.919	204.689.836	34,5 %	1.032,8 %
Oceania	34.700.201	7.620.480	21.263.990	61,3 %	179,0 %
TOTAL	**6.845.609.960**	**360.985.492**	**1.966.514.816**	**28,7 %**	**444,8 %**

Dados de 30/06/2010.

Seguindo esta tendência, no Brasil o percentual de domicílios com acesso à Internet, via computadores domésticos, aumentou de 12,93% em 2005, para 27% em 2009. O Brasil também alcançou, em Junho de 2009, a marca de 3,5 milhões de conexões em banda larga móvel, crescendo 77% em um ano. Já o número de conexões através de banda larga fixa alcançam o total de 11 milhões.

Como conseqüência deste crescimento, a demanda por endereços IP também cresce consideravelmente. Em 2010, 10 blocos /8 já foram atribuídos pela IANA aos RIRs, restando, no momento, 16 blocos não-alocados dos 256 /8 possíveis, ou seja, 6% do total. Este índice, reforça a projeção para 2011 como data para o esgotamento de seus blocos de endereços IPv4, principalmente porque o número de solicitações de blocos de endereços aumenta a cada ano.

Em setembro de 2008, os RIRs chegaram a um acordo quanto à política que será adotada pela IANA quando sua reserva de endereços atingir o limite de 5 blocos /8. Estes últimos /8 serão imediatamente atribuídos a cada RIR, que os atribuirão entre os ISPs e Registros Nacionais. Caso o estoque da IANA ocorra em 2011 como previsto, o número de endereços IPv4 disponíveis só se esgotará quando as reservas regionais terminarem, o que poderá ocorrer dois ou três anos após o fim das reservas da IANA.

Embora a Internet seja conhecida por ser uma rede mundial livre de uma coordenação central, existem órgãos responsáveis por administrar os principais aspectos técnicos necessários ao seu funcionamento. As atribuições desses órgãos são distribuídas em nível mundial, respeitando uma estrutura hierárquica:

A *Internet Assigned Numbers Authority* (**IANA**) é a responsável pela coordenação e alocação global do espaço de endereços IP e dos ASNs. Desde 1998, a IANA passou a operar através da *Internet Corporation for Assigned Names and Numbers* (**ICANN**), uma organização internacional dirigida por um conselho de diretores, provenientes de diversos países, que supervisiona a ICANN e as políticas elaboradas por ela, trabalhando em parceria com governos, empresas e organizações criadas por tratados internacionais, envolvidos na construção e manutenção da Internet.

A responsabilidade pela distribuição local dos endereços IP é dos Registros de Internet (*Internet Registry* - **IR**), que são classificados de acordo com sua função principal e alcance territorial dentro da estrutura hierárquica da Internet.

Os Registros Regionais de Internet (*Regional Internet Registries* - **RIR**) são órgãos, reconhecidos pela IANA, que atuam e representam grandes regiões geográficas. A função primordial de um RIR é gerenciar e distribuir endereços IP públicos dentro de suas respectivas regiões. Existem cinco RIRs: o *African Network Information Centre* (**AfriNIC**), que atua na região da África; o *Asia-Pacific Network Information Centre* (**APINIC**), atuando na Ásia e na região do Pacífico; o *American Registry for Internet Numbers* (**ARIN**), responsável pela América do Norte; o *Latin American and Caribbean Internet Addresses Registry* (**LACNIC**), que atua na América Latina e em algumas ilhas do Caribe; e o *Réseaux IP Européens Network Coordination Centre* (**RIPE NCC**), que serve a Europa e países da Ásia Central.

Um Registro Nacional de Internet (*National Internet Registry* - **NIR**) é o responsável pela alocação dos endereços IP em nível nacional, distribuindo-os aos Registros Locais de Internet (*Local Internet Registry* - **LIR**). Os LIRs são geralmente ISPs, que por sua vez podem oferecer estes endereços aos usuários finais ou outros ISPs.

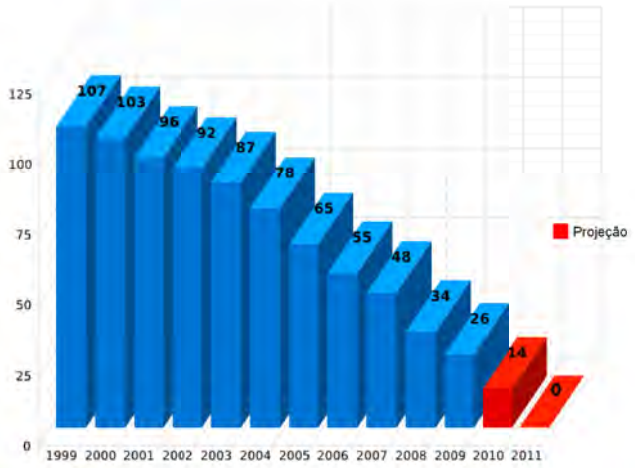
No Brasil, o NIR responsável pela distribuição de endereços IP e do registro de nomes de domínios usando ".br" é o Núcleo de Informação e Coordenação do ponto br (**NIC.br**).

nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

Por que utilizar IPv6 hoje?

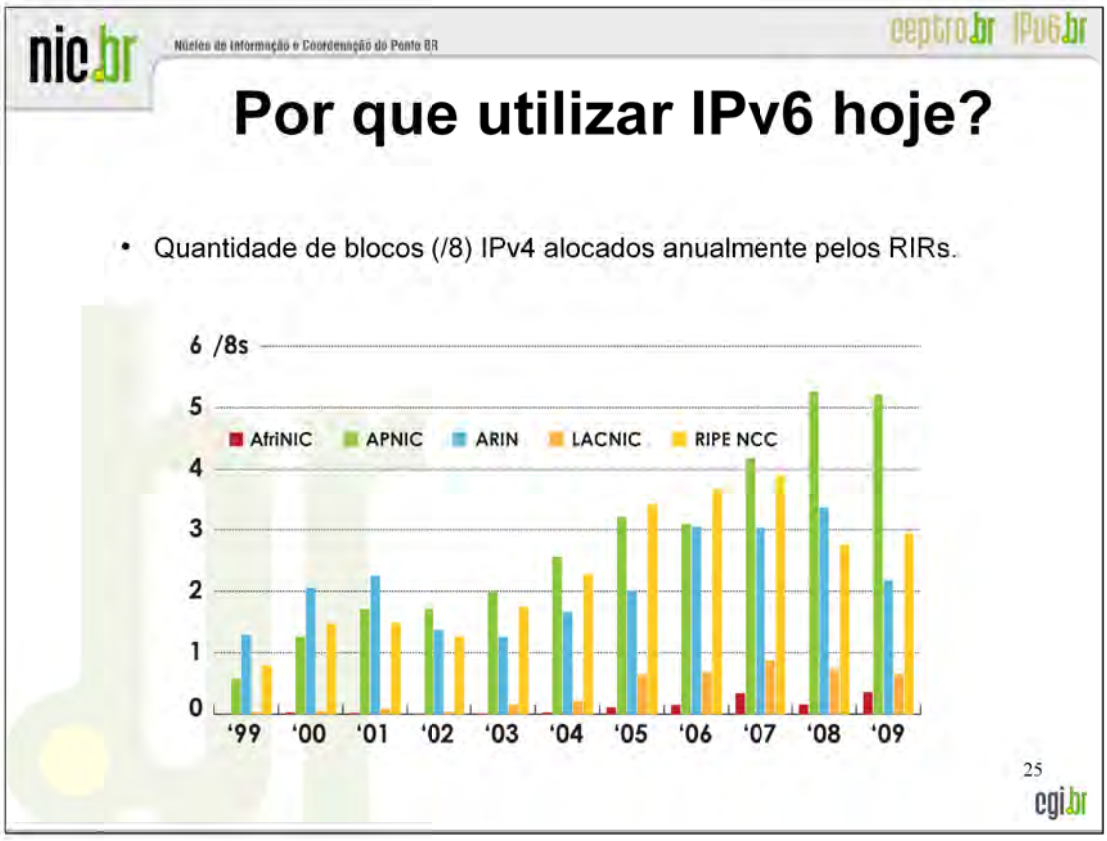
- Evolução do estoque de blocos IP na IANA.



24

cgi.br

Este gráfico apresenta a evolução do estoque de blocos IP na IANA ao longo dos últimos anos, além de uma projeção dos próximos dois anos.



Este gráfico apresenta a quantidade de blocos /8 IPv4 alocados anualmente pelos RIRs.

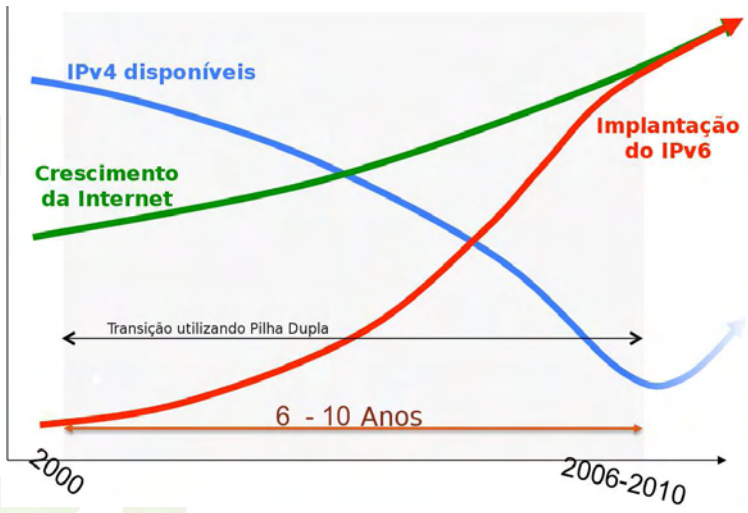
Mais informações:

- http://www.internetworldstats.com/stats.htm
- http://cetic.br/usuarios/tic/2009/rel-geral-04.htm
- http://www.cisco.com/web/BR/barometro/barometro.html
- https://www.isc.org/solutions/survey
- http://www.nro.net/statistics
- http://www.abiresearch.com

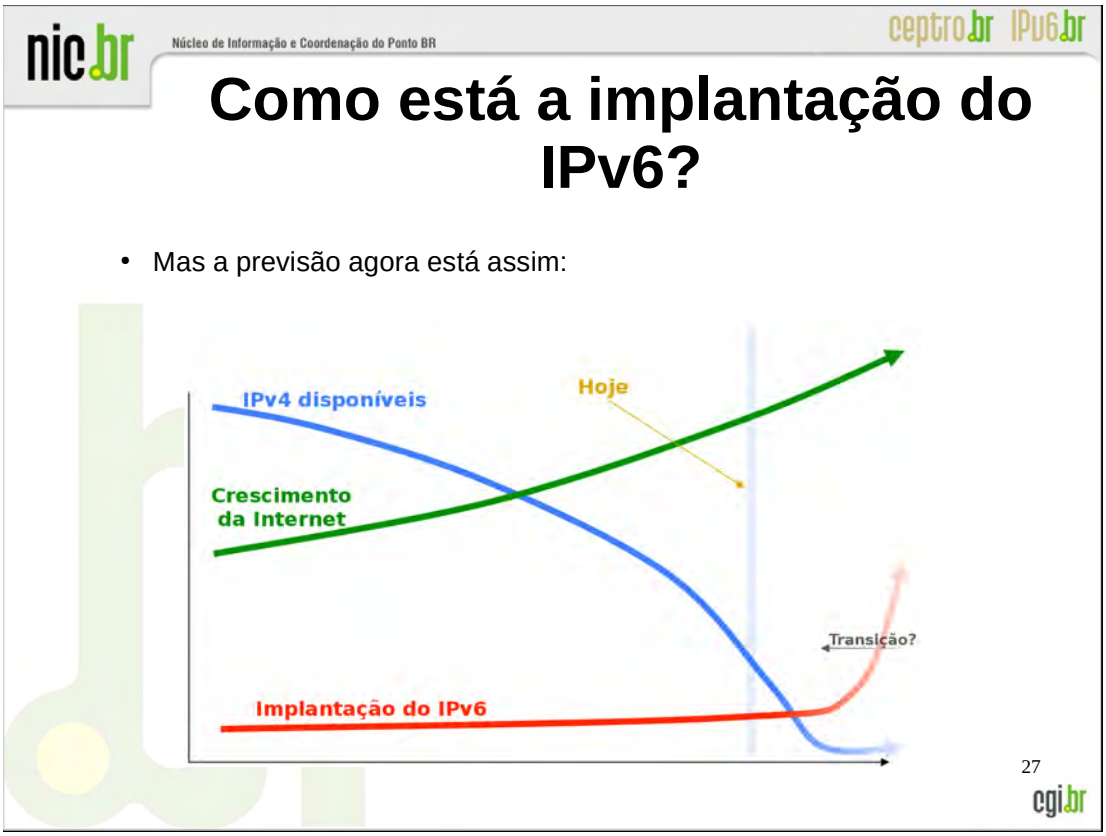
nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Como está a implantação do IPv6?

- A previsão inicial era que fosse assim:



Embora todos os números confirmem a necessidade mais endereços IP, questão essa que é prontamente resolvida com a adoção do IPv6, o ritmo de implantação da nova versão do protocolo IP não está ocorrendo da forma que foi prevista no início de seu desenvolvimento, que apontava o IPv6 como protocolo padrão da Internet aproximadamente dez anos após sua definição, ou seja, se isso realmente tivesse ocorrido, o objetivo desse curso provavelmente seria outro.



Ainda há muitos debates em torno da implantação do IPv6 e alguns fatores têm atrasado a implantação do novo protocolo.

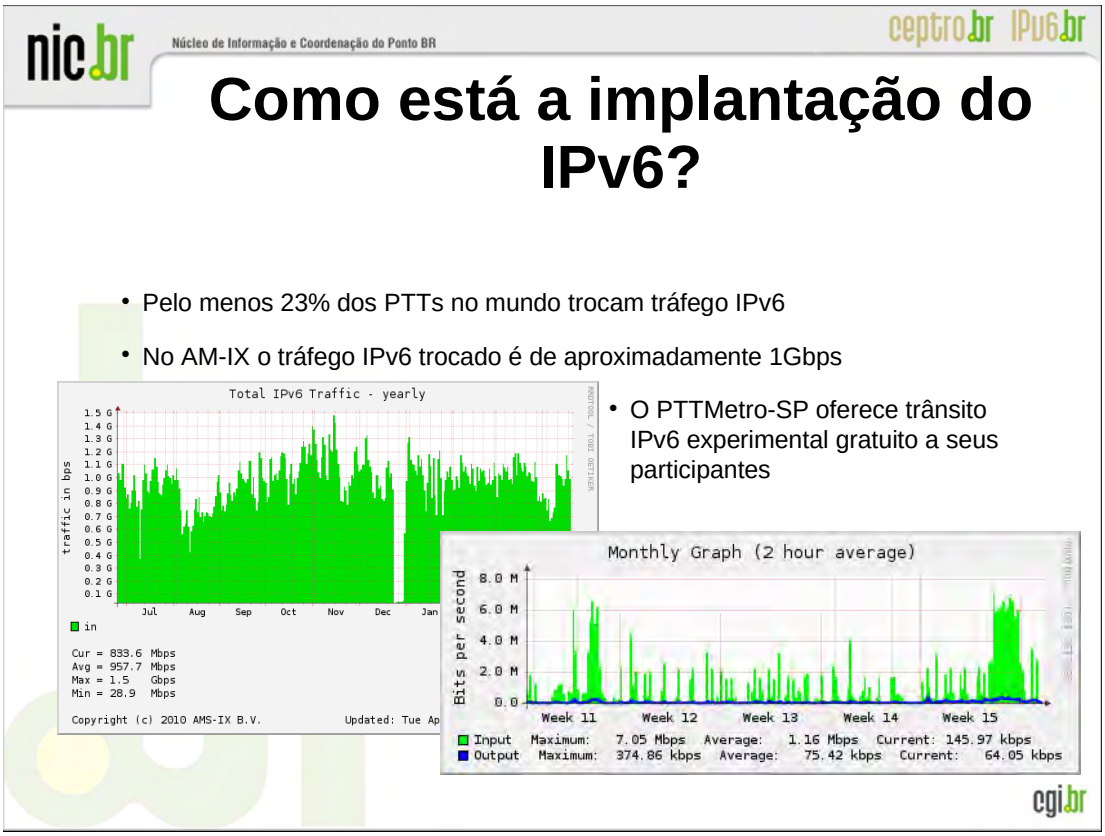
Técnicas como NAT e o DHCP, apesar de dar-nos tempo para desenvolver o IPv6, colaboraram para a demora em sua adoção. Aliado a isso, há o fato de o IPv4 não apresentar graves problemas de funcionamento.

Também é preciso destacar que até o momento, a utilização do IPv6 está ligada principalmente a área acadêmica, e para que a Internet passe a utilizar IPv6 em grande escala, é necessário que a infraestrutura dos principais ISPs seja capaz de transmitir tráfego IPv6 de forma nativa. No entanto, sua implantação em redes maiores tem encontrado dificuldades devido, entre outras coisas, ao receio de grandes mudanças na forma de se gerenciá-las, na existência de gastos devido a necessidade de troca de equipamentos como roteadores e *switches*, e gastos com o aprendizado e treinamento para a área técnica.

Diversos estudos estão sendo realizados buscando mensurar a quantidade de informação que trafega na Internet sobre o protocolo IPv6. Análises sobre o número de ASs anunciando IPv6, análise de consultas a servidores DNS e a quantidade de páginas na Internet usando IPv6, são alguns exemplos de como tem se tentado medir a evolução da implantação da versão 6 do Protocolo Internet.

O Google tem realizado uma avaliação do estado atual do uso de IPv6 por usuários comuns, coletando informações fornecidas pelos navegadores de uma parcela de usuários de seus serviços. Com isso, foi possível determinar que aproximadamente 0,2% de seus clientes IPv6 ativado, e que a quantidade de acessos utilizando IPv6 subiu de 0,189% em agosto de 2008, para 0,261% em janeiro de 2009.

Outros dados interessantes apontam que apenas 6,9% dos ASs trabalham sobre IPv6. Destaca-se também que 9 dos 13 *root DNS servers* são acessíveis via IPv6 (A, B, F, H, I, J, K, L e M).



Ponto fundamental da infraestrutura da Internet, 23% dos PTTs (Pontos de Troca de Tráfego, ou em inglês, IXP - Internet eXchange Point) no mundo trocam tráfego IPv6, e em um dos maiores IXPs, o AM-IX (Amsterdam Internet Exchange), o tráfego IPv6 trocado é de aproximadamente 1Gbps, o que corresponde a 0,3% do tráfego total.

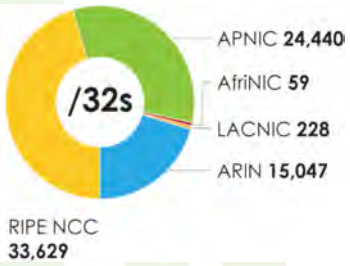
No Brasil, desde fevereiro de 2010 o NIC.br oferece para os participantes do PTTMetro São Paulo o serviço de trânsito IPv6 experimental gratuitamente. Com a iniciativa, o NIC.br tem o objetivo de promover o uso do protocolo, reduzindo o tempo entre a atribuição dos blocos para as entidades e seu efetivo uso, permitindo experimentação e facilitando sua implantação.

nic.br Núcleo de Informação e Coordenação do Ponto BR

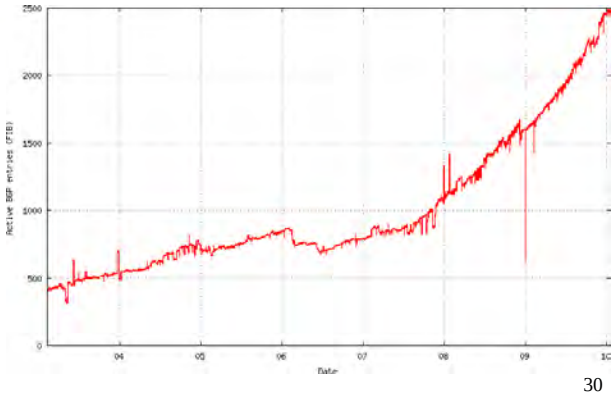
ceptro.br IPv6.br

Como está a implantação do IPv6?

- Dos ~73.000 blocos /32 já alocados pelos RIR, apenas 3% são efetivamente utilizados.



Alocações feitas pelos RIRs



Entradas IPv6 na tabela de rotas global

Dados de 15/01/2010

30

cgi.br

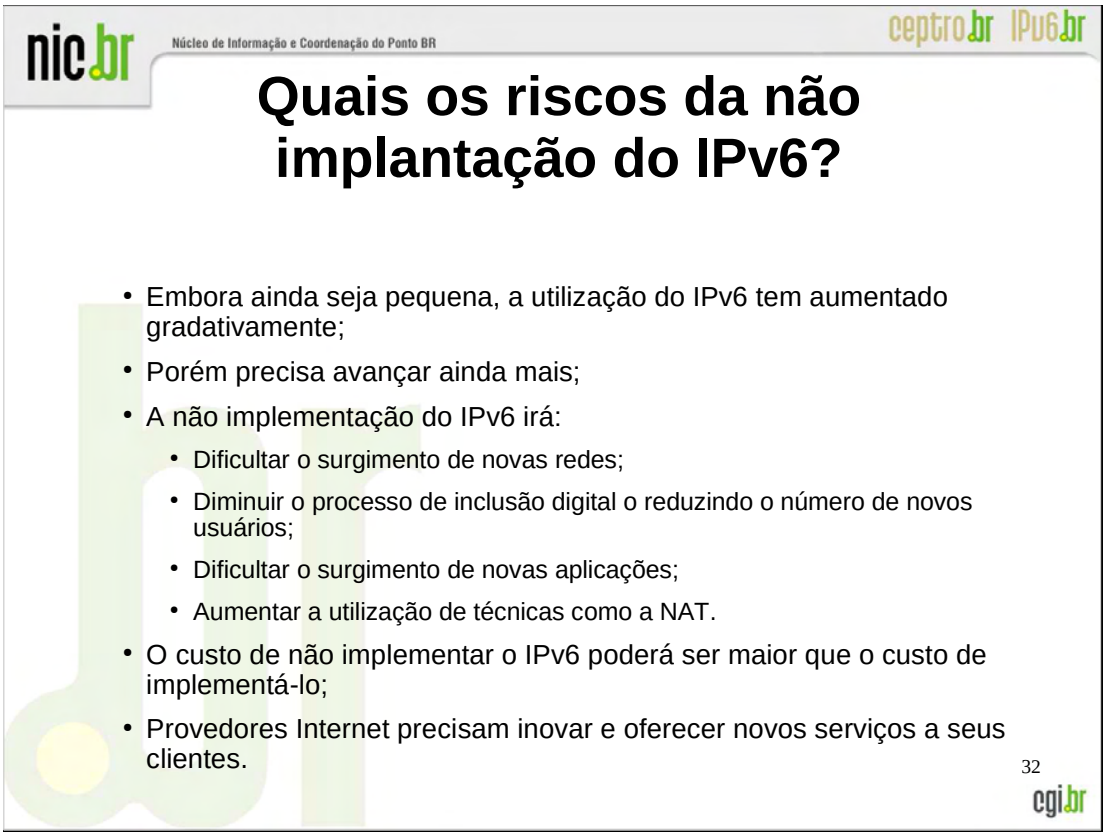
Blocos de endereços IPv6 vêm sendo alocados pelos RIRs há aproximadamente dez anos. No entanto, o fato dos RIRs alocarem endereços aos Registros Nacionais ou aos ISPs, não significa que estes endereços estejam sendo utilizados. Ao cruzar os dados sobre a quantidade de blocos /32 IPv6 já alocados com o número de rotas anunciadas na tabela de roteamento, nota-se que apenas 3% desses recursos estão sendo efetivamente utilizados, isto é, dos 76.000 blocos já alocados, apenas pouco mais 2.500 estão presentes na tabela global de roteamento.

No Brasil, e em toda a América Latina, a situação é similar.

O LACNIC, RIR que atua na América Latina e Caribe, já alocou aproximadamente 230 blocos /32 IPv6, o que corresponde a aproximadamente 0,4% do total de blocos já alocados mundialmente. Destes, 35,3% estão alocados no Brasil, porém, apenas 30% estão sendo efetivamente utilizado.

Mais informações:

- http://www.arbornetworks.com/IPv6research
- **https://sites.google.com/site/ipv6implementors/conference2009/agenda/10_Lees_Google_IPv6_User_Measurement.pdf**
- http://www.oecd.org/dataoecd/48/51/44953210.pdf
- http://www.ipv6.br/IPV6/MenuIPv6Transito
- http://www.ams-ix.net/sflow-stats/ipv6/
- http://bgp.he.net/ipv6-progress-report.cgi
- http://portalipv6.lacnic.net/pt-br/ipv6/estat-sticas
- http://bgp.potaroo.net/v6/as2.0/index.html
- ftp://ftp.registro.br/pub/stats/delegated-ipv6-nicbr-latest



É importante observar que, embora a utilização do IPv6 ainda não tenha tanta representatividade, todos os dados apresentados mostram que sua penetração nas redes tem aumentado gradativamente. No entanto, é preciso avançar ainda mais. Adiar por mais tempo a implantação do IPv6 pode trazer diversos prejuízos para o desenvolvimento de toda a Internet.

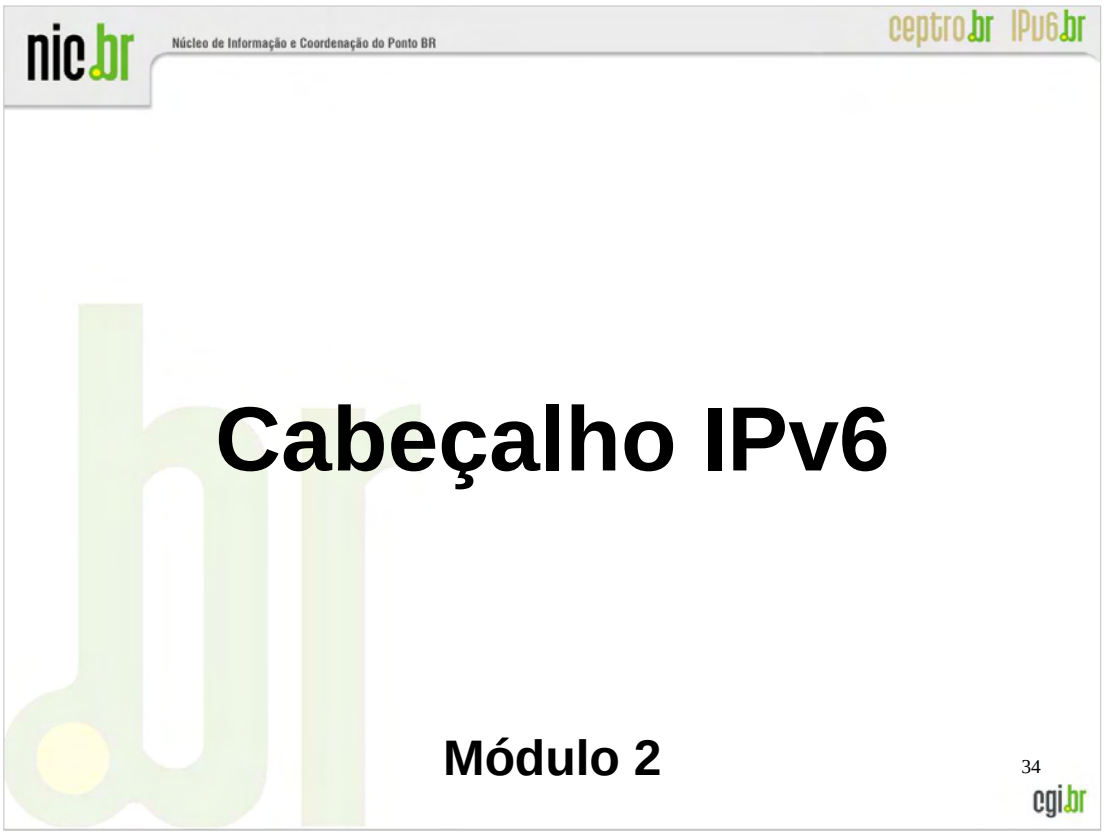
Como vimos, existe hoje uma demanda muito grande por mais endereços IP, e mesmo que a Internet continue funcionando sem novos endereços, ela terá muita dificuldade para crescer. A cada dia surgem novas redes, graças a expansão das empresas e ao surgimento de novos negócios; iniciativas de inclusão digital tem trazido muitos novos usuários para a Internet; e o crescimento das redes 3G, e a utilização da Internet em dispositivos eletrônicos e eletrodomésticos são exemplos de novas aplicações que colaboram com seu crescimento.

A não implantação do IPv6 provavelmente impedira o desenvolvimento de todas essas áreas, e além disso, com o IPv6 elimina-se a necessidade da utilização de NATs, favorecendo o funcionamento de várias aplicações. Deste modo, o custo de não se utilizar, ou adiar ainda mais a implantação do protocolo IPv6, será muito maior do que o de utilizá-lo.

Para os Provedores Internet, é importante que estes ofereçam novos serviços a seus clientes, e principalmente, porque inovar é a chave para competir e manter-se à frente da concorrência.

IPv6.br

A Nova Geração do Protocolo Internet



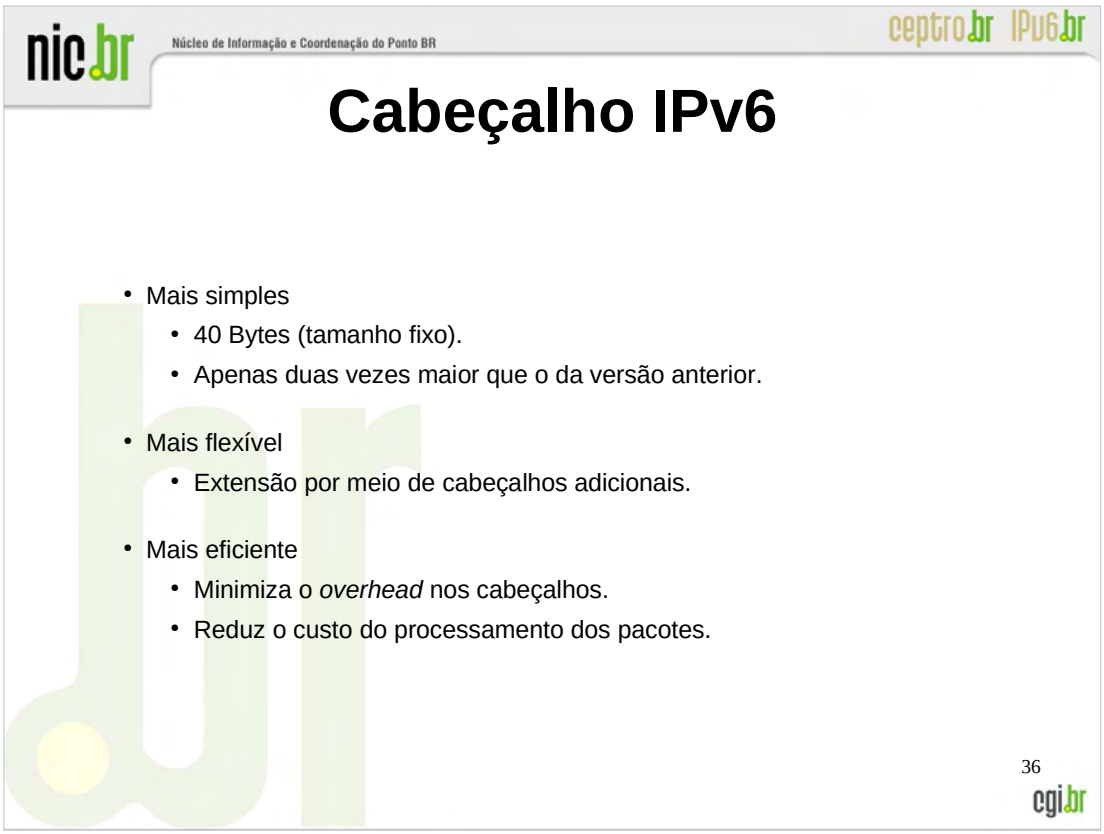
A partir desse ponto, iniciaremos o estudo das principais características do IPv6, começando pela análise das mudanças ocorridas na estrutura de seu cabeçalho, apresentando as diferenças entre os cabeçalhos IPv4 e IPv6, e de que forma essas mudanças aprimoraram o funcionamento do protocolo. Também será detalhado o funcionamento dos cabeçalhos de extensão, mostrando porque sua utilização pode melhorar o desempenho dos roteadores.

Cabeçalho IPv4

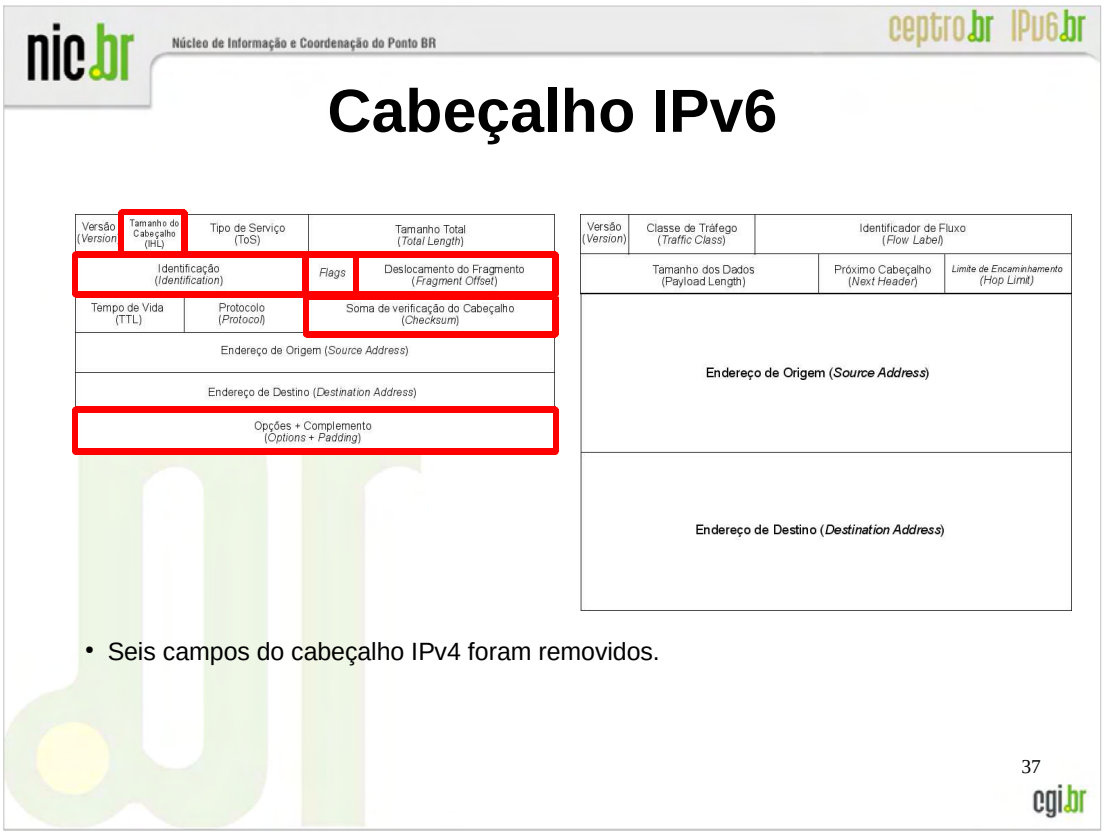
O cabeçalho IPv4 é composto por 12 campos fixos, podendo conter ou não opções, fazendo com que seu tamanho possa variar entre 20 e 60 Bytes.

O cabeçalho IPv4 é composto por 12 campos fixos, podendo conter ou não opções, fazendo com que seu tamanho possa variar entre 20 e 60 Bytes. Estes campos são destinados transmitir informações sobre:

- a versão do protocolo;
- o tamanho do cabeçalho e dos dados;
- a fragmentação;
- o tipo de dados;
- o tempo de vida do pacote;
- o protocolo da camada seguinte (TCP, UDP, ICMP);
- a integridade dos dados;
- a origem e o destino do pacote.



Algumas mudanças foram realizadas no formato do cabeçalho base do IPv6 de modo a torná-lo mais simples, com apenas oito campos e com tamanho fixo de 40 Bytes, além de mais flexível e eficiente, prevendo sua extensão por meio de cabeçalhos adicionais que não precisam ser processados por todos os roteadores intermediários. Estas alterações permitiram que, mesmo com um espaço para endereçamento de 128 bits, quatro vezes maior que os 32 bits do IPv4, o tamanho total do cabeçalho IPv6 seja apenas duas vezes maior que o da versão anterior.



Entre essas mudanças, destaca-se a remoção de seis campos do cabeçalho IPv4, visto que suas funções não são mais necessárias ou são implementadas pelos cabeçalhos de extensão.

No IPv6, as opções adicionais agora fazem parte dos cabeçalhos de extensão do IPv6. Deste modo, os campos Opções e Complementos puderam ser removidos.

O campo Tamanho do Cabeçalho também foi removido, porque o tamanho do cabeçalho IPv6 é fixo.

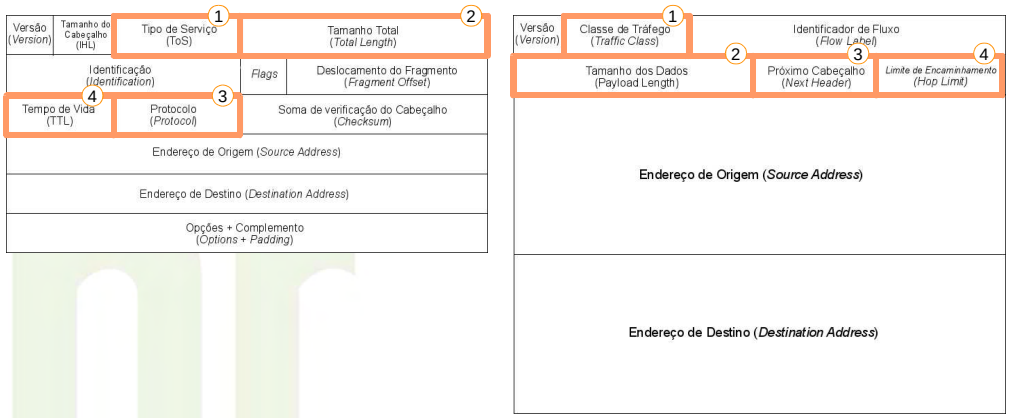
Os campos Identificação, *Flags* e Deslocamento do Fragmento, foram removidos porque as informações referentes a fragmentação são indicadas agora em um cabeçalho de extensão apropriado.

Com o intuito de aumentar a velocidade do processamento dos roteadores, o campo Soma de Verificação foi retirado, pois esse cálculo já é realizado pelos protocolos das camadas superiores.

nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

Cabeçalho IPv6



- Seis campos do cabeçalho IPv4 foram removidos.
- Quatro campos tiveram seus nomes alterados e seus posicionamentos modificados.

cgi.br

Outra mudança refere-se a alteração do nome e do posicionamento de outros quatro campos.

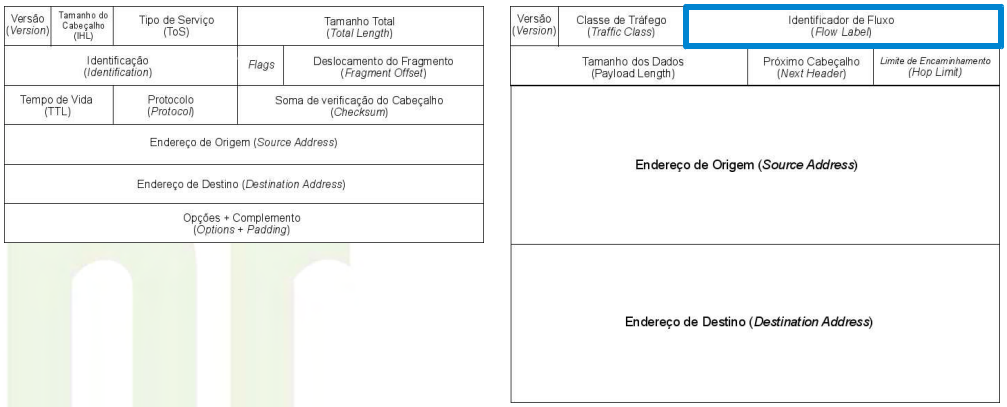
IPv4		IPv6
Tipo de Serviço	⟶	Classe de Tráfego
Tamanho Total	⟶	Tamanho dos Dados
Tempo de Vida (TTL)	⟶	Limite de Encaminhamento
Protocolo	⟶	Próximo Cabeçalho

Esses reposicionamentos foram definidos para facilitar o processamento dessas informações pelos roteadores.

nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

Cabeçalho IPv6



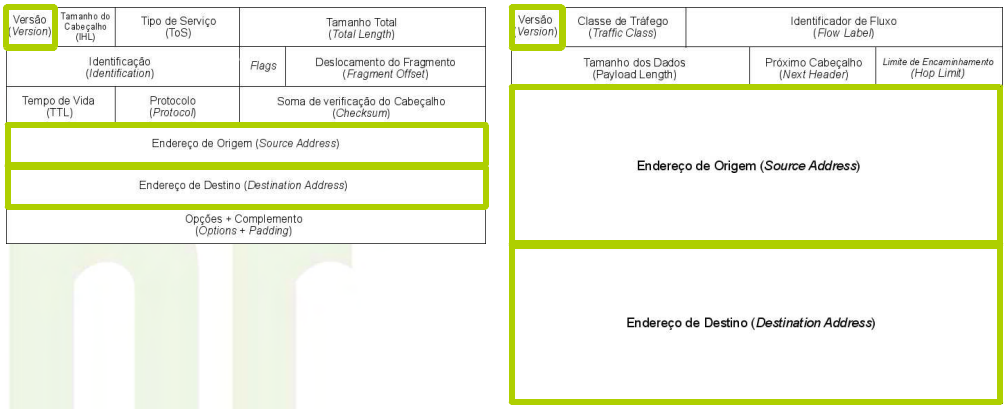
- Seis campos do cabeçalho IPv4 foram removidos.
- Quatro campos tiveram seus nomes alterados e seus posicionamentos modificados.
- O campo Identificador de Fluxo foi acrescentado.

Também foi adicionado um novo campo, o Identificador de Fluxo, acrescentado um mecanismo extra de suporte a QoS ao protocolo IP. Mais detalhes sobre este campo e de como o protocolo IPv6 trata a questão do QoS serão apresentados nos próximos módulos deste curso.

nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

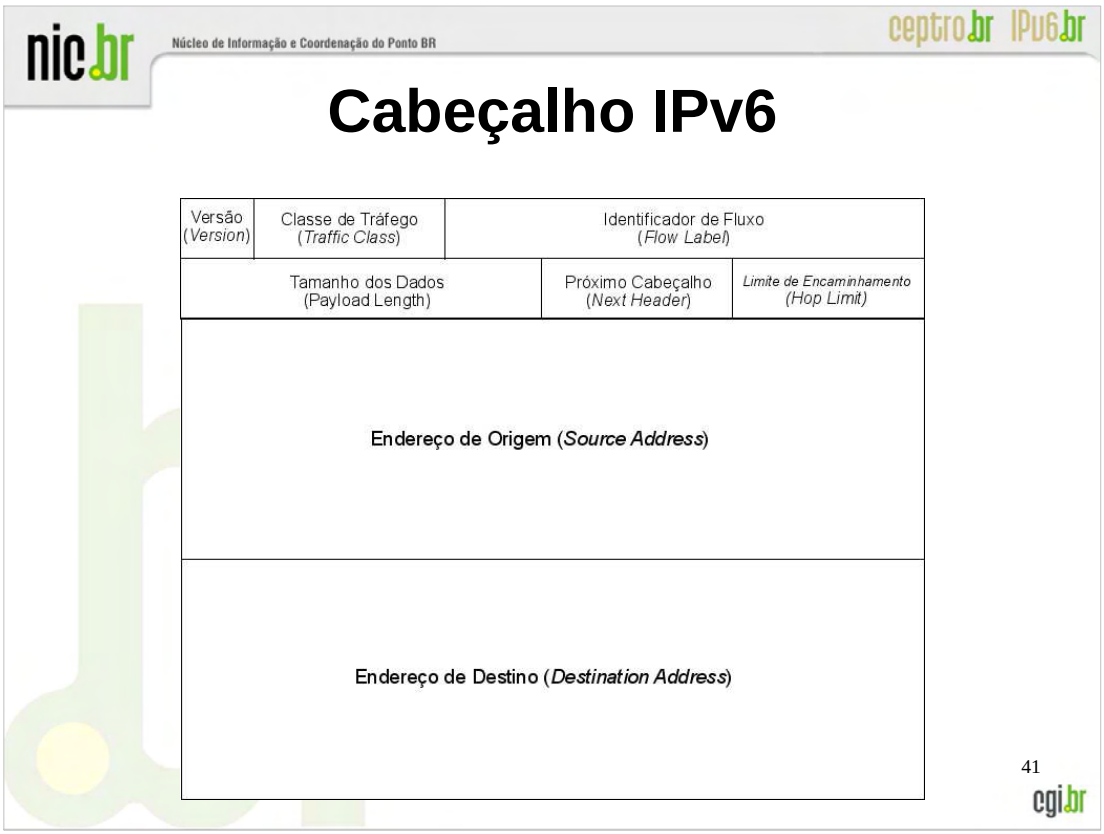
Cabeçalho IPv6



- Seis campos do cabeçalho IPv4 foram removidos.
- Quatro campos tiveram seus nomes alterados e seus posicionamentos modificados.
- O campo Identificador de Fluxo foi acrescentado.
- Três campos foram mantidos.

cgi.br

Os campo Versão, Endereço de Origem e Endereço de Destino foram mantidos, alterando apenas o tamanho do espaço reservado para o endereçamento que passa a ter 128 bits.



Vamos conhecer um pouco sobre cada campo do cabeçalho base do IPv6:

•**Versão** (4 bits) - Identifica a versão do protocolo IP utilizado. No caso do IPv6 o valor desse campo é 6.

•**Classe de Tráfego** (8 bits) - Identifica e diferencia os pacotes por classes de serviços ou prioridade. Ele continua provendo as mesmas funcionalidades e definições do campo Tipo de Serviço do IPv4.

•**Identificador de Fluxo** (20 bits) - Identifica e diferencia pacotes do mesmo fluxo na camada de rede. Esse campo permite ao roteador identificar o tipo de fluxo de cada pacote, sem a necessidade de verificar sua aplicação.

•**Tamanho do Dados** (16 bits) - Indica o tamanho, em Bytes, apenas dos dados enviados junto ao cabeçalho IPv6. Substituiu o campo Tamanho Total do IPv4, que indica o tamanho do cabeçalho mais o tamanho dos dados transmitidos. Os cabeçalhos de extensão também são incluídos no calculo do tamanho.

•**Próximo Cabeçalho** (8 bits) - Identifica cabeçalho que se segue ao cabeçalho IPv6. Este campo foi renomeado (no IPv4 chamava-se Protocolo) refletindo a nova organização dos pacotes IPv6, pois agora este campo não contém apenas valores referentes a outros protocolos, mas também indica os valores dos cabeçalhos de extensão.

•**Limite de Encaminhamento** (8 bits) - Indica o número máximo de roteadores que o pacote IPv6 pode passar antes de ser descartado, sendo decrementado a cada salto. Padronizou o modo como o campo Tempo de Vida (TTL) do IPv4 tem sido utilizado, apesar da definição original do campo TTL, dizer que este deveria indicar, em segundos, quanto tempo o pacote levaria para ser descartado caso não chegasse ao seu destino.

•**Endereço de origem** (128 bits) - Indica o endereço de origem do pacote.

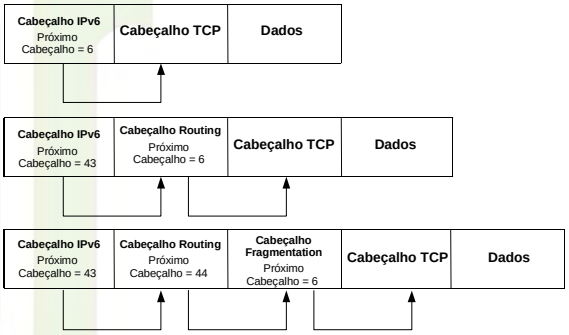
•**Endereço de Destino** (128 bits) - Indica o endereço de destino do pacote.

nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

Cabeçalhos de Extensão

- No IPv6, opções adicionais são tratadas por meio de cabeçalhos de extensão.
- Localizam-se entre o cabeçalho base e o cabeçalho da camada de transporte.
- Não há nem quantidade, nem tamanho fixo para estes cabeçalhos.

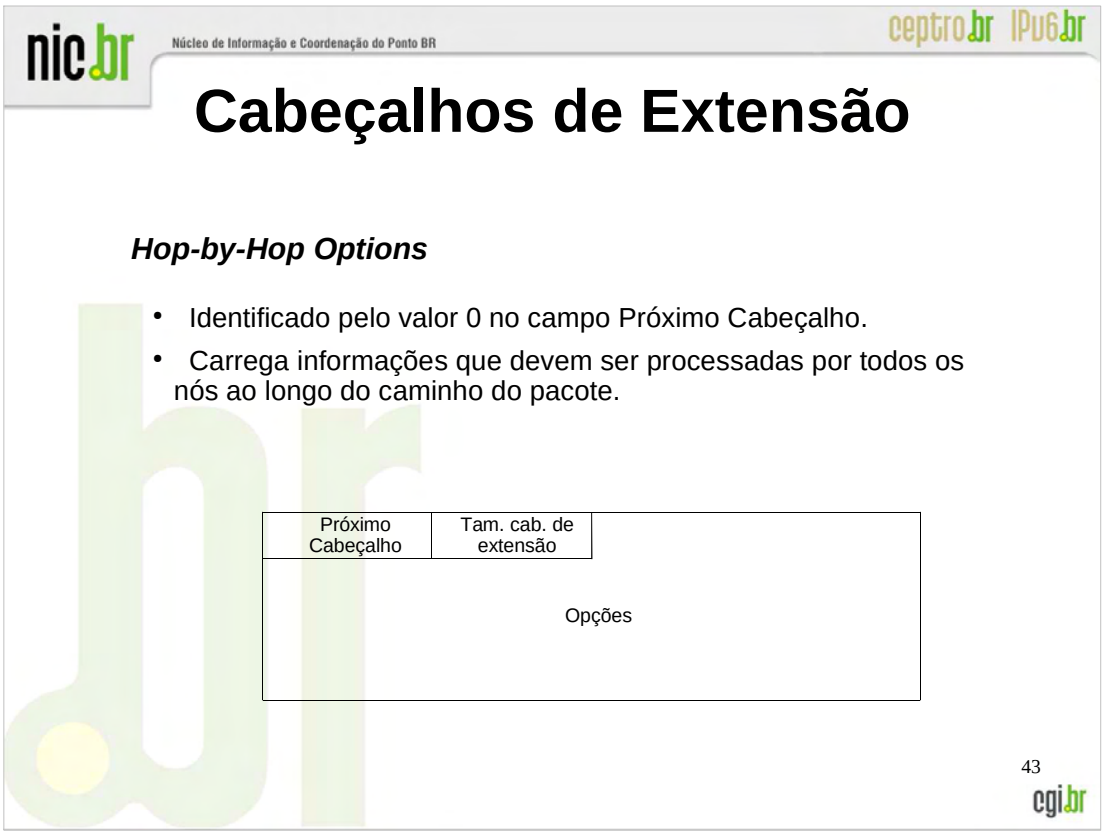


cgi.br

Diferente do IPv4, que inclui no cabeçalho base todas as informações opcionais, o IPv6 trata essas informações através de cabeçalhos de extensão. Estes cabeçalhos localizam-se entre o cabeçalho base e o cabeçalho da camada imediatamente acima, não havendo nem quantidade, nem tamanho fixo para eles. Caso existam múltiplos cabeçalhos de extensão no mesmo pacote, eles serão adicionados em série formando uma “cadeia de cabeçalhos”.

As especificações do IPv6 definem seis cabeçalhos de extensão: *Hop-by-Hop Options*, *Destination Options*, *Routing*, *Fragmentation*, *Authentication Header* e *Encapsulating Security Payload.*

A utilização dos cabeçalhos de extensão do IPv6, visa aumentar a velocidade de processamento nos roteadores, visto que, o único cabeçalho de extensão processado em cada roteador é o *Hop-by-Hop;* os demais são tratados apenas pelo nó identificado no campo Endereço de Destino do cabeçalho base. Além disso, novos cabeçalhos de extensão podem ser definidos e usados sem a necessidade de se alterar o cabeçalho base.



Identificado pelo valor 0 no campo Próximo Cabeçalho, o cabeçalho de extensão *Hop-by-Hop* deve ser colocado imediatamente após o cabeçalho base IPv6. As informações carregadas por ele devem ser examinadas por todos os nós intermediários ao longo do caminho do pacote até o destino. Na sua ausência, o roteador sabe que não precisa processar nenhuma informação adicional e assim pode encaminha o pacote para o destino final imediatamente.

As definições de cada campo do cabeçalho são as seguintes:

- **Próximo Cabeçalho** (1 Byte): Identifica o tipo de cabeçalho que segue ao *Hop-by-Hop*.
- **Tamanho do Cabeçalho** (1 Byte): Indica o tamanho do cabeçalho *Hop-by-Hop* em unidades de 8 Bytes, excluído o oito primeiros.
- **Opções**: Contem uma ou mais opções e seu tamanho é variável. Neste campo, o primeiro Byte contem informações sobre como estas opções devem ser tratadas no caso o nó que esteja processando a informação não a reconheça. O valor dos primeiros dois bits especifica as ações a serem tomadas:

    - 00: ignorar e continuar o processamento.
    - 01: descartar o pacote.
    - 10: descartar o pacote e enviar uma mensagem ICMP *Parameter Problem* para o endereço de origem do pacote.
    - 11: descartar o pacote e enviar uma mensagem ICMP *Parameter Problem* para o endereço de origem do pacote, apenas se o destino não for um endereço de *multicast*.

O terceiro bit deste campo especifica se a informação opcional pode mudar de rota (valor 01) ou não (valor 00).

Até o momento existem dois tipos definidos para o cabeçalho *Hop-by-Hop*: a *Router Alert* e a *Jumbogram*.

- ***Router Alert***: Utilizado para informar aos nós intermediários que a mensagem a ser encaminhada exige tratamento especial. Está opção é utilizada pelos protocols MLD (*Multicast Listener Discovery*) e RSVP (*Resource Reservation Protocol*).
- ***Jumbogram***: Utilizado para informa que o tamanho do pacote IPv6 é maior do que 64KB.

Mais informações:

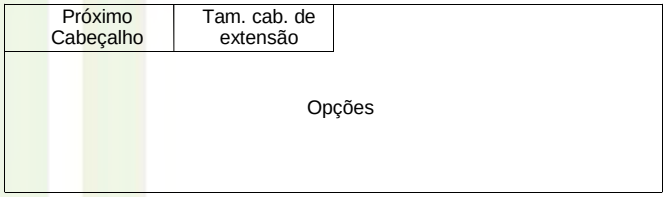
- RFC 2711 - *IPv6 Router Alert Option*

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Cabeçalhos de Extensão

Destination Options

- Identificado pelo valor 60 no campo Próximo Cabeçalho.
- Carrega informações que devem ser processadas pelo nó de destino do pacote.



45

cgi.br

Identificado pelo valor 60 no campo Próximo Cabeçalho, o cabeçalho de extensão *Destination Options* carrega informações que devem ser processadas pelo nó de destino do pacote, indicado no campo Endereço de Destino do cabeçalho base. A definição de seus campos é igual as do cabeçalho *Hop-by-Hop.*

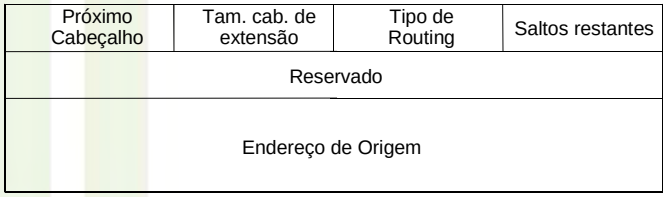
Este cabeçalho é utilizado no suporte a mobilidade do IPv6 através da opção *Home Address*, que contem o Endereço de Origem do Nó Móvel quando este está em transito.

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Cabeçalhos de Extensão

Routing

- Identificado pelo valor 43 no campo Próximo Cabeçalho.
- Desenvolvido inicialmente para listar um ou mais nós intermediários que deveriam ser visitados até o pacote chegar ao destino.
- Atualmente utilizado como parte do mecanismo de suporte a mobilidade do IPv6.



Identificado pelo valor 43 no campo Próximo Cabeçalho, o cabeçalho de extensão *Routing* foi desenvolvido inicialmente para listar um ou mais nós intermediários que deveriam ser visitados até o pacote chegar ao destino, semelhante às opções *Loose Source* e *Record Route* do IPv4. Esta função, realizada pelo cabeçalho *Routing Type 0*, tornou-se obsoleta pela RFC5095 devido a problemas de segurança.

Um novo cabeçalho *Routing*, o *Type 2*, foi definido para ser utilizado como parte do mecanismo de suporte a mobilidade do IPv6, carregando o Endereço de Origem do Nó Móvel em pacotes enviados pelo Nó Correspondente.

As definições de cada campo do cabeçalho são as seguintes:

- **Próximo Cabeçalho** (1 Byte): Identifica o tipo de cabeçalho que segue ao cabeçalho *Routing*.
- **Tamanho do Cabeçalho** (1 Byte): Indica o tamanho do cabeçalho *Routing* em unidades de 8 Bytes, excluído o oito primeiros.
- ***Routing Type*** (1 Byte): Identifica o tipo de cabeçalho *Routing*. Atualmente apenas o *Type* 2 está definido.
- **Saltos restantes**: Definido para ser utilizado com o *Routing Type* 0, indica o número de saltos a serem visitados antes do pacote atingir seu destino final.
- **Endereço de Origem**: Carrega o Endereço de Origem de um Nó Móvel.

Mais informações:

- RFC 3775 - *Mobility Support in IPv6 - 6.4. Type 2 Routing Header*
- RFC 5095 - *Deprecation of Type 0 Routing Headers in IPv6*

Cabeçalhos de Extensão

Fragmentation

- Identificado pelo valor 44 no campo Próximo Cabeçalho.
- Carrega informações sobre os fragmentos dos pacotes IPv6.

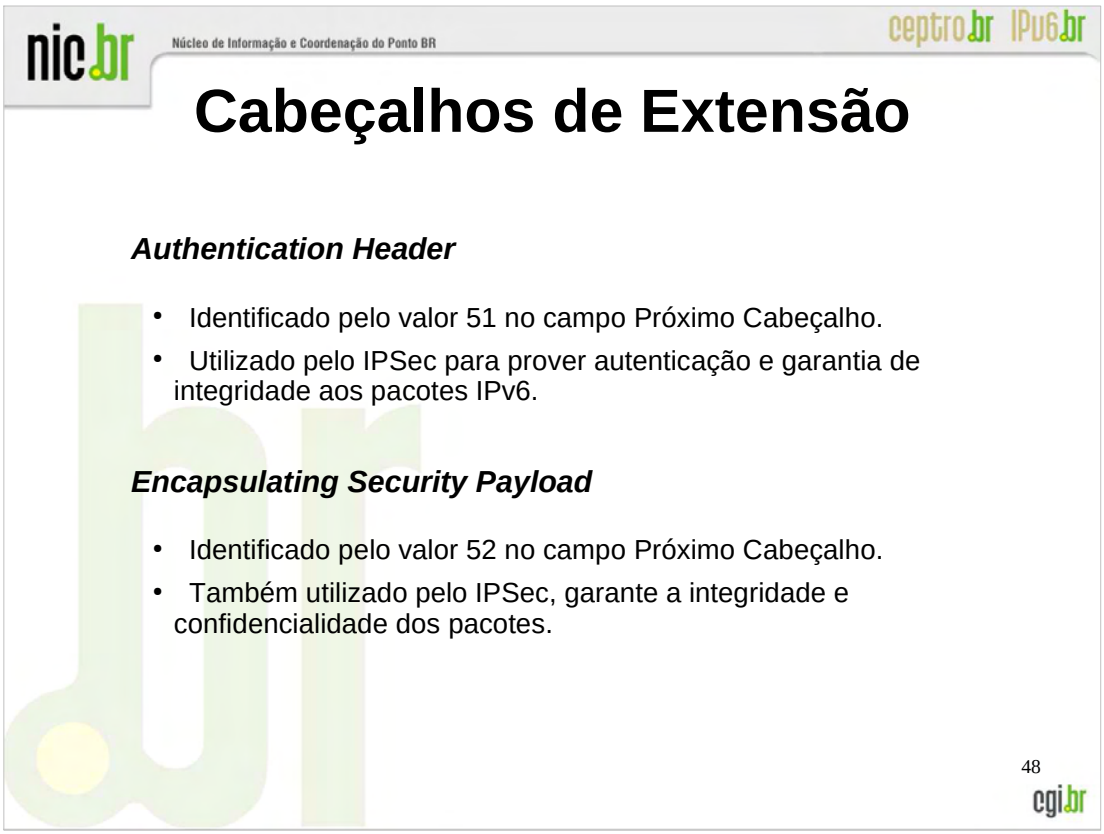
Próximo Cabeçalho	Reservado	Deslocamento do Fragmento	Res	M
Identificação				

Identificado pelo valor 44 no campo Próximo Cabeçalho, o cabeçalho de extensão *Fragmentation* é utilizado quando o pacote IPv6 a ser enviado é maior que o *Path MTU.*

As definições de cada campo do cabeçalho são as seguintes:

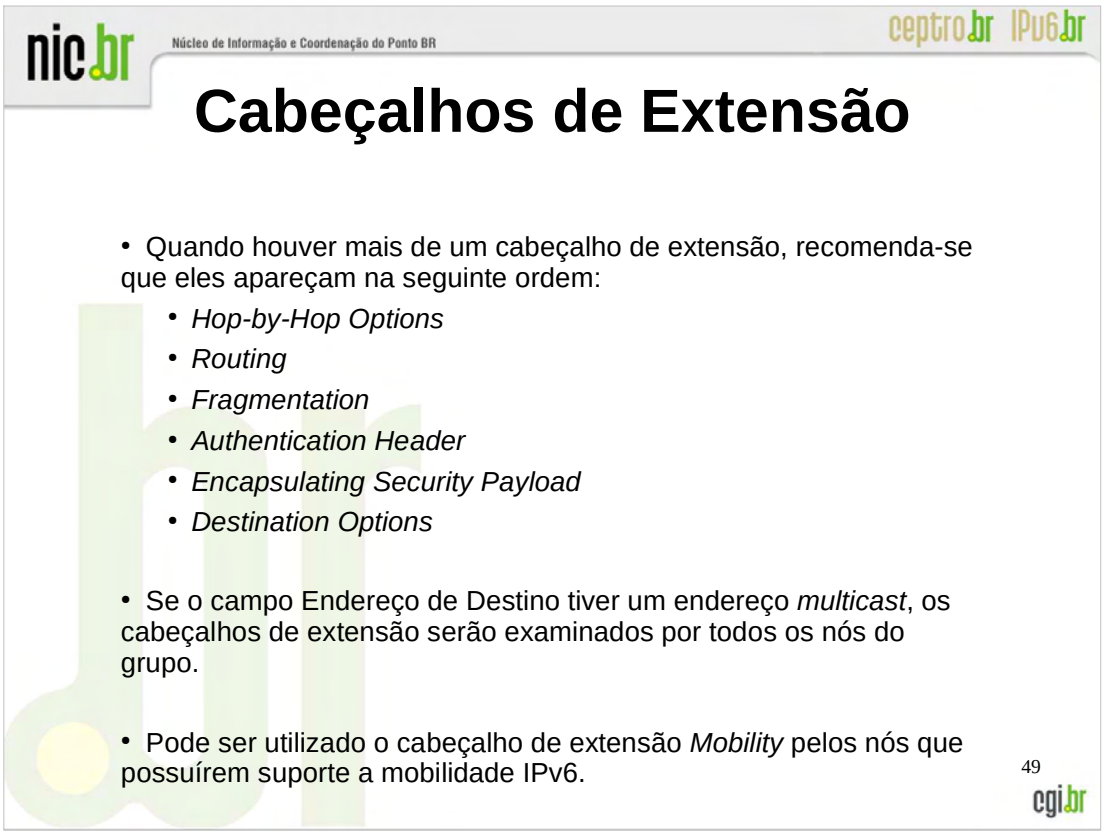
- **Próximo Cabeçalho** (1 Byte): Identifica o tipo de cabeçalho que segue ao cabeçalho *Fragmentation*.
- **Deslocamento do Fragmento** (13 bits): Indica, em unidades de oito Bytes, a posição dos dados transportados pelo fragmento atual em relação ao início do pacote original.
- ***Flag* M** (1 bit): Se marcado com o valor 1, indica que há mais fragmentos. Se marcado com o valor 0, indica que é o fragmento final.
- **Identificação** (4 Bytes): Valor único gerado pelo nó de origem, para identificar o pacote original. É utilizado para detectar os fragmentos de um mesmo pacote.

O processo de fragmentação de pacotes do IPv6 será detalhado nos próximos módulos.



Os cabeçalhos de extensão *Authentication Header e Encapsulating Security Payload*, indicados respectivamente pelos valores 51 e 52 no campo Próximo Cabeçalho, fazem parte do cabeçalho IPSec.

Embora as funcionalidades do IPSec sejam idênticas tanto no IPv4 quanto no IPv6, sua utilização com IPv6 é facilitada pelo fato de seu principais elementos serem parte integrante da nova versão do protocolo IP. Outros aspectos também facilitam essa utilização, como o fato de não se utilizar NAT com IPv6, no entanto essa questão será detalhada nos próximos módulos, juntamente com o detalhamento dos cabeçalhos de extensão AH e ESP.



Alguns aspectos sobre os cabeçalhos de extensão devem ser observados.

Primeiramente é importante destacar que, para evitar que os nós existentes ao longo do caminho do pacote tenham que percorrer toda a cadeia de cabeçalhos de extensão para conhecer quais informações deverão tratar, estes cabeçalhos devem ser enviados respeitando um determinada ordem. Geralmente, os cabeçalhos importantes para todos os nós envolvidos no roteamento devem ser colocados em primeiro lugar, cabeçalhos importantes apenas para o destinatário final são colocados no final da cadeia. A vantagem desta seqüência é que o nó pode parar de processar os cabeçalhos assim que encontrar algum cabeçalho de extensão dedicado ao destino final, tendo certeza de que não há mais cabeçalhos importantes a seguir. Com isso, é possível melhorar significativamente o processamento dos pacotes, porque, em muitos casos, apenas o processamento do cabeçalho base será suficiente para encaminhar o pacote. Deste modo, a sequência a ser seguida é:

- *Hop-by-Hop Options*
- *Routing*
- *Fragmentation*
- *Authentication Header*
- *Encapsulating Security Payload*
- *Destination Options*

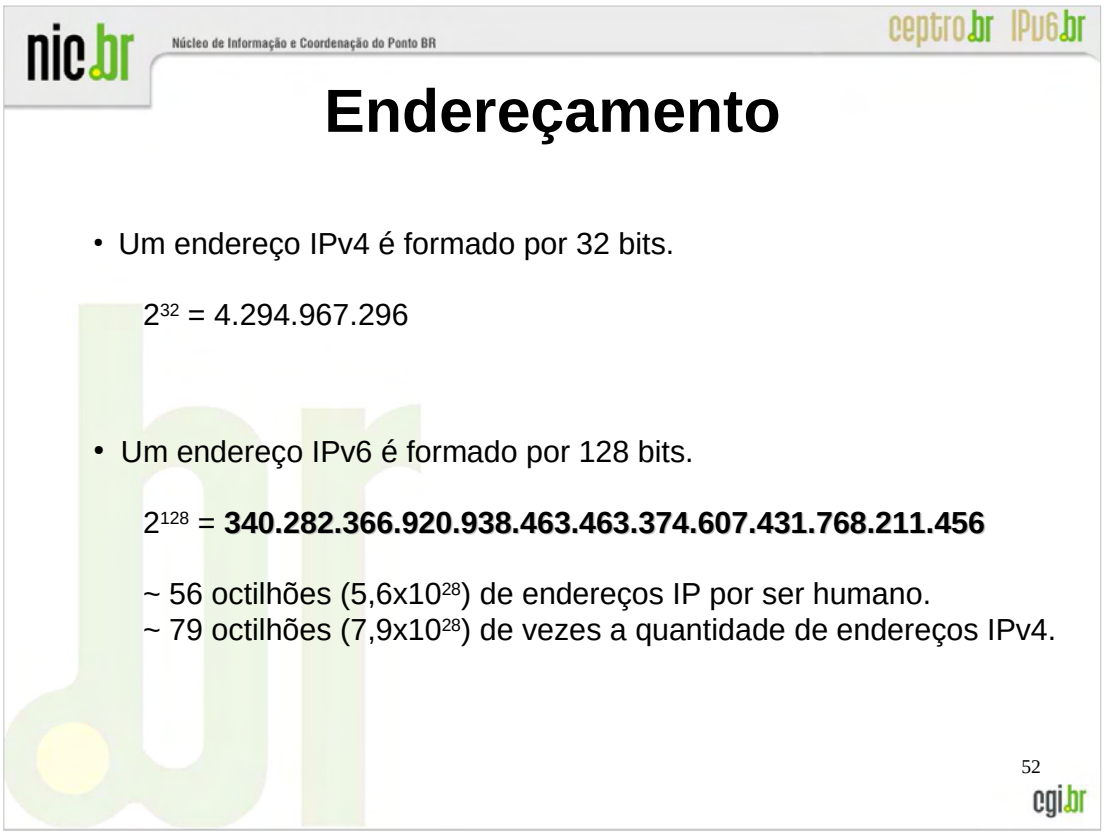
Também é vale observar, que se um pacote for enviado para um endereço *multicast*, os cabeçalhos de extensão serão examinados por todos os nós do grupo.

Em relação à flexibilidade oferecida pelos cabeçalhos de extensão, merece destaque o desenvolvido o cabeçalho *Mobility*, utilizado pelos nós que possuem suporte a mobilidade IPv6.

IPv6.br

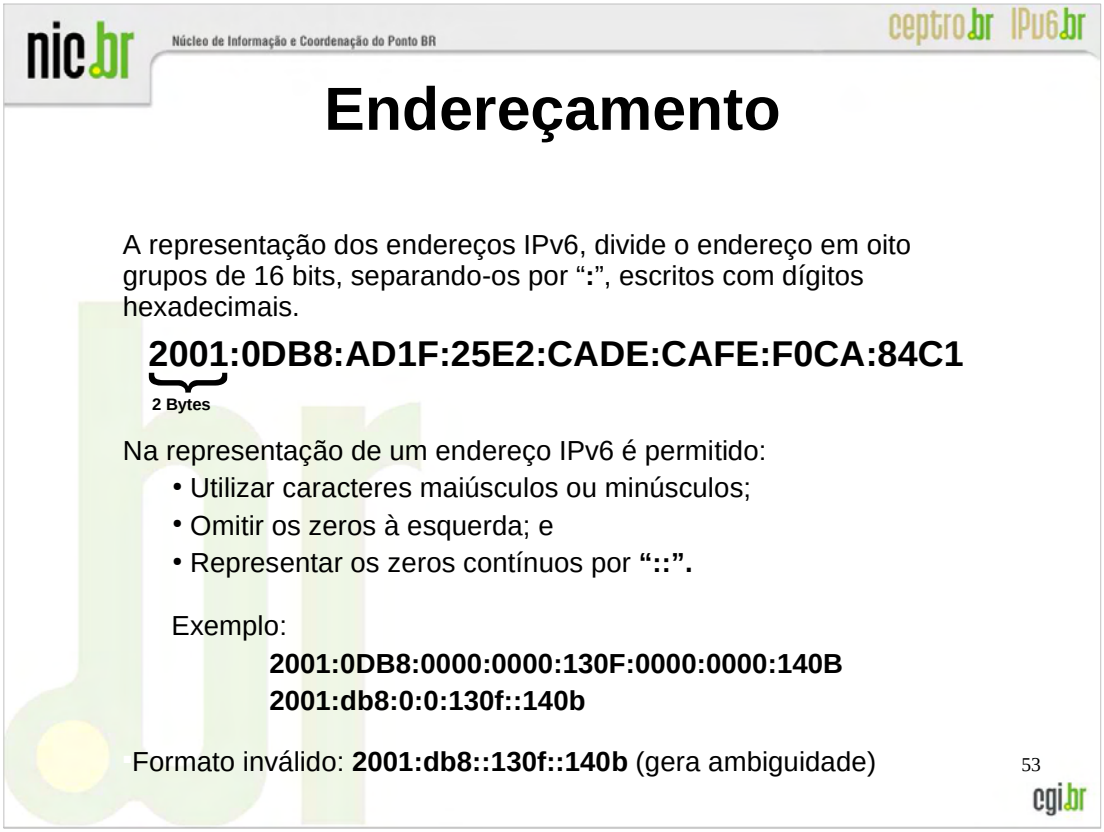
A Nova Geração do Protocolo Internet

O protocolo IPv6 apresenta como principal característica e justificativa maior para o seu desenvolvimento, o aumento no espaço para endereçamento. Por isso, é importante conhecermos as diferenças entre os endereços IPv4 e IPv6, saber reconhecer a sintaxe dos endereços IPv6 e conhecer os tipos de endereços IPv6 existentes e suas principais características.



No IPv4, o campo do cabeçalho reservado para o endereçamento possui 32 bits. Este tamanho possibilita um máximo de 4.294.967.296 (2^{32}) endereços distintos. A época de seu desenvolvimento, está quantidade era considerada suficiente para identificar todos os computadores na rede e suportar o surgimento de novas sub-redes. No entanto, com o rápido crescimento da Internet, surgiu o problema da escassez dos endereços IPv4, motivando a a criação de uma nova geração do protocolo IP.

O IPv6 possui um espaço para endereçamento de 128 bits, sendo possível obter 340.282.366.920.938.463.463.374.607.431.768.211.456 endereços (2^{128}). Este valor representa aproximadamente 79 octilhões ($7,9x10^{28}$) de vezes a quantidade de endereços IPv4 e representa, também, mais de 56 octilhões ($5,6x10^{28}$) de endereços por ser humano na Terra, considerando-se a população estimada em 6 bilhões de habitantes.



Os 32 bits dos endereços IPv4 são divididos em quatro grupos de 8 bits cada, separados por “**.**”, escritos com dígitos decimais. Por exemplo: **192.168.0.10**.

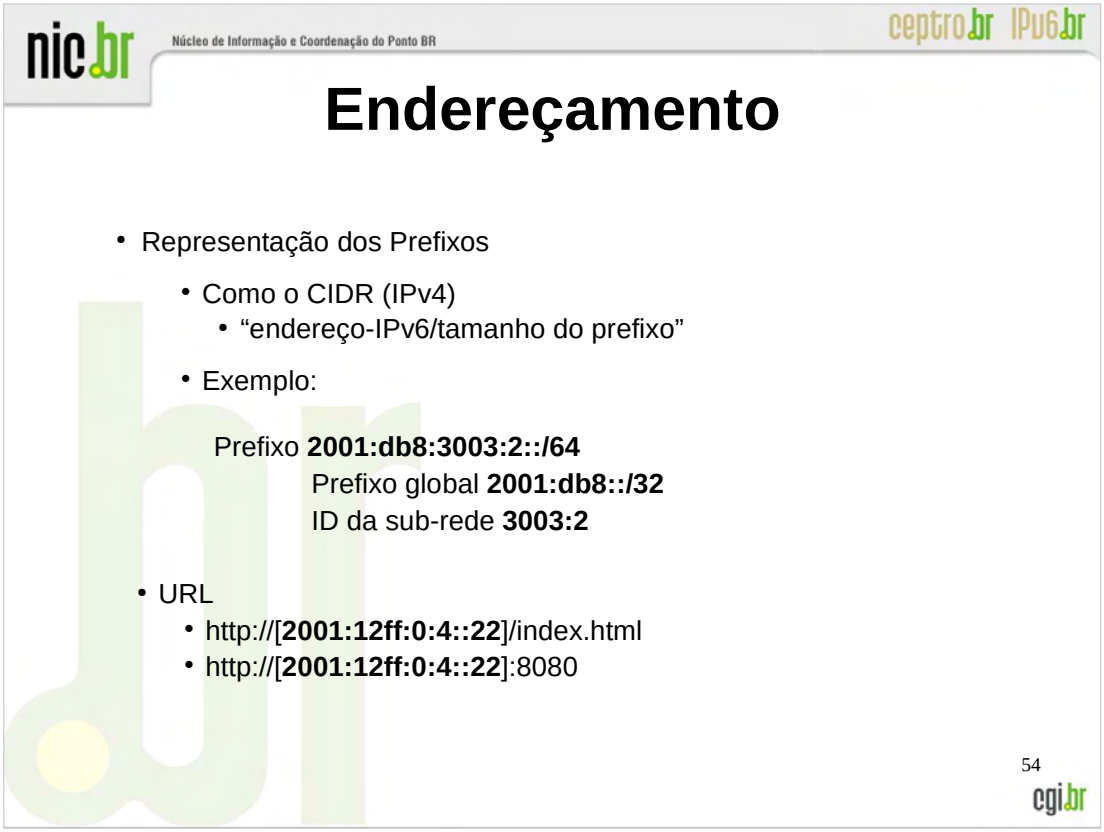
A representação dos endereços IPv6, divide o endereço em oito grupos de 16 bits, separando-os por “**:**”, escritos com dígitos hexadecimais (0-F). Por exemplo:

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

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

Além disso, regras de abreviação podem ser aplicadas para facilitar a escrita de alguns endereços muito extensos. É permitido omitir os zeros a esquerda de cada bloco de 16 bits, além de substituir uma sequência longa de zeros por “**::**”. Por exemplo, o endereço **2001:0DB8:0000:0000:130F:0000:0000:140B** pode ser escrito como **2001:DB8:0:0:130F::140B** ou **2001:DB8::130F:0:0:140B**. Neste exemplo é possível observar que a abreviação do grupo de zeros só pode ser realizada uma única vez, caso contrário poderá haver ambigüidades na representação do endereço. Se o endereço acima fosse escrito como **2001:DB8::130F::140B**, não seria possível determinar se ele corresponde a **2001:DB8:0:0:130F:0:0:140B**, a **2001:DB8:0:0:0:130F:0:140B** ou **2001:DB8:0:130F:0:0:0:140B.**

Esta abreviação pode ser feita também no fim ou no início do endereço, como ocorre em **2001:DB8:0:54:0:0:0:0** que pode ser escrito da forma **2001:DB8:0:54::**.



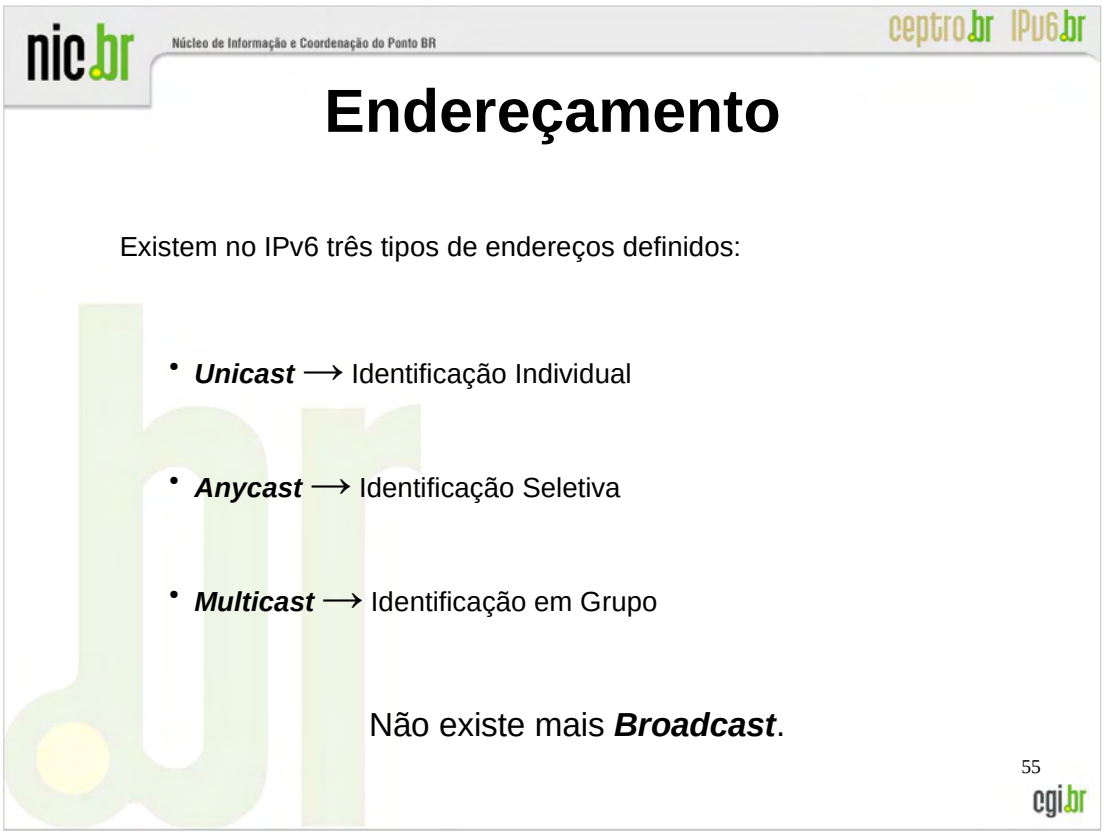
Outra representação importante é a dos prefixos de rede. Em endereços IPv6 ela continua sendo escrita do mesmo modo que no IPv4, utilizando a notação CIDR. Esta notação é representada da forma "endereço-IPv6/tamanho do prefixo", onde "tamanho do prefixo" é um valor decimal que especifica a quantidade de bits contíguos à esquerda do endereço que compreendem o prefixo. O exemplo de prefixo de sub-rede apresentado a seguir indica que dos 128 bits do endereço, 64 bits são utilizados para identificar a sub-rede.

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

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

Com relação a representação dos endereços IPv6 em URLs (*Uniform Resource Locators*), estes agora passam a ser representados entre colchetes. Deste modo, não haverá ambiguidades caso seja necessário indicar o número de uma porta juntamente com a URL. Observe os exemplos a seguir:

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



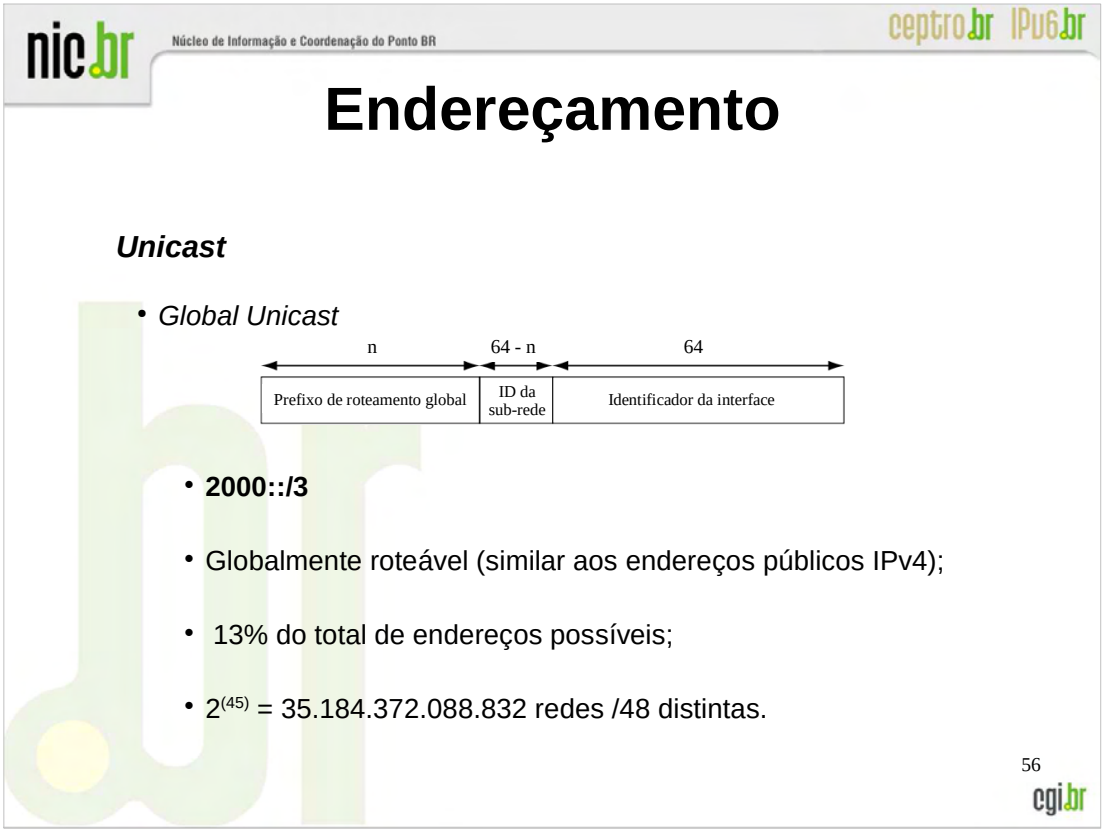
Existem no IPv6 três tipos de endereços definidos:

- ***Unicast*** – este tipo de endereço identifica uma única interface, de modo que um pacote enviado a um endereço *unicast* é entregue a uma única interface;

- ***Anycast*** – identifica um conjunto de interfaces. Um pacote encaminhado a um endereço *anycast* é entregue a interface pertencente a este conjunto mais próxima da origem (de acordo com distância medida pelos protocolos de roteamento). Um endereço *anycast* é utilizado em comunicações de um-para-um-de-muitos.

- ***Multicast*** – também identifica um conjunto de interfaces, entretanto, um pacote enviado a um endereço multicast é entregue a todas as interfaces associadas a esse endereço. Um endereço *multicast* é utilizado em comunicações de um-para-muitos.

Diferente do IPv4, no IPv6 não existe endereço *broadcast*, responsável por direcionar um pacote para todos os nós de um mesmo domínio. No IPv6, essa função foi atribuída à tipos específicos de endereços *multicast*.



Os endereços *unicast* são utilizados para comunicação entre dois nós, por exemplo, telefones VoIPv6, computadores em uma rede privada, etc., e sua estrutura foi definida para permitir agregações com prefixos de tamanho flexível, similar ao CIDR do IPv4.

Existem alguns tipos de endereços *unicast* IPv6: *Global Unicast*; *Unique-Local*; e *Link-Local* por exemplo. Existem também alguns tipos para usos especiais, como endereços IPv4 mapeados em IPv6, endereço de *loopback* e o endereço não-especificado, entre outros.

– ***Global Unicast*** - equivalente aos endereços públicos IPv4, o endereço *global unicast* é globalmente roteável e acessível na Internet IPv6. Ele é constituído por três partes: o prefixo de roteamento global, utilizado para identificar o tamanho do bloco atribuído a uma rede; a identificação da sub-rede, utilizada para identificar um enlace em uma rede; e a identificação da interface, que deve identificar de forma única uma interface dentro de um enlace.

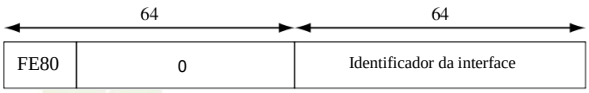
  Sua estrutura foi projetada para utilizar os 64 bits mais a esquerda para identificação da rede e os 64 bits mais a direita para identificação da interface. Portanto, exceto casos específicos, todas as sub-redes em IPv6 tem o mesmo tamanho de prefixo, 64 bits (/64), o que possibilita 2^{64} = 18.446.744.073.709.551.616 dispositivos por sub-rede.

  Atualmente, está reservada para atribuição de endereços a faixa **2000::/3** (001), que corresponde aos endereços de **2000::** a **3fff:ffff:ffff:ffff:ffff:ffff:ffff:ffff**. Isto representa 13% do total de endereços possíveis com IPv6, o que nos permite criar $2^{(64-3)}$ = 2.305.843.009.213.693.952 ($2{,}3 \times 10^{18}$) sub-redes (/64) diferentes ou $2^{(48-3)}$ = 35.184.372.088.832 ($3{,}5 \times 10^{13}$) redes /48.

Endereçamento

Unicast

- *Link local*



  - **FE80::/64**
  - Deve ser utilizado apenas localmente;
  - Atribuído automaticamente (autoconfiguração *stateless*);

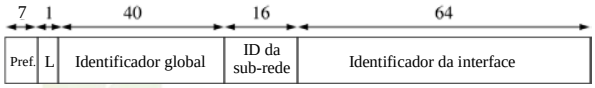
– ***Link Local*** – podendo ser usado apenas no enlace específico onde a interface está conectada, o endereço *link local* é atribuído automaticamente utilizando o prefixo **FE80::/64**. Os 64 bits reservados para a identificação da interface são configurados utilizando o formato IEEE EUI-64. Vale ressaltar que os roteadores não devem encaminhar para outros enlaces, pacotes que possuam como origem ou destino um endereço *link-local*;

nic.br Núcleo de Informação e Coordenação do Ponto BR — ceptro.br IPv6.br

Endereçamento

Unicast

- *Unique local*



  - **FC00::/7**
  - Prefixo globalmente único (com alta probabilidade de ser único);
  - Utilizado apenas na comunicação dentro de um enlace ou entre um conjunto limitado de enlaces;
  - Não é esperado que seja roteado na Internet.

58

cgi.br

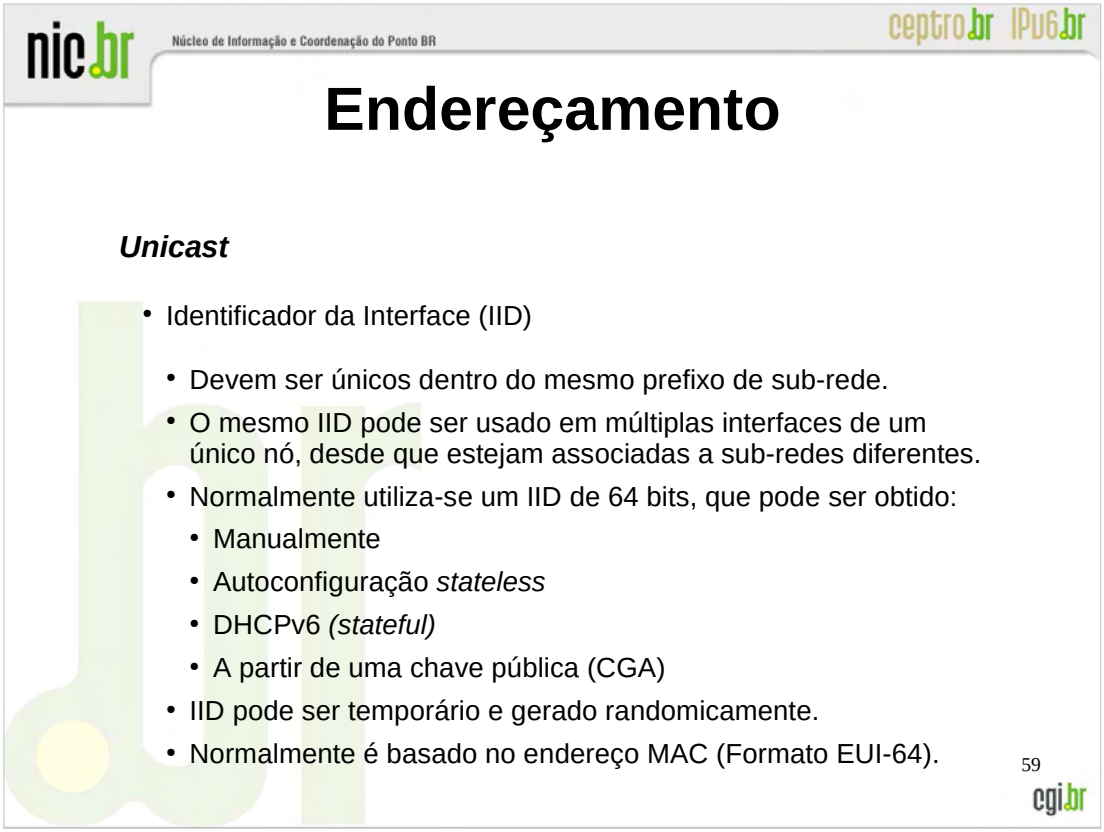
- ***Unique Local Address*** (ULA) – endereço com grande probabilidade de ser globalmente único, utilizado apenas para comunicações locais, geralmente dentro de um mesmo enlace ou conjunto de enlaces. Um endereço ULA não deve ser roteável na Internet global.

  Um endereço ULA, criado utilizado um ID globa alocado pseudo-randomicamente, é composto das seguintes partes:

  - **Prefixo**: **FC00::/7**.
  - **Flag Local (L)**: se o valor for 1 (**FD**) o prefixo é atribuído localmente. Se o valor for 0 **(FC),** o prefixo deve ser atribuído por uma organização central (ainda a definir).
  - **Identificador global**: identificador de 40 bits usado para criar um prefixo globalmente único.
  - **Identificador da Interface**: identificador da interface de 64 bits.

  Deste modo, a estrutura de um endereço ULA é **FDUU:UUUU:UUUU:<ID da sub-rede>:<Id da interface>** onde **U** são os bits do identificador único, gerado aleatoriamente por um algoritmo específico.

  Sua utilização permite que qualquer enlace possua um prefixo /48 privado e único globalmente. Deste modo, caso duas redes, de empresas distintas por exemplo, sejam interconectadas, provavelmente não haverá conflito de endereços ou necessidade de renumerar a interface que o esteja usando. Além disso, o endereço ULA é independente de provedor, podendo ser utilizado na comunicação dentro do enlace mesmo que não haja uma conexão com a Internet. Outra vantagem, é que seu prefixo pode ser facilmente bloqueado, e caso um endereço ULA seja anunciado acidentalmente para fora do enlace, através de um roteador ou via DNS, não haverá conflito com outros endereços.



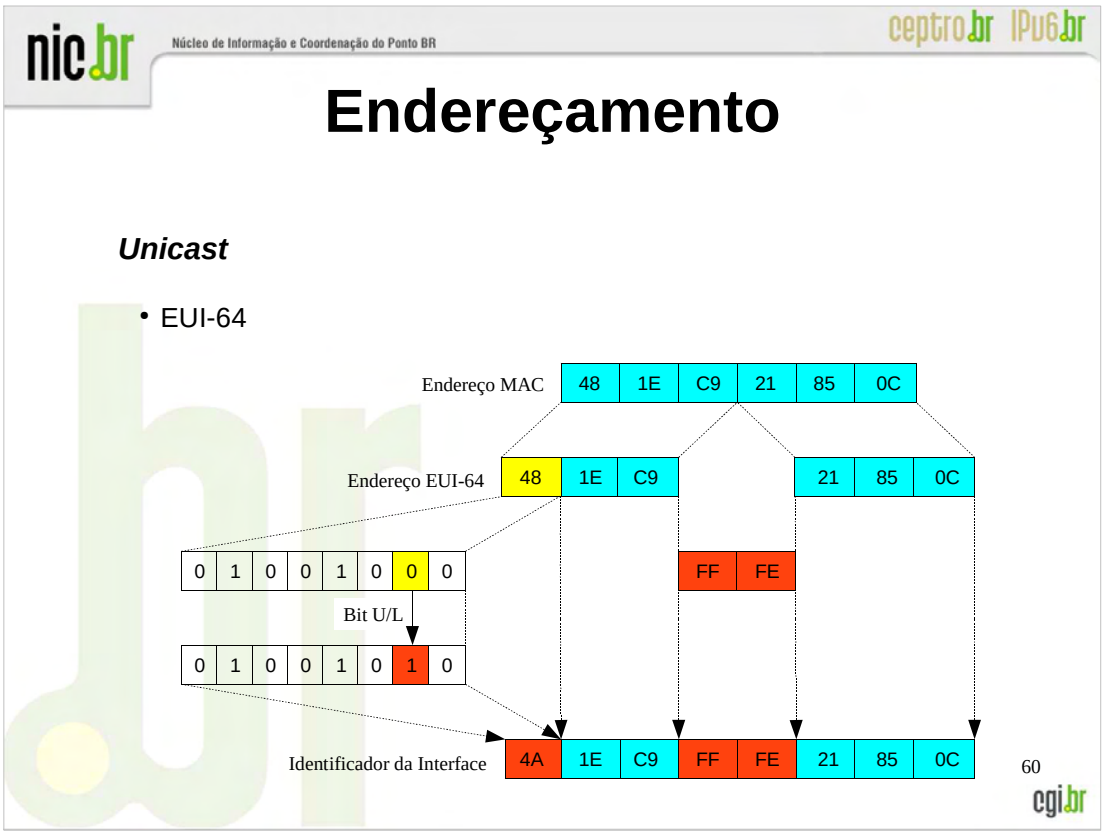
Os identificadores de interface (IID), utilizados para distinguir as interfaces dentro de um enlace, devem ser únicos dentro do mesmo prefixo de sub-rede. O mesmo IID pode ser usado em múltiplas interfaces em um único nó, porém, elas dever estar associadas a deferentes sub-redes.

Normalmente utiliza-se um IID de 64 bits, que pode ser obtido de diversas formas. Ele pode ser configurado manualmente, a partir do mecanismo de autoconfiguração *stateless* do IPv6, a partir de servidores DHCPv6 (*stateful*), ou formados a partir de uma chave pública (CGA). Estes métodos serão detalhados no decorrer deste curso.

Embora eles possam ser gerados randomicamente e de forma temporária, recomenda-se que o IID seja construído baseado no endereço MAC da interface, no formato EUI-64.

