UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
CURSO DE ESPECIALIZAÇÃO SEMIPRESENCIAL EM CONFIGURAÇÃO E GERENCIAMENTO DE SERVIDORES E EQUIPAMENTOS DE REDES

BRUNO TOKARSKI DE CARVALHO

IMPLEMENTAÇÃO DE IPV6 NA REDE DO CÂMPUS CURITIBA DA UTFPR: UM ESTUDO DE CASO

MONOGRAFIA DE ESPECIALIZAÇÃO

CURITIBA
2020

BRUNO TOKARSKI DE CARVALHO

IMPLEMENTAÇÃO DE IPV6 NA REDE DO CÂMPUS CURITIBA DA UTFPR: UM ESTUDO DE CASO

Monografia de Especialização, apresentada ao Curso de Especialização Semipresencial em Configuração e Gerenciamento de Servidores e Equipamentos de Redes, do Departamento Acadêmico de Eletrônica – DAELN, da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para obtenção do título de Especialista.

Orientador: Prof. Dr. Kleber Kendy Horikawa Nabas

CURITIBA
2020

Ministério da Educação
Universidade Tecnológica Federal do Paraná
Câmpus Curitiba
Diretoria de Pesquisa e Pós-Graduação
Departamento Acadêmico de Eletrônica
Curso de Especialização Semipresencial em Configuração e Gerenciamento de Servidores e Equipamentos de Redes

TERMO DE APROVAÇÃO

IMPLEMENTAÇÃO DE IPV6 NA REDE DO CÂMPUS CURITIBA DA UTFPR: UM ESTUDO DE CASO

por

BRUNO TOKARSKI DE CARVALHO

Esta monografia foi apresentada em 30 de Julho de 2020 como requisito parcial para a obtenção do título de Especialista em Configuração e Gerenciamento de Servidores e Equipamentos de Redes. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

_____
Prof. Dr. Kleber Kendy Horikawa Nabas
Orientador

_____
Prof. Dr. Edenilson José da Silva
Membro titular

_____
Prof. M. Sc. Omero Francisco Bertol
Membro titular

- O Termo de Aprovação assinado encontra-se na Coordenação do Curso -

AGRADECIMENTO

Agradeço a minha esposa Gabrieli Pereira da Cruz pela disponibilidade em ajudar durante todos os momentos durante o curso e o desenvolvimento deste trabalho, além do apoio incondicional que fazem com que eu possa ser todos os dias a melhor versão de mim mesmo.

RESUMO

CARVALHO, Bruno Tokarski de. **Implementação de IPv6 na rede do Câmpus Curitiba da UTFPR: um estudo de caso**. 2020. 60 p. Monografia de Especialização em Configuração e Gerenciamento de Servidores e Equipamentos de Redes, Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2020.

A evolução das redes de computadores formou o que chamamos de Internet: uma grande rede de redes. A base da comunicação na Internet é um protocolo desenvolvidos na década de sessenta que, com pequenas modificações, opera hoje em escala mundial como padrão de comunicação de todos os serviços conectados. O IPv4 é este protocolo e um dos principais déficits desta versão é o tamanho dos endereços, que impacta diretamente na quantidade disponível. Para sanar este problema, foi desenvolvido o IPv6 que, entre outras melhorias ao protocolo antigo, aumenta em grande quantidade o número de endereços disponíveis. Como a nova versão do protocolo não é diretamente compatível com o antigo, as redes de computadores existentes terão de migrar para IPv6 para que a Internet possa continuar sua expansão sem limitações ocasionadas pela falta de endereços. Este trabalho analisa a implementação do protocolo IPv6 em pilha dupla na rede do Câmpus Curitiba da UTFPR.

Palavras-chave: Redes de computadores. TCP/IP (Protocolo de rede de computador). IPv6. Pilha Dupla.