Mais informações:

- RFC 3986 - *Uniform Resource Identifier (URI): Generic Syntax*
- RFC 4291 - *IP Version 6 Addressing Architecture*
- RFC 4193 - *Unique Local IPv6 Unicast Addresses*
- RFC 5156 - *Special-Use IPv6 Addresses*
- RFC 3587 - *IPv6 Global Unicast Address Format*
- *Internet Protocol Version 6 Address Space* - http://www.iana.org/assignments/ipv6-address-space



Um IID baseado no formato EUI-64 é criado da seguinte forma:

– Caso a interface possua um endereço MAC de 64 bits (padrão EUI-64), basta complementar o sétimo bit mais a esquerda (chamado de bit U/L – Universal/Local) do endereço MAC, isto é, se for 1, será alterado para 0; se for 0, será alterado para 1. Caso a interface utilize um endereço MAC de 48 bits (padrão IEEE 802), primeiro adiciona-se os dígitos hexadecimais FF-FE entre o terceiro e quarto Byte do endereço MAC (transformando no padrão EUI-64), e em seguida, o bit U/L é complementado. Por exemplo:

– Se endereço MAC da interface for:
  • 48-1E-C9-21-85-0C

– adiciona-se os dígitos FF-FE na metade do endereço:
  • 48-1E-C9-**FF**-**FE**-21-85-0C

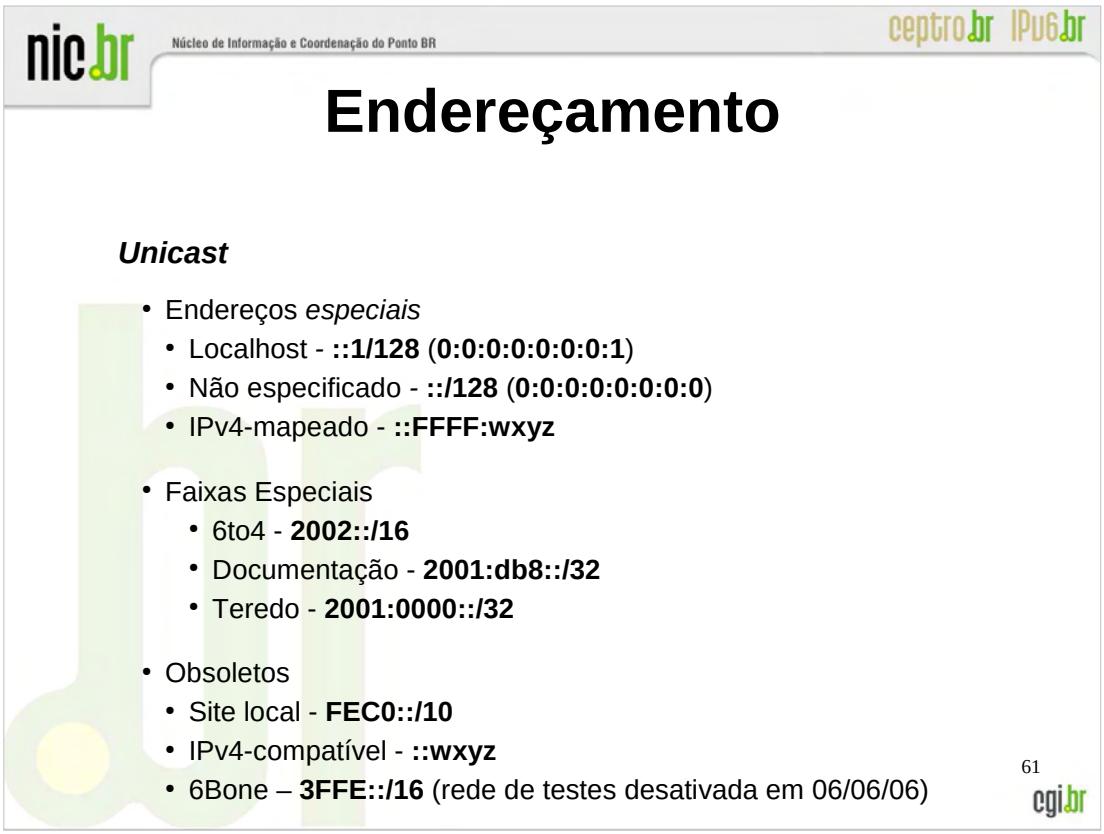
– complementa-se o bit U/L:
  • 48 = 01001000
  • 01001000 → 01001010
  • 01001010 = 4A

– IID = 4A-1E-C9-FF-FE-21-85-0C

Um endereço *link local* atribuído à essa interface seria **FE80::4A1E:C9FF:FE21:850C**.

Mais informações:

• *Guidelines for 64-bit Global Identifier (EUI-64) Registration Authority* - http://standards.ieee.org/regauth/oui/tutorials/EUI64.html



Existem alguns endereços IPv6 especiais utilizados para fins específicos:

- **Endereço Não-Especificado (*Unspecified)***: é representado pelo endereço **0:0:0:0:0:0:0:0** ou **::0** (equivalente ao endereço IPv4 *unspecified* **0.0.0.0**). Ele nunca deve ser atribuído a nenhum nó, indicando apenas a ausência de um endereço. Ele pode, por exemplo, ser utilizado no campo Endereço de Origem de um pacote IPv6 enviado por um *host* durante o processo de inicialização, antes que este tenha seu endereço exclusivo determinado. O endereço *unspecified* não deve ser utilizado como endereço de destino de pacotes IPv6;
- **Endereço *Loopback***: representado pelo endereço *unicast* **0:0:0:0:0:0:0:1** ou **::1** (equivalente ao endereço IPv4 *loopback* **127.0.0.1**). Este endereço é utilizado para referenciar a própria máquina, sendo muito utilizado para teste internos. Este tipo de endereço não deve ser atribuído a nenhuma interface física, nem usado como endereço de origem em pacotes IPv6 enviados para outros nós. Além disso, um pacote IPv6 com um endereço *loopback* como destino não pode ser enviado por um roteador IPv6, e caso um pacote recebido em uma interface possua um endereço *loopback* como destino, este deve ser descartado;
- **Endereços IPv4-mapeado**: representado por **0:0:0:0:0:FFFF:wxyz** ou **::FFFF:wxyz**, é usado para mapear um endereço IPv4 em um endereço IPv6 de 128-bit, onde **wxyz** representa os 32 bits do endereço IPv4, utilizando dígitos decimais. É aplicado em técnicas de transição para que nós IPv6 e IPv4 se comuniquem. Ex. **::FFFF:192.168.100.1**.

Algumas faixas de endereços também são reservadas para uso específicos:

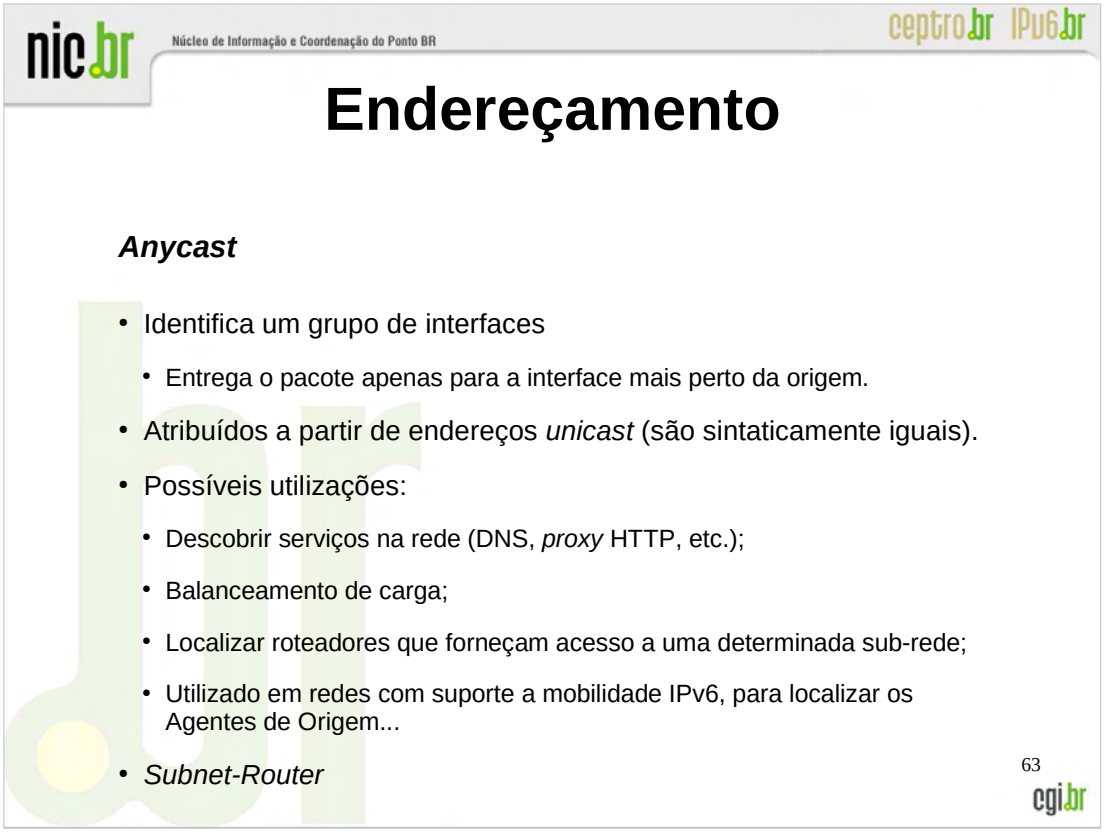
- **2002::/16**: prefixo utilizado no mecanismo de transição 6to4;
- **2001:0000::/32**: prefixo utilizado no mecanismo de transição TEREDO;
- **2001:db8::/32**: prefixo utilizado para representar endereços IPv6 em textos e documentações.

Outros endereços, utilizados no início do desenvolvimento do IPv6 tornaram-se obsoletos e não devem mais ser utilizados:

- **FEC0::/10**: prefixo utilizado pelos endereços do tipo *site local*, desenvolvidos para serem utilizados dentro de uma rede específica sem a necessidade de um prefixo global, equivalente aos endereços privados do IPv4. Sua utilização foi substituída pelos endereços ULA;
- **::wxyz**: utilizado para representar o endereço IPv4-compatível. Sua função é a mesma do endereço IPv4-mapeado, tornando-se obsoleto por desuso;
- **3FFE::/16**: prefixo utilizado para representar os endereços da rede de teste 6Bone. Criada para ajudar na implantação do IPv6, está rede foi desativada em 6 de junho de 2006 (06/06/06).

Mais informações:

- RFC 3849 - *IPv6 Address Prefix Reserved for Documentation*
- RFC 3879 - *Deprecating Site Local Addresses*



Um endereço IPv6 *anycast* é utilizado para identificar um grupo de interfaces, porém, com a propriedade de que um pacote enviado a um endereço *anycast* é encaminhado apenas a interface do grupo mais próxima da origem do pacote.

Os endereços *anycast* são atribuídos a partir da faixa de endereços *unicast* e não há diferenças sintáticas entre eles. Portanto, um endereço *unicast* atribuído a mais de uma interface transforma-se em um endereço *anycast*, devendo-se neste caso, configurar explicitamente os nós para que saibam que lhes foi atribuído um endereço *anycast.* Além disso, este endereço deve ser configurado nos roteadores como uma entrada separada (prefixo /128 – *host route*).

Este esquema de endereçamento pode ser utilizado para descobrir serviços na rede, como servidores DNS e *proxies* HTTP, garantindo a redundância desses serviços. Também pode-se utilizar para fazer balanceamento de carga em situações onde múltiplos *hosts* ou roteadores provem o mesmo serviço, para localizar roteadores que forneçam acesso a uma determinada sub-rede ou para localizar os Agentes de Origem em redes com suporte a mobilidade IPv6.

Todos os roteadores devem ter suporte ao endereço *anycast Subnet-Router*. Este tipo de endereço é formado pelo prefixo da sub-rede e pelo IID preenchido com zeros (ex.: **2001:db8:cafe:dad0::/64**). Um pacote enviado para o endereço *Subnet-Router* será entregue para o roteador mais próximo da origem dentro da mesma sub-rede.

Também foi definido um endereço *anycast* para ser utilizado no suporte a mobilidade IPv6. Este tipo de endereço é formado pelo prefixo da sub-rede seguido pelo IID **dfff:ffff:ffff:fffe** (ex.: **2001:db8::dfff:ffff:ffff:fffe**). Ele é utilizado pelo Nó Móvel, quando este precisar localizar um Agente Origem em sua Rede Original.

Mais informações:

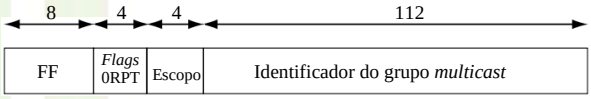
• *Internet Protocol Version 6 Anycast Addresses* - http://www.iana.org/assignments/ipv6-anycast-addresses

nic.br Núcleo de Informação e Coordenação do Ponto BR — ceptro.br IPv6.br

Endereçamento

Multicast

- Identifica um grupo de interfaces.
- O suporte a *multicast* é obrigatório em todos os nós IPv6.
- O endereço *multicast* deriva do bloco **FF00::/8**.
- O prefixo **FF** é seguido de quatro bits utilizados como *flags* e mais quatro bits que definem o escopo do endereço *multicast*. Os 112 bits restantes são utilizados para identificar o grupo *multicast*.



Endereços *multicast* são utilizados para identificar grupos de interfaces, sendo que cada interface pode pertencer a mais de um grupo. Os pacotes enviados para esses endereço são entregues a todos as interfaces que compõe o grupo.

No IPv4, o suporte a *multicast* é opcional, já que foi introduzido apenas como uma extensão ao protocolo. Entretanto, no IPv6 é requerido que todos os nós suportem *multicast*, visto que muitas funcionalidades da nova versão do protocolo IP utilizam esse tipo de endereço.

Seu funcionamento é similar ao do *broadcast*, dado que um único pacote é enviado a vários *hosts*, diferenciando-se apenas pelo fato de que no *broadcast* o pacote é enviado a todos os *hosts* da rede, sem exceção, enquanto que no *multicast* apenas um grupo de *hosts* receberá esse pacote. Deste modo, a possibilidade de transportar apenas uma cópia dos dados a todos os elementos do grupo, a partir de uma árvore de distribuição, pode reduzir a utilização de recurso de uma rede, bem como otimizar a entrega de dados aos *hosts* receptores. Aplicações como videoconferência, distribuição de vídeo sob demanda, atualizações de *softwares* e jogos *on-line*, são exemplos de serviços que vêm ganhando notoriedade e podem utilizar as vantagens apresentadas pelo *multicast*.

Os endereços *multicast* não devem ser utilizados como endereço de origem de um pacote. Esses endereços derivam do bloco **FF00::/8**, onde o prefixo **FF**, que identifica um endereço *multicast*, é precedido por quatro bits, que representam quatro *flags*, e um valor de quatro bits que define o escopo do grupo *multicast.* Os 112 bits restantes são utilizados para identificar o grupo multicast.

nic.br Núcleo de Informação e Coordenação do Ponto BR — ceptro.br IPv6.br

Endereçamento

Multicast

- ***Flags***

Flag	Valor (binário)	Descrição
Primeiro bit	0	Marcado como 0 (Reservado para uso futuro)
R	1	Endereço de um Ponto de Encontro (*Rendezvous Point*)
R	0	Não representa um endereço de Ponto de Encontro
P	1	Endereço *multicast* baseado no prefixo da rede
P	0	Endereço *multicast* não baseado no prefixo da rede
T	1	Endereço *multicast* temporário (não alocado pela IANA)
T	0	Endereço *multicast* permanente (alocado pela IANA)

- **Escopo**

Valor (4 bits hex)	Descrição
1	Interface
2	Enlace
3	Sub-rede
4	Admin
5	Site
8	Organização
E	Global
(0, F)	Reservados
(6, 7, 9, A, B, C, D)	Não-alocados

65

cgi.br

As *flags* são definidas da seguinte forma:

– O primeiro bit mais a esquerda é reservado e deve ser marcado com 0;

– ***Flag* R**: Se o valor for 1, indica que o endereço *multicast* "carrega" o endereço de um Ponto de Encontro (*Rendezvous Point*). Se o valor for 0, indica que não há um endereço de Ponto de Encontro embutido;

– ***Flag* P**: Se o valor for 1, indica que o endereço *multicast* é baseado em um prefixo de rede. Se o valor for 0, indica que o endereço não é baseado em um prefixo de rede;

– ***Flag* T**: Se o valor for 0, indica que o endereço *multicast* é permanente, ou seja, é atribuído pela IANA. Se o valor for 1, indica que o endereço *multicast* não é permanente, ou seja, é atribuído dinamicamente.

Os quatro bits que representam o escopo do endereço *multicast*, são utilizados para delimitar a área de abrangência de um grupo *multicast.* Os valores atribuídos a esse campo são o seguinte:

– 1 - abrange apenas a interface local;

– 2 - abrange os nós de um enlace;

– 3 - abrange os nós de uma sub-rede

– 4 - abrange a menor área que pode ser configurada manualmente;

– 5 - abrange os nós de um site;

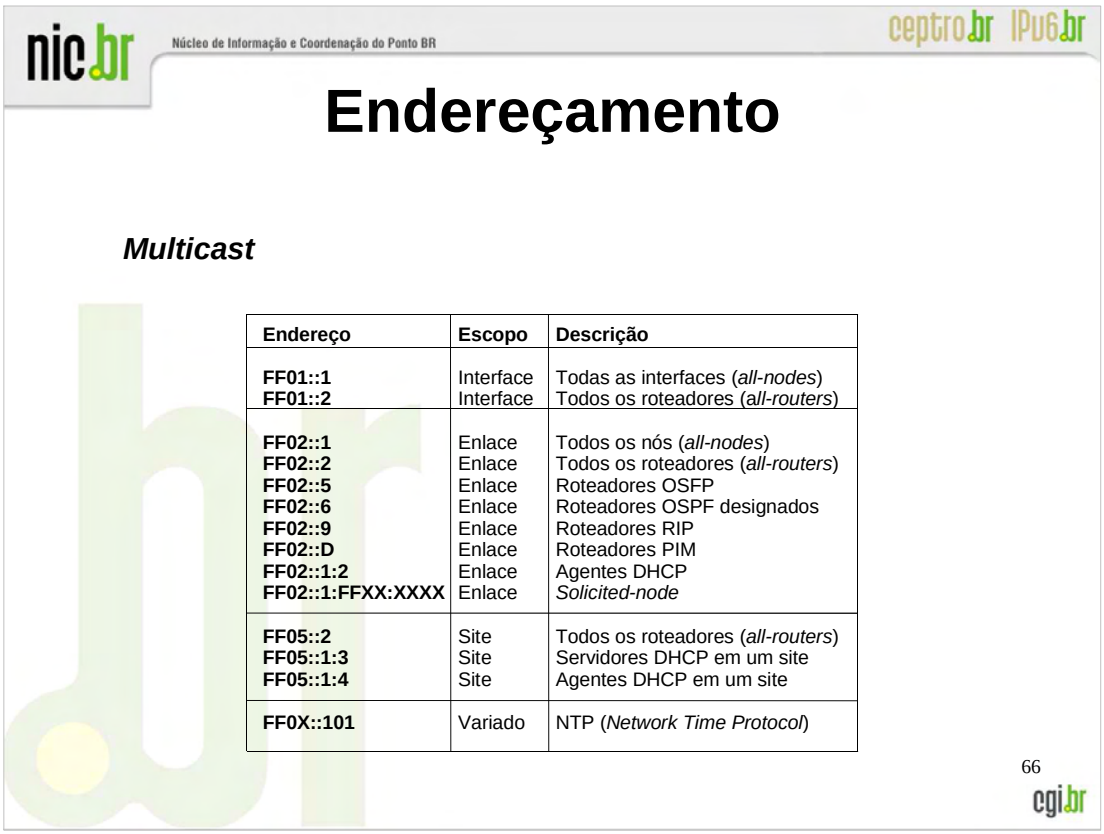
– 8 - abrange vários sites de uma mesma organização;

– E - abrange toda a Internet;

– 0, F - reservados;

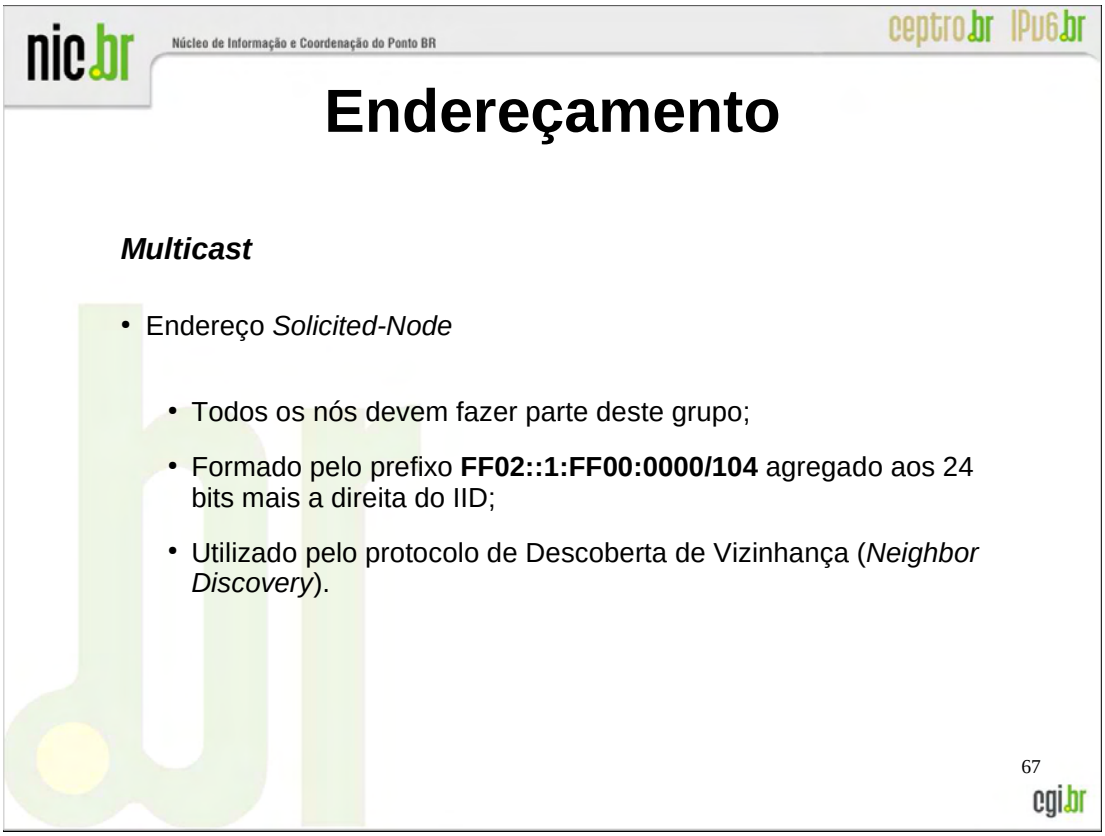
– 6, 7, 9, A, B, C, D - não estão alocados.

Deste modo, um roteador ligado ao *backbone* da Internet não encaminhará pacotes com escopo menor do que 14 (E em hexa), por exemplo. No IPv4, o escopo de um grupo *multicast* é especificado através do campo TTL do cabeçalho.



A lista abaixo apresenta alguns endereços *multicast* permanentes:

Endereço	Escopo	Descrição
FF01::1	Interface	Todas as interfaces em um nó (*all-nodes*)
FF01::2	Interface	Todos os roteadores em um nó (*all-routers*)
FF02::1	Enlace	Todos os nós do enlace (*all-nodes*)
FF02::2	Enlace	Todos os roteadores do enlace (*all-routers*)
FF02::5	Enlace	Roteadores OSFP
FF02::6	Enlace	Roteadores OSPF designados
FF02::9	Enlace	Roteadores RIP
FF02::D	Enlace	Roteadores PIM
FF02::1:2	Enlace	Agentes DHCP
FF02::1:FFXX:XXXX	Enlace	*Solicited-node*
FF05::2	Site	Todos os roteadores em um site
FF05::1:3	Site	Servidores DHCP em um site
FF05::1:4	Site	Agentes DHCPem um site
FF0X::101	Variado	NTP (*Network Time Protocol*)



O endereço *multicast solicited-node* identifica um grupo *multicast* que todos os nós passam a fazer parte assim que um endereço *unicast* ou *anycast* lhes é atribuído. Um endereço *solicited-node* é formado agregando-se ao prefixo **FF02::1:FF00:0000/104** os 24 bits mais a direita do identificador da interface, e para cada endereço *unicast* ou *anycast* do nó, existe um endereço *multicast solicited-node* correspondente.

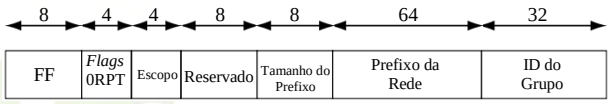
Em redes IPv6, o endereço *solicited-node* é utilizado pelo protocolo de Descoberta de Vizinhança para resolver o endereço MAC de uma interface. Para isso, envia-se uma mensagem *Neighbor Solicitation* para o endereço *solicited-node*. Com isso, apenas as interfaces registradas neste grupo examinam o pacote. Em uma rede IPv4, para se determinar o endereço MAC de uma interface, envia-se uma mensagem *ARP Request* para o endereço *broadcast* da camada de enlace, de modo que todas as interfaces do enlace examinam a mensagem.

nic.br Núcleo de Informação e Coordenação do Ponto BR — ceptro.br IPv6.br

Endereçamento

Multicast

- Endereço *multicast* derivado de um prefixo *unicast*



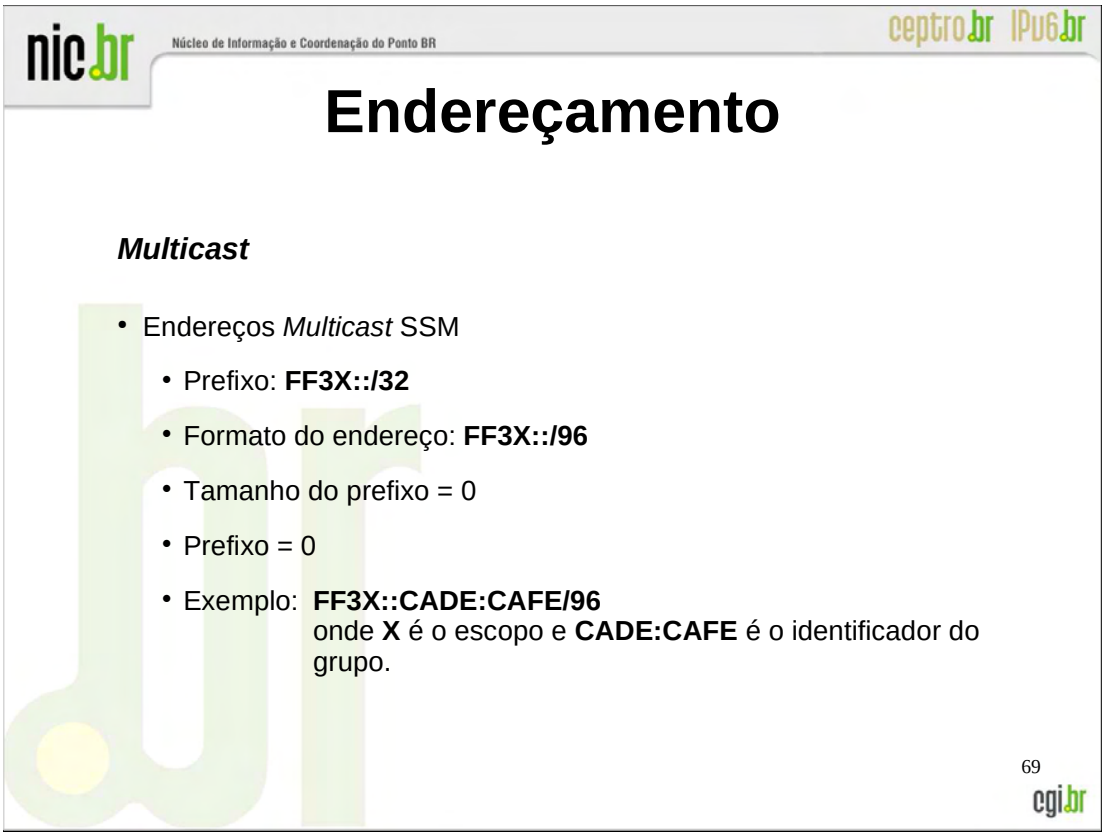
  - *Flag* P = 1
  - *Flag* T = 1
    - Prefixo **FF30::/12**
  - Exemplo:
    prefixo da rede = **2001:DB8::/32**
    endereço = **FF3E:20:2001:DB8:0:0:CADE:CAFE**

68

cgi.br

Com o intuito de reduzir o número de protocolos necessários para a alocação de endereços *multicast*, foi definido um formato estendido de endereço *multicast*, que permite a alocação de endereços baseados em prefixos *unicast* e de endereços SSM (*source-specific multicast*).

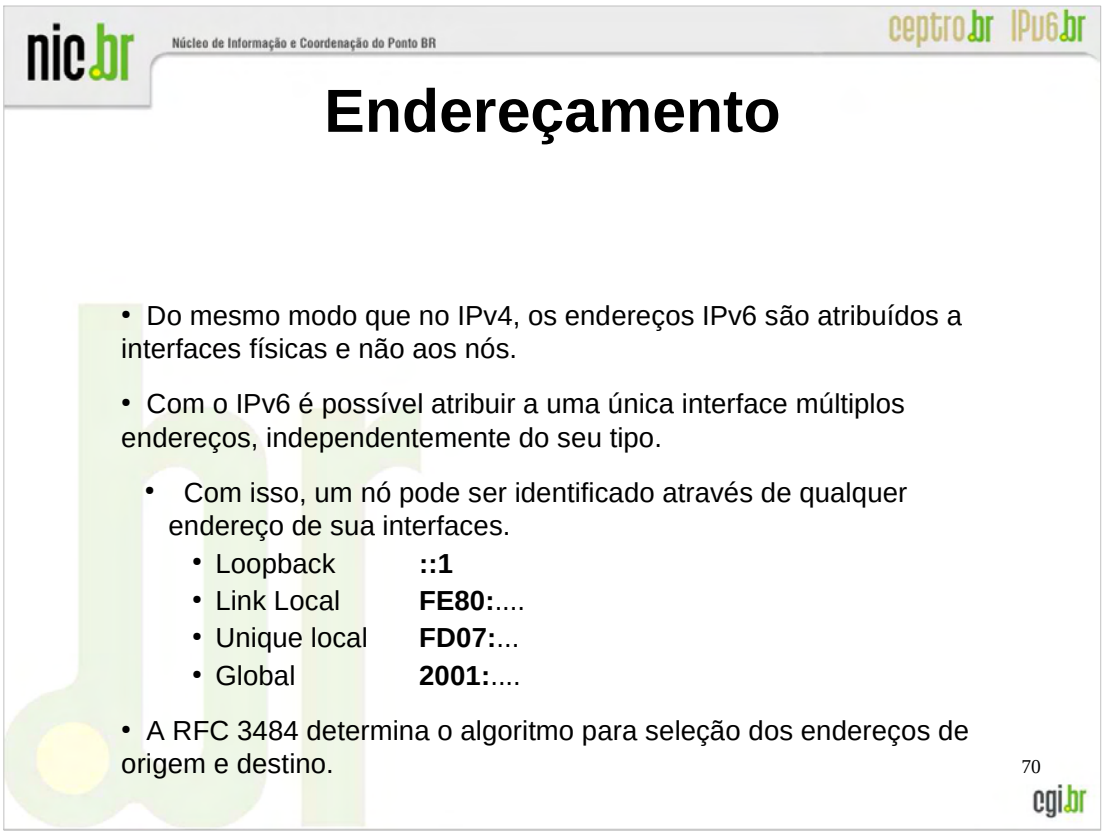
Em endereços baseados no prefixo da rede, a *flag* P é marcada com o valor 1. Neste caso, o uso do campo escopo não altera, porém, o escopo deste endereço *multicast* não deve exceder o escopo do prefixo *unicast* "carregado" junto a ele. Os 8 bits após o campo escopo, são reservados e devem ser marcados com zeros. Na sequência, há 8 bits que especificam o tamanho do prefixo da rede indicado nos 64 bits que os seguem. Caso o prefixo da rede seja menor que 64 bits, os bits não utilizados no campo tamanho do prefixo, devem ser marcados com zeros. O campo identificador do grupo utiliza os 32 bits restantes. Note que, em um endereço onde a *flag* P é marcada com o valor 1, a *flag* T também deve ser marcada com o valor 1, pois este não representa um endereço definido pela IANA.



No modelo tradicional de *multicast*, chamado de *any-source multicast* (ASN), o participante de um grupo *multicast* não controla de que fonte deseja receber os dados. Com o SSM, uma interface pode registrar-se em um grupo *multicast* e especificar as fontes de dados. O SSM pode ser implementado utilizando o protocolo MLDv2 (*Multicast Listener Discovery version 2*).

Para um endereço SSM, as *flags* P e T são marcadas com o valor 1. Os campos tamanho do prefixo e o prefixo da rede são marcados com zeros, chegando ao prefixo **FF3X::/32**, onde **X** é o valor do escopo. O campo Endereço de Origem do cabeçalho IPv6 identifica o dono do endereço *multicast*. Todo endereço SSM tem o formato **FF3X::/96**.

Os métodos de gerenciamento dos grupos *multicast* serão abordados no próximo módulo deste curso.



Também é importante destacar algumas características relacionadas ao endereço apresentadas pela nova arquitetura do protocolo IPv6. Assim como no IPv4, os endereços IPv6 são atribuídos às interfaces físicas, e não aos nós, de modo que cada interface precisa de pelo menos um endereço *unicast*. No entanto, é possível atribuir a uma única interface múltiplos endereços IPv6, independentemente do tipo (*unicast*, *multicast* ou *anycast*) ou sub-tipo (*loopback*, *link local*, 6to4, etc.). Deste modo um nó pode ser identificado através de qualquer endereço das suas interfaces, e com isso, torna-se necessário escolher entre seus múltiplos endereços qual utilizará como endereço de origem e destino ao estabelecer uma conexão.

Para resolver esta questão, foram definidos dois algoritmos, um para selecionar o endereço de origem e outro para o de destino. Esses algoritmos, que devem ser implementados por todos os nós IPv6, especificam o comportamento padrão desse nós, porém não substituem as escolhas feitas por aplicativos ou protocolos da camada superior.

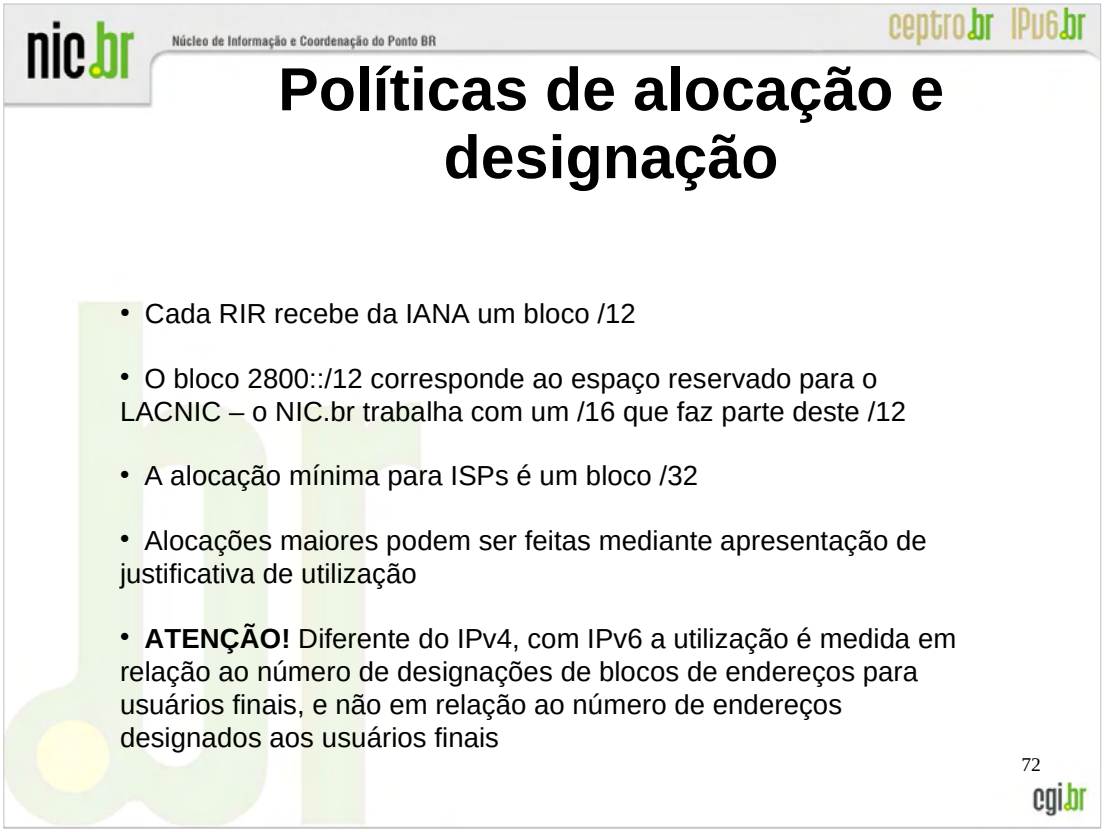
Entre as regras mais importantes destacam-se:

- Pares de endereços do mesmo escopo ou tipo têm preferência;
- O menor escopo para endereço de destino tem preferência (utiliza-se o menor escopo possível);
- Endereços cujo tempo de vida não expirou tem preferência sobre endereços com tempo de vida expirado;
- Endereços de técnicas de transição (ISATAP, 6to4, etc.) não podem ser utilizados se um endereço IPv6 nativo estiver disponível;
- Se todos os critérios forem similares, pares de endereços com o maior prefixo comum terão preferência;
- Para endereços de origem, endereços globais terão preferência sobre endereços temporários;
- Em um Nó Móvel, o Endereço de Origem tem preferência sobre um Endereço Remoto.

Estas regras devem ser utilizadas quando não houver nenhuma outra especificação. As especificações também permitem a configuração de políticas que possam substituir esses padrões de preferências com combinações entre endereços de origem e destino.

Mais informações:

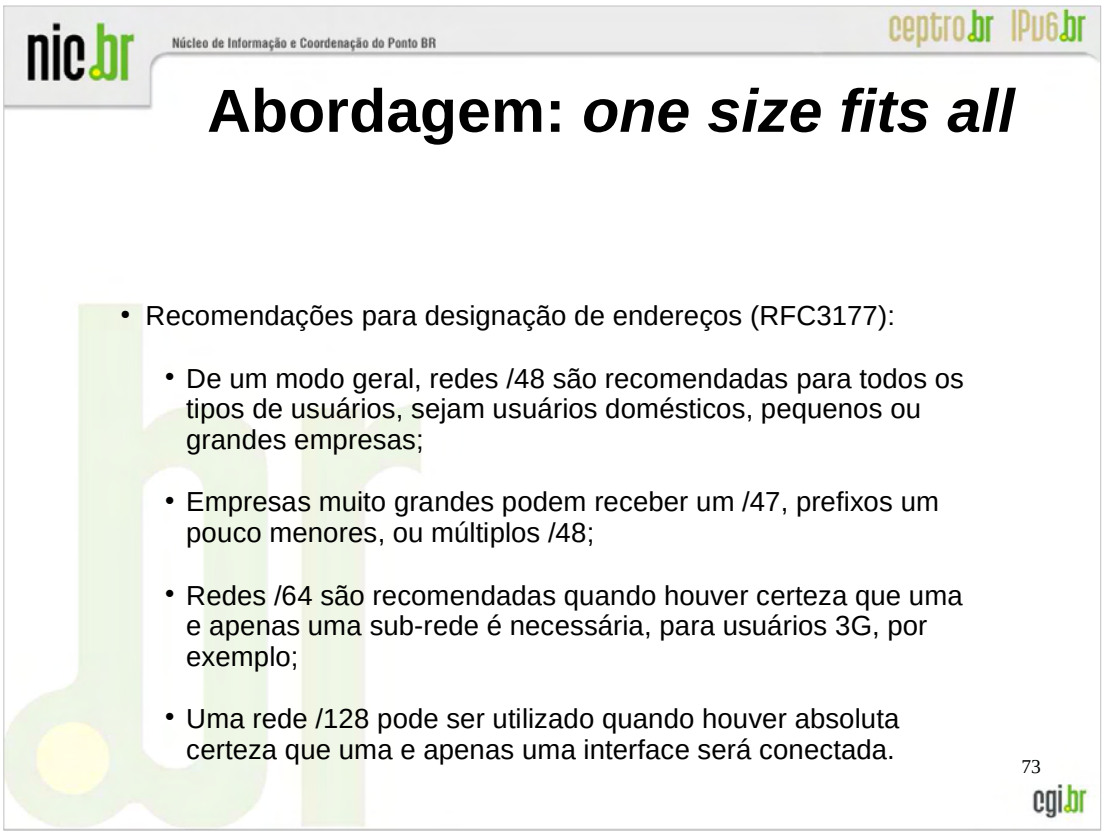
- RFC 2375 - *IPv6 Multicast Address Assignments*
- RFC 3306 - *Unicast-Prefix-based IPv6 Multicast*
- RFC 3307 - *Allocation Guidelines for IPv6 Multicast Addresses*
- RFC 3484 - *Default Address Selection for Internet Protocol version 6* (IPv6)
- RFC 3569 - *An Overview of Source-Specific Multicast* (SSM)
- RFC 3956 - *Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address*
- RFC 4007 - *IPv6 Scoped Address Architecture*
- RFC 4489 - *A Method for Generating Link-Scoped IPv6 Multicast Addresses*
- *Internet Protocol Version 6 Multicast Addresses* - http://www.iana.org/assignments/ipv6-multicast-addresses/



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

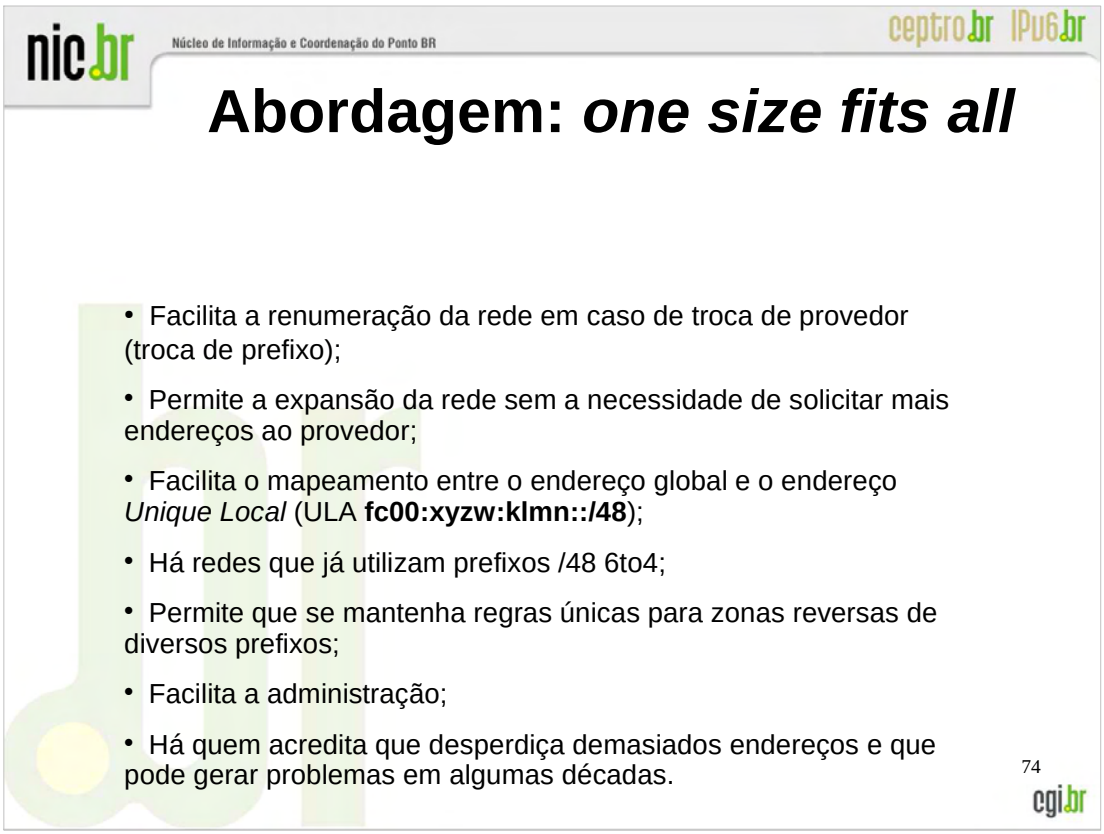
O bloco **2800::/12** corresponde ao espaço reservado para o LACNIC alocar na América Latina. O NIC.br por sua vez, trabalha com um /16 que faz parte deste /12.

A alocação mínima para ISPs é um bloco /32, no entanto, alocações maiores podem ser feitas mediante apresentação de justificativa de utilização. Um aspecto importante que merece destaque é que diferente do IPv4, com IPv6 a utilização é medida em relação ao número de designações de blocos de endereços para usuários finais, e não em relação ao número de endereços designados aos usuários finais.



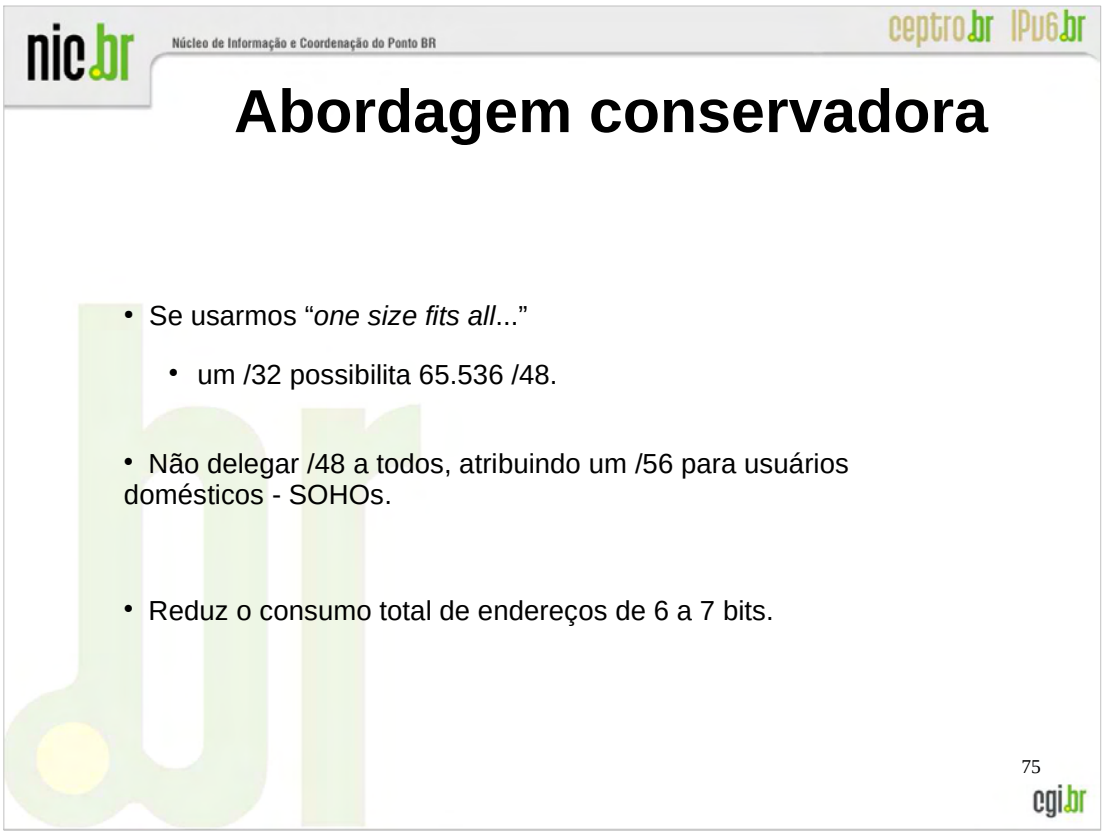
Em relação a alocação e designação de endereços a usuários finais, a RFC 3177 recomenda que seja seguida uma abordagem conhecida como *one size fits all*, que apresenta as seguintes características:

- De um modo geral, redes /48 são recomendadas para todos os tipos de usuários, sejam usuários domésticos, pequenos ou grandes empresas;
- Empresas muito grandes podem receber um /47, prefixos um pouco menores, ou múltiplos /48;
- Redes /64 são recomendadas quando houver certeza que uma e apenas uma sub-rede é necessária, para usuários 3G, por exemplo;
- Uma rede /128 pode ser utilizado quando houver absoluta certeza que uma e apenas uma interface será conectada. Ex.: conexões PPoE.



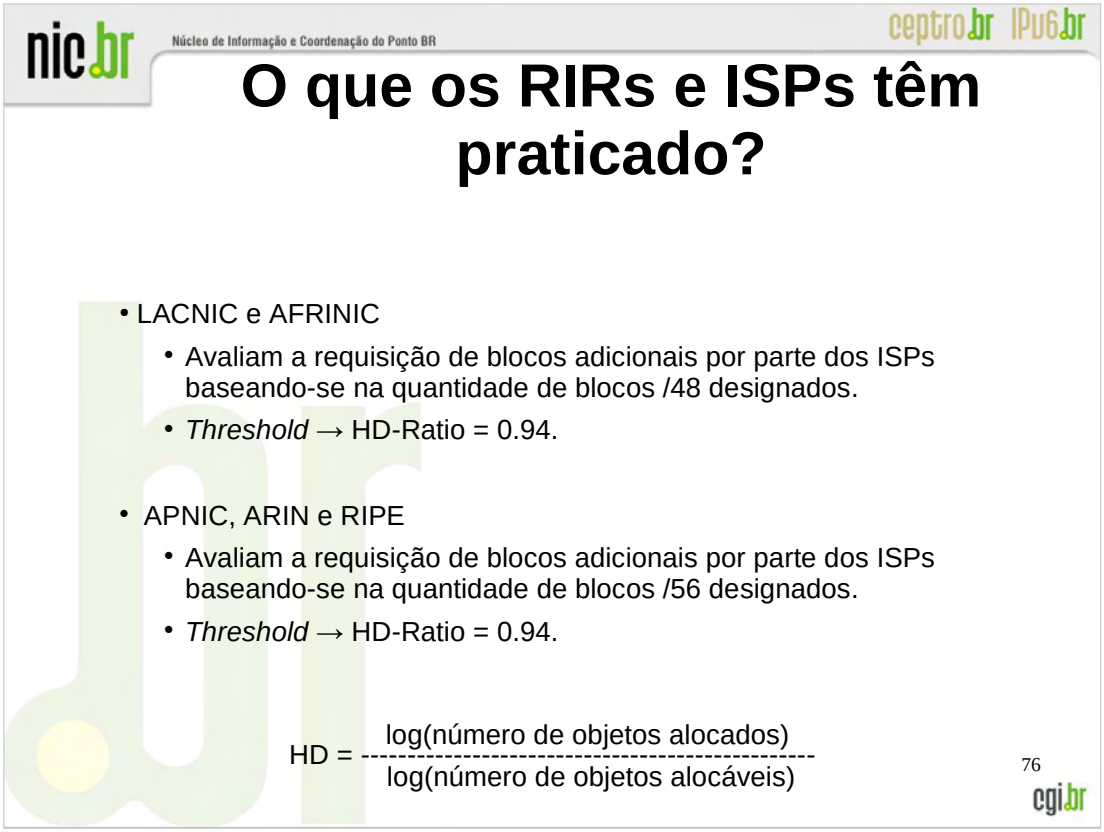
A abordagem *one size fits all* apresenta algumas vantagens:

- Facilita a renumeração da rede em caso de troca de provedor (troca de prefixo);
- Permite a expansão da rede sem a necessidade de solicitar mais endereços ao provedor;
- Facilita o mapeamento entre o endereço global e o endereço *Unique Local* (ULA **fc00:xyzw:klmn::/48**);
- Há redes que já utilizam prefixos /48 6to4;
- Permite que se mantenha regras únicas para zonas reversas de diversos prefixos;
- Facilita a administração;
- Há quem acredita que desperdiça demasiados endereços e que pode gerar problemas em algumas décadas.



Uma abordagem mais conservadora, oposta a *one size fits all*, recomenda que não sejam delegados /48 a todo tipo de usuário, atribuindo /56 para usuários domésticos e SOHO. Deste modo, se reduz o consumo total de endereços de 6 a 7 bits.

Além disso, um /32 possibilita “apenas” 65,536 /48, que no caso de grandes provedores, não seria suficiente para atender toda a sua demanda.



Entre os RIRs, temos duas políticas distintas sendo praticadas em relação a recomendação de uso aos ISPs e ao critério para alocação de blocos de endereços adicionais.

Os RIRs LACNIC e AFRINIC, seguem a recomendação *one size fits all*, sugerindo que os provedores de suas regiões também o sigam. A avaliação das requisições de blocos adicionais por parte dos ISPs também segue essa abordagem, baseando-se na quantidade de blocos /48 designados por eles.

Já os RIRs APNIC, ARIN e RIPE, seguem uma abordagem mais conservadora, utilizando a quantidade de blocos /56 designados pelos provedores como base para avaliação das requisições de blocos adicionais.

Em todos os casos, é utilizada como medida para avaliação o valor do HD-Ratio (*Host-Density ratio*). O HD-Ratio é um modo de medir o uso do espaço de endereçamento, onde seu valor é relacionado a porcentagem de uso. A fórmula para se calcular o HD-Ratio é a seguinte:

$$HD = \frac{\log(\text{número de objetos alocados})}{\log(\text{número de objetos alocáveis})}$$

Todos os RIRs utilizam como valor de *Threshold* (limite) o HD-Ratio = 0,94, mas o LACNIC e o AFRINIC sobre o a utilização de blocos /48, e o APNIC, ARIN e o RIPE sobre a utilização de blocos /56.

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

O que os RIRs e ISPs têm praticado?

Bloco	Qtd. /48	Threshold (HD=0,94)	% de Utilização
/32	65.536	33.689	51,41%
/31	131.072	64.634	49,31%
/30	262.144	124.002	47,30%
/29	524.288	237.901	45,38%
/28	1.048.576	456.419	43,53%
/27	2.097.152	875.653	41,75%
/26	4.194.304	1.679.965	40,05%
/25	8.388.608	3.223.061	38,42%
/24	16.777.216	6.183.533	36,86%
/23	33.554.432	11.863.283	35,36%
/22	67.108.864	22.760.044	33,92%
/21	134.217.728	43.665.787	32,53%
/20	268.435.456	83.774.045	31,21%

77

cgi.br

Esta tabela apresenta a porcentagem de utilização de blocos /48 baseando-se no cálculo do HD-Ratio igual a 0,94.

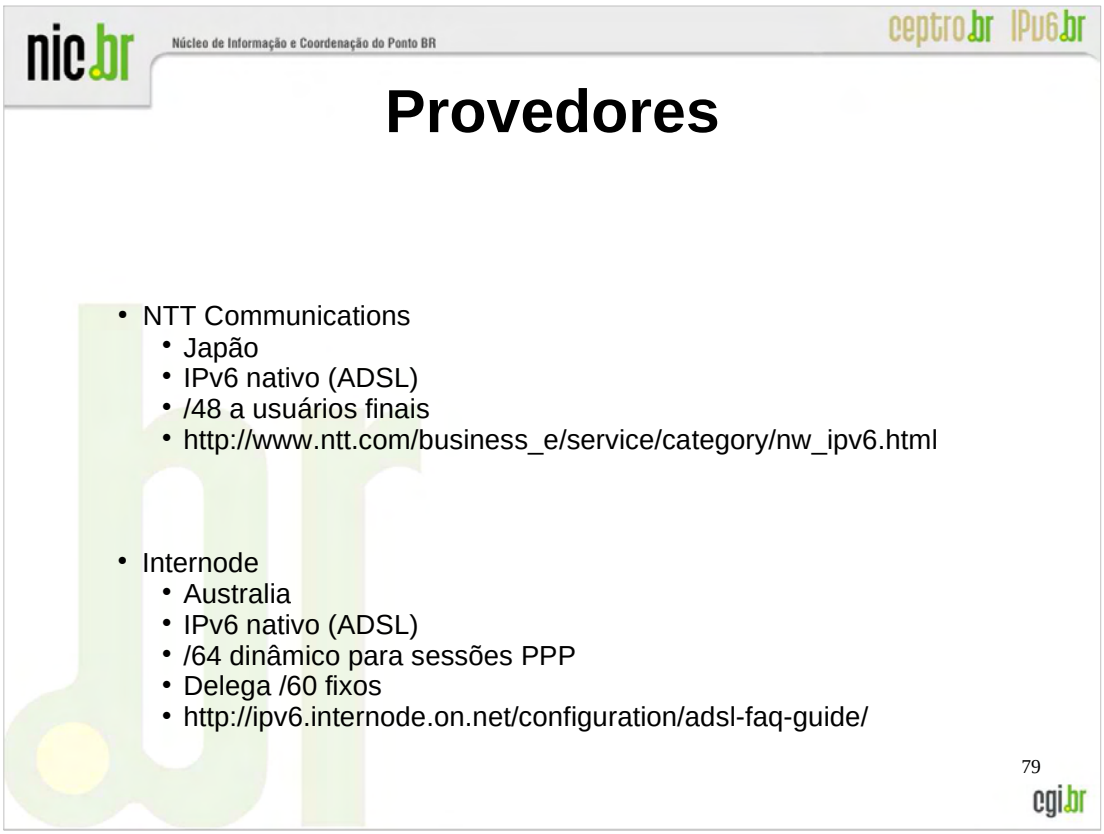
nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

O que os RIRs e ISPs têm praticado?

Bloco	Qtd. /56	Threshold (HD=0,94)	% de Utilização
/32	16.777.216	6.183.533	36,86%
/31	33.554.432	11.863.283	35,36%
/30	67.108.864	22.760.044	33,92%
/29	134.217.728	43.665.787	32,53%
/28	268.435.456	83.774.045	31,21%
/27	536.870.912	160.722.871	29,94%
/26	1.073.741.824	308.351.367	28,72%
/25	2.147.483.648	591.580.804	27,55%
/24	4.294.967.296	1.134.964.479	26,43%
/23	8.589.934.592	2.177.461.403	25,35%
/22	17.179.869.184	4.177.521.189	24,32%
/21	34.359.738.368	8.014.692.369	23,33%
/20	68.719.476.736	15.376.413.635	22,38%

cgi.br

Esta tabela apresenta a porcentagem de utilização de blocos /56 baseando-se no calculo do HD-Ratio igual a 0,94.



Em relação a política seguida por alguns provedores no mundo que já fornecem a seus clientes endereços IPv6, temos algumas visões diferentes. Analise os seguinte exemplos:

- NTT Communications
  - Japão
  - IPv6 nativo (ADSL)
  - /48 a usuários finais
  - http://www.ntt.com/business_e/service/category/nw_ipv6.html
- Internode
  - Australia
  - IPv6 nativo (ADSL)
  - /64 dinâmico para sessões PPP
  - Delega /60 fixos
  - http://ipv6.internode.on.net/configuration/adsl-faq-guide/
- IIJ
  - Japão
  - Túneis
  - /48 a usuários finais
  - http://www.iij.ad.jp/en/service/IPv6/index.html
- Arcnet6
  - Malásia
  - IPv6 nativo (ADSL) ou Túneis
  - /48 a usuários finais
  - /40 e /44 podem ser alocados (depende de aprovação)
  - http://arcnet6.net.my/how.html

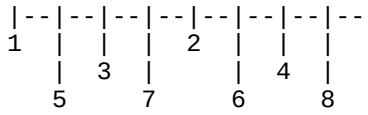
Provedores

- IIJ
  - Japão
  - Túneis
  - /48 a usuários finais
  - http://www.iij.ad.jp/en/service/IPv6/index.html
- Arcnet6
  - Malásia
  - IPv6 nativo (ADSL) ou Túneis
  - /48 a usuários finais
  - /40 e /44 podem ser alocados (depende de aprovação)
  - http://arcnet6.net.my/how.html

nic.br Núcleo de Informação e Coordenação do Ponto BR — ceptro.br IPv6.br

Considerações

- /32 =
  - 65 mil redes /48 (33 mil, se considerarmos desperdício)
  - 16 milhões de redes /56 (6 milhões, se cons. hd ratio)
  - é suficiente para seu provedor?
  - Reservar um bloco (/48 ?) para infraestrutura…
- Links ponto a ponto:
  - /64? /112? /120? /126? /127?
- RFC 3531



81

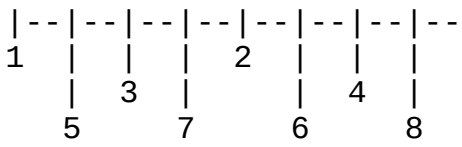
egi.br

Antes de solicitar um bloco de endereços IPv6 é preciso elaborar cuidadosamente o plano de endereçamento da rede. Neste momento, alguns aspectos importantes precisam ser considerados:

Um bloco /32 permite 65 mil redes /48, ou 33 mil, se considerarmos desperdício; 16 milhões de redes /56, ou 6 milhões, se considerarmos o HD-ratio. Cada provedor deverá analisar se esses valores atendem as suas necessidades, ou se será necessário solicitar ao Registro de Internet um bloco de endereços maior.

Também é preciso considerar os endereços que serão utilizados para infraestrutura, qual tamanho de bloco será reservado para as redes internas, para *links* ponto-a-ponto, etc.

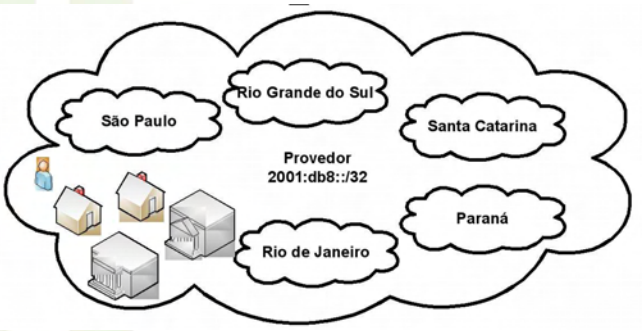
A RFC 3531 propõe um método para gerenciar a atribuição dos bits de um bloco de endereços IPv6 ou de um prefixo, de modo que estes sejam alocados reservando um espaço maior entre eles. Com isso, caso seja necessário alocar novos blocos a um usuário, a probabilidade de se alocar blocos contíguos ao que ele já possui será maior.



nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Exercício de endereçamento IPv6

- Você é um provedor e recebeu o bloco **2001:0db8::/32**
- Você está presente em várias localidades (5 estados diferentes) e tem planos de expansão.
- Você atende a usuários domésticos, a pequenas, médias e grandes empresas.



82

cgi.br

Será proposto agora um exercício rápido de planejamento de endereçamento:

- Você é um provedor e recebeu o bloco **2001:0db8::/32**
- Você está presente em várias localidades (5 estados diferentes) e tem planos de expansão.
- Você atende a usuários domésticos, a pequenas, médias e grandes empresas.

Trabalhe com os seguintes pontos:

(1) Você decidiu que a melhor forma de dividir os endereços é hierarquicamente... Qual o tamanho do bloco de cada estado?

(2) Qual o tamanho do bloco a ser designado para cada tipo de usuário?

(3) Quantos usuários de cada tipo poderão ser atendidos dessa forma?

(4) Indique o bloco correspondente a cada localidade.

(5) Escolha uma localidade e indique os blocos correspondentes a cada tipo de usuário

(6) Nessa mesma localidade, indique o primeiro e o segundo blocos designados para cada tipo de usuário (os 2 primeiros usuários de cada tipo)

(7) Para o segundo bloco/usuário de cada tipo, indique o primeiro e o último endereços.

Exercício de endereçamento IPv6

(1) Você decidiu que a melhor forma de dividir os endereços é hierarquicamente... Qual o tamanho do bloco de cada estado?

(2) Qual o tamanho do bloco a ser designado para cada tipo de usuário?

(3) Quantos usuários de cada tipo poderão ser atendidos dessa forma?

(4) Indique o bloco correspondente a cada localidade.

(5) Escolha uma localidade e indique os blocos correspondentes a cada tipo de usuário

(6) Nessa mesma localidade, indique o primeiro e o segundo blocos designados para cada tipo de usuário (os 2 primeiros usuários de cada tipo)

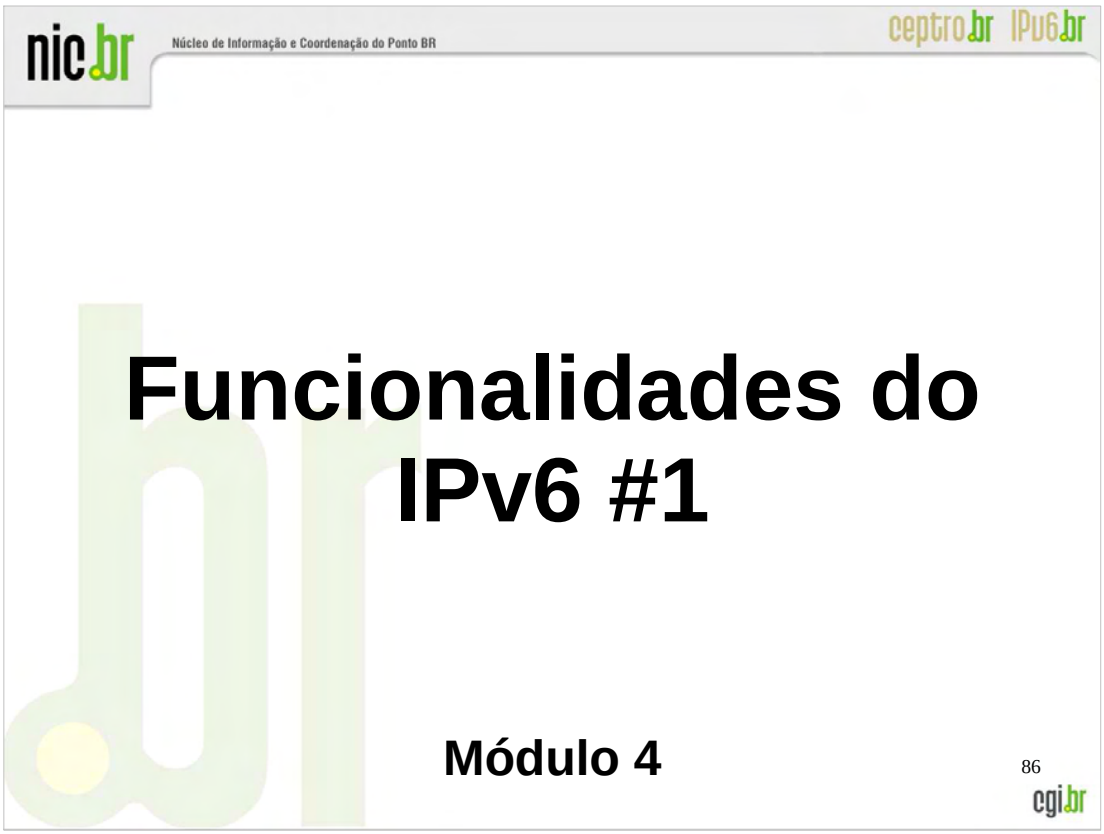
(7) Para o segundo bloco/usuário de cada tipo, indique o primeiro e o último endereços.

Mais Informações:

- RFC 5375 - *IPv6 Unicast Address Assignment Considerations*
- RFC 3177 - IAB/IESG Recommendations on IPv6 Address Allocations to Sites
- RFC 3531 - *A Flexible Method for Managing the Assignment of Bits of an IPv6 Address Block*
- RFC 3627 - *Use of /127 Prefix Length Between Routers Considered Harmful*
- RFC 3194 - *The Host-Density Ratio for Address Assignment Efficiency: An update on the HD ratio*
- RFC 4692 - *Considerations on the IPv6 Host Density Metric*
- http://www.potaroo.net/ispcol/2005-07/ipv6size.html
- http://www.lacnic.net/en/politicas/manual12.html
- http://tools.ietf.org/html/draft-narten-ipv6-3177bis-48boundary-04
- https://www.arin.net/policy/proposals/2005_8.html
- http://www.apnic.net/policy/ipv6-address-policy#2.7
- http://www.ripe.net/ripe/docs/ipv6-sparse.html
- http://www.ipv6book.ca/allocation.html
- http://tools.ietf.org/html/draft-kohno-ipv6-prefixlen-p2p-00
- http://www.swinog.ch/meetings/swinog18/swissix_swinog_18.pdf
- https://www.arin.net/participate/meetings/reports/ARIN_XXIV/PDF/wednesday/ipv6_implementation_fundamentals.pdf
- http://www.6deploy.org/workshops/20090921_bogota_colombia/Consulintel_IPv6_3-Direccionamiento_IPv6.pdf

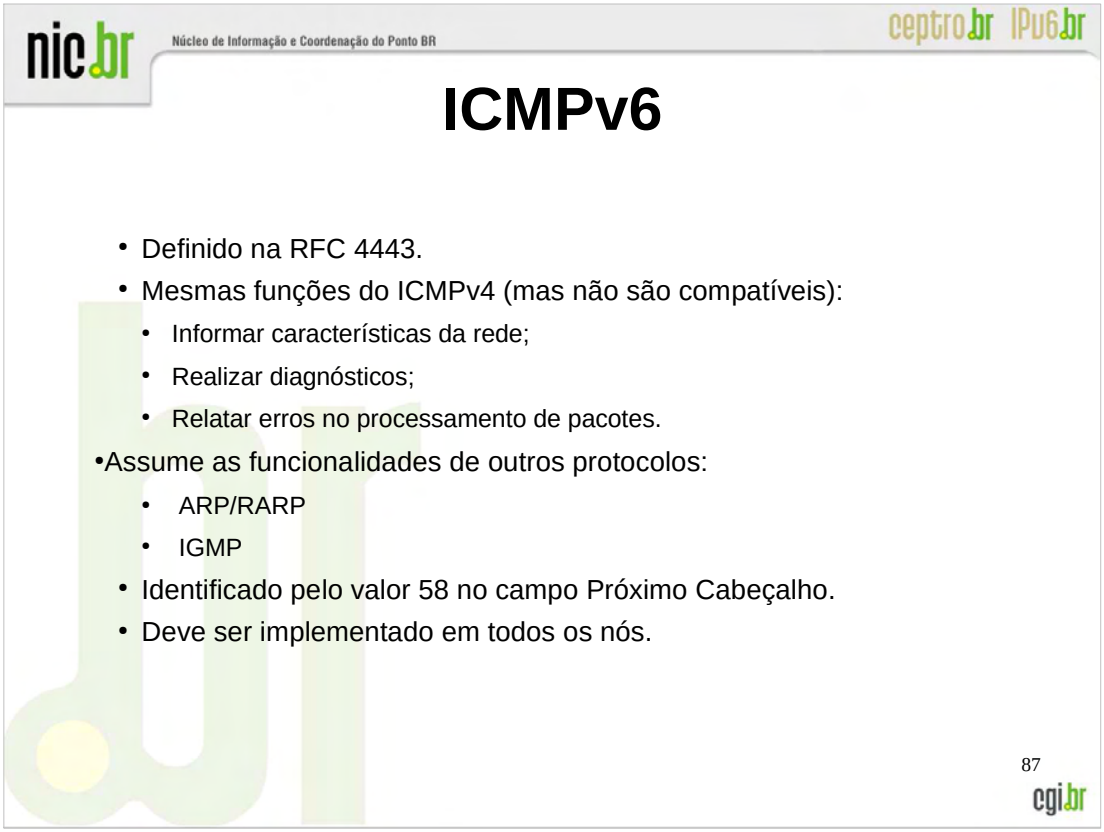
IPv6.br

A Nova Geração do Protocolo Internet



O protocolo IPv6 apresenta uma série de novas funcionalidades e outras aprimoradas em relação ao IPv4.

Na primeira parte deste módulo, conheceremos um pouco mais sobre essas funcionalidades, começando pelo estudo do protocolo ICMPv6 (*Internet Control Message Protocol version 6*), peça fundamental para a execução de ferramentas como o protocolo de Descoberta de Vizinhança (*Neighbor Discovery*) e o mecanismos de autoconfiguração *stateless*, também tratados aqui. Por analisaremos o funcionamento do protocolo DHCPv6 e como estas funcionalidades podem auxiliar no trabalho de renumeração de redes.



Definido na RFC 4443 para ser utilizado com o IPv6, o ICMPv6 é uma versão atualizada do ICMP (*Internet Control Message Protocol*) utilizado com IPv4.

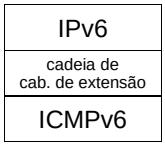
Esta nova versão do ICMP, embora apresente as mesma funções que o ICMPv4, como reportar erros no processamento de pacotes e enviar mensagens sobre o status a as características da rede, ela não é compatível com seu antecessor, apresentando agora um número maior de mensagens e funcionalidades.

O ICMPv6, é agora o responsável por realizar as funções dos protocolos ARP (*Address Resolution Protocol*), que mapeia os endereços da camada dois para IPs e vice-versa no IPv4, e do IGMP (*Internet Group Management Protocol*), que gerencia os membros dos grupos *multicast* no IPv4.

O valor no campo Próximo Cabeçalho, que indica a presença do protocolo ICMPv6, é 58, e o suporte a este protocolo deve ser implementado em todos os nós.

ICMPv6

- É precedido pelos cabeçalhos de extensão, se houver, e pelo cabeçalho base do IPv6.



- Protocolo chave da arquitetura IPv6.
- Essencial em funcionalidades do IPv6:
  - Gerenciamento de grupos *multicast*;
  - Descoberta de Vizinhança (*Neighbor Discovery*);
  - Mobilidade IPv6;
  - Descoberta do *Path* MTU.

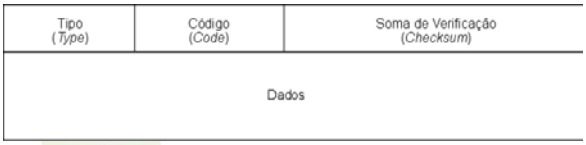
Em um pacote IPv6, o ICMPv6 posiciona-se logo após o cabeçalho base do IPv6, e dos cabeçalhos de extensão, se houver.

O ICMPv6, é um protocolo chave na arquitetura IPv6, visto que, além do gerenciamento dos grupos *multicast*, através do protocolo MLD (*Multicast Listener Discovery*), e da resolução de endereços da camada dois, suas mensagens são essenciais para o funcionamento do protocolo de Descoberta de Vizinhança (*Neighbor Discovery*), responsável por localizar roteadores vizinhos na rede, detectar mudanças de endereço no enlace, detectar endereços duplicados, etc.; no suporte à mobilidade, gerenciando Endereços de Origem dos *hosts* dinamicamente; e no processo de descoberta do menor MTU (*Maximum Transmit Unit*) no caminho de uma pacote até o destino.

nic.br Núcleo de Informação e Coordenação do Ponto BR — ceptro.br IPv6.br

ICMPv6

- Cabeçalho simples

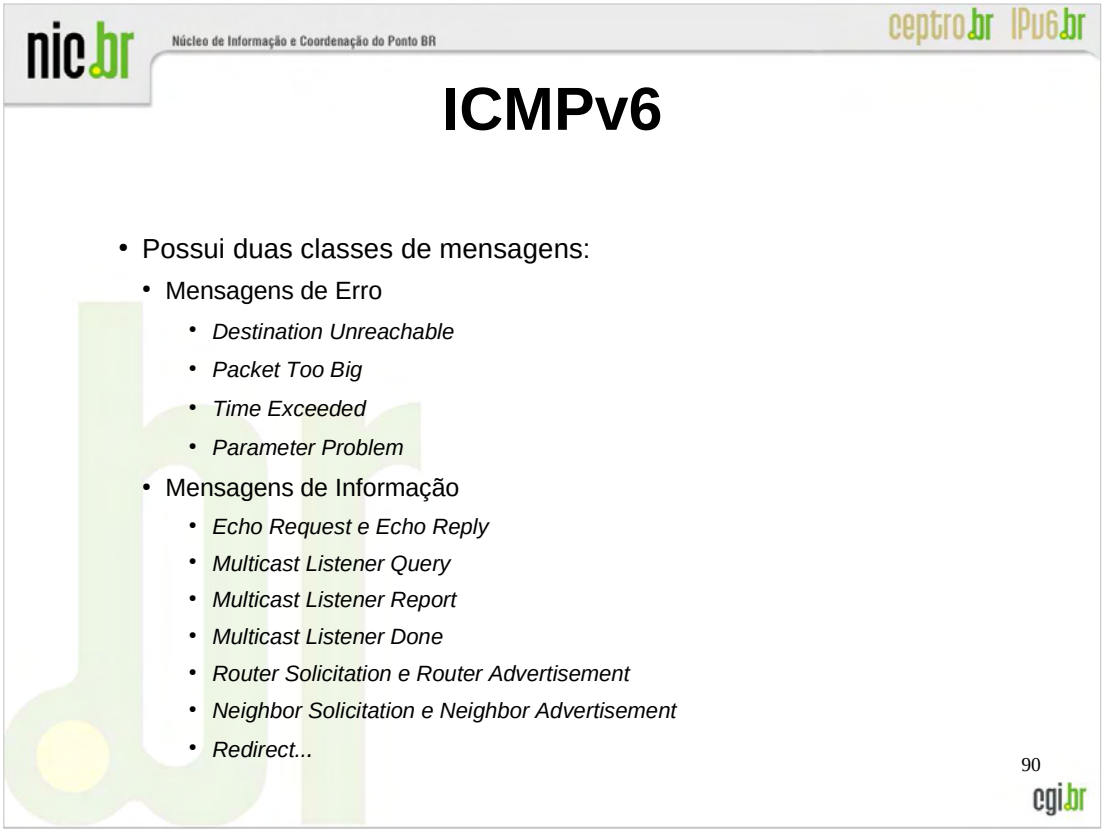


- **Tipo** (8 bits): especifica o tipo da mensagem.
- **Código** (8 bits): oferece algumas informações adicionais para determinados tipos de mensagens.
- **Soma de Verificação** (16 bits): é utilizado para detectar dados corrompidos no cabeçalho ICMPv6 e em parte do cabeçalho IPv6.
- **Dados:** apresenta as informações de diagnóstico e erro de acordo com o tipo de mensagem. Seu tamanho pode variar de acordo com a mensagem.

cgi.br

O cabeçalho de todas as mensagens ICMPv6 tem a mesma estrutura simples, sendo composto por quatro campos:

- **Tipo:** especifica o tipo da mensagem, o que determinará o formato do corpo da mensagem. Seu tamanho é de oito bits;
- **Código:** oferece algumas informações adicionais para determinados tipos de mensagens. Também possui oito bits de tamanho;
- **Soma de Verificação:** é utilizado para detectar dados corrompidos no cabeçalho ICMPv6 e em parte do cabeçalho IPv6. Seu tamanho é de 16 bits;
- **Dados:** apresenta as informações de diagnóstico e erro de acordo com o tipo de mensagem. Para ajudar na solução de problemas, as mensagens de erro trarão neste campo, o pacote que invocou a mensagem, desde que o tamanho total do pacote ICMPv6 não exceda o MTU mínimo do IPv6, que é 1280 Bytes.



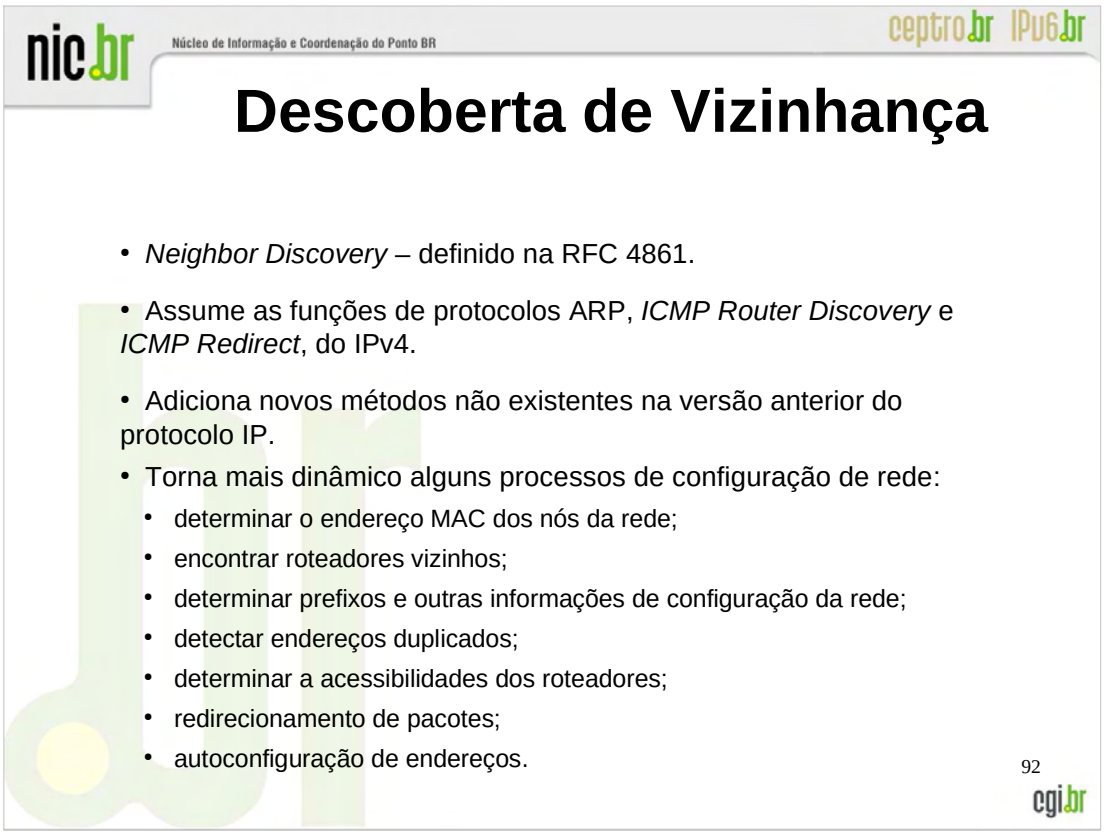
As mensagens ICMPv6 são divididas em duas classes, cada uma composta por diversos tipos de mensagens, conforme as tabelas a seguir:

Mensagens de Erro:

Tipo	Nome	Descrição
1	Destination Unreachable	Indica falhas na entrega do pacote como endereço ou porta desconhecida ou problemas na comunicação.
2	Packet Too Big	Indica que o tamanho do pacote é maior que a Unidade Máxima de Transito (MTU) de um enlace.
3	Time Exceeded	Indica que o Limite de Encaminhamento ou o tempo de remontagem do pacote foi excedido.
4	Parameter Problem	Indica erro em algum campo do cabeçalho IPv6 ou que o tipo indicado no campo Próximo Cabeçalho não foi reconhecido.
100-101		Uso experimental
102-126		Não utilizado
127		Reservado para expansão das mensagens de erro ICMPv6

Mensagens de Informação

Tipo	Nome	Descrição
128 129	Echo Request Echo Reply	Utilizadas pelo comando ping.
130 131 132	Multicast Listener Query Multicast Listener Report Multicast Listener Done	Utilizadas no gerenciamento de grupos *multicast.*
133 134 135 136 137	Router Solicitation Router Advertisement Neighbor Solicitation Neighbor Advertisement Redirect Message	Utilizadas com o protocolo Descoberta de Vizinhança.
138	Router Renumbering	Utilizada no mecanismo de Re-endereçamento (*Renumbering*) de roteadores.
139 140	ICMP Node Information Query ICMP Node Information Response	Utilizadas para descobrir informações sobre nomes e endereços, são atualmente limitadas a ferramentas de diagnóstico, depuração e gestão de redes.
141 142	Inverse ND Solicitation Message Inverse ND Advertisement Message	Utilizadas em uma extensão do protocolo de Descoberta de Vizinhança.
143	Version 2 Multicast Listener Report	Utilizada no gerenciamento de grupos *multicast.*
144 145 146 147	HA Address Discovery Req. Message HA Address Discovery Reply Message Mobile Prefix Solicitation Mobile Prefix Advertisement	Utilizadas no mecanismo de Mobilidade IPv6.
148 149	Certification Path Solicitation Message Cert. Path Advertisement Message	Utilizadas pelo protocolo SEND.
150		Utilizada experimentalmente com protocolos de mobilidade como o *Seamoby*.
151 152 153	Multicast Router Advertisement Multicast Router Solicitation Multicast Router Termination	Utilizadas pelo mecanismo *Multicast Router Discovery*
154	FMIPv6 Messages	Utilizada pelo protocolo de mobilidade *Fast Handovers*
200-201		Uso Experimental
255		Reservado para expansão das mensagens de erro ICMPv6



Definido pela RFC4861, o protocolo de Descoberta de Vizinhança torna mais dinâmicos alguns processos de configuração de rede em relação ao IPv4, combinando as funções de protocolos como ARP, *ICMP Router Discovery* e *ICMP Redirect*, além de adicionar novos métodos não existentes na versão anterior do protocolo IP.

O protocolo de Descoberta de Vizinhança do IPv6 é utilizado por *hosts* e roteadores para os seguintes propósitos:

· determinar o endereço MAC dos nós da rede;

· encontrar roteadores vizinhos;

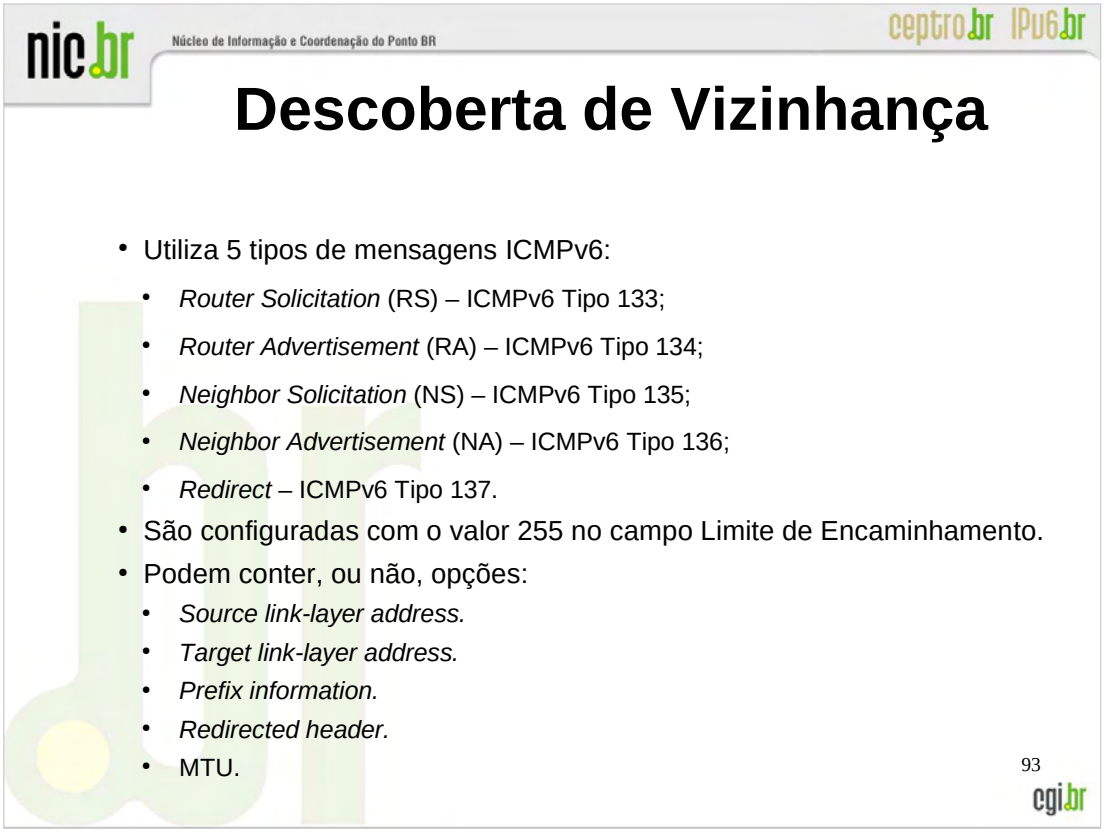
· determinar prefixos e outras informações de configuração da rede;

· detectar endereços duplicados;

· determinar a acessibilidades dos roteadores;

· redirecionamento de pacotes;

· autoconfiguração de endereços.



As mensagens *Neighbor Discovery* são configuradas com um Limite de Encaminhamento de 255 para assegurar que as mensagens recebidas são originadas de um nó do mesmo enlace, descartando as mensagens com valores diferentes.

O *Neighbor Discovery* utiliza cinco mensagens ICMPv6:

- ***Router Solicitation*** **(ICMPv6 tipo 133)**: utilizada por *hosts* para requisitar aos roteadores mensagens *Router Advertisements* imediatamente. Normalmente é enviada para o endereço *multicast* **FF02::2** (*all-routers on link*);
- ***Router Advertisement*** **(ICMPv6 tipo 134)**: enviada periodicamente, ou em resposta a uma *Router Solicitation*, é utilizada pelos roteadores para anunciar sua presença em um enlace. As mensagens periódicas são enviadas para o endereço *multicast* **FF02::1** (*all-nodes on link*) e as solicitadas são enviadas diretamente para o endereço do solicitante. Uma RA carrega diversas informações referentes à configurações da rede como:
  - O valor padrão do enlace para o campo Limite de Encaminhamento;
  - Uma *flag* especificando se deve ser utilizado autoconfiguração *stateless* ou *stateful;*
  - Outra *flag* que especifica se os nós devem utilizar configurações *stateful* para obter outras informações sobre a rede;
  - Uma terceira *flag* é utilizada em redes com suporte à mobilidade IPv6, para indicar se o roteador é um Agente de Origem;

    - Por quanto tempo, em segundos, o roteador será considerado o roteador padrão do enlace. Caso não seja o roteador padrão o valor será zero;
    - O tempo que um *host* pressupõe que os vizinhos são alcançáveis após ter recebido uma confirmação de acessibilidade;
    - O intervalo entre o envio de mensagens *Neighbor Solicitation*.
- ***Neighbor Solicitation* (ICMPv6 tipo 135)**: mensagem *multicast* enviada por um nó para determinar o endereço MAC e a acessibilidade de um vizinho, além de detectar a existência de endereços duplicados. Esta mensagem possui um campo para indicar o endereço de origem da mensagem;
- ***Neighbor Advertisement* (ICMPv6 tipo 136)**: enviada como resposta a uma *Neighbor Solicitation*, pode também ser enviada para anunciar a mudança de algum endereço dentro do enlace. Esta mensagem possui três *flags:*
    - A primeira indica se quem esta enviando a mensagem é um roteador;
    - A segunda indica se a mensagem é uma resposta a uma NS;
    - A terceira indica se a informação carregada na mensagem é uma atualização de endereço de algum nó da rede.
- ***Redirect* (ICMPv6 tipo 137)**: utilizada por roteadores para informar ao *host* o melhor roteador para encaminhar o pacote ao destino. Esta mensagem traz como informação, o endereço do roteador considerado o melhor salto, o endereço do nó que está sendo redirecionado.

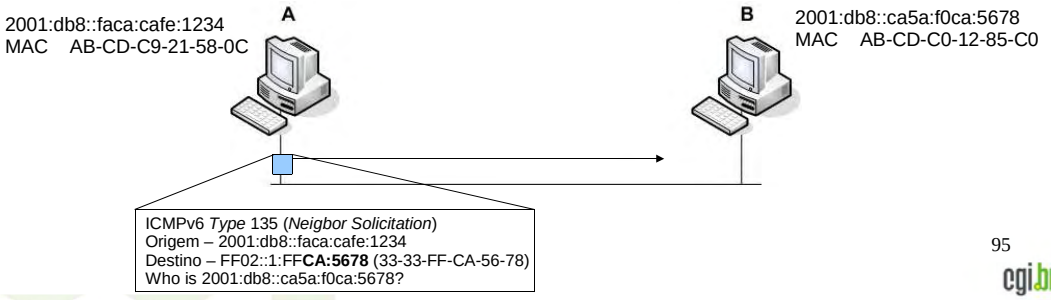
Estas mensagens podem trazer zero ou mais opções, definidas na RFC 4861:

- ***Source link-layer address****:* contém o endereço MAC do remetente do pacote. É utilizada nas mensagens NS, RS, e RA;
- ***Target link-layer address***: contém o endereço MAC de destino do pacote. É utilizada nas mensagens NA e *Redirect*;
- ***Prefix information***: forneça a *hosts* os prefixos do enlace e os prefixos para autoconfiguração do endereço. É utilizada nas mensagens RA;
- ***Redirected header***: contém todo ou parte do pacote que está sendo redirecionado. É utilizada nas mensagens *Redirect*; e
- **MTU**: indica o valor do MTU do enlace. É utilizada nas mensagens RA.

Novas opções foram definidas para novas funcionalidades do protocolo de Descoberta de Vizinhança. Estas opções serão detalhadas conforme essas novas funções forem apresentadas.

Descoberta de Vizinhança

- ***Descoberta de Endereços da Camada de Enlace***
  - Determina o endereço MAC dos vizinhos do mesmo enlace.
  - Substitui o protocolo ARP.
  - Utiliza o endereço *multicast solicited-node* em vez de *broadcast.*
    - O *host* envia uma mensagem NS informando seu endereço MAC e solicita o endereço MAC do vizinho.

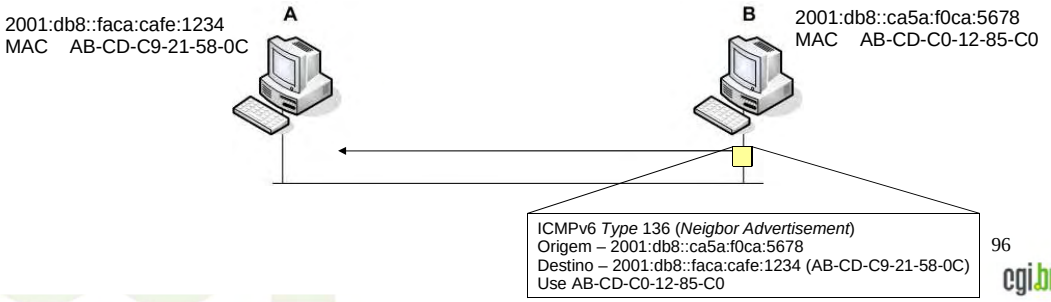


Esta funcionalidade é utilizada para determinar o endereço MAC dos vizinhos do mesmo enlace, onde um *host* envia uma mensagem NS para o endereço *multicast solicited node* do vizinho, informando seu endereço MAC.

nic.br Núcleo de Informação e Coordenação do Ponto BR — ceptro.br IPv6.br

Descoberta de Vizinhança

- ***Descoberta de Endereços da Camada de Enlace***
  - Determina o endereço MAC dos vizinhos do mesmo enlace.
  - Substitui o protocolo ARP.
  - Utiliza o endereço *multicast solicited-node* em vez de *broadcast.*
    - O *host* envia uma mensagem NS informando seu endereço MAC e solicita o endereço MAC do vizinho.
    - O vizinho responde enviando uma mensagem NA informando seu endereço MAC.



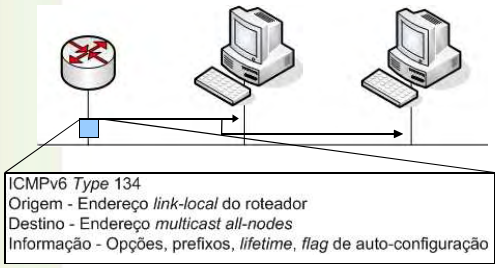
cgi.br

Ao receber a mensagem, o vizinho a responde enviando uma mensagem NA informando seu endereço MAC.

Esta característica do protocolo Descoberta de Vizinhança substitui, no IPv6, o protocolo ARP do IPv4, utilizando no lugar de um endereço *broadcast*, o endereço *multicast solicited-node* como endereço de destino.

Descoberta de Vizinhança

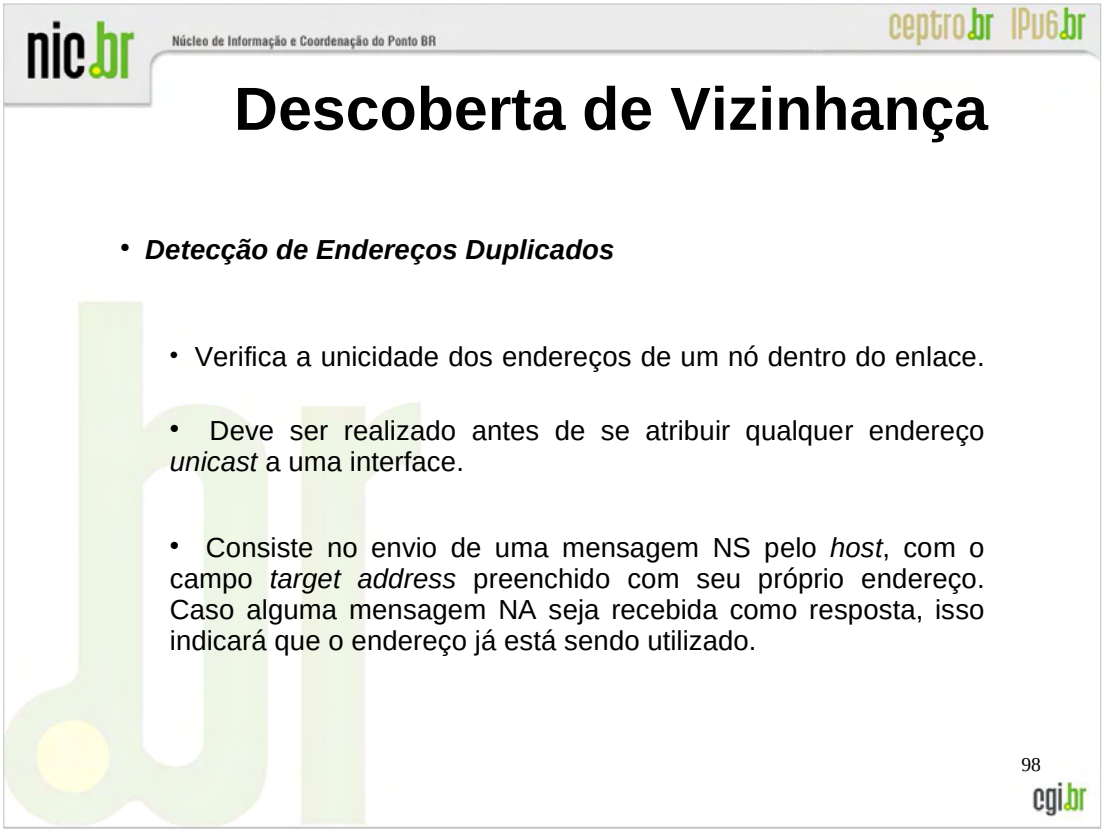
- ***Descoberta de Roteadores e Prefixos***
  - Localizar roteadores vizinhos dentro do mesmo enlace.
  - Determina prefixos e parâmetros relacionados à autoconfiguração de endereço.
  - No IPv4, está função é realizada pelas mensagens *ARP Request*.
  - Roteadores enviam mensagens RA para o endereço *multicast all-nodes.*



Esta funcionalidade do protocolo de Descoberta de Vizinhança é utilizada para localizar roteadores vizinhos dentro do mesmo enlace, bem como aprender prefixos e parâmetros relacionados à autoconfiguração de endereço.