ABSTRACT

CARVALHO, Bruno Tokarski de. **Implementation of IPv6 in the UTFPR Câmpus Curitiba network: a case study**. 2020. 60 p. Monografia de Especialização em Configuração e Gerenciamento de Servidores e Equipamentos de Redes, Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2020.

The evolution of computer networks has formed what is called Internet: a big network of networks. The basis of communication on the Internet is a protocol developed in the sixties that, with minor modifications, operates today on a worldwide scale as the communication standard for all connected services. IPv4 is this protocol and one of the main deficits of this version is the size of the addresses, which directly impacts the quantity available. To remedy this problem, IPv6 was developed which, among other improvements to the old protocol, greatly increases the number of available addresses. As the new version of the protocol is not directly compatible with the old one, the existing computer networks will have to migrate to the new protocol so that the Internet can continue its expansion without limitations caused by the lack of addresses. This work analyzes the implementation of the IPv6 protocol in dual stack in the UTFPR Câmpus Curitiba network.

Keywords: Computer networks. TCP/IP (Computer network protocol). IPv6. Dual Stack.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

1.1 CONTEXTO

O protocolo *Transmission Control Protocol/Internet Protocol* (TCP/IP) surgiu em 1969, com a *Advanced Research Projects Agency Network* (ARPANET) uma rede do Departamento de Defesa dos Estados Unidos da América. O *Internet Protocol* (IP) como construído naquele momento é o IPv4, que chegou em funcionamento até os dias de hoje com pouquíssimas mudanças (PYLES; CARRELL; TITTEL, 2016).

Com a definição de endereços de 32 *bits* no IPv4, o problema de escassez de endereços já se mostrava real no início da década de 1990. Com intenção de resolver este problema e melhorar o protocolo, foi construído o IPv6, o novo protocolo da Internet (SANTOS *et al.*, 2014).

Com um custo elevado para transição para o novo protocolo de forma global, o IPv6 foi desenvolvido com a possibilidade de operar em paralelo ao antigo protocolo, propondo uma transição gradual. Isso permite que o protocolo IPv6 seja aos poucos configurado e passe a operar nas redes. Da mesma forma, quando o IPv4 for desativado, bastará a configuração dos equipamentos para que não encaminhem pacotes da versão antiga (SANTOS *et al.*, 2014).

Com essas características de operação paralela, várias técnicas de transição foram desenvolvidas para que sejam aplicadas em diferentes escopos de rede, baseado nos sistemas dos *hosts*, nos equipamentos de rede disponíveis e na oferta de IPv6 pelos *Internet Service Providers* (ISP) (IPV6.BR, 2012b).

1.2 PROBLEMA

Devido a boas práticas de alocação e técnicas para economia de endereços, o Câmpus Curitiba da Universidade Tecnológica Federal do Paraná (UTFPR) ainda possui diversos endereços IPv4 livres. Isso não é suficiente para oferecer endereços válidos com conexão fim a fim para todos os usuários da rede, mas permite que o IPv6 não seja implementado com urgência.

Mesmo assim, a ausência da disponibilização do novo protocolo no Câmpus pode estar limitando diversas atividades de pesquisa que envolvam IPv6, bem como não permite oferecer os serviços hospedados na infraestrutura de Tecnologia da Informação (TI) deste Câmpus com IPv6 nativo, tornando a conexão muito mais eficiente para todos os usuários na Internet que já possuem acesso em pilha dupla.

A Rede Nacional de Pesquisa (RNP), vários ISP do Brasil e do mundo e a maioria dos serviços de rede consolidados já oferecem conectividade IPv6 nativa e o Câmpus Curitiba da UTFPR deve passar a oferecer a conectividade com IPv6 nas suas três sedes, para continuar o caminho de aprimoramento da rede que vem sendo realizado nos últimos anos pela equipe de TI deste órgão.

1.3 OBJETIVOS

1.3.1 Objetivo Geral

Contribuir com a disseminação do IPv6 através da implementação em um dos Câmpus da UTFPR, oferecendo este novo protocolo em pilha dupla nas conexões de todos os *hosts* conectados na rede.