Estas informações são enviadas a partir de um roteador local, através de mensagens RA encaminhadas para o endereço *multicast all-nodes*.

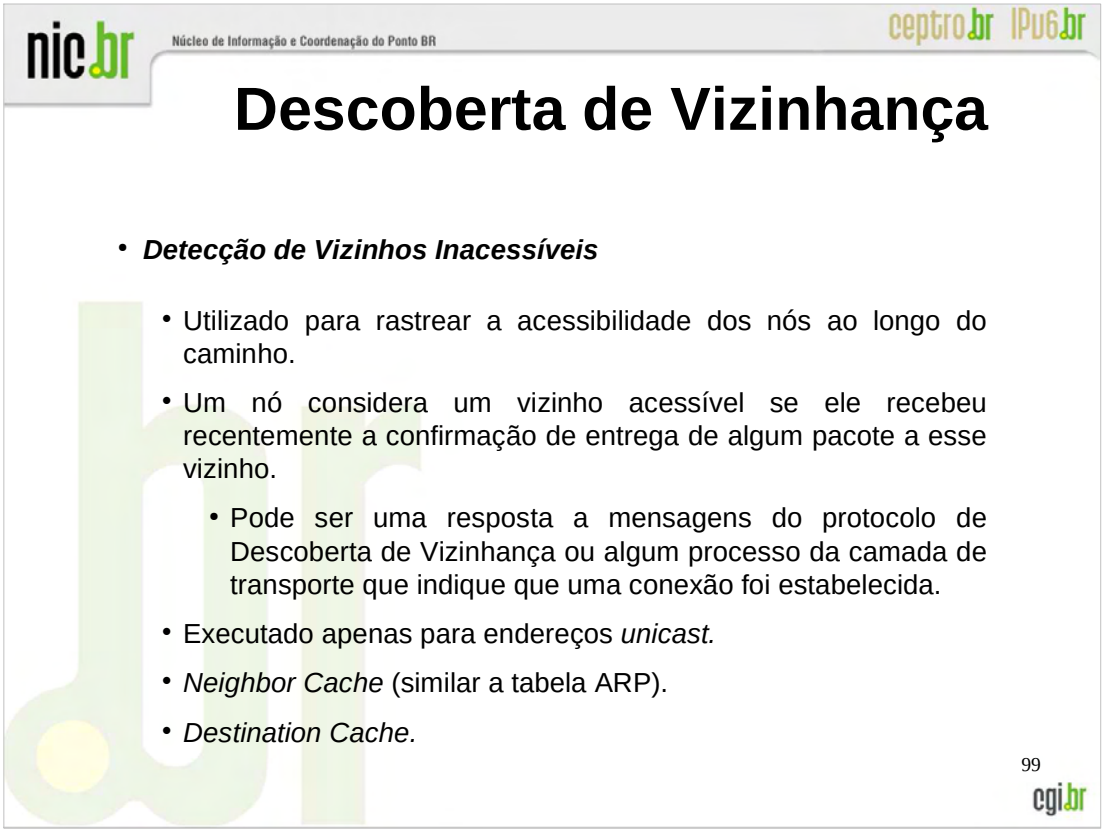
No IPv4, o mapeamento dos endereços da rede local são realizados através de mensagens *ARP Request*.



A Detecção de Endereços Duplicados é o procedimento utilizado pelos nós para verificar a unicidade dos endereços em um enlace, devendo ser realizado antes de se atribuir qualquer endereço *unicast* a uma interface, independentemente de este ter sido obtido através de autoconfiguração *stateless*, DHCPv6, ou configuração manual.

Este mecanismo consiste no envio de uma mensagem NS pelo *host*, com o campo *target address* preenchido com seu próprio endereço. Caso alguma mensagem NA seja recebida em resposta, isso indicará que o endereço já está sendo utilizado e o processo de configuração deve ser interrompido.

No IPv4, os nós utilizam mensagens *ARP Request* e o método chamado *gratuitous ARP* para detectar endereços *unicast* duplicados dentro do mesmo enlace, definindo os campos *Source Protocol Address* e *Target Protocol Address*, do cabeçalho da mensagem *ARP Request*, com o endereço IPv4 que está sendo verificado.



Este mecanismo é utilizado na comunicação *host-a-host*, *host-a-roteador*, e *roteador-a-host* para rastrear a acessibilidade dos nós ao longo do caminho.

Um nó considera um vizinho acessível se ele recebeu recentemente a confirmação de entrega de algum pacote a esse vizinho. Essa confirmação pode ocorrer de dois modos: ser uma resposta a uma mensagem do protocolo de Descoberta de Vizinhança; ou algum processo da camada de transporte que indique que uma conexão foi estabelecida.

Esse processo apenas é executado quando os pacotes são enviados a um endereço *unicast*, não sendo utilizado no envio para endereços *multicast*.

Para acompanhar os estados de um vizinho, o nó IPv6 utiliza duas importantes tabelas:

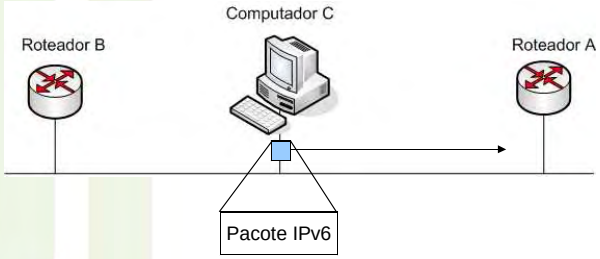
• ***Neighbor Cache*** – mantem uma lista de vizinhos locais para os quais foi enviado tráfego recentemente, armazenado seus endereços IP, informações sobre o endereço MAC e um *flag* indicando se o vizinho é um roteador ou um *host.* Também informa se ainda há pacotes na fila para serem enviados, a acessibilidade dos vizinhos e a próxima vez que um evento de detecção de vizinhos inacessíveis está agendado. Esta tabela pode ser comparada a tabela ARP do IPv4.

• ***Destination Cache*** – mantem informações sobre destinos para os quais foi enviado tráfego recentemente, incluindo tanto destinos locais quanto remotos, sendo atualizado com informações recebidas por mensagens *Redirect*. O *Neighbor Cache* pode ser considerado um subconjunto das informações do *Destination Cache*.

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Descoberta de Vizinhança

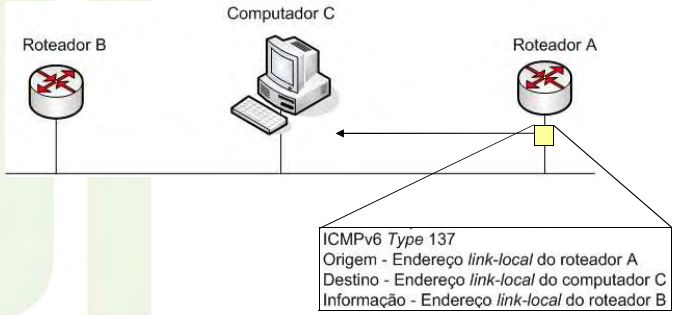
- ***Redirecionamento***
  - Envia mensagens *Redirect*
  - Redireciona um *host* para um roteador mais apropriado para o primeiro salto.
  - Informar ao *host* que destino encontra-se no mesmo enlace.
  - Este mecanismo é igual ao existente no IPv4.



Mensagens *Redirect* são enviadas por roteadores para redirecionar um *host* automaticamente a um roteador mais apropriado ou para informar ao *host* que destino encontra-se no mesmo enlace. Este mecanismo é igual ao que existe no IPv4.

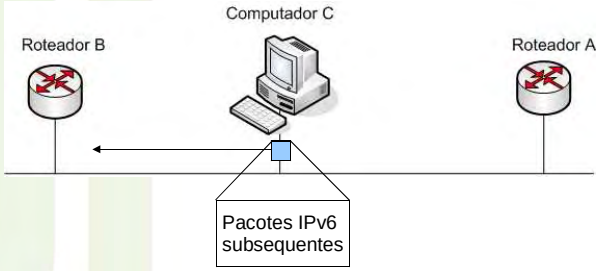
Descoberta de Vizinhança

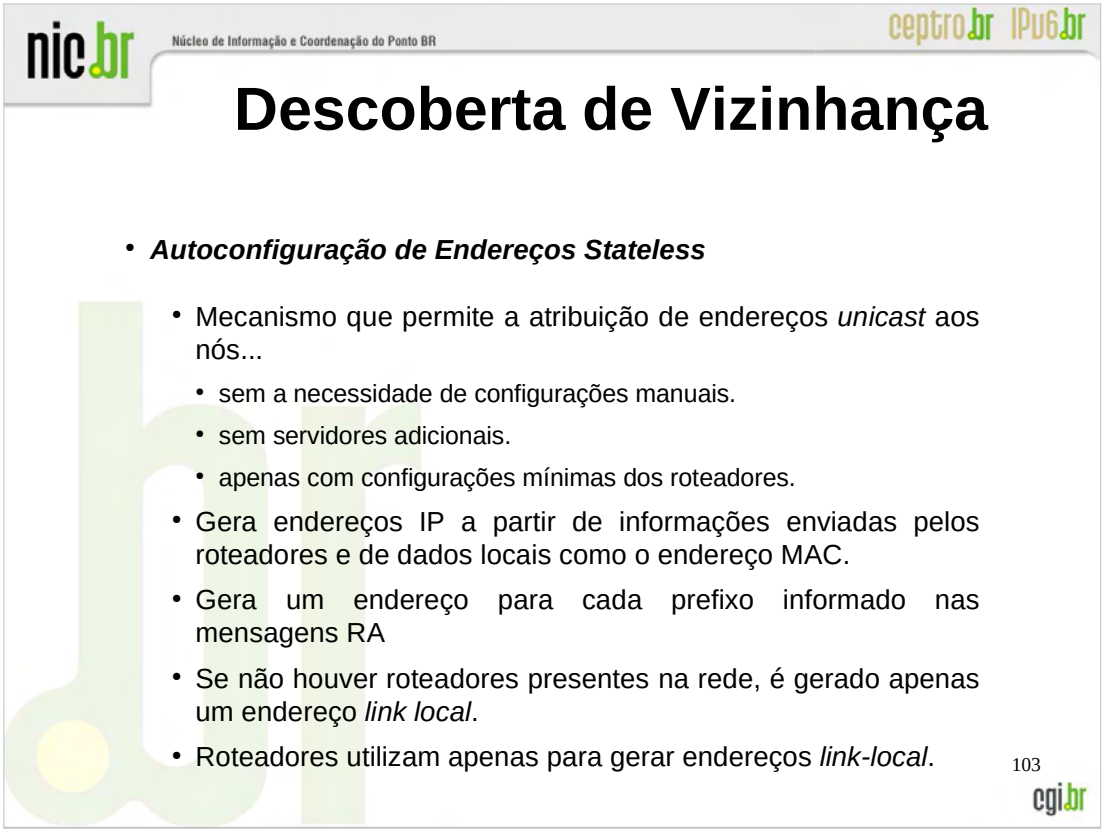
- ***Redirecionamento***
  - Envia mensagens *Redirect*
  - Redireciona um *host* para um roteador mais apropriado para o primeiro salto.
  - Informar ao *host* que destino encontra-se no mesmo enlace.
  - Este mecanismo é igual ao existente no IPv4.



Descoberta de Vizinhança

- ***Redirecionamento***
  - Envia mensagens *Redirect*
  - Redireciona um *host* para um roteador mais apropriado para o primeiro salto.
  - Informar ao *host* que destino encontra-se no mesmo enlace.
  - Este mecanismo é igual ao existente no IPv4.

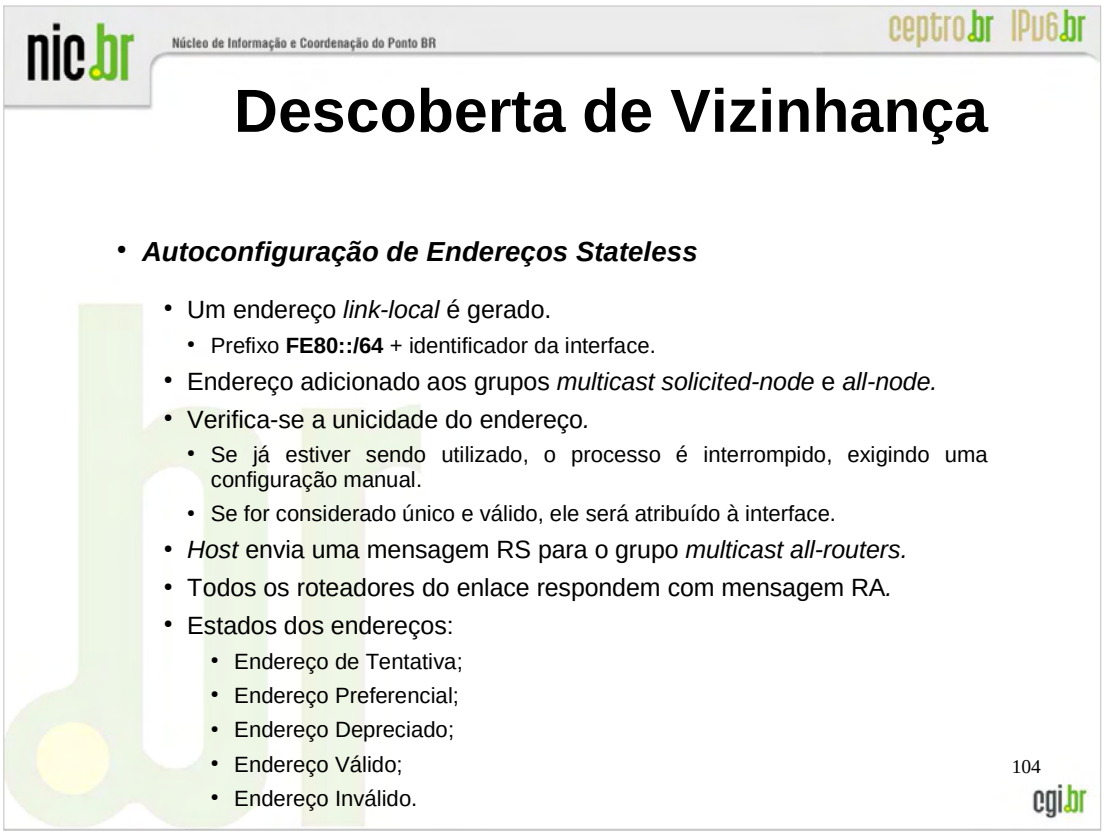




O mecanismo de autoconfiguração *stateless*, definido na RFC 4862, permite que endereços IPv6 sejam atribuídos às interfaces sem a necessidade de configurações manuais, sem a utilização de servidores adicionais (DHCP), apenas com configurações mínimas de roteadores.

Para gerar o endereço IP, um *host* utiliza uma combinação entre dados locais, como o endereço MAC da interface ou um valor randômico para gerar o ID, e informações recebidas dos roteadores, como múltiplos prefixos. Se não houver roteadores presentes, o *host* gera apenas o endereço *link local* com o prefixo **FE80::**.

Roteadores só utilizam este mecanismo para gerar endereços *link-local*. Seus endereços globais devem ser configurados de outra forma.

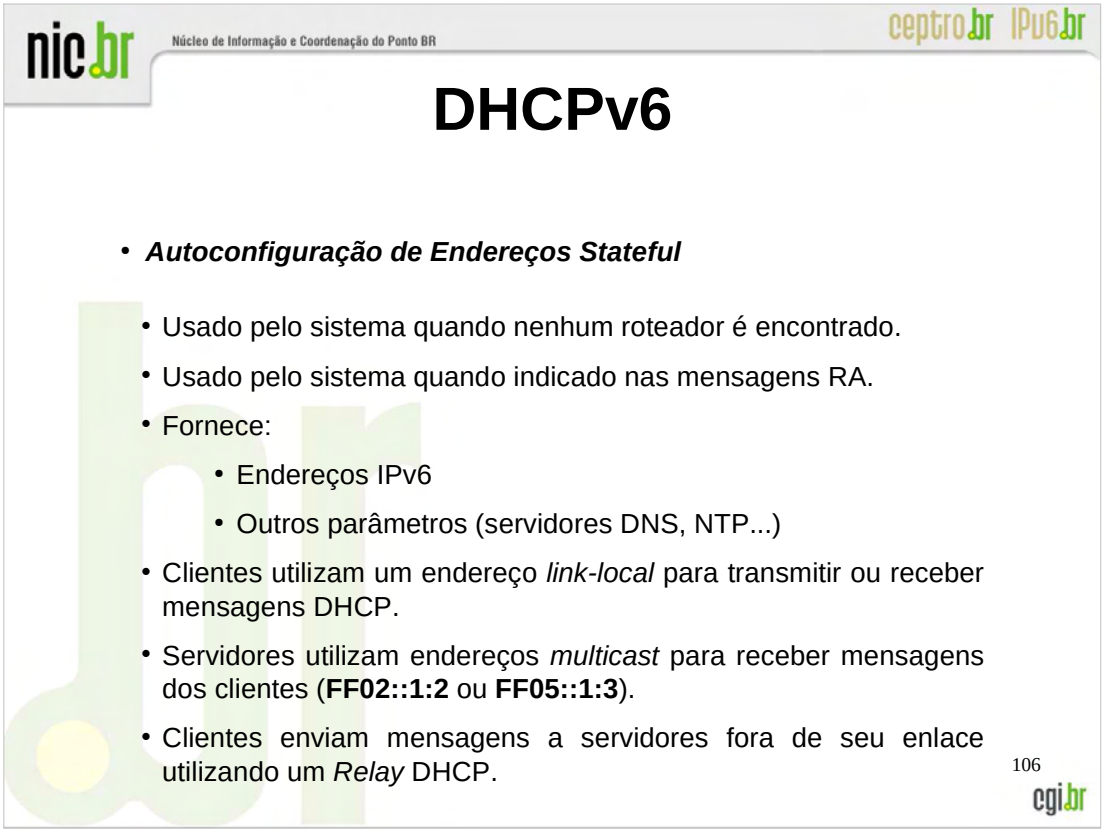


O mecanismo de autoconfiguração de endereços é executado respeitando os seguintes passo:

- Um endereço *link-local* é gerado anexando ao prefixo **FE80::/64** o identificador da interface;
- Esse endereço passa a fazer parte dos grupos *multicast solicited-node* e *all-node;*
- É feita a verificação da unicidade do endereço de *link-local* gerado*;*
  - Caso outro nó no enlace esteja utilizando o mesmo endereço, o processo de autoconfiguração é interrompido, exigindo uma configuração manual;
- Se o endereço for considerado único e válido, ele será automaticamente inicializado para a interface;
- O *host* envia uma mensagem *Router Solicitation* para o grupo *multicast all-routers;*
- Todos os roteadores do enlace respondem com uma mensagem *Router Advertisement* informando: os roteadores padrão; um valor predefinido para o campo Limite de Encaminhamento; o MTU do enlace; a lista de prefixos da rede, para os quais também serão gerados endereços automaticamente.

Um endereço IPv6 pode assumir diferentes estados:

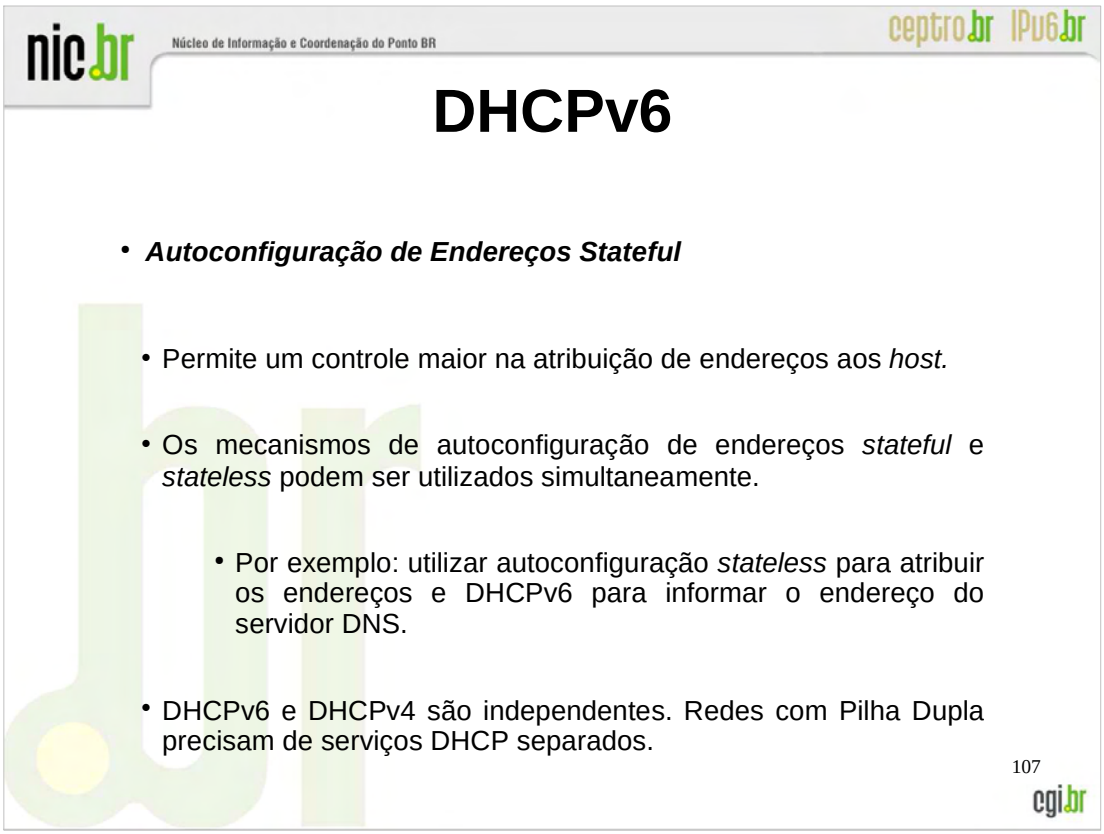
- Endereço de Tentativa – endereço qua ainda não foi atribuído. É o estado anterior a atribuição, enquanto o processo de DAD é realizado. Não pode ser utilizado na comunicação do nó, apenas por mensagens relativas à Descoberta de Vizinhança;
- Endereço Preferencial – endereço atribuído a interface e pode ser utilizado sem restrições, até expirar seu tempo de vida;
- Endereço Depreciado – endereço cujo tempo de vida expirou. Pode ser utilizado para continuar as comunicações abertas por ele, mas não para iniciar novas comunicações;
- Endereço Válido – termo utilizado para designar tanto os endereços preferenciais quanto os depreciados;
- Endereço Inválido – endereço que não pode ser atribuído a uma interface. Um endereço se torna inválido quando seu tempo de vida expira.



O *Dynamic Host Configuration Protocol* (DHCP) é um protocolo de autoconfiguração *stateful* utilizado na distribuição de endereços IP dinamicamente em uma rede, a partir de um servidor DHCP, fornecendo um controle maior na atribuição de endereços aos *host.*

Definido na RFC 3315 o DHCPv6 é uma opção ao mecanismo de autoconfiguração *stateless* do IPv6, podendo ser utilizado quando não há roteadores na rede, ou quando seu uso for indicado nas mensagens RA, sendo capaz de fornecer endereços IPv6 e diversos parâmetros de rede, como endereços de servidores DNS, NTP, SIP, etc.

No DHCPv6, a troca de mensagens entre cliente e servidor é realizada utilizando-se o protocolo UDP. Os clientes utilizam um endereço *link-local* para transmitir ou receber mensagens DHCP, enquanto que os servidores utilizam um endereço *multicast* reservado (**FF02::1:2** ou **FF05::1:3**) para receber mensagens dos clientes. Caso o cliente necessite enviar uma mensagem a um servidor que esteja fora de sua sub-rede, é utilizado um *Relay* DHCP.



A utilização de DHCPv6 oferece um controle maior na atribuição de endereços, visto que, além de fornecer opções de configuração de rede, é possível definir políticas de alocação de endereços e atribuir endereços aos *hosts* que não sejam derivados do endereço MAC.

Em uma rede IPv6, é possível combinar o uso de autoconfiguração *stateless* com servidores DHCP. Neste cenário, é possível por exemplo, utilizar autoconfiguração *stateless* na atribuição de endereços aos *hosts* e servidores DHCPv6 para fornecer informações adicionais de configuração, como o endereço de servidores DNS.

Os protocolos DHCPv6 e DHCPv4 são independentes, de modo que, em uma rede com Pilha Dupla, será necessário rodar um serviço para cada protocolo. Com DHCPv4, é preciso configurar no cliente se este usará DHCP, enquanto que com o DHCPv6, sua utilização é indicada através das opções das mensagens RA.

nic.br Núcleo de Informação e Coordenação do Ponto BR — ceptro.br IPv6.br

Renumeração da Rede

- *Hosts* – Autoconfiguração *stateless* ou DHCPv6
- Roteadores – *Router Renumbering*
  - Mensagens ICMPv6 Tipo 138
  - Formato da Mensagem
    - Cabeçalho RR + Corpo da Mensagem

Tipo	Código	Soma de Verificação
Número Sequencial		
Número de Segmento	Flags	Atraso Máximo
Reservado		
Corpo da Mensagem Mensagem de Comando / Mensagem de Resultado		

O endereçamento de uma rede muitas vezes é baseado nos prefixos atribuídos por ISPs. No caso de uma mudança de provedor, é necessário renumerar todos os endereços da rede.

No IPv6, o processo de reendereçamento dos *hosts* pode ser feito de forma relativamente simples. Através dos mecanismos do protocolo de Descoberta de Vizinhança, um novo prefixo pode ser anunciado pelo roteador a todos os *hosts* do enlace. É possível também, a utilização de servidores DHCPv6. Para tratar a configuração e reconfiguração dos prefixos nos roteadores tão facilmente quanto nos *hosts*, foi definido na RFC 2894, o protocolo *Router Renumbering*.

O mecanismo *Router Renumbering* utiliza mensagens ICMPv6 do tipo 138, enviadas aos roteadores, através do endereço *multicast all-routers*, contendo as instruções de como atualizar seus prefixos.

As mensagens *Router Renumbering* são formadas pelos seguintes campos:

- Tipo - 138 (decimal);
- Código - 0 para mensagens de Comando;

  1 para mensagens de Resultado;

  255 para Zerar Número Sequencial;

- Soma de Verificação – verifica a integridade da mensagem ICMPv6 e de parte do cabeçalho IPv6;
- Número Sequencial – identifica as operações;
- Número de Segmento – enumera diferentes mensagens RR válidas que tenham o mesmo Número Sequencial.
- *Flags* – T: indica se a configuração do roteador deve ser modificada, ou se é um teste;

  R: indica se uma mensagem de Resultado deve ser enviada;

  A: indica se o comando deve ser aplicado a todas as interfaces, independente do seu estado;

  S: indica que o comando deve ser aplicado a todas as interfaces, independente de qual sub-rede pertençam;

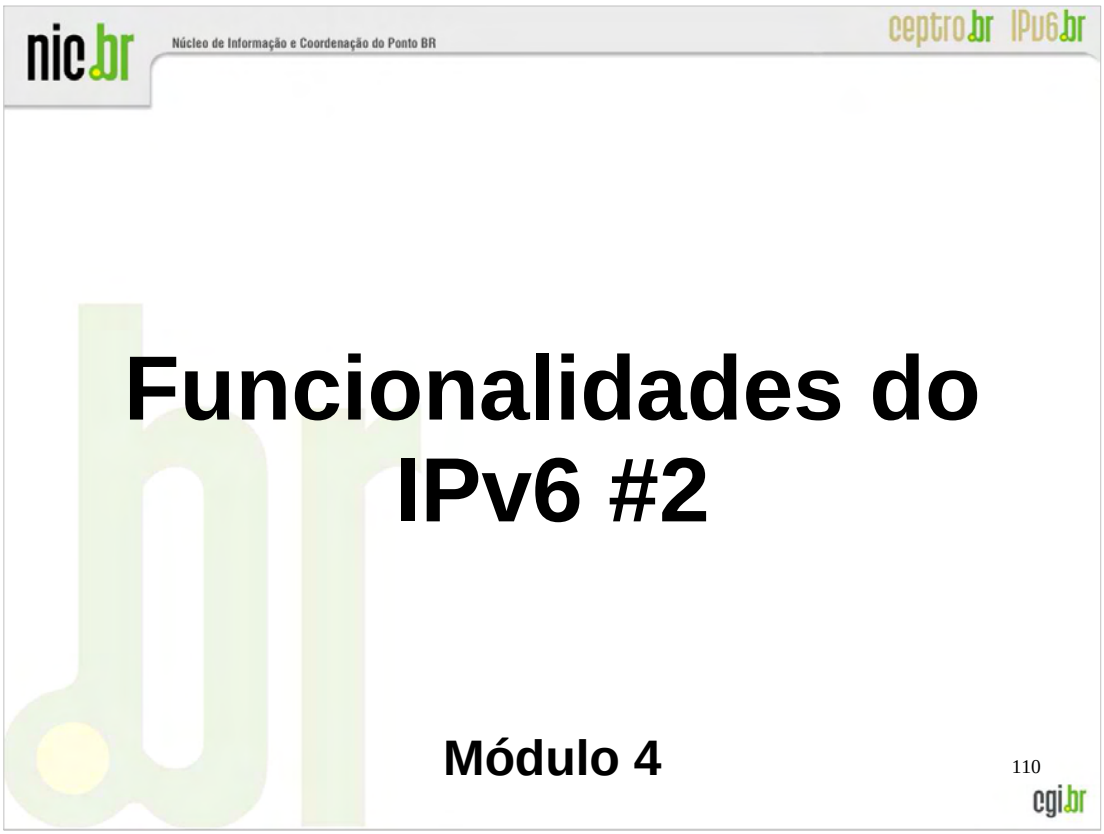
  P: indica que a mensagem de Resultado contém o relatório completo do processamento da mensagem de Comando, ou que a mensagem de Comando foi previamente tratada (e não é um teste) e que o roteador não está processando-a novamente.
- Atraso Máximo - especifica o tempo máximo, em milisegundos, que um roteador deve atrasar o envio de qualquer resposta a mensagem de Comando.

As mensagens de Comando são formadas por sequencias de operações, *Match-Prefix* e *Use-Prefix.* O *Match-Prefix* indica qual prefixo deve ser modificado, e o *Use-Prefix* indica o novo prefixo. As operações podem ser ADD, CHANGE, ou SET-GLOBAL, que instruem, respectivamente, o roteador a adicionar os prefixos indicados em *Use-Prefix* ao conjunto de prefixos configurados; a remover o prefixo indicado em *Match-Prefix*, se existirem, e trocá-los pelos contidos em *Use-Prefix*; ou substituir todos os prefixos de escopo global, pelos prefixos do *Use-Prefix*. Se o conjunto de *Use-Prefix* for vazio, a operação não ADD não faz nenhuma adição e as outras duas operações apenas apagam o conteúdo indicado.

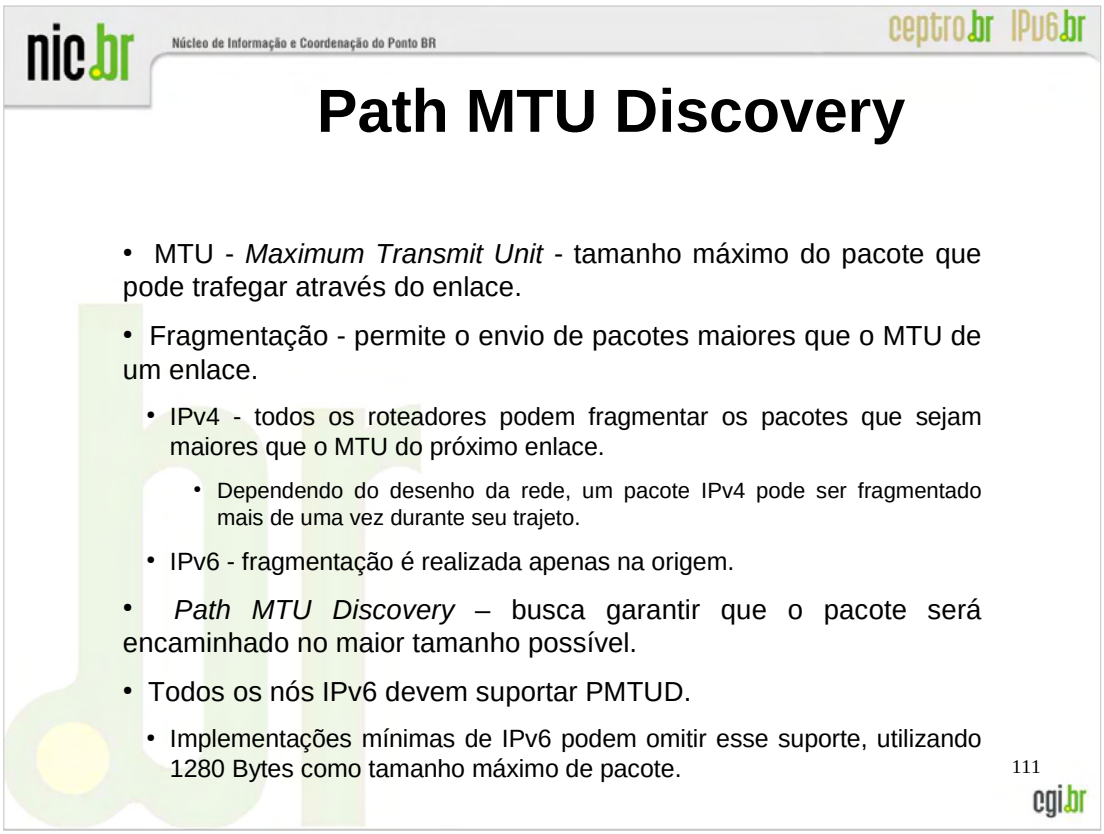
Os roteadores também enviam mensagens de Resultados contendo um *Match Report* para cada prefixo igual aos enviados na mensagem de Comando.

Mais informações:

- RFC 3315 - *Dynamic Host Configuration Protocol for IPv6* (DHCPv6)
- RFC 4443 - *Internet Control Message Protocol* (ICMPv6) *for the Internet Protocol Version 6* (IPv6) *Specification*
- RFC 4861 - *Neighbor Discovery for IP version 6* (IPv6)
- RFC 5006 - *IPv6 Router Advertisement Option for DNS Configuration*



Dando continuidade ao estudo das funcionalidades do protocolo IPv6, veremos a seguir como o novo protocolo trata a questão da fragmentação de pacotes e o gerenciamento de grupos *multicast*. Veremos também, as alterações feitas no protocolo DNS e as melhorias apresentadas pelo protocolo IPv6 referentes à aplicação de QoS e ao suporte à mobilidade.

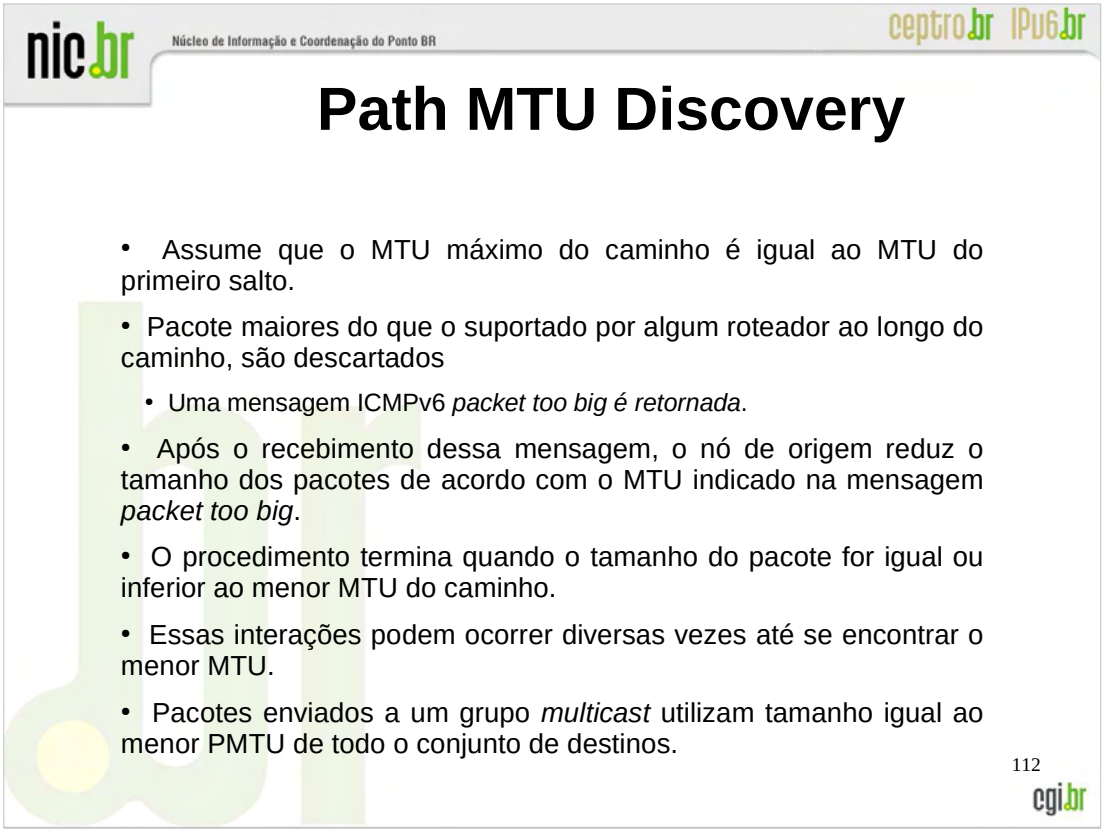


Determinado pelos protocolos de roteamento, cada enlace na rede pode possuir um valor diferente de MTU, ou seja, uma limitação distinta em relação ao tamanho máximo do pacote que pode trafegar através dele. Para que pacotes maiores que o MTU de enlace seja encaminhado, ele deve ser fragmentado em pacotes menores, que serão remontados ao chegarem em seu destino.

Na transmissão de um pacote IPv4, cada roteador ao longo do caminho pode fragmentar os pacotes, caso estes sejam maiores do que o MTU do próximo enlace. Dependendo do desenho da rede, um pacote IPv4 pode ser fragmentado mais de uma vez durante seu trajeto através da rede, sendo reagrupado no destino final.

No IPv6, a fragmentação dos pacotes é realizada apenas na origem, não sendo permitida em roteadores intermediários. Este processo tem o intuito de reduzir o *overhead* do cálculo dos cabeçalhos alterados nos roteadores intermediários.

Para isso, é utilizado, no início do processo de fragmentação, o protocolo *Path MTU Discovery*, descrito na RFC 1981, que descobre de forma dinâmica qual o tamanho máximo permitido ao pacote, identificando previamente os MTUs de cada enlace no caminho até o destino. O protocolo PMTUD deve ser suportado por todos os nós IPv6. No entanto, Implementações mínimas de IPv6 podem omitir esse suporte, utilizando 1280 Bytes como tamanho máximo de pacote.



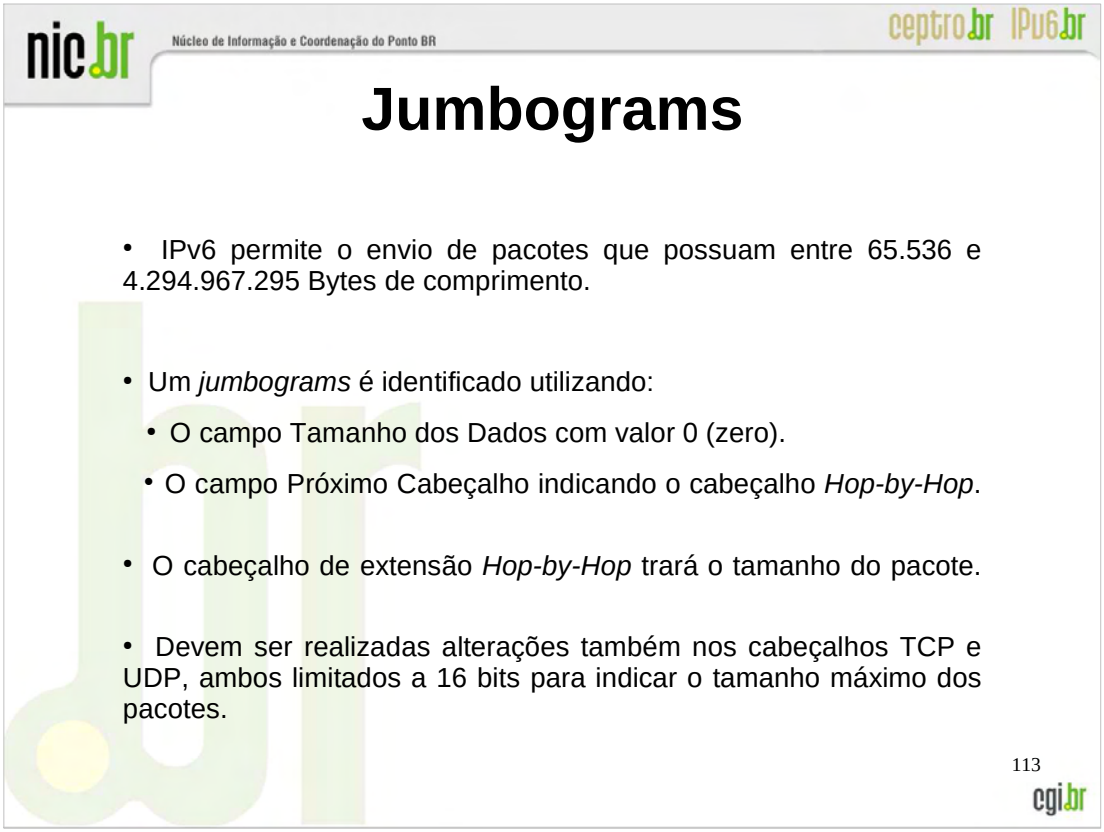
O processo de *Path MTU Discovery* se inicia assumindo que o MTU de todo o caminho é igual ao MTU do primeiro salto. Se o tamanho dos pacotes enviados for maior do que o suportado por algum roteador ao longo do caminho, este irá descartá-lo e retornará uma mensagem ICMPv6 *packet too big*, que devolve juntamente com a mensagem de erro, o valor do MTU do enlace seguinte. Após o recebimento dessa mensagem, o nó de origem reduzirá o tamanho dos pacotes de acordo com o MTU indicado na mensagem *packet too big*.

Esse procedimento termina quando o tamanho do pacote for igual ou inferior ao menor MTU do caminho, sendo que estas iterações, de troca de mensagens e redução do tamanho dos pacotes, podem ocorrer diversas vezes até se encontrar o menor MTU. Caso o pacote seja enviado a um grupo *multicast*, o tamanho utilizado será o menor PMTU de todo o conjunto de destinos.

De um ponto de vista teórico, o PMTUD pode parecer imperfeito, dado que o roteamento dos pacotes é dinâmico, e cada pacote pode ser entregue através de uma rota diferente. No entanto, essas mudanças não são tão frequentes, e caso o valor do MTU diminua devido a uma mudança de rota, a origem receberá a mensagem de erro e reduzirá o valor do *Path MTU*.

Mais informações:

- RFC 1981 - *Path MTU Discovery for IP version 6*



A RFC 2675 define uma opção do cabeçalho de extensão *Hop-By-Hop* chamada Jumbo Payload. Esta opção permitir envio de pacotes IPv6 com cargas úteis entre 65.536 e 4.294.967.295 Bytes de comprimento, conhecidos como *jumbograms*.

Ao enviar *jumbograms*, o cabeçalho IPv6 trará os campos Tamanho dos Dados e Próximo Cabeçalho com o valor zero. Este último indicará que as opções do cabeçalho de extensão *Hop-By-Hop* devem ser processadas pelos nós, onde são indicados os tamanhos dos pacotes *jumbograms*.

O cabeçalho UDP possui um campo de 16 bits chamado Tamanho, que indica o tamanho do cabeçalho UDP mais o tamanho dos dados, não permitindo o envio de pacotes com mais 65.536 Bytes. Entretanto, é possível o envio de *jumbograms* definindo o campo Tamanho como zero, deixando que o receptor extraia o tamanho real do pacote UDP a partir do tamanho do pacote IPv6.

Nos pacotes TCP, as opções *Maximum Segment Size* (MSS), que negocia no início da conexão o tamanho máximo do pacote TCP a ser enviado, e *Urgent Pointer*, que indica um deslocamento de Bytes a partir do número de seqüência em que dados com alta prioridade devem ser encontrados, também não podem referenciar pacotes maiores que 65.535 Bytes. Deste modo, para se enviar *jumbograms* é preciso no caso do MSS, determinar seu valor como 65.535, que será tratado como infinito pelo receptor do pacote. Em relação ao *Urgent Pointer*, a solução é semelhante, visto que se pode determinar o valor do *Urgent Pointer* como 65.535, indicando que este está além do final deste pacote.

Mais informações:
- RFC 2675 - *IPv6 Jumbograms*

Recapitulando o que foi dito no módulo anterior, *multicast* é uma técnica que permite endereçar múltiplos nós como um grupo, possibilitando o envio de pacotes a todos os nós que o compõe a partir de um endereço único que o identifica.

Os membros de um grupo *multicast* são dinâmicos, sendo que os nós podem entrar e sair de um grupo a qualquer momento, não existindo limitações para o tamanho de um grupo *multicast*.

O gerenciamento dos grupos *multicast* no IPv6 é realizado pelo *Multicast Listener Discovery* (MLD), definido na RFC 2710. Este protocolo é o responsável por informar aos roteadores *multicast* locais o interesse de nós em fazer parte ou sair de um determinado grupo *multicast*. No IPv4, este trabalho é realizado pelo protocolo *Internet Group Management Protocol* (IGMPv2).

O MLD utiliza três tipos de mensagens ICMPv6:

- *Multicast Listener Query* (Tipo 130) - as mensagens *Query* possuem dois subtipos. A *General Query* é utilizada por roteadores verificar periodicamente os membros do grupo, solicitando a todos os nós *multicast* reportem todos os grupos de que fazem parte. A *Multicast-Address-Specific Query* é utilizada por roteadores para descobrir se existem nós fazendo parte de um determinado grupo;
- *Multicast Listener Report* (Tipo 131) - mensagens *Report* não solicitadas são enviadas por um nó quando este começa a fazer parte de grupo *multicast*. Elas também são geradas em resposta a mensagens *Query*;
- *Multicast Listener Done* (Tipo 132) – enviada pelos nós quando estes estão deixando um determinado grupo.

Estas mensagens são enviadas com um endereço de origem *link-local* e com o valor 1 no campo Limite de Encaminhamento, garantindo que elas permaneçam na rede local. Caso o pacote possua um cabeçalho *Hop-by-Hop*, a *flag Router Alert* será marcada, deste modo, os roteadores não descartarão o pacote, ainda que o endereço do grupo *multicast* em questão, não esteja sendo ouvido por eles.

Uma nova versão do protocolo MLD, chamada de MLDv2, foi definida na RFC3810. Equivalente ao IGMPv3, além de incorporar as funcionalidades de gerenciamento de grupos do MLD, esta nova versão introduziu o suporte a filtragem de origem, que permite a um nó especificar se não deseja receber pacotes de uma determinada origem, ou informar o interesse em receber pacotes somente de endereços específicos. Por padrão, os membros de um grupo recebem pacotes de todos os membros deste grupo.

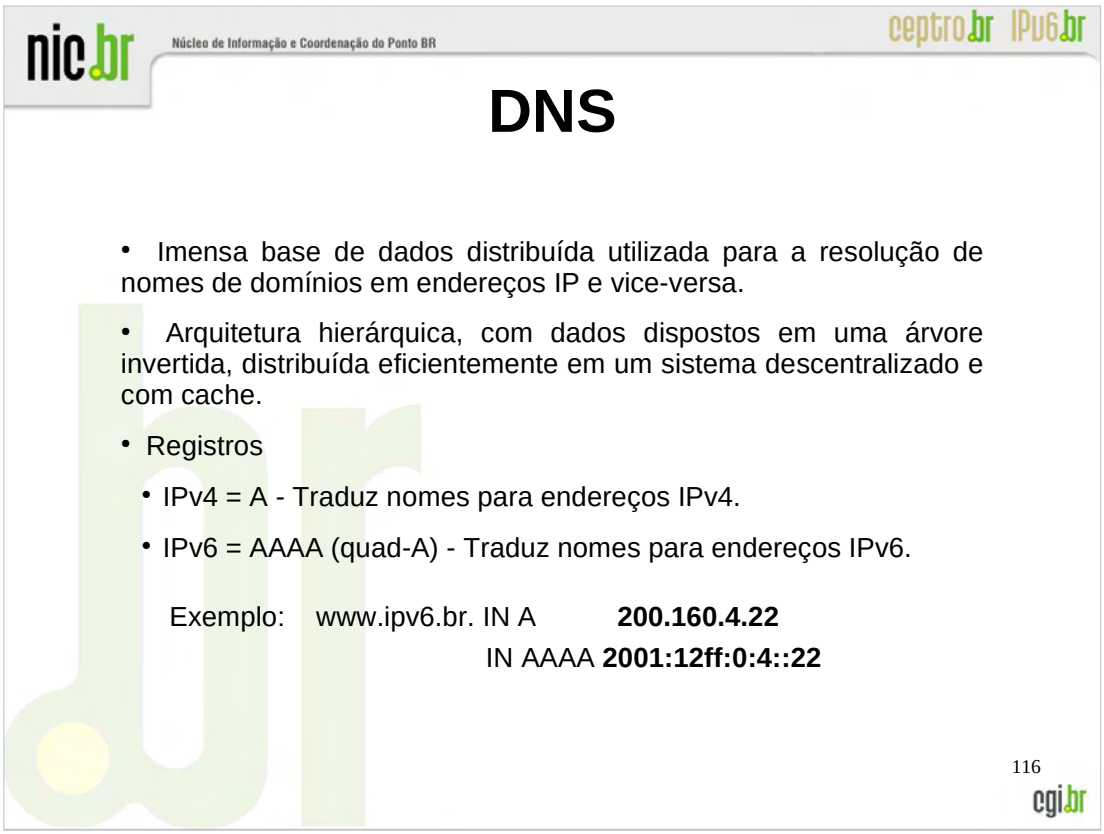
Outro mecanismo importante para o funcionamento dos grupos *multicast*, é o *Multicast Router Discovery* (MRD). Definido na RFC 4286, ele é utilizado na descoberta de roteadores *multicast* na rede. Ele utiliza três mensagens ICMPv6:

- *Multicast Router Advertisement* (Tipo 151)- esta mensagem é enviada por roteadores para anunciar que o roteamento IP *multicast* está habilitado. Ela é enviada a partir do endereço *link-local* do roteador para o endereço *multicast all-snoopers* (**FF02::6A**);
- *Multicast Router Solicitation* (Tipo 152) – esta mensagem é enviada pelos dispositivos para solicitar mensagens *Multicast Router Advertisement* aos roteadores *multicast.* Ela é enviada a partir do endereço *link-local* do dispositivo para o endereço *multicast all-routers* (**FF02::2**);
- *Multicast Router Termination* (Tipo 153) – Esta mensagem é enviada por roteadores para anunciar que suas interfaces não estão mais encaminhando pacotes IP *multicast*. Ela é enviada a partir do endereço *link-local* do roteador para o endereço *multicast all-snoopers* (**FF02::6A**).

Todas as mensagens MRD também são enviadas com um Limite de Encaminhamento igual a 1 e com contendo a opção *Router Alert*.

Mais informações:

- RFC 2710 - *Multicast Listener Discovery (MLD) for IPv6*
- RFC 4286 - *Multicast Router Discovery*



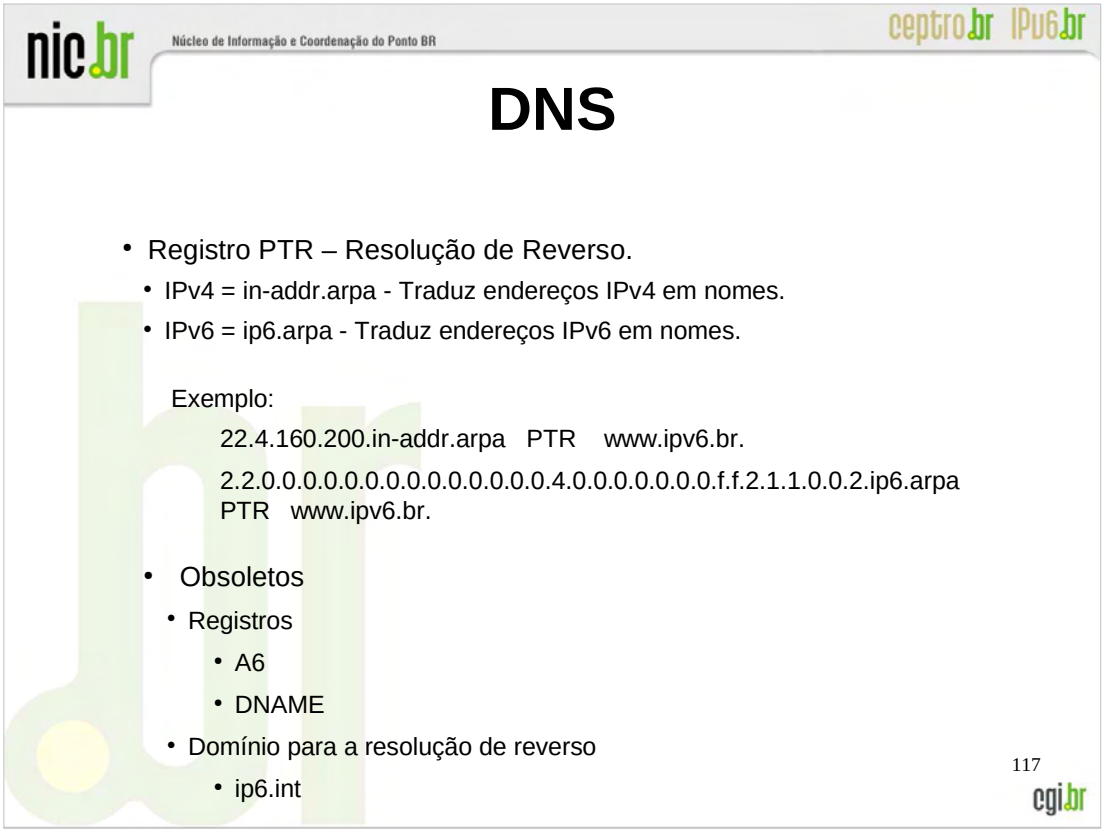
O protocolo *Domain Name System* (DNS) é uma imensa base de dados distribuída utilizada para a resolução de nomes de domínios em endereços IP e vice-versa. Possui uma arquitetura hierárquica, com dados dispostos em uma árvore invertida, distribuída eficientemente em um sistema descentralizado e com cache.

Para que o DNS trabalhe com a versão 6 do protocolo IP, algumas mudanças foram definidas na RFC 3596.

Um novo registro foi criado para armazenar os endereços IPv6 de 128 bits, o AAAA ou quad-A. Sua função é traduzir nomes para endereços IPv6, equivalente ao registro A utilizado com o IPv4. Caso um *host* possua mais de um endereço IPv6, ele terá um registro quad-A para cada endereço. Os registros são representados como se segue:

Exemplo: www.ipv6.br. IN A **200.160.4.22**

IN AAAA **2001:12ff:0:4::22**



Para resolução de reverso, foi adicionado o registro PTR ip6.arpa, responsável por traduzir endereços IPv6 em nomes. Em sua representação, omitir sequência de zeros não é permitido e o bit menos significativo é colocado mais a esquerda, como é possível observar no exemplo a seguir:

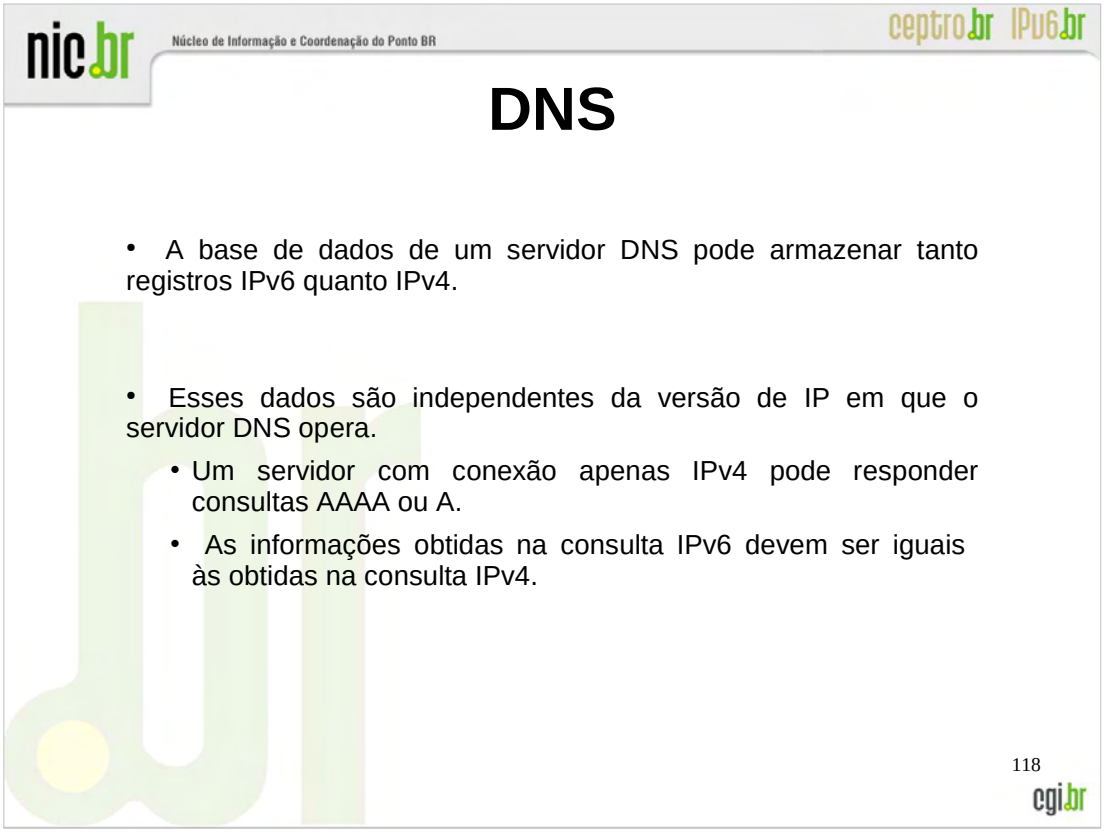
Exemplo:

22.4.160.200.in-addr.arpa PTR www.ipv6.br.

2.2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.4.0.0.0.0.0.0.0.f.f.2.1.1.0.0.2.ip6.arpa PTR www.ipv6.br.

Os outros tipos de registro DNS não sofreram alterações, apenas foram adaptados para suportar o novo tamanho dos endereços.

A RFC 2874 introduziu os registros A6 e DNAME, com o intuito de facilitar renumeração de redes, onde cada *nameserver* detém apenas uma parte do endereço IPv6. Inicialmente o domínio para resolução de reverso, definido na RFC 1886, era o ip6.int, no entanto, houve manifestações contrárias a sua utilização, pois, o .int significa "internacional", e não deve servir para fins administrativos na Internet. Os registros A6 e DNAME tornaram-se obsoletos pelo desuso, e o domínio .int foi substituído pelo .arpa, respectivamente nas RFCs 3363 e 3152,



O suporte a IPv6 do DNS deve ser observado por dois aspectos. O primeiro, é que um servidor DNS deve ser capaz de armazenar registros quad-A para endereços IPv6. O segundo aspecto, é se um servidor DNS é capaz de transportar consultas e respostas através de conexões IPv6. Isto é, a base de dados de um servidor DNS pode armazenar tanto registros IPv6 quanto IPv4, independente versão de IP em que este servidor opera.

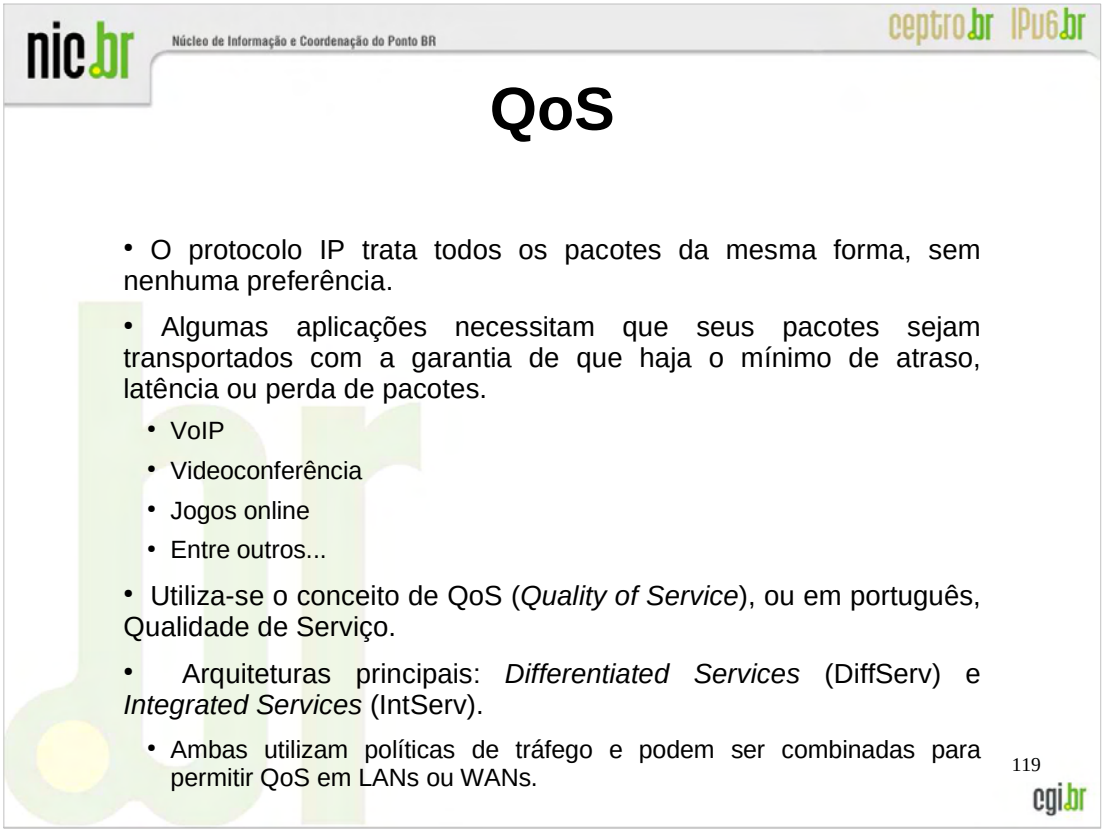
Com isso, um servidor com conexão apenas IPv4 pode responder tanto consultas AAAA quanto A. No entanto, as informações obtidas na consulta via IPv6 devem ser iguais às obtidas na consulta IPv4.

Mais informações:

RFC 3596 - *DNS Extensions to Support IP Version 6*

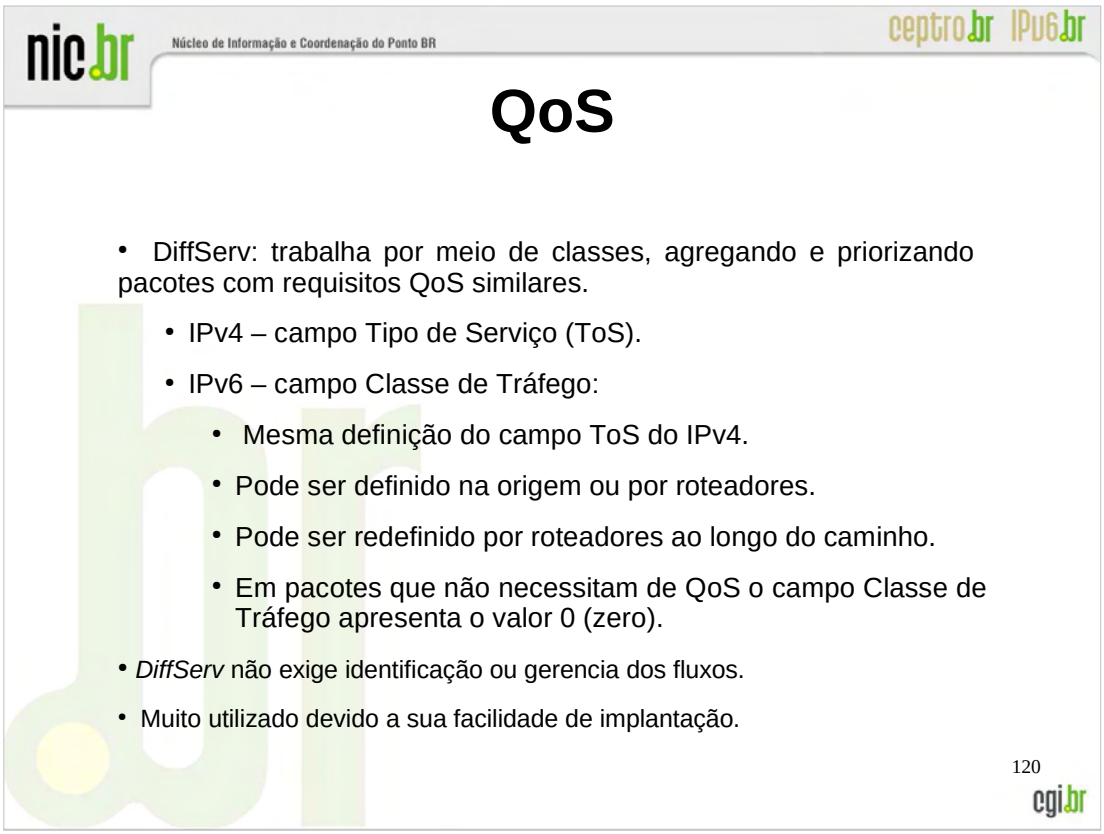
RFC 3363 - *Representing Internet Protocol version 6 (IPv6) Addresses in the Domain Name System (DNS)*

RFC 3364 - *Tradeoffs in Domain Name System (DNS) Support for Internet Protocol version 6 (IPv6)*



A princípio, o protocolo IP trata todos os pacotes da mesma forma, sem nenhuma preferência no momento de encaminhá-los. Isto pode acarretar diversas implicações no desempenho de uma aplicação, visto que, atualmente muitas dessas aplicações, como voz e vídeo sobre IP, requerem transmissão e reprodução praticamente em tempo real, podendo ter sua qualidade diminuída devido à ocorrência de perda de pacotes, entrega fora de ordem, atraso ou variação de sinal. Estes problemas podem acontecer devido à forma como o tráfego chega e é manipulado pelos roteadores, dado que, vindo de diferentes interfaces e diversas redes, o roteador processa os pacotes na ordem em que são recebidos.

O conceito de QoS (*Quality of Service*), ou em português, Qualidade de Serviço, é empregado para em protocolos cuja a tarefa é prover a transmissão de determinados tráfegos de dados com prioridade e garantia de qualidade. Existem atualmente, duas arquiteturas principais: a *Differentiated Services* (DiffServ) e a *Integrated Services* (IntServ). Ambas utilizam políticas de tráfego e podem ser combinadas para permitir QoS em LANs ou WANs.

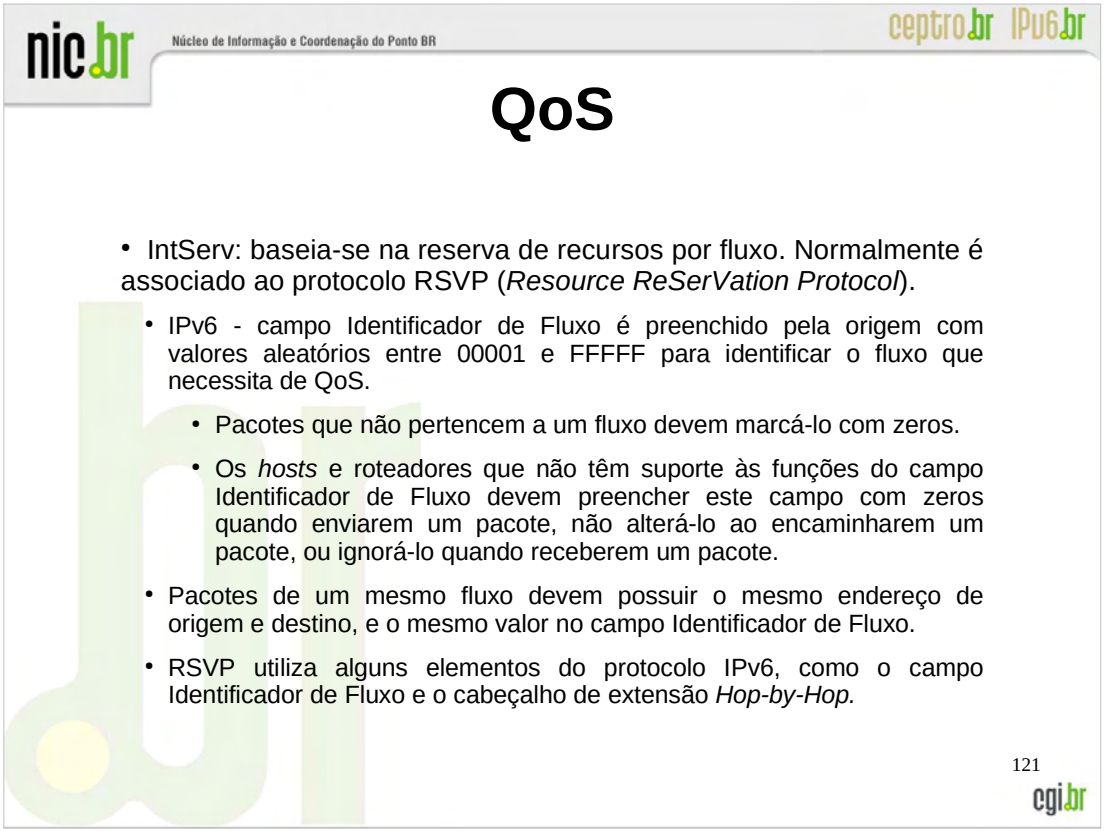


O *DiffServ* trabalha por meio de classes, agregando e priorizando pacotes com requisitos QoS similares.

Pacotes *DiffServ* são identificados pelos oito bits dos campos Tipo de Serviço do IPv4 e Classe de Tráfego do IPv6, com o intuito de identificar e distinguir as diferentes classes ou prioridades de pacotes que necessitem de QoS.

Ambos os campos possuem as mesmas definições e as prioridades atribuídas a cada tipo de pacote podem ser definidos tanto na origem quanto nos roteadores, podendo também, serem redefinidas ao longo do caminho por roteadores intermediários. Pacotes que não necessitem de QoS o campo Classe de Tráfego apresenta o valor zero.

Comparado com o *IntServ*, o *DiffServ* não exige qualquer identificação ou gerencia dos fluxos, além de ser geralmente mais utilizado nas redes, devido a sua facilidade de implantação.



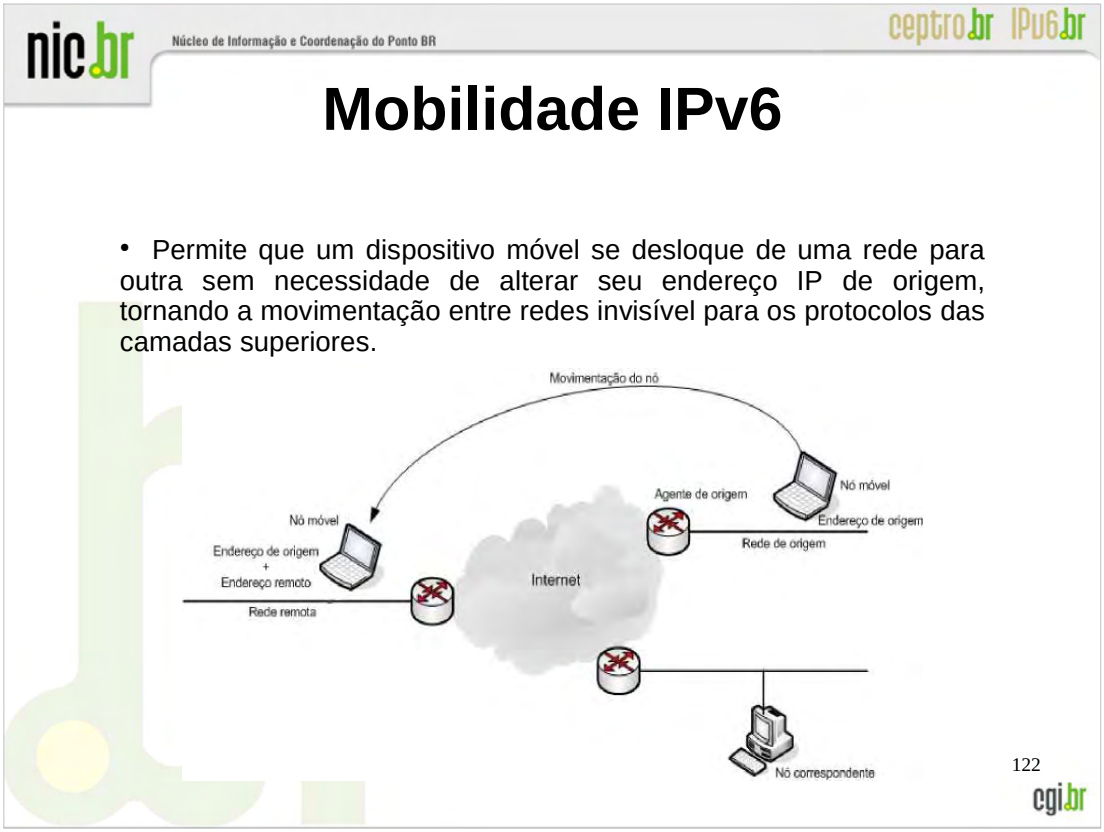
O modelo IntServ baseia-se na reserva de recursos por fluxo e sua utilização está normalmente associada ao protocolo RSVP. O RSVP é utilizado para reservar o recurso ao longo do caminho da fonte até o destino de um fluxo que requer QoS.

No IPv6, para identificar os fluxos que necessitam de QoS são utilizados os 20 bits do campo Identificador de Fluxo, que são preenchidos com valores aleatórios entre 00001 e FFFFF. Pacotes que não pertencem a um fluxo devem marcar o campo Identificador de Fluxo com zeros. Os *hosts* e roteadores que não têm suporte as funções deste campo devem preencher este campo com os zeros quando enviarem um pacote, não alterá-lo ao encaminharem um pacote, ou ignorá-lo quando receberem um pacote. Pacotes de um mesmo fluxo devem possuir o mesmo endereço de origem e destino, e o mesmo valor no campo Identificador de Fluxo.

O RSVP utiliza alguns elementos do protocolo IPv6, como o campo Identificador de Fluxo e o cabeçalho de extensão *Hop-by-Hop*. Pacotes RSVP são enviados com o mesmo valor no campo Identificador de Fluxo, junto com o cabeçalho de Extensão *Hop-By-Hop*, usado para transportar uma mensagem *Router Alert*, indicando para cada roteador no caminho do tráfego QoS, que o pacote IP deverá ser processado.

Mais informações:

- RFC 1633 - *Integrated Services in the Internet Architecture: an Overview*
- RFC 2205 - *Resource ReSerVation Protocol* (RSVP)
- RFC 2475 - *An Architecture for Differentiated Services*
- RFC 3260 - *New Terminology and Clarifications for Diffserv*



O suporte à mobilidade permite que um dispositivo móvel se desloque de uma rede para outra sem necessidade de alterar seu endereço IP de origem, tornando a movimentação entre redes invisível para os protocolos das camadas superiores. Com isso, todos os pacotes enviados para este nó móvel, continuarão sendo encaminhados a ele usando o endereço de origem.

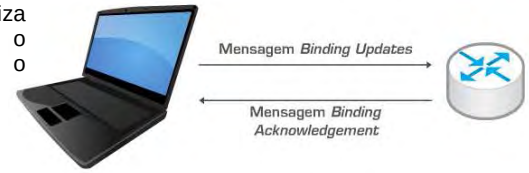
No suporte à mobilidade IPv6 existem alguns componentes chave para o seu funcionamento:

- **Nó Móvel** - dispositivo que pode mudar de uma rede para outra enquanto continua recebendo pacotes através de seu Endereço de Origem;
- **Rede de Origem –** rede que atribui o Endereço de Origem ao Nó Móvel;
- **Agente de Origem** - roteador localizado na Rede de Origem, que mantem a associação entre o Endereço de Origem e o Endereço Remoto do Nó Móvel.
- **Endereço de Origem –** endereço *global unicast* atribuído pela Rede de Origem ao Nó Móvel. É utilizado como endereço permanente, para o qual os pacotes são encaminhados.
- **Rede Remota –** qualquer rede, diferente da origem, onde o Nó Móvel se encontra;
- **Endereço Remoto –** endereço *global unicast* atribuído ao Nó Móvel pela Rede Remota;
- **Nó Correspondente** - nó que se comunica com o Nó Móvel. Este pode ser móvel ou estacionário.

nic.br Núcleo de Informação e Coordenação do Ponto BR — ceptro.br IPv6.br

Mobilidade IPv6

- Funcionamento
  - O Nó Móvel utiliza Endereço de Origem para receber pacotes na Rede de Origem.
  - Deslocamento
    - Adquire Endereço Remoto via autoconfiguração *stateless* ou *stateful*.
      - Agente de Origem realiza a associação entre o Endereço Remoto e o Endereço de Origem.



  - O Nó Móvel também pode se registrar diretamente com o Nó Correspondente.

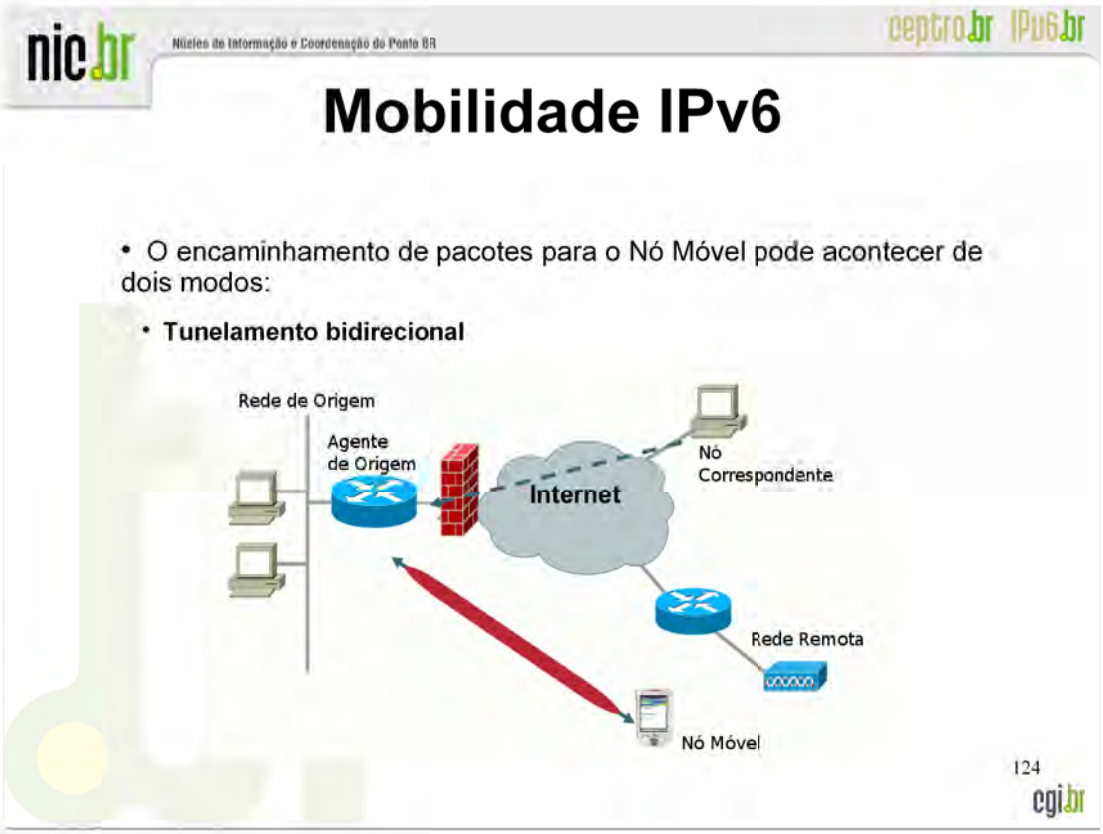
O Nó Móvel possui um Endereço de Origem fixo, que lhe é atribuído pela sua Rede de Origem. Mesmo quando o nó se desloca de sua Rede de Origem, este endereço é mantido.

Ao ingressar em uma Rede Remota, o Nó Móvel recebe um ou mais Endereços Remotos através dos mecanismos de autoconfiguração, constituídos de um prefixo válido na Rede Remota. Para assegurar que os pacotes IPv6 destinados ao seu Endereço de Origem sejam recebidos, o nó realiza uma associação entre o Endereço de Origem e o Endereço Remoto, registrando seu novo endereço no Agente de Origem, através do envio de uma mensagem *Binding Updates*. Como resposta a essa mensagem, o roteador da Rede de Origem envia uma mensagem *Binding Acknowledgement.*

Essa associação de endereços também pode ser feita diretamente com o Nó Correspondente, com o intuito de otimizar a comunicação.

Para o Nó Móvel detectar que retornou a sua rede, ele utiliza o processo de Descoberta de Vizinhos Inacessíveis, para detectar se o seu roteador padrão está ativo. Caso ele localize um novo roteador padrão, ele irá gerar um novo endereço baseado no prefixo anunciado na mensagem RA. No entanto, encontrar um novo roteador padrão não significa necessariamente que ele esteja em uma nova rede, pode ser apenas uma renumeração em sua rede ou a adição de um novo roteador. Com isso, antes de realizar a associação de endereços com o Agente de Origem e com os Nós Correspondentes, o Nó Móvel tenta localizar novamente seu roteador padrão e irá comparar se o intervalo entre o envio de mensagens RA não solicitadas é o mesmo que o configurado em sua Rede Original.

Quando o Nó Móvel retorna a sua Rede de Origem, ele envia uma mensagem *Binding Updates* informando ao Agente de Origem o seu retorno e que este não precisa mais lhe encaminhar os pacotes.



As comunicações entre nós móveis e nós correspondentes podem acontecer de dois modos, tunelamento bidirecional e otimização de rota.

No Tunelamento Bidirecional, os pacotes enviados pelo Nó Correspondente para o Endereço Original do Nó Móvel, são interceptados pelo Agente de Origem, que os encaminhará, através de um túnel, para o Nó Móvel utilizando o Endereço Remoto. Em seguida, o Nó Móvel responde ao Agente de Origem, através do túnel, que reenvia o pacote ao Nó Correspondente. Neste caso, o Nó Correspondente não necessita ter suporte à mobilidade IPv6 e o Nó Móvel não precisa se registrar no Nó Correspondente.

Mobilidade IPv6

- O encaminhamento de pacotes para o Nó Móvel pode acontecer de dois modos:
  - **Otimização de rota**

No modo Otimização de Rota, a comunicação entre o Nó Móvel e o Nó Correspondente ocorre diretamente, sem a necessidade da utilização do Agente de Origem. Para que esta comunicação ocorra, o Nó Móvel registra seu Endereço Remoto no Nó Correspondente, que associa os Endereços de Origem e Remoto do Nó Móvel.

A troca de mensagens entre os dois nós funciona do seguinte modo:

- O Nó Correspondente envia pacotes com o campo Endereço de Destino do cabeçalho base preenchido com o Endereço Remoto do Nó Móvel. O cabeçalho base é seguido pelo cabeçalho de extensão *Routing* Tipo 2, que carrega o Endereço de Origem do Nó Móvel;
- Ao receber o pacote, o Nó Móvel processa o cabeçalho *Routing* e insere o Endereço de Origem do cabeçalho *Routing* no campo Endereço de Destino do cabeçalho base. As camadas superiores continuam o processamento do pacote normalmente;
- Os pacotes enviados pelo Nó Móvel, têm o campo Endereço de Origem do cabeçalho base preenchido com o Endereço Remoto. O cabeçalho base é seguido pelo cabeçalho de extensão *Destination Options*, que carrega na opção *Home Address* o Endereço de Origem do Nó Móvel;
- Ao receber o pacote , o Nó Correspondente insere o Endereço de Origem do cabeçalho *Destination Options* no campo Endereço de Origem do cabeçalho base. As camadas superiores continuam o processamento do pacote normalmente.

nic.br Núcleo de Informação e Coordenação do Ponto BR — ceptro.br IPv6.br

Mobilidade IPv6

- Identificado no campo Próximo Cabeçalho pelo valor 135.
- Utilizado nas trocas de mensagens relacionadas à criação e gerenciamento das associações de endereços.
- Cabeçalho de extensão *Mobility*

Protocolo dos dados	Tam. cab. de extensão	Tipo de Mensagem *Mobility*	Reservado
Soma de Verificação			
Dados			

- Principais tipos de mensagem *Mobility*:
  - *Binding Refresh Request* (Tipo 0)
  - *Binding Update* (Tipo 5)
  - *Binding Ack* (Tipo 6)
  - *Binding Error* (Tipo 7)

126

cgi.br

Para otimizar o funcionamento deste serviço foi adicionado à especificação do IPv6 um novo cabeçalho de extensão, o *Mobility.*

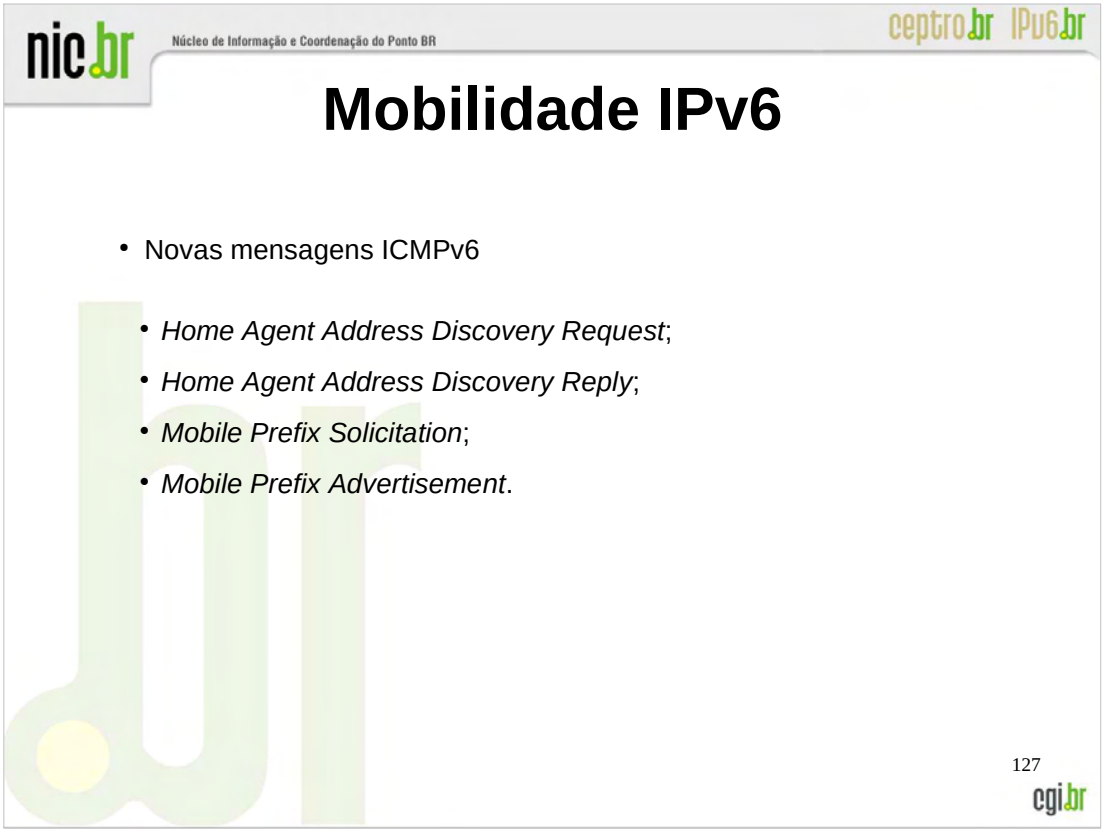
O cabeçalho de extensão *Mobility* é indicado no campo Próximo Cabeçalho pelo valor 135. Ele é utilizado pelo Nó Móvel, pelo Agente Remoto e pelo Nó Correspondente, nas trocas de mensagens relacionadas à criação e gerenciamento das associações de endereços.

Este cabeçalho possui os seguintes campos:

- Protocolo de dados – Corresponde ao campo Próximo Cabeçalho. Atualmente apenas o valor 59, em decimal, é utilizado, indicando que não há próximo cabeçalho;
- Tamanho do cabeçalho de extensão – contem o tamanho do cabeçalho *Mobility* em unidades de 8 Bytes. O tamanho desse cabeçalho deve ser sempre múltiplo de 8;
- Tipo de Mensagem *Mobility* – indica o tipo da mensagem enviada;
- Soma de Verificação – verifica a integridade do cabeçalho *Mobility*;
- Dados – o seu formato e tamanho dependem do tipo de mensagem *Mobility* que está sendo enviada.

Principais tipos de mensagem *Mobility* utilizadas são:

- *Binding Refresh Request* (Tipo 0) – enviada pelo Nó Correspondente, solicitando ao Nó Móvel a atualização da associação de endereços;
- *Binding Update* (Tipo 5) – enviada pelo Nó Móvel notificando ao Agente de Origem ou ao Nó Correspondente sobre um novo Endereço Remoto;
- *Binding Ack* (Tipo 6) – enviada como confirmação de recebimento de uma mensagem *Binding Update;*
- *Binding Error* (Tipo 7) – enviada pelo Nó Correspondente para relatar erros.

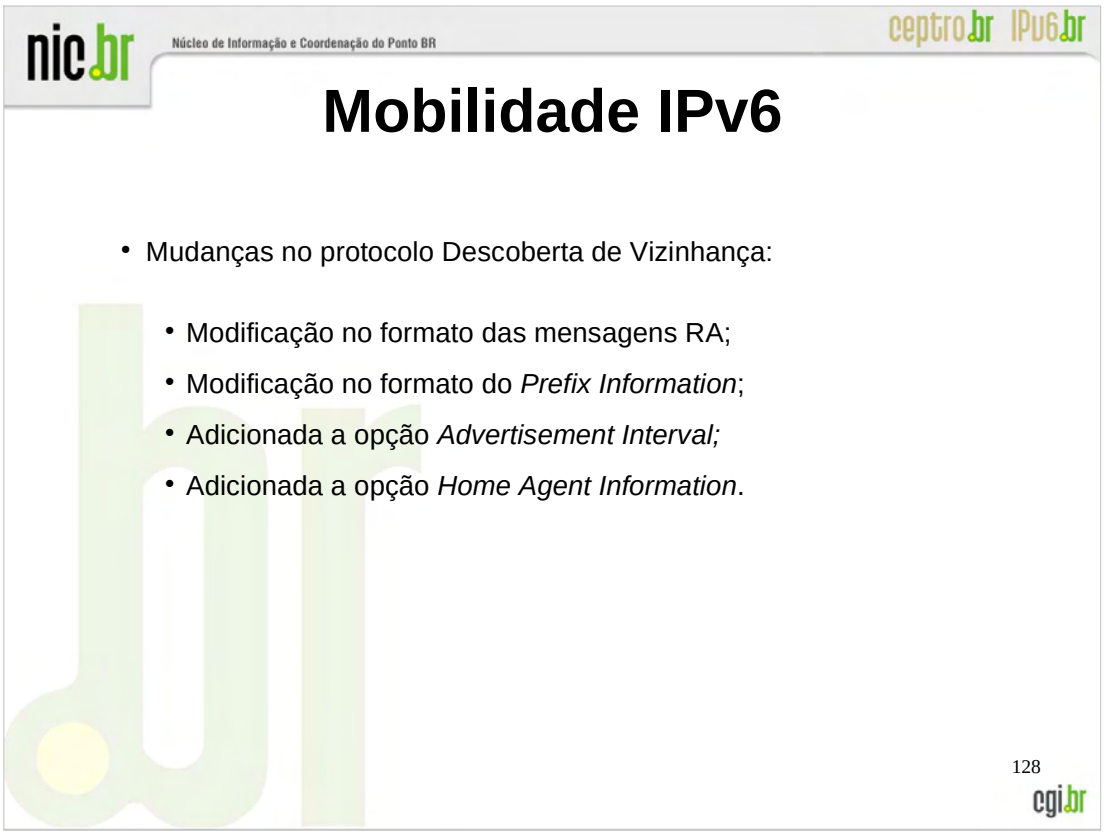


Também foram criadas quatro novas mensagens ICMPv6 utilizadas na configuração de prefixos na Rede de Origem e na descoberta de Agentes de Origem.

O par de mensagens *Home Agent Address Discovery Request* e *Home Agent Address Discovery Reply* são utilizadas para descobrir dinamicamente um Agente de Origem em sua rede. Isto evita a necessidade de configurações manuais e problemas no caso da renumeração do Agente de Origem.

O Nó Móvel envia uma mensagem *Discovery Request* para o endereço *anycast* do Agente de Origem em sua rede. O campo Endereço de origem do cabeçalho base carrega o Endereço remoto do Nó Móvel. O Agente de Origem responde enviando uma mensagem *Discovery Reply*. As mensagens *Discovery Request* e *Reply* são identificadas, respectivamente, pelos valores 150 e 151 no campo Próximo Cabeçalho.

Já a mensagem *Mobile Prefix Solicitation* é enviada pelo Nó Móvel ao Agente de origem, para determinar mudanças nas configurações de prefixo em sua rede. O Agente Remoto responde enviando uma mensagem *Mobile Prefix Advertisement*. Baseado nessa resposta, o Nó Móvel pode ajustar seu Endereço de Origem. As mensagens *Mobile Prefix Solicitation* e *Advertisement*, são identificadas, respectivamente, no campo Próximo Cabeçalho pelo valor 152 e 153.



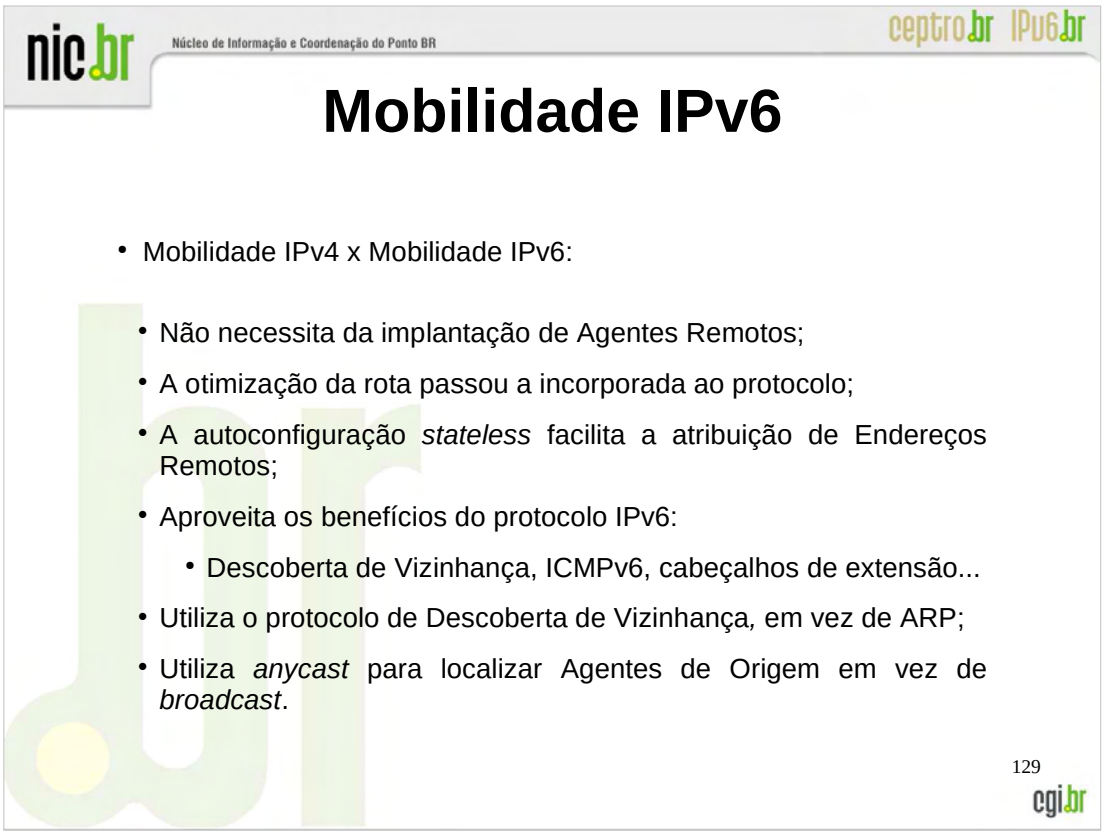
Algumas modificações também foram feitas no protocolo de Descoberta de Vizinhança.

Foi adicionada à mensagem RA a *flag* H, que permite a um roteador anunciar se este atua como um Agente de Origem na rede. A partir desse anuncio, um Nó Móvel forma uma lista de Agentes de Origem da sua rede.

No entanto, para manter essa lista atualizada, o Nó Móvel precisa saber os endereços *global unicast* dos roteadores, porém, as mensagens RA trazem apenas o endereço *link-local*. Para resolver este problema, foi adicionada uma nova *flag à* opção *Prefix Information*, a *flag* R. Quando esta *flag* é marcada, ela indica que a opção *Prefix Information* não contém um prefixo, mas sim o endereço *global unicast* do roteador.

Também foram criadas duas novas opções ao protocolo, a *Advertisement Interval* e a *Home Agent Information*. A primeira indica o intervalo entre mensagens RA não solicitadas, informação esta, que é utilizada no algoritmo de detecção de mudança de rede. Nas especificações do protocolo de Descoberta de Vizinhança, o intervalo mínimo entre o envio dessas mensagens deve ser de três segundos. No entanto, para garantir que o Nó Móvel detecte a mudança de rede e aprenda as informações sobre a nova rede o mais rápido possível, roteadores com suporte à mobilidade IPv6 podem ser configurados com um intervalo de tempo menor para o anuncio de mensagens RA.

A opção *Home Agent Information* é utilizada para indicar o nível de preferencia de associação de cada Agente de Origem.



As principais diferenças entre o suporte à mobilidade do IPv6 e do IPv4 podem ser resumidas nos seguintes tópicos:

- Não há mais a necessidade de se implantar roteadores especiais atuando como agentes remotos;
- A otimização da rota passou a incorporada ao protocolo, em vez de fazer parte de um conjunto de extensões opcionais;
- A autoconfiguração *stateless* facilita a atribuição de Endereços Remotos;
- Aproveita os benefícios do protocolo IPv6 como o protocolo de Descoberta de Vizinhança, as mensagens ICMPv6, e os cabeçalhos de extensão;
- O uso do protocolo de Descoberta de Vizinhança, em vez de ARP, permite que o processo de interceptação dos pacotes destinados ao nó móvel não dependa da camada de enlace, simplificando o protocolo e aumentando sua eficiência;
- A busca por agentes de origem realizada pelo nó móvel passou a ser feita utilizando *anycast*. Desta forma, o nó móvel receberá apenas a reposta de um único agente de origem. Com o IPv4, utiliza-se *broadcast*, o que implica em uma resposta separada para cada agente de origem existente.

Mais informações:

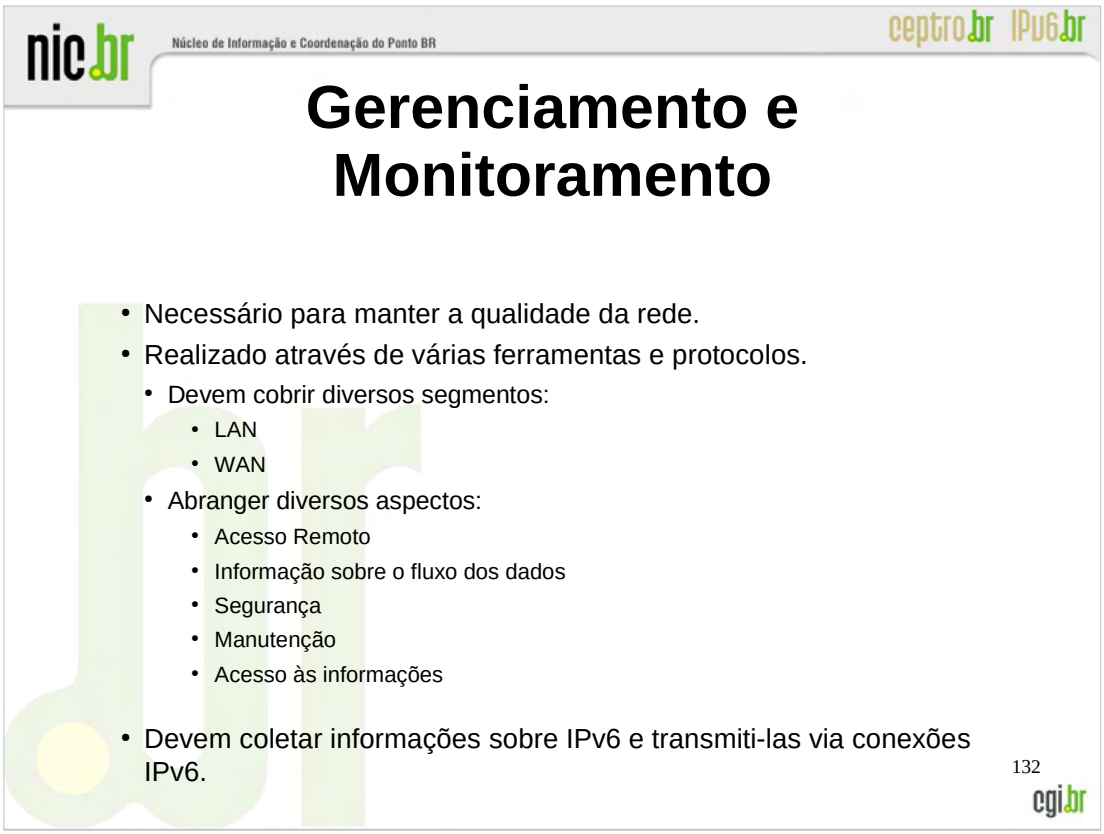
- RFC 3775 - *Mobility Support in IPv6*

IPv6.br

A Nova Geração do Protocolo Internet

A utilização de protocolos e ferramentas de gerenciamento e monitoramento de redes é muito importante para a manutenção da qualidade e obtenção do máximo desempenho desta rede. Com a adoção do novo Protocolo Internet, é preciso conhecer quais dessas ferramentas são capazes de coletar informações sobre IPv6 e se estão aptas a obtê-las através da rede IPv6.

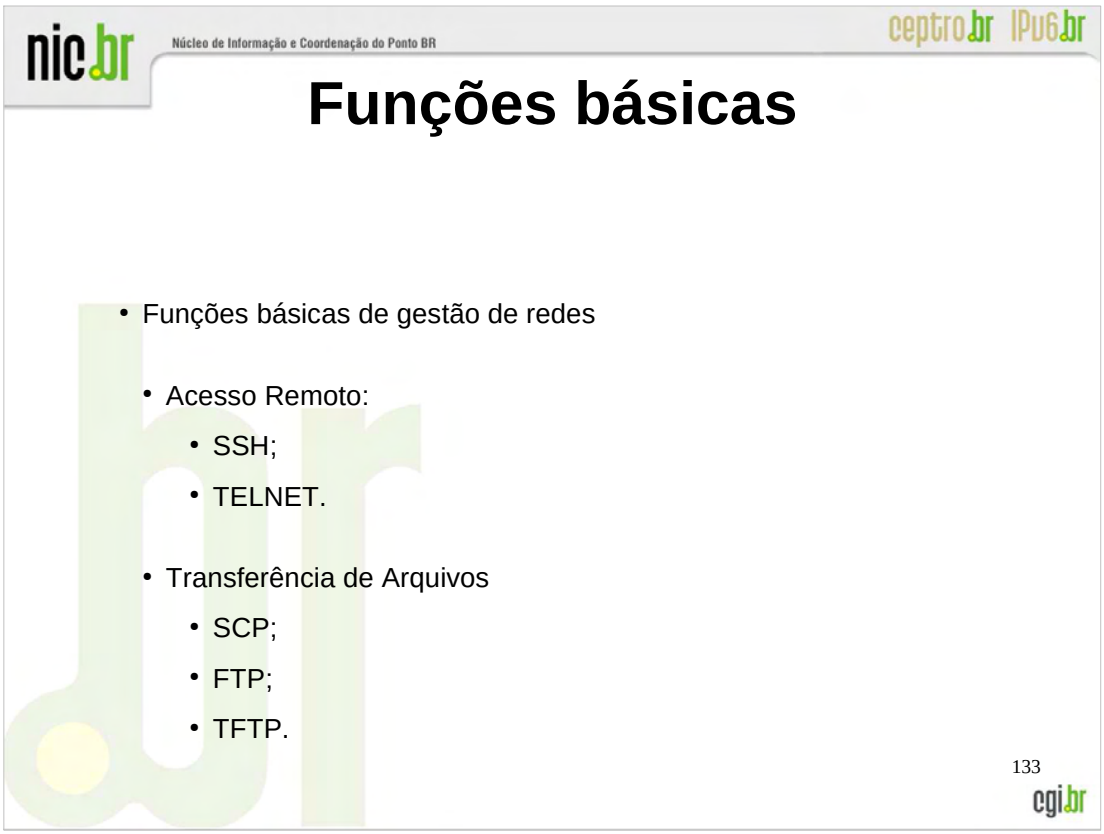
Neste módulo estes aspectos serão abordados em relação a alguns protocolos utilizados para estes fins como SSH, FTP, SNMP, entre outros, e alguns aplicativos como Argus, Nagios, MRTG, Rancid, Wireshark e Looking Glass.



Necessária para manter a qualidade e garantir o máximo de eficiência de seu funcionamento, o gerenciamento e o monitoramento de redes de computadores é uma parte importante na operação, independente do tamanho dessa rede.

Atualmente existem inúmeras ferramentas e protocolos que realizam essas funções, que podem diferenciar-se de acordo com o segmento de rede que atuam, sejam LANs ou WANs, e funcionalidades como garantir acesso remoto e seguro à nós da rede, coletar informações sobre o fluxo dos dados, autenticação de usuários, testes e manutenções e acesso as informações.

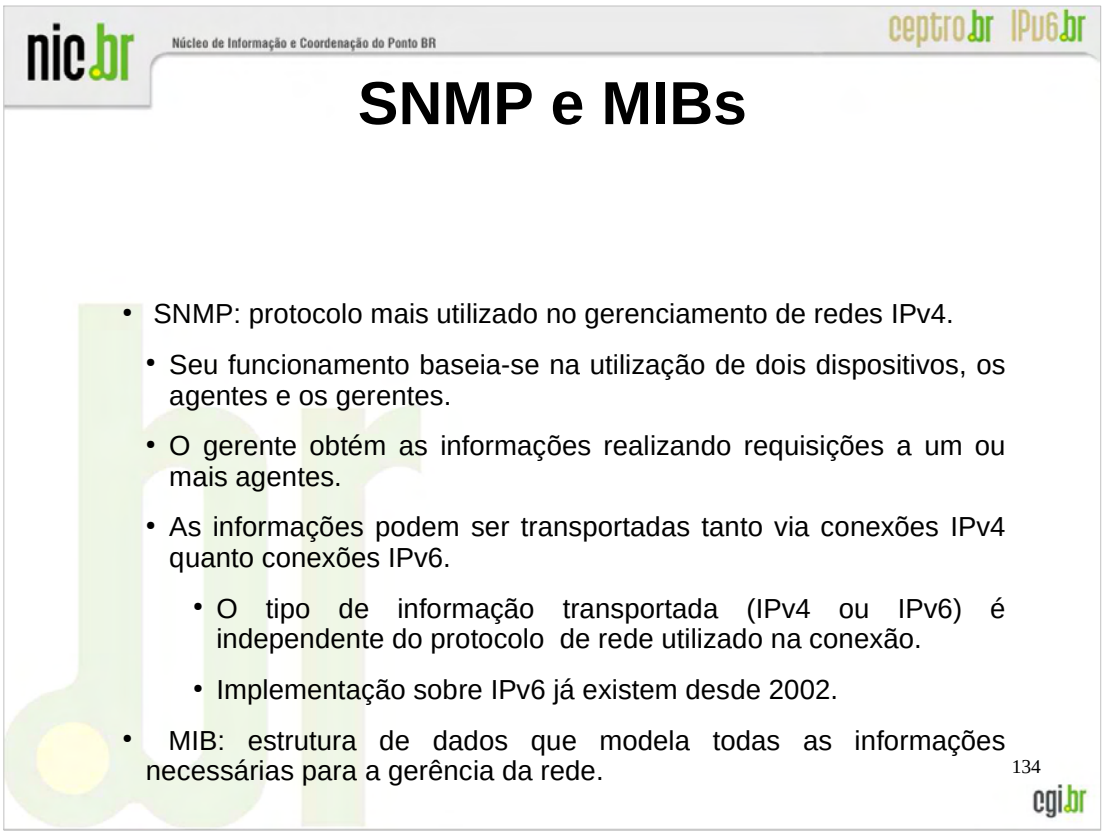
Portanto, neste momento de transição entre o IPv4 e o IPv6, é importante que essas ferramentas sejam capazes de suportar as duas versões do protocolo IP, estando aptas a coletar informações sobre IPv6 e transmiti-las via conexões IPv6.



Umas das funções mais básicas de gestão de redes, é o acesso remoto a outros dispositivos. Neste aspecto, os principais protocolos existentes já são capazes de operar sobre IPv6.

Os protocolos Telnet e SSH (Secure Shell), utilizados para estabelecer conexões remotas a outros dispositivos da rede, já permitem o acesso via conexões IPv6. Aplicativos como OpenSSH e PuTTy, por exemplo, já oferecem essa funcionalidade.

Do mesmo modo, realizar a transferência de arquivos entre dispositivos remotos via IPv6, já é possível através de protocolos como SCP, TFTP e FTP. O FTP inclusive, foi um dos primeiros protocolos a serem adaptados para trabalhar sobre IPv6.

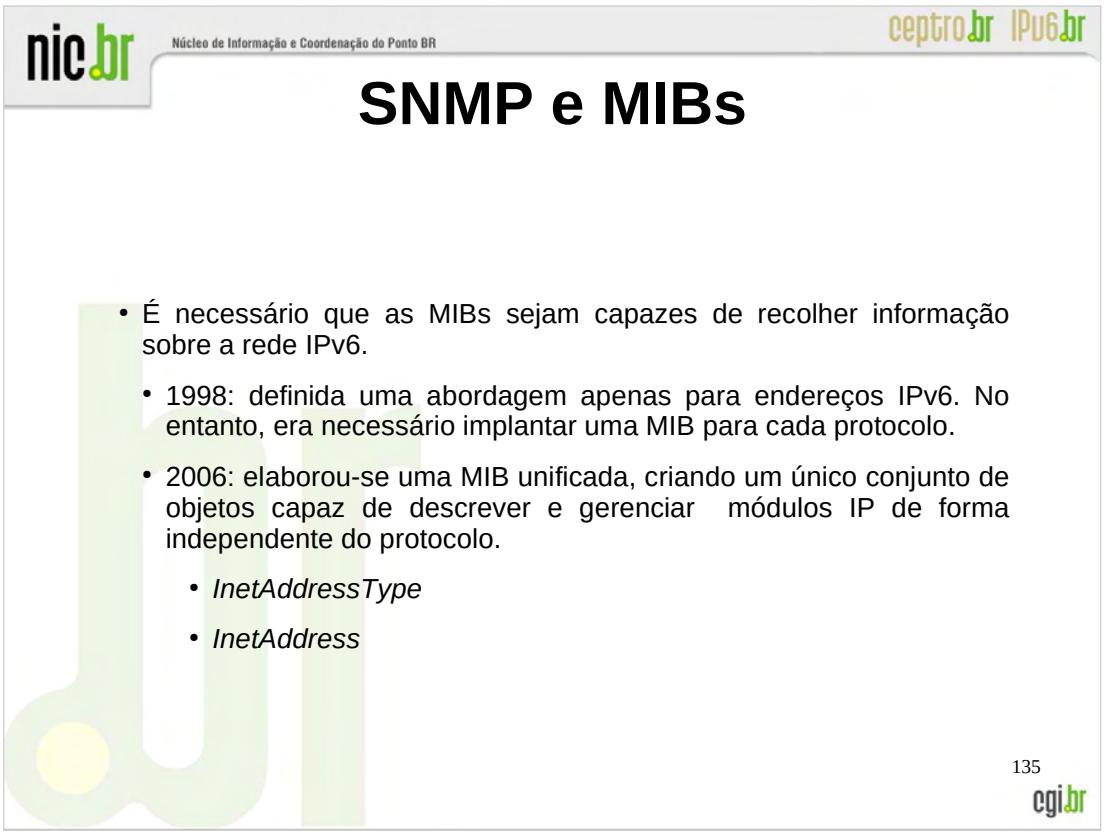


Em redes IPv4, o protocolo SNMP (*Simple Network Management Protocol*), definido na RFC 1157, é uma das ferramentas de gerenciamento mais utilizadas, devido a sua flexibilidade e facilidade de implantação.

O funcionamento do SNMP baseia-se na utilização de dois dispositivos, um agente e um gerente. Cada dispositivo gerenciado deve possuir um agente e uma base de dados referente ao seu estado atual, que pode ser consultada e alterada pelo gerente. O conjunto desses dados, ou objetos gerenciados, é conhecido como MIB (*Management Information Base*), uma estrutura de dados que modela todas as informações necessárias para a gerência da rede.

O agente é o responsável pela manutenção das informações gerenciadas. Ele é quem deve responder as requisições feitas pelo gerente, lhe enviando as informações necessárias para que este possa realizar o monitoramento do sistema.

O envio dessas informações, armazenadas nas MIBs, pode ser realizado tanto via conexões IPv4 ou IPv6, visto que, o tipo de informação transportada é independente do protocolo de rede utilizado. No entanto, já existe desde de 2002, implementações de SNMPs capazes de monitorar redes que possuam somente conexões IPv6.



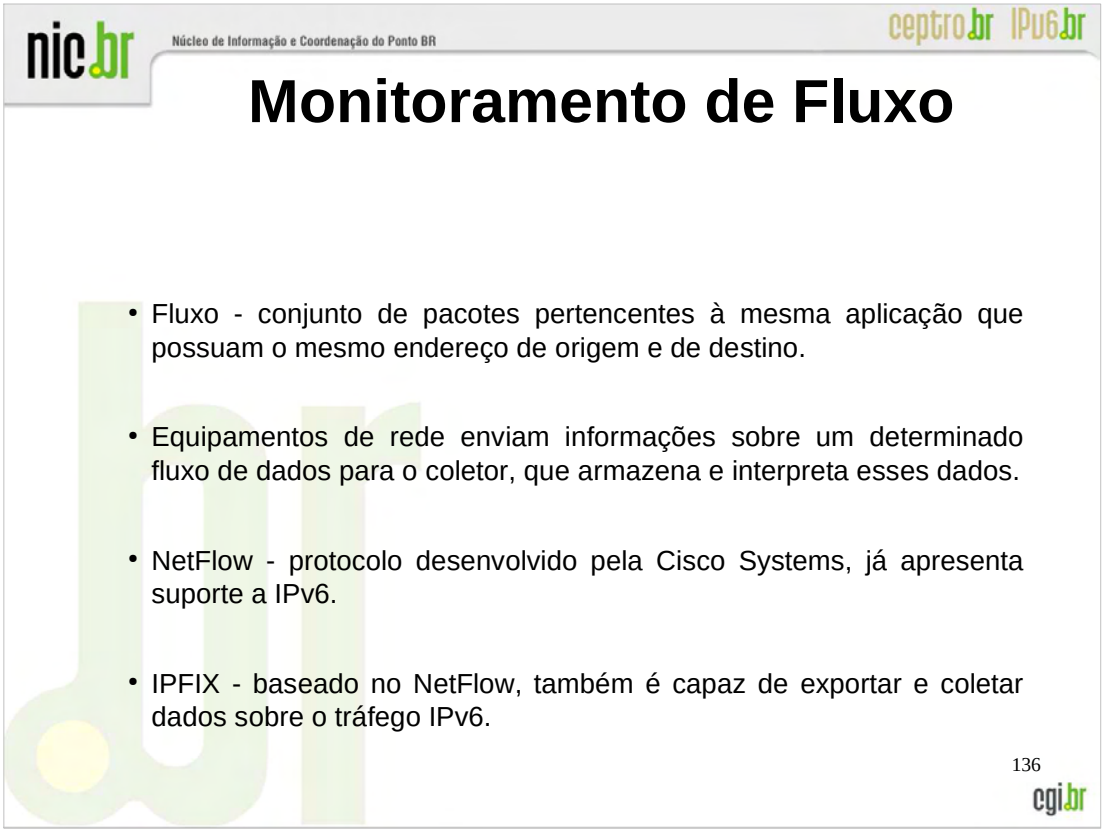
Embora o tipo de protocolo de rede utilizado na transmissão dos dados não interfira no envio das mensagens SNMP, é necessário que as MIBs sejam capazes de armazenar informações sobre a rede IPv6.

Em vista disso, em 1998, na RFC 2465 (tornou-se obsoleta pela RFC 4292), foi definida uma abordagem apenas para endereços IPv6. No entanto, era necessário implantar uma MIB para cada protocolo. Em 2002, a RFC 2851 (tornou-se obsoleta pelas RFCs 3291 e 4001), estabeleceu uma MIB unificada, criando um único conjunto de objetos capaz de descrever e gerenciar módulos IP de forma independente do protocolo.

Esta nova convenção define um endereço IP como uma estrutura {*inetAddressType*, *inetAddress*}, onde o primeiro, é um valor inteiro que determina a forma como o segundo será codificado.

Mais informações:

- RFC 1157 - *A Simple Network Management Protocol* (SNMP)
- RFC 4001 - *Textual Conventions for Internet Network Addresses*
- RFC 4292 - *IP Forwarding Table MIB*



Para análises mais detalhadas de uma rede, podemos aplicar uma abordagem alternativa que consiste em recolher informações sobre cada pacote. Neste método, equipamentos de rede, por exemplo, um roteador, enviam periodicamente informações sobre um determinado fluxo de dados para um dispositivo chamado coletor, que armazena e interpreta esses dados.

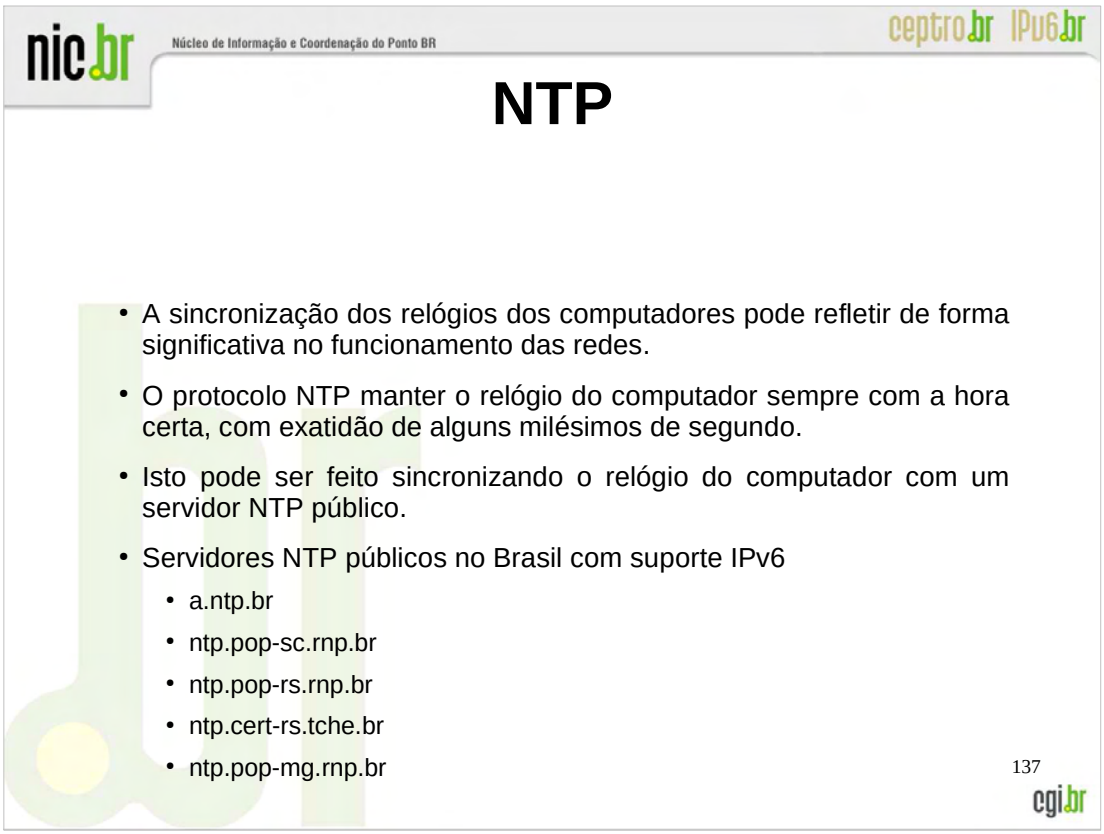
Um fluxo de dados pode ser definido como um conjunto de pacotes pertencentes à mesma aplicação que possuem o mesmo endereço de origem e destino. Os principais protocolos utilizados para a transmissão de informações sobre um fluxo IP de uma rede, também já estão preparados para coletar dados sobre o tráfego IPv6.

O NetFlow, protocolo desenvolvido pela Cisco Systems definido na RFC 3954, é uma eficiente ferramenta utilizada, entre outras coisas, para contabilização e caracterização de tráfego de redes, planejamento de redes e detecção de ataques DoS e DDoS. O protocolo NetFlow com suporte IPv6 encontra-se implementado a partir do Cisco IOS 12.3(7)T, no entanto, esta implementação ainda utiliza o protocolo IPv4 para a exportação de dados.

Do mesmo modo, o protocolo IPFIX (*IP Flow Information Export*), proposto pela IETF na RFC 3917, também é capaz de exportar e coletar dados sobre o tráfego IPv6.

Mais informações:

- RFC 3917 - *Requirements for IP Flow Information Export* (IPFIX)
- RFC 3954 - Cisco Systems NetFlow Services Export V9



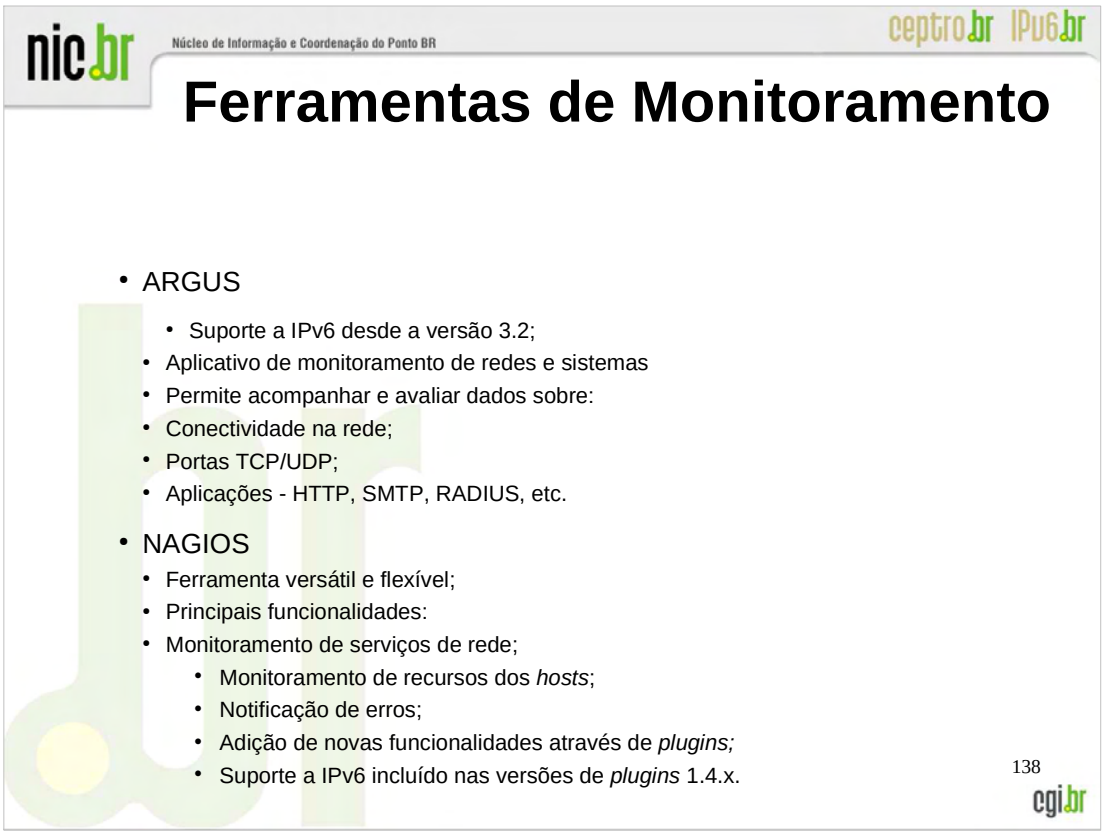
Um quesito muito importante no gerenciamento de redes, a sincronização dos relógios dos computadores pode refletir de forma significativa no funcionamento de diversos *softwares* e sistemas, além da segurança dos computadores, redes e da própria Internet. Com a utilização do protocolo NTP (*Network Time Protocol*) é possível manter o relógio do computador sempre com a hora certa, com exatidão de alguns milésimos de segundo. Isto pode ser feito sincronizando o relógio do computador com um servidor NTP público.

A sincronização dos relógios de uma rede IPv6 é possível conectando-se a servidores que possuam suporte ao novo protocolo IP. A lista a seguir apresenta o endereço de uma série de servidores NTP públicos no Brasil que já trabalham com conexão IPv6:

- a.ntp.br
- ntp.pop-sc.rnp.br
- ntp.pop-rs.rnp.br
- ntp.cert-rs.tche.br
- ntp.pop-mg.rnp.br

Mais informações:

- RFC 1305 - *Network Time Protocol (Version 3) Specification, Implementation and Analysis*



Existem diversas ferramentas que auxiliam no monitoramento de uma rede. Utilitários para gerenciamento de tráfego, elaboração de gráficos e relatórios sobre o status de equipamentos e links, análise de tráfego e diversas outras tarefas, são utilizados frequentemente por administradores de redes, sendo que muitas dessas ferramentas já possuem suporte o IPv6. Entre essas ferramentas podemos destacar:

- ARGUS – aplicativo de monitoramento de redes e sistemas, que permite acompanhar e avaliar dados referentes à conectividade na rede, portas TCP/UDP e de aplicações como HTTP, SMTP, RADIUS, etc.. Apresenta suporte a IPv6 desde a versão 3.2;
- NAGIOS - ferramenta versátil e flexível que apresenta inúmeras funcionalidades como monitoramento de serviços de rede; de recursos dos *hosts*; notificação de erros; etc.. Possui a vantagem da possibilidade de adição de novas funcionalidades através de *plugins*. O suporte IPv6 foi incluído nas versões de *plugins* 1.4.x;

Ferramentas de Monitoramento

- NTOP
  - Detalhar a utilização da rede;
  - Visualização de estatísticas do tráfego;
  - Análise do tráfego IP;
  - Detecção de violações de segurança;
  - Possui suporte a tráfego IPv6.
- MRTG
  - Desenvolvido em C e Perl;
  - Utiliza SNMP para obter informações dos dispositivos gerenciados;
  - Análisde dos dados através de gráficos visualizados em formato HTML;
  - Suporte a IPv6 desde a versão 2.10.0.

- NTOP (*Network Traffic Probe*) - capaz de detalhar a utilização da rede por *host*, protocolo, etc., permitindo a visualização de estatísticas do tráfego, análise do tráfego IP, detecção de violações de segurança na rede, entre outras funções. Possui suporte a tráfego IPv6;
- MRTG (*Multi Router Traffic Grapher*) - Desenvolvido em C e Perl, utiliza o SNMP para obter informações de tráfego dos dispositivos gerenciados. Todos os dados obtidos através do protocolo SNMP podem ser monitorados por esta ferramenta e analisados através de gráficos visualizados em formato HTML. Suporte a IPv6 desde a versão 2.10.0.

Ferramentas de Monitoramento

- Pchar
  - Ferramenta de avaliação de performance;
  - Análise de largura de banda;
  - Análise de latência
  - Análise de perda de conexões;
  - Permite a análise de redes IPv6.
- Rancid
  - Monitora configurações de equipamentos;
  - Desenvolvida nas linguagens Perl, Shell e C;
  - Disponibiliza um *looking glass;*
  - É capaz de caracterizar o caminho entre dois *hosts* em redes IPv6.

- Pchar – ferramenta de avaliação de performance da rede. Analisa aspectos como largura de banda, latência e perda de conexões. Permite a análise de redes IPv6.
- Rancid – permite o monitoramento de configurações de equipamentos (*software* e *hardware*), utilizando CVS. Desenvolvida nas linguagens Perl, Shell e C, disponibiliza além das funcionalidades tradicionais de uma ferramenta de monitoramento de rede, possui um *looking glass*. É capaz de caracterizar o caminho entre dois *hosts* em redes IPv6.

Ferramentas de Monitoramento

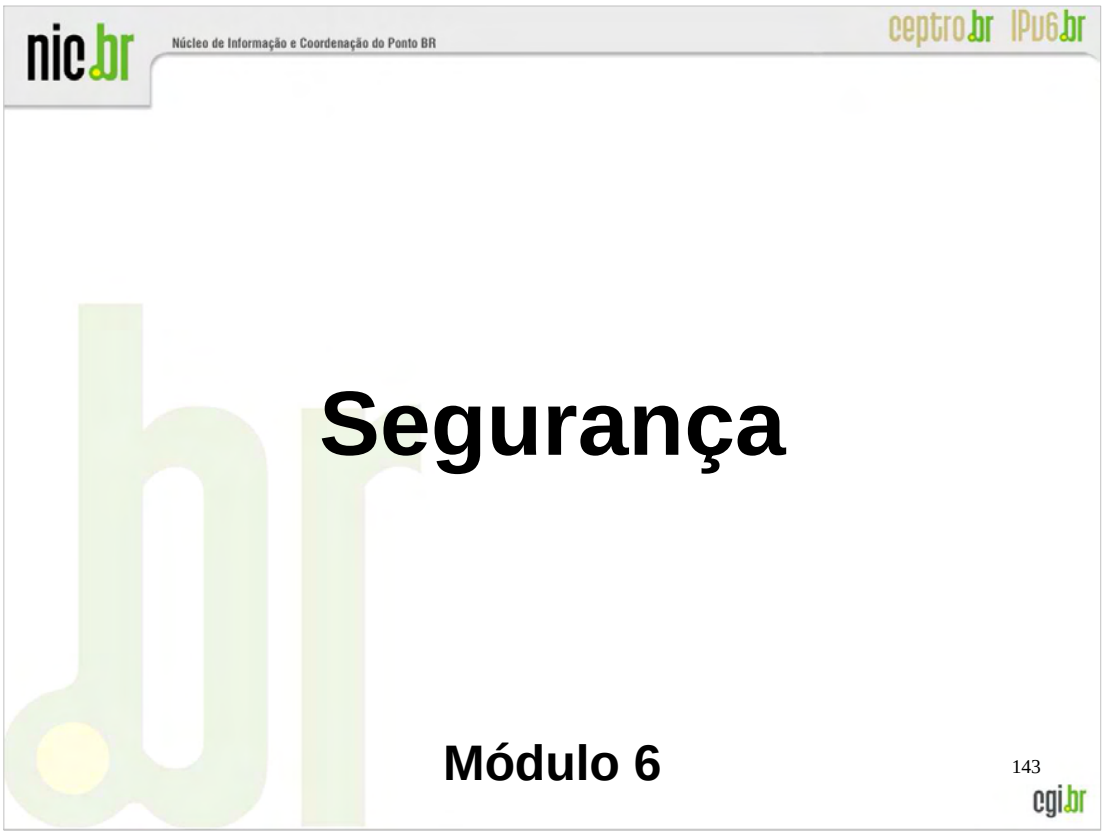
- Wireshark
  - Analisador de tráfego de rede (*sniffer*);
  - Possui interface gráfica
  - Apresenta informação sobre:
  - Árvore de protocolos do pacote
  - Conteúdo dos pacotes;
  - Permite a captura de pacotes IPv6.
- Looking Glass
  - Permite a obtenção de informações sobre roteadores sem necessidade de acesso direto ao equipamento;
  - Pode ser acessado através de uma interface Web, facilitando o diagnostico de problemas na rede;
  - Permite o acesso via conexões IPv6.

- Wireshark - analisador de tráfego de rede (*sniffer*) através da captura de pacotes. Possui interface gráfica e apresenta informação sobre a árvore de protocolos do pacote e o seu conteúdo. Permite a captura de pacotes IPv6;

- Looking Glass – permite a obtenção de informações sobre um roteador sem necessidade de acesso direto ao equipamento. Pode ser acessado através de uma interface Web, facilitando o diagnostico de problemas na rede. Permite o acesso via conexões IPv6.

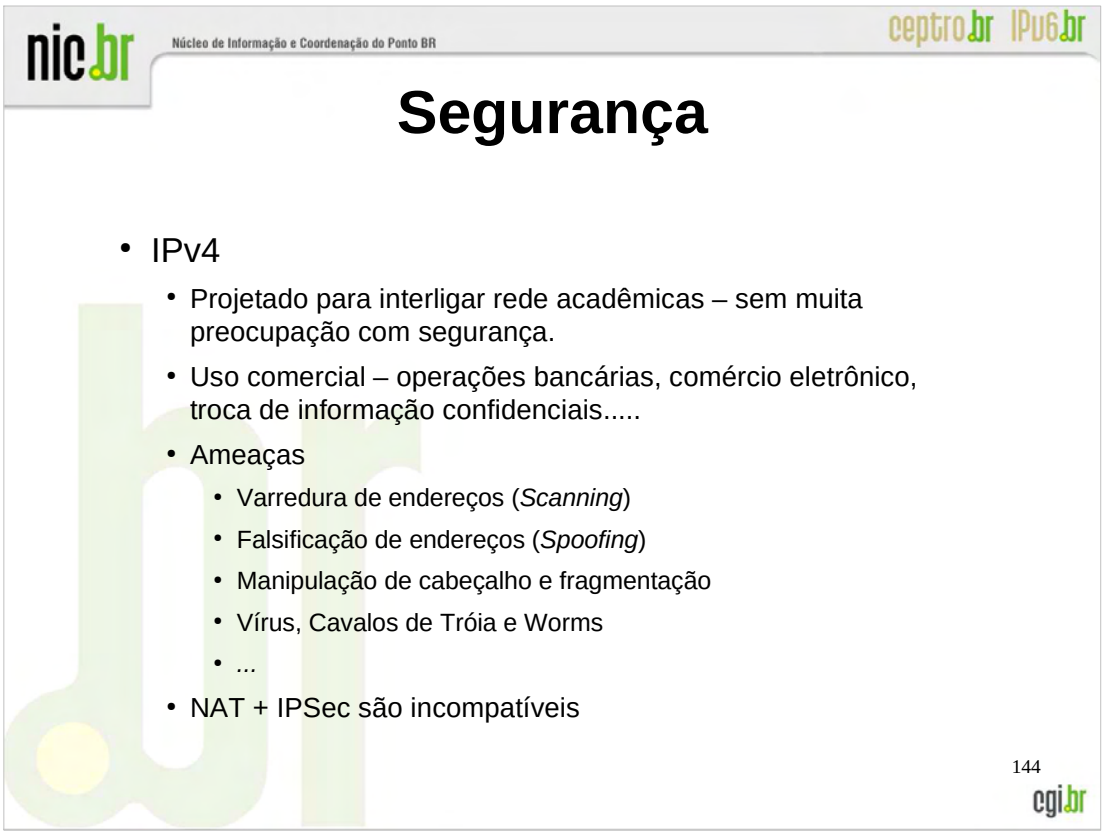
IPv6.br

A Nova Geração do Protocolo Internet



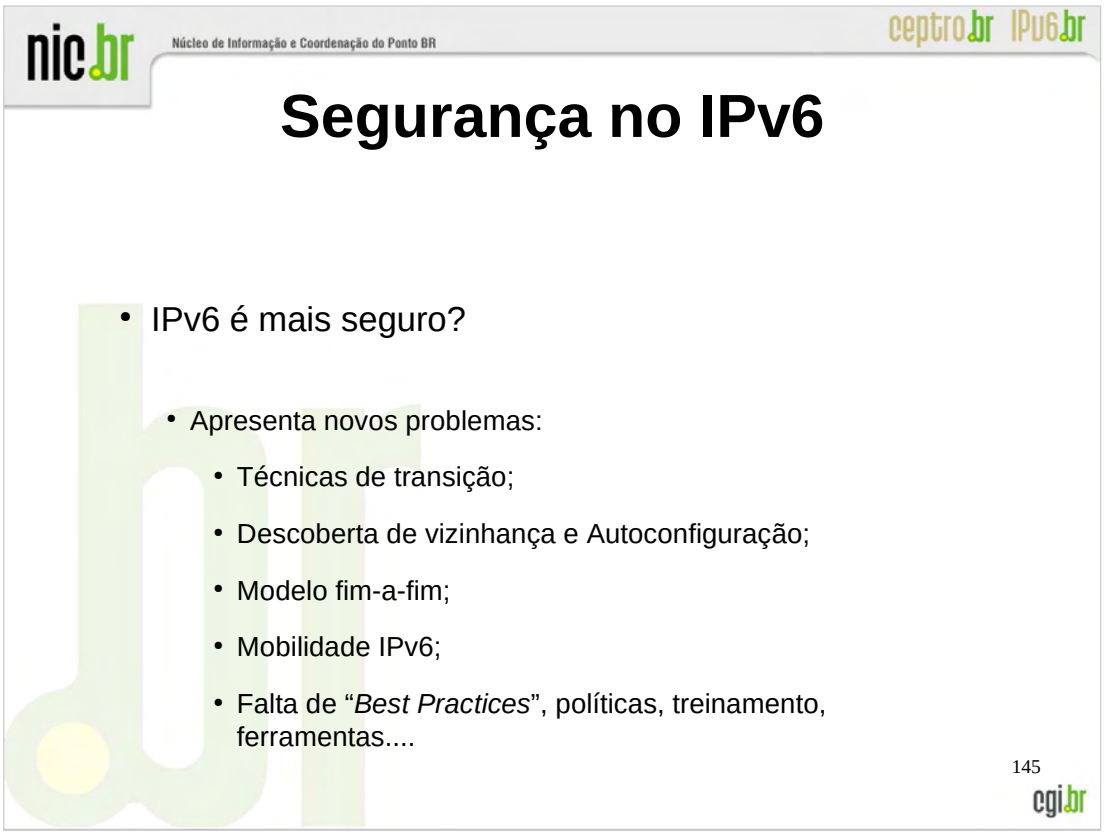
No projeto do IPv6 a questão da segurança foi pensada desde o início. Mecanismos de autenticação e encriptação passaram a fazer parte do protocolo IPv6, disponibilizando para qualquer par de dispositivos de uma conexão fim-a-fim, métodos que visam garantir a segurança dos dados que trafegam pela rede. No entanto, inúmeros problemas ainda existem e novas falhas de segurança também surgiram.

Neste módulo abordaremos cada um desses novos cenários, analisando as novas ferramentas de segurança do IPv6 e traçando um paralelo entre as ameaças existentes no IPv4 e em seu sucessor.

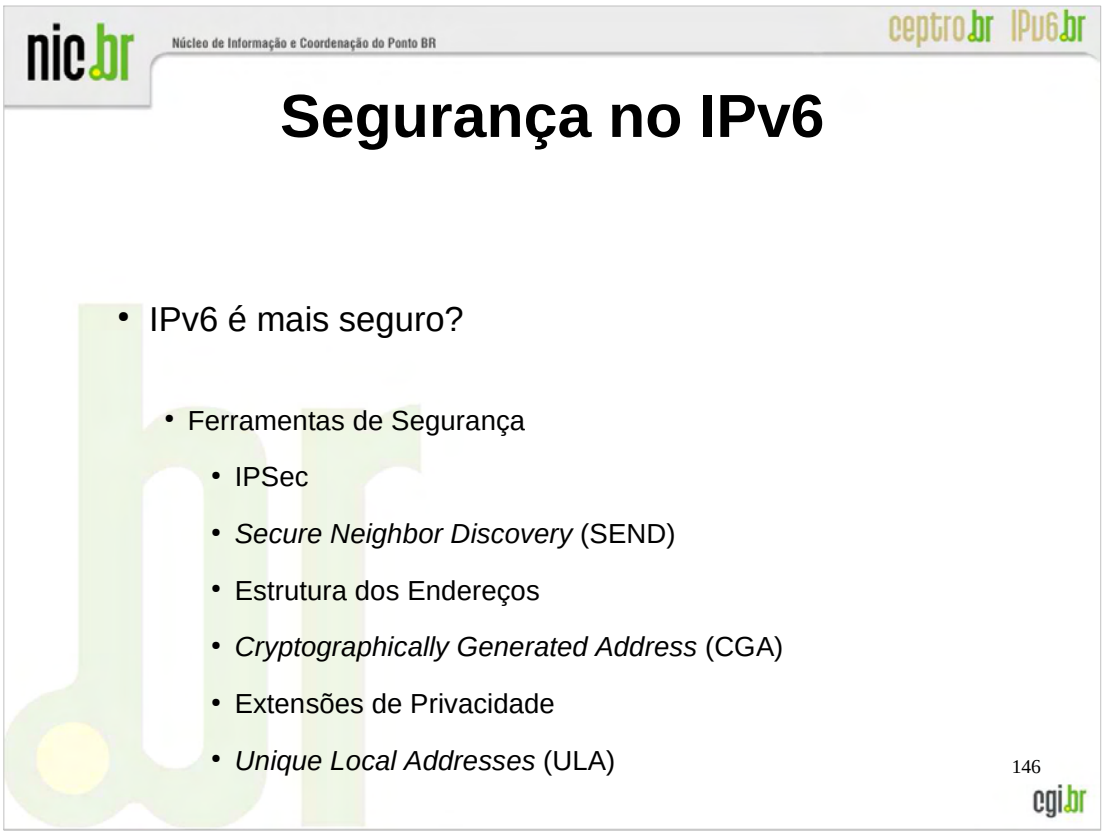


Destinado principalmente a interligar redes de pesquisa acadêmicas, o projeto original do IPv4 não apresentava nenhuma grande preocupação com a questão da segurança das informações transmitidas. No entanto, o aumento da importância da Internet para a realização de transações entre empresas e consumidores, por exemplo, fez com que um nível maior de segurança passasse a ser exigido, como identificação de usuários e criptografia de dados, tornando necessário anexar novos mecanismos ao protocolo original, que garantissem tais serviços. Contudo, as soluções adotadas normalmente são específicas para cada aplicação.

Este fato é bastante claro na Internet atual. Há quem defenda que deveria haver uma nova Internet, projetada do zero (abordagem *clean slate*) para resolver esse problema.



Alguns aspectos de segurança foram abordados no projeto do IPv6, mas suas implementações ainda estão imaturas. Apesar de ter mais de dez anos, não há boa experiência de uso. As melhores práticas ainda são adaptadas do IPv4, e isso nem sempre funciona bem.



No IPv6 a segurança foi uma preocupação desde o início, várias ferramentas de segurança foram implementadas no protocolo.

Mais informações:

- RFC 4864 - *Local Network Protection for IPv6*

nic.br Núcleo de Informação e Coordenação do Ponto BR — ceptro.br IPv6.br

Segurança no IPv6

- Estratégia de Implantação
  - Sem planejamento... Ex:
    - Roteadores *wireless*
      Sistemas sem *firewall*
      Projetos de última hora / demonstrações
      Projetos sem o envolvimento de especialistas em segurança
  - Ou...
    - *Planejamento (Plan)*
    - *Implantação (Do)*
    - *Verificação (Check)*
    - *Ação (Act)*

cgi.br

Antes de entrar em detalhes sobre as ferramentas, vamos pensar um pouco sobre a estratégia de implantação do IPv6.

Podemos pensar em vários exemplos históricos de implantações de tecnologias novas em que não houve uma preocupação com aspectos de segurança desde o início, como quando surgiram os roteadores *wireless*. Podemos pensar, certamente, em vários exemplos do dia a dia, de projetos de última hora, demonstrações para clientes, que acabam tornando-se sistemas em produção e apresentando várias vulnerabilidades.

O ideal é que a implantação do IPv6 seja feita de uma forma planejada e organizada, com a preocupação com a segurança presente desde o início.

Este e os slides seguintes foram retirados da apresentação de Joe Klein, na Google IPv6 *Implementors Conference* 2009. O original pode ser encontrado em: https://sites.google.com/site/ipv6implementors/conference2009/agenda/03_Klein_IPv6_Security.pdf?attredirects=0

nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

Desenvolvimento de Sistemas com IPv6 Habilitado

Data	Produtos	Suporte ao IPv6	IPv6 Habilitado
1996	OpenBSD / NetBSD / FreeBSD	Sim	Sim
	Linux Kernel 2.1.6	Sim	Não
1997	AIX 4.2	Sim	Não
2000	Windows 95/98/ME/NT 3.5/NT 4.0	Sim (pacotes adicionais)	Não
	Windows 2000	Sim	Não
	Solaris 2.8	Sim	Sim
2001	Cisco IOS (12.x e superior)	Sim	Não
2002	Juniper (5.1 e superior)	Sim	A maioria
	IBM z/OS	Sim	Sim
	Apple OS/10.3	Sim	Sim
	Windows XP	Sim	Não
	Linux Kernel 2.4	Sim	Não
	AIX 6	Sim	Sim
	IBM AS/400	Sim	Sim
2006	Roteadores Linksys (Mindspring)	Sim	Não
	Telefones Celulares (Vários)	Sim	Sim
	Solaris 2.10	Sim	Sim
	Linux Kernel 2.6	Sim	Sim
2007	Apple Airport Extreme	Sim	Sim
	BlackBerry (Telefone Celular)	Sim	Não
	Windows Vista	Sim	Sim
	HP-UX 11iv2	Sim	Sim
	Open VMS	Sim	Sim
	Mac OS/X Leopard	Sim	Sim
2009	Cloud Computing e Sistemas embarcados	Sim	Sim

cgi.br

É interessante notar quantos sistemas são capazes de executar o IPv6, e que muitos deles vêm com o protocolo habilitado por padrão. A lista inclui Sistemas Operacionais, telefones celulares, equipamentos de redes, entre outros.

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Incidentes de segurança no IPv6

2001	Revisão de logs, após anuncio do Projeto Honeynet
2002	Projeto Honeynet: Lance Spitzner: Solaris Snort: Martin Roesch: IPv6 adicionado, depois removido
2003	Worm: W32.HLLW.Raleka: Download de arquivos de um local pré-definido e conecta em um IRC server
2005	Trojam: Troj/LegMir-AT: Conecta em um IRC server CERT: Backdoors usando Teredo IPv6 Mike Lynn: Blackhat: captura de pacotes IPv6
2006	CAMSECWest: THC IPv6 Hacking Tools RP Murphy: DefCon: Backdoors IPv6
2007	Rootkit: W32/Agent.EZM!tr.dldr: TCP HTTp SMTP James Hoagland: Blackhat: falha relatada no Teredo IPv6 do Vista
2008	HOPE: Vulnerabilidade em telefones móveis com IPv6 Novembro: “Atacantes estão tentando ou usando-o como mecanismo de transporte para botnets. IPv6 tornou-se um problema do lado operacional.”Arbor Networks

Incidentes de Segurança envolvendo IPv6 vêm sendo reportados há tempos.

Ex:

From: Lance Spitzner <lance_at_honeynet.org>
Date: Tue, 17 Dec 2002 20:34:33 -0600 (CST)

Recently one of the Honeynet Project's Solaris Honeynets was compromised.

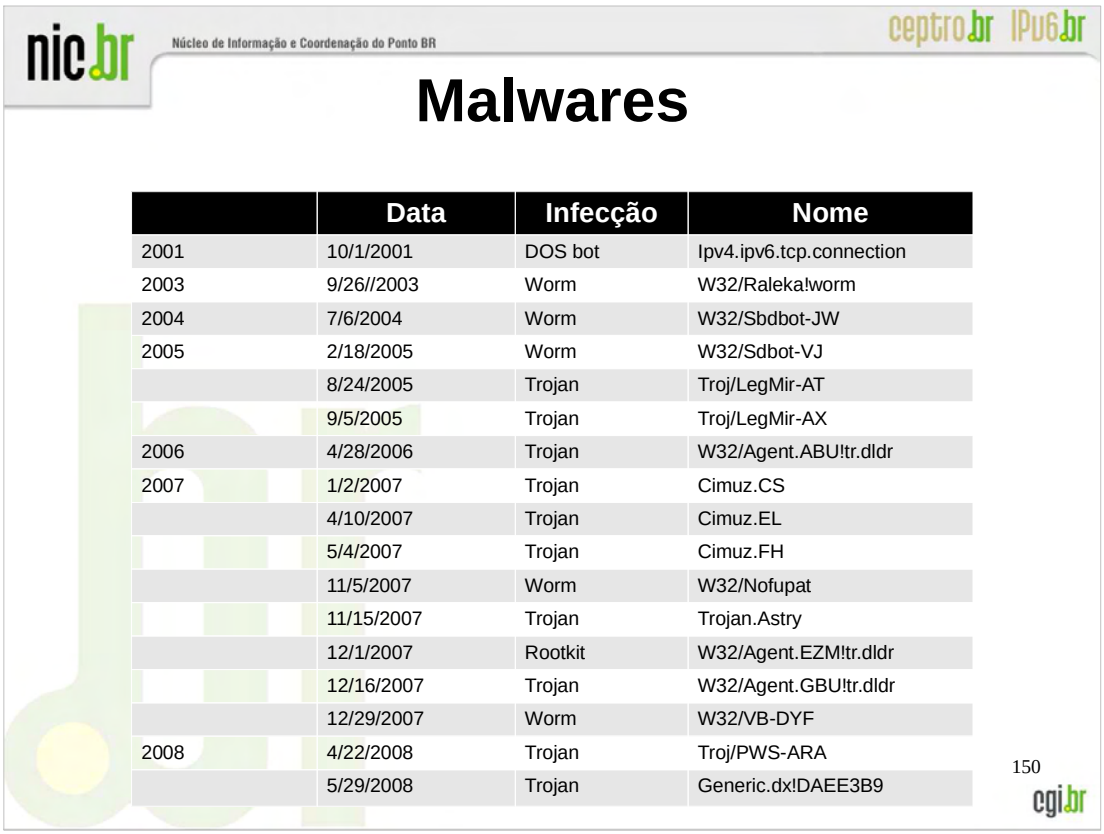
What made this attack unique was after breaking into the system, the attackers enabled IPv6 tunneling on the system, with communications being forwarded to another country. The attack and communications were captured using Snort, however the data could not be decoded due to the IPv6 tunneling. Also, once tunneled, this could potentialy disable/bypass the capabilities of some IDS systems.

Marty is addressing this issue and has added IPv6 decode support to Snort. Its not part of Snort current (2.0) yet, its still in the process of testing. If you would like to test this new capability, you can find it online at http://www.snort.org/~roesch/

Marty's looking for feedback. As IPv6 usage spreads, especially in Asia, you will want to be prepared for it. Keep in mind, even in IPv4 environments (as was our Solaris Honeynet) attackers can encode their data in IPv6 and then tunnel it through IPv4. We will most likely being seeing more of this type of behavior.

Just a friendly heads-up :)

-- Lance Spitzner http://www.tracking-hackers.com

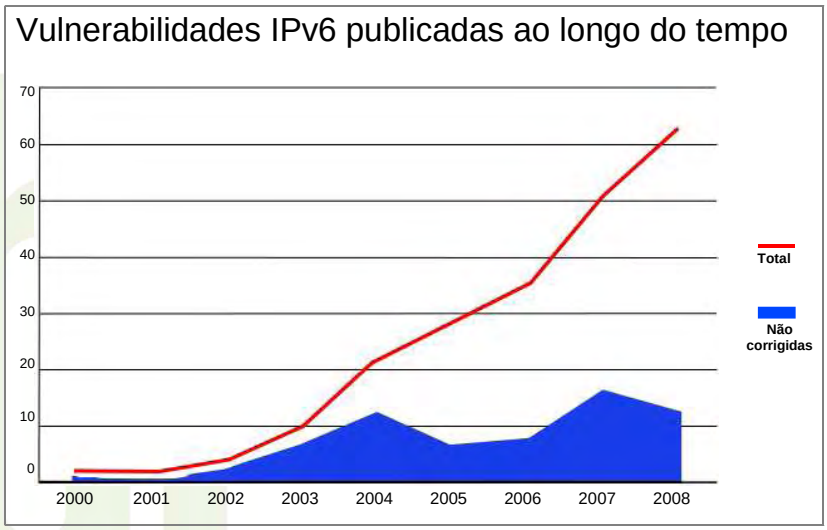


Da mesma forma, *malwares* que fazem uso do IPv6 ou atacam sistemas IPv6 vêm sendo reportados pelo menos desde 2001.

nic.br Núcleo de Informação e Coordenação do Ponto BR

cetro.br IPv6.br

Vulnerabilidades IPv6



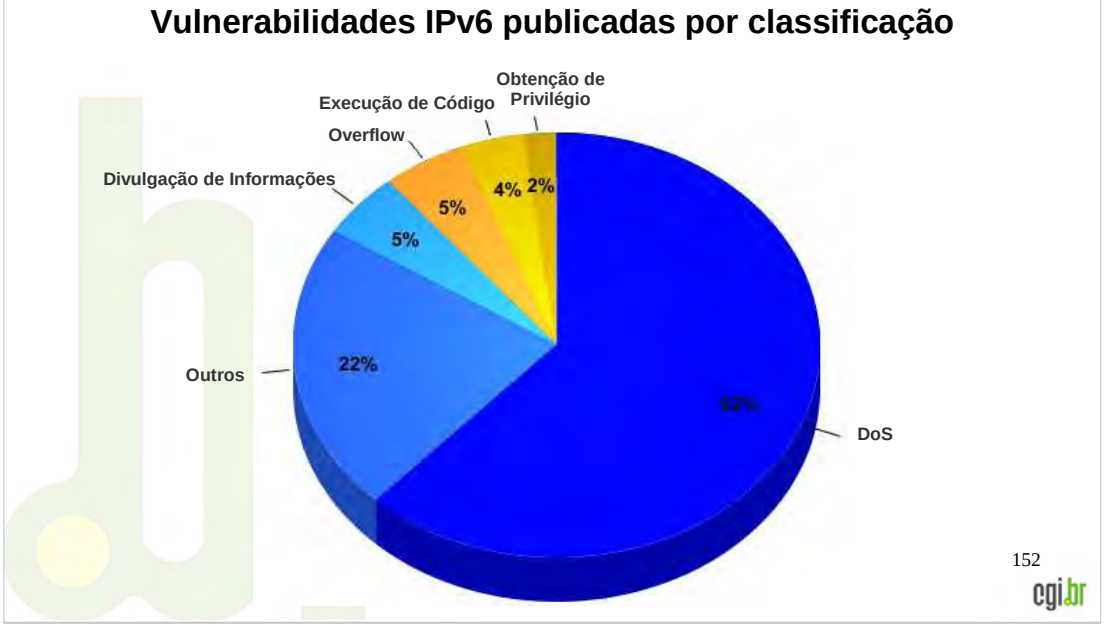
151

cgi.br

O número de vulnerabilidades (encontradas) deve crescer, com o aumento do uso do IPv6.

nic.br Núcleo de Informação e Coordenação do Ponto BR cetro.br IPv6.br

Impactos das Vulnerabilidades



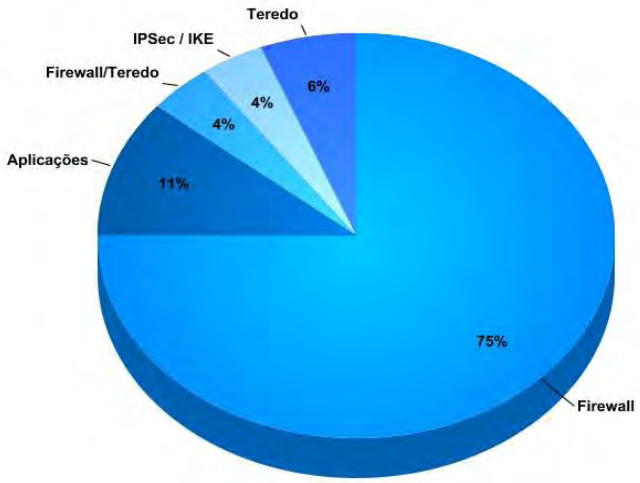
cgi.br

A maior parte das vulnerabilidades torna os sistemas sujeitos a ataques DoS.

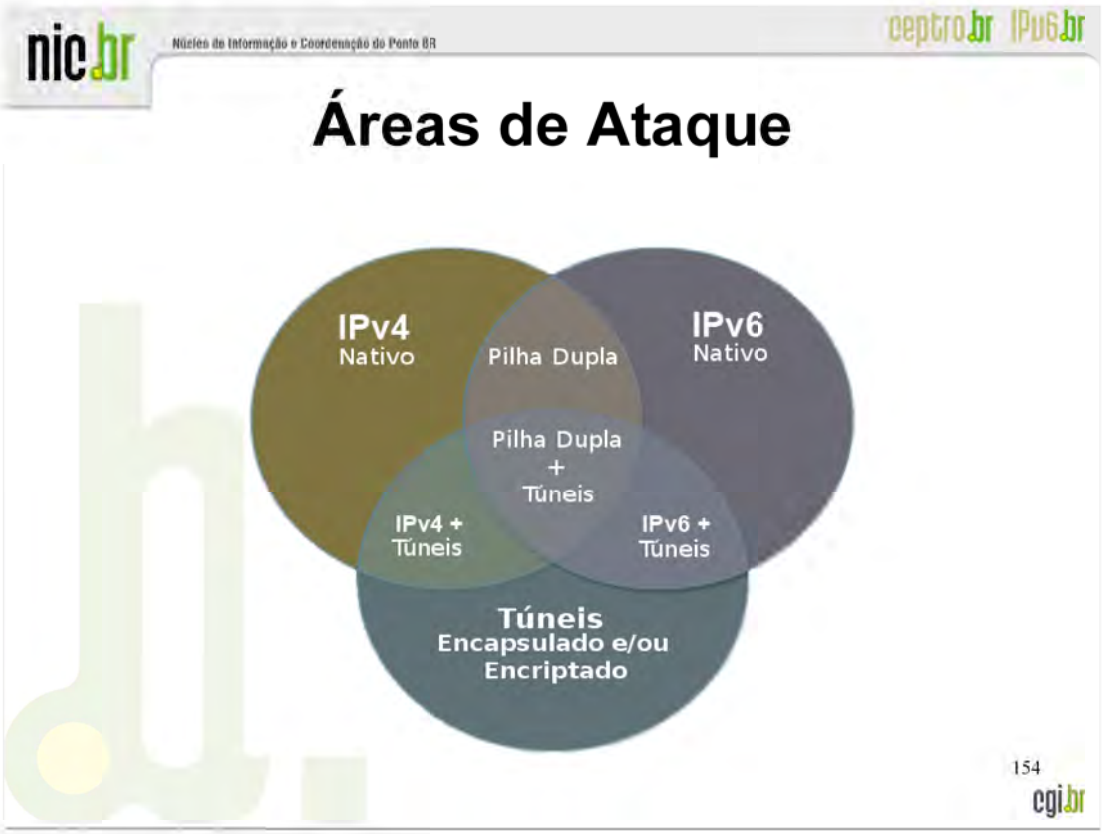
nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Núcleo dos Problemas

Vulnerabilidades IPv6 publicadas por tecnologia



A maior parte das vulnerabilidades publicadas afeta equipamentos de rede



Durante a transição de IPv4 para IPv6 há o uso de ambas as tecnologias nativamente, e de túneis. Observando a figura vê-se 7 superfícies de ataque possíveis, nesse contexto.

Apesar do IP tratar da camada de rede, implicações nas outras camadas podem levar também a vulnerabilidades ou problemas.

Segurança no IPv6

The brave new world of IPv6

O IPv6 oferece várias ferramentas tanto para defesa, quanto para o ataque:
Defesa:

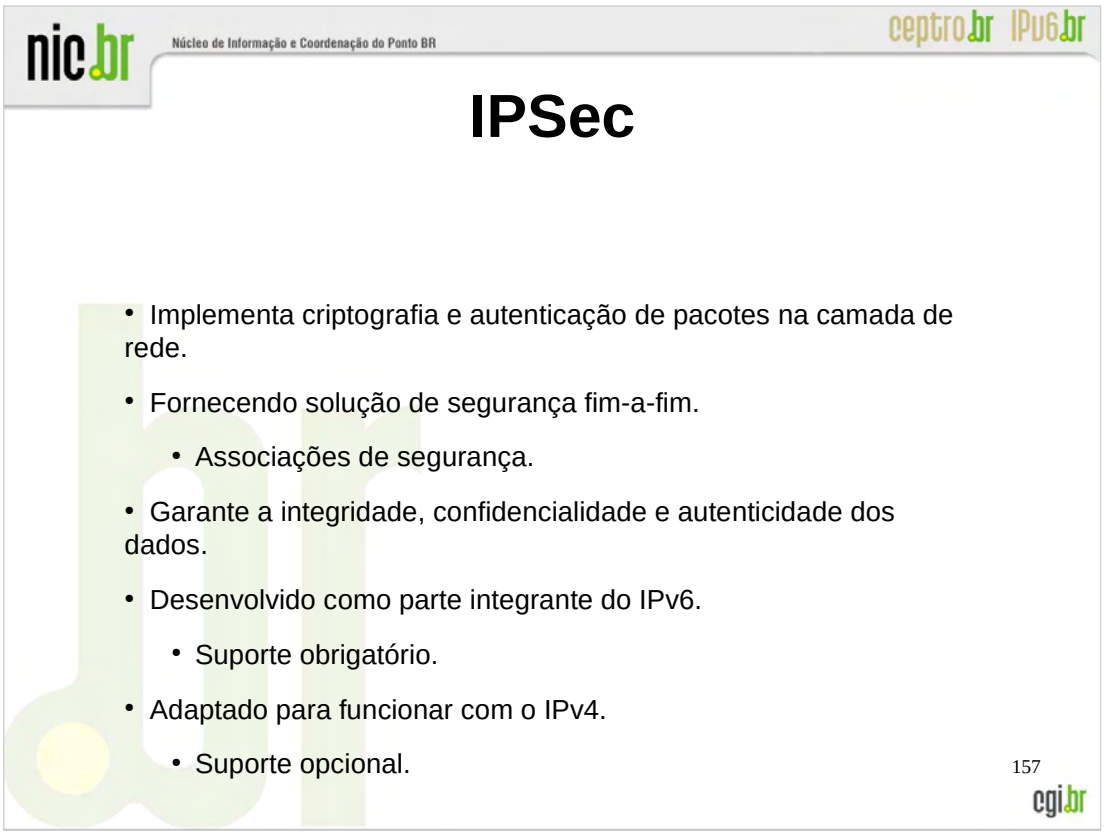
- IPsec
- SEND
- *Crypto-generated Address*
- *Unique Local Addresses*
- *Privacy Addresses*

Ataque:

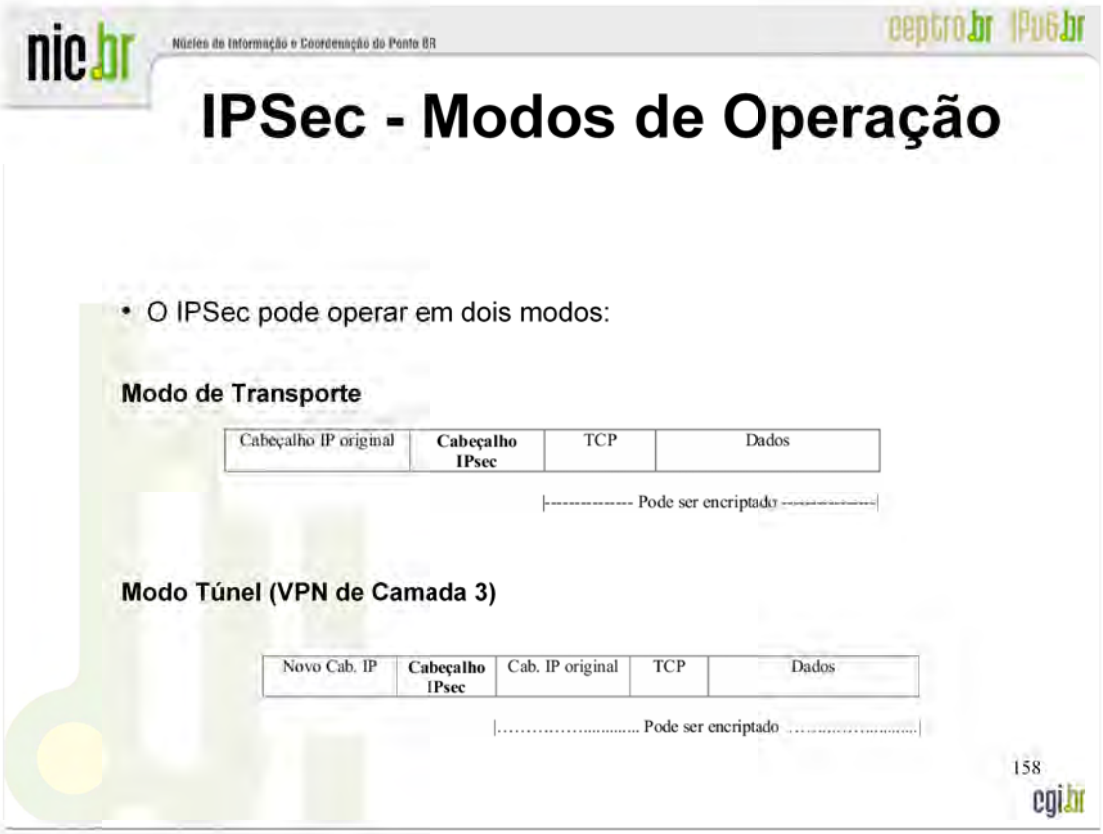
- Túnel automático
- *Neighbor Discovery* e autoconfiguração
- Modelo fim a fim
- Novidade / Complexidade
- Falta de políticas, treinamentos e ferramentas.

Deve-se:

- Ter preocupação com segurança e envolver a equipe de segurança desde o início
- Obter equipamentos certificados
- Educação / Treinamento
- Fazer upgrade das ferramentas e processos de segurança
- Desenvolver práticas de programação adequadas (e seguras) para IPv6
- Procurar auditorias / equipes de teste que conheçam IPv6



O IPSec foi desenvolvido para o IPv4 e pouca coisa muda com o IPv6. Contudo o suporte passa a ser mandatório, e não há a NAT para atrapalhar o funcionamento.



O IPSec pode ser utilizado de duas formas distintas, em modo de transporte, ou em modo de túnel. No modo de transporte, ambos os extremos da comunicação necessitam de suporte IPSec, entre os quais se realiza a comunicação segura.

Ao contrário do modo de transporte, no modo de túnel (também conhecido por VPN) o IPSec é implementado em dispositivos próprios (ex: concentradores VPN), entre os quais será efetuada a comunicação IPSec encapsulando todos os pacotes IP dos respectivos extremos.

É de salientar, que no caso de uma comunicação em modo de transporte, o cabeçalho do pacote IP original mantém-se. No modo de túnel, este é codificado e é criado um novo cabeçalho tornando possível a ligação entre o dispositivo emissor, com o dispositivo receptor (do túnel).

- **Transporte** - protege apenas os protocolos das camadas superiores, pois o cabeçalho de segurança aparece imediatamente após o cabeçalho IP e antes dos cabeçalhos dos protocolos das camadas superiores;
- **Túnel** - protege todo o pacote IP, encapsulando-o dentro de outro pacote IP, deixando visível apenas o cabeçalho IP externo.

IPSec

- *Framework* de segurança - utiliza recursos independentes para realizar suas funções.
  - *Authentication Header* (AH)
    - Integridade de todo o pacote;
    - Autenticação da origem;
    - Proteção contra o reenvio do pacote.
  - *Encapsulating Security Payload* (ESP)
    - Confidencialidade;
    - Integridade do interior do pacote;
    - Autenticação da origem;
    - Proteção contra o reenvio do pacote.
  - *Internet Key Exchange* (IKE)
    - Gerar e gerenciar chaves de segurança.

IPSec - AH

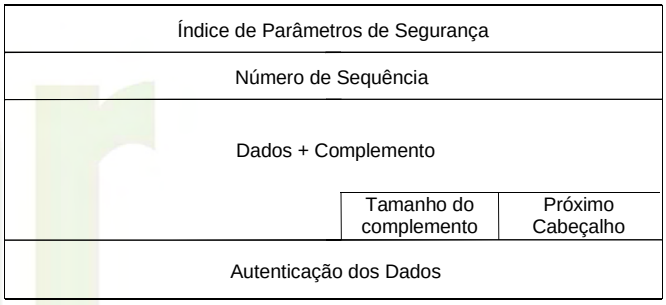
- *Authentication Header* (AH)

Próximo Cabeçalho	Tam. cab. de extensão	Reservado
Índice de Parâmetros de Segurança		
Número de Sequência		
Autenticação dos Dados		

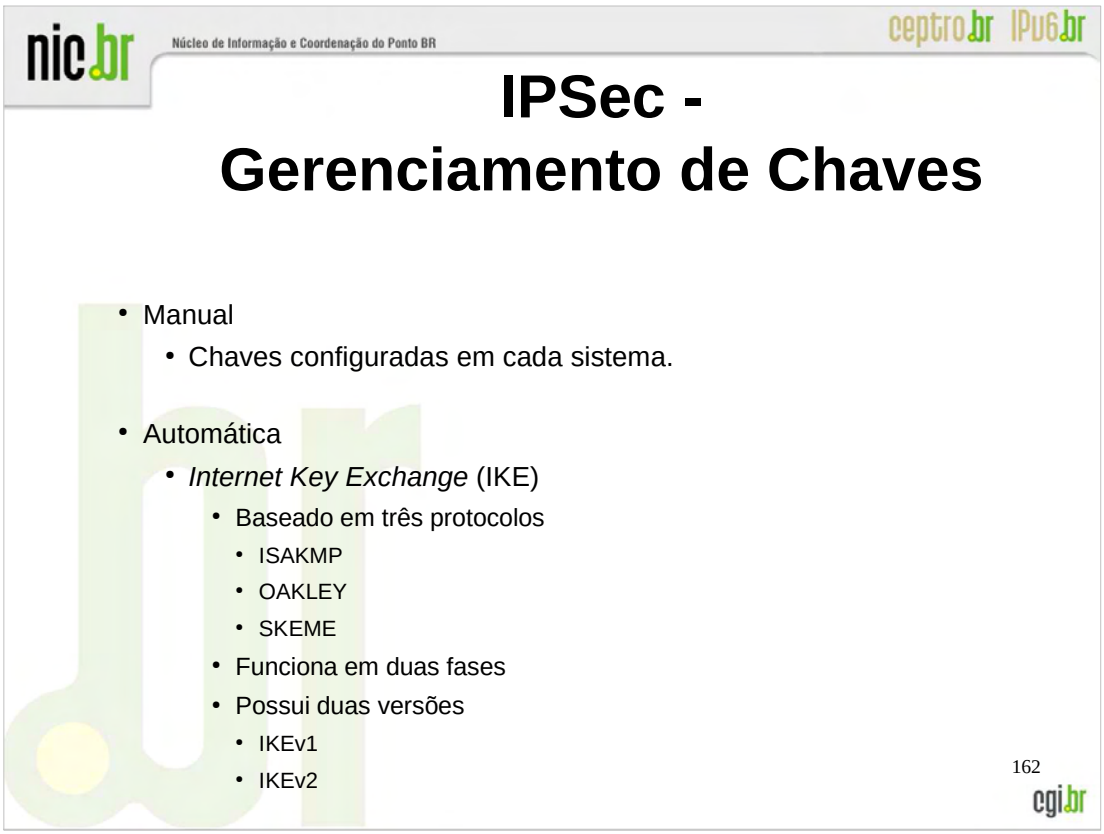
- É adicionado após os cabeçalhos *Hop-by-Hop*, *Routing* e *Fragmentation* (se houver);
- Pode ser utilizado em ambos os modos de operação.

IPSec - ESP

- *Encapsulating Security Payload* (ESP)



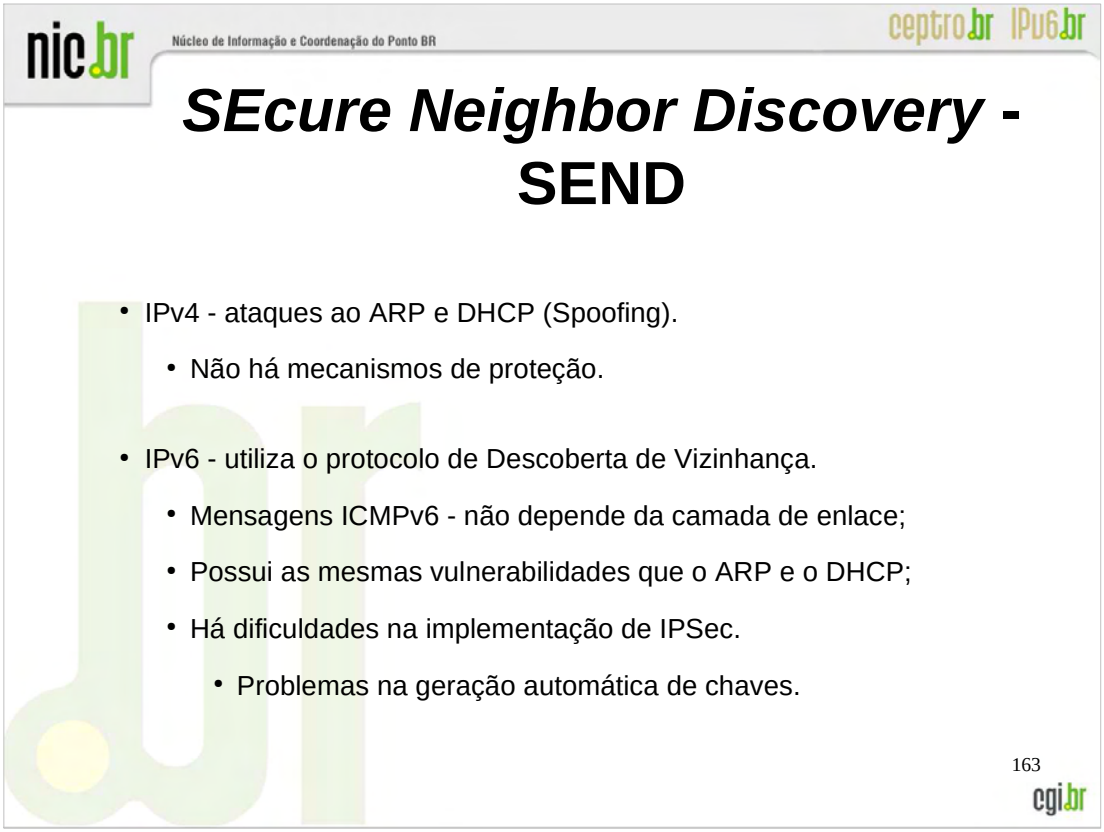
- Responsável pela criptografia dos dados (opcional);
- Pode ser utilizado em ambos os modos de operação;
- Pode ser combinado com o AH.



O Gerenciamento de Chaves é uma das dificuldades operacionais para implantar-se o IPSec no IPv4, e continua tendo o mesmo nível de complexidade no IPv6.

Mais informações:

- RFC 4301 - *Security Architecture for the Internet Protocol*



Mais informações:

- RFC 3756 - *IPv6 Neighbor Discovery (ND) Trust Models and Threats*
- RFC 3971 - *SEcure Neighbor Discovery* (SEND)

SEND

- Cadeia de certificados.
  - Utilizados para certificar a autoridade dos roteadores.
- Utilizar endereços CGA.
  - Gerados criptograficamente.
- Nova opção do protocolo de Descoberta de Vizinhança.
  - *RSA signature* - protege as mensagens relativas ao *Neighbor Discovery* e ao *Router Discovery.*
- Duas novas opções do protocolo de Descoberta de Vizinhança.
  - *Timestamp* e *nonce* - preveni ataques de reenvio de mensagens.

Estrutura dos Endereços

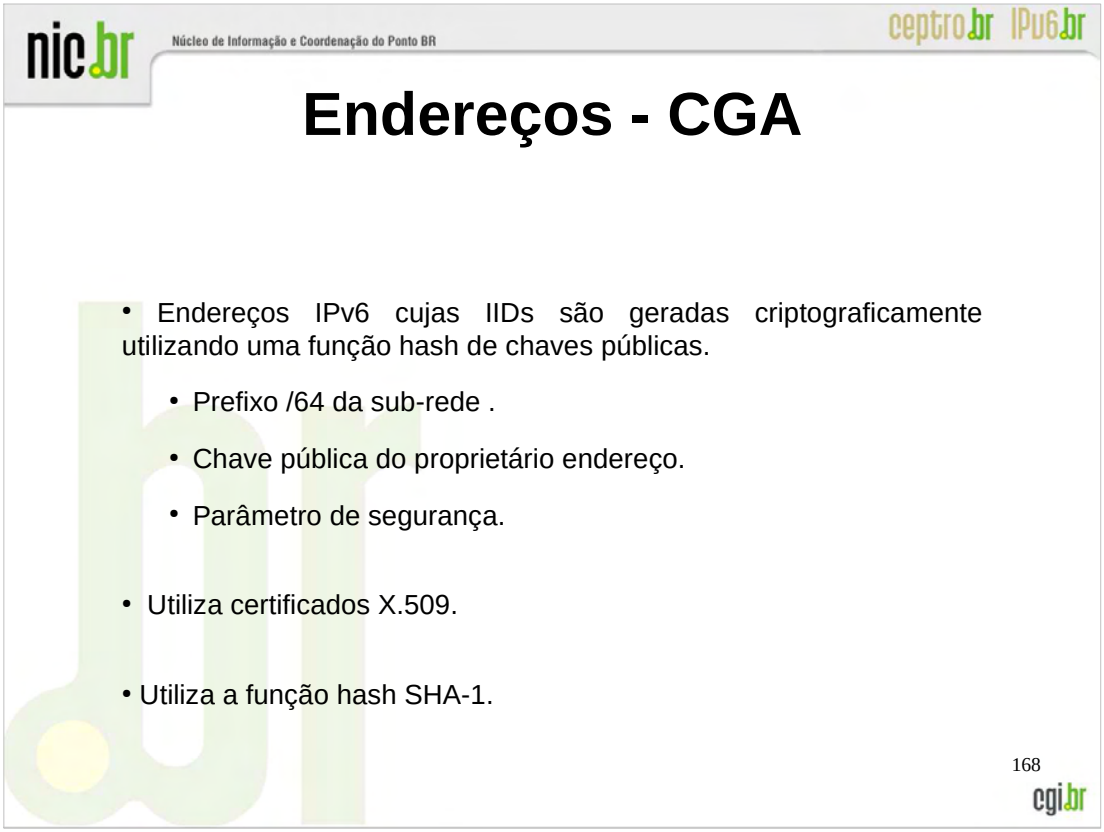
- Os 128 bits de espaço para endereçamento podem dificultar alguns tipos de ataques.
- Novas formas de gerar IID.
- A filtragem dos endereços também muda.
  - Endereços *bogons.*
- Novos tipos de ataques.

Estrutura dos Endereços

- Varredura de endereços (*Scanning*)
  - Tornou-se mais complexo, mas não impossível.
  - Com uma mascara padrão /64, são possíveis 2^{64} endereços por sub-rede.
  - Percorrendo 1 milhão de endereços por segundo, seria preciso mais de 500.000 anos para percorrer toda a sub-rede.
    - NMAP só tem suporte para escanear um único *host* de cada vez.
  - Worms que utilizam essa técnica para infectar outros dispositivos, também terão dificuldades para continuar se propagando.

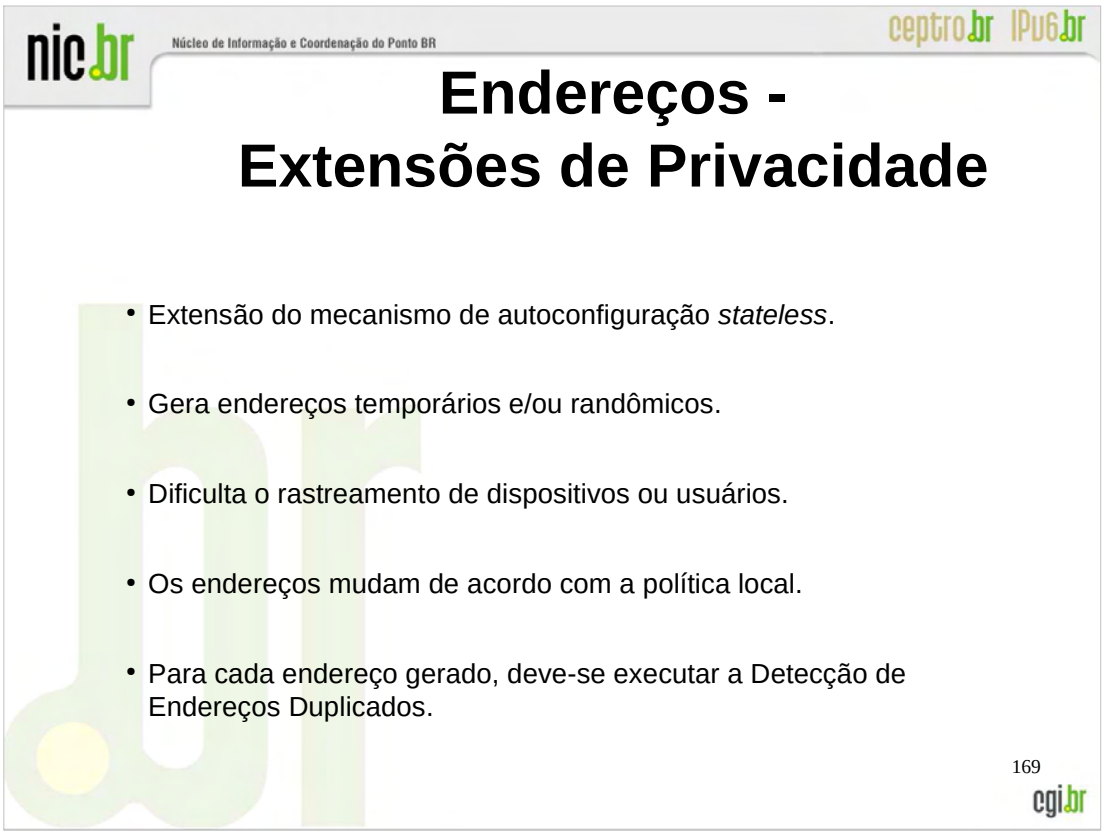
Estrutura dos Endereços

- Varredura de endereços (*Scanning*)
  - Devem surgir novas técnicas:
    - Explorar endereços de servidores públicos divulgados no DNS.
    - Procura por endereços fáceis de memorizar utilizados por administradores de redes.
      - **::10**, **::20**, **::DAD0**, **::CAFE**.
      - Último Byte do endereço IPv4.
    - Explorar endereços atribuídos automaticamente com base no MAC, fixando a parte do número correspondente ao fabricante da placa de rede.



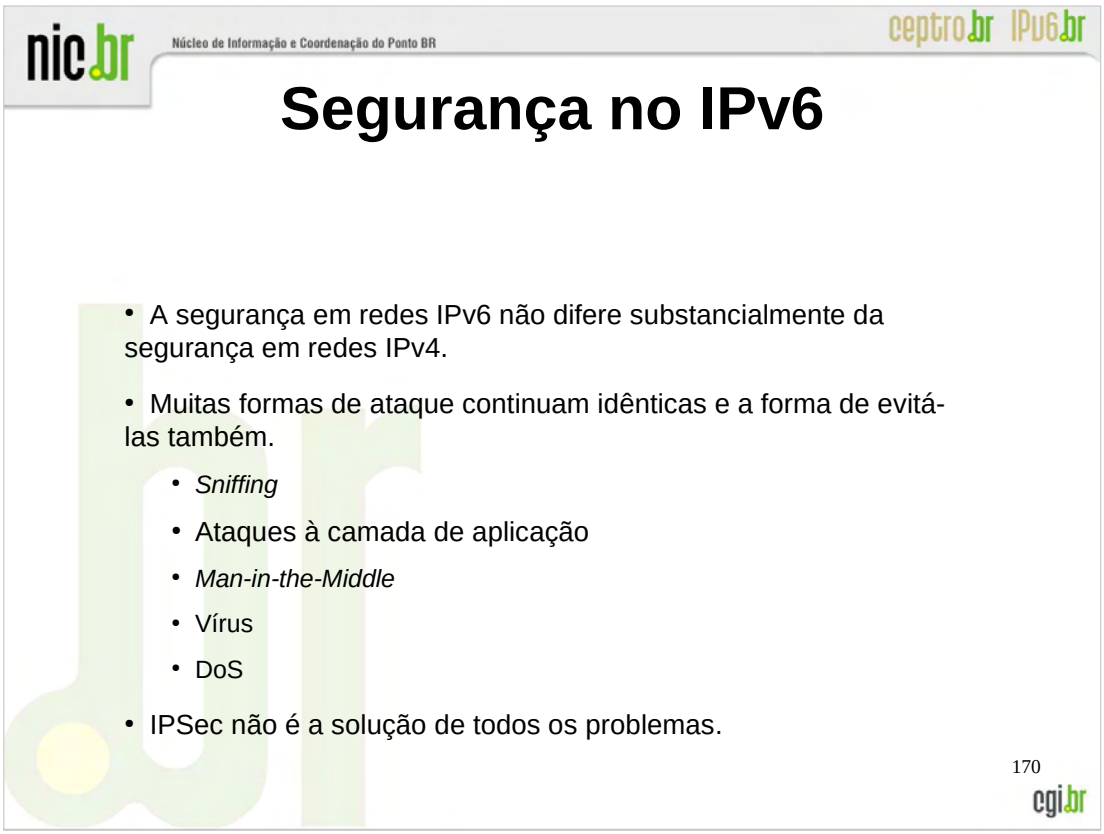
Mais informações:

- RFC 3972 - *Cryptographically Generated Addresses* (CGA)



Mais informações:

- RFC 4864 - *Local Network Protection for IPv6*
- RFC 4941 - *Privacy Extensions for Stateless Address Autoconfiguration in IPv6*



Muitas implementações da pilha IPv6 ainda não suportam IPSec integralmente. Assim, ele vem sendo usado sem o suporte criptográfico. Mesmo quando este é utilizado, várias questões de segurança em redes IP ainda estão presentes. Mas o IPv6 pode potencialmente melhorar a segurança na Internet.

• DoS: Não existem endereços *broadcast* em IPv6

– Evita ataques através do envio de pacotes ICMP para o endereço de *broadcast.*

• As especificações do IPv6 proíbem a geração de pacotes ICMPv6 em resposta a mensagens enviadas para endereços globais *multicast* (com a exceção da mensagem *packet too big*).

– Muitos Sistemas Operacionais seguem a especificação;

– Ainda há alguma incerteza sobre o perigo que pode ser criado por pacotes ICMPv6 com origem em endereços *multicast* globais.

Recomendações

- Implementar extensões de privacidade apenas em comunicações externas.
  - Cuidado com o uso indiscriminado. Pode dificultar auditorias internas.
- Endereços de uso interno devem ser filtrados nos roteadores de borda.
  - Endereços *multicast* como **FF02::1**(*all-nodes on link*), **FF05::2** (*all-routers on link*) e **FF05::5** (*all DHCPv6 servers*) podem se tornar novos vetores de ataque.
- Filtrar tráfego ingresso de pacotes com endereços de origem *multicast.*

Recomendações

- Não usar endereços óbvios;
- Filtrar serviços desnecessários no firewall.
- Filtrar mensagens ICMPv6 não essenciais.
- Filtrar endereços *bogon.*
  - Essa filtragem no IPv6 é diferente da feita no IPv4.
    - No IPv4, bloqueia-se as faixa não-alocadas (há poucas).
    - No IPv6 é o inverso. É mais fácil liberar apenas as faixas alocadas.

Mais informações:

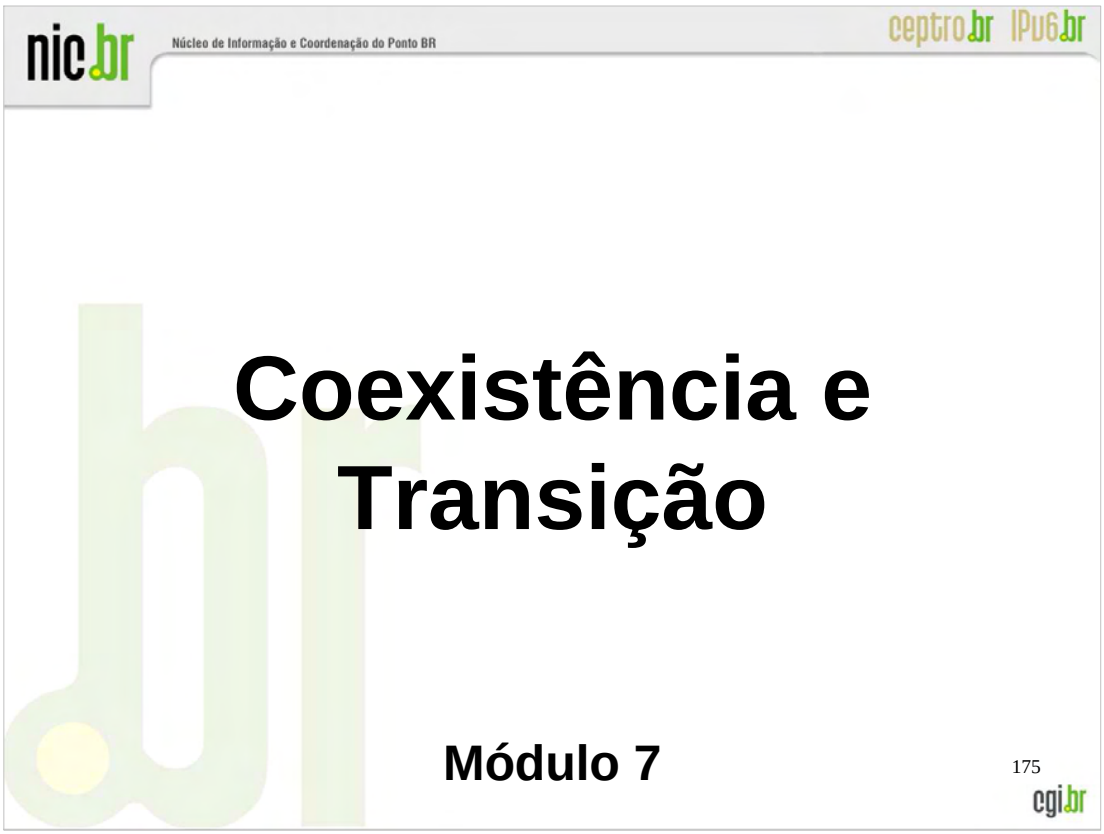
- RFC 3704 - *Ingress Filtering for Multihomed Networks*
- RFC 4890 - *Recommendations for Filtering ICMPv6 Messages in Firewalls*

Recomendações

- Bloquear fragmentos de pacotes IPv6 com destino a equipamentos de rede.
- Descartar pacotes com tamanho menor do que 1280 Bytes (exceto o último).
- Os mecanismos de segurança do BGP e do IS-IS não mudam.
- Com OSPFv3 e RIPng deve-se utilizar IPSec.
- Limitar o número de saltos para proteger dispositivos de rede.
- E utilizar IPSec sempre que necessário.

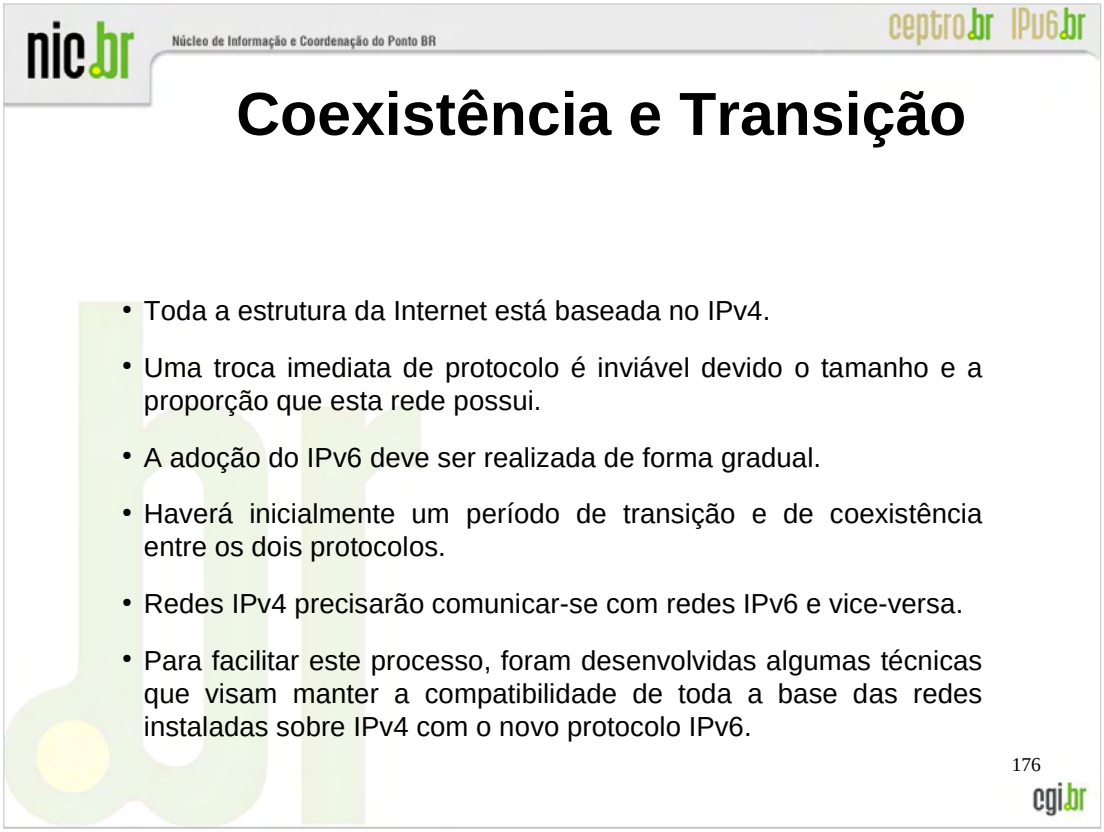
IPv6.br

A Nova Geração do Protocolo Internet



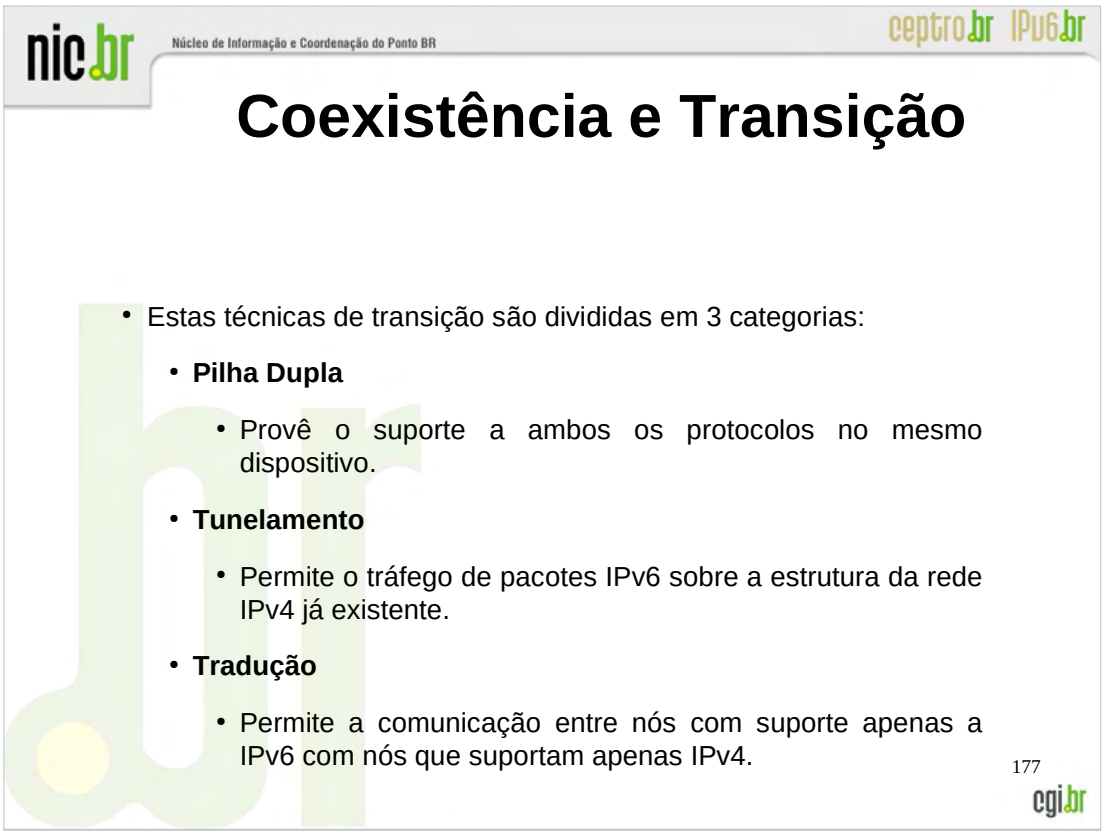
Para que a transição entre os dois Protocolo Internet ocorra de forma gradativa e sem maiores impactos no funcionamento das redes, é necessário que exista um período de coexistência entre os protocolos IPv4 e IPv6.