1.3.2 Objetivos Específicos

Para atender ao objetivo geral neste trabalho de conclusão de curso de especialização os seguintes objetivos específicos serão abordados:

- Contextualizar historicamente a Internet e seu protocolo principal, o TCP/IP;
- Especificar o funcionamento do IPv4 e do IPv6;
- Detalhar as melhores práticas recomendadas pelo Comitê Gestor da Internet no Brasil (CGI.BR) nas descrições das técnicas de transição;
- Descrever o procedimento de implementação do IPv6 no Câmpus Curitiba da UTFPR, baseado nas melhores práticas;
- Apresentar o estado atual de utilização do IPv6 no Câmpus Curitiba.

1.4 JUSTIFICATIVA

É importante destacar que a Internet está em expansão desde a abertura, saindo do escopo apenas militar, primeiro para instituições acadêmicas e depois para fins comerciais. Mesmo com técnicas para melhorar o aproveitamento dos endereços IPv4 ainda livres, estas não foram suficientes para que a ocupação de endereços fosse diminuída substancialmente, tornando a mudança do protocolo inevitável num futuro próximo. Com o esgotamento dos endereços IPv4 mais próximo dia após dia, a adoção do IPv6 já está avançando nos ISP.

Assim também, realizar a implementação do IPv6 nas organizações é essencial, sendo pelo esgotamento dos endereços do protocolo antigo ou pela necessidade de manter atualizados as tecnologias em uso. A demora na troca do protocolo pode ser fator determinante ao gasto de recursos computacionais com técnicas para economizar endereços IPv4, bem como pode estar represando o avanço de novas tecnologias, como *Internet of Things* (IoT).

Num ambiente acadêmico em que diversos laboratórios de ensino e pesquisa conduzem atividades envolvendo o funcionamento de redes IPv4 e IPv6, oferecer o novo protocolo é fundamental, bem como seguir padrões e melhores práticas de implementação.

1.5 ESTRUTURA DO TRABALHO

Esta monografia de especialização está dividida em seis capítulos. Neste primeiro capítulo foi introduzido o tema geral do trabalho, seus objetivos e justificativa.

O segundo capítulo, “TCP/IP, a Internet e o funcionamento da Rede de Computadores”, tratará da Internet historicamente, bem como seu funcionamento básico.

No terceiro capítulo, “O IPv4, estrutura e escassez de endereços”, descreve a estrutura do protocolo IPv4, a forma de distribuição dos endereços e o panorama do esgotamento dos endereços no mundo.

O quarto capítulo, “O IPv6, estrutura e técnicas de transição”, apresentará o protocolo IPv6, algumas de suas diferenças em relação à versão anterior e quais as recomendações e técnicas de implementação do IPv6 juntamente da rede IPv4.

O quinto capítulo, “A implementação de IPv6 na rede do Câmpus Curitiba da UTFPR”, descreve padrões da rede do Câmpus Curitiba, como eles auxiliaram a implementação do novo protocolo, como a implementação foi realizada e qual o estado de utilização após a finalização da fase de implantação.

O sexto é último capítulo, “Considerações finais”, serão retomadas os objetivos e a problemática do capítulo introdutório, apontado os resultados atingidos, por meio do trabalho realizado.

2 TCP/IP, A INTERNET E O FUNCIONAMENTO DA REDE DE COMPUTADORES

Em 1969, um dos setores do Departamento de Defesa dos Estados Unidos, denominado *Defense Advanced Research Projects Agency* (DARPA), começou a desenvolver um projeto para comunicação via rede utilizando o princípio de comutação por pacotes. O projeto levou o nome de ARPANET (PYLES; CARRELL; TITTEL, 2016).

Nesse tipo de comutação, o transmissor e o receptor possuem endereços únicos, o que permite que a rede encaminhe os pacotes, que são pequenos pedaços de dados, para o destino, mesmo que eles não percorram uma rota fixa igual aos pacotes anteriores ou posteriores (PYLES; CARRELL; TITTEL, 2016).

O projeto, com viés de segurança governamental, tinha como premissas permitir a comunicação distribuída, através de longas distâncias e suportar diferentes sistemas. Essas três características vinham da ideia de evitar interrupção diante de ataques a bases militares ou instituições, interligando diversos setores das forças armadas, universidades e outros setores governamentais que, naquele momento, já possuíam computadores e sistemas heterogêneos (PYLES; CARRELL; TITTEL, 2016).

O resultado do projeto foi o desenvolvimento da combinação de protocolos chamada *Transmission Control Protocol*/*Internet Protocol* (TCP/IP, ou Protocolo de Controle de Transmissão/Protocolo de Internet) (TANENBAUM, 2003). Com conclusão em 1978, o IP a qual refere-se aqui é o IPv4, usado até hoje sem muitas modificações substanciais (PYLES; CARRELL; TITTEL, 2016).

Em 1983, todos os protocolos experimentais foram considerados inaptos para acesso a ARPANET, tornando o TCP/IP padrão. Isso, junto da mudança da operação da DARPA para a *Defense Communications Agency* (DCA), facilitou a conexão de novas instituições a rede, como outras agências governamentais e universidades. Essas ações consolidaram o entendimento da ARPANET como uma “rede de redes” e também favoreceram a adoção do TCP/IP como implementação padrão em vários sistemas, disseminando ainda mais o protocolo. No mesmo ano, a separação da parte militar da rede e a criação do *Domain Name System* (DNS), tornou a comunicação mais livre para novos participantes e mais acessível, já que agora não era mais necessário decorar os endereços numéricos

dos serviços que se queria acessar (PYLES; CARRELL; TITTEL, 2016).

O modelo de referência TCP/IP é construído em estrutura de camadas que controlam cada etapa da comunicação dos equipamentos com a rede, definindo padrões. Basicamente, os padrões de trabalho na Internet são publicados pela *Internet Engineering Task Force* (IETF) em forma de *Requests for Comments* (RFC). Segundo Tanenbaum (2003), as camadas são:

- Camada de aplicação: Reúne os diversos protocolos utilizados pelos aplicativos para se comunicarem, os quais realizam operações como, por exemplo, envio e recebimento de e-mails, navegação na Internet, resolução de nomes de domínio ou transferência de arquivos. Cada *software* utiliza o protocolo mais adequado para sua função;
- Camada de transporte: Segmenta os dados recebidos da camada de aplicação em pacotes que serão repassados a camada de Internet, controlando nesse processo o fluxo ordenado dos dados e garantindo a integridade;
- Camada de Internet: acrescenta as informações para o envio dos pacotes da origem ao destino (endereçamento), encaminhando em seguida os dados a próxima camada para envio;
- Camada de acesso à rede: Responsável pelo gerenciamento do envio e recebimento dos dados através do meio físico e, portanto, dependente deste. A arquitetura de camadas neste caso, possibilita a utilização de diferentes *hardwares* e meios de transmissão.

Os protocolos IPv4 e IPv6 trabalham na camada de Internet, a qual é responsável por duas principais tarefas: o endereçamento e a definição das rotas para encaminhar os pacotes.

Esse conjunto proporciona justamente a mais básica tarefa realizada no TCP/IP, pensada como básica no desenvolvimento da ARPANET: a conexão ponto-a-ponto. Através da padronização da linguagem, códigos e regras de comunicação entre o emissor e receptor e de regras que definem o melhor caminho para que a mensagem seja transmitida, ainda que haja redes diferentes entre o comunicantes, foi possível a criação do que hoje é a Internet, provendo comunicação a nível global (PYLES; CARRELL; TITTEL, 2016; TANENBAUM, 2003).