Neste módulo, aprenderemos as diferentes técnicas de transição utilizadas, analisando os conceitos básicos do funcionamento da Pilha Dupla, dos Túneis e das Traduções, de modo a entender em qual situação cada técnica é melhor aplicada.



Com o intuito de facilitar o processo de transição entre as duas versões do Protocolo Internet, algumas técnicas foram desenvolvidas para que toda a base das redes instaladas sobre IPv4 mantenha-se compatível com o protocolo IPv6, sendo que nesse primeiro momento de coexistência entre os dois protocolo, essa compatibilidade torna-se essencial para o sucesso da transição para o IPv6.

Cada uma dessa técnicas apresenta uma característica específica, podendo ser utilizada individualmente ou em conjunto com outras técnicas, de modo a atender as necessidades de cada situação, seja a migração para o IPv6 feita passo a passo, iniciando por um único *host* ou sub-rede, ou até de toda uma rede corporativa.



Estes mecanismos de transição podem ser classificados nas seguintes categorias:

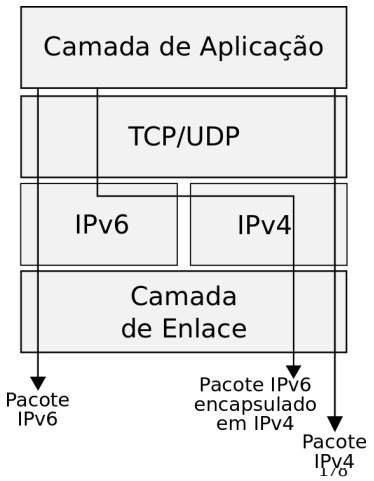
- **Pilha Dupla**: que provê o suporte a ambos os protocolos no mesmo dispositivo;
- **Tunelamento**: que permite o tráfego de pacotes IPv6 sobre estruturas de rede IPv4; e
- **Tradução**: que permite a comunicação entre nós com suporte apenas a IPv6 com nós que suportam apenas IPv4.

Como o período de coexistência entre os dois protocolos pode durar indefinidamente, a implementação de métodos que possibilitem a interoperabilidade entre o IPv4 e o IPv6, poderá garantir uma migração segura para o novo protocolo, através da realização de testes que permitam conhecer as opções que estes mecanismos oferecem, além de evitar, no futuro, o surgimento de “ilhas” isoladas de comunicação.

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Pilha Dupla

- Os nós tornam-se capazes de enviar e receber pacotes tanto para o IPv4, quanto para o IPv6.
- Um nó IPv6/IPv4, ao se comunicar com um nó IPv6, se comporta-se como um nó IPv6 e na comunicação com um nó IPv4, como nó IPv4.
- O precisa de pelo menos um endereço para cada pilha.
- Utiliza mecanismos IPv4, como por exemplo DHCP, para adquirir endereços IPv4, e mecanismos do IPv6 para endereços IPv6.



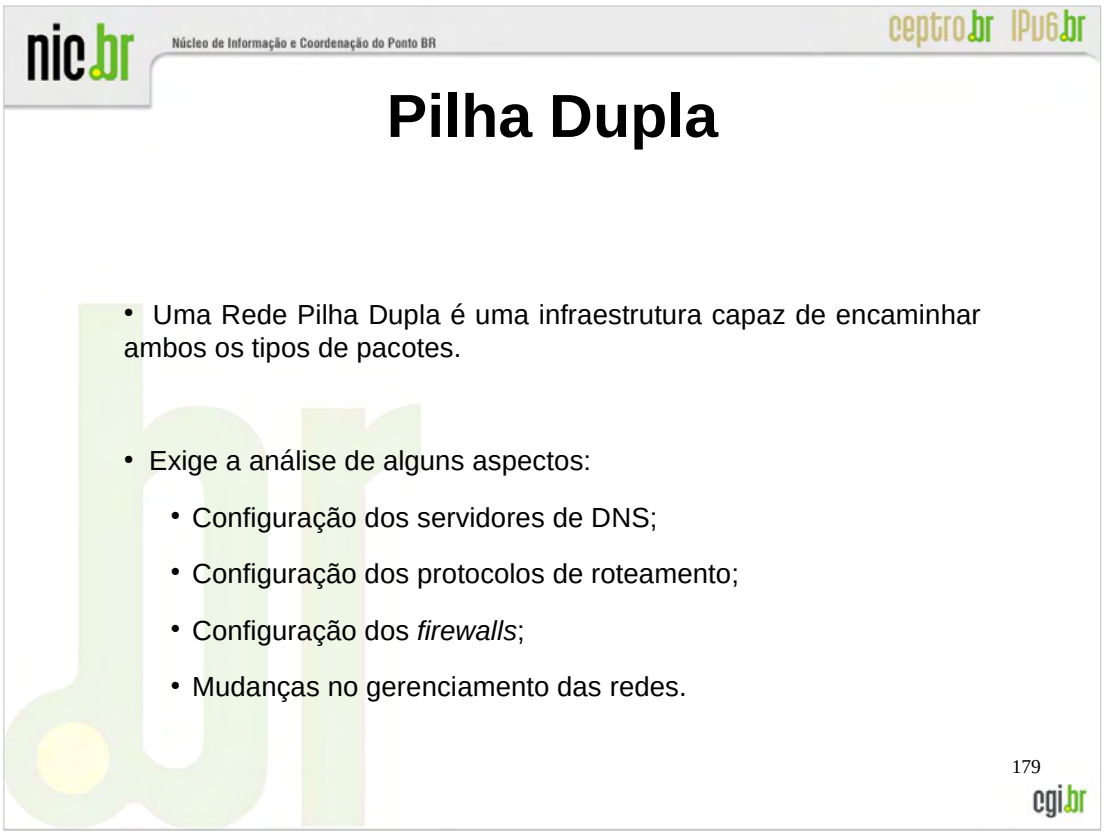
cgi.br

Nesta fase inicial de implementação do IPv6, ainda não é aconselhável ter nós com suporte apenas a esta versão do protocolo IP, visto que muitos serviços e dispositivos de rede ainda trabalham somente sobre IPv4. Deste modo, uma possibilidade é a de se introduzir o método conhecido como pilha dupla.

A utilização deste método permite que *hosts* e roteadores estejam equipados com pilhas para ambos os protocolos, tendo a capacidade de enviar e receber os dois pacotes, IPv4 e IPv6. Com isso, um nó Pilha Dupla, ou nó IPv6/IPv4, na comunicação com um nó IPv6, se comportará como um nó apenas IPv6, e na comunicação com um nó IPv4, se comportará como um nó apenas IPv4.

Cada nó IPv6/IPv4 é configurado com ambos endereços, utilizando mecanismos IPv4 (ex. DHCP) para adquirir seu endereço IPv4, e mecanismos do protocolo IPv6 (ex. autoconfiguração e/ou DHCPv6) para adquirir seu endereço IPv6.

Este método de transição pode facilitar o gerenciamento da implantação do IPv6, por permitir que este seja feito de forma gradual, configurando pequenas seções do ambiente de rede de cada vez. Além disso, caso no futuro o IPv4 não seja mais usado, basta simplesmente desabilitar a pilha IPv4 de cada nó.

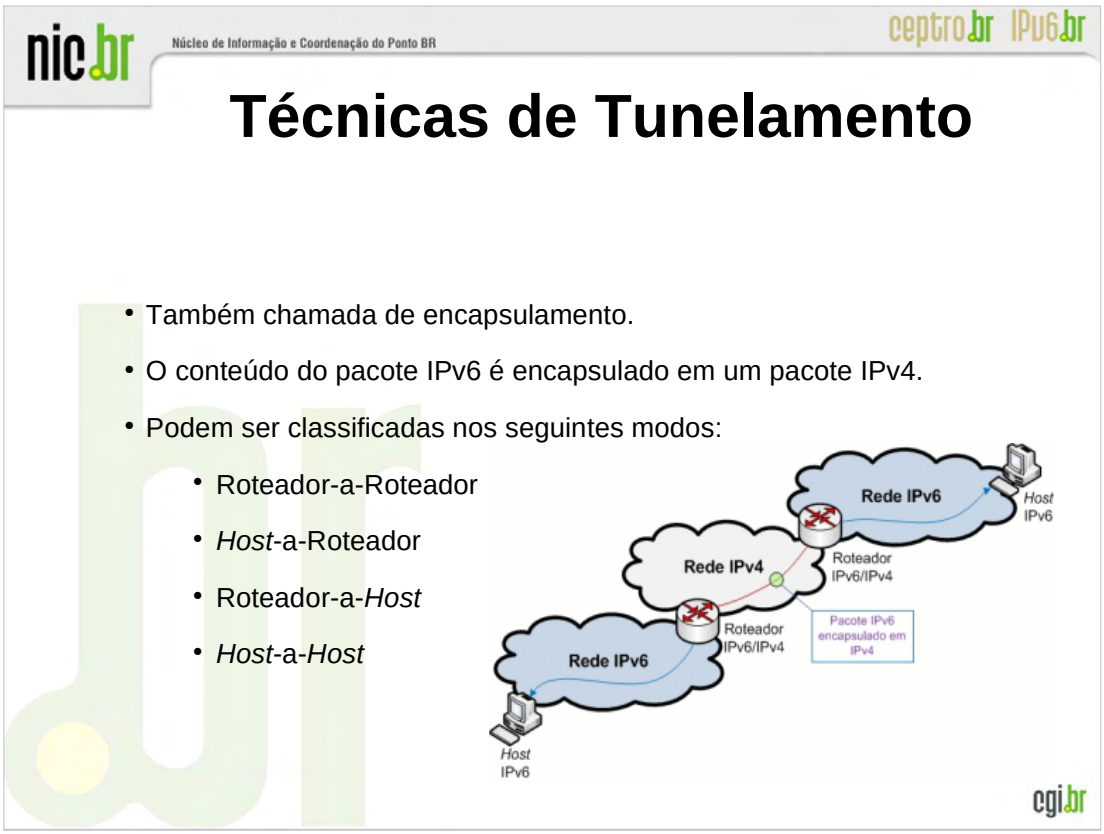


Alguns aspectos devem ser considerados ao se implementar a técnica de pilha dupla. A necessidade de mudanças na infra-estrutura da rede deve ser analisada, como a estruturação do serviço de DNS e a configuração dos protocolos de roteamento e de *firewalls*.

Em relação ao DNS, é preciso que este esteja habilitado para resolver nomes e endereços de ambos os protocolos. No caso do IPv6, é preciso responder a consultas de registros do tipo AAAA (quad-A), que armazenam endereços no formato do IPv6, e para o domínio criado para a resolução de reverso, o ip6.arpa. Para mais detalhes sobre o suporte do DNS ao IPv6, consulte a RFC 3596.

Em uma rede IPv6/IPv4, a configuração do roteamento IPv6 normalmente é independente da configuração do roteamento IPv4. Isto implica no fato de que, se a rede antes de ser implementada a pilha dupla utilizava apenas o protocolo de roteamento interno OSPFv2, com suporte apenas ao IPv4, será necessário migrar para um protocolo de roteamento que suporte tanto IPv6 quanto IPv4, como IS-IS por exemplo, ou forçar a execução de um IS-IS ou OSPFv3 paralelamente com o OSPFv2.

A forma como é feita a filtragem dos pacote que trafegam na rede, pode depender da plataforma que se estiver utilizando. Em um ambiente Linux, por exemplo, os filtros de pacotes são totalmente independentes um dos outros, de modo que o iptables filtra apenas pacotes IPv4 e o ip6tables apenas IPv6, não compartilhando nenhuma configuração. No FreeBSD, as regras são aplicadas a ambos os protocolos, a menos que se restrinja explicitamente a qual família de protocolo as regras devem ser aplicadas, usando inet ou inet6.

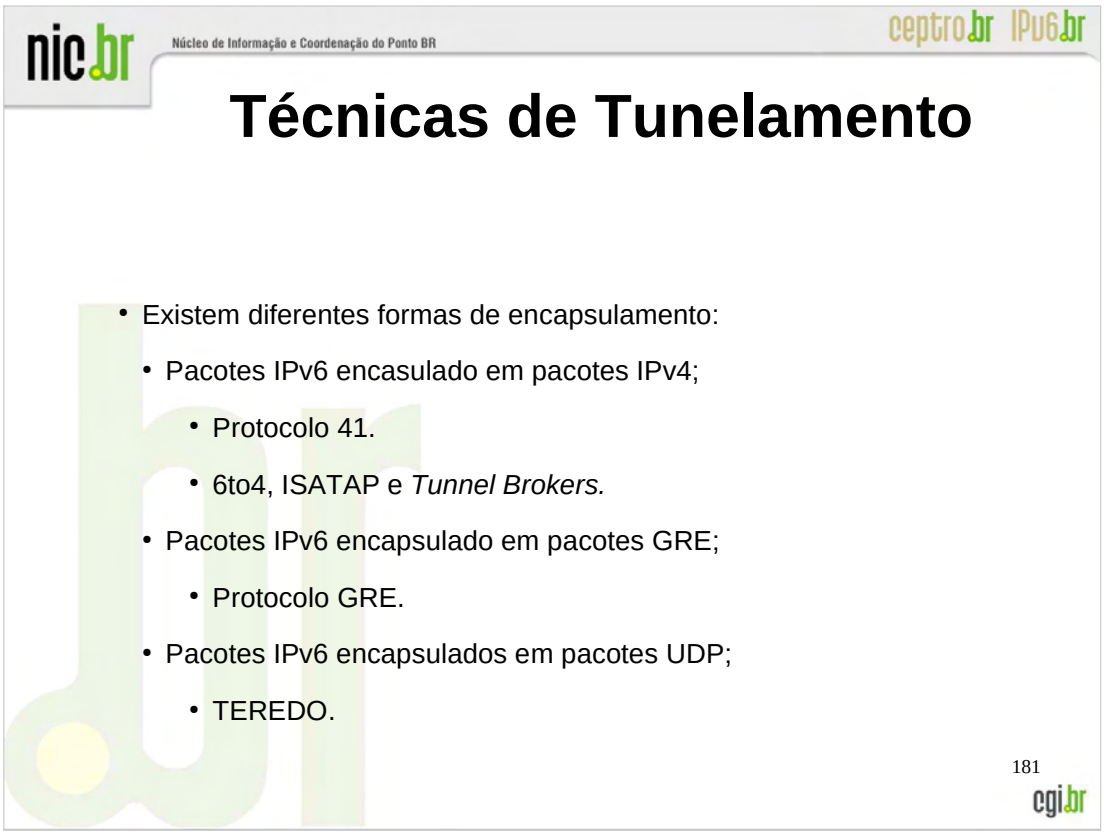


A técnica de criação de túneis, ou tunelamento, permite transmitir pacotes IPv6 através da infra-estrutura IPv4 já existente, sem a necessidade de realizar qualquer mudança nos mecanismos de roteamento, encapsulando o conteúdo do pacote IPv6 em um pacote IPv4.

Essas técnicas, tratadas na RFC 4213, têm sido as mais utilizadas na fase inicial de implantação do IPv6, por serem facilmente aplicadas em teste, onde há redes não estruturadas para oferecer tráfego IPv6 nativo.

Mais informações:

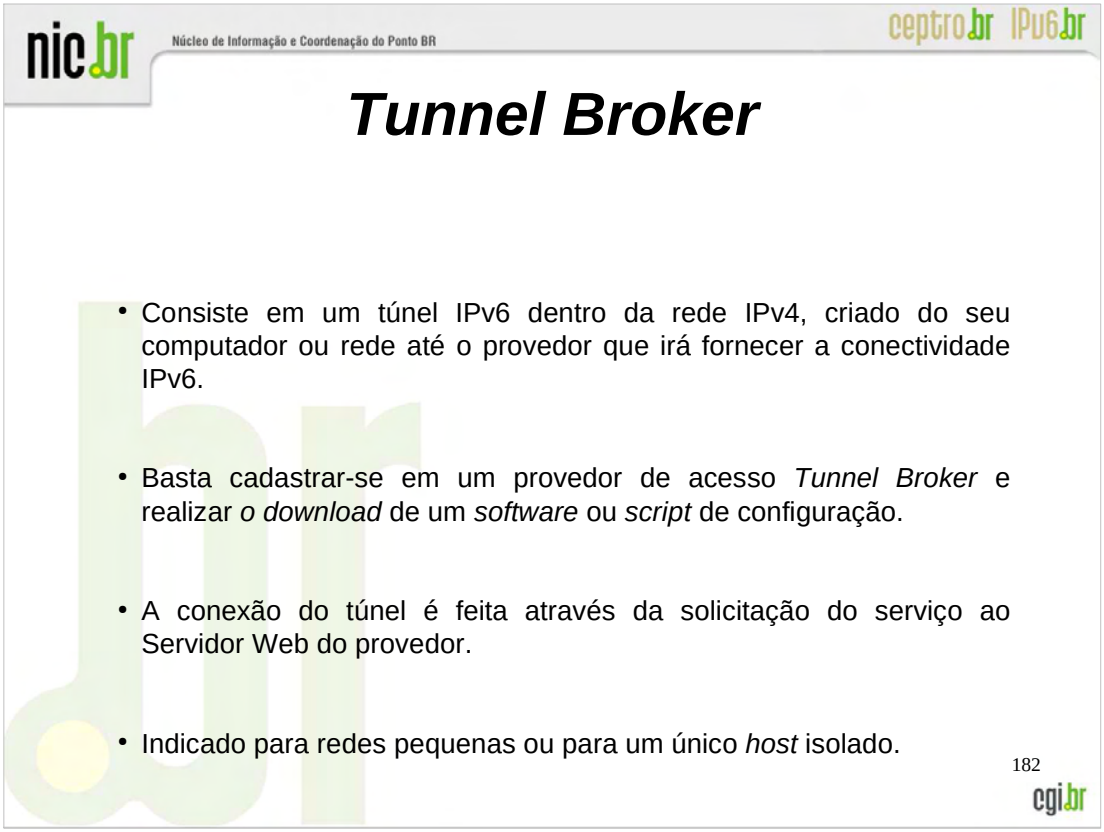
- RFC 4213 - *Basic Transition Mechanisms for IPv6 Hosts and Routers*



Existem diversas técnicas de tunelamento disponíveis. Os cenários onde podem ser aplicados, as dificuldades de implementação e a diferença de performance variam significativamente entre os modelo, necessitando uma análise detalhada de cada um. As principais técnicas de tunelamento são:

- Tunnel Broker
- 6to4
- ISATAP
- Teredo
- GRE

Vamos agora, analisar detalhadamente cada uma dessas técnicas.



Descrita na RFC 3053, essa técnica permite que *hosts* IPv6/IPv4 isolados em uma rede IPv4 acessem redes IPv6. Seu funcionamento é bastante simples, primeiramente é necessário cadastrar-se em um provedor de acesso Tunnel Broker e realizar o *download* de um *software* ou *script* de configuração. A conexão do túnel é estabelecida através da solicitação do serviço ao Servidor Web do provedor, que após autenticação, verifica qual tipo de conexão o cliente está utilizado (IPv4 público ou NAT) e lhe atribui um endereço IPv6. A partir desse ponto, o cliente pode acessar qualquer *host* na Internet.

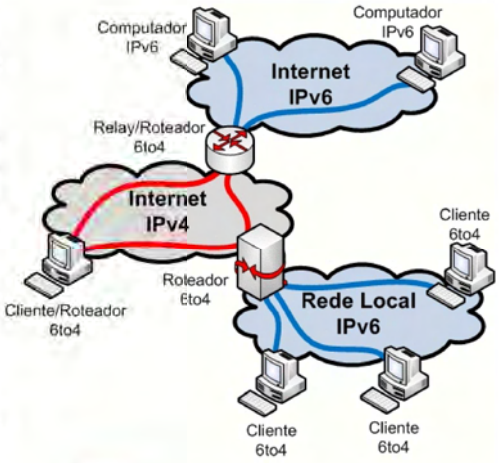
Mais informações:

- RFC 3053 - *IPv6 Tunnel Broker*

nic.br Núcleo de Informação e Coordenação do Ponto BR — ceptro.br IPv6.br

6to4

- Forma de tunelamento roteador-a-roteador.
- Fornecendo um endereço IPv6 único ao *host*.
- O endereço é formado pelo prefixo de endereço global **2002:wwxx:yyzz::/48**, onde **wwxx:yyzz** é o endereço IPv4 público do *host* convertido para hexadecimal.
- O *relay* 6to4 pode ser identificado pelo endereço *anycast* **192.88.99.1**.
- Encaminhamento Assimétrico.
- Pode ser utilizada com Relays públicos, quando não há conectividade v6 nativa.
- Quando há conectividade nativa e serviços, deve ser implementada para facilitar a comunicação com clientes 6to4.



cgi.br

Definida na RFC 3056, a técnica de tunelamento automática 6to4 permite a interconexão ponto-a-ponto entre roteadores, subredes ou computadores IPv6 através da rede IPv4, fornecendo um endereço IPv6 único formado a partir de endereços IPv4 públicos. Este endereçamento 6to4 utiliza o prefixo de endereço global **2002:wwxx:yyzz::/48**, onde **wwxx:yyzz** é o endereço IPv4 público do cliente convertido para hexadecimal.

* Cliente/Roteador 6to4: cliente que possui um endereço IPv4 público e conectividade direta 6to4, ou seja, ele tem uma interface virtual 6to4 pela qual acessa diretamente a Internet IPv6 sem necessidade de utilização de roteador 6to4. Ele necessita apenas de um *Relay* 6to4;

* Roteador 6to4: roteador que suporta 6to4, possibilitando aos clientes que não suportam este tipo de endereço, acessarem outros *hosts* 6to4 IPv6 através dele. No caso dos acessos à Internet IPv6, ele direcionará o tráfego até o *Relay Router* mais próximo, que encaminhará o pacote para a rede IPv6;

* *Relay* 6to4: roteador com suporte ao 6to4 e que também possui conexão nativa à Internet IPv6. Com isso, ele consegue rotear e se comunicar com a rede IPv6 nativa, com a rede IPv4 e com a rede 6to4;

* Cliente 6to4: equipamento de rede ou computador que possui apenas endereço IPv6 no formato 6to4, mas que não tem uma interface virtual 6to4. Com isso, há a necessidade de um Roteador 6to4 que faça a comunicação com outras redes IPv6 e 6to4.

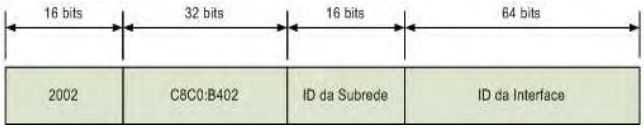
Mais informações:

- RFC 3056 - *Connection of IPv6 Domains via IPv4 Clouds*

nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

6to4



• O prefixo 6to4 é sempre **2002**.

• O próximo campo, IPv4 público do cliente, é criado convertendo-se o endereço para hexadecimal.

• O ID da subrede pode ser usado para segmentar a rede IPv6 6to4 em até 2^{16} subredes com 2^{64} endereços cada, pode se utilizar por exemplo 0, 1, 2, 3, 4...

• O ID da interface pode ser igual ao segundo campo (Windows faz assim) ou qualquer outro número no caso de configuração manual (no Linux, usa-se sequencial 1, 2, 3, 4...).

- O prefixo 6to4 é sempre **2002**, conforme definição da IANA;
- O próximo campo, IPv4 público do cliente, é criado conforme o seguinte exemplo:

Endereço IPv4: **200.192.180.002**

Primeiro convertemos cada número decimal para hexadecimal:

200=C8

192=C0

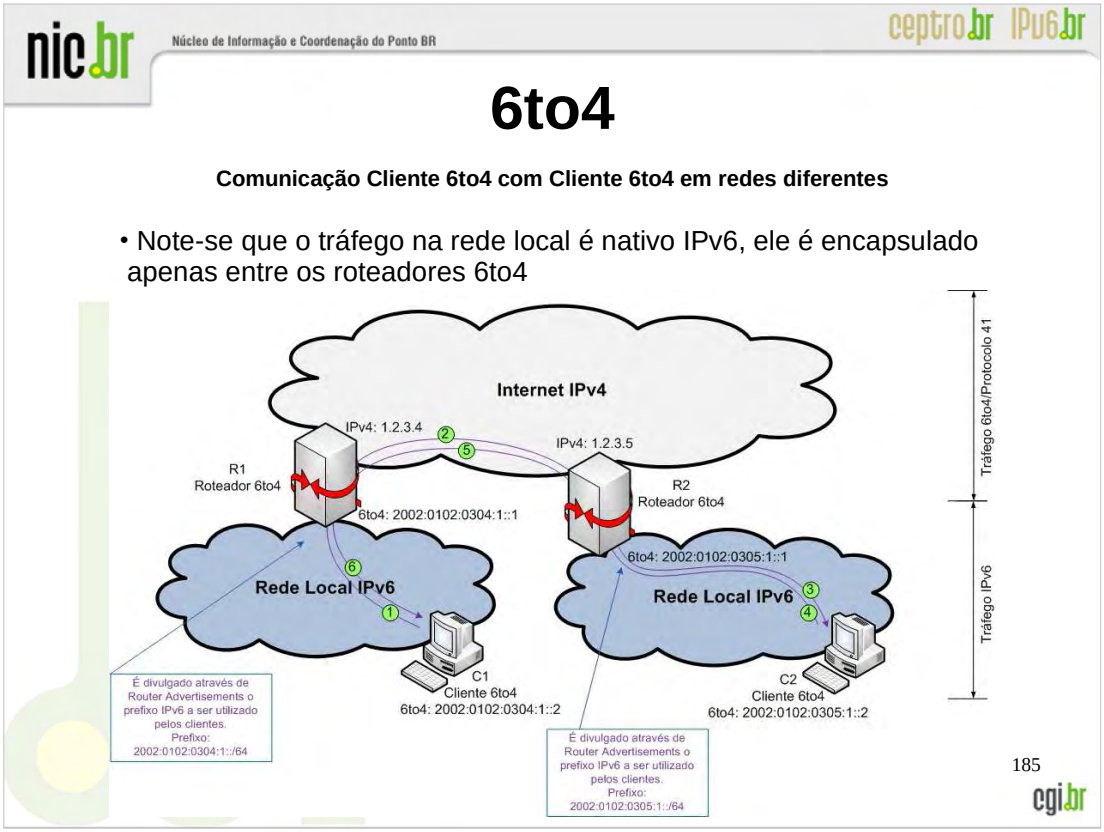
180=B4

002=02

Com isso, o endereço convertido é C8C0:B402

- O ID da subrede é utilizado apenas para segmentar a rede 6to4 associada ao IPv4 público em várias subredes(2^{16} subredes com 2^{64} endereços cada), pode se utilizar por exemplo 0, 1, 2, 3, 4...;

- O ID da interface pode ser igual ao segundo campo(IPv4 convertido para hexadecimal) no caso da configuração automática do Windows Vista e Server 2008 ou então 1, 2, 3, 4... no caso de configuração manual ou do Linux e BSD. Como o comprimento deste campo é de 64 bits, podemos ter até 2^{64} endereços por subrede.



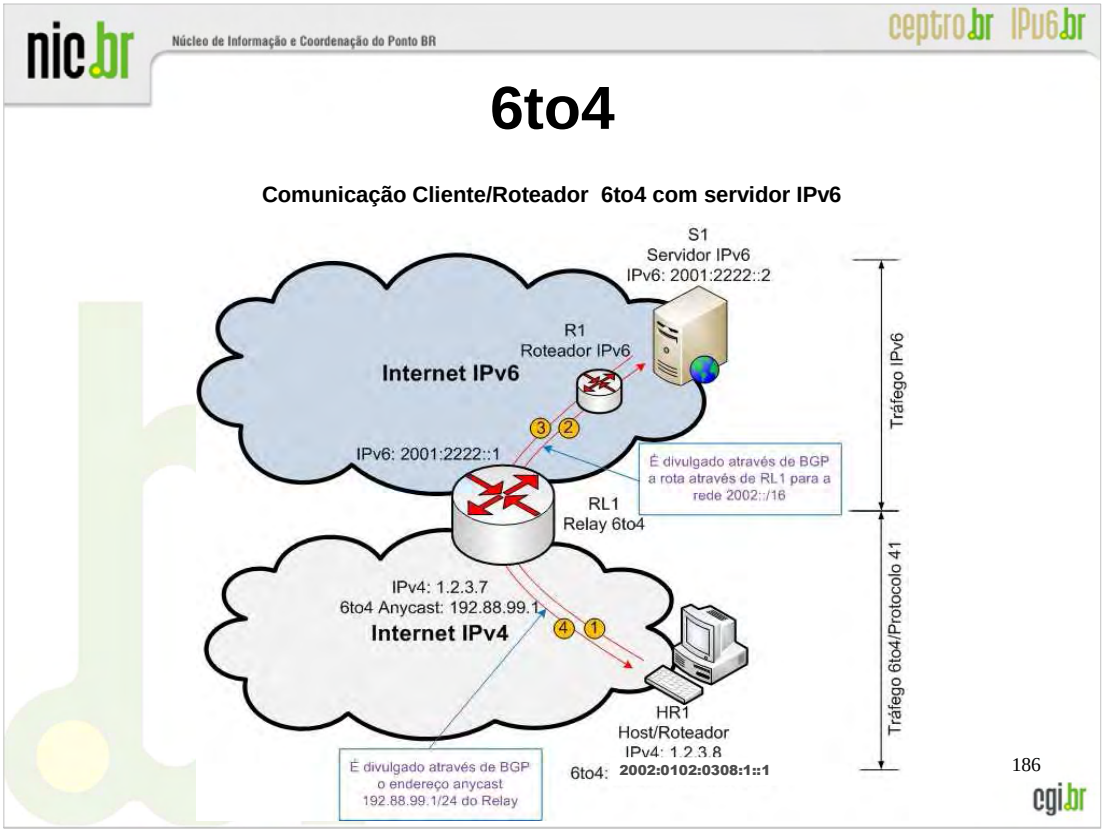
Equipamento	Rota
C1	**::/0** através de R1
	2002:0102:0304:1::/64 através da interface LAN
R1	**::/0** através do Relay 6to4 utilizando a interface virtual 6to4
	2002::/16 através da interface virtual 6to4
	2002:0102:0304:1/64 para a rede local através da interface LAN
R2	**::/0** através de R2
	2002:0102:0305:1:/64 para a rede local através da interface LAN
C2	**::/0** através do Relay 6to4 utilizando a interface virtual 6to4
	2002::/16 através da interface Virtual 6to4
	2002:0102:0305:1:/64 para a rede local através da interface LAN

1- Analisando a tabela de roteamento, nota-se que o pacote é enviado através da rede local IPv6 para o roteador R1 utilizando a rota **::/0**;

2- O pacote IPv6 é recebido por R1 através da interface LAN. R1 verifica sua tabela de roteamento e descobre que deve enviar o pacote para a sua interface virtual 6to4 (rota para rede **2002::/16**). Nesta interface o pacote IPv6 é encapsulado em um pacote IPv4 (protocolo tipo 41) que é endereçado ao roteador R2 (endereço extraído do endereço IPv6 do destinatário do pacote original);

3- O pacote IPv6 encapsulado em IPv4 é recebido por R2 através da sua interface IPv4 ou WAN. Como o pacote é do tipo 41, ele é encaminhado à interface virtual 6to4, que o desencapsula. Consultando a sua tabela de roteamento, R2 descobre que o pacote é destinado à sua rede local **2002:0102:03:05:1::/64**, sendo assim, ele encaminha através da sua rede local o pacote IPv6 ao computador C2.

Nos passos seguintes o sistema de comunicação é o mesmo, mudando apenas a direção de encaminhamento do pacote.



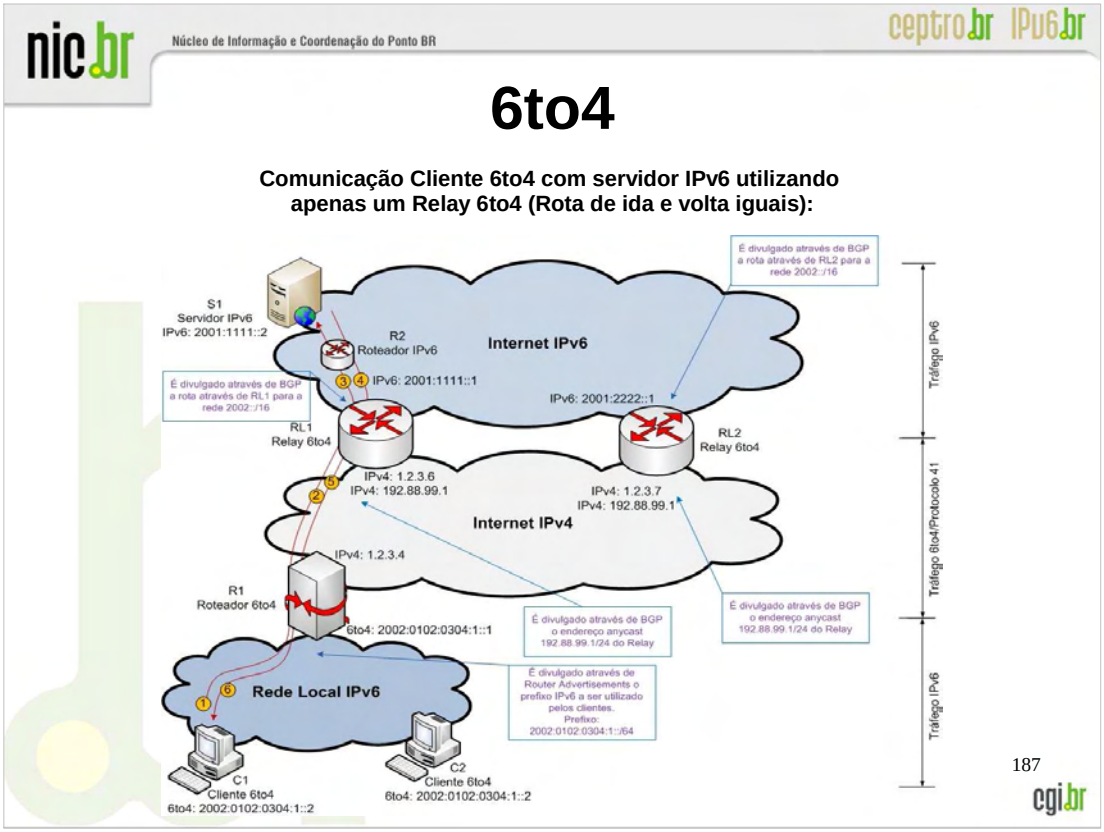
Equipamento	Rota
HR1	**::/0** através da interface virtual 6to4
	2002::/16 através da interface virtual 6to4
RL1	**::/0** rede IPv6 através da interface LAN
	2002::/16 através da interface virtual 6to4
S1	Rota padrão através de R1
R1	**2002::/16** através do Relay RL1 (rota descoberta através da divulgação via BGP)

1- HR1 envia um pacote IPv6 para S1, através da tabela de roteamento o pacote é direcionado para a interface virtual 6to4. Esta encapsula o pacote IPv6 em um pacote IPv4 (protocolo 41) e coloca como destino o endereço do Relay, que pode ser especificado manualmente ou então descoberto automaticamente através do encaminhamento do pacote para o endereço *anycast* **192.88.99.1**;

2- O Relay RL1 recebe o pacote encapsulado através do seu IPv4 ou *anycast*, como o protocolo do pacote é 41, ele desencapsula o pacote IPv6, e através da sua tabela de roteamento, descobre que o pacote deve ser enviado para S1 através da sua interface LAN na rede IPv6;

3- Depois de recebido o pacote, S1 responde utilizando a sua rota padrão através do roteador R1 da sua rede. O roteador R1 recebeu via BGP a rota para a rede **2002::/16**, e encaminha o pacote para o Relay RL1;

4- RL1 recebe o pacote e vê que ele é destinado à rede 6to4, sendo assim, ele encaminha o pacote para a interface virtual 6to4, que o encapsula em um pacote IPv4 (protocolo 41) e através do endereço IPv4 implícito no endereço IPv6 do destinatário, o pacote é encaminhado para HR1. HR1 recebe o pacote na sua interface IPv4, vê que está sendo utilizado o protocolo 41 e desencapsula o pacote IPv6 através da interface virtual 6to4.



Equipamento	Rota
RL1	**::/0** rede IPv6 através da interface LAN / **2002::/16** através da interface virtual 6to4
RL2	**::/0** rede IPv6 através da interface LAN / **2002::/16** através da interface virtual 6to4
S1	Rota padrão através de R2
R2	**2002::/16** através do Relay RL1 (rota descoberta através da divulgação via BGP)
R1	**::/0** através do Relay 6to4 RL1 ou RL2 utilizando a interface virtual 6to4
	2002::/16 através da interface virtual 6to4
	2002:0102:0304:1/64 para a rede local através da interface LAN
C1	**::/0** através de R1 / **2002:0102:0304:1::/64** através da interface LAN
C2	**::/0** através de R1 / **2002:0102:0304:1::/64** através da interface LAN

1- De acordo com a tabela de roteamento, o pacote é enviado através da rede local IPv6 para o roteador R1 utilizando a rota **::/0**;

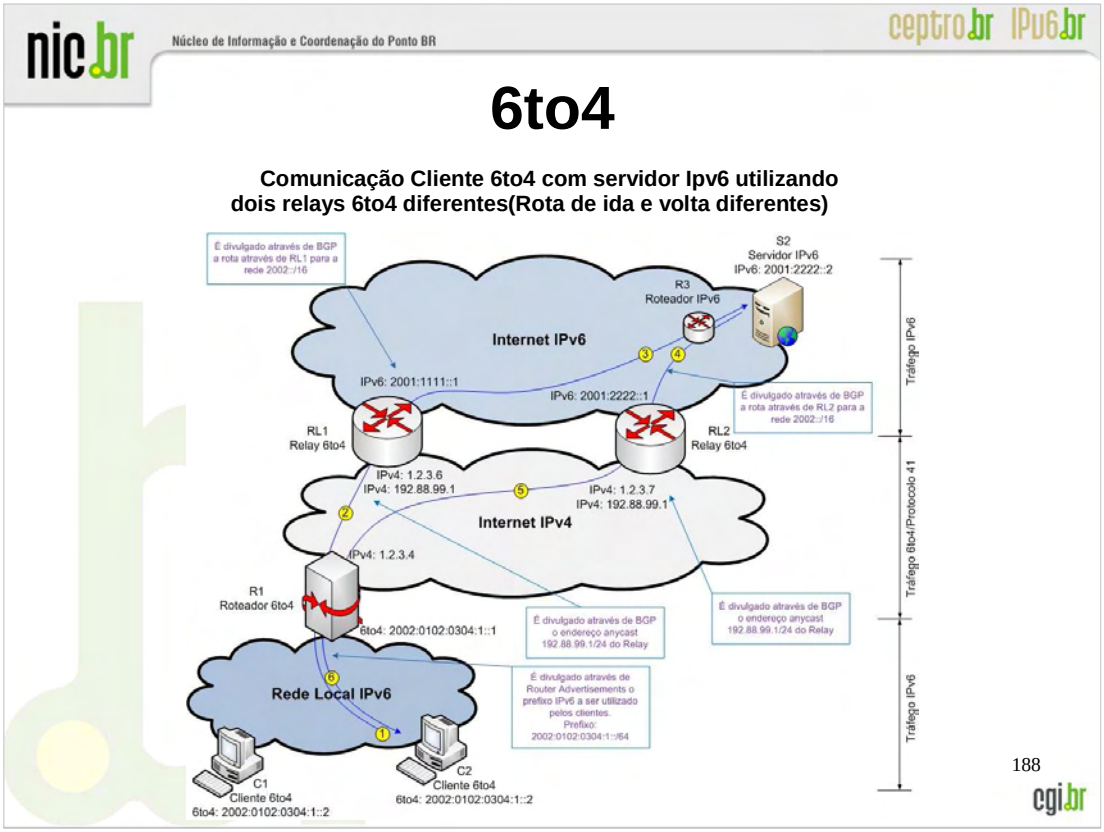
2- O pacote IPv6 é recebido por R1 através da interface LAN, que verifica a sua tabela de roteamento e descobre que o pacote deve ser encaminhado para a interface virtual 6to4 (rota para rede **2002::/16**). Nesta interface o pacote IPv6 é encapsulado em um pacote IPv4 (protocolo tipo 41) é enviado ao Relay RL1 ou RL2 (O *Relay* 6to4 pode ser definido manualmente no roteador 6to4 ou então automaticamente através da utilização do endereço *anycast* **192.88.99.1**). Vamos supor que o pacote foi enviado para o *Relay* RL1;

3- RL1 recebe o pacote 6to4 através de sua interface IPv4, e como o pacote utiliza o protocolo 41, ele o encaminha para a interface virtual, que desencapsula o pacote IPv6 e verifica na tabela de roteamento que deve enviá-lo pela interface LAN através do roteador R2, que simplesmente repassa o pacote IPv6 ao servidor S1;

4- S1 responde com o envio de outro pacote IPv6 com destino ao Cliente C1 utilizando a sua rota padrão que aponta para o roteador R2. R2 recebe o pacote e através da rota recebida via BGP, ele sabe que deve enviá-lo para o *relay* mais próximo, neste caso é RL1;

5- RL1 recebe o pacote IPv6 e verifica que o destino é a rede 6to4 (**2002::/16**). Sendo assim, de acordo com sua tabela de roteamento, o pacote é encaminhado para a interface virtual 6to4, que o empacota em um pacote IPv4 (protocolo 41) e o envia ao endereço IPv4 implícito no endereço IPv6 do destinatário do pacote;

6- O roteador R1 recebe o pacote através de seu endereço IPv4, como o pacote está utilizando o protocolo 41, este é encaminhado à interface virtual 6to4, que o desencapsula e verifica o endereço de destino. De acordo com sua tabela de roteamento ela envia o pacote IPv6 através da sua interface LAN para o Cliente 6to4 C1.



Equipamento	Rota
RL1	**::/0** rede IPv6 através da interface LAN / **2002::/16** através da interface virtual 6to4
RL2	**::/0** rede IPv6 através da interface LAN / **2002::/16** através da interface virtual 6to4
S2	Rota padrão através de R3
R3	**2002::/16** através do Relay RL2 (rota descoberta através da divulgação via BGP)
R1	**::/0** através do Relay 6to4 RL1 ou RL2 utilizando a interface virtual 6to4
	2002::/16 através da interface virtual 6to4
	2002:0102:0304:1/64 para a rede local através da interface LAN
C1	**::/0** através de R1 / **2002:0102:0304:1::/64** através da interface LAN
C2	**::/0** através de R1 / **2002:0102:0304:1::/64** através da interface LAN

1- De acordo com a tabela de roteamento, o pacote é enviado através da rede local IPv6 para o roteador R1 utilizando a rota **::/0**;

2- O pacote IPv6 é recebido por R1 através da interface LAN, que verifica sua tabela de roteamento e descobre que deve enviar o pacote para a interface virtual 6to4 (rota para rede **2002::/16**). Nesta interface o pacote IPv6 é encapsulado em um pacote IPv4 (protocolo tipo 41) e enviado ao Relay RL1 ou RL2 (O *Relay* 6to4 pode ser definido manualmente no roteador 6to4 ou então automaticamente através da utilização do endereço *anycast* **192.88.99.1**). Vamos supor que o pacote foi enviado para o *Relay* RL1;

3- RL1 recebe o pacote 6to4 através de sua interface IPv4, vê que o pacote utiliza o protocolo 41 e o encaminha para a interface virtual. Esta desencapsula o pacote IPv6 e verifica na sua tabela de roteamento que deve enviá-lo pela interface LAN através do roteador R3, que simplesmente repassa o pacote IPv6 ao servidor S2;

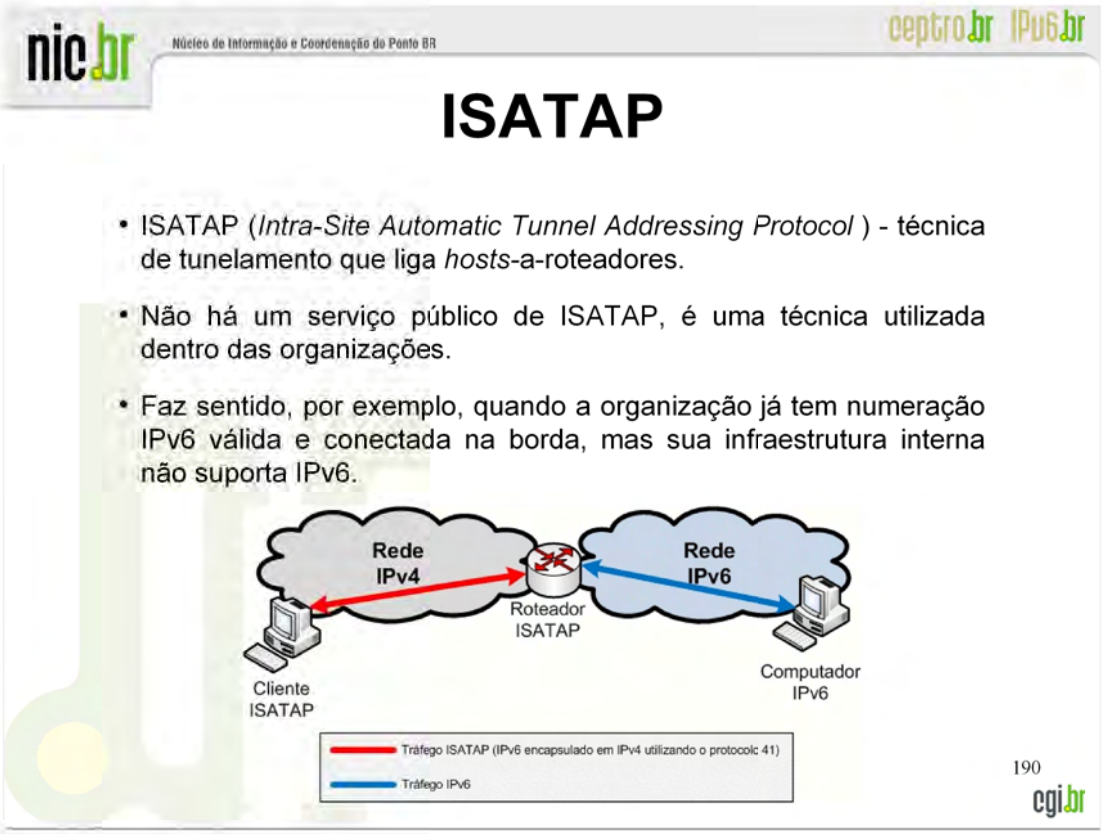
4- S2 responde com o envio de outro pacote IPv6 com destino ao Cliente C2 utilizando a sua rota padrão que aponta para o roteador R3. R3 recebe o pacote e através da rota recebida via BGP, ele sabe que deve enviá-lo para o relay mais próximo que é RL2;

5- RL2 recebe o pacote IPv6 e verifica que o destino é a rede 6to4 (**2002::/16**). Deste modo, de acordo com sua tabela de roteamento, ele encaminha o pacote para a interface virtual 6to4, que o empacota em um pacote IPv4 (protocolo 41) e o envia ao endereço IPv4 implícito no endereço IPv6 do destinatário do pacote;

6- O roteador R1 recebe o pacote através de seu endereço IPv4, verifica que o pacote está utilizando o protocolo 41 e o encaminha à interface virtual 6to4. Esta o desencapsula e verifica o endereço de destino. De acordo com sua tabela de roteamento e o endereço de destino, o pacote IPv6 é enviado através da sua interface LAN para o Cliente 6to4 C2.

6to4

- **Segurança**
  - Os Relay roteador não verifica os pacotes IPv6 que estão encapsulados em IPv4, apesar dele os encapsular e desencapsular;
  - O spoofing de endereço é um problema grave de túneis 6to4, podendo ser facilmente explorado;
  - Não há um sistema de autenticação entre o roteador e o Relay roteador, facilitando assim a exploração de segurança através da utilização de Relays roteadores falsos.



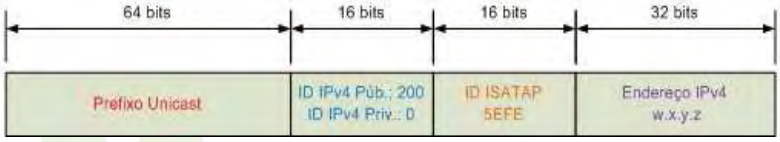
A técnica de transição *Intra-Site Automatic Tunnel Addressing Protocol* (ISATAP), definida na RFC 5214, é baseada em túneis IPv6 criados automaticamente dentro da rede IPv4, e em endereços IPv6 associados aos clientes de acordo com o prefixo especificado no roteador ISATAP e no IPv4 do cliente. Para a criação destes túneis são utilizadas as especificações da seção 3 da RFC 4213, que trata o tunelamento através do protocolo IPv4 tipo 41 ou 6in4.

Mais informações:

- RFC 5214 - *Intra-Site Automatic Tunnel Addressing Protocol* (ISATAP)

ISATAP

- Endereçamento
  - Nesta técnica, o endereço IPv4 dos clientes e roteadores são utilizados como parte dos endereços ISATAP. Com isso, um nó ISATAP pode determinar facilmente os pontos de entrada e saída dos túneis IPv6, sem utilizar nenhum protocolo ou recurso auxiliar.
  - O formato do endereço ISATAP segue o seguinte formato:



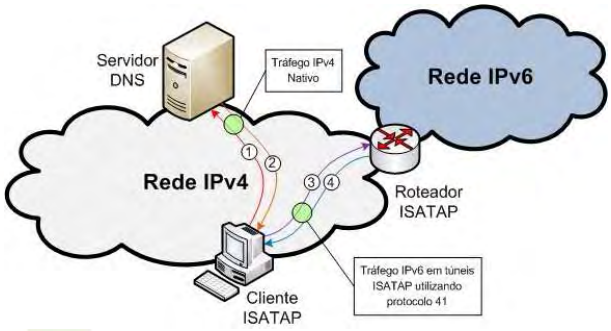
  - **Prefixo unicast :** É qualquer prefixo unicast válido em IPv6, que pode ser link-local (FE80::/64) ou global;
  - **ID IPv4 público ou privado:** Se o endereço IPv4 for público, este campo deve ter o valor "200" e se for privado (192.168.0.0/16, 172.16.0.0/12 e 10.0.0.0/8) o valor do campo é zero;
  - **ID ISATAP:** Sempre tem o valor 5EFE;
  - **Endereço IPv4:** É o IPv4 do cliente ou roteador em formato IPv4;

Exemplos de endereços ISATAP:

Endereço IPv4	Endereço IPv6/ISATAP
250.140.80.1	**2001:10fe:0:8003:200:5efe:250.140.80.1** **fe80::200:5efe:250.140.80.1**
192.168.50.1	**2001:10fe:0:8003:0:5efe:192.168.50.1** **fe80:0:5efe:250.140.80.1**

ISATAP

Início



1- Consulta ao DNS (no caso do Windows, procura por ISATAP.domínio-local)

2- O servidor DNS retorna o IPv4 do roteador ISATAP

3- *Router Solicitation* (encapsulada em v4)

4- *Router Advertisement* (encapsulada em IPv4)

1- Neste passo o cliente tenta determinar o endereço IPv4 do roteador, se o endereço IPv4 já não estiver determinado na sua configuração. No caso do Windows ele tenta por padrão resolver o nome ISATAP e ISATAP.domínio-local via resolução local ou servidor DNS;

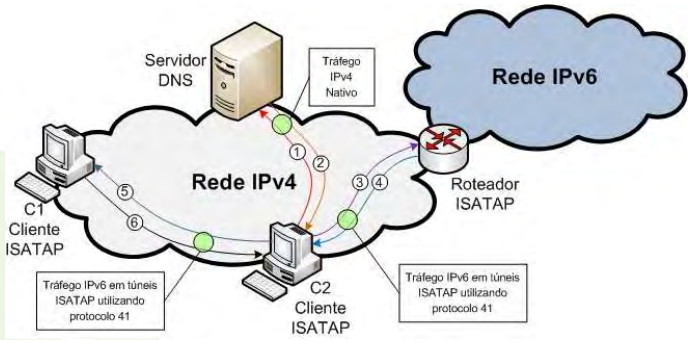
2- O servidor DNS retorna o IPv4 do roteador ISATAP (se for o caso);

3- O cliente envia uma mensagem de solicitação de roteador (RS) encapsulada em IPv4 ao roteador ISATAP;

4- O Roteador ISATAP responde com uma mensagem de Anúncio de Roteador (RA) encapsulada em IPv4, com isso, o cliente já pode configurar os seus endereços IPv6/ISATAP.

ISATAP

Comunicação entre clientes ISATAP na mesma rede



- A comunicação entre os clientes ISATAP numa mesma rede é feita diretamente, sem a interferência do Roteador ISATAP (após autoconfiguração inicial). O tráfego na rede é sempre IPv4, o IPv6 é encapsulado ou desencapsulado localmente nos clientes.

1- C2 solicita a resolução DNS do nome do roteador ISATAP (Se necessário);

2- C2 recebe o IPv4 do roteador ISATAP (Se necessário);

3- O cliente envia uma mensagem de solicitação de roteador (RS) encapsulada em IPv4 ao roteador ISATAP;

4- O Roteador ISATAP responde com uma mensagem de Anúncio de Roteador (RA) encapsulada em IPv4, com isso, o cliente já pode configurar os seus endereços IPv6/ISATAP;

Os processos de 1 a 4 são executados também por C1;

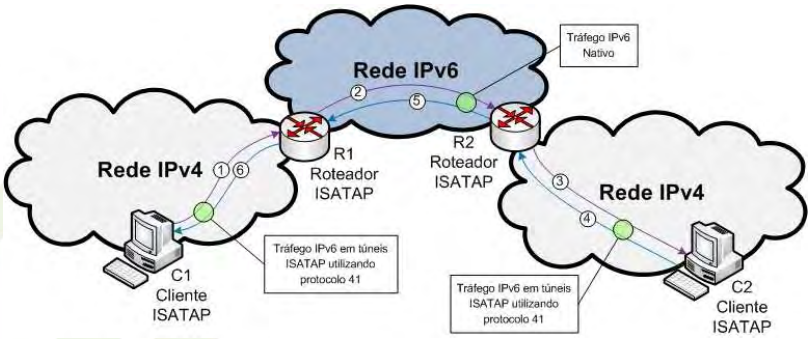
5- O cliente ISATAP C2 envia um pacote IPv6 encapsulado em IPv4 utilizando o protocolo 41 através da rede IPv4 com destino ao endereço IPv4 de C1;

6- O cliente ISATAP C1 recebe o pacote IPv4 e desencapsula o pacote IPv6 dele, após isto, ele responde com um outro pacote IPv6 encapsulado em IPv4 utilizando o protocolo 41 através da rede IPv4 com destino ao endereço IPv4 do cliente C2.

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

ISATAP

Comunicação entre clientes ISATAP em redes diferentes



- O tráfego ISATAP entre clientes de redes IPv4 diferentes, depende dos roteadores ISATAP.
- Na rede IPv4, o tráfego v6 está sempre encapsulado dentro de pacotes v4.
- Entre os roteadores ISATAP diferentes, o tráfego é v6 nativo.

194

cgi.br

1.O cliente ISATAP C1 quer enviar um pacote IPv6 para o cliente C2. Através de sua tabela de roteamento ele descobre que tem que enviá-lo utilizando a interface virtual ISATAP, com isso, o pacote é encapsulado em IPv4 (protocolo 41) e enviado ao endereço IPv4 do roteador R1;

2.O roteador R1 recebe o pacote através de sua interface IPv4, mas, como o pacote IPv6 está encapsulado utilizando o protocolo 41, ele o desencapsula (utilizando a interface virtual ISATAP) e verifica o endereço IPv6 do destino. Depois disso ele descobre que a rota para o destino é através da rede IPv6, sendo assim, o pacote desencapsulado (IPv6 nativo) é encaminhado para o roteador R2;

3.O roteador R2 recebe o pacote IPv6 em sua interface IPv6, mas verificando o endereço de destino, descobre que ele é para o cliente C2 que está na sua subrede ISATAP, sendo assim, ele envia o pacote através desta interface, que encapsula novamente o pacote IPv6 em um pacote IPv4 e o envia a C2 (baseado no IPv4 que está implícito no IPv6). O cliente ISATAP C1 recebe o pacote IPv4 e desencapsula o pacote IPv6 (através da interface virtual ISATAP);

4.O cliente ISATAP C2 quer responder ao cliente C1, sendo assim, ele verifica a sua tabela de rotas e conclui que tem que enviar o pacote através da interface virtual ISATAP, com isso, o pacote IPv6 é encapsulado em IPv4 e encaminhado ao roteador R2;

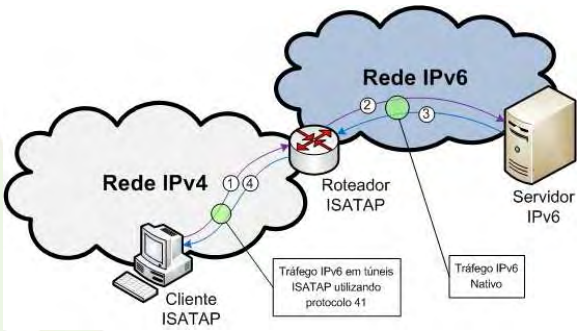
5.O roteador R2 recebe o pacote através de sua interface IPv4, mas, como o pacote está utilizando o protocolo 41, ele desencapsula o pacote IPv6 dele e verificando na sua tabela de roteamento, o encaminha através da sua interface IPv6;

6.O roteador R1 recebe o pacote IPv6, mas, verificando em sua tabela de roteamento descobre que o pacote tem que ser enviado através de sua interface virtual ISATAP, a qual encapsula o pacote IPv6 em IPv4 utilizando o protocolo 41 e o encaminha o IPv4 de C1;

C1 recebe o pacote, mas, como ele está encapsulado utilizando o protocolo 41, ele extrai o pacote IPv6 enviado por C2 e o recebe.

ISATAP

Comunicação entre clientes ISATAP e computadores IPv6



1- O cliente ISATAP quer enviar um pacote IPv6 para o servidor IPv6. Através de sua tabela de roteamento ele descobre que tem que enviá-lo utilizando a interface virtual ISATAP, com isso, o pacote é encapsulado em IPv4(protocolo 41) e enviado ao endereço IPv4 do roteador ISATAP;

2- O roteador ISATAP recebe o pacote através de sua interface IPv4, mas, como o pacote IPv6 está encapsulado utilizando o protocolo 41, ele o desencapsula(utilizando a interface virtual ISATAP) e verifica o endereço IPv6 do destino. Depois disso ele descobre que a rota para o destino é através da rede IPv6, sendo assim, o pacote desencapsulado(IPv6 nativo) é encaminhado para o servidor IPv6. O servidor recebe o pacote IPv6 destinado a ele;

3- O servidor IPv6 quer responder ao cliente ISATAP, sendo assim, verificando sua tabela de roteamento ele descobre que tem que enviar através de sua rota padrão, que é através da rede IPv6;

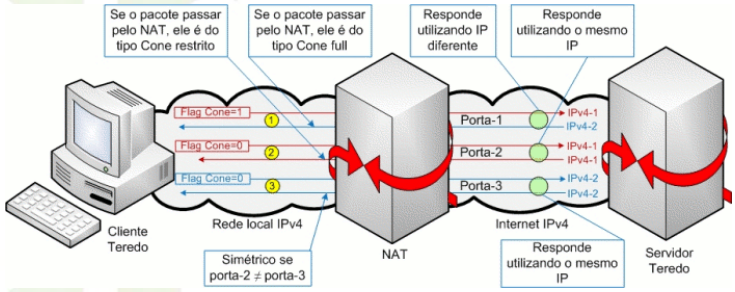
4- Como a rota para a rede do cliente ISATAP é através do roteador ISATAP, o pacote é encaminhado a ele através de sua interface IPv6. Verificando em sua tabela de roteamento o roteador descobre que tem que enviar o pacote através de sua interface virtual ISATAP, sendo assim, o pacote é encapsulado em IPv4 e encaminhado ao cliente ISATAP através da rede IPv4. O cliente recebe o pacote IPv4, mas, como ele está utilizando o protocolo 41, ele desencapsula e recebe o pacote IPv6.

nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

Teredo

- Encapsula o pacote IPv6 em pacotes UDP.
- Funciona através de NAT tipo Cone e Cone Restrito.
- Envia pacotes *bubles* periodicamente ao Servidor para manter as configurações iniciais da conexão UDP.
- Seu funcionamento é complexo e apresenta *overhead.*



cgi.br

A técnica de tunelamento automática Teredo, definida na RFC 4380, permite que nós localizados atrás de *Network Address Translations* (NAT), obtenham conectividade IPv6 utilizando o protocolo UDP.

A conexão é realizada através de um Servidor Teredo que a inicializa, e determinar o tipo de NAT usada pelo cliente. Em seguida, caso o *host* de destino possua IPv6 nativo, um *Relay* Teredo é utilizado para criar uma interface entre o Cliente e o *host* de destino. O *Relay* utilizado será sempre o que estiver mais próximo *host* de destino, e não o mais próximo ao cliente.

Esta técnica não é muito eficiente devido ao *overhead* e a complexidade de seu funcionamento, entretanto, quando o *host* está atrás de NAT, ela é uma das únicas opções.

Por padrão, o Windows Vista já traz o Teredo instalado e ativado, enquanto que no Windows XP, 2003 e 2008, ele vem apenas instalado. Quanto ao FreeBSD e ao Linux, ele não vem instalado.

Para facilitar a compreensão do funcionamento deste tipo de túnel, apresentaremos no quadro a seguir um resumo dos quatro tipos de NAT existentes:

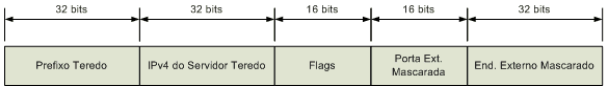
NAT Cone - todas as requisições originadas de um mesmo endereço e porta internos são mapeadas para uma mesma porta do NAT. Com isso, é preciso apenas conhecer o endereço público do NAT e a porta associada a um nó interno para que um nó externo estabeleça uma comunicação, não importando seu endereço ou porta, podendo assim, enviar arbitrariamente pacotes para o nó interno. É também conhecido como NAT Estático.

NAT Cone Restrito - todas as requisições originadas de um mesmo endereço e porta internos são mapeadas para uma mesma porta do NAT. No entanto, o acesso externo é permitido apenas em respostas a requisições feitas previamente, sendo que o endereço do nó externo, que está respondendo a requisição, deve ser o mesmo utilizado como endereço de destino da requisição. É também conhecido como NAT Dinâmico.

NAT Cone Restrito por Porta - Têm as mesmas características de mapeamento do NAT Cone Restrito, porém, a restrição para a comunicação considera também a porta do nó externo. Com isso, um nó externo só poderá estabelecer uma comunicação com um nó interno se este último houver lhe enviado previamente um pacote através da mesma porta e endereço.

NAT Simétrico - Além de possuir as mesmas restrições do NAT tipo Cone Restrito por Porta, cada requisição realizada a partir de um endereço e porta internos para um endereço e porta externos é mapeada unicamente no NAT. Ou seja, se o mesmo endereço interno envia uma requisição, com a mesma porta, porém para um endereço de destino diferente, um mapeamento diferente será criado no NAT. Este tipo de NAT é também conhecido como NAT *Masquerade* ou NAT *Stateful*.

Teredo



- Utiliza o prefixo **2001:0000::/32**.
- Os 32 bits seguintes contém o endereço IPv4 do Servidor Teredo.
- Os 16 bits seguintes são utilizados para definir *flags* que indicam o tipo de NAT utilizado e introduzem uma proteção adicional ao nó contra ataques de *scan*.
- Os próximos 16 bits indicam a porta UDP de saída do NAT.
- Os últimos 32 bits representam o endereço IPv4 público do Servidor NAT.

Baseado nas mensagens RA recebidas dos Servidores, o cliente constrói seu endereço IPv6, utilizando o seguinte padrão:

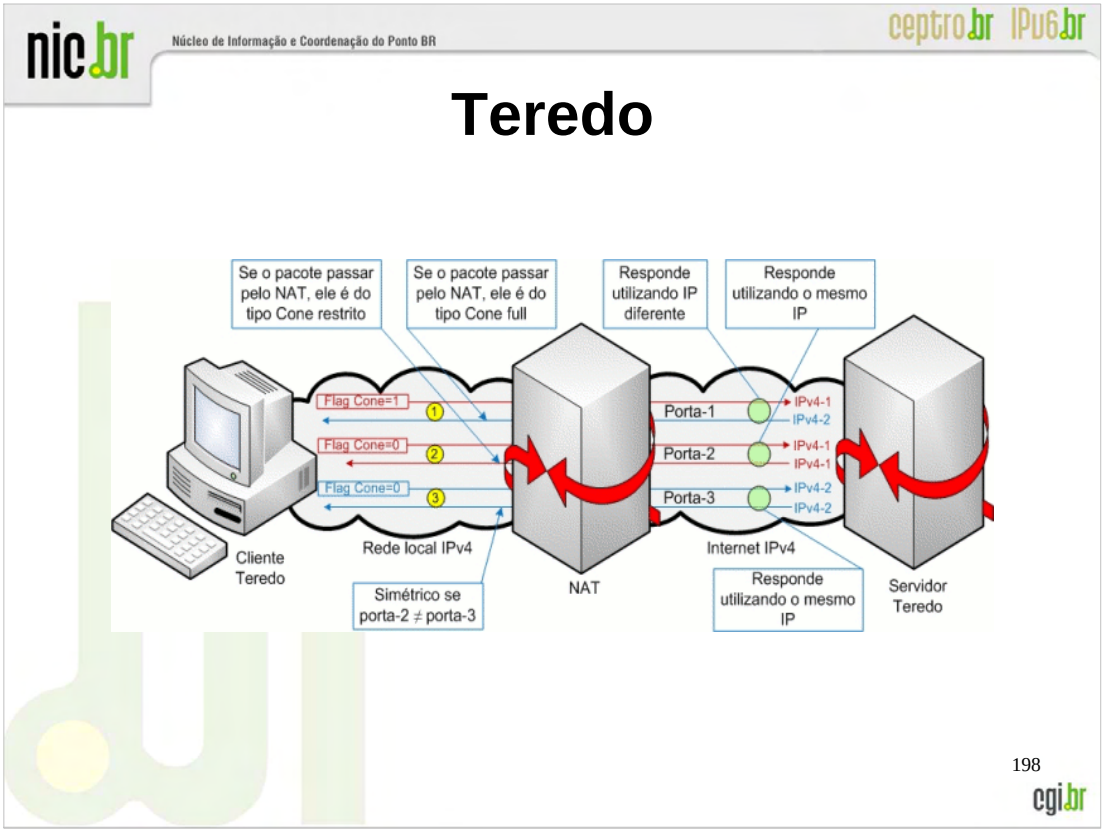
* Os bits 0 a 31 são o prefixo do Teredo recebido do Servidor através das mensagens RA; o padrão é **2001:0000**;

* Os bits 32 a 63 são o endereço IPv4 primário do Servidor Teredo que enviou a primeira mensagem RA;

* Os bits 64 a 79 são utilizados para definir alguns *flags*. Normalmente, somente o bit mais significativo é utilizado, sendo que ele é marcado como 1 se o Cliente está atrás de NAT do tipo Cone, caso contrário ele é marcado como 0. Apenas o Windows Vista e Windows Server 2008 utilizam todos os 16 bits, que seguem o formado "CRAAAAUG AAAAAAAA", onde "C" continua sendo a *flag* Cone; o bit R é reservado para uso futuro; o bit U define a *flag* Universal/Local (o padrão é 0); o bit G define a *flag* Individual/Group (o padrão é 0); e os 12 bits “A” são randomicamente determinados pelo Cliente para introduzir uma proteção adicional ao nó contra ataques de *scan*;

* Os bits 80 a 95 são a porta UDP de saída do NAT, recebida nas mensagens RA e mascarada através da inversão de todos os seus bits. Este mascaramento é necessário porque alguns NATs procuram portas locais dentro dos pacotes e os substituem pela porta pública ou vice-versa;

* Os bits 96 a 127 são o endereço IPv4 público do Servidor NAT, mascarado através da inversão de todos os seus bits. Este mascaramento é necessário porque alguns NATs procuram endereços IPs locais dentro dos pacotes e os substituem pelo endereço público ou vice-versa.



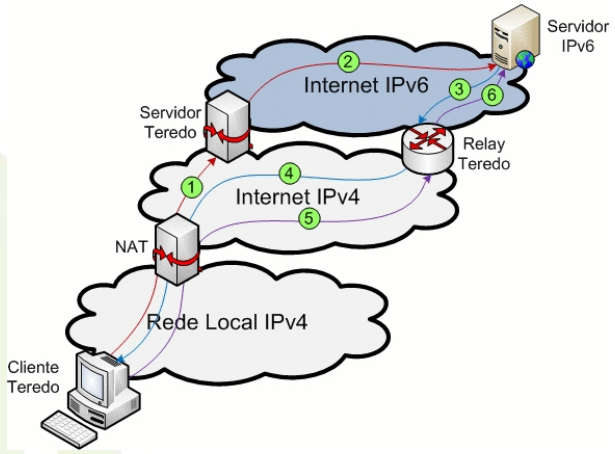
1- Uma mensagem *Router Solicitation* (RS) é enviada ao servidor Teredo 1 (servidor primário) com o *flag* de NAT tipo Cone ativado, o servidor Teredo 1 então responde com uma mensagem de *Router Advertisement* (RA). Como a mensagem RS estava com o Cone *flag* ativado, o servidor Teredo 1 envia a mensagem RA utilizando um endereço IPv4 alternativo. Com isso o cliente conseguirá determinar se o NAT que ele está utilizando é do tipo Cone, se ele receber a mensagem de RA;

2- Se a mensagem RA do passo anterior não for recebida, o cliente Teredo envia uma outra mensagem RS, mas, agora com o Cone *flag* desativado. O servidor Teredo 1 responde novamente com uma mensagem RA, mas, como o Cone *flag* da mensagem RS estava desativado, ele responde utilizando o mesmo endereço IPv4 em que ele recebeu a mensagem RS. Se agora o cliente receber a mensagem de RA, então ele conclui que está utilizando NAT do tipo restrita;

3- Para ter certeza que o cliente Teredo não está utilizando um NAT do tipo simétrico, ele envia mais uma mensagem RS, mas, agora para o servidor secundário Teredo 2, o qual responde com uma mensagem do tipo RA. Quando o cliente recebe a mensagem RA do servidor Teredo 2, ele compara o endereço e a porta UDP de origem contidos na mensagem RA recebidas dos dois servidores, se forem diferentes o cliente conclui que está utilizando NAT do tipo simétrico, o qual não é compatível com o Teredo.

Teredo

Comunicação através de NAT tipo CONE



1- Para iniciar a comunicação, primeiro o cliente Teredo tem que determinar o endereço IPv4 e a porta UDP do Relay Teredo que estiver mais próximo do *host* IPv6, para isto, ele envia uma mensagem ICMPv6 *echo request* para o *host* IPv6 via o seu servidor Teredo;

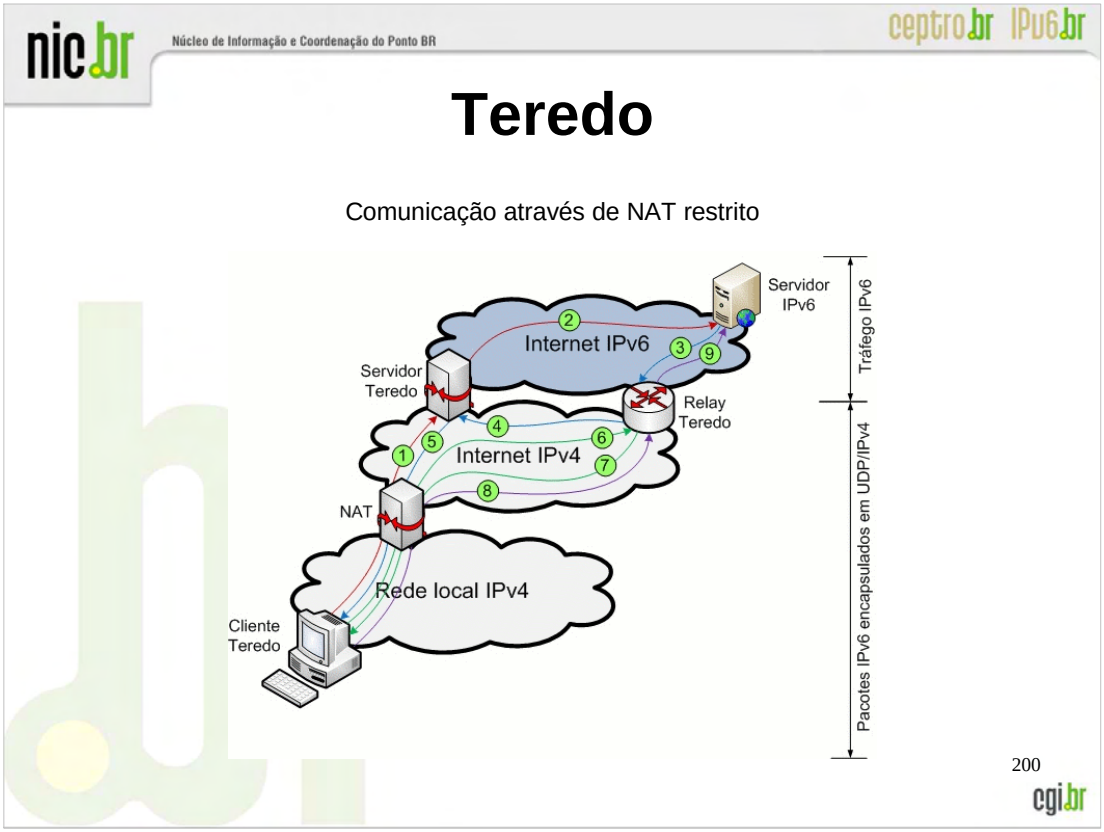
2- O servidor Teredo recebe a mensagem ICMPv6 *echo request* e a encaminha para o *host* IPv6 através da rede IPv6;

3- O *host* IPv6 responde ao cliente Teredo com uma mensagem ICMPv6 *Echo Reply* que é roteada através do *Relay* Teredo mais próximo dele;

4- O *Relay* Teredo então encapsula a mensagem ICMPv6 *Echo Reply* e envia diretamente ao cliente Teredo. Como o NAT utilizado pelo cliente é do tipo Cone, o pacote enviado pelo *Relay* Teredo é encaminhado para o cliente Teredo;

5- Como o pacote retornado pelo *Relay* Teredo contém o endereço IPv4 e a porta UDP utilizada por ele, o cliente Teredo extrai estas informações do pacote. Depois disso um pacote inicial é encapsulado e enviado diretamente pelo cliente Teredo para o endereço IPv4 e porta UDP do *Relay* Teredo;

6- O *Relay* Teredo recebe este pacote, remove o cabeçalho IPv4 e UDP e o encaminha para o *host* IPv6. Depois disso toda a comunicação entre o cliente Teredo e o *host* IPv6 é feita via o *relay* Teredo através deste mesmo método.



1- Para iniciar a comunicação, primeiro o cliente Teredo tem que determinar o endereço IPv4 e a porta UDP do *Relay* Teredo que estiver mais próximo do *host* IPv6, para isto, ele envia uma mensagem ICMPv6 *echo request* para o *host* IPv6 via o seu servidor Teredo;

2- O servidor Teredo recebe a mensagem ICMPv6 *echo request* e a encaminha para o *host* IPv6 através da rede IPv6;

3- O *host* IPv6 responde ao cliente Teredo com uma mensagem ICMPv6 *Echo Reply* que é roteada através do *Relay* Teredo mais próximo dele;

4- Através do pacote recebido, o *Relay* Teredo descobre que o cliente Teredo está utilizando um NAT do tipo restrito, sendo assim, se o *Relay* Teredo enviar o pacote ICMPv6 diretamente para o cliente Teredo, ele será descartado pelo NAT porque não há mapeamento pré-definido para tráfego entre o cliente e o *Relay* Teredo, com isso, o *Relay* Teredo envia um pacote "*bubble*" para o cliente Teredo através do Servidor Teredo utilizando a rede IPv4;

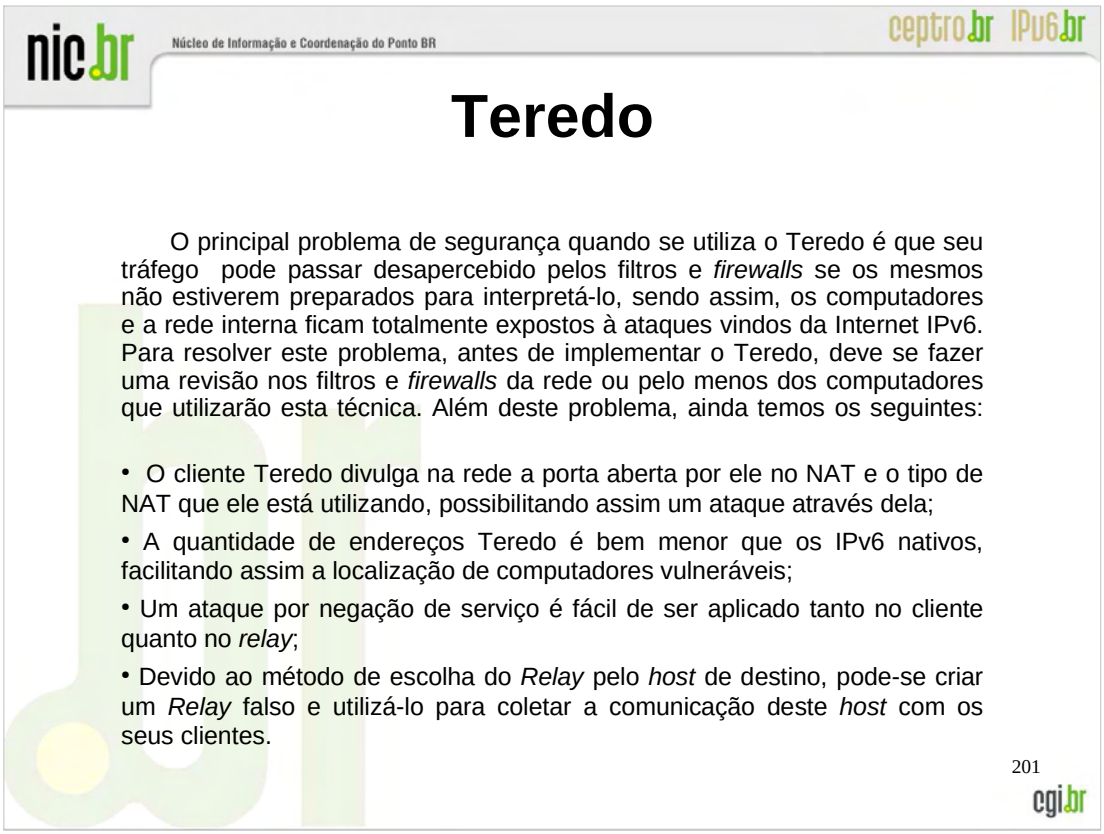
5- O servidor Teredo recebe o pacote "*Bubble*" do *Relay* Teredo e o encaminha para o cliente Teredo, mas coloca no indicador de origem o IPv4 e a porta UDP do *Relay* Teredo. Como já havia um mapeamento de tráfego entre o servidor Teredo e o Cliente Teredo, o pacote passa pelo NAT e é entregue ao Cliente Teredo;

6- O Cliente Teredo extrai do pacote "*Bubble*" recebido o IPv4 e a porta UDP do *Relay* Teredo mais próximo do *host* IPv6, com isso, o Cliente Teredo envia um pacote "*Bubble*" para o *Relay* Teredo, para que seja criado um mapeamento de conexão entre eles no NAT;

7- Baseado no conteúdo do pacote "*Bubble*" recebido, o *Relay* Teredo consegue determinar que ele corresponde ao pacote ICMPv6 *Echo Reply* que está na fila para a transmissão e também que a passagem através do NAT restrito já está aberta, sendo assim, ele encaminha o pacote ICMPv6 *Echo Reply* para o cliente Teredo;

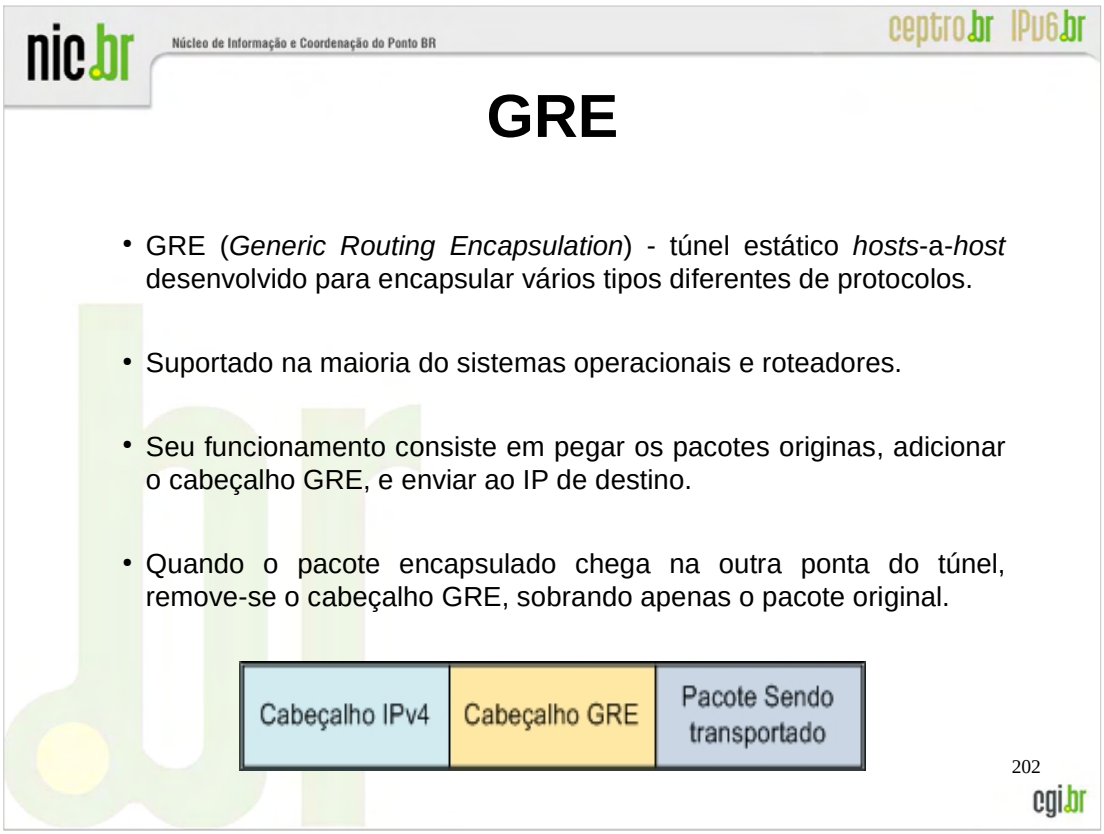
8- Depois de recebido o pacote ICMPv6, um pacote inicial é então enviado do Cliente Teredo para o *host* IPv6 através do *Relay* Teredo mais próximo dele;

9- O *relay* Teredo remove os cabeçalhos IPv4 e UDP do pacote e o encaminha através da rede IPv6 para o *host* IPv6. Após isto os pacotes subseqüentes são enviados através do *Relay* Teredo.



Mais informações:

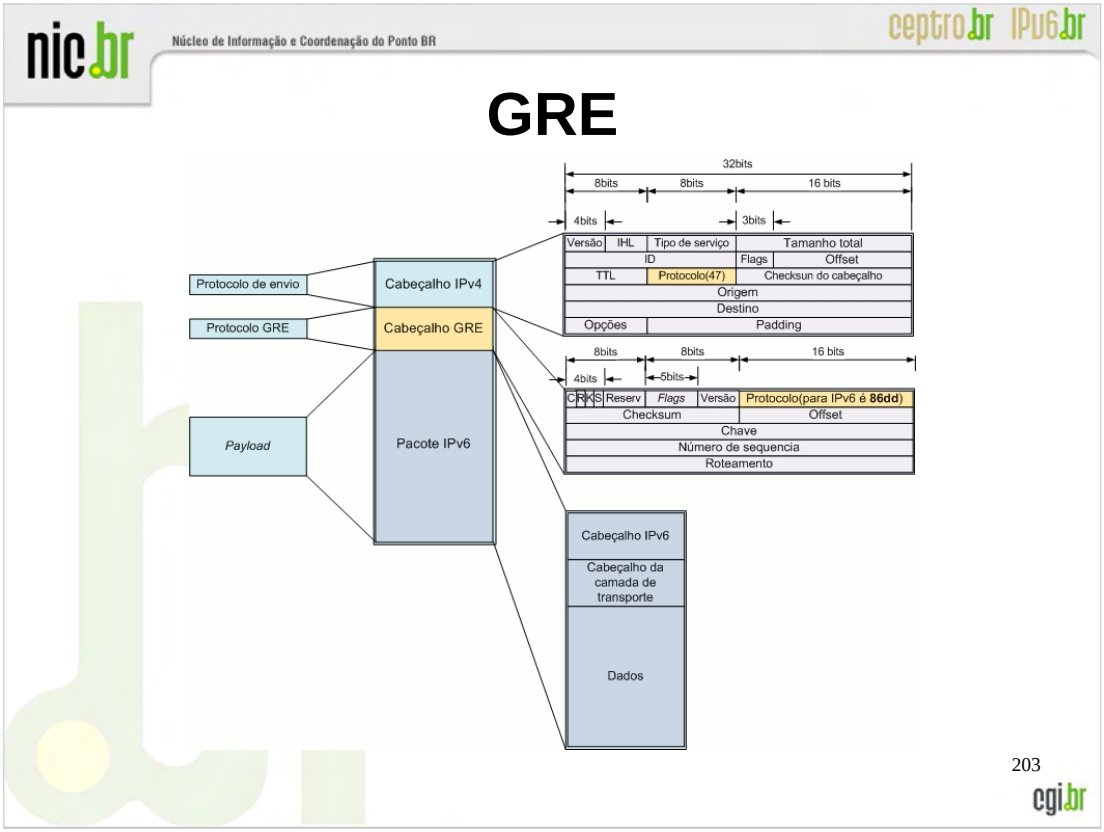
- RFC 4380 - *Teredo: Tunneling IPv6 over UDP through Network Address Translations* (NATs)



O GRE (*Generic Routing Encapsulation*) é um túnel estático entre dois *hosts* originalmente desenvolvido pela Cisco com a finalidade de encapsular vários tipos diferentes de protocolos, como por exemplo IPv6 e IS-IS (veja a lista completa de protocolos suportados em http://www.iana.org/assignments/ethernet-numbers). Este tipo de encapsulamento é suportado na maioria do sistemas operacionais e roteadores e consiste em um *link* ponto a ponto. A principal desvantagem do túnel GRE é a configuração manual, que de acordo com a quantidade de túneis, gerará um grande esforço na sua manutenção e gerenciamento.

Mais informações:

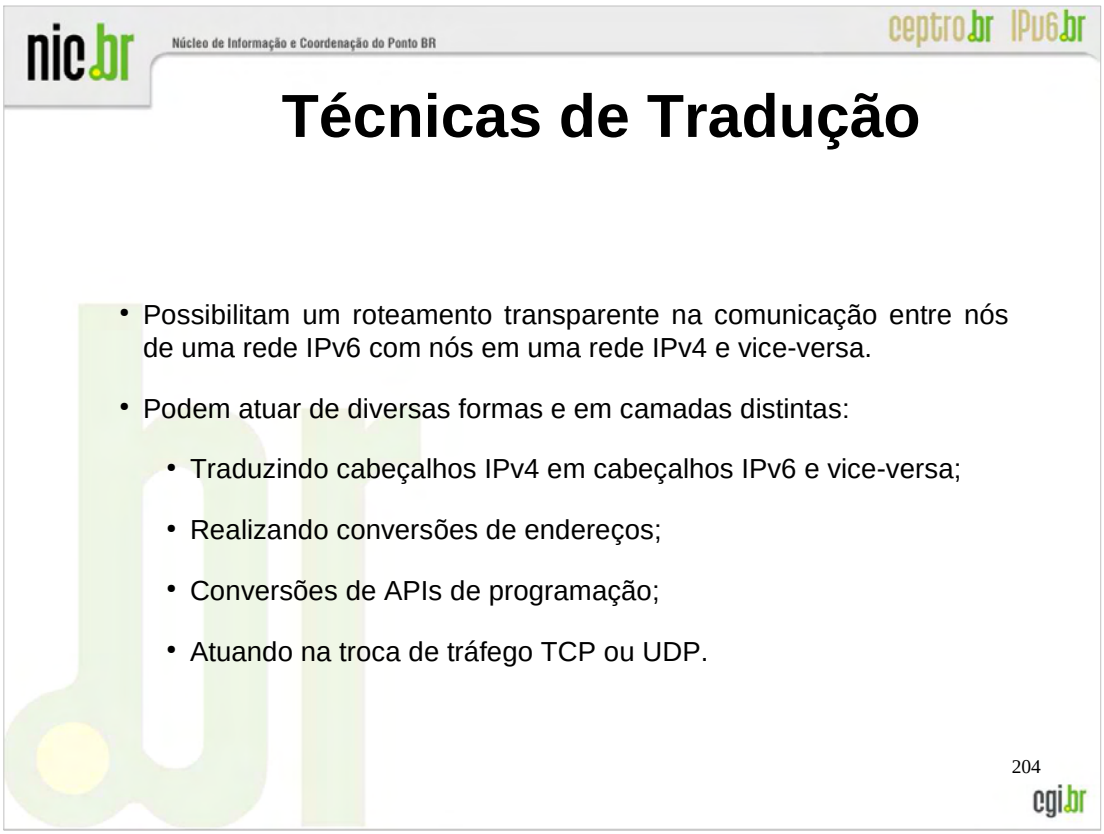
- RFC 2784 - *Generic Routing Encapsulation* (GRE)



O funcionamento deste tipo de túnel é muito simples, e consiste em pegar os pacotes originas, adicionar o cabeçalho GRE, e enviar ao IP de destino (o endereço do destino é especificado no cabeçalho GRE), quando o pacote encapsulado chega na outra ponta do túnel (IP de destino) é removido dele o cabeçalho GRE, sobrando apenas o pacote original, o qual é encaminhado normalmente ao destinatário. Como estamos mais preocupados com os pacotes IPv6, no desenho abaixo podemos ver a estrutura de um pacote IPv6 sendo transportado em um túnel GRE:

Os campos mais importantes do cabeçalho GRE são os seguintes:

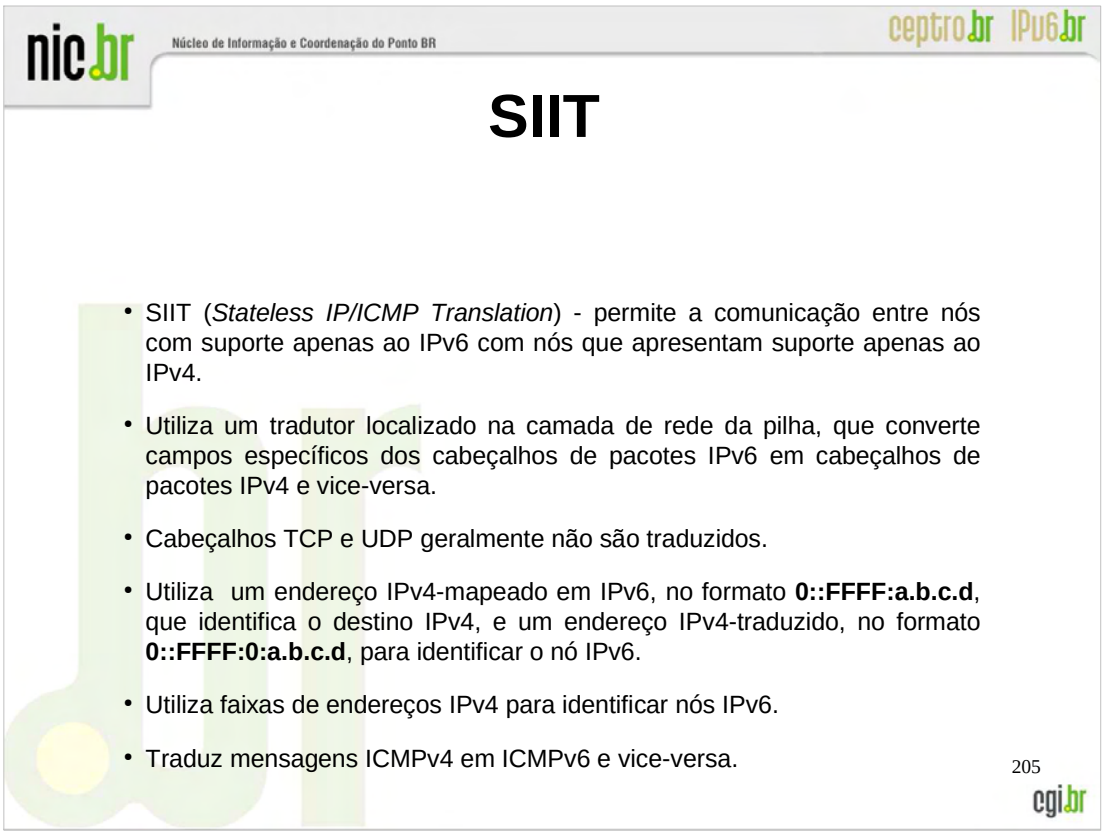
- C (*Checksum*): Se for 1, indica que existe o campo *Checksum* e que há informações válidas nele e no Offset;
- R (*Routing*): Se for 1, indica que existe o campo Roteamento e que há informações de roteamento válidas nele e no *Offset*;
- K (*Key*): Se for 1, indica que o campo Chave existe e está sendo utilizado;
- S (*Sequence*): Se for 1, indica que o campo Número de seqüência existe e está sendo utilizado;
- Versão: Geralmente preenchido com 0;
- Protocolo: É preenchido com o código do protocolo sendo transportado, de acordo com os tipos de pacotes ethernet (http://www.iana.org/assignments/ethernet-numbers);
- *Offset*: Indica a posição onde inicia o campo de roteamento;
- *Checksum*: Contém o *checksum* IP (complemento de 1) do cabeçalho GRE e do pacote sendo transportado;
- Chave (*Key*): Contém um número de 32 bits que é inserido pelo encapsulador. Ele é utilizado pelo destinatário para identificar o remetente do pacote;
- Número de seqüência (*Sequence number*): Contém um número inteiro de 32 bits que é inserido pelo remetente do pacote. Ele é utilizado pelo destinatário para seqüenciar os pacotes recebidos;
- Roteamento (*Routing*): Contém uma lista de entradas de roteamento, mas, geralmente não é utilizado.



As técnicas de tradução possibilitam um roteamento transparente na comunicação entre nós que apresentem suporte apenas a uma versão do protocolo IP, ou utilizem pilha dupla. Estes mecanismos podem atuar de diversas formas e em camadas distintas, traduzindo cabeçalhos IPv4 em cabeçalhos IPv6 e vice-versa, realizando conversões de endereços, de APIs de programação, ou atuando na troca de tráfego TCP ou UDP.

Mais informações:

• RFC 4966 - *Reasons to Move the Network Address Translator - Protocol Translator* (NAT-PT) *to Historic Status*



SIIT (*Stateless IP/ICMP Translation Algorithm*) - definido na RFC 2765, o SIIT é um mecanismo de tradução *stateless* de cabeçalhos IP/ICMP, permitindo a comunicação entre nós com suporte apenas ao IPv6 com nós que apresentam suporte apenas ao IPv4. Ele utiliza um tradutor localizado na camada de rede da pilha, que converte campos específicos dos cabeçalhos de pacotes IPv6 em cabeçalhos de pacotes IPv4 e vice-versa. Para realizar este processo, o tradutor necessita de um endereço IPv4-mapeado em IPv6, no formato **0::FFFF:a.b.c.d**, que identifica o destino IPv4, e um endereço IPv4-traduzido, no formato **0::FFFF:0:a.b.c.d**, para identificar o nó IPv6. Quando o pacote chega ao SIIT, o cabeçalho é traduzido, convertendo o endereço para IPv4 e encaminhado ao nó de destino;

Mais informações:

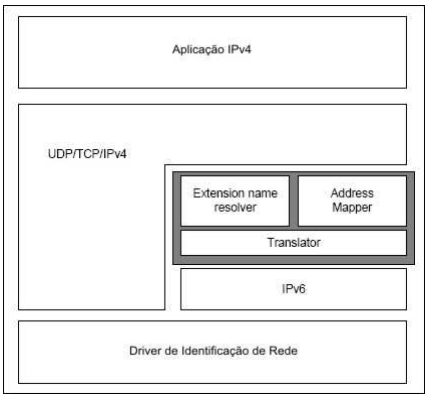
- RFC 2765 - *Stateless IP/ICMP Translation Algorithm (SIIT)*

nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

BIS

- BIS *(Bump-in-the-Stack)* - funciona entre a camada de aplicação e a de rede.
- Utilizada para suportar aplicações IPv4 em redes IPv6.
- Adiciona a pilha IPv4 três módulos:
  - *Translator* - traduz cabeçalhos IPv4 em cabeçalhos IPv6 e vice-versa;
  - *Address mapper* - possui uma faixa de endereços IPv4 que são associados a endereços IPv6 quando o *translator* receber um pacote IPv6;
  - *Extension name resolver* - atua nas consultas DNS realizadas pela aplicação IPv4.
  - Não funciona em comunicações *multicast*.



206

cgi.br

BIS (*Bump in the Stack*) - esse método possibilita a comunicação de aplicações IPv4 com nós IPv6. Definida na RFC 2767, o BIS funciona entre a camada de aplicação e a de rede, adicionando à pilha IPv4 três módulos: translator, que traduz os cabeçalhos IPv4 enviados em cabeçalhos IPv6 e os cabeçalhos IPv6 recebidos em cabeçalhos IPv4; *extension name resolver*, que atua nas DNS *queries* realizadas pelo IPv4, de modo que, se o servidor DNS retorna um registro AAAA, o resolver pede ao *address mapper* para atribuir um endereço IPv4 correspondente ao endereço IPv6; e *address mapper*, que possui uma certa quantidade de endereços IPv4 para associar a endereços IPv6 quando o translator receber um pacote IPv6. Como os endereços IPv4 não são transmitidos na rede, eles podem ser endereços endereços privados. Esse método permite apenas a comunicação de aplicações IPv4 com *hosts* IPv6, e não o contrário, além de não funcionar em comunicações *multicast*.

Como os endereços IPv4 não são transmitidos na rede, eles podem ser endereços privados. Esse método permite apenas a comunicação de aplicações IPv4 com *hosts* IPv6, e não o contrário, além de não funcionar em comunicações *multicast*.

Mais informações:

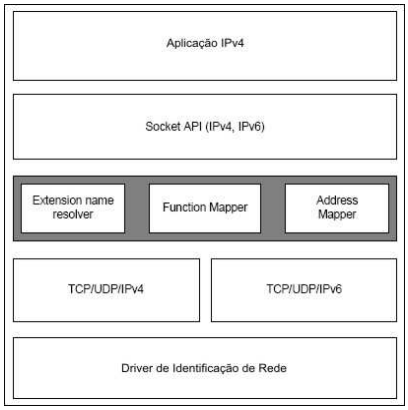
- RFC 2767 - *Dual Stack Hosts using the "Bump-In-the-Stack" Technique (BIS)*

nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

BIA

- BIA (*Bump in the API*) - similar ao BIS, traduz funções da API IPv4 em funções da API IPv6 e vice-versa.
- Três módulos são adicionados, ***extension name resolver*** e ***address mapper***, que funcionam da mesma forma que no BIS, e o ***function mapper***, que detecta as chamadas das funções do *socket* IPv4 e invoca as funções correspondentes do *socket* IPv6 e vice-versa.
- Também utiliza faixas de endereços IPv4.
- Não suporta comunicações *multicast;*



BIA (*Bump in the API*) - similar ao BIS, esse mecanismo adiciona uma API de tradução entre o *socket* API e os módulos TPC/IP dos *hosts* de pilha dupla, permitindo a comunicação de aplicações IPv4 com *hosts* IPv6, traduzindo as funções do *socket* IPv4 em funções do *socket* IPv6 e vice-versa. Conforme descrito na RFC 3338, três módulos são adicionados, *extension name resolver* e *address mapper*, que funcionam da mesma forma que no BIS, e o *function mapper*, que detecta as chamadas das funções do *socket* IPv4 e invoca as funções correspondentes do *socket* IPv6 e vice-versa. O BIA apresenta duas vantagens em relação ao BIS: não depender do *driver* da interface de rede e não introduzir *overhead* na tradução dos cabeçalhos dos pacotes. No entanto, ele também não suporta comunicações *multicast*.

Mais informações:

- RFC 3338 - *Dual Stack Hosts Using "Bump-in-the-API* (BIA)

TRT (*Transport Relay Translator*) - atuando como um tradutor de camada de transporte, esse mecanismo possibilita a comunicação entre *hosts* apenas IPv6 e *hosts* apenas IPv4 através de tráfego TCP/UDP. Sem a necessidade de se instalar qualquer tipo de *software*, o TRT roda em máquinas com pilha dupla que devem ser inseridas em um ponto intermediário dentro da rede. Na comunicação de um *host* IPv6 com um *host* IPv4, conforme definição na RFC 3142, é adicionado um prefixo IPv6 falso ao endereço IPv4 do destino. Quando um pacote com esse prefixo falso passa pelo TRT, esse pacote é interceptado e enviado ao host IPv4 de destino em um pacote TCP ou UDP. Na tradução TCP e UDP o *checksum* deve ser recalculado e apenas no caso das conexões TCP, o estado do *socket* sobre o qual o *host* está conectado deve ser mantido, removendo-o quando a comunicação for finalizada. Para que o mecanismo funcione de forma bidirecional, é necessário a adição de um bloco de endereços IPv4 públicos e o uso de um servidor DNS-ALG para mapear os endereços IPv4 para IPv6.

Mais informações:

- RFC 3142 - *An IPv6-to-IPv4 Transport Relay Translator*

ALG e DNS-ALG

- ALG (*Application Layer Gateway*) - trabalha como um *proxy* HTTP.
- O cliente inicia a conexão com o ALG, que estabelece uma conexão com o servidor, retransmitindo as requisições de saída e os dados de entrada.
- Em redes apenas IPv6, o ALG habilita a comunicação dos *hosts* com serviços em redes apenas IPv4, configurando o ALG em nós com pilha dupla.
- É normalmente utilizado quando o *host* que deseja acessar a aplicação no servidor IPv4, está atrás de NAT ou de um *firewall*.

- DNS-ALG – traduz consultas DNS do tipo AAAA vindo de um *host* IPv6, para uma consulta do tipo A, caso o servidor de nomes a ser consultado se encontre no ambiente IPv4, e vice-versa.

Segurança

- Com a utilização de Pilha Dupla as aplicações ficam expostas aos ataques a ambos os protocolos, IPv6 e IPv4, o que pode ser resolvido configurando *firewalls* específicos para cada protocolo.
- As técnicas de Túneis e Tradução são as que causam maiores impacto do ponto de vista de segurança.
- Mecanismos de tunelamento são suscetíveis a ataques de DoS, falsificação de pacotes e de endereços de roteadores e *relays* utilizados por essas técnicas, como 6to4 e TEREDO.
- As técnicas de tradução implicam em problemas relacionados a incompatibilidade dessas técnicas com alguns mecanismos de segurança existentes. Similar ao que ocorre com o NAT no IPv4.

Segurança

- Como se proteger:
  - Utilizar pilha dupla na migração, protegendo as duas pilhas com *firewall*;
  - Dar preferência aos túneis estáticos, no lugar dos automáticos;
  - Permitir a entrada de tráfego apenas de túneis autorizados.

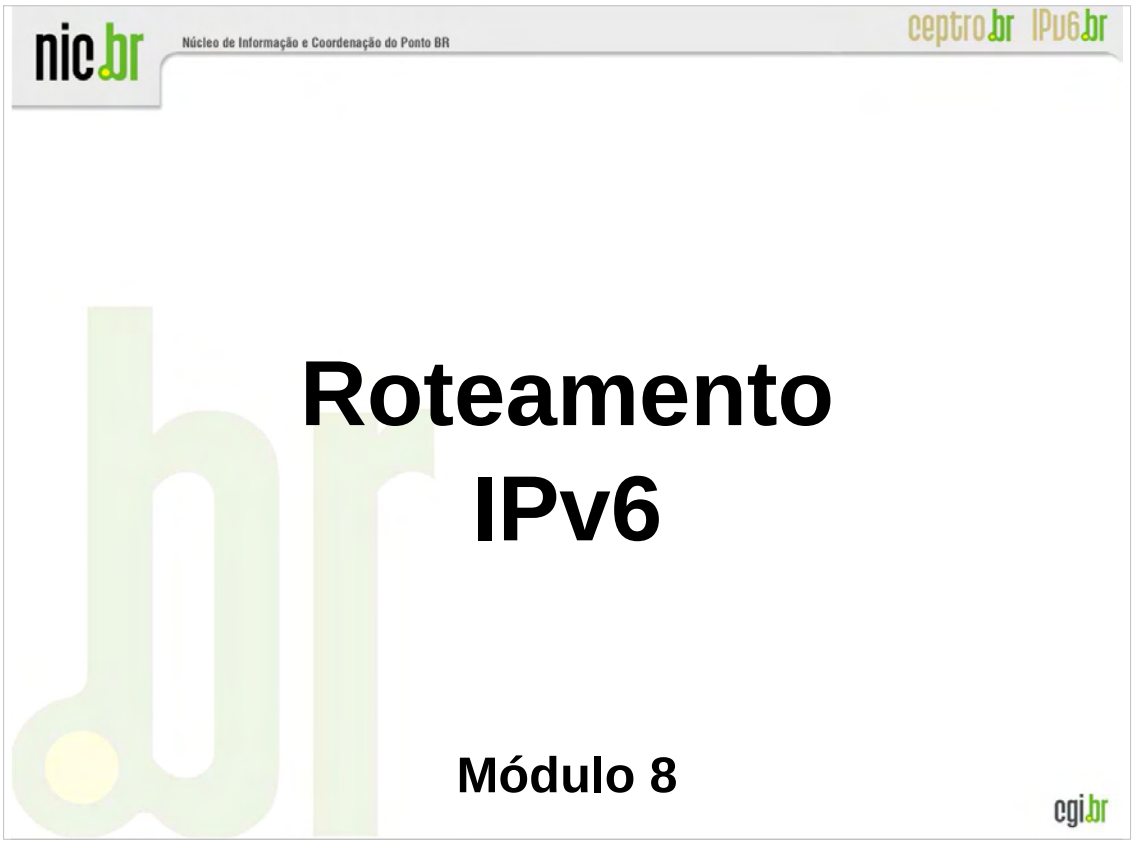
nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

IPv6.br

A Nova Geração do Protocolo Internet

cgi.br



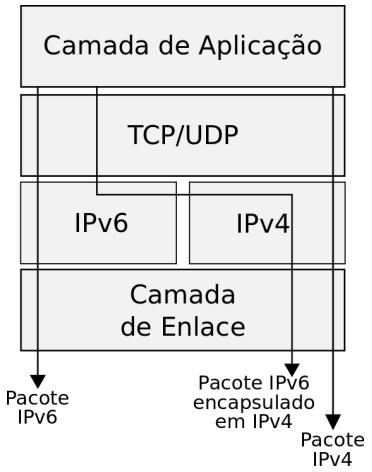
Neste módulo apresentaremos algumas características básicas sobre o funcionamento dos mecanismos de roteamento, tanto interno (IGP) quanto externo (EGP), sempre destacando as principais mudanças em relação ao IPv6. Serão abordados os protocolos de roteamento RIP, OSPF, IS-IS e BGP.

nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

Considerações Importantes

- IPv4 e IPv6 → Camada de Rede
- Duas redes distintas
  - Planejamento
  - Suporte
  - *Troubleshooting*
  - Arquitetura dos equipamentos
  - ...

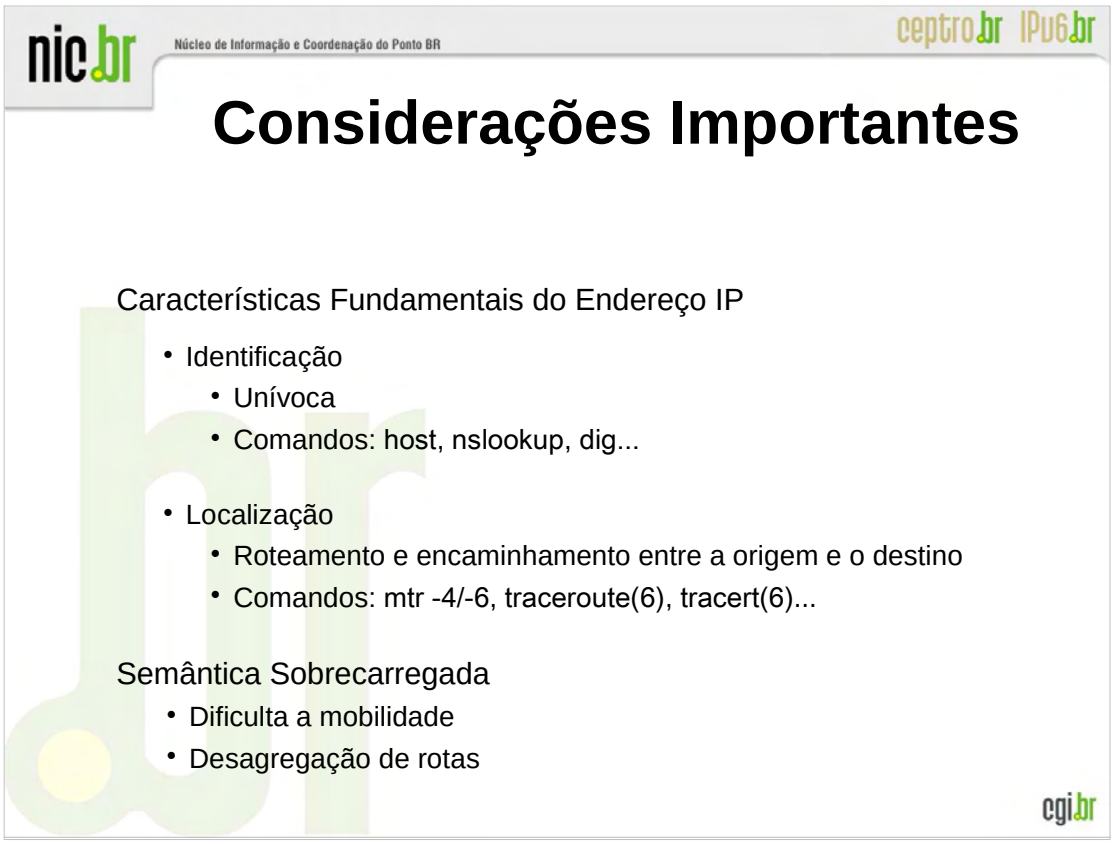


cgi.br

O IPv4 e o IPv6 são protocolos da Camada de Rede, de modo que esta é a única camada diretamente afetada com a implantação do IPv6, sem a necessidade de alterações no funcionamento das demais.

Porém, é preciso compreender que são duas Camadas de Rede distintas e independentes. Isto implica em algumas considerações importantes:

- Como atuar no planejamento e estruturação das redes:
  - Migrar toda a estrutura para Pilha Dupla; migrar apenas áreas críticas; manter duas estruturas distintas, uma IPv4 e outra IPv6; etc.
  - Em redes com Pilha Dupla algumas configurações devem ser duplicadas como DNS, *firewall* e protocolos de roteamento.
- No suporte e resolução de problemas será preciso detectar se há falhas na conexão da rede IPv4 ou da rede IPv6;
- Novos equipamentos e aplicações precisam ter suporte às funcionalidades dos dois protocolos.

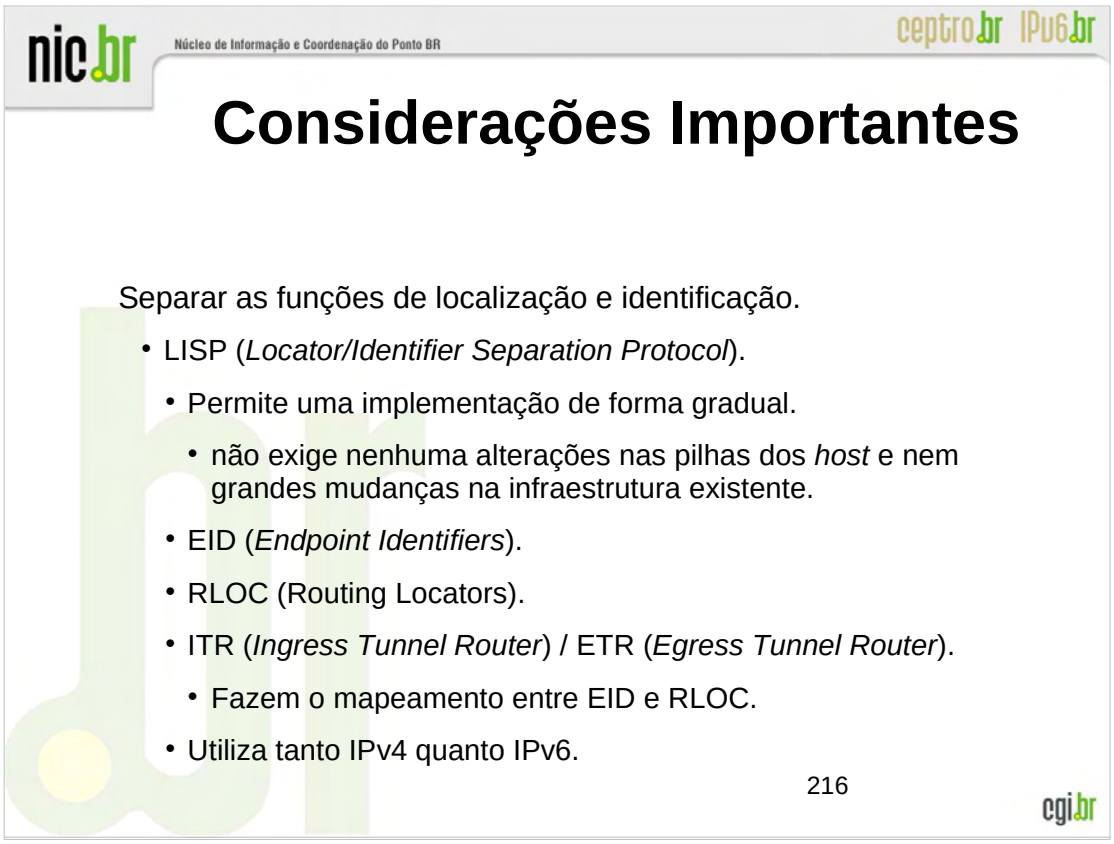


A Camada de Rede está associada principalmente a duas características:

- **Identificação** – deve garantir que cada dispositivo da rede seja identificado de forma unívoca, sem chance de erro. Isto é, o endereço IP deve ser único no mundo. Utilizando o comando host, em plataformas UNIX, ou nslookup, em plataformas Windows, pode-se ver a identificação de um serviço, por exemplo. Em redes com Pilha Dupla, um nó será identificado pelos dois endereços.

- **Localização** - indica como chegar ao destino, tomando as decisões de encaminhamento dos pacotes baseando-se no endereçamento, ocorrendo da mesma forma tanto em IPv4 quanto em IPv6. Podemos verificar esta funcionalidade utilizando comandos como *mtr -4* e *-6*, ou *traceroute (traceroute6)*, ou *tracert (tracert6)*. Estes comandos mostram a identificação e a localização de um nó.

A união dessas duas características na Camada de Rede torna a semântica do endereço IP sobrecarregada. Isto implica em questões como a desagregação de rotas, agravando o problema do crescimento da tabela de roteamento global. Uma forma de impedir isso é separar as funções de localização e identificação.



Existe um grupo de trabalho no IETF que discute uma forma de separar essas duas funções (identificação e localização). O LISP (*Locator/Identifier Separation Protocol*) é um protocolo simples que busca separar os endereços IP em *Endpoint Identifiers* (EIDs) e *Routing Locators* (RLOCs). Ele não exige nenhuma alteração nas pilhas dos *host* e nem grandes mudanças na infraestrutura existente, podendo ser implementado em um número relativamente pequeno de roteadores.

Seus principais elementos são:

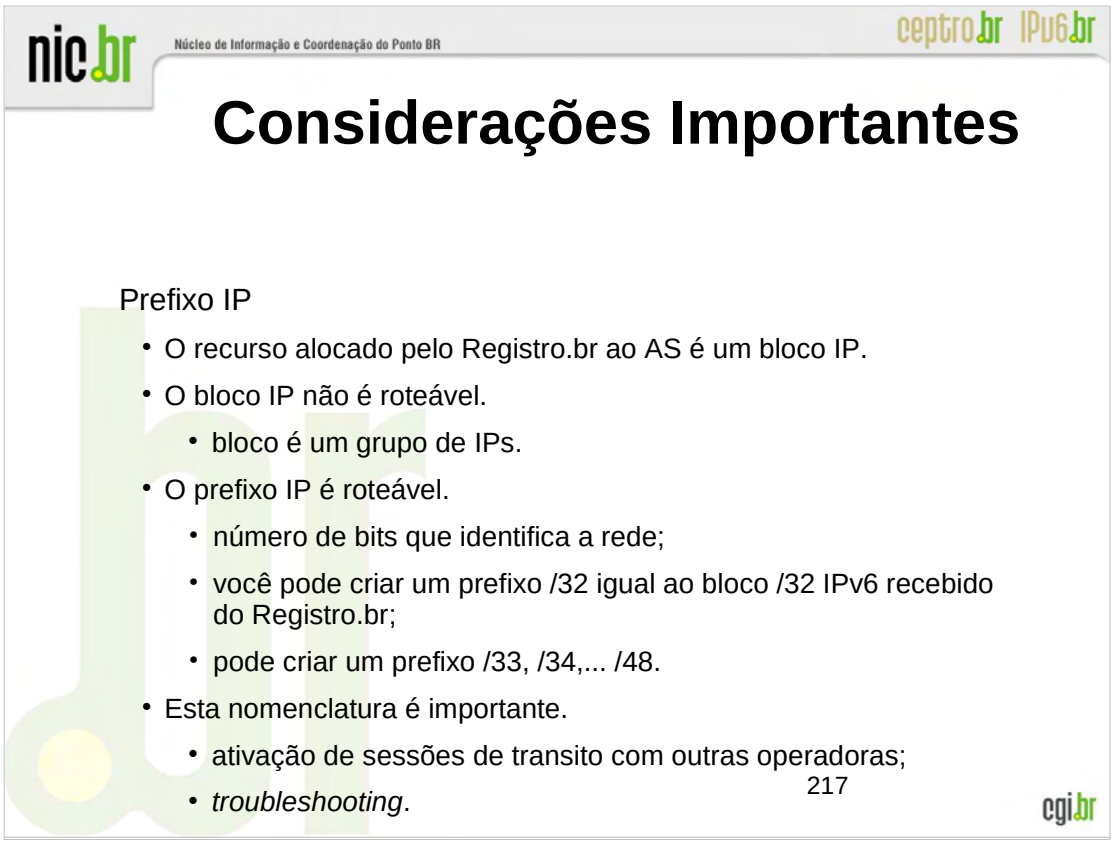
- *Endpoint ID* (EID): um identificador de 32 bits (para IPv4) ou 128 bits (para IPv6) usado nos campos de endereço de origem e destino do primeiro cabeçalho LISP (mais interno) de um pacote. O *host* obtém um EID de destino da mesma forma que obtém um endereço de destino hoje, por exemplo através de uma pesquisa de DNS. O EID de origem também é obtido através dos mecanismos já existentes, usados para definir o endereço local de um *host;*
- *Routing Locator* (RLOC): endereço IPv4 ou IPv6 de um ETR (*Egress Tunnel Router*). RLOCs são numerados a partir de um bloco topologicamente agregado, e são atribuídos a uma rede em cada ponto em que haja conexão com a Internet global;
- *Ingress Tunnel Router* (ITR): roteador de entrada do túnel que recebe um pacote IP (mais precisamente, um pacote IP que não contêm um cabeçalho LISP), trata o endereço de destino desse pacote como um EID e executa um mapeamento entre o EID e o RLOC. O ITR, em seguida, anexa um cabeçalho “IP externo” contendo um de seus RLOCs globalmente roteáveis, no campo de endereço de origem, e um RLOC, resultado do mapeamento, no campo de endereço de destino;
- *Egress Tunnel Router* (ETR): roteador de saída do túnel que recebe um pacote IP onde o endereço de destino do cabeçalho “IP externo” é um de seus RLOCs. O roteador retira o cabeçalho externo e encaminha o pacote com base no próximo cabeçalho IP encontrado.

Mais informações

• *Locator/ID Separation Protocol (LISP)* - http://www.ietf.org/id/draft-ietf-lisp-06.txt

•*LISP Networking: Topology, Tools, and Documents* - http://www.lisp4.net (apenas conexões IPv4)

•*LISP Networking: Topology, Tools, and Documents* - http://www.lisp6.net (apenas conexões IPv6)

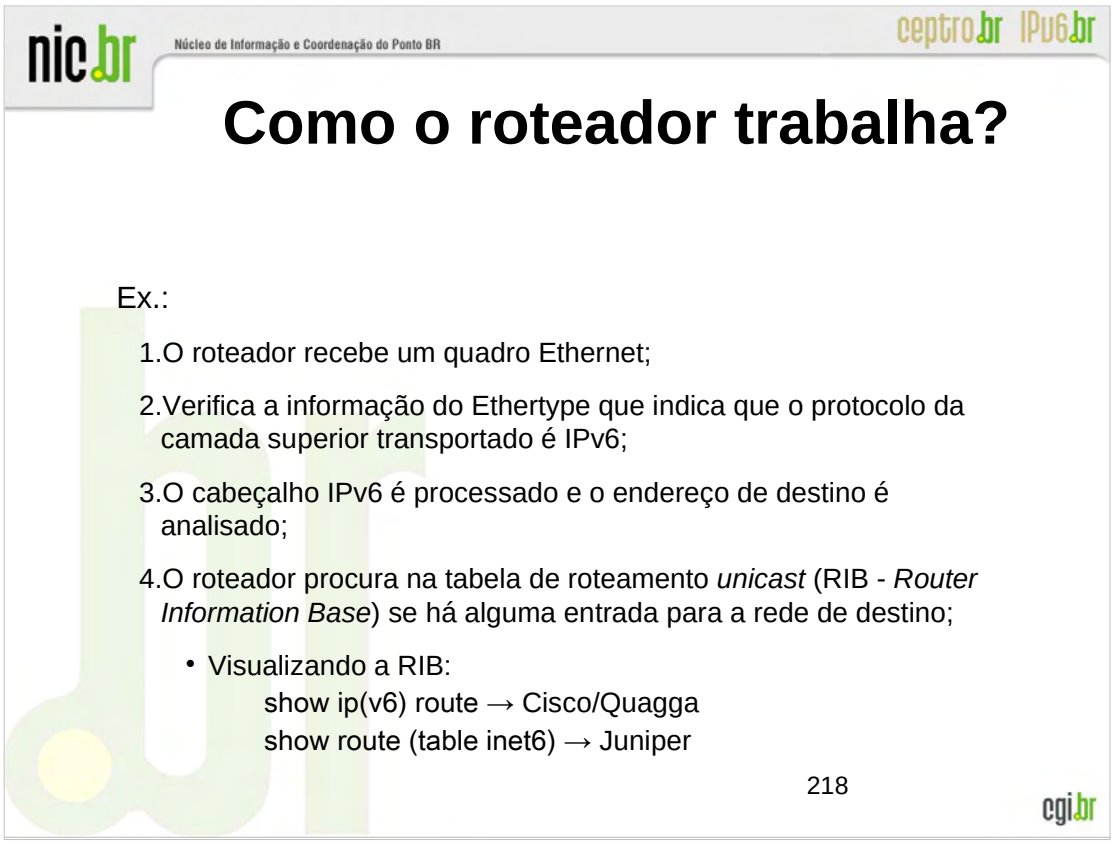


Um definição importante é a de prefixo IP.

O recurso alocado pelo Registro.br aos ASs é um bloco IP, que representa um grupo de endereços IP. O bloco não é um elemento roteável, o que é roteável é o prefixo. O que é possível, por exemplo, é criar um prefixo IPv6 /32 igual ao bloco /32 recebido do Registro.br e anunciar esse prefixo na tabela de rotas. Porém também é possível criar prefixos /33, /34, /48 etc a partir do bloco recebido.

O prefixo representa o número de bits de um endereço que identifica a rede.

Apesar de ser apenas mais uma nomenclatura, essa definição é importante na hora de enviar informações para ativar sessões de trânsito com outras operadoras e na detecção de problemas de conectividade.



Também é importante compreender o funcionamento básico de um roteador, de que forma ele processa os pacotes recebidos e efetua as decisões de encaminhamento. Analise o seguinte exemplo:

- O roteador recebe um quadro Ethernet através de sua interface de rede;
- Verifica a informação do Ethertype, que indica que o protocolo da camada superior transportado é IPv6;
- O cabeçalho IPv6 é processado e o endereço de destino é analisado;
- O roteador procura na tabela de roteamento *unicast* (RIB - *Router Information Base*) se há alguma entrada para a rede de destino;
-

  Visualizando a RIB IPv6:

  Cisco/Quagga → show ipv6 route
  Juniper → show route table inet6

  Visualizando a RIB IPv4:

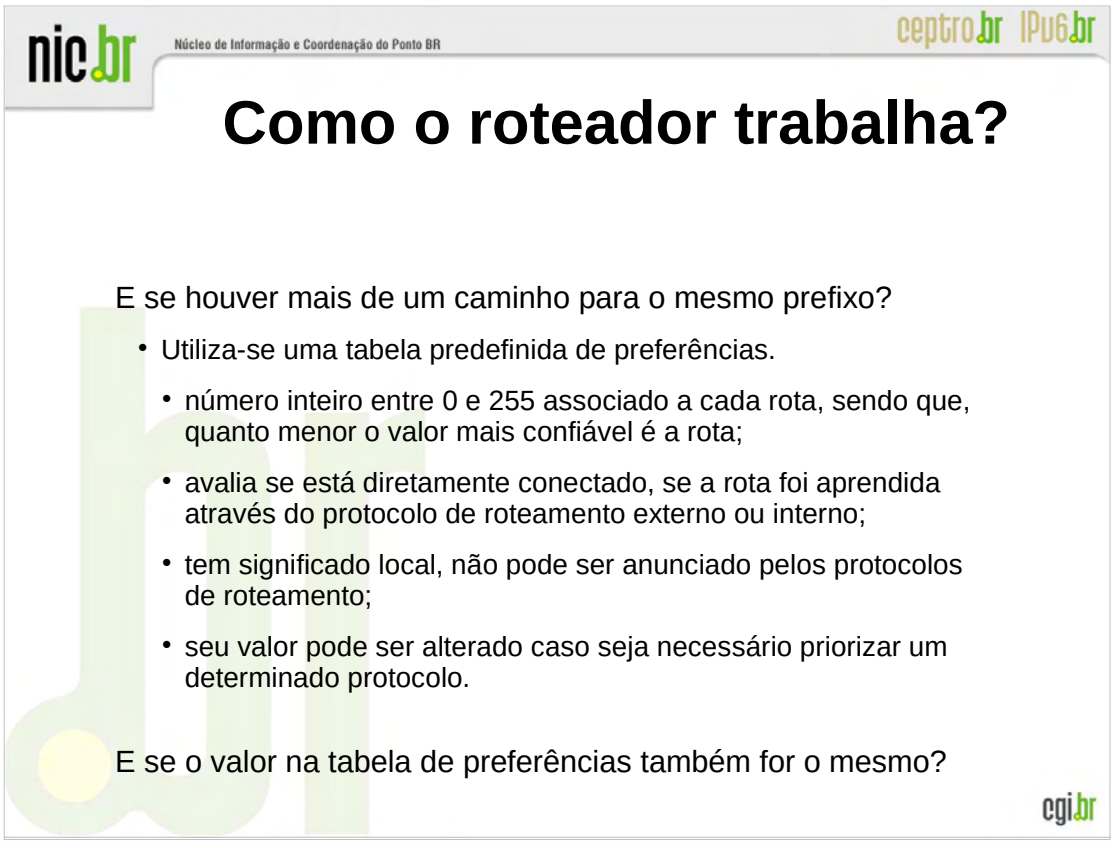
  Cisco/Quagga → show ip route
  Juniper → show route

Como o roteador trabalha?

5. *Longest Match* - procura a entrada mais específica. Ex.:
- O IP de destino é 2001:0DB8:0010:0010::0010
- O roteador possui as seguintes informações em sua tabela de rotas:
  - 2001:DB8::/32 via interface A
  - 2001:DB8::/40 via interface B
  - 2001:DB8:10::/48 via interface C
- Os três prefixos englobam o endereço de destino, porém o roteador sempre irá preferir o mais específico, neste caso, o /48;
- Qual é a entrada mais específica IPv4 e IPv6?

6. Uma vez identificado o prefixo mais específico, o roteador decrementa o *Hop-Limit*, monta o quadro Ethernet de acordo a interface, e envia o pacote.

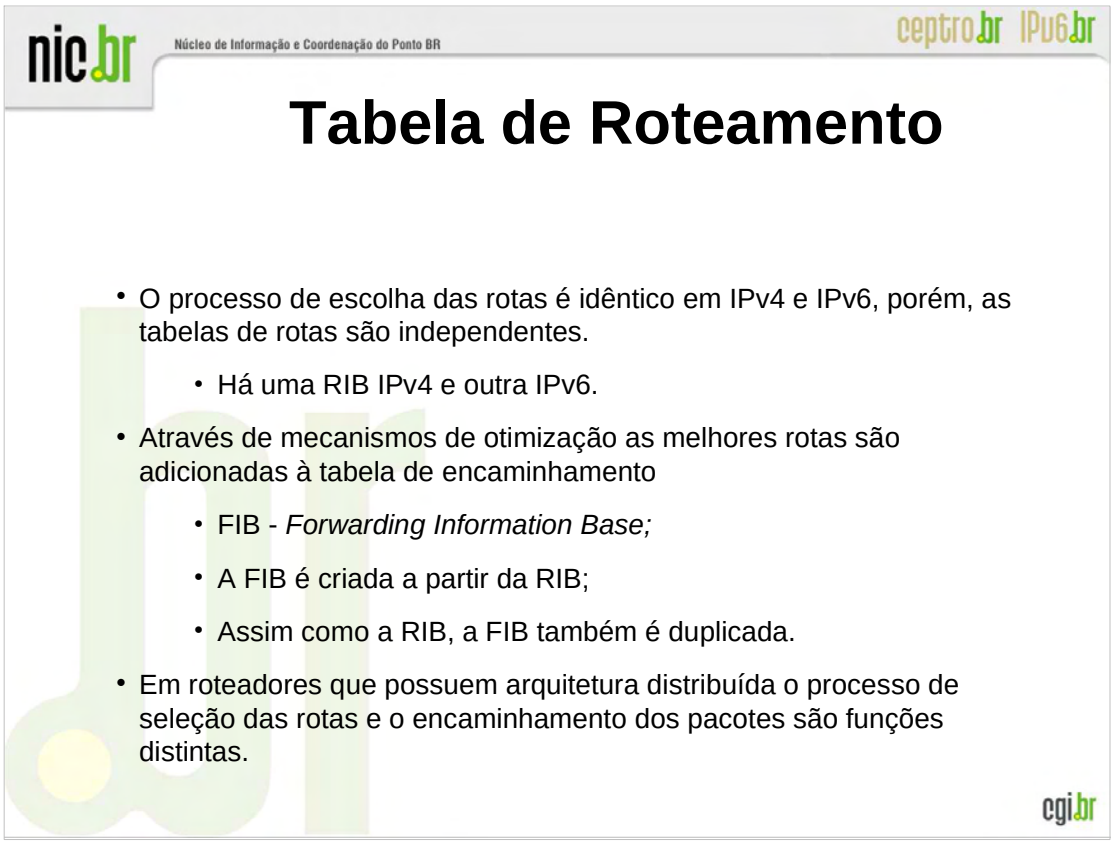
- *Longest Match* - procura a entrada mais específica. Ex.:
  - O IP de destino é 2001:0DB8:0010:0010::0010
  - O roteador possui as seguintes informações em sua tabela de rotas:
  - 2001:DB8::/32 via interface A
  - 2001:DB8::/40 via interface B
  - 2001:DB8:10::/48 via interface C
- Os três prefixos englobam o endereço de destino, porém o roteador sempre irá preferir o mais específico, neste caso, o /48;
- Uma vez identificado o prefixo mais específico, o roteador decrementa o *Hop-Limit*, monta o quadro Ethernet de acordo a interface, e envia o pacote.



Se o roteador localizar mais de uma caminho para o mesmo destino e com o mesmo valor de *longest match*, ele utilizará uma tabela predefinida de preferências (conceito de *Distância Administrativa* da Cisco).

Os valores desta tabela são números inteiros entre 0 e 255 associados a cada rota, sendo que quanto menor o valor mais confiável é a rota. Os valores são atribuídos avaliando se a rota está diretamente conectada, se foi aprendida através do protocolo de roteamento externo ou interno, etc. Estes valores têm significado apenas local, não podendo ser anunciados pelos protocolos de roteamento, e caso seja necessário, podem ser alterados para priorizar um determinado protocolo.

Caso também seja encontrado um mesmo valor na tabela de preferências, há equipamentos e implementações que, por padrão, realizam balanceamento de tráfego.



O processo de escolha das rotas é idêntico em IPv4 e IPv6, porém, as tabelas de rotas são independentes. Por exemplo: há uma RIB IPv4 e outra IPv6.

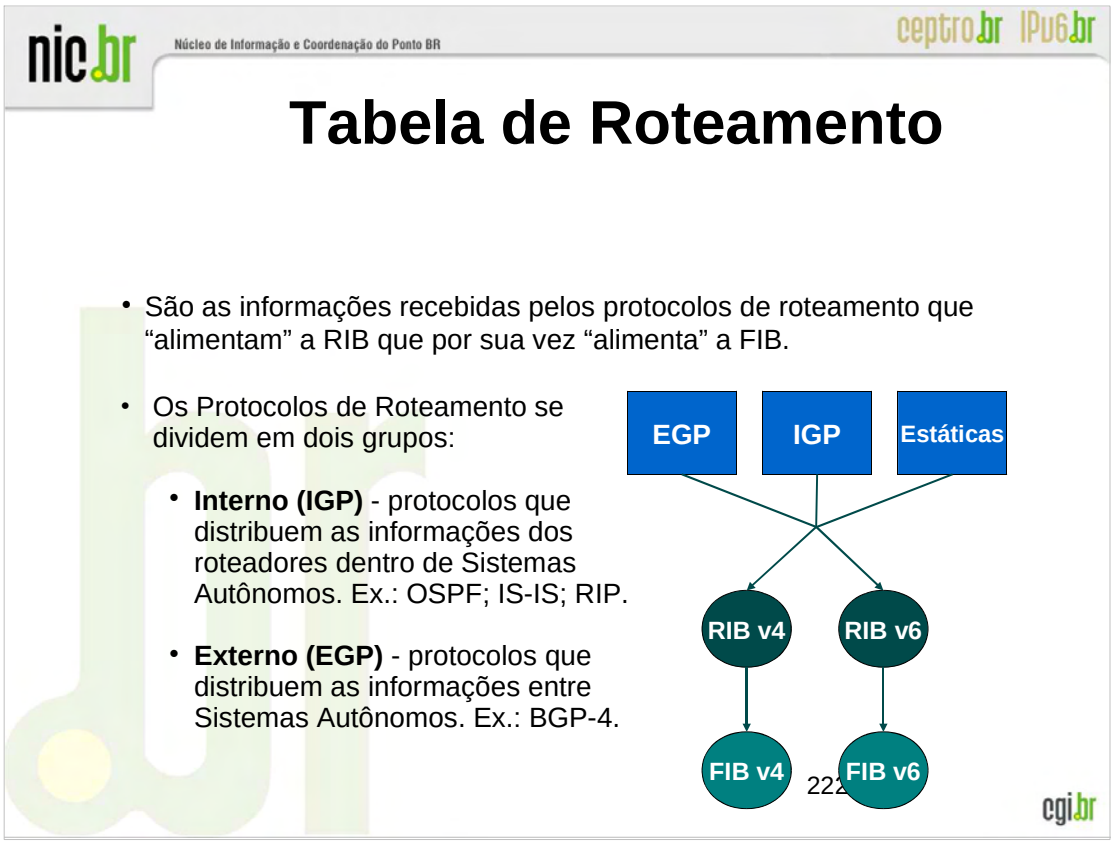
Para otimizar o envio dos pacotes, existem mecanismos que adicionam apenas as melhores rotas a uma outra tabela, a tabela de encaminhamento (FIB - *Forwarding Information Base*). Um exemplo deste mecanismo é o CEF (*Cisco Express Forwarding*) da Cisco.

A FIB é criada a partir da RIB, e assim como a RIB, ela também é duplicada se a rede estiver configurada com Pilha Dupla. Com isso, há mais informações para serem armazenadas e processadas.

Em roteadores que possuem arquitetura distribuída, o processo de seleção das rotas e o encaminhamento dos pacotes são funções distintas.

Ex.:

- Roteadores 7600 da Cisco, a RIB reside no módulo central de roteamento e a FIB nas placas das interfaces.
- Roteadores Juniper da série M, a *Router Enginie* é a responsável pela RIB, e a FIB também reside nas placas das interfaces (*Packet Forwarding Engine* - PFE).



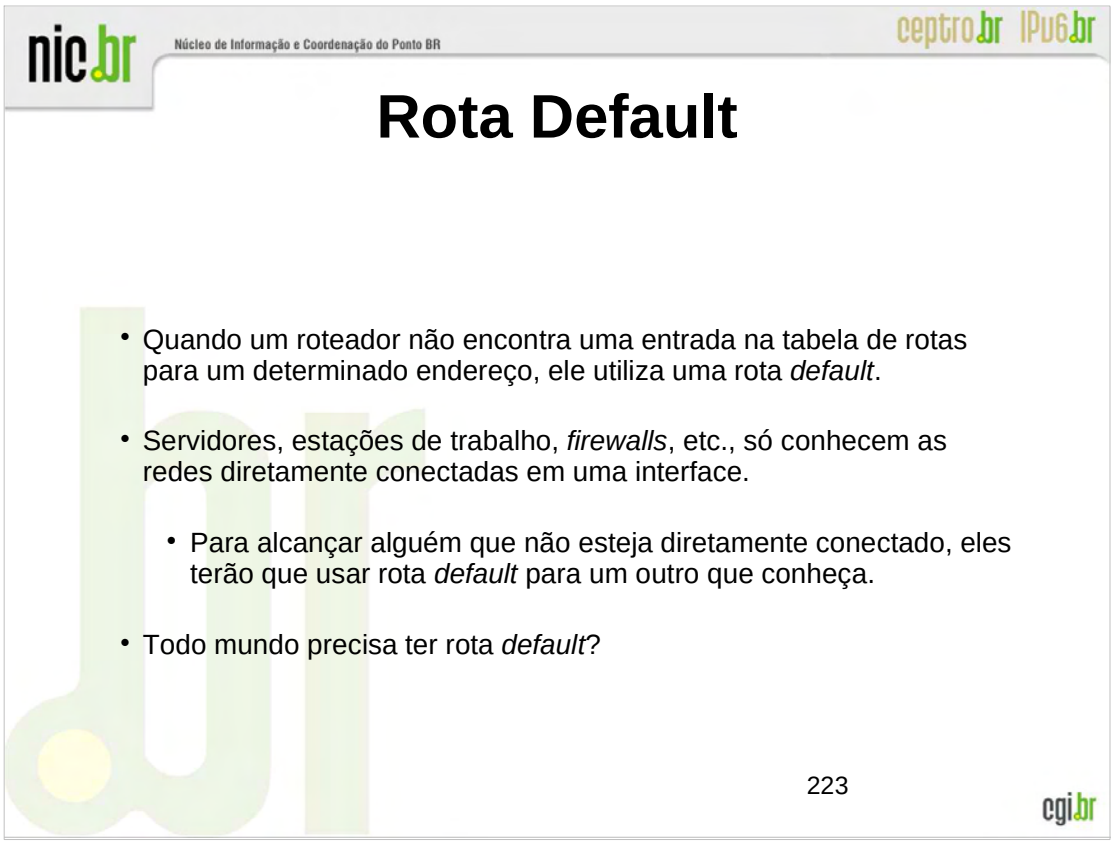
É o mecanismo de roteamento que possibilita o encaminhamento de pacotes de dados entre quaisquer dois dispositivos conectados à Internet.

Para atualizar as informações utilizadas pelos roteadores para encontrar o melhor caminho disponível no encaminhamento dos pacotes até o seu destino, utilizam-se os protocolos de roteamento. São as informações recebidas pelos protocolos de roteamento que "alimentam" a RIB, que por sua vez "alimenta" a FIB.

Estes protocolos se dividem em dois grupos:

- **Interno (IGP)** - protocolos que distribuem as informações dos roteadores dentro de Sistemas Autônomos. Como exemplo desses protocolos podemos citar: OSPF; IS-IS; e RIP.

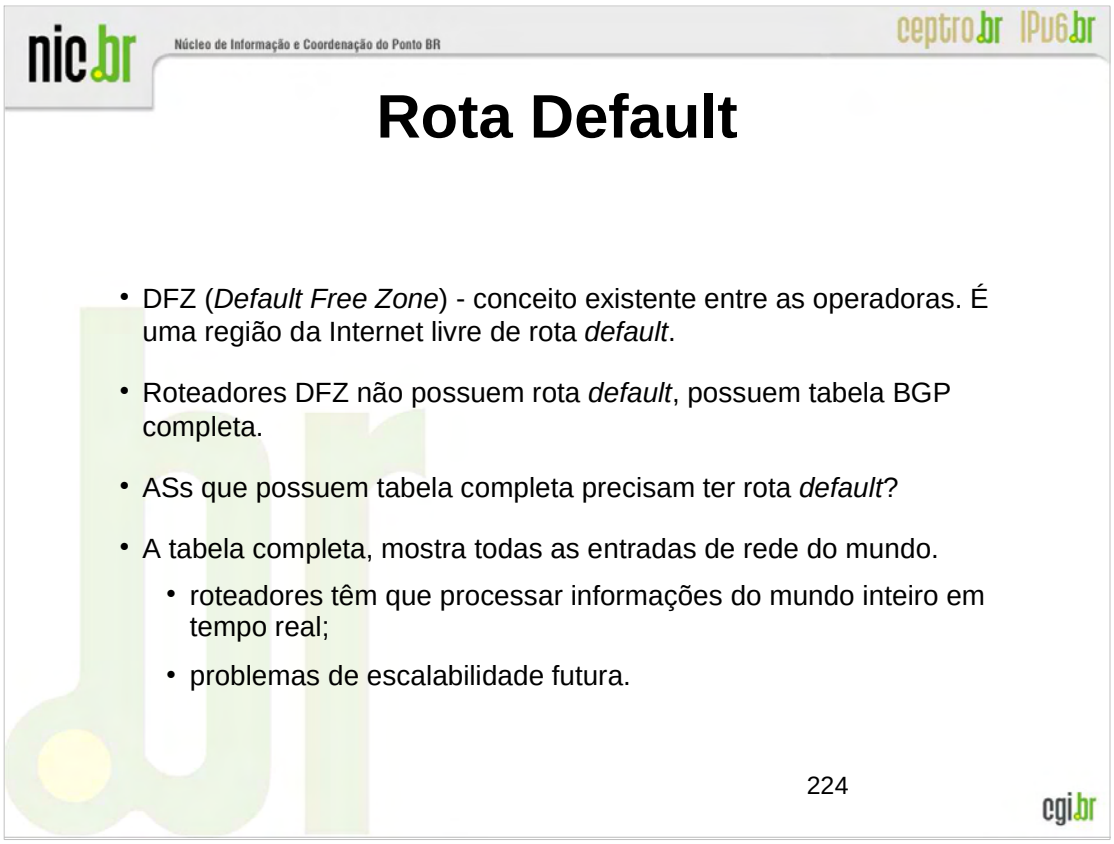
- **Externo (EGP)** - protocolos que distribuem as informações entre Sistemas Autônomos. Como exemplo podemos citar o BGP-4.



Caso o roteador receba um pacote cujo o endereço de destino não esteja explicitamente listado na tabela de rotas, ele utilizará sua rota *default.*

*S*ervidores e estações de trabalho, naturalmente precisam de uma rota *default*. Eles não são equipamentos de rede, eles só conhecem as redes diretamente conectadas em suas interfaces. Se eles quiserem alcançar alguém que não esteja diretamente conectado, eles terão que usar rota *default* para um outro equipamento que conheça.

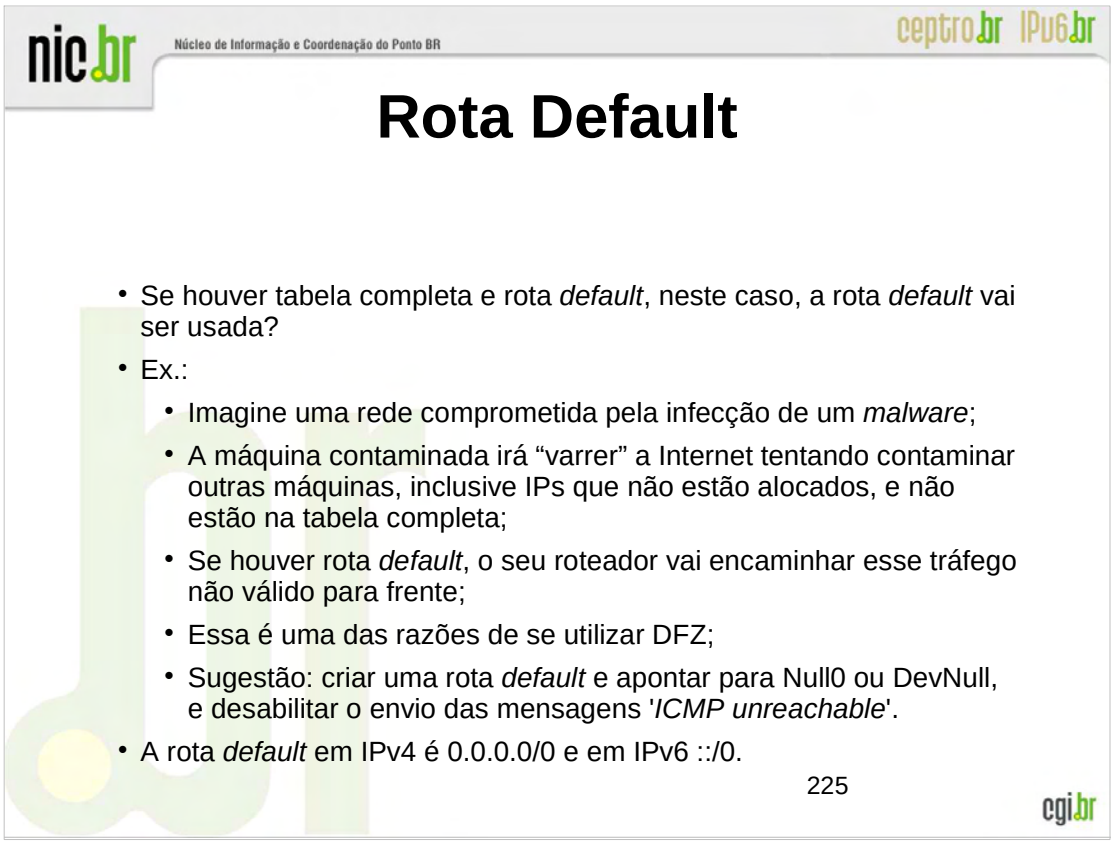
Ai surge a questão: todo mundo precisa ter rota *default*?



Existe um conceito entre as operadoras que delimita uma região da Internet livre de rota *default, a* DFZ (*Default Free Zone*).

Um AS que possua tabela completa não precisa ter rota *default*, pois a tabela completa, mostra todas as entradas de rede do mundo.

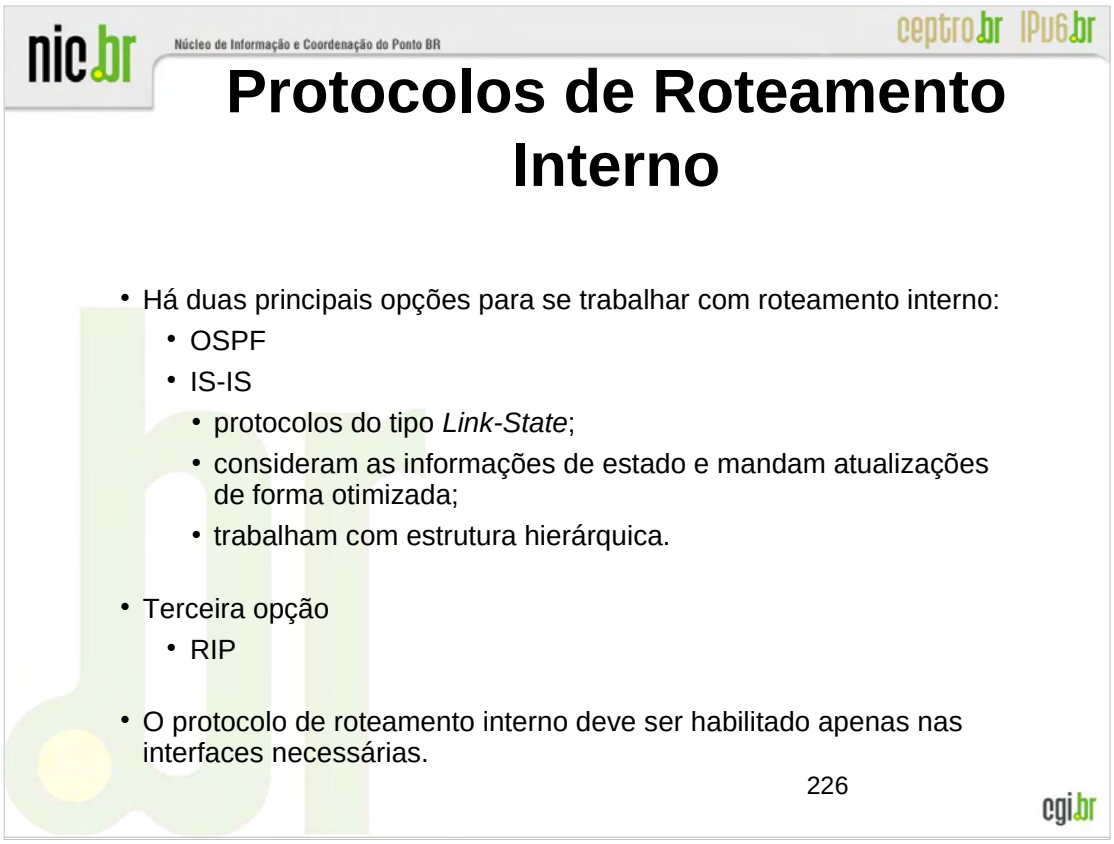
Esse modelo é bom e funcional, porém, isso pode acarretar alguns problemas. Os roteadores têm que processar informações do mundo inteiro em tempo real; e há também problemas de escalabilidade futura.



A utilização de rota *default* por roteadores que possuam tabela completa pode ocasionar alguns problemas.

Imagine a seguinte situação como exemplo: uma rede foi comprometida pela infecção de um *malware*. A máquina contaminada irá "varrer" a Internet tentando contaminar outras máquinas, inclusive IPs que não estão alocados, e não estão na tabela completa. Se houver rota *default*, o seu roteador vai encaminhar esse tráfego não válido para frente. Essa é uma das razões de se utilizar DFZ. Uma sugestão para solucionar esse problema é criar uma rota *default* e apontar para Null0 ou DevNull. Além disso, deve-se desabilitar o envio das mensagens 'ICMP unreachable', porque quando um roteador descarta um pacote, ele envia uma mensagem 'ICMP unreachable' avisando, porém, se o destino não é válido, não há necessidade de avisar a origem, isso apenas consome CPU desnecessariamente.

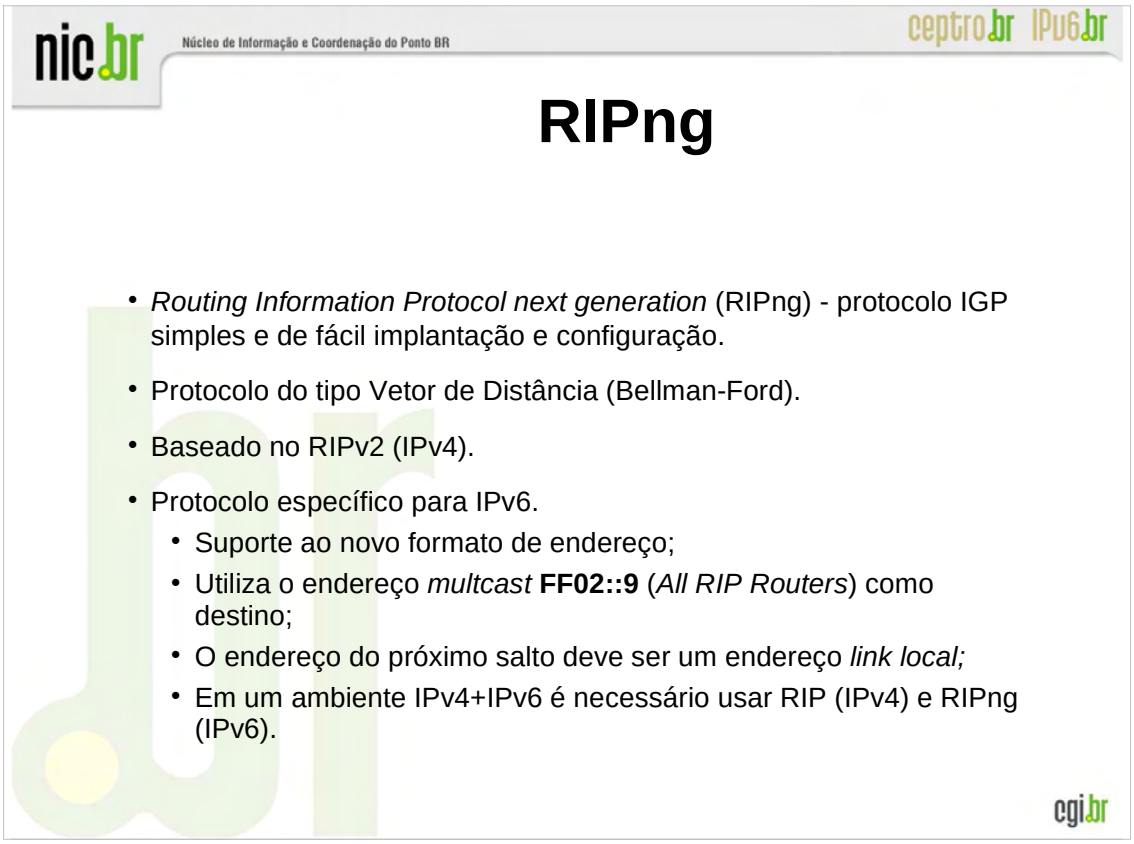
A rota *default* em IPv4 é 0.0.0.0/0 e em IPv6 ::/0.



Hoje há duas principais opções para se trabalhar com roteamento interno, o OSPF e o IS-IS. Esses dois protocolos são do tipo *Link-State*, isto é consideram as informações de estado do enlace, e mandam atualizações de forma otimizada, apenas quando há mudança de estado. Eles também permitem que se trabalhe com estrutura hierárquica, separando a rede por regiões. Isso é um ponto fundamental para IPv6.

Uma outra opção é o protocolo RIP (*Routing Information Protocol*). É um protocolo do tipo Vetor de Distância (Bellman-Ford), de fácil implementação e de funcionamento simples, porém apresenta algumas limitações como o fato de enviar sua tabela de estados periodicamente, independente de mudanças ou não na rede.

É importante que o protocolo de roteamento interno seja habilitado apenas nas interfaces onde são necessárias. Embora pareça óbvio, há quem configure de forma errada fazendo com que os roteadores fiquem tentando estabelecer vizinhança com outros ASs.



Para tratar o roteamento interno IPv6 foi definida uma nova versão do protocolo RIP, o *Routing Information Protocol next generation* (RIPng). Esta versão foi baseada no RIPv2 utilizado em redes IPv4, porém, ela é específica para redes IPv6.

Como mudanças principais destaca-se:

- Suporte ao novo formato de endereço;
- Utiliza o endereço *multcast* FF02::9 (*All RIP Routers*) como destino;
- O endereço do próximo salto deve ser um endereço *link local*.

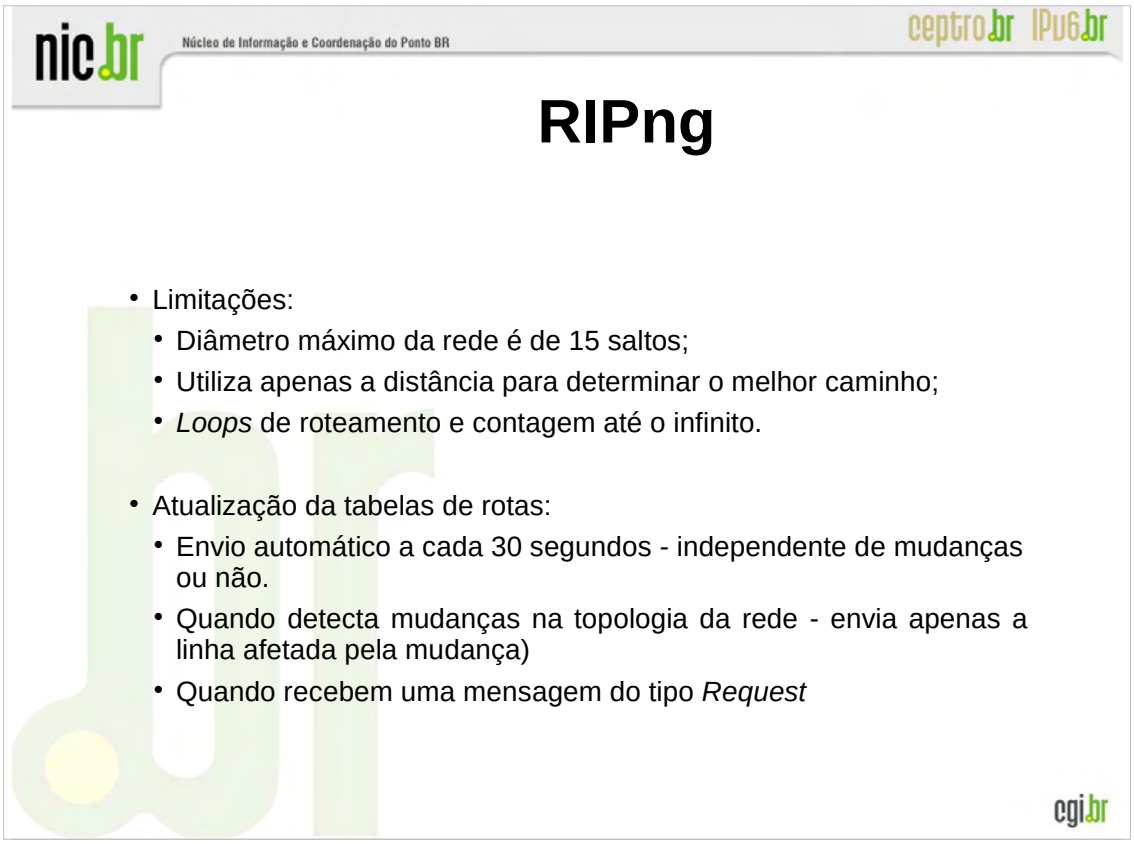
Em um ambiente com Pilha Dupla (IPv4+IPv6) é necessário usar uma instância do RIP para IPv4 e uma do RIPng para o roteamento IPv6.

Apesar de ser um protocolo novo, o RIPng ainda apresenta as mesma limitações das versões anteriores utilizadas com IPv4, como:

- Diâmetro máximo da rede é de 15 saltos;
- Utiliza apenas a distância para determinar o melhor caminho;
- *Loops* de roteamento e contagem até o infinito.

Mais informações:

- RFC 2080 - *RIPng for IPv6*



As informações presentes na tabela de rotas são:

- Prefixo do destino
- Métrica
- Próximo salto
- Identificação da rota (*route tag*)
- Mudança de rota
- Tempo até a rota expirar (padrão 180 segundos)
- Tempo até a “coleta de lixo” (*garbage collection*) (padrão 120 segundos)

A atualização da tabelas de rotas pode ocorrer de três formas: através do envio automático dos dados a cada 30 segundos; quando é detectada alguma mudanças na topologia da rede, enviando apenas a linha afetada pela mudança; e quando é recebida uma mensagem do tipo *Request.*

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

RIPng

- Mensagens *Request* e *Response*

8 bits	8 bits	16 bits
Comando	Versão	Reservado
Entrada 1 da tabela de rotas (RTE)		
....		
Entrada n da tabela de rotas		

- RTE
  - Prefixo IPv6 (128 bits)
  - Identificação da rota (16 bits)
  - Tamanho do prefixo (8 bits)
  - Métrica (8 bits)
- Diferente do RIPv2, o endereço do próximo salto aparece apenas uma vez, seguido de todas as entradas que devem utilizá-lo.

cgi.br

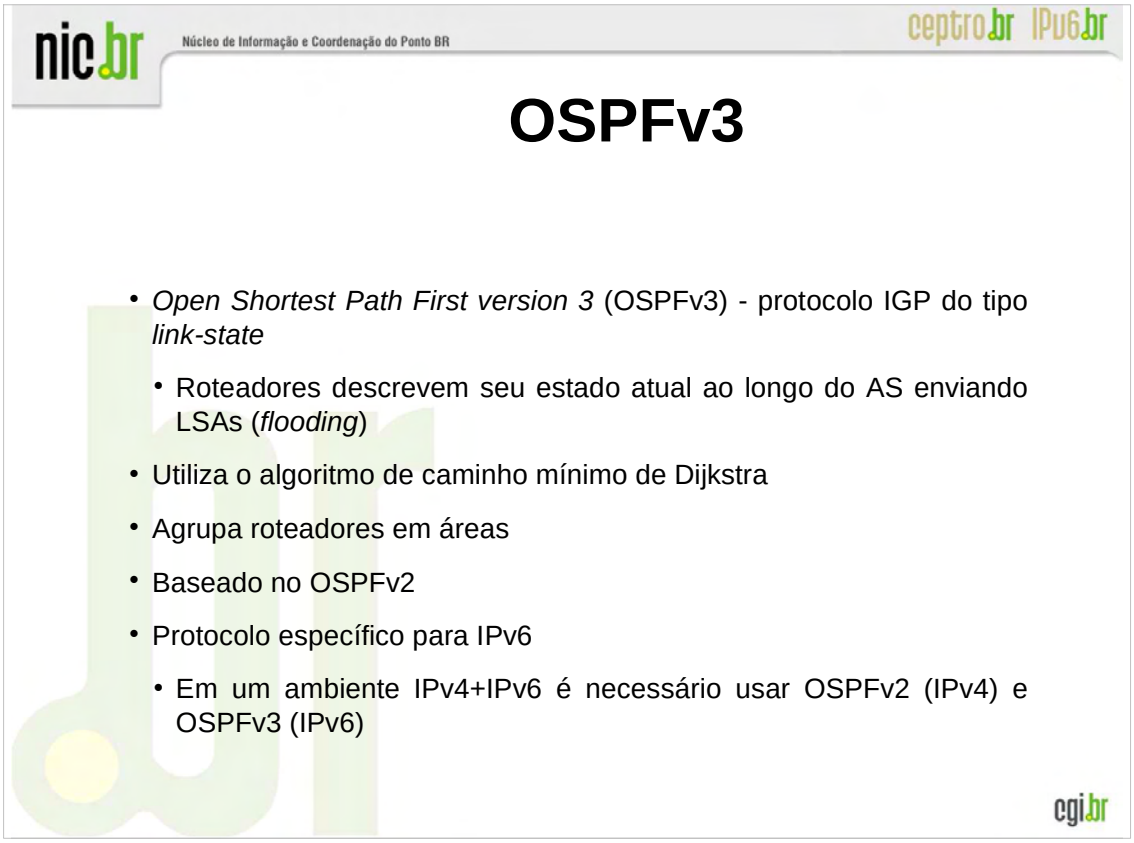
O cabeçalho das mensagens do RIPng é bem simples, composto pelos seguintes campos:

- Comando (*command*) – indica se a mensagem é do tipo *Requet* ou *Response*;
- Versão (*version*) – indica a versão do protocolo que atualmente é 1.

Esses campos são seguidos das entradas da tabela de rotas (*Route Table Entry* – RTE):

- Prefixo IPv6 (128 bits);
- Identificação da rota (16 bits);
- Tamanho do prefixo (8 bits);
- Métrica (8 bits).

Diferente do RIPv2, o endereço do próximo salto aparece apenas uma vez, seguido de todas as entradas que devem utiliza-lo.

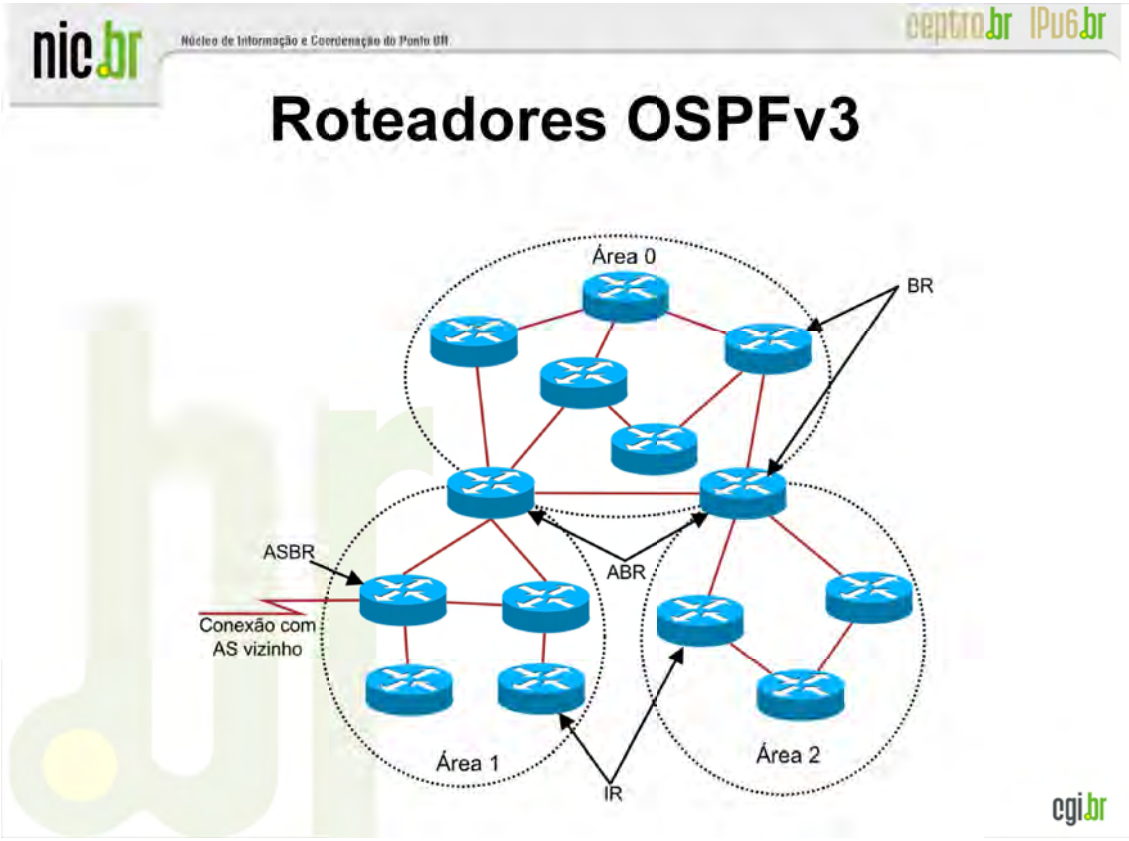


O OSPF é um protocolo do tipo *link-state* onde, através do processo de *flooding* (inundação), os roteadores enviam *Link State Advertisements* (LSA) descrevendo seu estado atual ao longo do AS. O *flooding* consiste no envio de um LSA por todas as interfaces de saída do roteador, de modo que todos os roteadores que receberem um LSA também o enviam por todas as suas interfaces. Com isso, o conjunto dos LSAs de todos os roteadores forma um banco de dados de estado do enlace, onde cada roteador participante do AS possui um banco de dados idêntico. Com as informações desse banco, o roteador, através do protocolo OSPF, constrói um mapa da rede que será utilizado para determinar uma árvore de caminhos mais curtos dentro de toda a sub-rede, tendo o próprio nó como raiz. Ele utiliza o algorítimo de Dijkstra para a escolha do melhor caminho e permite agrupar os roteadores em áreas.

O OSPF pode ser configurado para trabalhar de forma hierárquica, dividindo os roteadores de um AS em diversas áreas. A cada uma dessas áreas é atribuído um identificador único (Area-ID) de 32 bits e todos os roteadores de uma mesma área mantém um banco de dados de estado separado, de modo que a topologia de uma área é desconhecida fora dela, reduzindo a quantidade de tráfego de roteamento entre as partes do AS. A área de *backbone* é a responsável por distribuir as informações de roteamento entre as áreas *nonbackbone* e é identificada pelo ID 0 (ou 0.0.0.0). Em ASs onde não há essas divisões a área de *backbone* geralmente é a única a ser configurada.

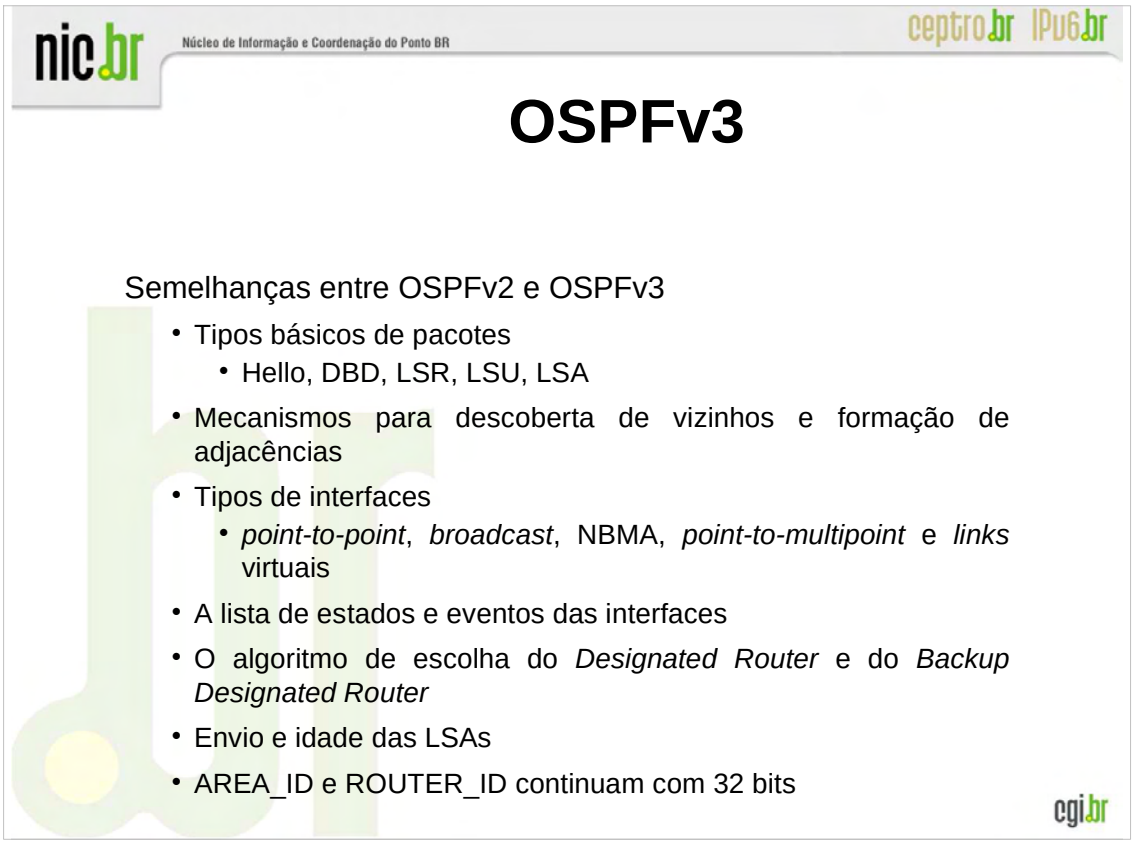
O OSPFv3 é um protocolo específico para IPv6, apesar de ter sido baseado na versão do OSPFv2, utilizada em redes IPv4. Deste modo, em uma rede com Pilha Dupla, é necessário utilizar OSPFv2 para o roteamento IPv4 e OSPFv3 para realizar o roteamento IPv6.

Mais informações:
- RFC 5340 - *OSPF for IPv6*



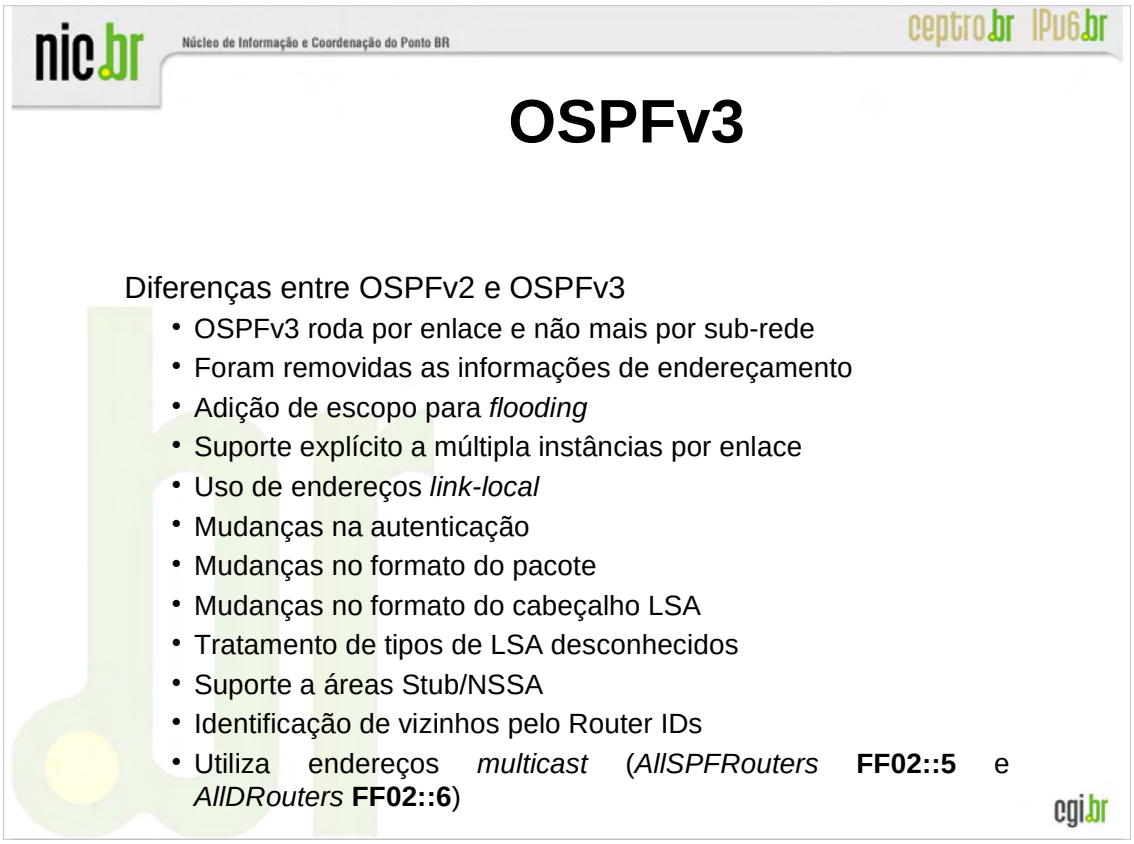
Os roteadores OSPF podem ser classificados como:

- *Internal Router* (IR) – roteadores que se relacionam apenas com vizinhos OSPF de uma mesma área;
- *Area Border Router* (ABR) – roteadores que conectam uma ou mais áreas ao *backbone*. Eles possuem múltiplas cópias dos bancos de dados de estado, uma para cada área, e são responsáveis por condensar as informações destas áreas e enviá-las ao *backbone*;
- *Backbone Router* (BR) – roteadores pertencentes a área *backbone*. Um ABR é sempre um BR, desde que todas suas áreas estejam diretamente conectadas ao *backbone* ou conectadas via *virtual link* - túnel que conecta uma área ao *backbone* passando através de outra área; e
- *Autonomous System Border Router* (ASBR) – roteadores que trocam informações de roteamento com roteadores de outro AS e distribuem as rotas recebidas ao longo do seu próprio AS.



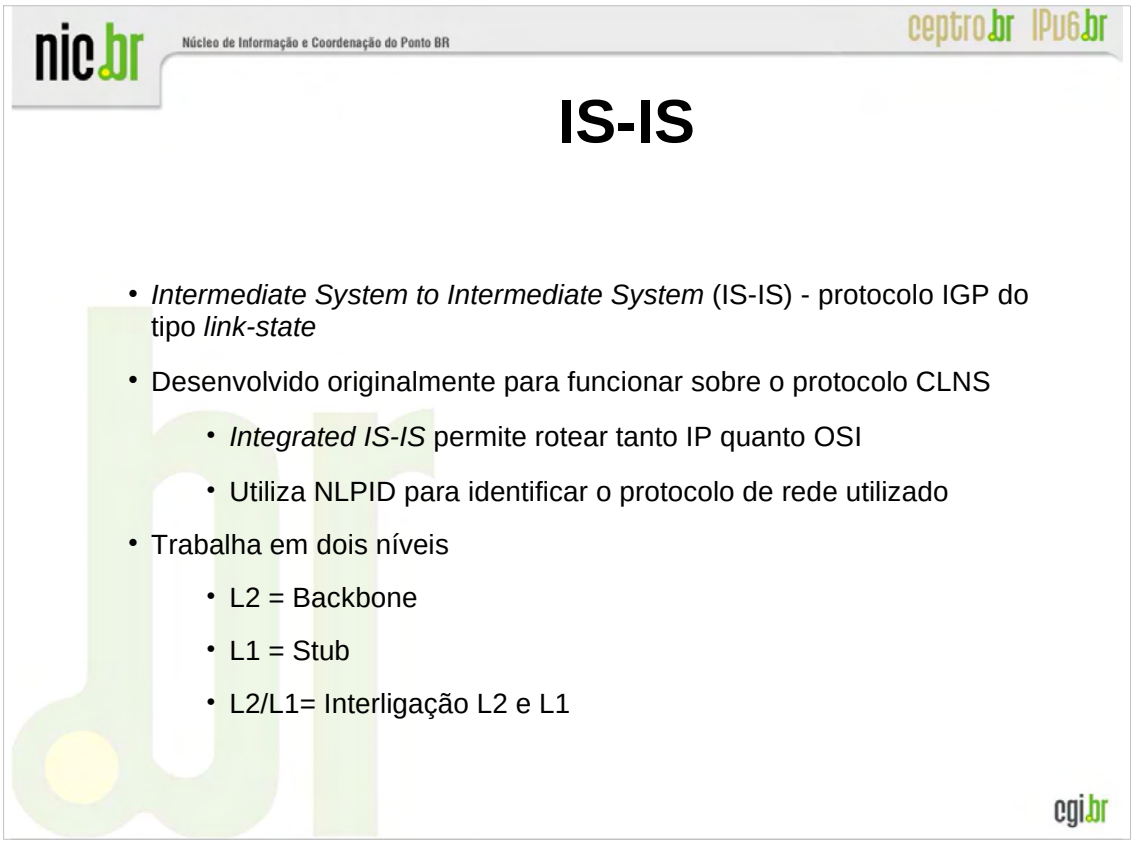
Algumas características do OSPFv2 ainda são encontradas no OSPFv3:

- Tipos básicos de pacotes
  - Hello, DBD, LSR, LSU, LSA
- Mecanismos para descoberta de vizinhos e formação de adjacências
- Tipos de interfaces
  - *point-to-point*, *broadcast*, NBMA, *point-to-multipoint* e *links* virtuais
- A lista de estados e eventos das interfaces
- O algoritmo de escolha do *Designated Router* e do *Backup Designated Router*
- Envio e idade das LSAs
- AREA_ID e ROUTER_ID continuam com 32 bits



Entras as principais diferenças entre o OSPFv2 e o OSPFv3 destacam-se:

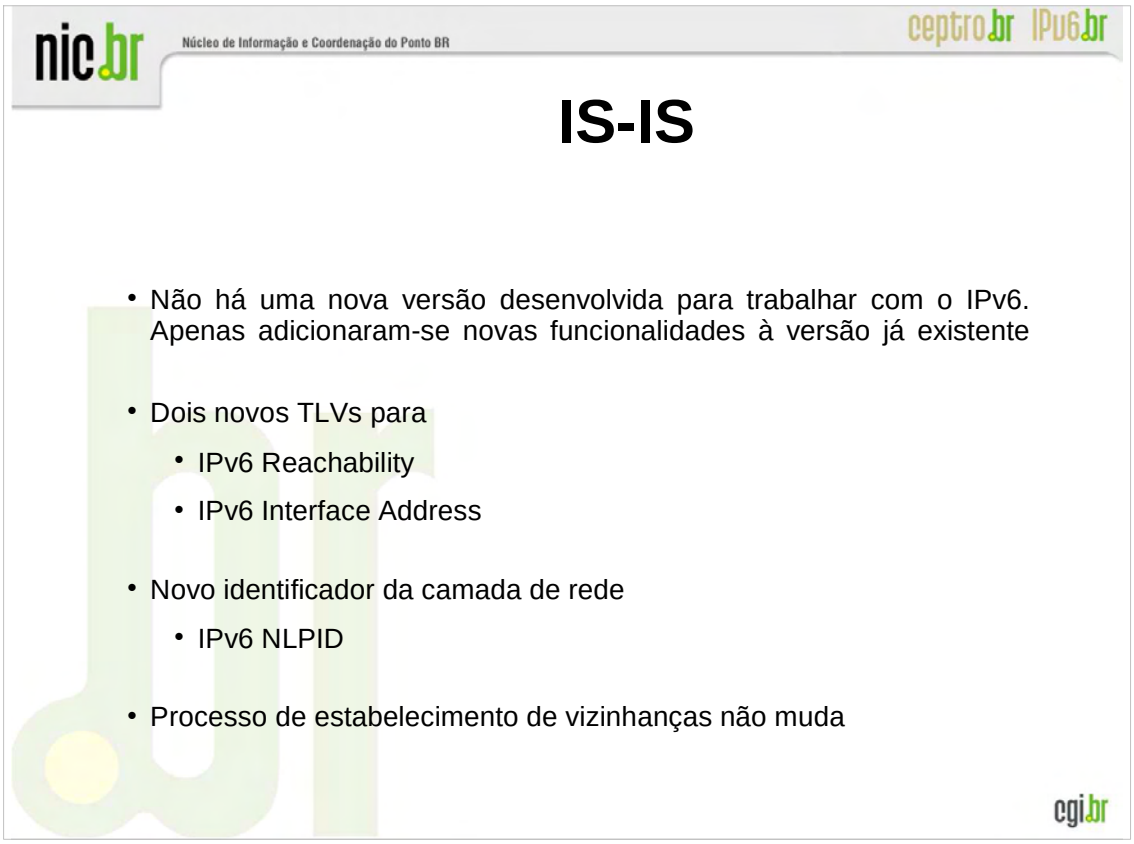
- OSPFv3 roda por enlace e não mais por sub-rede
- Foram removidas as informações de endereçamento
- Adição de escopo para *flooding*
- Suporte explícito a múltipla instâncias por enlace
- Uso de endereços *link-local*
- Mudanças na autenticação
- Mudanças no formato do pacote
- Mudanças no formato do cabeçalho LSA
- Tratamento de tipos de LSA desconhecidos
- Suporte a áreas Stub/NSSA
- Identificação de vizinhos pelo Router IDs
- Utiliza endereços *multicast* (*AllSPFRouters* **FF02::5** e *AllDRouters* **FF02::6**)



Assim como o OSFP, o *Intermediate System to Intermediate System* (IS-IS) é um protocolo IGP do tipo *link-state*, que utiliza o algorítimo de Dijkistra para calcular as rotas.

O IS-IS foi desenvolvido originalmente para funcionar sobre o protocolo CLNS, mas a versão *Integrated IS-IS* permite rotear tanto pacotes de rede IP quanto OSI. Para isso, utiliza-se um identificador de protocolo, o NLPID, para informar qual protocolo de rede está sendo utilizado.

Assim como o OSPF, o IS-IS também permite trabalhar a rede de forma hierárquica, atuando com os roteadores em dois níveis, o L1 (Stub) e o L2 (Backbone), além de roteadores que integram essas áreas, os L2/L1.



Para tratar o roteamento IPv6, não foi definida uma nova versão do protocolo IS-IS, apenas foram adicionadas novas funcionalidades à versão já existente.

Duas novas TLVs (*Type-Length-Values*) foram adicionadas:

- **IPv6 Reachability** (type 236) – carrega as informações das rede acessíveis;
- **IPv6 Interface Address** (type 232) – traz os endereços IP da interface que está transmitindo o pacote.

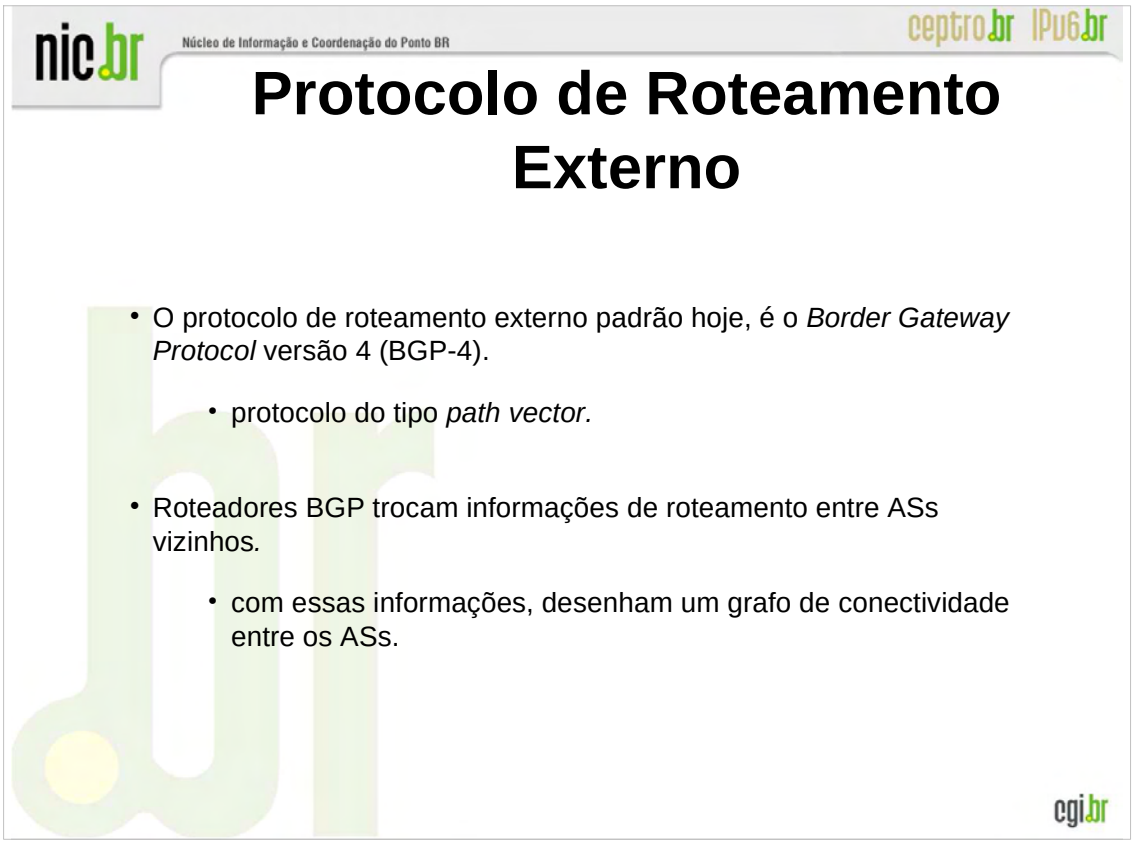
Também foi adicionado um novo identificador da Camada de Rede

- **IPv6 NLPID** – seu valor é 142.

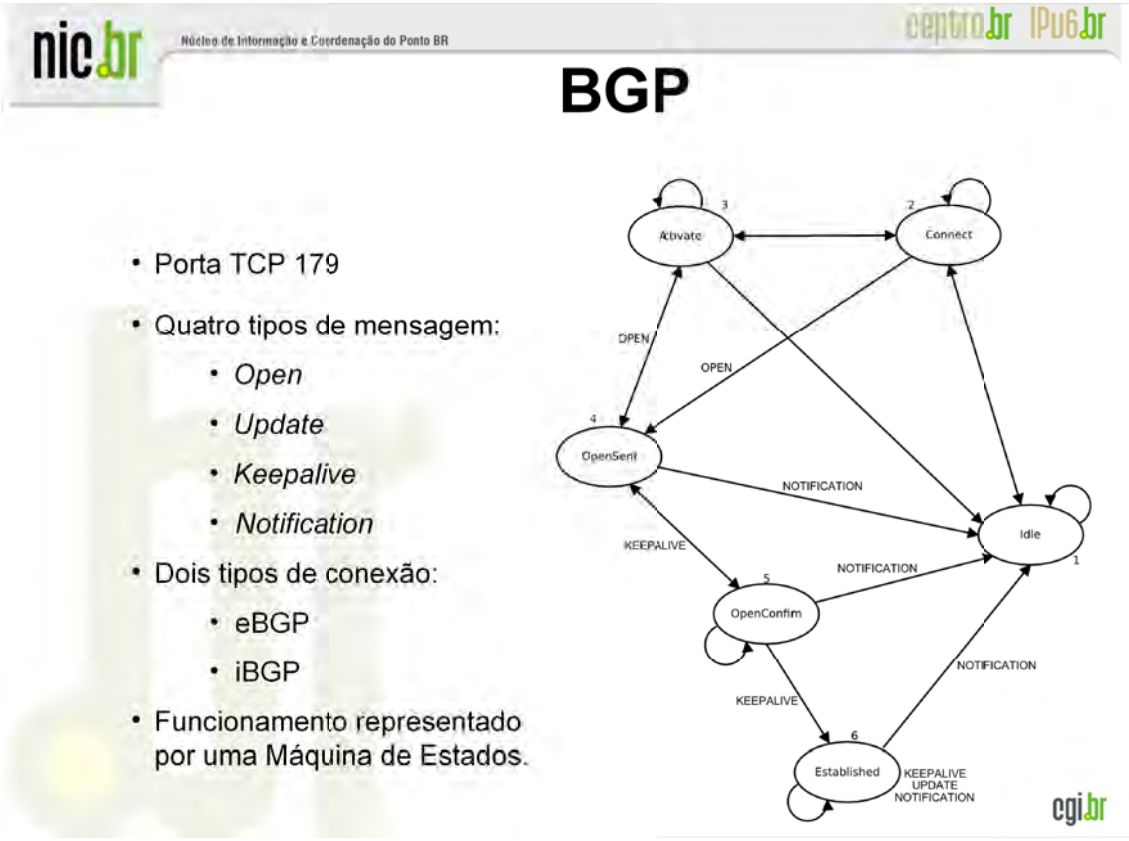
O processo de estabelecimento de vizinhanças não muda.

Mais informações:

- RFC 1195 - *Use of OSI IS-IS for Routing in TCP/IP and Dual Environments*
- RFC 5308 - *Routing IPv6 with IS-IS*



O protocolo de roteamento externo padrão hoje, é o *Border Gateway Protocol* versão 4 (BGP-4). É um protocolo do tipo *path vector*, onde roteadores BGP trocam informações de roteamento entre ASs vizinhos desenhando um grafo de conectividade entre os ASs.



O BGP é um protocolo extremamente simples baseado em sessões TCP ouvindo na porta 179.

Quatro tipos de mensagens BGP são utilizadas para troca de informações e manter o estado da conexão TCP:

- *Open* - enviada pelos dois vizinhos logo após o estabelecimento da conexão TCP, ela carrega as informações necessárias para o estabelecimento da sessão BGP, como ASN, versão do BGP, etc;
- *Update* – usada para transferir as informações de roteamento entre os vizinhos BGP, que serão utilizadas para construir o grafo que descreve o relacionamento entre vários ASs;
- *Keepalive* – são enviadas frequentemente para evitar que a conexão TCP expire;
- *Notification* – é enviada quando um erro é detectado, fechando a conexão BGP imediatamente após o seu envio.