3 O IPV4, ESTRUTURA E ESCASSEZ DE ENDEREÇOS

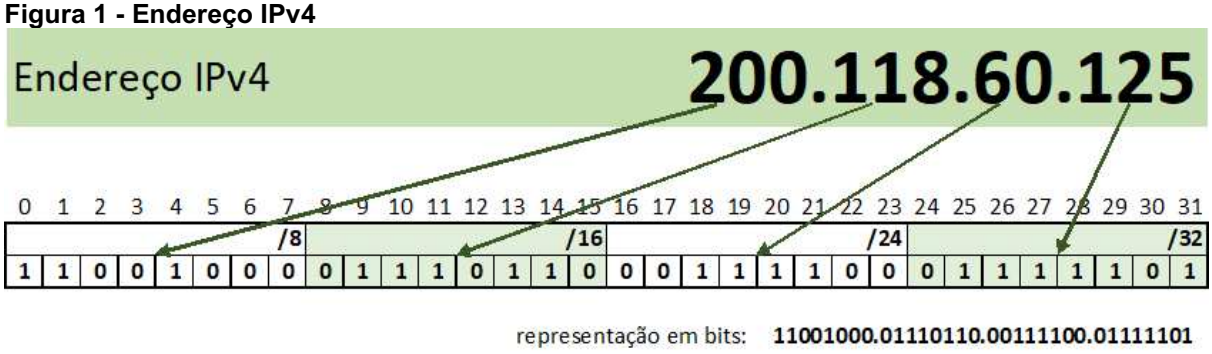
O TCP/IP ganha destaque principalmente pelo fato de permitir conectar uma grande quantidade de equipamentos em rede, tendo sido posto a prova em grande escala. A interconexão atende hoje todas as áreas povoadas do mundo, formando a grande rede mundial de computadores, Internet. Por ter se tornado um padrão mundial, permite que qualquer pessoa do mundo acesse serviços disponibilizados em qualquer outro ponto do globo (COMER, 2014).

Origem e destino desejam se comunicar usando TCP/IP, o que nos permite agora chamá-los de *hosts*. *Host* é qualquer dispositivo que se comunique pela rede com outros dispositivos. Para permitir que a rota correta seja seguida pelos pacotes entre a origem e o destino existem roteadores (COMER, 2014).

O princípio da comutação de pacotes está em dizer qual o caminho que o pacote enviado deve tomar para que chegue ao serviço e receba resposta, considerando os endereços dos *hosts* envolvidos. Desta forma, é necessário que a origem e o destino possuam endereços únicos (TANENBAUM, 2003).

O endereço no IPv4 é formado por 32 *bits* e é dividido em duas partes: o prefixo indica qual a rede e o sufixo indica um *host* daquela rede. Um exemplo de endereço IPv4 e sua representação é ilustrada na Figura 1.

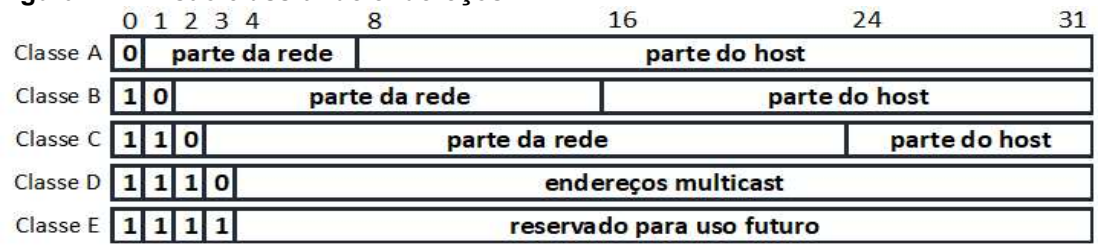
Figura 1 - Endereço IPv4



Fonte: Autoria própria.