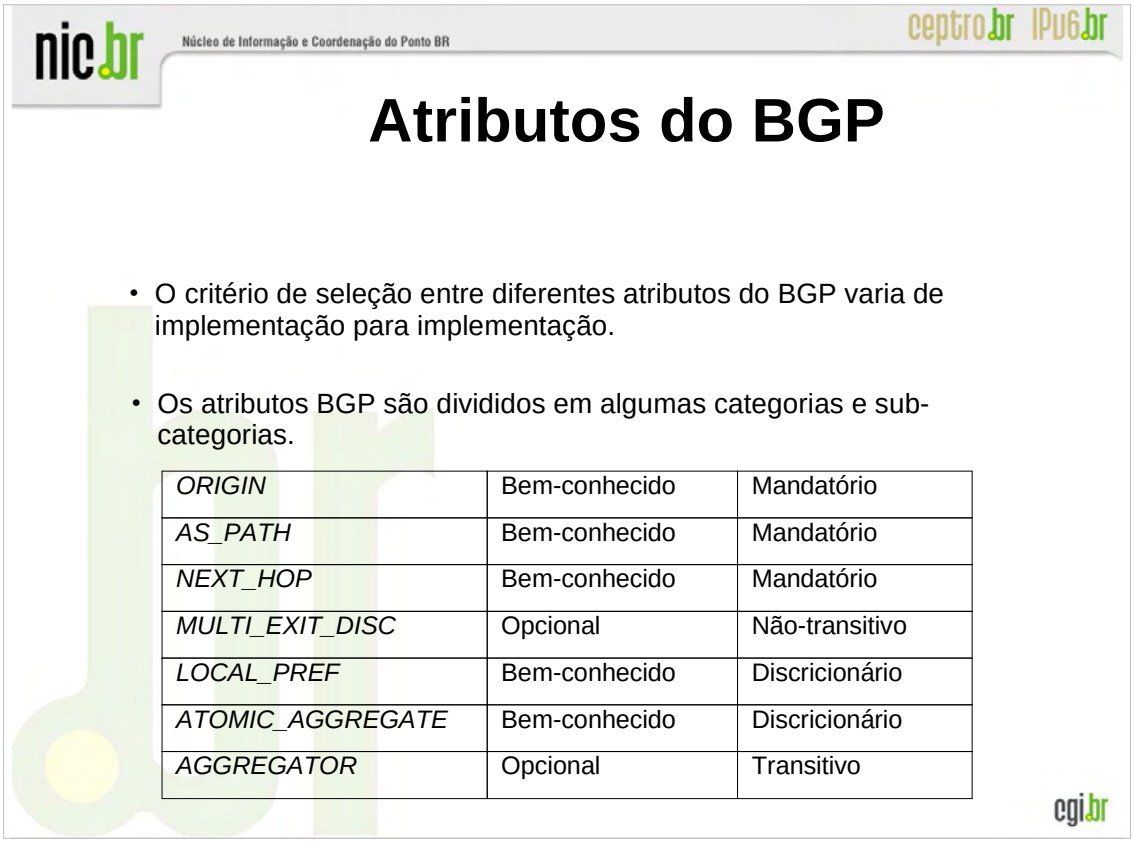
Você pode estabelecer dois tipos de conexão BGP:

- externa (eBGP) – conexão entre dois ASs vizinhos;
- interna (iBGP) – conexão entre roteadores dentro de um mesmo AS. O estabelecimento do iBGP é muito importante para se manter uma visão consistente das rotas externas em todos os roteadores de um AS.

O funcionamento do BGP pode ser representado por uma Máquina de Estados Finitos. Para quem não está familiarizado com o BGP, ao verificar que o estado de uma conexão está “Active” ou “Established”, pode ter a falsa impressão de que a conexão está “ativa” ou “estabelecida”, mas em geral, em BGP quando há “palavras” representando o estado, significa que a sessão BGP ainda não está ok. A sessão só estará efetivamente estabelecida quando for observada a quantidade de prefixos que se está recebendo do vizinho. Esse nomes representam estados intermediários da sessão BGP. Identificar esses estados ajuda na análise e resolução de problemas.

Mais informações:

RFC 4271 - *A Border Gateway Protocol 4* (BGP-4)
RFC 4760 - *Multiprotocol Extensions for* BGP-4



Apesar da RFC do BGP recomendar alguns pontos, o critério de seleção entre diferentes atributos do BGP pode variar de implementação para implementação. No entanto, a maior parte das implementações segue os mesmos padrões.

Os atributos BGP podem ser divididos em duas grandes categorias:

- **Bem-Conhecidos** (*Well-know*) – são atributos definidos na especificação original do protocolo BGP. Eles se sub-dividem em outras duas categorias:
  - **Mandatórios** (*Mandatory*) - devem estar sempre presentes nas mensagens do tipo UPDATE e devem ser obrigatoriamente reconhecidos em todas as implementações do protocolo;
  - **Descricionário** (*Discretionary*) - não precisam estar obrigatoriamente presente em todas as mensagens UPDATE.
- **Opcionais** (*Optional*) - não são obrigatoriamente suportados por todas as implementações de BGP. Eles se sub-dividem em outras duas categorias:
  - **Transitivos** (Transitive) – devem ser repassados nas mensagens UPDATE;
  - **Não-Transitivo** (*Non-transitive*) – não deve ser repassado.

A RFC do BGP apresenta os seguintes atributos:

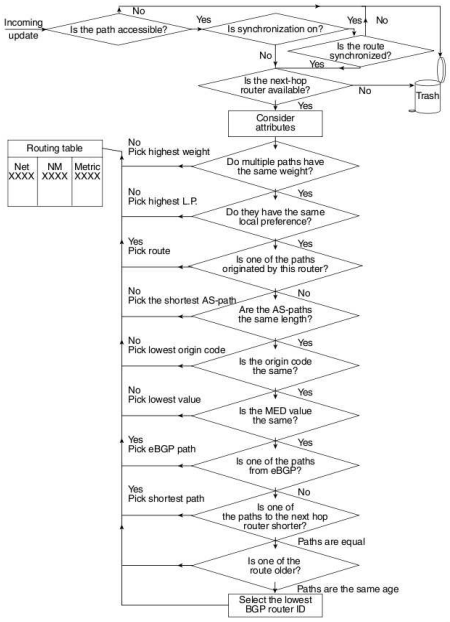
- *ORIGIN* – é Bem-Conhecido e Mandatório. Indica se o caminho foi aprendido via IGP ou EGP;
- *AS_PATH* - é Bem-Conhecido e Mandatório. Indica o caminho para se chegar a um destino, listando os ASN pelos quais se deve passar;
- *NEXT_HOP* – é Bem-Conhecido e Mandatório. Indica o endereço IP da interface do próximo roteador;
- *MULTI_EXIT_DISC* – é Opcional e Não-Transitivo. Indica para os vizinhos BGP externos qual o melhor caminho para uma determinada rota do próprio AS, influenciando-os, assim, em relação a qual caminho deve ser seguido no caso do AS possuir diversos pontos de entrada;
- *LOCAL_PREF* – é Bem-Conhecido e Discricionário. Indica um caminho preferencial de saída para uma determinada rota, destinada a uma rede externa ao AS;
- *ATOMIC_AGGREGATE* – é Bem-Conhecido e Discricionário. Indica se caminhos mais específicos foram agregados em menos específicos.
- *AGGREGATOR* - é Opcional e Transitivo. Indica o ASN do último roteador que formou uma rota agregada, seguido de seu próprio ASN e endereço IP.

nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

Atributos do BGP

- Os atributos são considerados se o caminho for conhecido, se houver conectividade, se for acessível e se o *next hop* estiver disponível.
- A forma de seleção pode variar de acordo com a implementação.
- O *LOCAL_PREFERENCE* é um atributo extremamente poderoso para influenciar o tráfego de saída.
- O valor do *LOCAL_PREFERENCE* é valido para todo o AS.

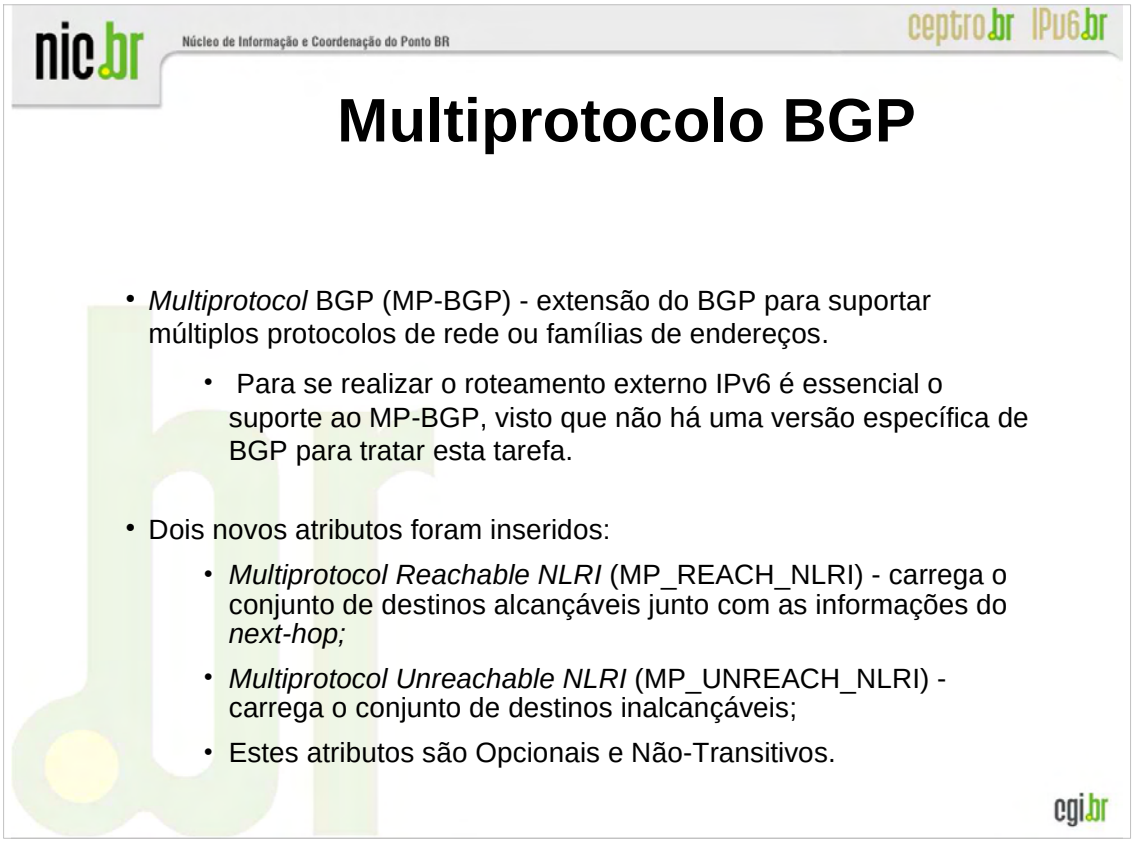


Seleção do caminho no BGP (CISCO).

cgi.br

Na decisão pela melhor rota, os atributos são considerados se o caminho for conhecido, se houver conectividade, se for acessível e se o *next hop* estiver disponível. Porém, a forma de seleção pode variar de acordo com a implementação.

Um atributo que merece destaque é o *LOCAL_PREFERENCE*. Ele é um atributo extremamente poderoso para influenciar o tráfego de saída. Seu valor é valido para todo o AS, sendo repassado apenas nas sessões iBGP.



Foram definidas extensões para o BGP-4 com o intuito de habilitá-lo a carregar informações de roteamento de múltiplos protocolo da Camada de Rede (ex., IPv6, IPX, L3VPN, etc.). Para se realizar o roteamento externo IPv6 é essencial o suporte ao MP-BGP, visto que não há uma versão específica de BGP para tratar esta tarefa.

Para que o BGP possa trabalhar com informações de roteamento de diversos protocolos, dois novos atributos foram inseridos:

- *Multiprotocol Reachable NLRI* (MP_REACH_NLRI): carrega o conjunto de destinos alcançáveis junto com as informações do *next-hop;*
- *Multiprotocol Unreachable NLRI* (MP_UNREACH_NLRI): carrega o conjunto de destinos inalcançáveis.

Estes atributos são Opcionais e Não-Transitivos, e no caso de um roteador BGP não suportar MBGP, este deve ignorar estas informações, não passando-as para seus vizinhos.

Mais informações:

- RFC 2545 - *Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing*
- RFC 4760 - *Multiprotocol Extensions for BGP-4*

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Multiprotocolo BGP

- MP_REACH_NLRI
  - *Address Family Identifier* (2 Bytes)
  - *Subsequent Address Family Identifier* (1 Byte)
  - *Length of Next Hop Network Address* (1 Byte)
  - *Network Address of Next Hop* (variável)
  - *Reserved* (1 Byte)
  - *Network Layer Reachability Information* (variável)
- MP_UNREACH_NLRI
  - *Address Family Identifier* (2 Bytes)
  - *Subsequent Address Family Identifier* (1 Byte)
  - *Withdrawn Routes* (variável)

cgi.br

As seguintes informações são carregadas por esses atributos:

MP_REACH_NLRI

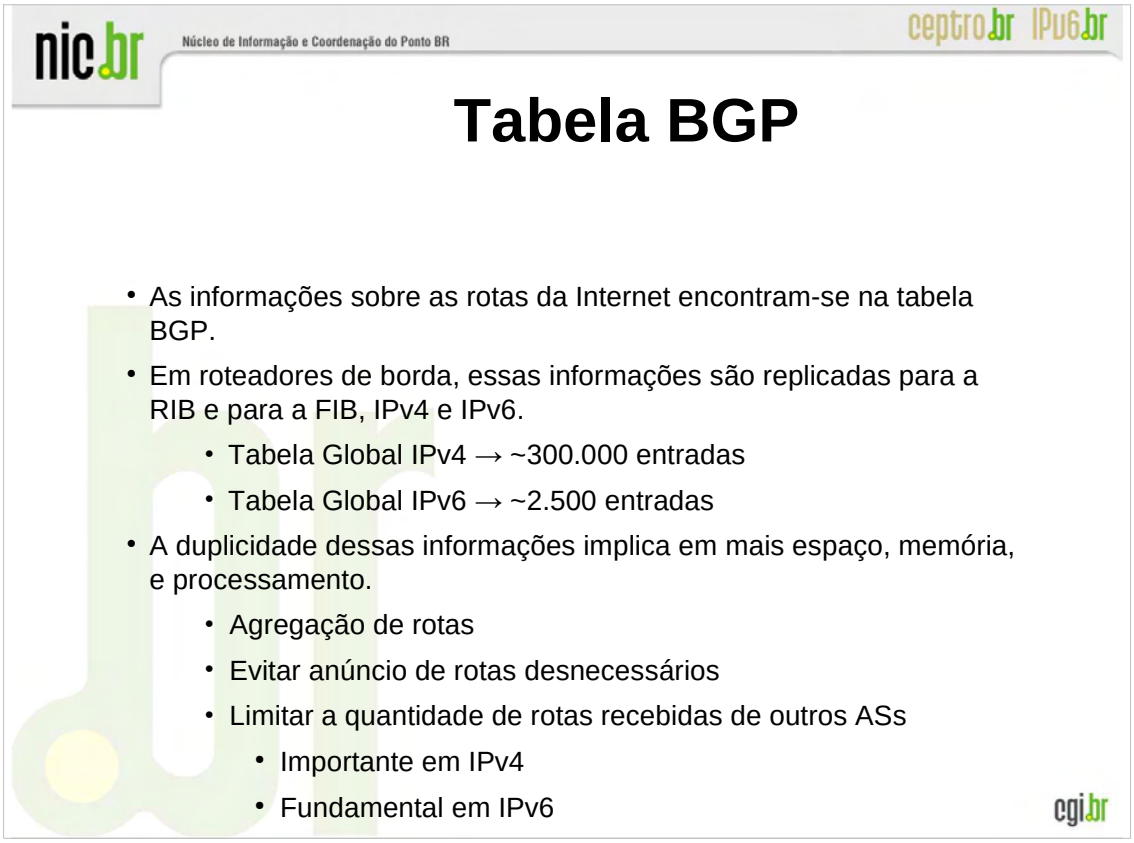
- *Address Family Identifier* (2 Bytes) - identifica o protocolo de rede a ser suportado;
- *Subsequent Address Family Identifier* (1 Byte) - identifica o protocolo de rede a ser suportado;
- *Length of Next Hop Network Address* (1 Byte) - valor que expressa o comprimento do campo *Network Address of Next Hop*, medida em Bytes;
- *Network Address of Next Hop* (variável) - contem o endereço do próximo salto;
- *Reserved* (1 Byte) - reservado;
- *Network Layer Reachability Information* (variável) - lista as informações das rotas acessíveis.

MP_UNREACH_NLRI

- *Address Family Identifier* (2 Bytes) - Identifica o protocolo de rede a ser suportado;
- *Subsequent Address Family Identifier* (1 Byte) - Identifica o protocolo de rede a ser suportado;
- *Withdrawn Routes* (variável) - lista as informações das rotas inacessíveis.

Códigos mais comuns para AFI e Sub-AFI

Código AFI	Código Sub-AFI	Significado
1	1	IPv4 Unicast
1	2	IPv4 Multicast
1	3	IPv4 based VPN
2	1	IPv6 Unicast
2	2	IPv6 Unicast e IPv6 Multicast RPF
2	3	Multicast RPF
2	4	IPv6 Label
2	128	IPv6 VPN
.....	....	...

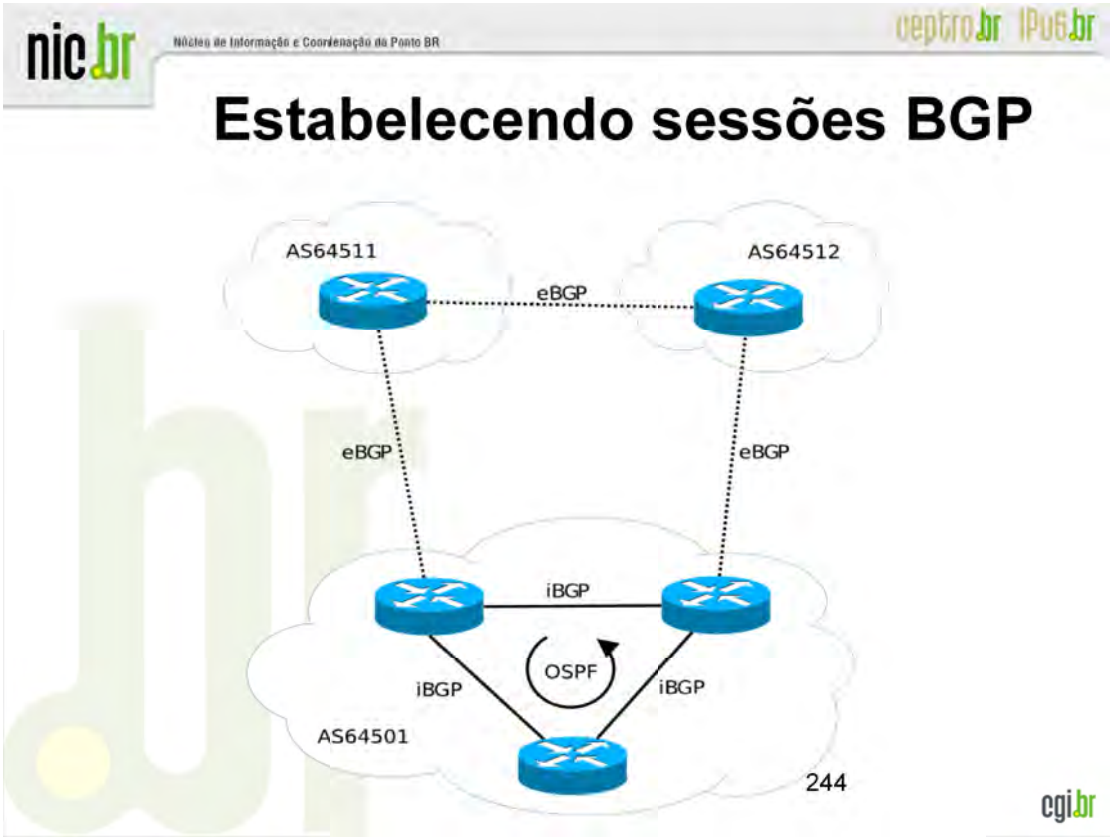


As informações sobre as rotas da Internet encontram-se na tabela BGP. Em roteadores de borda, que tratam da comunicação entre ASs, essas informações são replicadas para a RIB e para a FIB, IPv4 e IPv6.

A tabela global IPv4 possuiu hoje aproximadamente 300.000 entradas. A tabela IPv6 possui aproximadamente 2.500 entradas. A duplicidade dessas informações em arquiteturas distribuídas, implica na necessidade de mais espaço para armazenamento, mais memória e mais processamento, tanto no módulo central quanto nas placas das interfaces.

Este dados implicam em outro aspecto importante, a necessidade de se estabelecer um plano hierárquico de endereçamento para minimizar a tabela de rotas e otimizar o roteamento, evitando o anúncio de rotas desnecessárias e desagregadas.

Os ASs também podem controlar os anúncios recebidos de seus vizinhos BGP. É possível, por exemplo, limitar o tamanho dos prefixos recebidos entre /20 e /24 IPv4, e entre /32 e /48 IPv6. Porém, lembre-se que podemos anunciar até 31 prefixos IPv4 (considerando anúncios entre um /20 e um /24) e 131.071 prefixos IPv6 (considerando anúncios entre um /32 e um /48), com isso, há quem controle também a quantidade de prefixos recebidos de seus vizinhos BGP, através de comandos como `maximum-prefix` (Cisco) e `maximum-prefixes` (Juniper). Tratar esta quentão em redes IPv4 é muito importante, mas em redes IPv6 é fundamental.



Neste diagrama, podemos analisar as opções de configuração apresentadas até o momento.

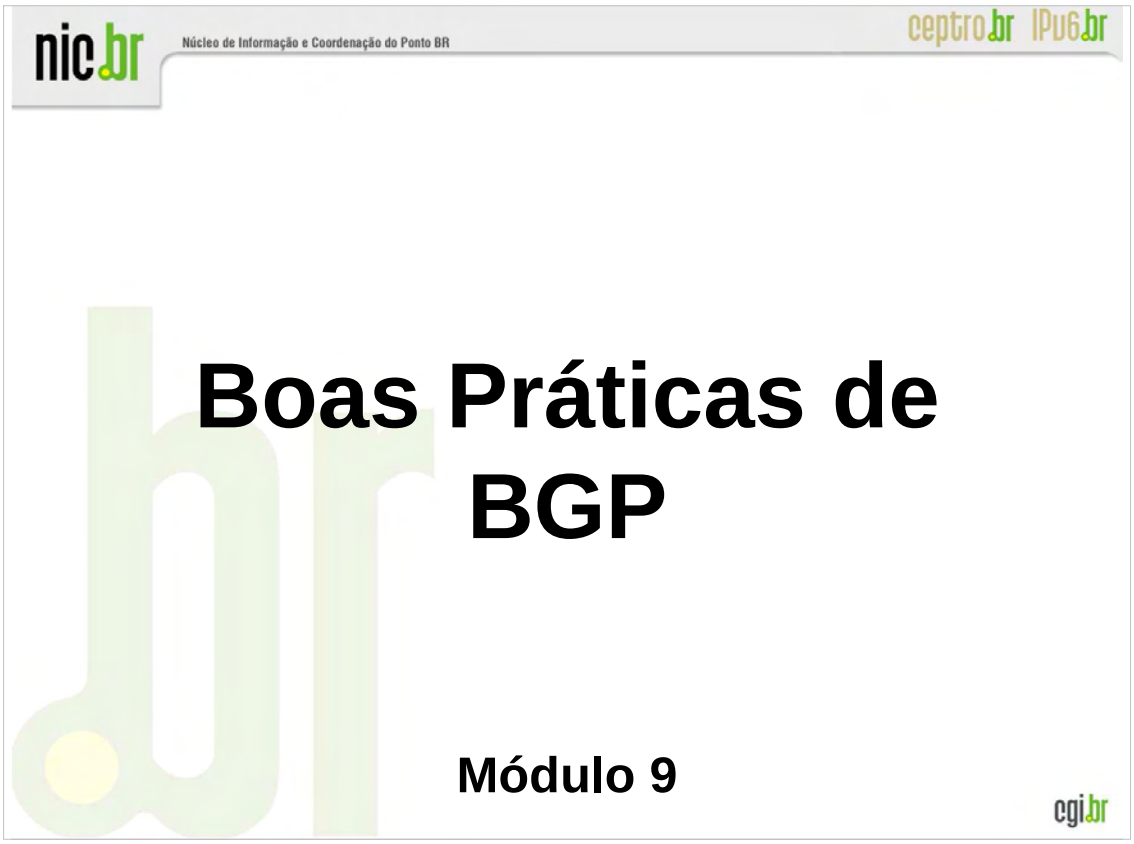
nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

IPv6.br

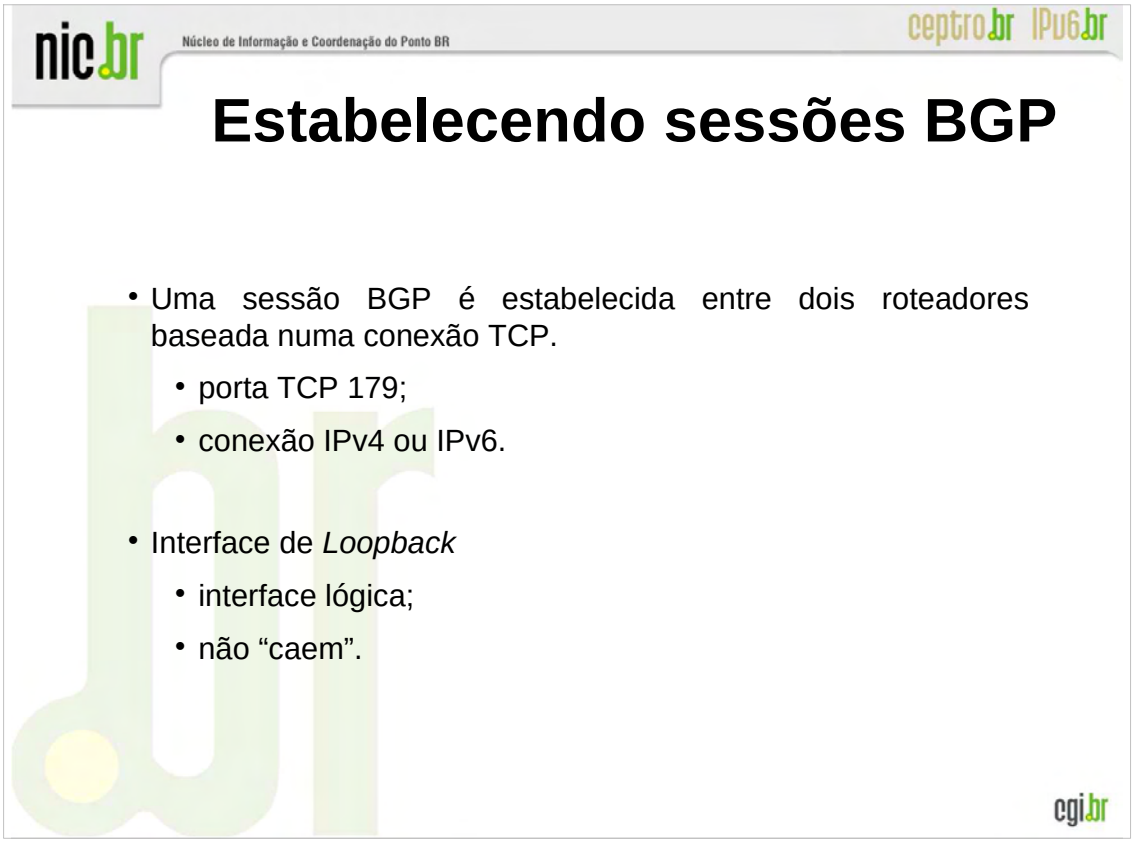
A Nova Geração do Protocolo Internet

cgi.br



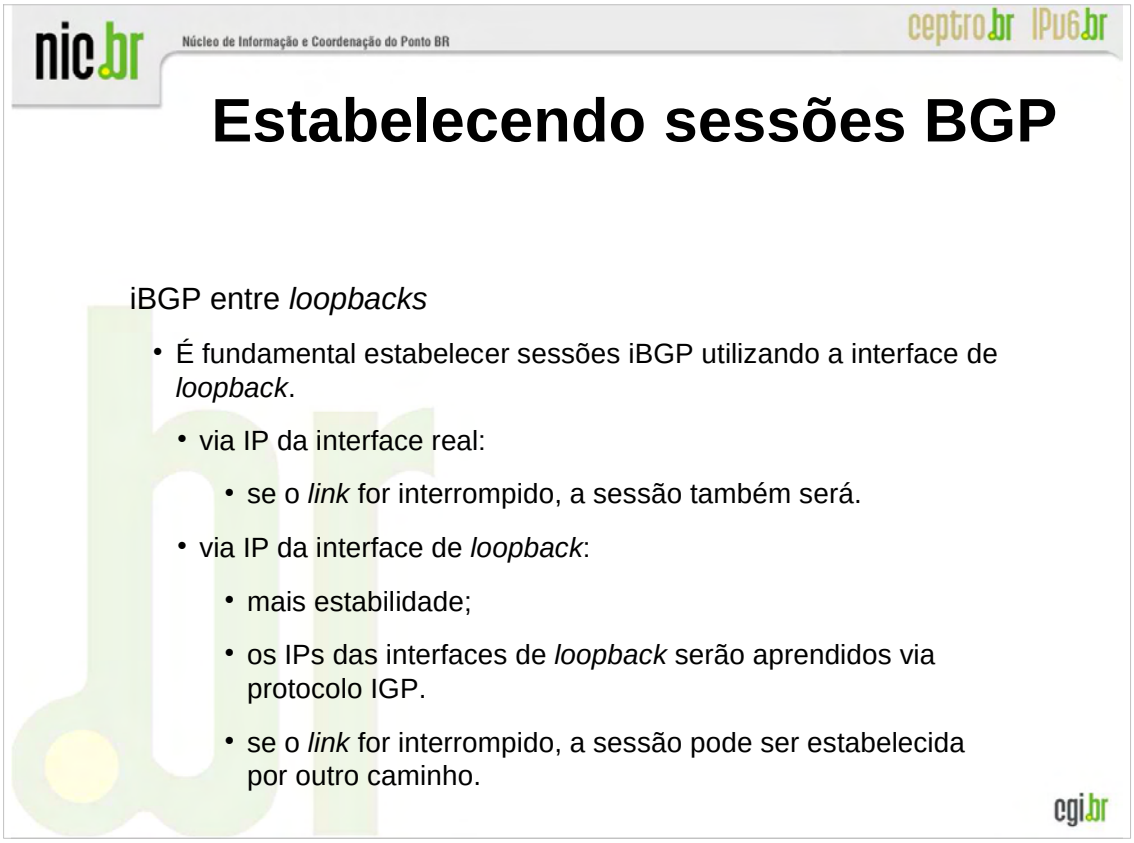
Neste módulo veremos alguns conceitos básicos de como as sessões BGP são estabelecidas; as vantagens na utilização de interfaces *loopbacks* em sessões iBGP e eBGP; alguns aspectos de segurança importantes que devem ser observados na comunicação entre ASs; formas de se garantir redundância e balanceamento de tráfego; além do detalhamento de uma série de comandos úteis para se verificar o estado das sessões BGP.

Todos esses tópicos serão abordados utilizando como base as plataformas Cisco, Quagga e Juniper, apresentando exemplos de configurações IPv6 e comparações a configurações IPv4.



Uma sessão BGP é estabelecida entre dois roteadores baseado-se em uma conexão TCP, utilizando como padrão a porta TCP 179, necessitando para isso, de uma conexão IP, seja IPv4 ou IPv6.

Uma das formas de se estabelecer essa comunicação, é através de interfaces *loopback.* Elas são interfaces lógicas, como a “null0”, ou seja, ela “não cai”, a não ser que se desligue o roteador, ou a interface seja desconfigurada.

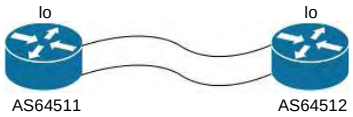


Por suas características, é fundamental que as sessões iBGP sejam estabelecidas utilizando a interface *loopback*. Caso a sessão seja estabelecida via IP da interface física, real, se o *link* for interrompido, a sessão também será. Se for estabelecida via interface *loopback*, a sessão poderá ser re-estabelecida por outro caminho aprendido através dos protocolos IGP.

A utilização de interfaces *loopback* no estabelecimento das sessões iBGP proporcionam maior estabilidade aos AS.

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Estabelecendo sessões BGP



eBGP entre *loopbacks*

- Balanceamento
  - Ex.:
    - Há dois roteadores e cada roteador representa um AS;
    - Eles estão conectados por dois *links;*
    - Utilizando o IP das interfaces reais:
      - Serão necessárias duas sessões BGP;
      - Eventualmente com políticas diferentes.
    - Utilizando o IP das interfaces de *loopbacks*
      - É estabelecida uma única sessão BGP;
      - Cria-se uma rota estática para o IP da interface *loopback* do vizinho através de cada *link.*

cgi.br

Também é recomendada a utilização de interfaces *loopback* no estabelecimento de sessões eBGP. Uma das finalidades de se estabelecer este tipo de conexão, é garantir balanceamento.

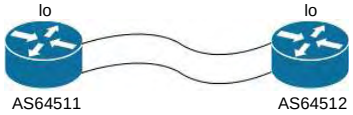
Analise o seguinte exemplo:

- Há dois roteadores, cada um representando um AS, e eles estão conectados através de dois *links*;
- Se for utilizado nesta comunicação o IP das interfaces reais será necessário estabelecer duas sessões BGP e para cada sessão, eventualmente haverá uma política diferente. Isso pode ocasionar complicações desnecessárias.
- Utilizar as interface *loopback* simplifica esse processo. Nesse caso, só será necessária uma sessão BGP, e a criação de rotas estáticas, apontando para o IP da *loopback* do vizinho através de cada *link.*

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Estabelecendo sessões BGP

eBGP entre *loopbacks*



- Essa rota estática deve ser via interface ou via IP?
  - Se for uma interface serial pode-se apontar a rota para a interface;
  - Se for uma interface Ethernet deve-se apontar para o IP.
- Em *link* serial, o tamanho de rede IPv4 normalmente utilizado é /30.
  - Um /30 possui 4 IP; rede; *broadcast;* e os dois lados;
  - Em *links* seriais pode-se utilizar /31.
- Qual o equivalente ao /31 em IPv6?
- Em IPv6 pode-se trabalhar com redes /64 em *links* seriais.
- Uma boa opção é trabalhar com /112.

cgi.br

No estabelecimento da sessão eBGP através da *loopback*, a rota estática deve ser criada via interface ou via IP?

Se for uma interface serial pode-se apontar a rota para a interface. Interfaces seriais são um "tubo", as informações que trafegam por ela chegam diretamente do outro lado, portanto, pode-se apontar a rota para a interface.

Caso seja um meio compartilhado, uma interface Ethernet por exemplo, deve-se apontar para o IP.

Outro ponto importante é o tamanho de rede utilizado nestes tipos de *links*. Em um *link* serial, normalmente utiliza-se prefixos de redes /30 IPv4. Com isso, são possíveis quatro endereços IP: o de rede; de *broadcast*; e os dois que vão identificar as interfaces. No entanto, em um *link* serial não são necessários os endereços de rede nem de *broadcast*, por isso, há uma abordagem que utiliza prefixos de rede /31 neste tipo de *link*. Este método permite também a economia de uma grande quantidade de endereços IPv4, principalmente em operadora que trabalham com milhares de *links* ponto-a-ponto. Porém, isto só é recomendado para *links* seriais, não em Ethernet.

Em redes IPv6, o equivalente ao /31 IPv4 seria um /127. A RFC 3627 não recomenda a utilização de /127, devido a possíveis problemas com o endereço *anycast Subnet-Router*, no entanto, existe um *draft* que questiona esse argumento.

Existem diversas possibilidades para se trabalhar em IPv6. Em *links* ponto-a-ponto pode-se utilizar um prefixo /64 ou /126, mas uma opção interessante é utilizar /112, de modo a se trabalhar apenas com o último grupo de Bytes do endereço.

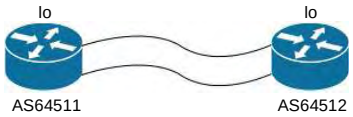
Mais informações:

- RFC 3021 - *Using 31-Bit Prefixes on IPv4 Point-to-Point Links*
- RFC 3627 - *Use of /127 Prefix Length Between Routers Considered Harmful*
- draft-kohno-ipv6-prefixlen-p2p-00.txt - *Use of /127 IPv6 Prefix Length on P2P Links Not Considered Harmful*

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Estabelecendo sessões BGP

eBGP entre *loopbacks*



- Normalmente utilizam-se na interface de *loopback* prefixo /32 IPv4 ou /128 IPv6.
- O IP da *loopback* é de responsabilidade do próprio AS.
  - Não se deve utilizar IP privado.
- O IP do *link* de trânsito é da responsabilidade do Provedor de Trânsito.
  - Esse IP deve ou pode ser roteável?
    - Se for uma relação IP interna com a operadora, pode ser um IP válido, da operadora e não roteável, ex. conexão MPLS;
    - Se for um serviço Internet, o IP DEVE ser roteável.

cgi.br

Em relação ao endereçamento das interface *loopback*, normalmente utilizam-se prefixos /32 IPv4 e /128 IPv6. Isso desmistifica a ideia de que só é possível utilizar prefixos /64. A utilização de /64 só é obrigatória quando utiliza-se o protocolo de Descoberta de Vizinhança.

O endereço IP utilizado na *loopback* deve fazer parte do bloco do próprio AS e deve ser um endereço válido, não pode-se utilizar IPs privados. Os endereços IPs privados são para uso interno na sua rede e a comunicação entre ASs é uma conexão Internet, o que exige IPs válidos.

O endereço da interface real, conectada ao *link* de trânsito, deve ser do bloco da operadora que fornece o serviço, do *UpStream Provider*. Além disso, esse endereço IP deve ser roteável. Está é uma questão bastante polêmica, pois há quem defenda que esse endereço não seja roteável devido a questões de segurança. Se houver uma relação IP interna com a operadora, por exemplo uma conexão MPLS, é interessante que se utilize um IP válido, da operadora, e não roteável. No entanto, se for um serviço Internet, a operadora tem obrigação de fornecer um endereço roteável, porque ter conectividade é um ponto fundamenta em um serviço Internet.

Ex. 1 – O tráfego gerado por um roteador sai com IP da interface de saída como IP de origem. Com isso, se o seu roteador se conectar a um servidor NTP, seja IPv4 ou IPv6, o IP de origem desse pacote será o IP da interface pela qual ele sabe chegar ao destino. Se for um IP não roteável, o pacote vai chegar ao servidor, mas o servidor não saberá como retornar a resposta.

Ex. 2 - Há quem peça para a operadora não rotear esse IP por questões de segurança. Se os IPs internos forem roteáveis, não há proteção alguma, pois se houver apenas um IP roteável, ele será conhecido pelo mundo e haverá outro caminho para se chegar até ele, por exemplo, entrando em outro elemento do AS que conheça o IP do roteador.

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Estabelecendo sessões BGP

eBGP entre *loopbacks*

- Segurança
  - A utilização de interfaces *loopbacks* em sessões eBGP não é necessária apenas para garantir balanceamento.
  - Estabelecer sessões eBGP utilizando o IP da interface, facilita muito ataques contra a infraestrutura.
  - É recomendável trabalhar eBGP entre *loopbacks* mesmo que só haja um *link*.

cgi.br

Há quem defenda que a utilização de interface *loopback* só é realmente necessária apenas para garantir o balanceamento do tráfego. Essa prática auxilia também em relação a questões de segurança.

Estabelecer sessões eBGP utilizando o IP da interface, pode facilitar ataques contra a infraestrutura de um AS. Por isso, é recomendável trabalhar sessões eBGP entre *loopbacks* independente da existência de apenas um *link*.

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Estabelecendo sessões BGP

eBGP entre *loopbacks*

- Segurança
  - Ex.:
    - Para estabelecer uma sessão eBGP sobre TCP são necessárias 4 informações básicas:
      - 2 IPs e 2 portas TCP (179 e >1024).
    - Se a sessão eBGP for estabelecida utilizando o IP da interface:
      - normalmente identifica-se um dos IP utilizando traceroute;
      - descobrindo o primeiro, descobre-se o segundo, visto que normalmente utiliza-se /30;
      - a terceira informação é uma porta padrão, a 179.
    - Ou seja, de 4 variáveis 3 podem ser descobertas de forma relativamente fácil.

cgi.br

Esta questão pode ser exemplificada com o cenário a seguir:

- Uma sessão eBGP entre dois roteadores é estabelecida através de uma conexão TCP;
- Para isso, são necessárias quatro variáveis básicas: dois endereços IP e duas portas TCP, uma padrão, a 179, e da mesma forma que uma aplicação HTTP, o roteador que iniciar a sessão BGP, vai sair por uma porta alta, maior que 1024, para fechar com a 179;
- Se a sessão eBGP for estabelecida utilizando o IP da interface, é possível identificar esse IP utilizando comandos como o *traceroute*. Como normalmente são utilizados prefixo /30 IPv4, se for descoberto um IP, descobrir o segundo torna-se mais simples. Com isso, das quatro variáveis, duas são fáceis de descobrir;
- A terceira é uma porta padrão, a 179;
- Ou seja de quatro variáveis três podem ser descobertas de forma relativamente fácil, e a única variável que falta, a porta alta, também não apresenta dificuldade para sua descoberta.

nic.br Núcleo de Informação e Coordenação do Ponto BR ceptro.br IPv6.br

Estabelecendo sessões BGP

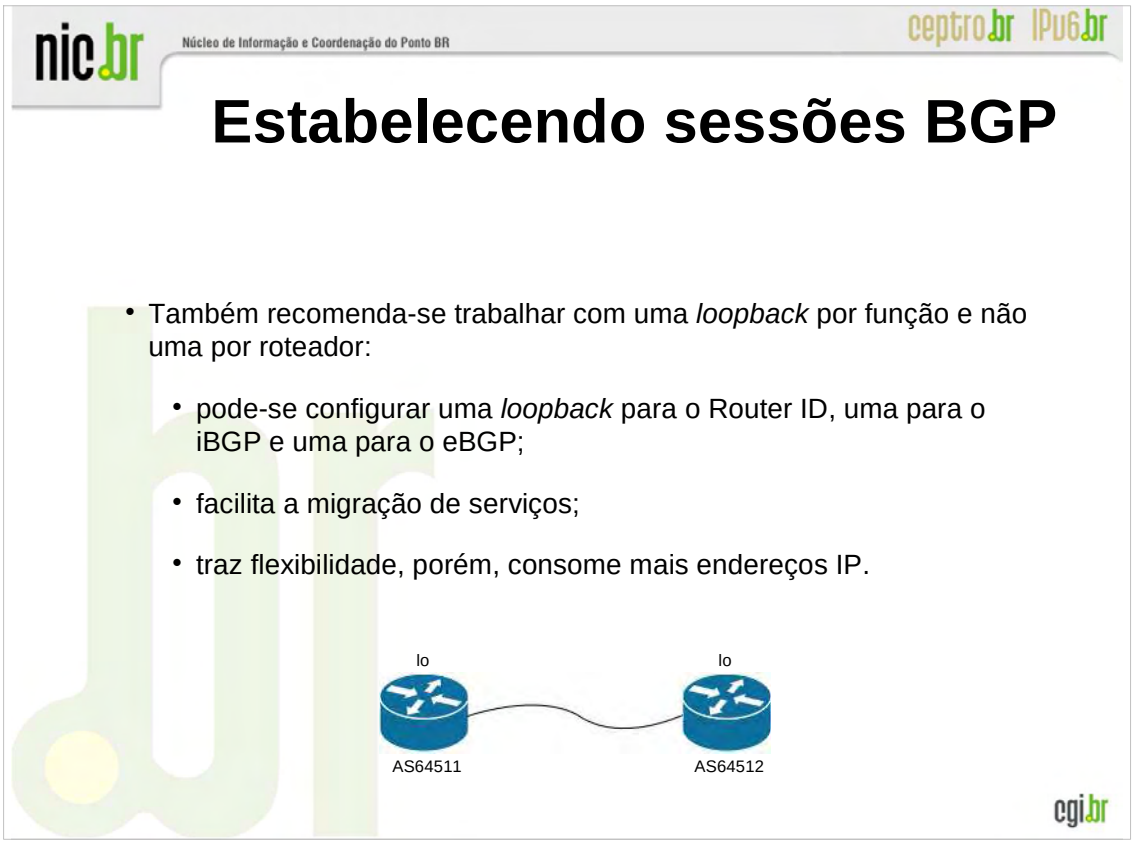
eBGP entre *loopbacks*

- Segurança
  - Uma das formas de derrubar um As ou um destino, é derrubar o AS que provê conectividade para ele.
  - Estabelecendo uma sessão eBGP utilizando *loopbacks:*
    - os IPs são das redes internas, não tendo relações entre eles;
    - dificulta a descoberta via traceroute.

cgi.br

Uma das formas de “derrubar” um AS ou um destino, é “derrubar” o AS que provê conectividade para ele e isso pode ser feito interrompendo as sessões eBGP.

Estabelecer a sessão eBGP utilizando a sessão *loopback* apresenta alguns pontos relativos a segurança. Os IPs da *loopback* são IPs da rede interna, não sendo descobertos com *traceroutes* facilmente, e o fato dos dois IPs serem totalmente distintos, não tendo relações entre eles, dificulta ainda mais.

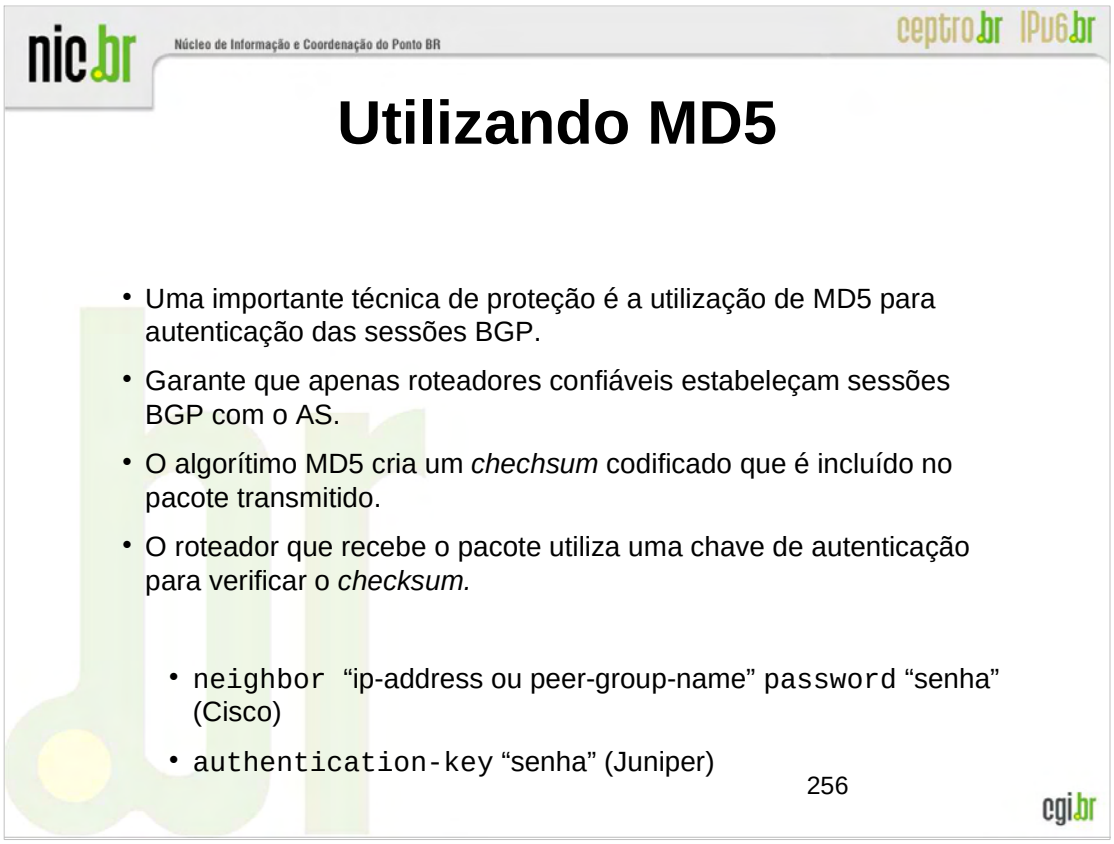


Outro aspecto que se deve destacar em relação à utilização da interface *loopback* é o de trabalhar com uma *loopback* por função e não uma única *loopback* por roteador. Por exemplo, pode-se configurar uma *loopback* para o Router ID, uma para o iBGP e uma para o eBGP.

Ex.:

- Há uma sessão eBGP estabelecida e é preciso migrar essa sessão para um outro roteador;
- Se além de ser a *loopback* usada para a sessão eBGP ela também for usada para *n* outras funções. Se for assim, a migração será mais complicada;
- Se for uma *loopback* por função, a migração poderá ser realizada sem interferir com as outras funções. Pode-se mudar o iBGP, o Router ID, alterar a sessão eBGP de um roteador para outro sem ter que avisar a operadora e não tem que mudar outros serviços internos.

Esta prática apresenta uma maior flexibilidade, apesar de consumir mais endereços IP. Entretanto, como normalmente são utilizados prefixos /32 ou /128, esta questão não é tão grave.



Uma importante técnica de proteção é a utilização de MD5 para autenticação das sessões BGP. Desta forma garante-se que apenas roteadores confiáveis estabeleçam sessões BGP com o AS.

O algorítimo MD5 cria um *chechsum* codificado que é incluído no pacote transmitido e o roteador que recebe o pacote utiliza uma chave de autenticação para verificar o *hash.*

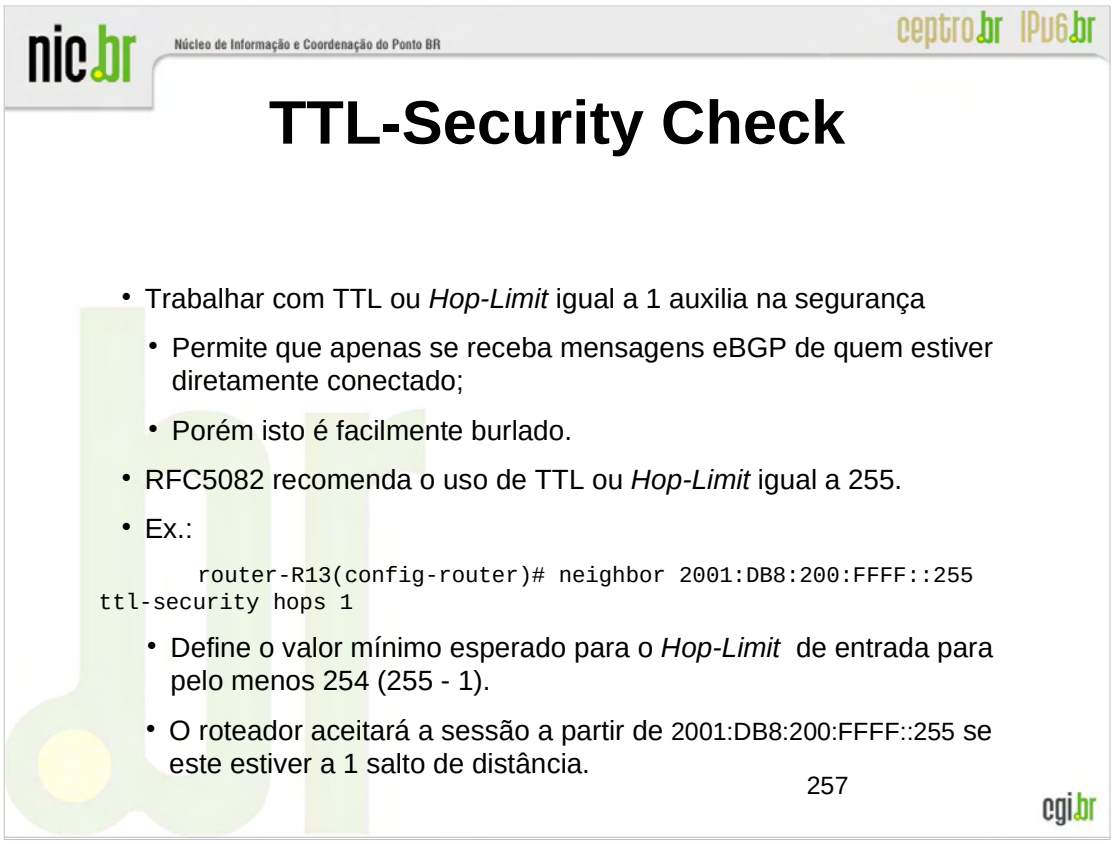
Utilize os seguintes comandos para habilitar essa funcionalidade:

neighbor "ip-address ou peer-group-name" password "senha" (Cisco)

authentication-key "senha" (Juniper)

Mais informações:

- RFC 1321 - *The MD5 Message-Digest Algorithm*
- RFC 2385 - *Protection of BGP Sessions via the TCP MD5 Signature Option*



Trabalhar com TTL ou *Hop-Limit* igual a 1 auxilia na segurança das sessões BGP, pois permite que apenas se receba mensagens eBGP de quem estiver diretamente conectado. Porém isso é facilmente burlado, basta utilizar comandos como traceroute para identificar quantos saltos são necessários para chegar ao roteador de destino e gerar um pacote como valor do TTL necessário para alcança-lo.

A RFC 5082 recomenda que se trabalhe com TTL ou *Hop-Limit* igual a 255 em vez de 1. Deste modo, em uma sessão eBGP diretamente conectada utilizando o IP da interface, é possível garantir que o vizinho BGP está a no máximo um salto de distância através da leitura do valor do TTL, que terá sido decrementado a cada hop.

Para utilizar essa funcionalidade é preciso configurar os dois vizinhos participantes da sessão eBGP. Quem envia a mensagem deve montar o pacote com TTL ou *Hop-Limit* igual a 255 e quem recebe deve habilitar a verificação desse campo. Em roteadores Cisco é possível fazer a verificação da seguinte forma:

```
router-R13(config-router)#neighbor 2001:DB8:200:FFFF::255 ttl-security hops 1
```

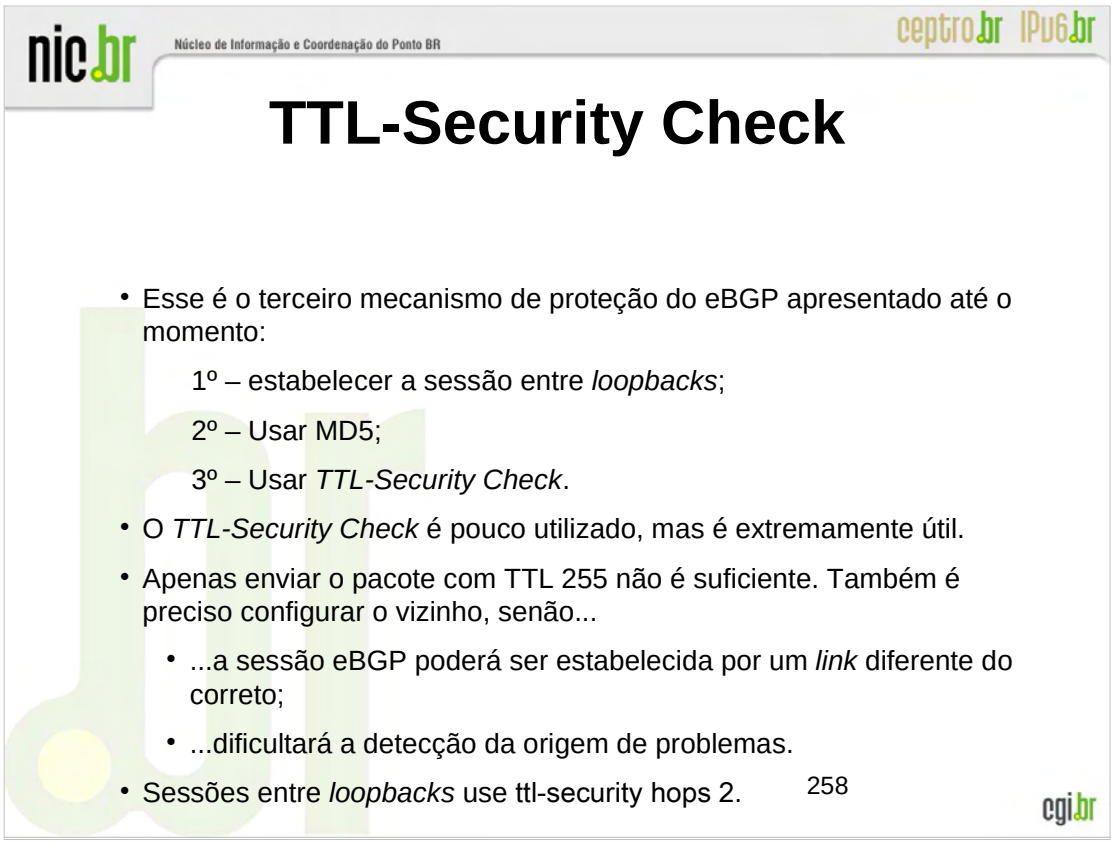
Deste modo, define-se o valor mínimo esperado para o *Hop-Limit* de entrada para pelo menos 254 (255 – 1). Com isso, o roteador aceitará a sessão a partir de 2001:DB8:200:FFFF::255 se este estiver a 1 salto de distância.

Com um vizinho BGP IPv4 essa linha seria:

```
router-R13(config-router)#neighbor 10.2.255.255 ttl-security hops 1
```

Mais informações:

- RFC 5082 - The Generalized TTL Security Mechanism (GTSM)
- http://www.cisco.com/en/US/docs/ios/12_3t/12_3t7/feature/guide/gt_btsh.html
- http://www.juniper.net/us/en/community/junos/script-automation/library/configuration/ttl-security/



Esse é o terceiro mecanismo de proteção do eBGP apresentado até o momento:

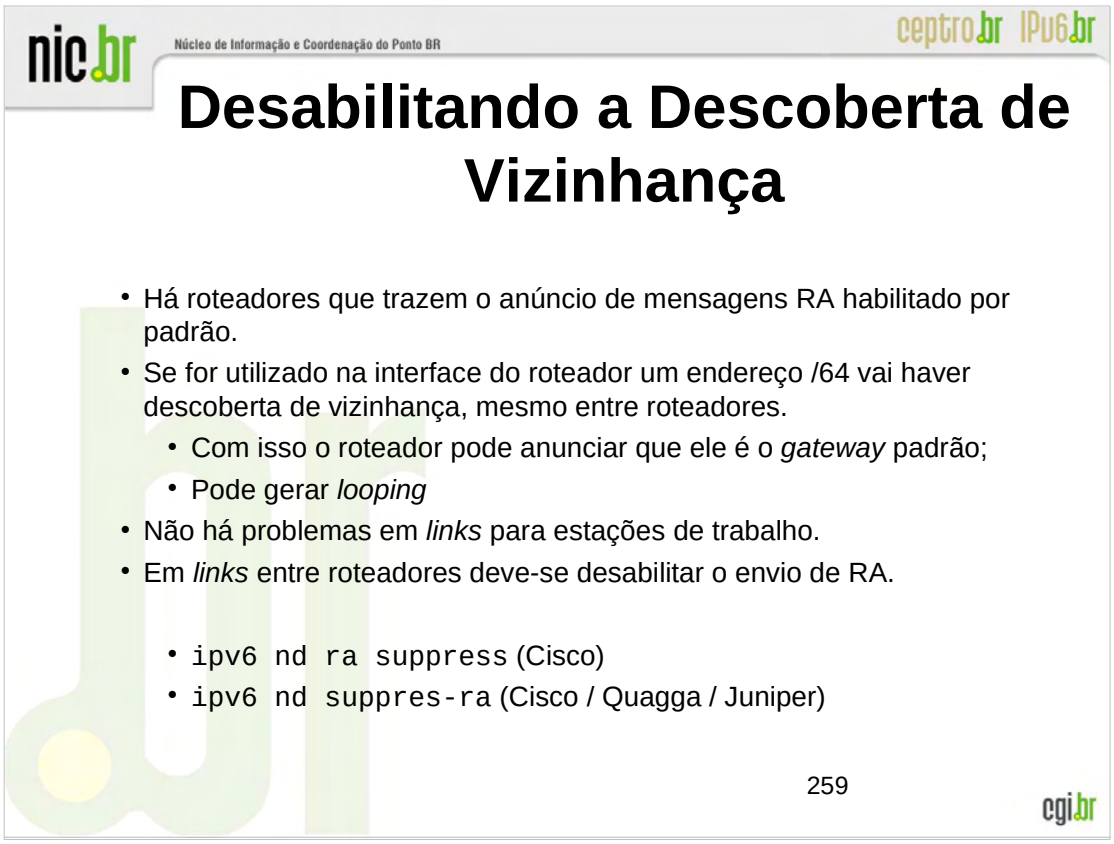
1º – estabelecer a sessão entre *loopbacks*;

2º – Usar MD5;

3º – Usar *TTL-Security Check.*

O *TTL-Security Check* é pouco utilizado, mas é extremamente útil. No entanto, apenas enviar o pacote com TTL 255 não é suficiente, também é preciso configurar o vizinho. Caso isso não ocorra, a sessão eBGP poderá ser estabelecida por um *link* diferente do correto, o que dificultará a detecção da origem de problemas.

Outro ponto importante, é que em sessões entre *loopbacks* deve-se usar `ttl-security hops 2`.



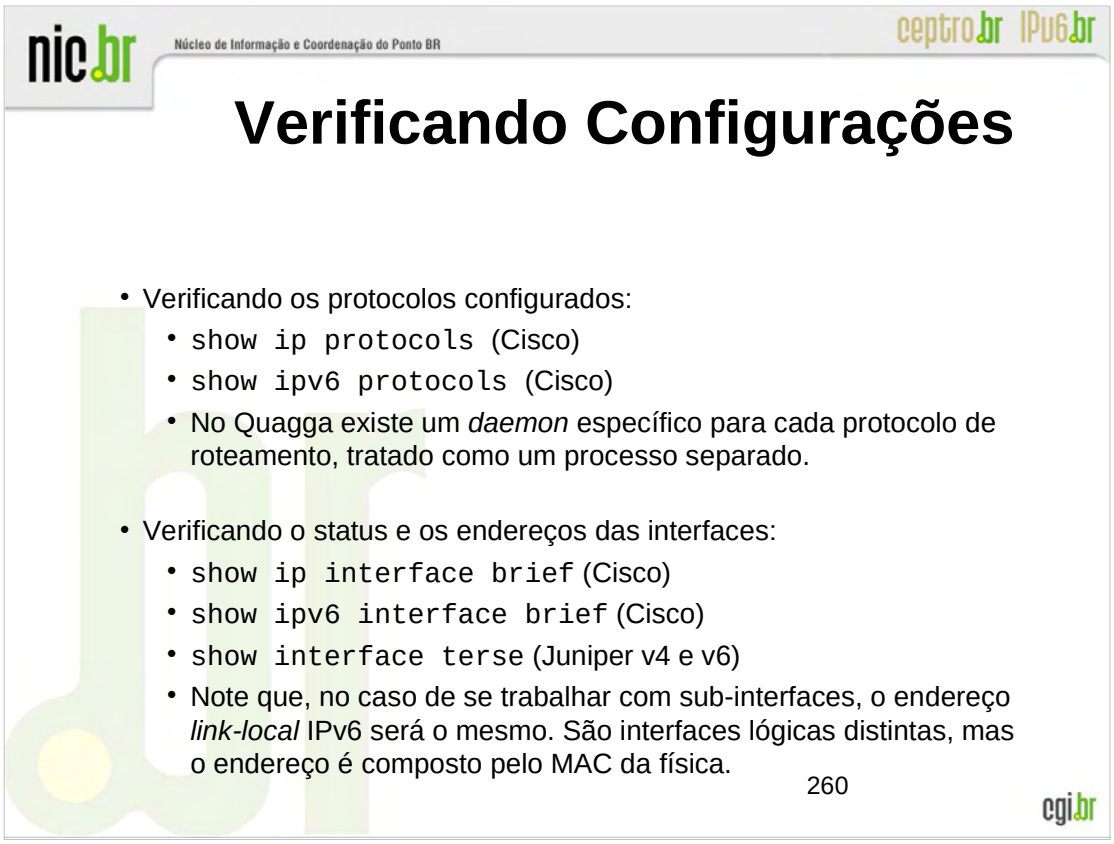
Um ponto importante na configuração de roteadores IPv6 é o funcionamento do protocolo de Descoberta de Vizinhança. Há roteadores que trazem o anúncio de mensagens RA (*Router Advertisements*) habilitado por padrão. Caso seja utilizado na interface do roteador um endereço / 64, vai haver descoberta de vizinhança, mesmo entre roteadores. Com isso o roteador pode anunciar que ele é o *gateway* padrão para os outros roteadores da rede, podendo gerar *loopings.*

Esta funcionalidade só deve ser habilitada em interfaces que estejam conectadas a estações de trabalho ou em alguns casos, a servidores. Em *links* entre roteadores deve-se desabilitar o envio das mensagens RA.

Esta funcionalidade pode ser desabilitada com a utilização dos seguintes comandos:

ipv6 nd ra suppress (Cisco)

ipv6 nd suppres-ra (Cisco / Quagga / Juniper)



Ter conhecimento das funcionalidades e configurações habilitas nos roteadores é muito importante principalmente quando se obtém um equipamento novo. Alguns comandos que podem facilitar essa tarefa são:

Para verificar os protocolos configurados podemos utilizar:

show ip protocols (Cisco)

show ipv6 protocols (Cisco)

No Quagga existe um *daemon* específico para cada protocolo de roteamento, tratado como um processo separado.

Para verificar o status e os endereços das interfaces utilizamos:

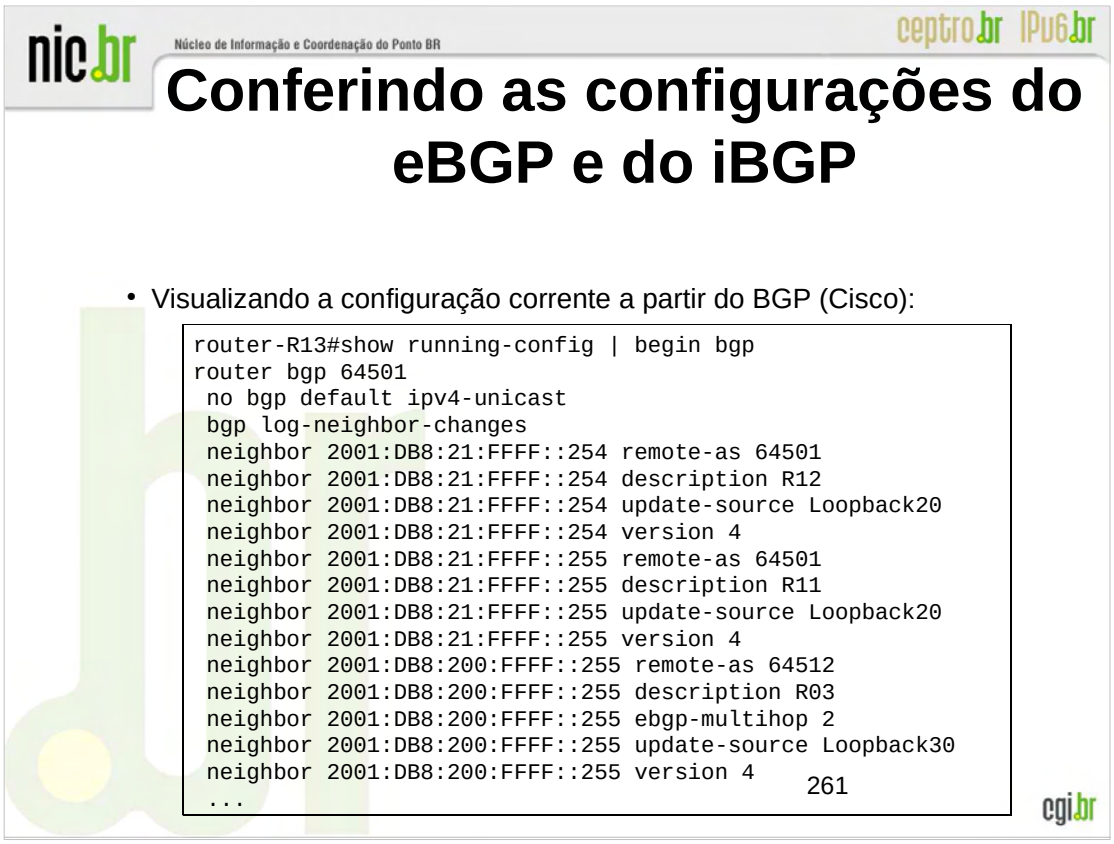
show ip interface brief (Cisco)

show ipv6 interface brief (Cisco)

show interface terse (Juniper v4 e v6)

Note que, no caso de se trabalhar com sub-interfaces, o endereço *link-local* IPv6 será o mesmo. São interfaces lógicas distintas, mas o endereço é composto pelo MAC da física.

Esses comandos são úteis porque mostram de forma resumida o está configurado no equipamento, que interfaces já estão habilitadas com IPv6 por exemplo, etc.



Existem alguns comando que podem facilitar a analisar e conferência das políticas de entrada e saída de um roteador.

Para visualiza a configuração corrente a partir do BGP em um roteador Cisco ou Quagga:

show running-config | begin bgp

Em nosso exemplo, a primeira linha da configuração BGP indica o ASN do próprio AS:

- router bgp 64501

Por padrão, os roteadores cisco e quagga só conhecem uma família de endereços, a *ipv4-unicast.* Para utilizar outras famílias de endereços, utiliza-se o conceito de *address-family*, e para habilitá-lo utiliza-se o comando:

- no bgp default ipv4-unicast

Para habilitar IPv6 em roteadores Cisco e Quagga recomenda-se que seja marcada uma janela de manutenção, para deste modo, poder interromper o tráfego, aplicando o comando

- no router bgp 64501

e refazer toda a configuração com *address-family.*

Mesmo utilizando *address-family*, no início da configuração sempre são apresentadas as informações gerais que independem da família. Por exemplo:

- `neighbor 2001:DB8:200:FFFF::255 remote-as 64512` – indica o IP e o ASN do vizinho. Se indicar o ASN do próprio AS, é porque trata-se de uma sessão iBGP;
- `neighbor 2001:DB8:200:FFFF::255 description R03` – apresenta um nome de identificação;

- `neighbor 2001:DB8:200:FFFF::255 ebgp-multihop 2` – especifica o número de saltos até se alcançar o vizinho. Uma diferença importante entre o iBGP e o eBGP, é que o quando o roteador gera uma mensagem eBGP, o pacote IP que carrega essa mensagem é enviado com o valor do TTL, se for IPv4, ou do *Hop_Limit*, se for IPv6, igual a 1, e com isso, ele só poderá alcançar um salto. Se um pacote tiver que ser roteado para uma interface *loopback*, e a mensagem sair como o TTL igual a 1, o roteador de destino ao abri-lo, irá decrementar o valor do TTL e não poderá realizar o roteamento interno para *loopback*. Por tanto, para levantar a sessão eBGP entre *loopbacks*, deve-se especificar qual o número de saltos;

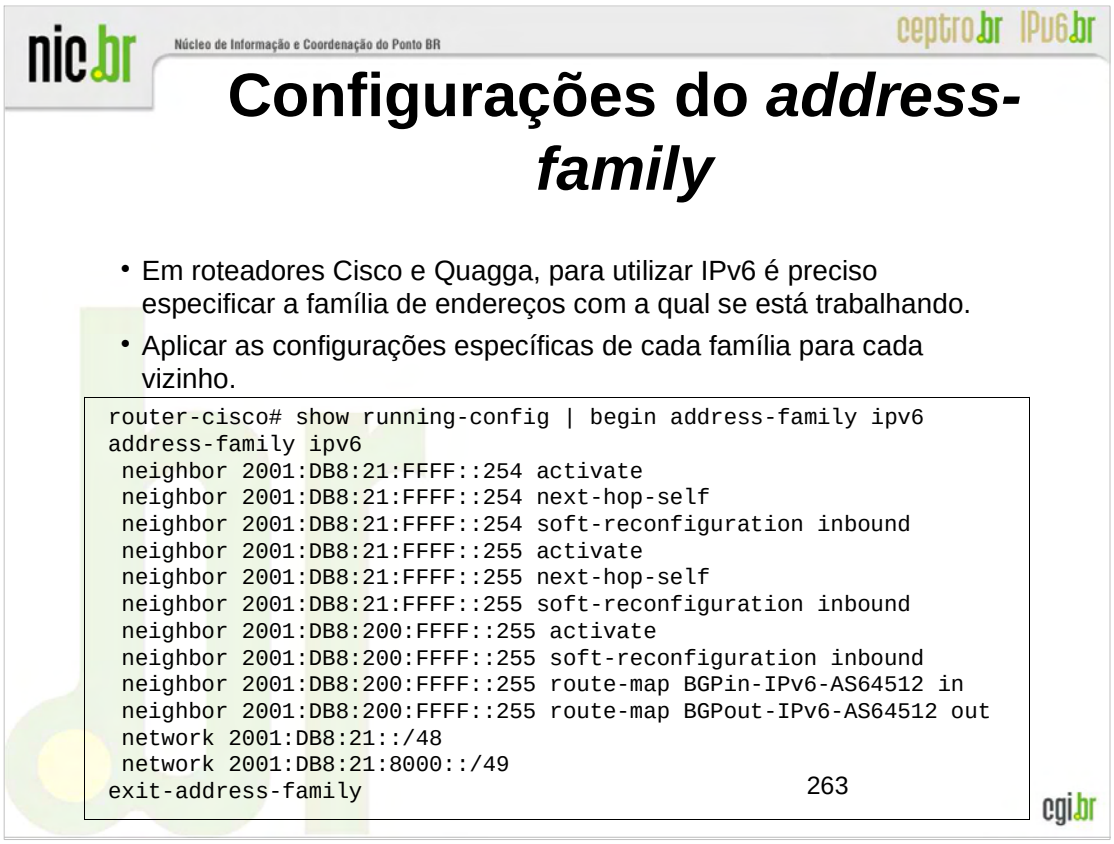
- `neighbor 2001:DB8:200:FFFF::255 update-source Loopback30` - configura o roteador para que o IP de origem dos pacotes seja o da *loopback*, porque ao enviar uma mensagem, o roteador adota por padrão, como endereço IP de origem o IP da interface por onde ela é enviada.

- `neighbor 2001:DB8:200:FFFF::255 version 4` – indica a versão de protocolo BGP utilizada. Essa informação agiliza o estabelecimento da sessão BGP, visto que, na primeira mensagem trocada entre os vizinhos, são passadas algumas informações, entre elas, há a negociação da versão. Se já for informado desde o início qual a versão utilizada, essa negociação não necessita ser feita.

Em relação a configuração do iBGP, as únicas diferenças são que o ASN é o do próprio AS e que diferente do eBGP, não precisa alterar o TTL. Por padrão, no iBGP o TTL não é alterado, presumindo que a sessão possa fazer um caminho longo, saindo com TTL igual a 255 ou outro valor intermediário dependendo da implementação.

Observe a seguir um exemplo das configurações de uma sessão BGP IPv4:

```
router-R13#show running-config | begin bgp
router bgp 64501
 no bgp default ipv4-unicast
 bgp log-neighbor-changes
 neighbor 10.2.255.255 remote-as 64512
 neighbor 10.2.255.255 description R03
 neighbor 10.2.255.255 ebgp-multihop 2
 neighbor 10.2.255.255 update-source Loopback30
 neighbor 10.2.255.255 version 4
 neighbor 172.21.15.254 remote-as 64501
 neighbor 172.21.15.254 description R12
 neighbor 172.21.15.254 update-source Loopback20
 neighbor 172.21.15.254 version 4
 neighbor 172.21.15.255 remote-as 64501
 neighbor 172.21.15.255 description R11
 neighbor 172.21.15.255 update-source Loopback20
 neighbor 172.21.15.255 version 4
 ...
```



Em roteadores Cisco e Quagga, para utilizar IPv6 é preciso especificar a família de endereços com a qual se está trabalhando. Diferente dos roteadores Juniper, as configurações do BGP são apresentadas dividas em configurações geais e nas configurações específicas de cada família para cada vizinho.

Para analisar as configurações do *address-family* IPv6 utiliza-se:

```
show running-config | begin address-family ipv6
```

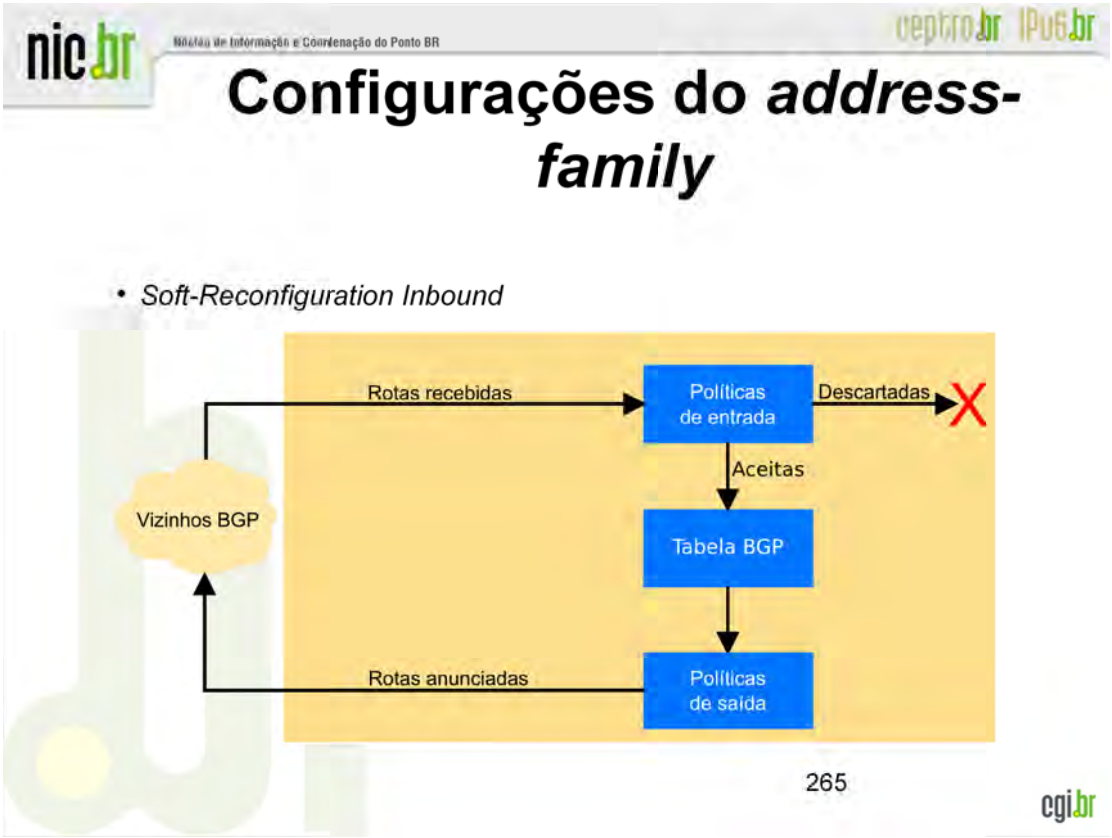
- `address-family ipv6` – indica a qual família pertence as configurações;
- `neighbor 2001:DB8:200:FFFF::255 activate` – ativa a sessão, necessário quando se trabalha com *address-family*. Uma pratica boa a se aplicar quando se configura uma sessão iBGP ou eBGP, é levantá-la em *shutdown*. Isso evita que se estabeleça a sessão sem que as políticas estejam configuradas, não permitindo que se envie informações indevidas;
- neighbor `2001:DB8:200:FFFF::255` soft-reconfiguration inbound – indica a forma que a tabela de rotas será atualizada;
- neighbor `2001:DB8:200:FFFF::255` prefix-list BGPout-IPv6-AS64512 out – indica a política de saída aplicada;
- neighbor `2001:DB8:200:FFFF::255` route-map BGPin-IPv6-AS64512 in – indica a política de saída aplicada.

Em relação as configurações do iBGP destaca-se a seguinte informação:

- `neighbor 2001:DB8:21:FFFF::254 next-hop-self` – indica quem é o próximo salto. Esta configuração ajuda a trazer mais estabilidade e facilita na operação dos ASs. Um roteador de borda pode repassar para os demais roteadores de seu AS, via iBGP, todos os prefixos que ele aprender de seus ASs vizinhos. Quando ele repassa para os demais roteadores, é mantido o atributo *next-hop*. No entanto, o *next-hop* desses prefixos sempre será o roteador de borda dos ASs vizinhos, aos quais os roteadores internos não possuem conectividade direta. Para solucionar esse problema, o roteador de borda do AS repassa os anúncios informando que o *next-hop* para os ASs vizinhos é ele mesmo através do comando next-hop-self. Com isso, os roteadores internos só precisam saber chegar no roteador de borda de seu AS, que é quem tem conectividade para a Internet.

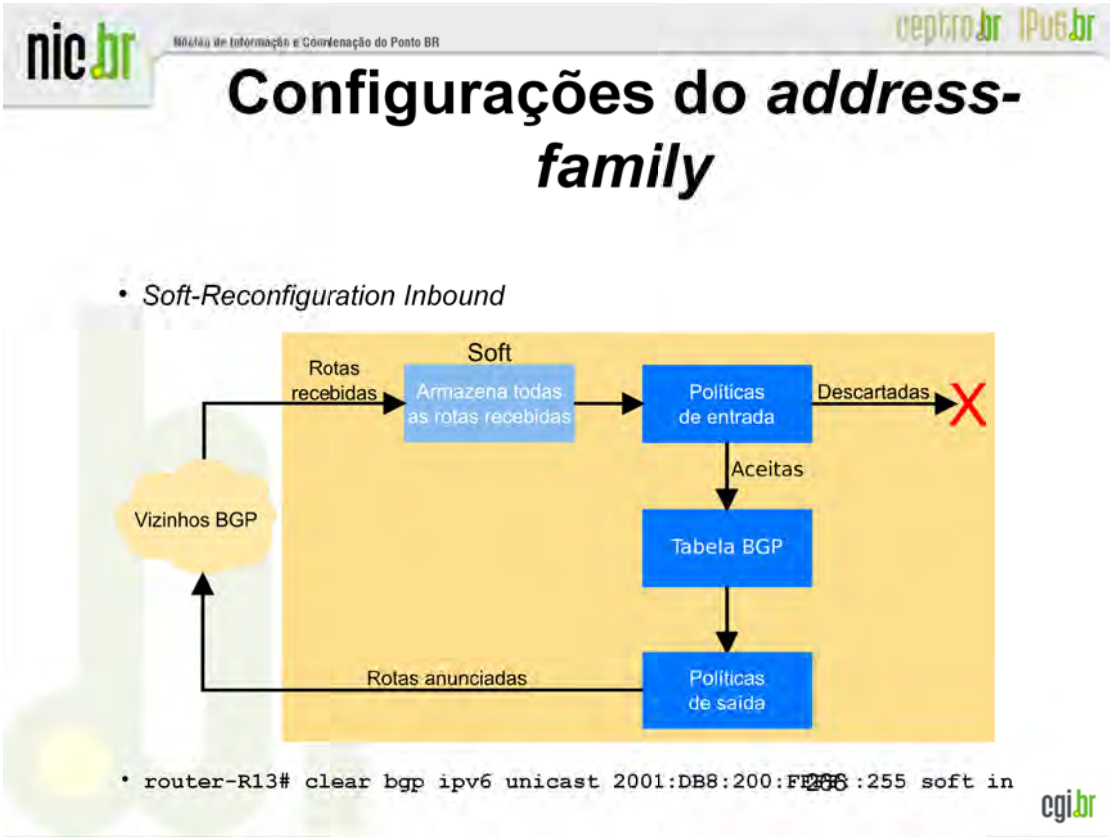
Um exemplo de configuração do *addres-family* IPv4 pode ser observado a seguir:

```
router-cisco# show running-config | begin address-family ipv4
address-family ipv4
 neighbor 10.2.255.255 activate
 neighbor 10.2.255.255 soft-reconfiguration inbound
 neighbor 10.2.255.255 prefix-list BGPout-IPv4-AS64512 out
 neighbor 10.2.255.255 route-map BGPin-IPv4-AS64512 in
 neighbor 172.21.15.254 activate
 neighbor 172.21.15.254 next-hop-self
 neighbor 172.21.15.254 soft-reconfiguration inbound
 neighbor 172.21.15.255 activate
 neighbor 172.21.15.255 next-hop-self
 neighbor 172.21.15.255 soft-reconfiguration inbound
 network 172.21.0.0 mask 255.255.240.0
 network 172.21.8.0 mask 255.255.248.0
exit-address-family
```



Um comando interessante é o `soft-reconfiguration inbound`. Analise o seguinte exemplo:

- O roteador R1 levanta uma sessão BGP com o roteador R2;
- Quando a sessão é estabelecida, o roteador envia todas as informações que ele conhece;
- Novas informações só serão enviadas quando houver a necessidade de se adicionar ou retirar entradas da tabela;
- Se for criada uma política de entrada no R2, a mensagem original, trocada no estabelecimento da sessão, será alterada;
- Caso seja necessário criar uma nova política em R2, não se terá mais as informações iniciais para poder aplicá-las;



Uma forma de se recuperar essas informações seria “derrubar” a sessão, para que assim o roteador mande novamente todos os seus prefixos. Essa prática era funcional quando não havia tantas entradas na tabela global de rotas. Hoje em dia ela não é mais efetiva.

Outra opção é utilizar o comando `soft-reconfiguration inbound`. Com isso, antes de se aplicar as políticas é criada uma outra tabela de entrada por vizinho, exatamente igual a recebida. Deste modo, tudo o que o R1 enviar, será gravado nesta pré-tabela, salvando os prefixos originais. Se for necessário alterar alguma configuração, basta utilizar, por exemplo, o comando:

```
router-R13# clear bgp ipv6 unicast 2001:DB8:200:FFFF::255 soft in
```

Este comando faz com que o roteador re-leia a pré-tabela sem interromper a sessão. Porém, isso também não é funcional nos dias de hoje, porque a tabela BGP possui entorno de 300 mil prefixos, e com esse comando duplica-se a tabela BGP para cada vizinho, consumindo muito mais memória do seu módulo de roteamento, na parte de controle.

Um exemplo de sua utilização com IPv4 seria:

```
router-R13# clear bgp ipv4 unicast 10.2.255.255 soft in
```

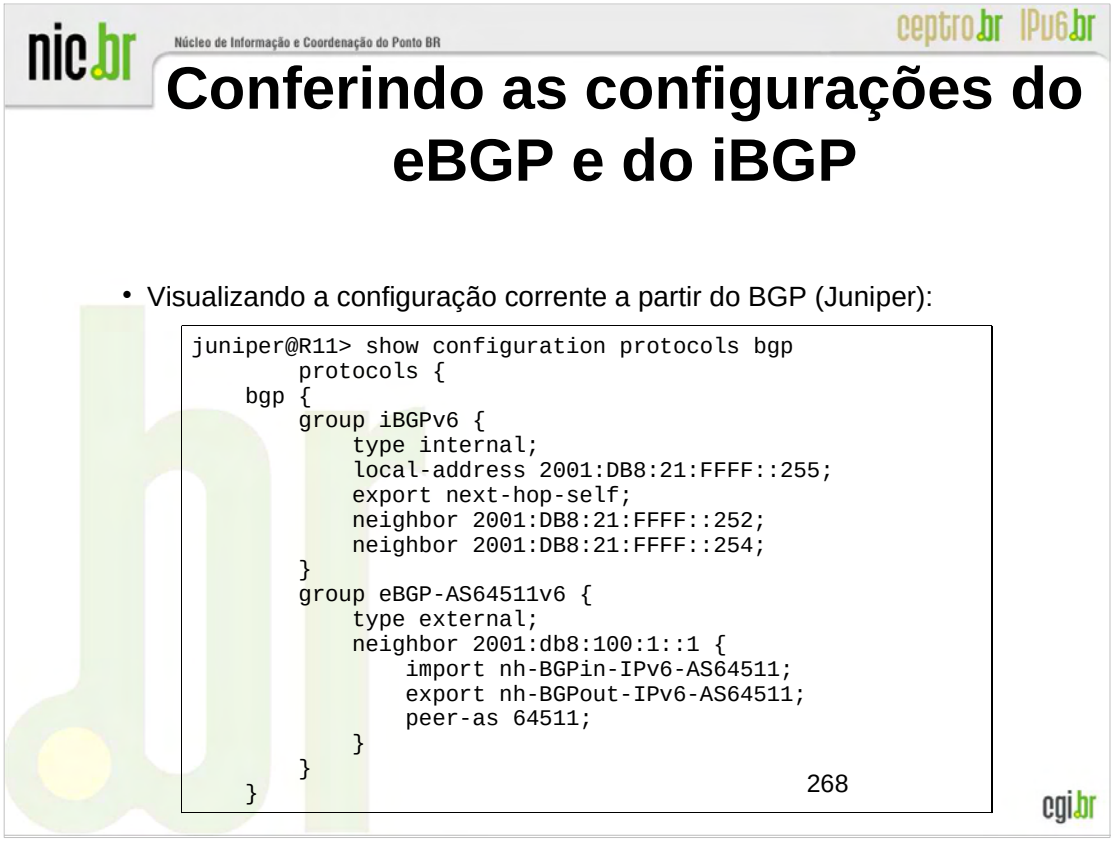
Quando os roteadores iniciam uma sessão BGP, cada roteador passa uma série de informações sobre os recursos que ele conhece, como: quais *capabilities* ele suporta. Uma delas é o `route-refresh`.

Este recurso permite recuperar as informações originais da tabela de rotas sem "derrubar" a sessão BGP e sem criar tabelas adicionais, apenas solicitando ao vizinho o reenvio da tabela de rotas.

Para saber se o roteador suporta `route-refresh` um exemplo seria:

```
show ipv6 bgp neighbor 2001:DB8:200:FFFF::255
show ip bgp neighbor 10.2.255.255
```

Com este comando também é possível ver o suporta a outras *capabilities* como o suporte a ASN de 32 bits (*New ASN Capability*).



Roteadores Juniper já trabalham com o conceito de *address-family* por padrão. (inet, inet6).

Para visualiza a configuração corrente a partir do BGP em um roteador Juniper:

show configuration protocols bgp

No primeiro grupo são apresentadas as configurações do iBGP informando os seguintes dados:

- group iBGPv6 – nome do grupo;
- type internal – indica que é iBGP;
- local-address 2001:DB8:21:FFFF::255 – endereço da interface de saída;
- export next-hop-self – propaga aos roteadores internos que o próximo salto para qualquer rota é o roteador de borda do próprio AS;
- neighbor 2001:DB8:21:FFFF::252 – indica o IP do vizinho iBGP;
- neighbor 2001:DB8:21:FFFF::254 - indica o IP do vizinho iBGP.

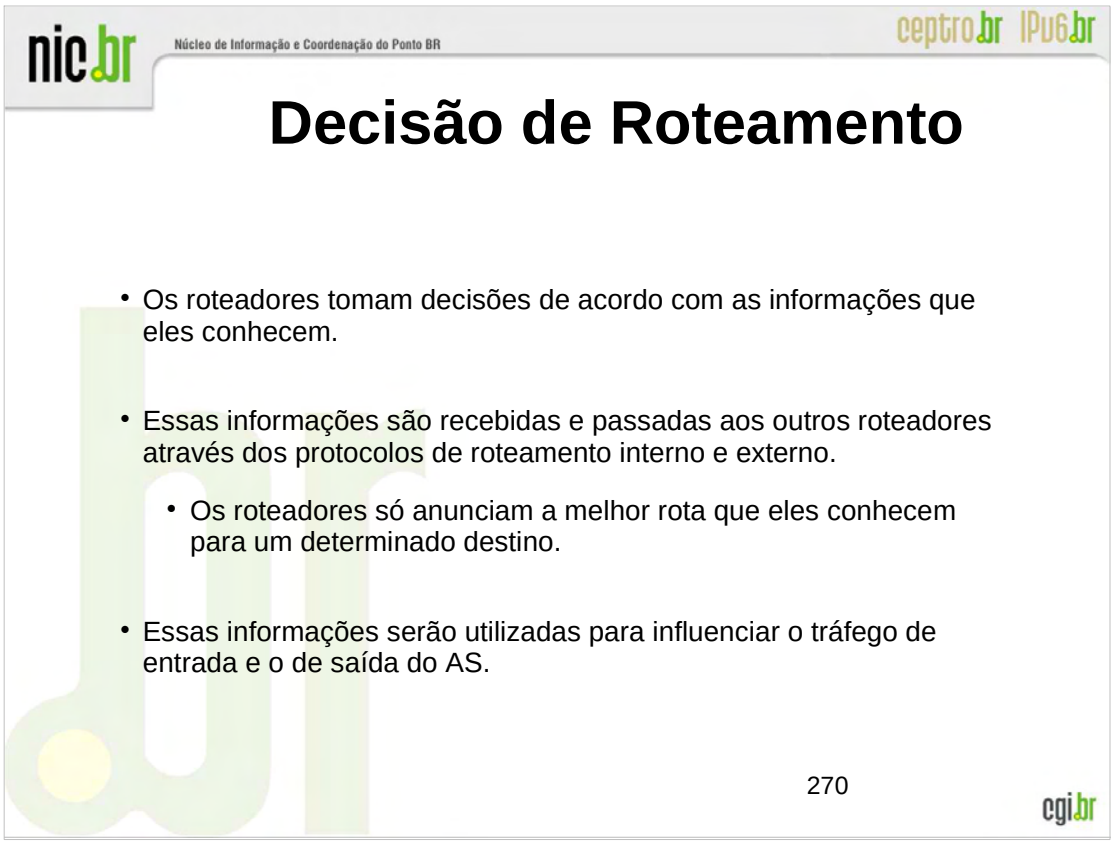
O segundo grupo traz as informações do eBGP:

- group eBGP-AS64511 – nome do grupo;
- type external – indica que é eBGP;
- neighbor 2001:db8:100:1::1 – indica o endereço do vizinho eBGP;
- import nh-BGPin-IPv6-AS64511 – política de entrada aplicada;
- export nh-BGPout-IPv6-AS64511 – política de saída aplicada;
- peer-as 64511 – ASN do vizinho.

Diferente da configuração do roteador Cisco apresentada anteriormente, no exemplo acima foi utilizado o endereço IP da interface real para se estabelecer as sessões BGP.

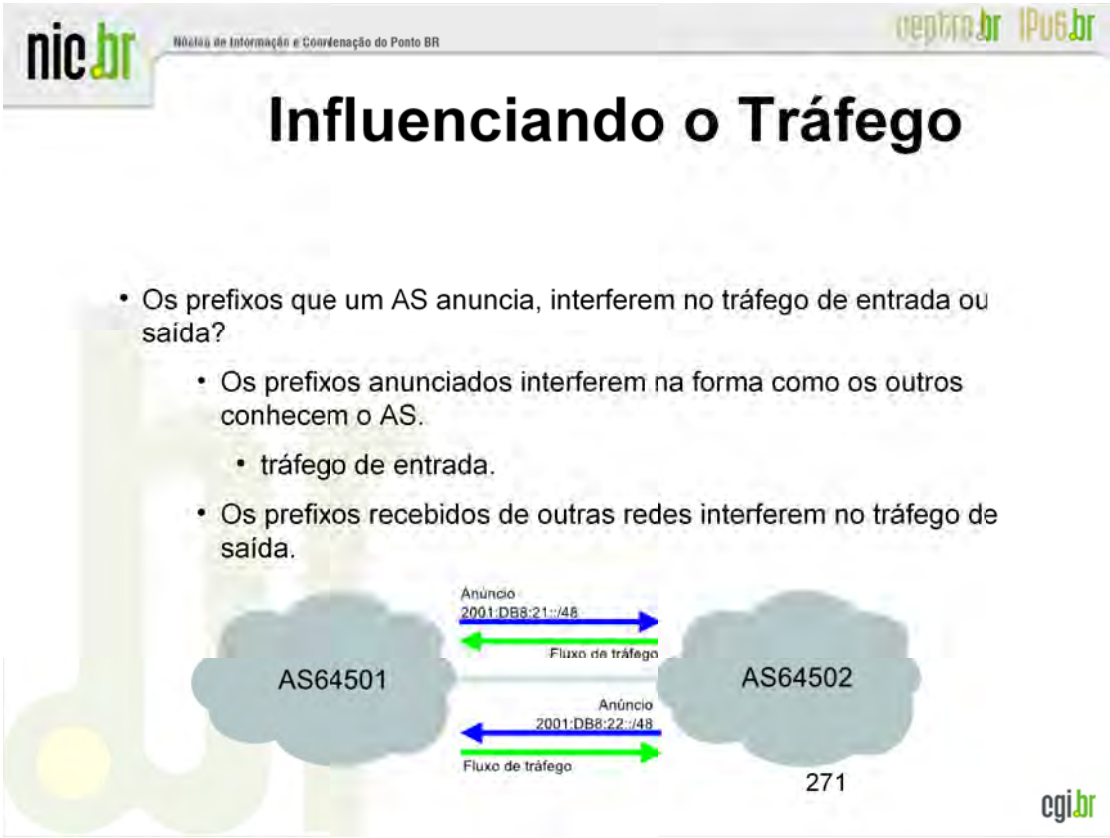
Observe a seguir um exemplo das configurações de uma sessão BGP IPv4 no Juniper:

```
juniper@R11> show configuration protocols bgp
protocols {
    bgp {
        group iBGP {
            type internal;
            local-address 172.21.15.255;
            export next-hop-self;
            neighbor 172.21.15.252;
            neighbor 172.21.15.254;
        }
        group eBGP-AS64511 {
            type external;
            neighbor 10.1.1.1 {
                import nh-BGPin-IPv4-AS64511;
                export nh-BGPout-IPv4-AS64511;
                peer-as 64511;
            }
        }
    }
```



Os roteadores tomam decisões de acordo com as informações que eles conhecem. Essas informações são recebidas e passadas aos outros roteadores através dos protocolos de roteamento interno e externo.

Ao enviar suas informações, os roteadores só anunciam a melhor rota que eles conhecem para um determinado destino. São essas informações que serão utilizadas para influenciar o tráfego de entrada e o de saída do AS.



Os prefixos anunciados interferem na forma como os outros conhecem o AS, isso é, interferem no tráfego de entrada. Do mesmo modo, os prefixos recebidos interferem no tráfego de saída.

nic.br Núcleo de Informação e Coordenação do Ponto BR

ceptro.br IPv6.br

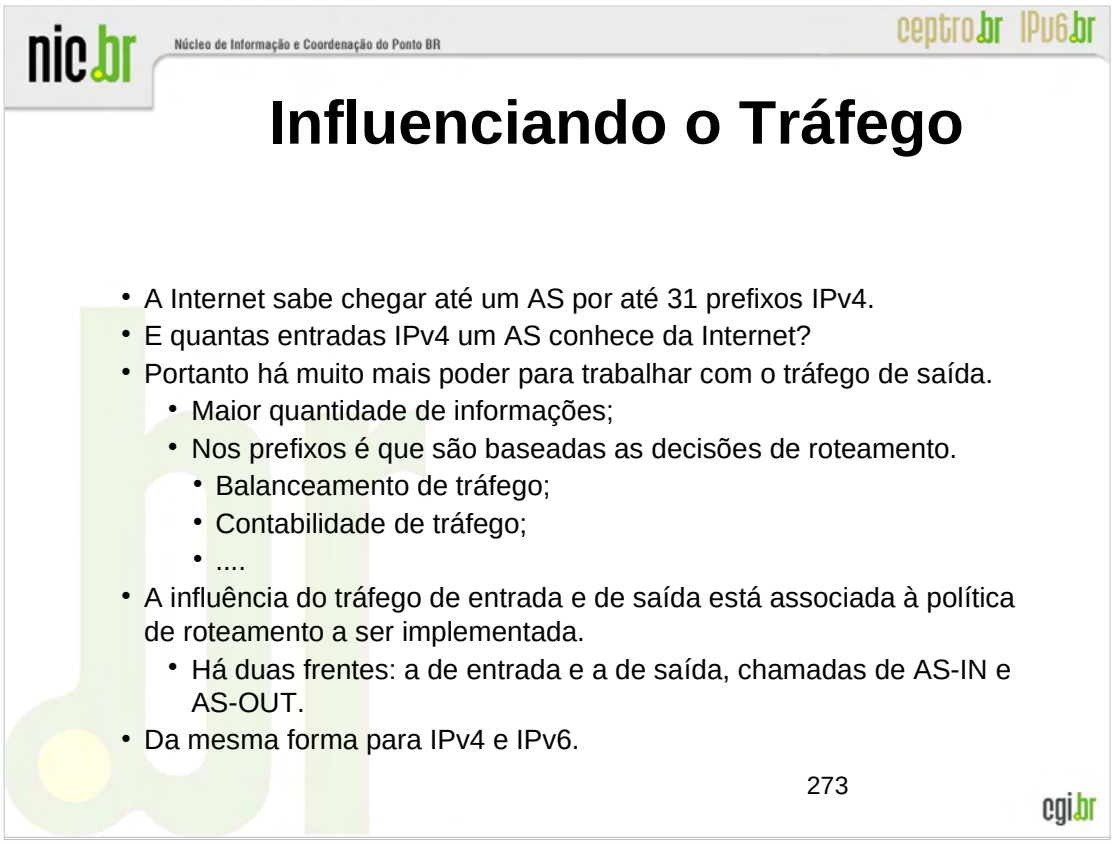
Influenciando o Tráfego

- O que é mais fácil, influenciar o tráfego de entrada ou de saída?
- Ex.:
  - Um AS possui um bloco IPv4 /20;
    - Este AS pode gerar para a Internet anúncios de prefixos até um /24, o prefixo IPv4 mais específico normalmente aceito pelas operadoras;
    - Quantos prefixos /24 podem ser gerados a partir de um /20?
    - E quantos prefixos podem ser gerados entre /20 e um /24?
    - E entre um /32 e um /48 IPv6?

cgi.br

O que é mais fácil em um AS, influenciar o tráfego de entrada ou de saída?
Analise as informações do exemplo a seguir:

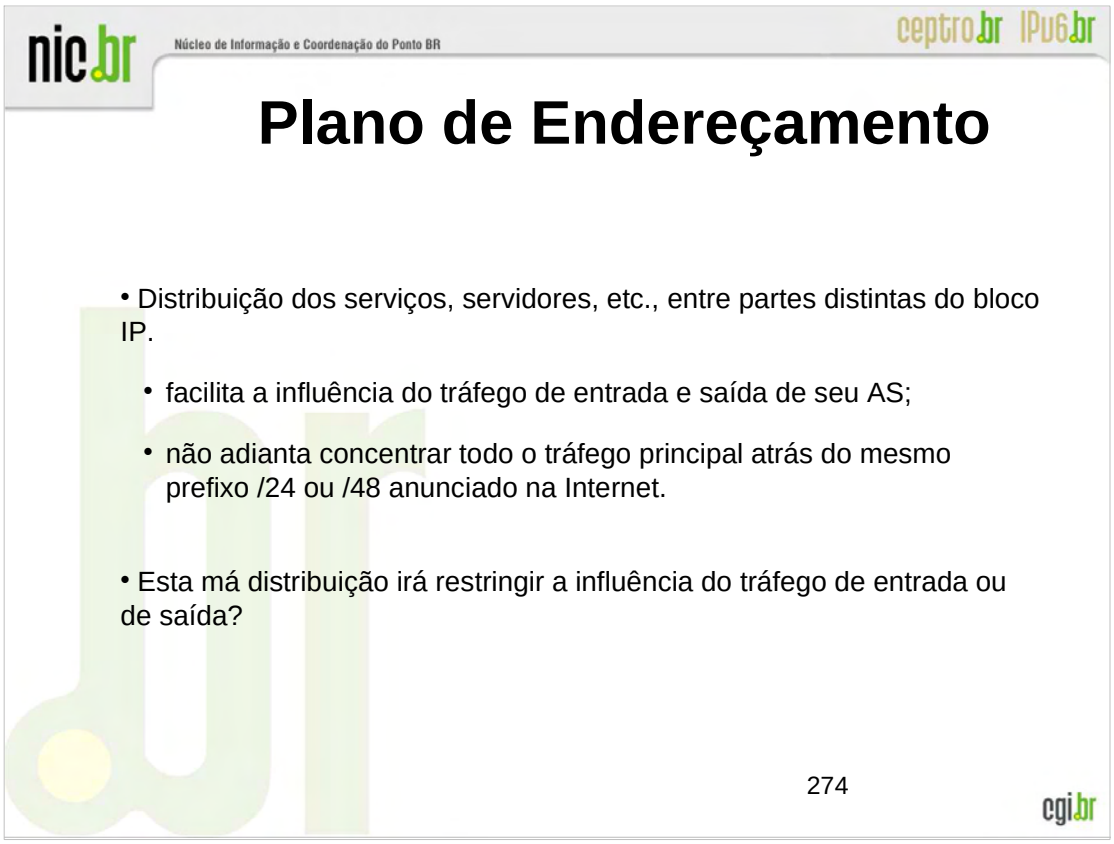
- Um AS possui um bloco IPv4 /20;
  - Este AS pode gerar para a Internet anúncios de prefixos até um /24, o prefixo IPv4 mais específico normalmente aceito pelas operadoras;
  - Com um /20 pode-se gerar 16 prefixos /24;
  - Se considerarmos a hipótese de se anunciar todos os prefixos possíveis entre o /20 até o /24, pode-se gerar um total de 31 prefixos;
  - E com um bloco de endereços IPv6, quantos prefixos podem ser gerados entre um /32 e um /48?



Deste modo, a Internet sabe chegar até o AS por até 31 prefixos IPv4 e o AS tem a opção de sair por aproximadamente 300.000 prefixos, que é o tamanho da Tabela de Roteamento Global IPv4 atualmente.

Portanto, se pensarmos na ideia de que “informação é poder”, podemos afirmar que é mais fácil influenciar o tráfego de saída, visto que há uma maior quantidade de prefixos para se trabalhar, e que são nos prefixos que são baseadas as decisões de roteamento, atuando sobre o balanceamento, contabilidade de tráfego etc.

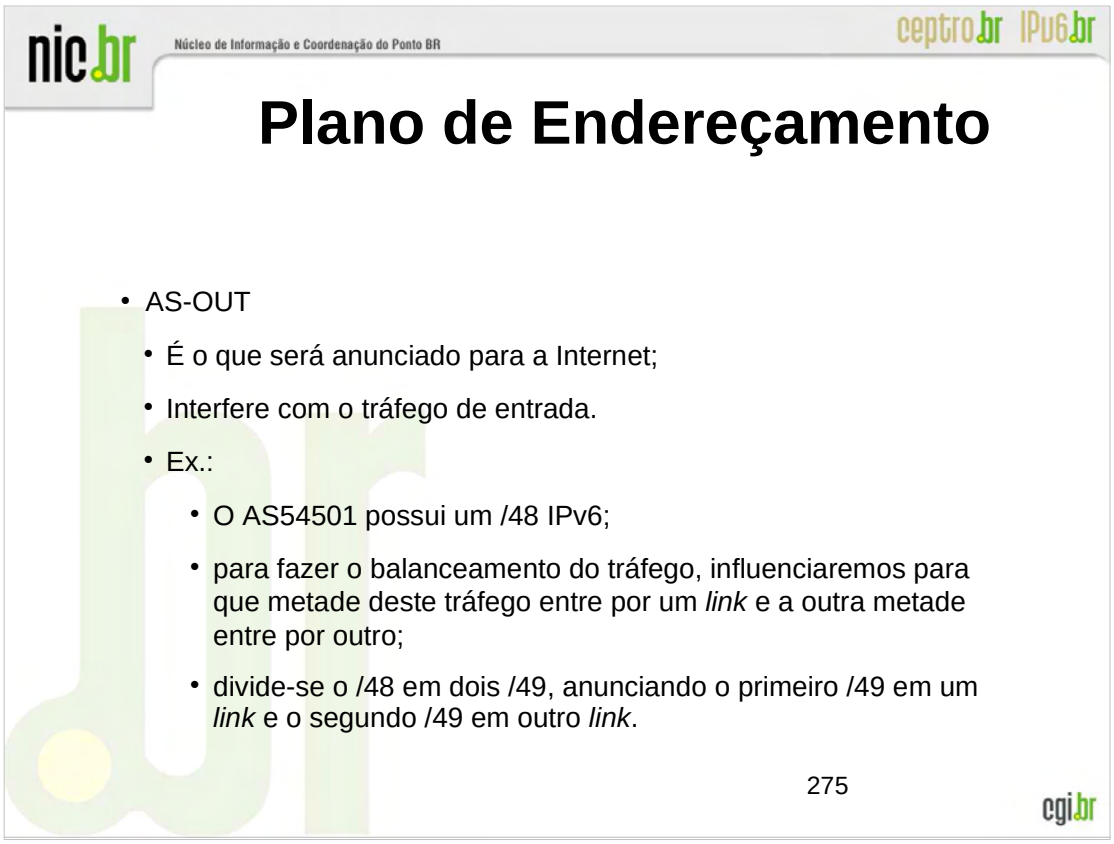
A influência dos tráfegos de entrada e de saída está associada à política de roteamento a ser implementada, sendo que há duas frentes: a de entrada e a de saída, chamadas de AS-IN e AS-OUT. Isto ocorre da mesma forma para IPv4 e IPv6.



Um ponto importante no plano de endereçamento IP, tanto IPv4 quanto IPv6, é a distribuição dos serviços, servidores, etc., entre partes distintas do bloco IP, de modo a facilitar a influência do tráfego de entrada e saída do AS.

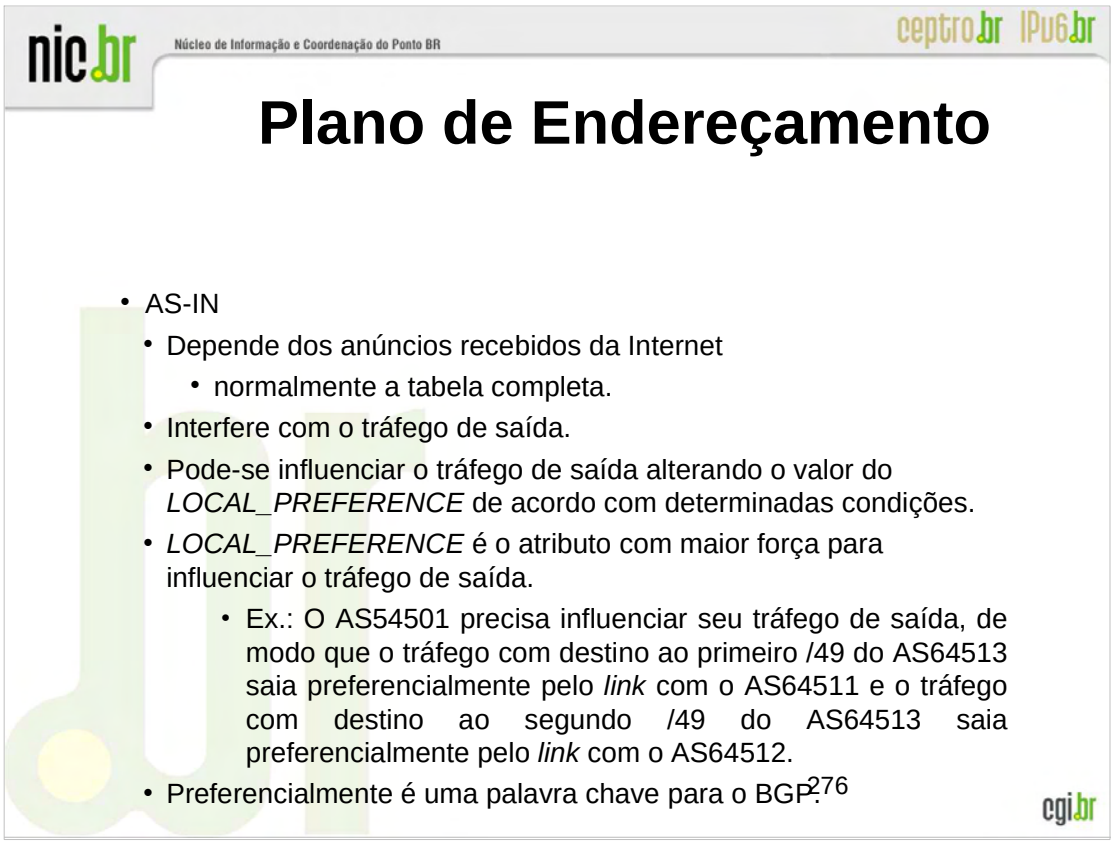
Não é recomendável posicionar todos os servidores, clientes importantes que geram a maior parte do tráfego, no mesmo /24 IPv4 ou /48 IPv6. Esta má distribuição restringirá a influencia do tráfego de entrada. Porque o tráfego de entrada é influenciado pelos anúncios gerados. Ou seja, é fundamental no plano de endereçamento ver como serão posicionados os consumidores de tráfego de entrada e de saída.

Consumidor de tráfego de entrada, que são os usuários de acesso a Internet, tem que se distribuir bem entre os prefixos anunciados.



AS-OUT é a política que trata do que será anunciado para a Internet. É o que vai interferir com o tráfego de entrada.

Por exemplo, o AS64501 possui um bloco /48 IPv6 e, para dividir e fazer o balanceamento do tráfego de entrada por dois links de acesso à Internet, deve-se influenciá-lo para que uma metade entre por um *link* e a outra metade entre por outro *link.* Para isso, divide-se o /48 em dois prefixos /49, anunciando o primeiro /49 em um *link* e o segundo /49 em outro *link*. Isto é utilizado para possibilitar um balanceamento do tráfego.

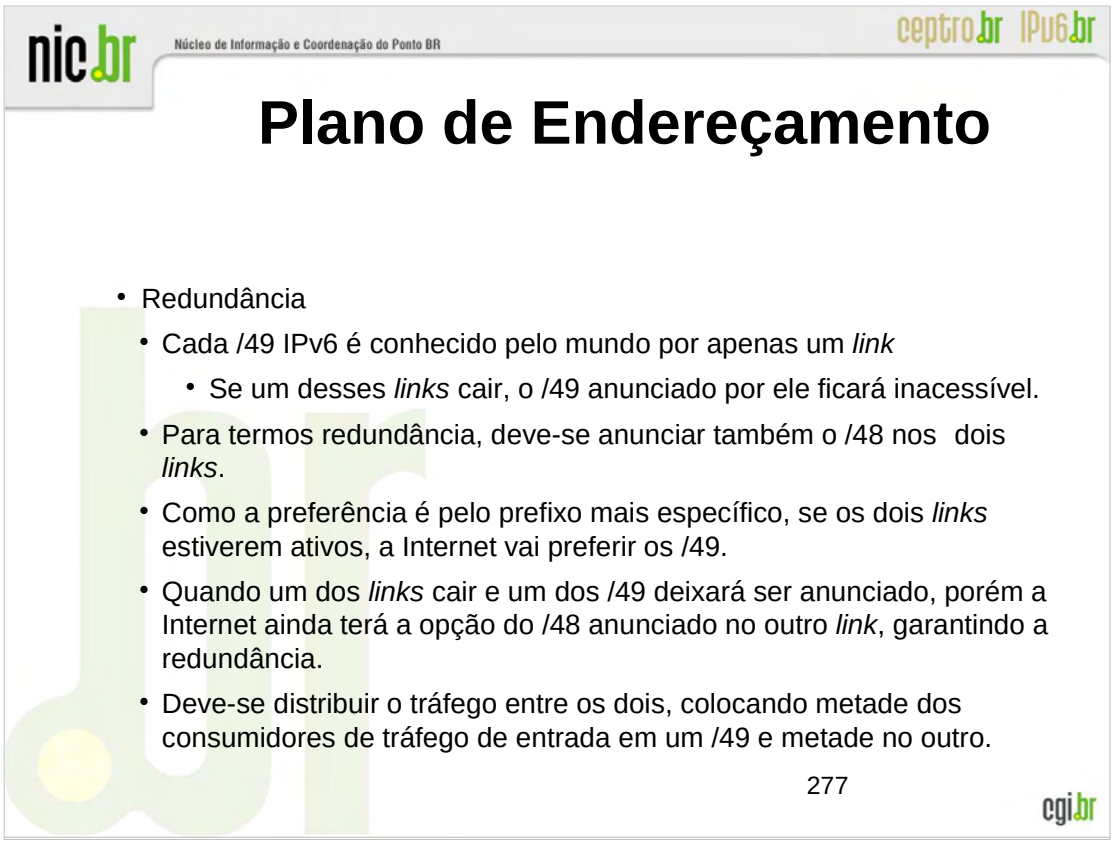


AS-IN é a política que irá interferir no tráfego de saída. Ela depende dos anúncios que são recebidos da Internet, normalmente a tabela de roteamento completa.

Para influenciar o tráfego de saída, pode-se alterar o valor do atributo *LOCAL_PREFERENCE* dos anúncios recebidos, de acordo com determinadas condições. Ele é o atributo com maior força para influenciar este tipo de tráfego.

> Ex.: O AS 54501 precisa influenciar seu tráfego de saída, de modo que o tráfego com destino ao primeiro /49 do AS64513 saia preferencialmente pelo *link* com o AS64511 e o tráfego com destino ao segundo /49 do AS64513 saia preferencialmente pelo *link* com o AS64512.

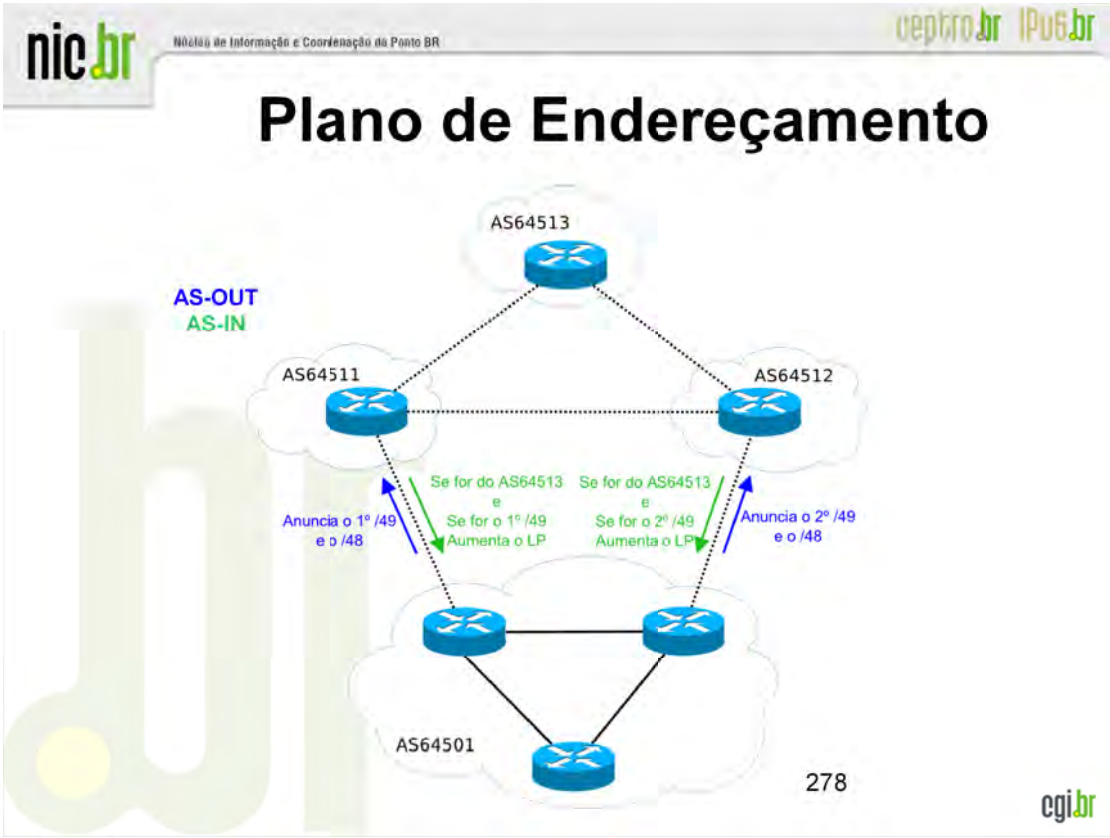
Preferencialmente é uma palavra chave para o BGP. É fundamental, com BGP, trabalhar com preferência e não com filtro. Existem várias formas de se fazer isso, mas, para que seja realmente efetiva uma configuração, especialmente quando ocorre uma queda de *link*, é importante não se descartar nada, deve-se preferenciar um caminho em relação ao outro, mas não descartar. O descarte tem um efeito imediato, mas quando não houver redundância, ele deixa de ser efetivo.



Para que haja redundância deve-se atuar da seguinte forma:

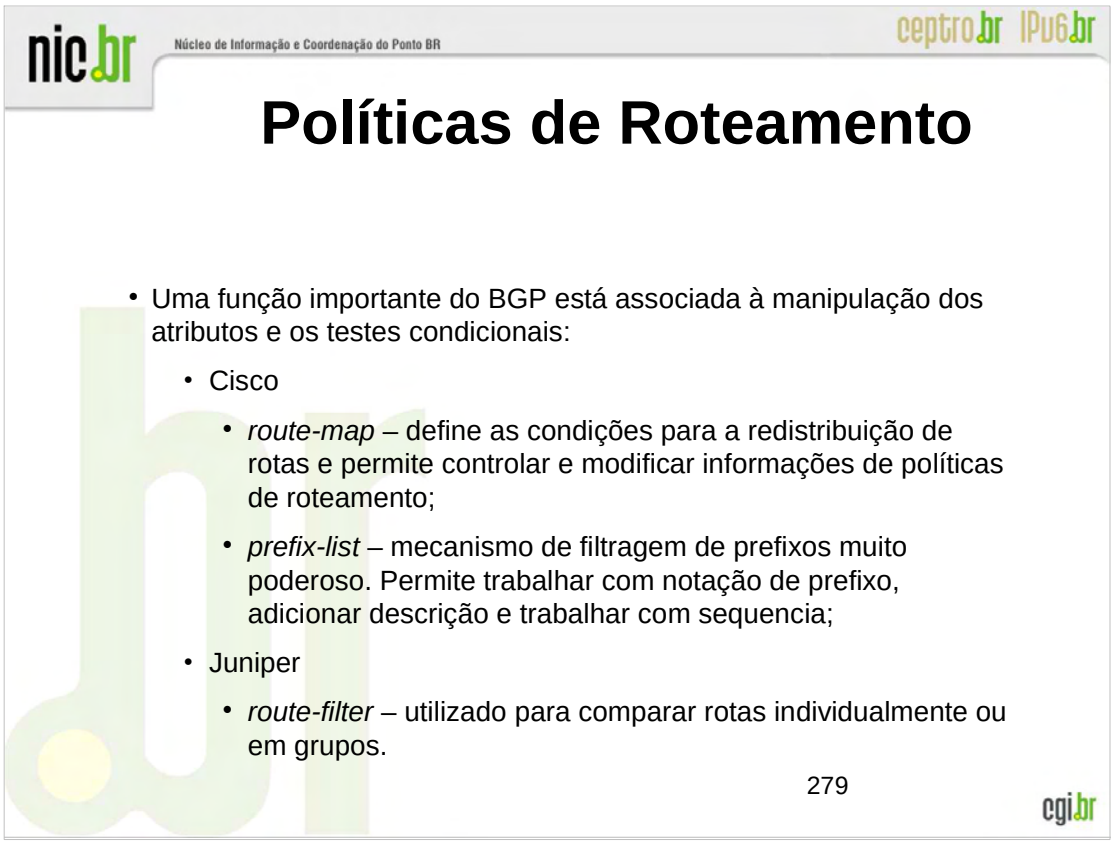
- Após dividir o bloco /48 IPv6 em dois prefixos /49, cada um deles é anunciado em um *link*, isso é, cada /49 é conhecido pelo mundo por apenas um *link*, um caminho;
- Se um desses *links* cair, o /49 anunciado por ele ficará inacessível;
- Para termos redundância, deve-se anunciar também o prefixo /48 nos dois *links;*
- Com isso, a Internet conhecerá o /48 pelos dois caminhos;
- Como a preferência é pelo prefixo mais específico, se os dois *links* estiverem ativos, a Internet vai preferir os /49 sempre, ou seja o balanceamento estará em operação;
- Quando um desses *links* cair, um dos prefixos /49 deixará de ser anunciado. No entanto, esse /49 está contido no /48, ou seja, mesmo com um dos *links* desativado, a Internet ainda terá a opção do /48 anunciado no outro *link*. Garantindo a redundância.

Para realmente distribuir o tráfego entre os dois, deve-se colocar metade dos consumidores de tráfego de entrada em um /49 e metade no outro. Isso faz parte do planejamento.



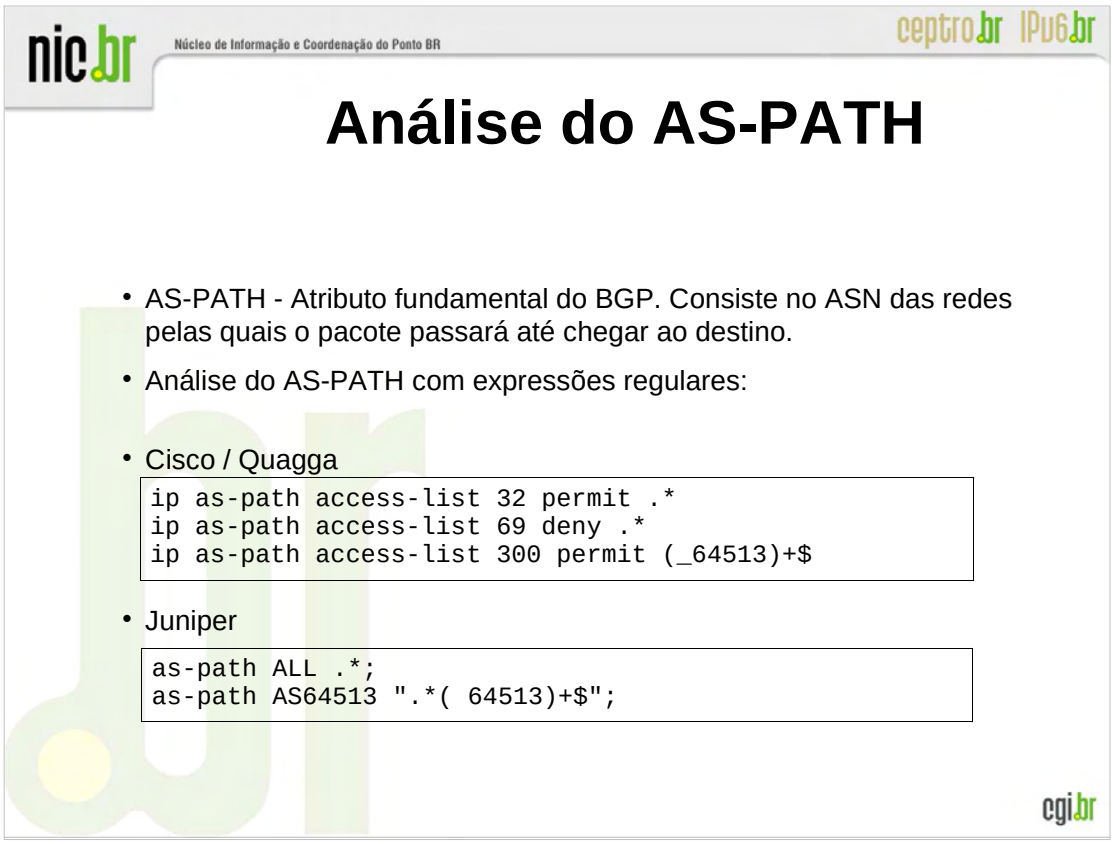
Neste diagrama podemos analisar exemplos das políticas de roteamento apresentadas até o momento.

- AS-OUT – Utiliza os anúncios enviados para interferir o tráfego de entrada;
- AS-IN – Utiliza os anúncios recebidos para influenciar o tráfego de saída.



Uma função importante do BGP reside na manipulação dos atributos e nos testes condicionais. Para tratar esses aspectos temos as seguintes funcionalidades:

- *route-map* (Cisco e Quagga) – define as condições para a redistribuição de rotas e permite controlar e modificar informações de políticas de roteamento;
- *prefix-list* (Cisco e Quagga) – mecanismo de filtragem de prefixos com muitos recursos. Permite trabalhar com notação de prefixo, adicionar descrição e trabalhar com sequencia, apresentando deste modo, uma vantagem em relação a utilização da função *distribute-list*, que por ser baseada em ACLs facilita a filtragem de pacotes, porém, torna-se de difícil gerenciamento;
- *route-filter* (Juniper) – utilizado para comparar rotas individualmente ou em grupos.



O BGP é um protocolo usado na comunicação entre os ASs. Por isso, o AS-PATH torna-se um atributo fundamental do BGP. Ele consiste no ASN das redes pelas quais o pacote passará até chegar ao destino.

Os roteadores Cisco, Quaga e Juniper oferecem comandos que permitem a análise do AS-PATH com expressões regulares. Em nosso exemplo, temos os seguintes termos nas expressões regulares:

- O caractere “ponto” significa 'qualquer elemento'
- O caractere “asterisco” significa 'zero ou varias ocorrências'
- O caractere “$” significa 'fim de linha'
- O caractere “+” significa 'uma ou mais ocorrências'

ip as-path access-list 32 permit .* (Cisco / Quagga)
as-path ALL .*; (Juniper)

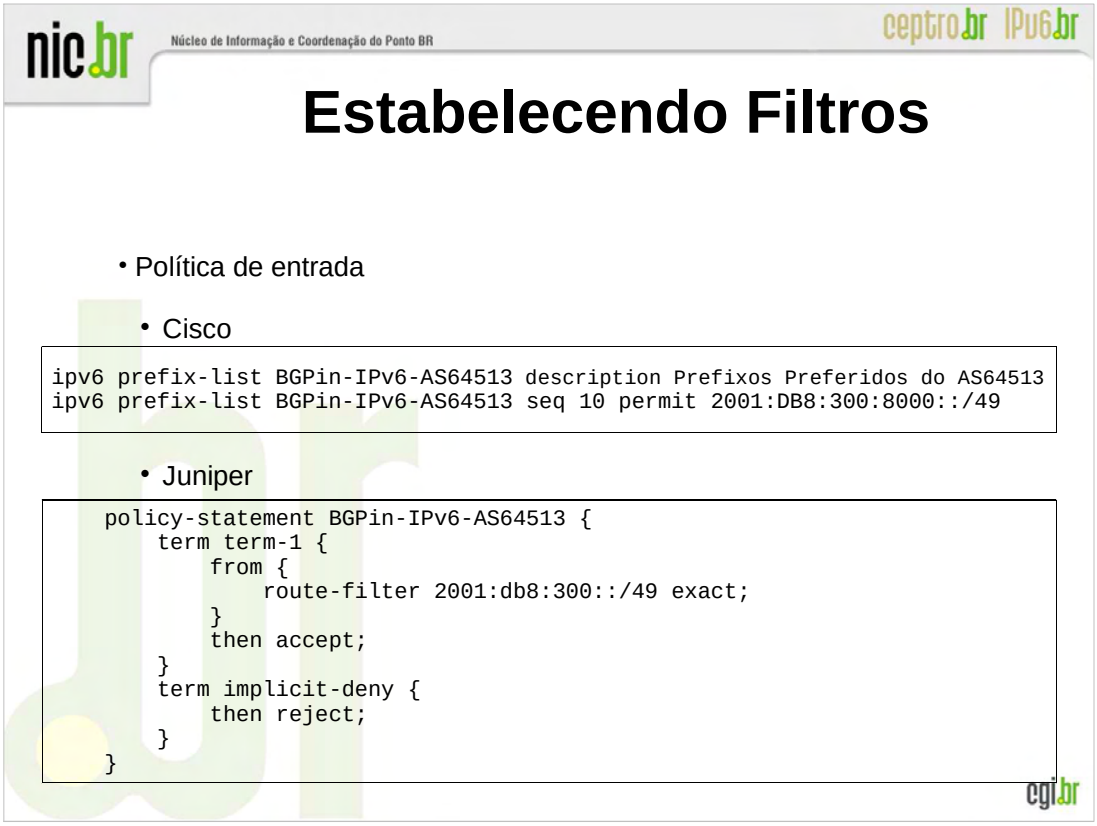
- As linhas acima indicam que qualquer AS-PATH será permitido.

ip as-path access-list 69 deny .* (Cisco / Quagga)

- Esta linha indica que qualquer bloco será negado.

ip as-path access-list 300 permit (_64513)+$ (Cisco / Quagga)
as-path AS64513 ".*(64513)+$"; (Juniper)

- Essas linhas indicam que todos os prefixos originados no AS 64513 serão permitidos. A permissão de uma ou mais ocorrências é para garantir *AS Path prepends*, ou seja, repetições de um mesmo ASN em sequencia ao longo do AS-PATH (no caso, o ASN 64513).



Este exemplo mostra através de um *prefix-list* em um roteador Cisco e um *route-filter* em um roteador Juniper, o estabelecimento de uma política de entrada.

No exemplo do roteador Cisco, o *prefix-list* identifica o segunda /49 do AS 64513 que será recebido.

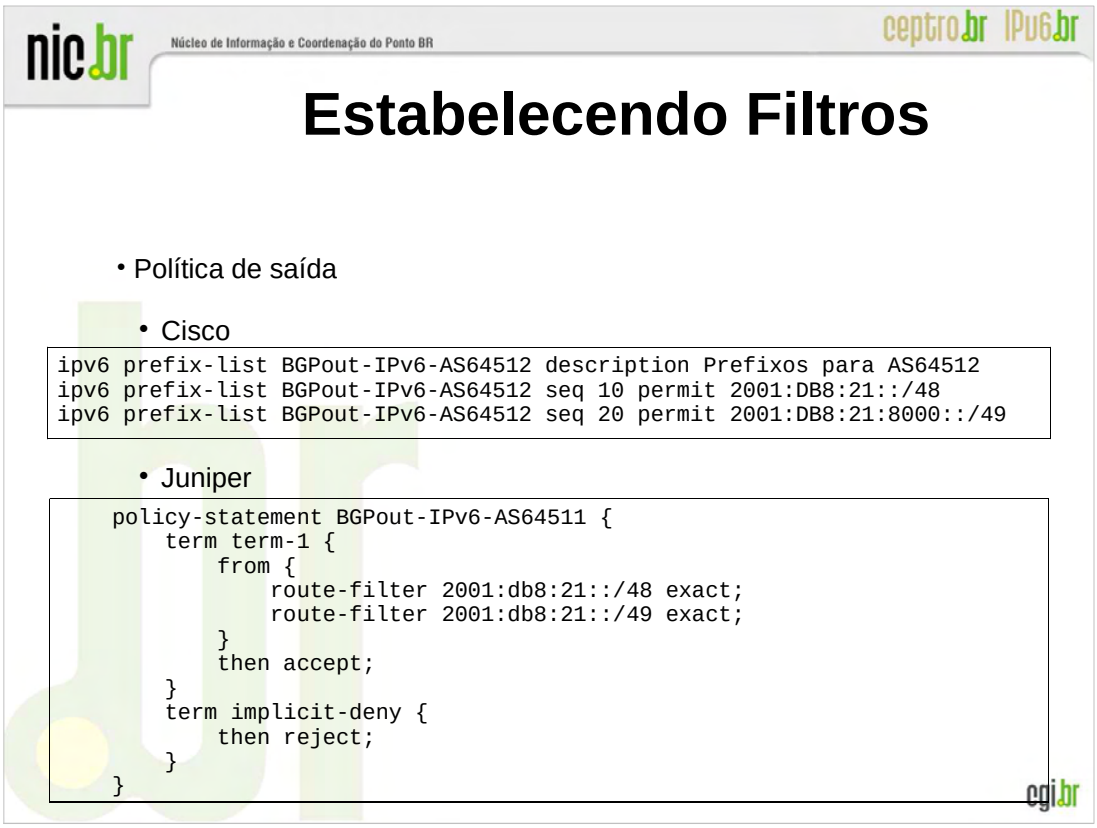
```
ipv6 prefix-list BGPin-IPv6-AS64513 seq 10 permit 2001:DB8:300:8000::/49
```

No exemplo do roteador Juniper, o *route-filter* identifica o primeiro /49 do AS 64513 que será recebido.

```
route-filter 2001:db8:300::/49 exact;
```

Um exemplo da implantação dessas políticas em uma configuração IPv4 seria:

- Cisco: `ip prefix-list BGPin-IPv6-AS64513 seq 10 permit 10.3.128.0/17`
- Juniper: `route-filter 10.3.0.0/17 exact;`



Este exemplo mostra através de *prefix-list* em um roteador Cisco e de *route-filter* em um roteador Juniper, o estabelecimento de uma política de saída.

No exemplo do roteador Cisco, os *prefix-list* indicam os prefixos que serão anunciados para o AS64512. Neste caso serão enviados o prefixo /48 IPv6 e segundo /49.

No exemplo do roteador Juniper, os *route-filter* indicam que serão anunciados para o AS 64511 o prefixo /48 IPv6 e o primeiro /49.

Um exemplo dessa política de saída em uma configuração IPv4 poderia ser:

- Cisco

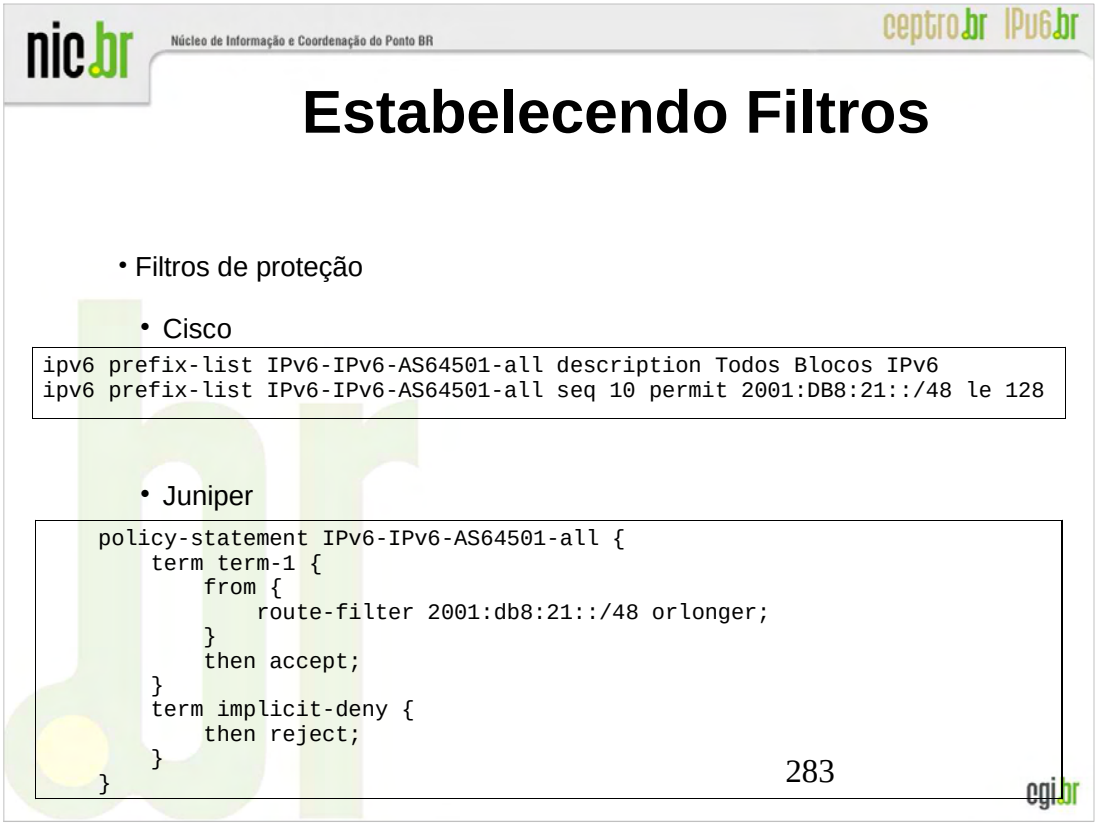
ip prefix-list BGPout-IPv4-AS64512 seq 10 permit 172.21.0.0/20

ip prefix-list BGPout-IPv4-AS64512 seq 20 permit 172.21.8.0/21

- Juniper

route-filter 172.21.0.0/20 exact;

route-filter 172.21.0.0/21 exact;



Outra política de proteção importante é a que impede que o AS receba anúncios de seus próprios prefixos.

Para IPv6 poderíamos ter algo similar a:

- Cisco/Quagga

```
ipv6 prefix-list IPv6-IPv6-AS64501-all seq 10 permit 2001:DB8:21::/48 le 128
```

- Juniper

```
route-filter 2001:db8:21::/48 orlonger;
```

Para IPv4 poderíamos ter algo similar a:

– Cisco / Quagga

```
ip prefix-list IPv4-AS64501-all seq 10 permit 172.21.0.0/20 le 32
```

– Juniper

```
route-filter 172.21.0.0/20 orlonger;
```

As regras exemplificadas acima indicam todos os prefixos possíveis dentro de um bloco /48 IPv6 ou /20 IPv4. Elas serão utilizadas na política de entrada, para dizer: “não aceito nenhum prefixo que seja meu”. Isso ajuda a evitar problemas como sequestro de prefixos.

Por padrão, o roteador rejeita todos os prefixos que tenha o seu ASN, para evitar *looping*. Porém, nada impede que outro AS na Internet, por intensão ou erro, gere anúncios do outro prefixo, até um mais específico. Se não houver proteção, o roteador vai aceitar e encaminhar todo o tráfego interno para fora.

Um exemplo deste tipo de ocorrência foio sequestro do prefixo do YouTube. Por determinação do Governo Paquistanês, o tráfego do YouTube deveria ser bloqueado para evitar o acesso ao trailer de um filme anti-Islamico. Para cumprir essa ordem, a operadora Pakistan Telecom gerou o anúncio de um prefixo mais específico do que o utilizado pelo YouTube, com o intuito de direcionar todos os acessos a ele para uma página que dizia “YouTube was blocked”.

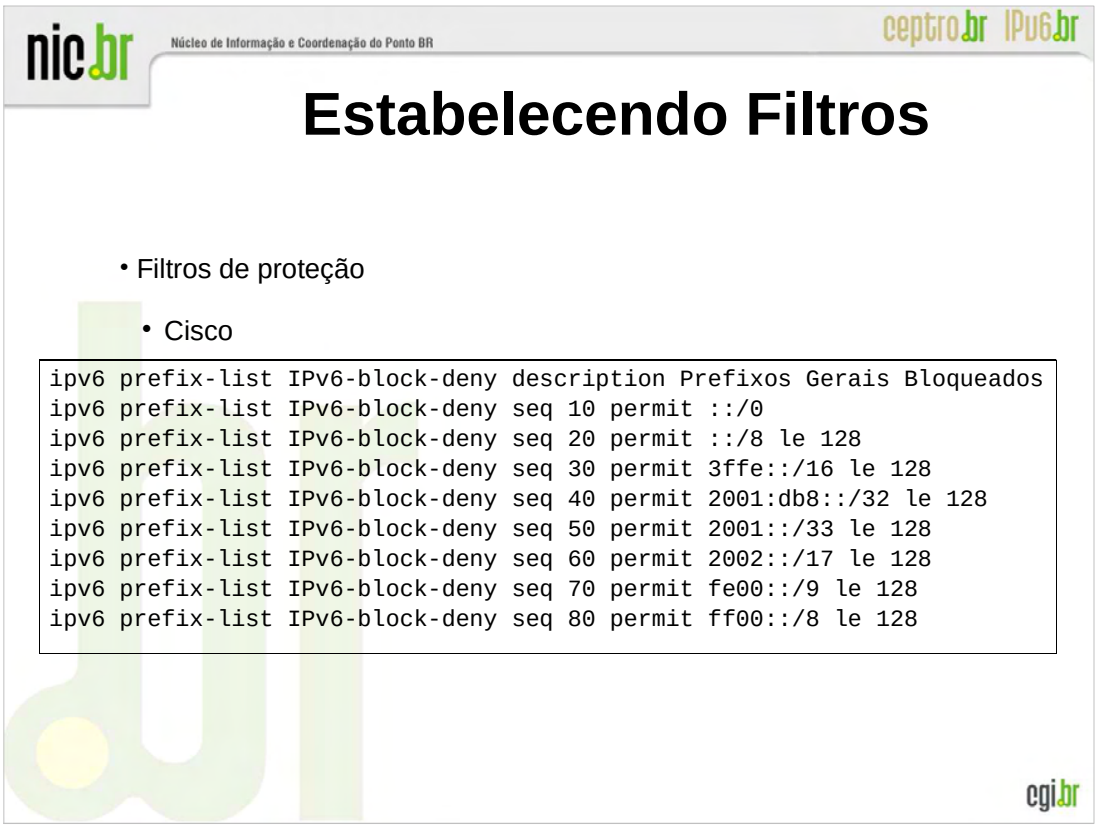
No entanto, a operadora anunciou essa nova rota a seu *upstream provider* (primeiro erro), que, além de não verificar a nova rota (segundo erro) a propagou por toda a Internet (terceiro erro). Com isso, todo o tráfego do YouTube passou a ser direcionado para o Paquistão e ser descartado.

Esse foi apenas o caso mais famoso, mas sequestros de blocos IP ocorrem diariamente, intencionalmente ou não. Isso ocorre porque toda a estrutura da Internet e o funcionamento do BGP foram definidos baseados em uma relação de confiança. Essa “inocência” ainda é presente e é o que mantem toda a estrutura atual.

Existem discussões sobre modos de verificar se o AS que está anunciando um determinado prefixo tem autoridade para faze-lo, similar ao ao que o DNSSec faz.

Mais informações:

- http://www.ietf.org/dyn/wg/charter/idr-charter.html
- http://www.ietf.org/dyn/wg/charter/sidr-charter.html
- http://www.youtube.com/watch?v=IzLPKuAOe50
- http://www.wired.com/threatlevel/2008/02/pakistans-accid/



Também é possível adicionar *prefix-list* de proteção. Observe o seguinte exemplo em um roteador Cisco:

```
ipv6 prefix-list IPv6-block-deny seq 10 permit ::/0
ipv6 prefix-list IPv6-block-deny seq 20 permit ::/8 le 128
ipv6 prefix-list IPv6-block-deny seq 30 permit 3ffe::/16 le 128
ipv6 prefix-list IPv6-block-deny seq 40 permit 2001:db8::/32 le 128
ipv6 prefix-list IPv6-block-deny seq 50 permit 2001::/33 le 128
ipv6 prefix-list IPv6-block-deny seq 60 permit 2002::/17 le 128
ipv6 prefix-list IPv6-block-deny seq 70 permit fe00::/9 le 128
ipv6 prefix-list IPv6-block-deny seq 80 permit ff00::/8 le 128
```

Estes *prefix-list* verificam respectivamente:

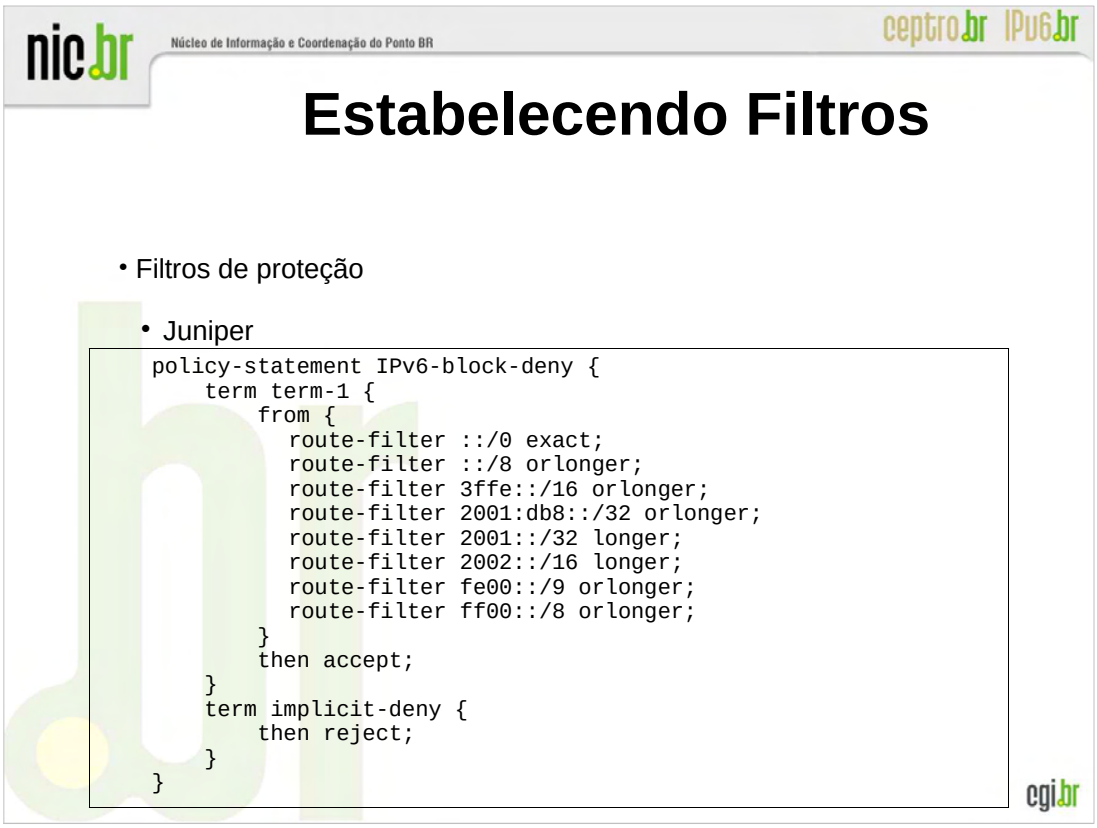
- A rota *default;*
- O primeiro prefixo /8;
- Endereçox da rede de testes 6bone;
- Endereços para documentação;
- Endereços dos túneis Teredo;
- Endereços dos túneis 6to4;
- Endereços *Link-local* (RFC 5735);
- Endereços Multicast.

Um exemplo da aplicação de um *prefix-list* de proteção para IPv4 poderia ser:

```
ip prefix-list IPv4-block-deny seq 10 permit 0.0.0.0/0
ip prefix-list IPv4-block-deny seq 20 permit 0.0.0.0/8
ip prefix-list IPv4-block-deny seq 30 permit 127.0.0.0/8
ip prefix-list IPv4-block-deny seq 40 permit 169.254.0.0/16
ip prefix-list IPv4-block-deny seq 50 permit 192.0.2.0/24
ip prefix-list IPv4-block-deny seq 60 permit 10.0.0.0/8
ip prefix-list IPv4-block-deny seq 60 permit 172.16.0.0/12
ip prefix-list IPv4-block-deny seq 80 permit 192.168.0.0/16
```

Estas *prefix-list* verificam respectivamente:

- A rota *default;*
- O primeiro prefixo /8;
- O endereço de *loopback*;
- Endereços *Link-local* (RFC 5735);
- Endereços da TEST-NET-1 (RFC 5737);
- Endereços privados (RFC 1918).



Também é possível adicionar *route-fiter* de proteção. Observe o seguinte exemplo em um roteador Juniper:

```
route-filter ::/0 exact;
route-filter ::/8 orlonger;
route-filter 3ffe::/16 orlonger;
route-filter 2001:db8::/32 orlonger;
route-filter 2001::/32 longer;
route-filter 2002::/16 longer;
route-filter fe00::/9 orlonger;
route-filter ff00::/8 orlonger;
```

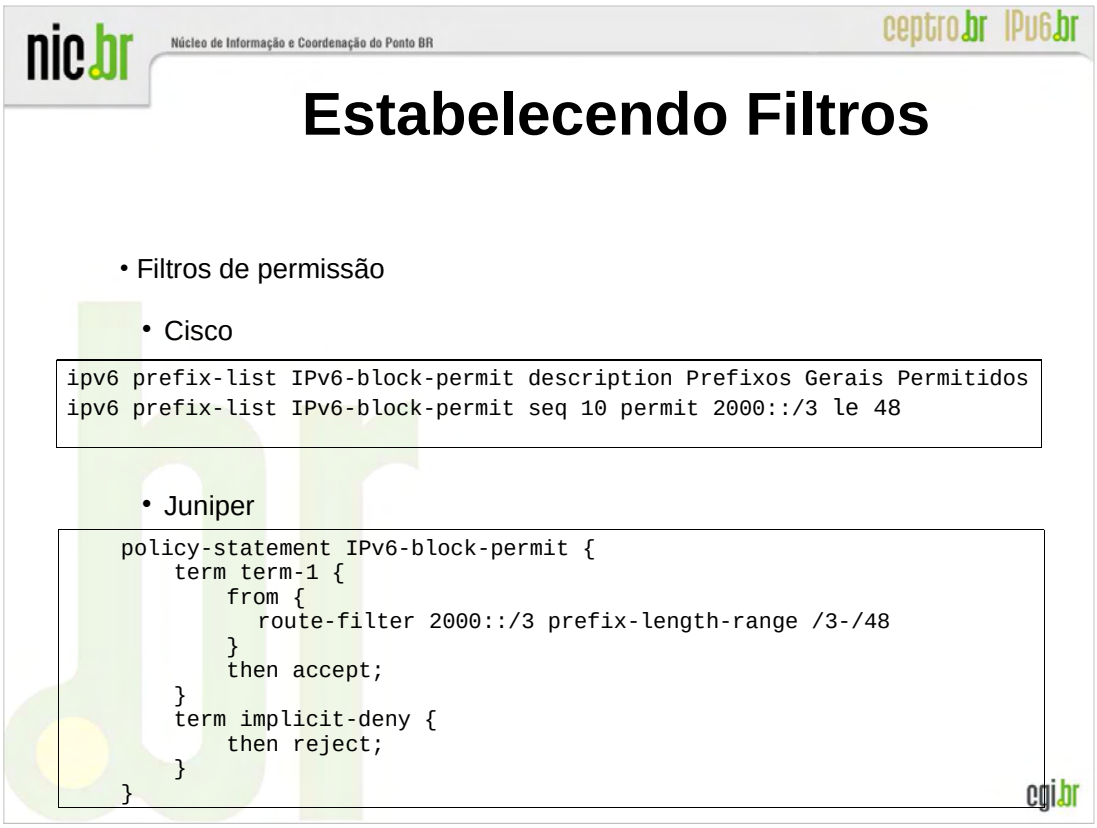
Estes *route-fiters* realizam as mesmas verificações apresentadas no exemplo anterior de roteadores Cisco.

Um exemplo da aplicação de um *prefix-list* de proteção para IPv4 poderia ser:

```
route-filter 0.0.0.0/0 exact;
route-filter 0.0.0.0/8 exact;
route-filter 127.0.0.0/8 exact;
route-filter 169.254.0.0/16 exact;
route-filter 192.0.2.0/24 exact;
route-filter 10.0.0.0/8 exact;
route-filter 172.16.0.0/12 exact;
route-filter 192.168.0.0/16 exact;
```

Mias informações:

- http://www.space.net/~gert/RIPE/ipv6-filters.html



Como vimos no módulo de Segurança IPv6, o modo de filtrarmos os endereços *bogons* no IPv6 é diferente da forma feita com IPv4. No IPv6 é mais fácil liberar as faixas de endereços alocados e bloquear o restante.

Os exemplos de `prefix-list` e `policy-statement` apresentados a seguir mostram uma forma flexível de liberar as faixas de endereços IPv6 disponíveis para alocação. Uma forma mais restrita de fazer esse mesmo tipo de filtro, é permitir uma a uma as faixas de endereços já atribuídas aos RIRs. Essas faixas podem ser obtidas em:

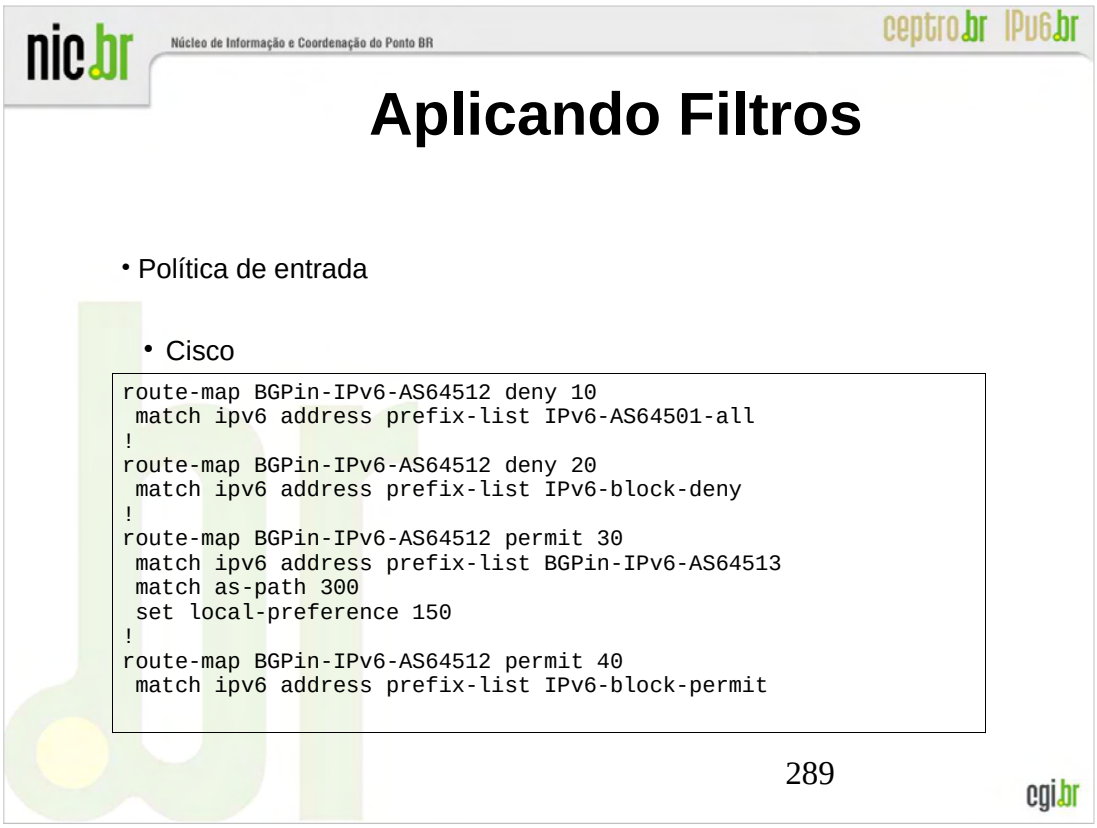
- http://www.iana.org/assignments/ipv6-unicast-address-assignments/ipv6-unicast-address-assignments.xhtml

Observe um exemplo de um roteador Cisco:

```
ipv6 prefix-list IPv6-block-permit description Prefixos Gerais Permitidos
ipv6 prefix-list IPv6-block-permit seq 10 permit 2000::/3 le 48
```

Observe um exemplo de um roteador Juniper:

```
policy-statement IPv6-block-permit {
    term term-1 {
        from {
               route-filter 2000::/3 prefix-length-range /3-/48
        }
        then accept;
    }
    term implicit-deny {
        then reject;
    }
}
```



Após serem estabelecidas as condições dos filtros via *prefix-list* nos roteadores Cisco, deve-se aplicá-las através de *route-maps*.

Ex.:

```
route-map BGPin-IPv6-AS64512 deny 10
 match ipv6 address prefix-list IPv6-AS64501-all
!
route-map BGPin-IPv6-AS64512 deny 20
 match ipv6 address prefix-list IPv6-block-deny
!
route-map BGPin-IPv6-AS64512 permit 30
 match ipv6 address prefix-list BGPin-IPv6-AS64513
 match as-path 300
 set local-preference 150
!
route-map BGPin-IPv6-AS64512 permit 40
 match ipv6 address prefix-list IPv6-block-permit
```

Esse é o *route-map* de entrada. Ele possui um nome para identificação de sua função, pois é possível ter vários *route-maps*. Neste exemplo, ele indica como tratar o que será recebido do AS 64512. Há 4 regras, a 10, 20, 30 e a 40.

Os *route-maps* trabalham com testes lógicos de “e” e “ou”, onde cada prefixo passa pelo *route-map*, e se a comparação entre a regra estabelecida e o prefixo coincidirem, ele é processado e a comparação é encerrada. Se a comparação não coincidir, a regra seguinte será analisada. Ou seja, cada regra é um teste “ou”.

A terceira regra possui dois testes de comparação. Quando há dois ou mais teste na mesma regra, é equivalente a condição “e”. Com isso, a terceira regra diz que tem que coincidir na primeira e na segunda linha.

O modo de funcionamento dos *route-map* é independente se a configuração é de uma sessão IPv4 ou IPv6.

A primeira regra descarta todos os prefixos que coincidirem com o que foi estabelecido no `prefix-list IPv6-AS64501-all`, que representa todos os prefixos do próprio AS. É a regra que protege do sequestro de blocos.

A segunda regra descarta os blocos de uso privado especificados no `prefix-list IPv6-block-deny`.

A terceira regra é onde será alterado o *LOCAL_PREFERENCE*. Ela verifica de qual AS vem o prefixo (`as-path 300`) e se é o prefixo esperado (`prefix-list BGPin-IPv6-AS64513`). Se coincidir com as duas condições, o valor do *LOCAL_PREFERENCE* é aumentado para 150. O valor padrão do *LOCAL_PREFERENCE* é 100 e, quanto maior seu valor, maior a preferência.

A última regra, especificada no `prefix-list IPv6-block-permit,` permite o recebimento de anúncios de prefixos de dentro da faixa reservada pela IANA para alocação **2000::/3**. Ela permite anúncios de prefixos até um /48, tamanho normalmente aceito pelas operadoras.

Qualquer outro anúncio recebido que não seja validado pelos *route-maps* serão descartados, pois os roteadores Cisco possuem um "*deny*" implícito como última regra.

Aplicando Filtros

- Política de entrada
  - Juniper

```
policy-statement nh-BGPin-IPv6-AS64511 {
    term term-1 {
        from policy IPv6-AS64501-all;
        then reject;
    }
    term term-2 {
        from policy IPv6-block-deny;
        then reject;
    }
    term term-3 {
        from {
            as-path AS64513;
            policy BGPin-IPv6-AS64513;
        }
        then {
            local-preference 150;
            next term;
        }
    }
    term term-4 {
        from policy IPv6-block-permit;
        then accept;
    }
 term implicit-deny {
        then reject;
    }
}
```


A política implementada no roteador Juniper é similar a do roteador Cisco apresentada anteriormente. A principal diferença é que as políticas são aplicadas aos anúncios recebidos do AS 64511.

Ex.:

```
policy-statement nh-BGPin-IPv6-AS64511 {
    term term-1 {
        from policy IPv6-AS64501-all;
        then reject;
    }
    term term-2 {
        from policy IPv6-block-deny;
        then reject;
    }
    term term-3 {
        from {
            as-path AS64513;
            policy BGPin-IPv6-AS64513;
        }
        then {
            local-preference 150;
            next term;
        }
    }
    term term-4 {
        from policy IPv6-block-permit;
        then accept;
    }
 term implicit-deny {
        then reject;
    }
}
```

nic.br Núcleo de Informação e Coordenação do Ponto BR — ceptro.br IPv6.br

Aplicando Filtros

- Política de saída
  - Cisco

```
route-map BGPout-IPv6-AS64512 permit 10
 match ipv6 address prefix-list BGPout-IPv6-AS64512
```

  - Juniper

```
policy-statement nh-BGPout-IPv6-AS64511 {
    term term-1 {
        from policy BGPout-IPv6-AS64511;
        then accept;
    }
    term implicit-deny {
        then reject;
    }
}
```


Do mesmo modo como foram aplicadas as políticas de entrada, podemos aplicar as políticas de saída através de um `route-map` e de um `policy-statement` no Juniper.

nic.br Núcleo de Informação e Coordenação do Ponto BR

cetro.br IPv6.br

Verificando a Vizinhança BGP

- Mostrando todos os vizinhos BGP IPv4:
  - `show ip bgp summary` (Cisco / Quagga)
  - `show bgp summary` (Juniper)
- Mostrando todos os vizinhos BGP de ambas as famílias:
  - `show bgp ipv4 unicast summary` (Cisco / Quagga)
  - `show bgp ipv6 unicast summary` (Cisco / Quagga)
  - `show bgp all summary` (Cisco / Quagga)

cgi.br

Os comandos a seguir listam uma série de informações de estado da tabela BGP:

Mostrando todos os vizinhos BGP IPv4:

show ip bgp summary (Cisco / Quagga)

show bgp summary (Juniper)

Mostrando todos os vizinhos BGP de ambas as famílias:

show bgp ipv4 unicast summary (Cisco / Quagga)

show bgp ipv6 unicast summary (Cisco / Quagga)

show bgp all summary (Cisco / Quagga)

A partir dessas informações pode-se detectar uma série de problemas da sessão BGP.

nic.br Núcleo de Informação e Coordenação do Ponto BR

cetro.br IPv6.br

Verificando a Vizinhança BGP

- Cisco

```
router-R13#show bgp ipv6 unicast summary
BGP router identifier 172.21.15.253, local AS number 64501
BGP table version is 45, main routing table version 45
28 network entries using 4368 bytes of memory
54 path entries using 4104 bytes of memory
45/17 BGP path/bestpath attribute entries using 7560 bytes of memory
34 BGP AS-PATH entries using 848 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
Bitfield cache entries: current 2 (at peak 3) using 64 bytes of memory
BGP using 16944 total bytes of memory
26 received paths for inbound soft reconfiguration
BGP activity 49/1 prefixes, 96/21 paths, scan interval 60 secs

Neighbor          V      AS MsgRcvd MsgSent  TblVer InQ OutQ Up/Down State/PfxRcd
2001:DB8:21::254  4   64501    1867    1856       45   0    0 1w0d    Active
2001:DB8:21::255  4   64501    4136    3642       45   0    0 1d07h          26
2001:DB8:20::255  4   64512    1896    1876       45   0    0 1d07h           0
```

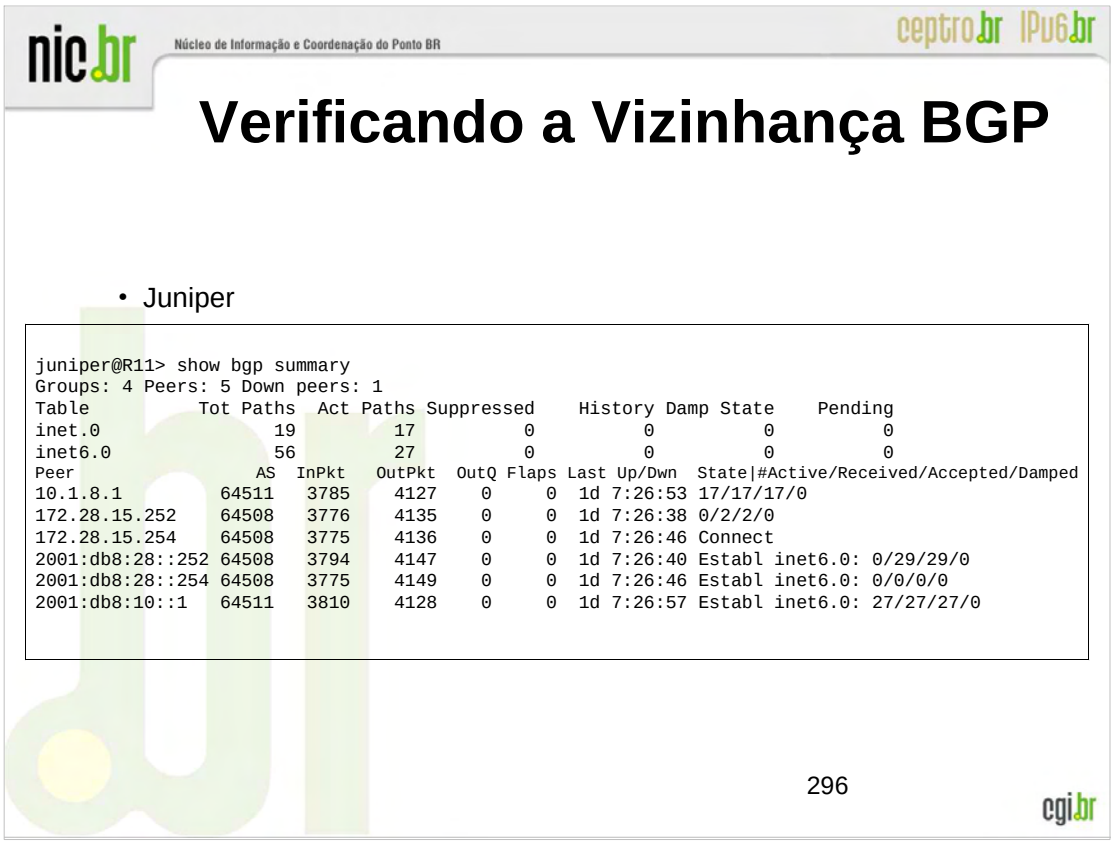

cgi.br

Neste exemplo, podemos observar os seguintes dados sobre os vizinhos das sessões BGP IPv6 em um roteador Cisco:

- Neighbor – IP do vizinho em que se estabeleceu a sessão BGP;
- V – versão do BGP;
- AS – ASN do vizinho;
- MsgRcvd – quantidade de mensagens recebidas do vizinho;
- MsgSent – quantidade de mensagens enviadas ao vizinho;
  - esses dois últimos campos normalmente não são muito verificados, mas são importantes. Muita variação desses campos pode identificar um problema. Receber muitas mensagens, se a tabela está sendo atualizada muitas vezes, pode indicar uma flutuação grande com seu vizinho.
- TblVer – versão da tabela;
- InQ – fila de entrada de pacotes;
- OutQ – fila de saída de pacotes;
- Up/Down - tempo da última mudança de estado;
- State/PfxRcd – indica o estado atual ou o número de prefixo aprendidos. Lembre-se, apesar de existirem estados como Active e Established, que aparentemente indicam que a sessão está ok, eles apenas representam estados intermediários da conexão. A sessão só estará plenamente estabelecida quando houver a indicação de quantos prefixos foram aprendidos.

Observe a saída do mesmo comando, mas agora para visualizar as informações sobre a vizinhança BGP IPv4 em um roteador Cisco:

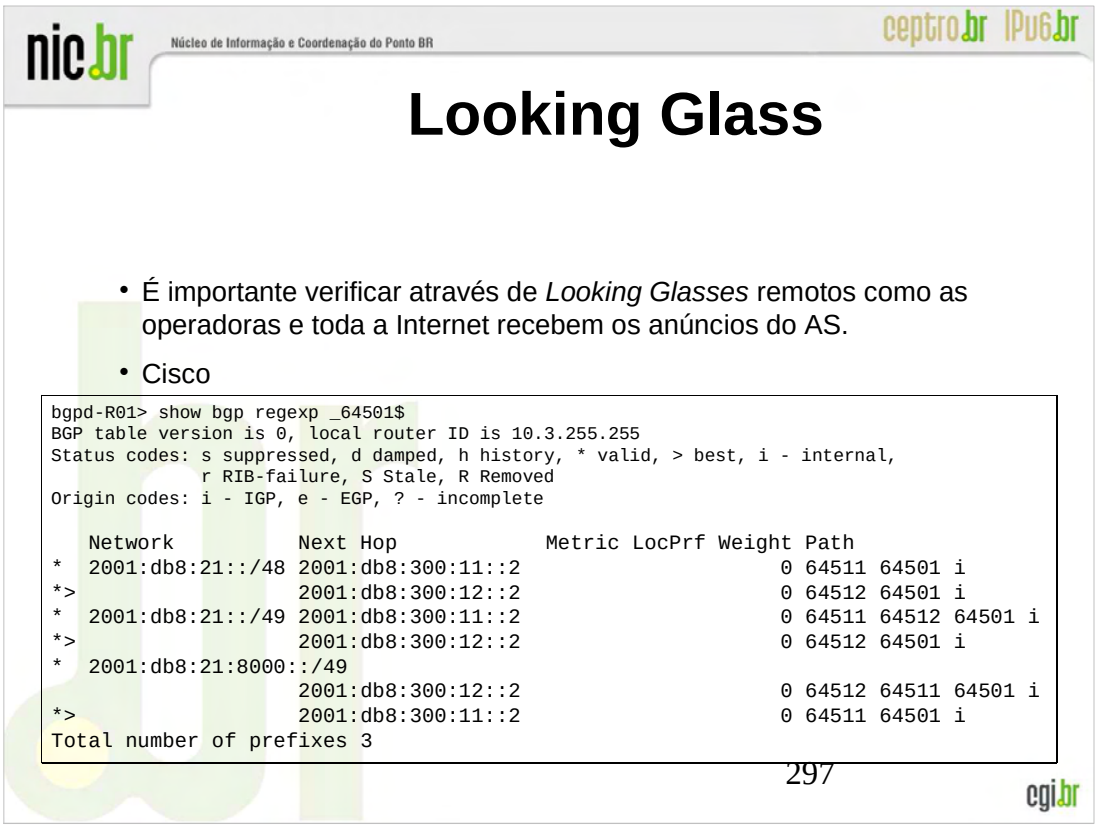
```
router-R13#show bgp ipv4 unicast summary
BGP router identifier 172.21.15.253, local AS number 64501
BGP table version is 80, main routing table version 80
31 network entries using 4092 bytes of memory
52 path entries using 2704 bytes of memory
33/22 BGP path/bestpath attribute entries using 5544 bytes of memory
28 BGP AS-PATH entries using 672 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
Bitfield cache entries: current 3 (at peak 4) using 96 bytes of memory
BGP using 13108 total bytes of memory
21 received paths for inbound soft reconfiguration
BGP activity 99/40 prefixes, 185/84 paths, scan interval 60 secs

Neighbor        V      AS MsgRcvd MsgSent  TblVer  InQ OutQ Up/Down
State/PfxRcd
10.2.255.255    4   64512   10578   10474       80    0    0 1w0d     Active
172.21.15.254   4   64501   10544   10490       80    0    0 1w0d            0
172.21.15.255   4   64501   10572   10490       80    0    0 1w0d           21
```



Neste exemplo, podemos observar os seguintes dados sobre os vizinhos das sessões BGP em um roteador Juniper:

- Groups - número de grupos BGP;
- Peers – números de vizinhos BGP;
- Down peers – número de vizinhos BGP desconectados;
- Table – nome da tabela de rotas;
- Tot Paths – número total de caminhos;
- Act Paths – número de rotas ativas;
- Suppressed - número de rotas atualmente inativas. Estas rotas não aparecem na tabela de encaminhamento e não são exportadas pelos protocolos de roteamento;
- History - número de rotas retiradas armazenadas localmente para manter o controle histórico de instabilidade;
- Damp State – número de rotas com uma figura de mérito maior que zero, mas que continuam ativas porque o valor não atingiu o limite em que a retirada ocorre;
- Pending – rotas em processamento pela política de importação do BGP;
- Peer – endereço de cada vizinho BGP;
- AS – ASN do vizinho;
- InPkt – número de pacotes recebidos do vizinho;
- OutPkt – númrero de pacotes enviados ao vizinho;
- OutQ - fila de saída de pacotes;
- Flaps – número de vezes que a sessão BGP foi interrompida e se re-estabeleceu;
- Last Up/Down – última vez em que ocorreu uma mudança de estado;
- State|#Active/Received/Accepted/Damped - indica o estado atual ou o número de prefixo aprendidos. Se a sessão não foi estabelecida, este campo mostra o estado atual da sessão: Active, Connect, ou Idle. Se a sessão foi estabelecida, o campo indica o número de rotas ativas, recebidas, aceitas ou instáveis.

Observe que o Juniper apresenta na saída do mesmo comando as informações sobre as sessões IPv4 e IPv6.



É importante verificar através de *Looking Glasses* remotos como as operadoras e toda a Internet recebem os anúncios do AS. Deste modo, é possível verificar também, se suas políticas de roteamento foram bem aplicadas.

Com o *Looking Glass*, é possível consultar como os prefixos de um AS, IPv4 e IPv6, estão sendo aprendidos pela Internet, ou seja, como os ASs conseguem chegar até a sua rede.

Ex.:

show bgp regexp _64501$ (Cisco / Quagga)

Neste exemplo, em um roteador Cisco, podemos observar como cada prefixo anunciado pelo AS 64501 foi aprendido. Nesta expressão regular, o “$” significa que é a origem (início de linha), e o “_” significa espaço, ou seja, o comando é aplicado para todo prefixo que tem o AS 64501 como origem no AS-PATH.

Na resposta a consulta podemos observar o balanceamento do tráfego, pois o bloco /48 foi dividido em dois prefixos /49 permitindo que metade do tráfego saia por um *link* e a outra metade saia por outro *link*, e também a redundância das rotas, pois além dos prefixos /49 o prefixo /48 também está sendo anunciado pelos dois *links*.

Looking Glass

- Juniper

```
juniper@R11> show route table inet6.0 aspath-regex .64513$

inet6.0: 59 destinations, 84 routes (59 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8:300::/48  *[BGP/170] 01:44:51, localpref 100
                      AS path: 64511 64513 I
                    > to 2001:db8:100:1::1 via ge-0/0/0.2105
                    [BGP/170] 01:44:13, MED 0, localpref 100, from 2001:db8:21:ffff::252
                      AS path: 64512 64513 I
                    > to fe80::224:97ff:fec1:c8bd via ge-0/0/0.2101
2001:db8:300::/49  *[BGP/170] 01:44:13, MED 0, localpref 150, from 2001:db8:21:ffff::252
                      AS path: 64512 64513 I
                    > to fe80::224:97ff:fec1:c8bd via ge-0/0/0.2101
                    [BGP/170] 01:44:51, localpref 100
                      AS path: 64511 64513 I
                    > to 2001:db8:100:1::1 via ge-0/0/0.2105
2001:db8:300:8000::/49
                    *[BGP/170] 01:44:51, localpref 150
                      AS path: 64511 64513 I
                    > to 2001:db8:100:1::1 via ge-0/0/0.2105
```

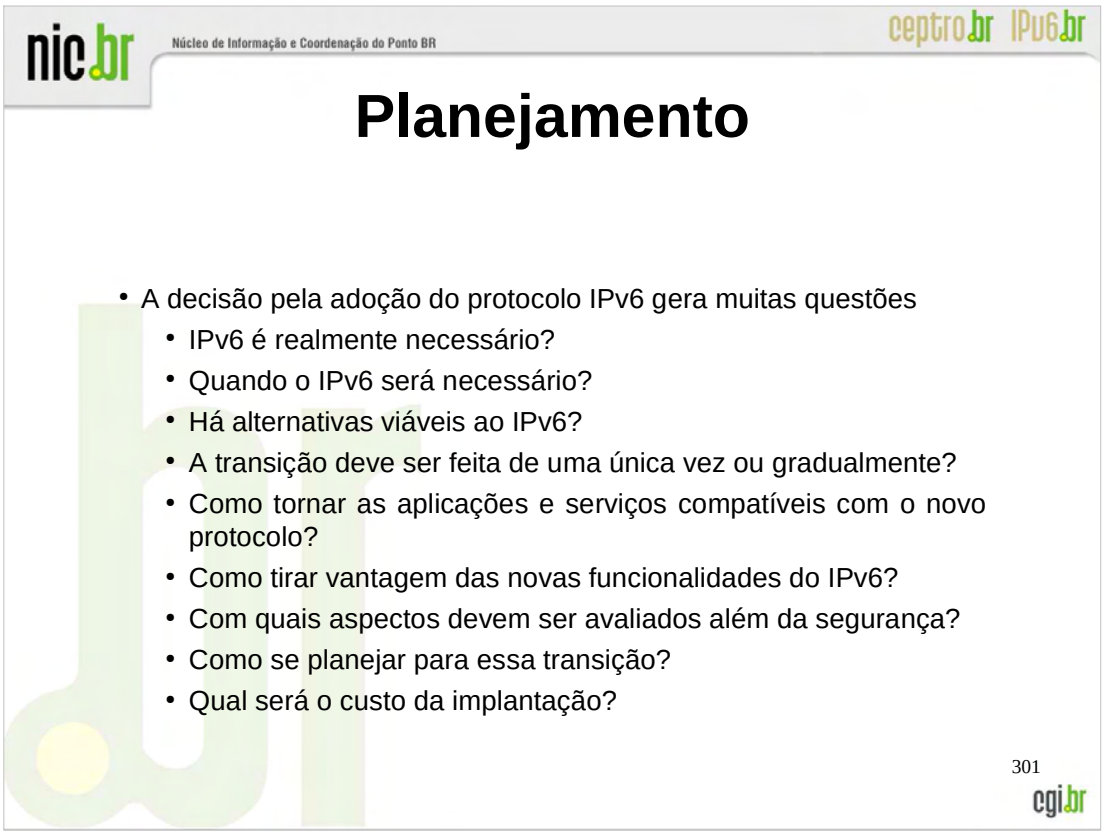

Neste exemplo podemos observar, em um roteador Juniper, como os prefixo anunciado pelo AS64513 foram aprendidos pelo AS64501. A expressão regular utilizada é similar a vista no exemplo anterior utilizando roteadores Cisco.

Na resposta a consulta podemos observar o balanceamento do tráfego e também a redundância das rotas, assim como no exemplo anterior, além de informações como: melhor rota, indicado pelo asterisco; caminho até o destino (*AS path*); próximo salto; e interface de saída do pacote.

IPv6.br

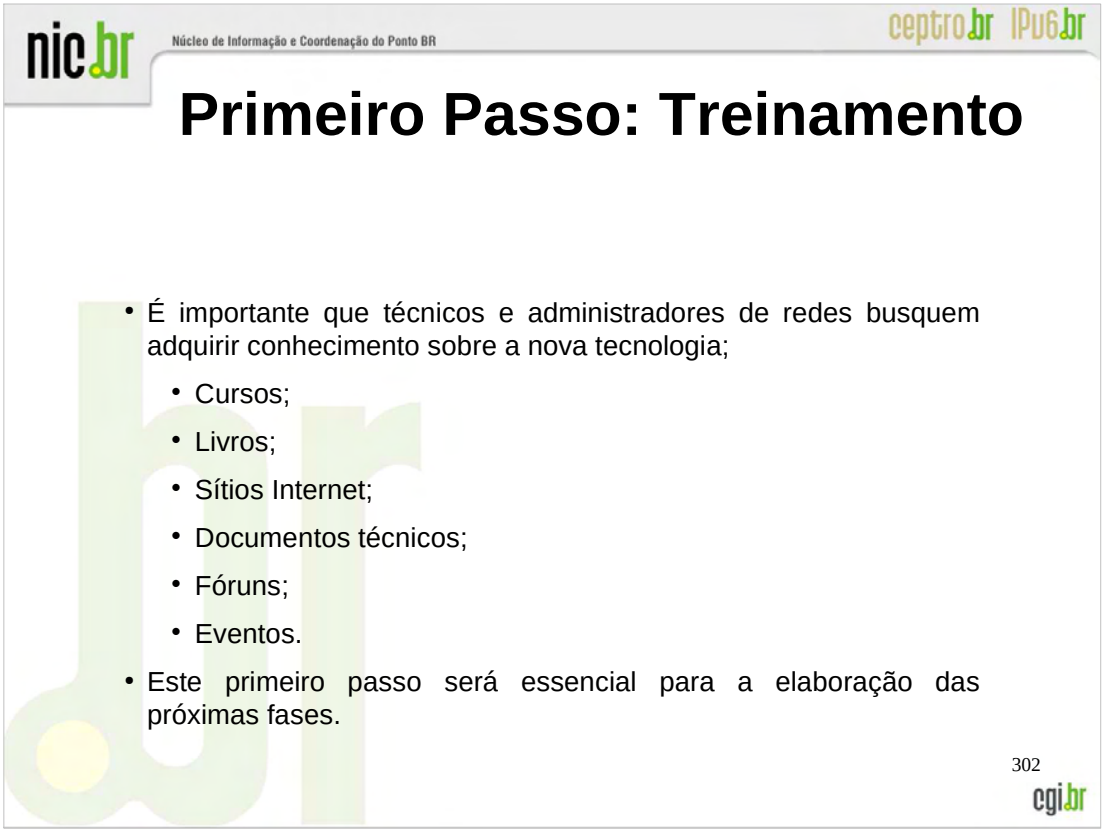
A Nova Geração do Protocolo Internet

Após conhecermos um pouco sobre o funcionamento e os novos recursos do protocolo IPv6, discutiremos neste módulo alguns conceitos importantes que precisam ser estudados na hora de implantar IPv6 nas grandes redes de computadores. Devemos compreender todos os aspectos que envolvem a implantação do IPv6 principalmente em redes corporativas, como análise de custos, troca de material e treinamento.



Conceitos Importantes:

- A troca de protocolo tem um caráter estrutural;
- Esta mudança não ocorrerá apenas porque o protocolo IPv6 apresenta melhorias em relação ao seu antecessor;
- A implantação do IPv6 é necessária e inevitável;
- O esgotamento dos endereços IPv4 não fará a Internet acabar, nem mesmo que ela deixe de funcionar;
- Mas deve haver uma diminuição na taxa de crescimento da rede e dificuldades no desenvolvimento de novas aplicações;
- Todos esses problemas podem ser evitados com a adoção do IPv6 antes do término do IPv4;
- A implantação do IPv6 não será algo rápido;
- Não haverá uma “data da virada” para a troca de protocolo;
- A migração do IPv4 para IPv6 acontecerá de forma gradual, com o IPv4 ainda em funcionamento;
- É importante que as redes estejam preparadas para o novo protocolo desde já. Quanto mais cedo a questão for entendida, e a implantação planejada, menores serão os gastos no processo.



O RIPE NCC disponibiliza alguns vídeos sobre “cases IPv6” que podem servir como fonte de conhecimento muito interessante. Esses vídeos estão no canal ripencc do youtube: www.youtube.com/ripencc, e alguns têm legendas em português:

- Randy Bush (Internet Initiative Japan Inc.) - http://www.youtube.com/watch?v=Qh3i6lDqWBM
- Lorenzo Colitti (Google) - http://www.youtube.com/watch?v=vFwStbTpr6E
- Marco Hogewoning (Dutch ISP XS4ALL) - http://www.youtube.com/watch?v=f3WcWBIQ11A
- Andy Davidson (NetSumo ISP Consultancy) - http://www.youtube.com/watch?v=QCcigLJJbvU
- David Freedman (Claranet) - http://www.youtube.com/watch?v=HQtbz1ahRxE

O impacto do IPv6

- Como o IPv6 pode afetar os negócios:
  - Novas aplicações;
  - Novas oportunidades;
  - Novos serviços.
- Como obter conexão;
- Que tipo de conexão oferecer aos clientes;
  - IPv6 nativo;
  - Túneis.
- Quais serviços internos serão migrados inicialmente;
- Entender esses aspectos é essencial para otimizar o retorno dos investimentos.

O impacto do IPv6

- Minimizar os custos da implantação.
  - Custo Relativo em um ISP*:
    - Com equipamentos (15%)
      - Roteadores - Médio;
      - Firewalls - Médio.
    - Com softwares (15%)
      - Softwares de gerenciamento e monitoramento de redes - Alto;
      - SOs - Médio.
    - Mão-de-obra (70%)
      - Pesquisa e desenvolvimento - Baixo;
      - Treinamento - Alto;
      - Implementação - Alto;
      - Manutenção – Médio/Alto;
      - Problemas de interoperabilidade - Médio/Alto.

*Fonte: U.S. DEPARTMENT OF COMMERCE

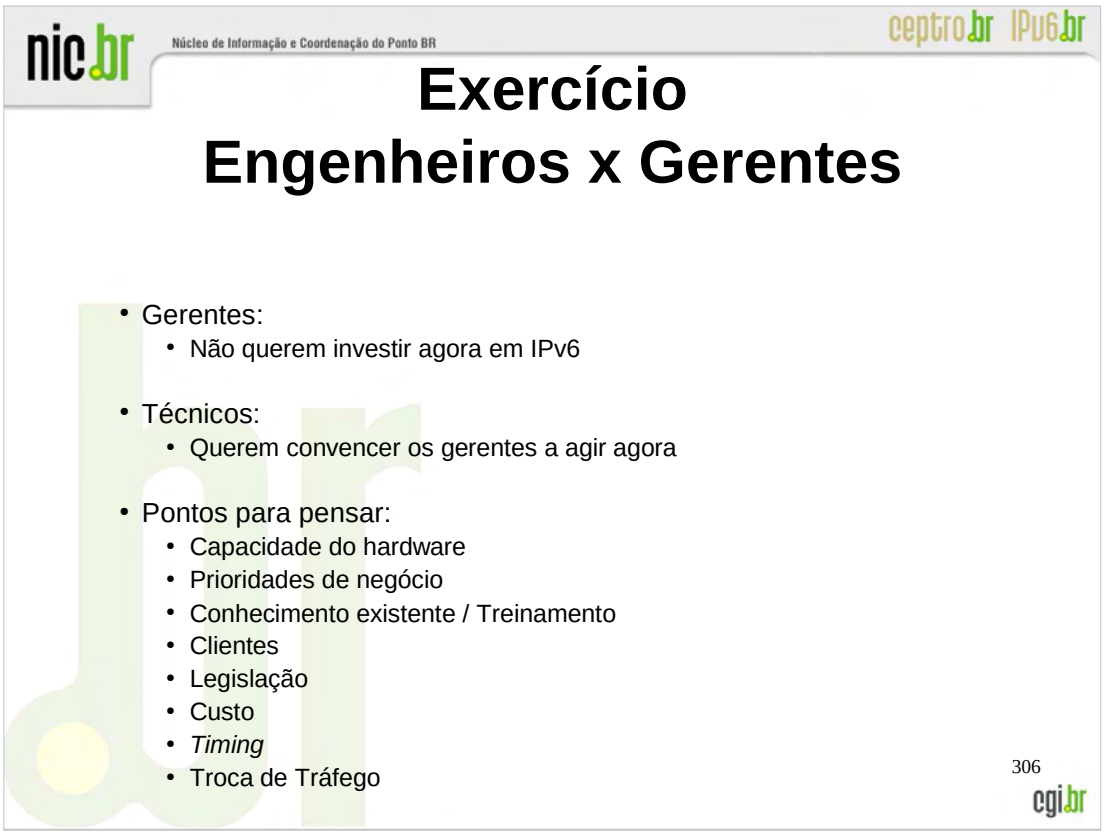
Exercício
Engenheiros x Gerentes

Gerentes/Gestores/Diretores

X

Técnicos/Engenheiros

- Você pode convencer os gestores de sua empresa?
- Vamos trabalhar em grupos...
  - 10 min – preparação
  - 10 min – apresentação e discussão



Este exercício foi aplicado no treinamento IPv6 oferecido pelo RIPE a provedores europeus. Algumas respostas apresentadas foram:

Técnicos:

- Um cliente pediu!
- Dispositivos móveis
- Nosso equipamento já suporta
- Se fizermos agora, podemos investir gradualmente
- Obter a alocação agora é um investimento, muito espaço, nenhum custo
- Há várias oportunidades de treinamento
- V6 é a única opção para crescermos
- V6 é fácil, não são necessários muitos recursos
- Compare o preço de fazer agora, com o de deixar pra depois

Gerentes:

- Vai custar muito, vamos precisar de 20% a mais no orçamento
- Não teremos clientes para isso
- Antes precisamos auditar o equipamento
- Precisamos de um plano, em fases
- Os equipamentos podem ter v6 mas não tem paridade nas funcionalidades
- Temos equipamentos e softwares obsoletos
- Empresas que se preocupam com segurança não querem v6 agora, porque hardware e software não estão maduros
- Por que temos que mudar?
- Como você pode garantir que o v4 vai acabar mesmo?

Cenário: Fazer nada!

- Nenhum problema nos próximos anos
- Com o passar do tempo, algumas pessoas não poderão fazer uso de seus serviços
- Nenhum custo extra
  - Até batermos no muro!
- Custos altos para uma implantação rápida
- Tempos de planejamento curtos, implicam em mais erros...

Cenário: Fazer tudo agora!

- Talvez o hardware tenha de ser trocado
- Investimento alto em tempo e outros recursos
- Sem retorno imediato
- Altos custos para uma implantação rápida
- Planejamento rápido significa mais possibilidade de erros...

Cenário: Comece agora, faça em etapas

- Defina metas e prazos a serem cumpridos.
- Identifique quais áreas e serviços serão afetados.
- Procedimento de compra
  - Paridade de funcionalidades
- Verifique seu hardware e software
  - Aplicações ou sistemas que não serão atualizados;
  - Serviços críticos.
- Faça testes
- Um serviço de cada vez:
  - Face primeiro
  - Core
  - Clientes
- Prepare-se para desligar o IPv4

Face primeiro

- Desenvolva um plano de endereçamento
- Obtenha os endereços
- Anuncie-os
- Web
- DNS autoritativo
- Servidores de email
- etc.

Obtendo um prefixo IPv6

- Todos os RIRs já distribuem endereços IPv6 em suas regiões.
- Preencha o formulário em:
  - *http://registro.br/info/pedido-form.txt*
- Enviar por email: *numeracao-pedido@registro.br*
- Receberá um ticket, ou uma mensagem indicando erros de preenchimento
- Quem tem IPv4 certamente justifica IPv6
- Gratuito, por hora
- 2 semanas entre análise e aprovação
- Dúvidas: *numeracao@registro.br*

Nãos

- Não separe as funcionalidades v6 do v4
- Não faça tudo de uma vez
- Não indique um “guru IPv6” para sua organização
  - Você tem um especialista v4?
- Não veja o IPv6 como um produto
  - O produto é a Internet, ou o acesso/conteúdos Internet.

Considerações

- O IPv4 não é mais igual a Internet
- Evitar o problema não fará ele desaparecer
- Quanto você está disposto a gastar agora, para economizar dinheiro depois?
- Somente o IPv6 permitirá o crescimento contínuo da rede

Comece agora!

BIBLIOGRAFIA

• Migrating to IPv6 : A Practical Guide to Implementing IPv6 in Mobile and Fixed Networks .
Autor: Marc Blanchet.

• 6 Net: An IPv6 Deployment Guide.
Autor: Martin Dunmore.

• IPv6 Essentials.
Autor: Silvia Hagem.

• Global IPv6 Strategies.
Autores: Patric Grossetete; Ciprian popovicius; Fred Wetting.

• Planning and Accomplishing the IPv6 Integration: Lessons Learned from a Global Construction and Project-Management Company .
Autor: Cisco Public Information.

• Technical and Economic Assessment of Internet Protocol Version 6.
Autor: U.S. DEPARTMENT OF COMMERCE.

• Introducción a IPv6.
Autor: Roque Gagliano.

• Planificando IPv6.
Autor: Roque Gagliano.

• Deliverable D 6.2.4: Final report on IPv6 management tools, developments and tests.
Autor: 6 Net.

• IPv6 Security: Are You Ready? You Better Be!
Autor: Joe Klein.

• IPv6 and IPv4 Threat Comparison and Best- Practice Evaluation (v1.0)
Autores: Sean Convery; Darrin Miller.

• BGP Routing Table Analysis Reports - http://bgp.potaroo.net/
Autores: Tony Bates; Philip Smith; Geoff Huston.

• Measuring IPv6 Deployment
Autores: Geoff Huston; George Michaelson.

• IPv6 at Google.
Autores: Angus Lees; Steinar H. Gunderson.

• Resumo do Barômetro Cisco Banda Larga Brasil 2005-2010
Autores: Mauro Peres; João Paulo Bruder.

• Tracking the IPv6 Migration. Global Insights From the Largest Study to Date on IPv6 Traffic on the Internet.
Autor: Craig Labovitz.

• ICE: Uma solução geral para a travessia de NAT
Autor: José Henrique de Oliveira Varanda

BIBLIOGRAFIA

• TIC Domicílios e Usuários Total Brasil - http://cetic.br/usuarios/tic/2009-total-brasil/index.htm
Autor: CETIC.br.

• IPv6.br - http://ipv6.br
Autor: CEPTRO.br.

• IPv6 Deployment and Support - http://www.6deploy.org
Autor: 6Deploy.

• World Internet Usage Statistics News and World Population Stats - http://www.internetworldstats.com/stats.htm
Autor: Internet World Stats.

• Barômetro – BRASIL - http://www.cisco.com/web/BR/barometro/barometro.html
Autor: Cisco Systems.

• The ISC Domain Survey - https://www.isc.org/solutions/survey
Autor: Internet Systems Consortium.

• Number Resource Organization – Statistics - http://www.nro.net/statistics
Autor: Number Resource Organization (NRO).

• Routing TCP/IP
Autor: Jeff Doyle,Jennifer DeHaven Carroll

• O Protocolo BGP4 - Parte 3 (Final) - http://www.rnp.br/newsgen/9907/pgbp4p3.html
Autor: RNP – Rede Nacional de Ensino e Pesquisa