A escolha por endereços de 32 *bits* representou a disponibilização de algo em torno de 4.294.967.296 (2^{32}) endereços para toda a Internet, divididos em 5 classes, conforme demonstrado na Figura 2.

Figura 2 - Divisão *classful* de endereços IPv4



Fonte: Adaptado de Comer (2014, p. 72).

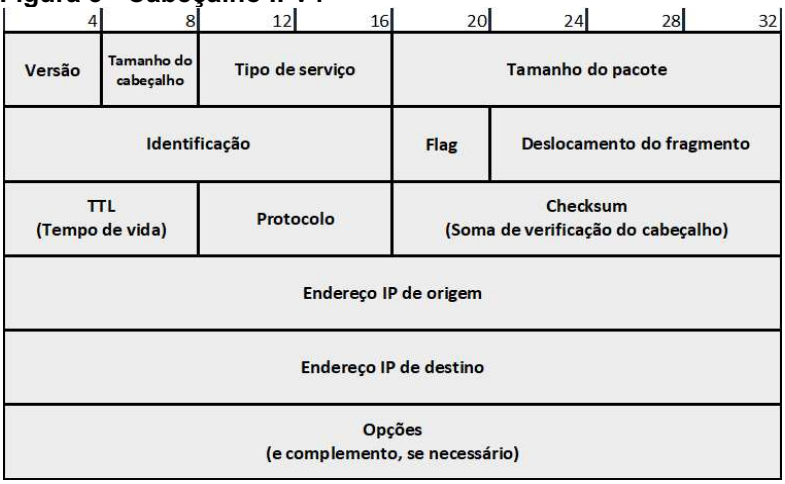
Os endereços são transmitidos no pacote, junto de diversas outras informações, formando o cabeçalho do pacote IPv4, representado na Figura 3, com tamanho variável entre 20 e 60 *bytes*, descrito por Kurose e Ross (2012) como a seguir:

- Versão (4 *bits*): versão do protocolo IP, neste caso preenchido com 0100 (4 em binário);
- Tamanho do cabeçalho (4 *bits*): indica onde os dados começam no datagrama IPv4, já que o cabeçalho possui informações opcionais;
- Tipo de serviço (8 *bits*): indica o tipo do pacote enviado, permitindo que, se oportuno, os pacotes sofram tratamento diferenciado para fins de qualidade do serviços (QoS);
- Tamanho do Pacote (16 *bits*): fornece o tamanho total do pacote em *bytes*, incluindo o cabeçalho e os dados;
- Identificação (16 *bits*): identificação dos fragmentos de um pacote IP enviado pela origem, para facilitar a remontagem;
- Flag (3 *bits*): fornecem informações de controle;
- Deslocamento do fragmento (13 *bits*): indica onde no pacote cada fragmento pertence;
- TTL (Tempo de vida - *Time to live*) (8 *bits*): indica o número máximo de saltos por roteadores que o pacote pode ser encaminhado. O campo é decrementado em cada roteador e o pacote é descartado quando o TTL é igual a zero;
- Protocolo (8 *bits*): indica qual protocolo foi utilizado para criar e enviar a mensagem do pacote. Alguns valores podem ser: 00000001 (01) para ICMP, 00000110 (06) para TCP, 00010001 (17) para *User Datagram Protocol* (UDP);
- Soma de verificação do cabeçalho (*checksum*) (16 *bits*): utilizado para

verificação da validade das informações do cabeçalho. Como o TTL se altera a cada roteador, cada encaminhamento do pacote possui um novo *checksum*. Se o cálculo do *checksum* estiver incorreto ao valor gravado, o pacote é descartado;

- Endereço IP de origem (32 *bits*): endereço IPv4 do *host* origem;
- Endereço IP de destino (32 *bits*): endereço IPv4 do *host* destino;
- Opções (tamanho variável - 0 a 320 *bits*): presença opcional nos pacotes, podendo conter informações para serviços adicionais de segurança, roteamento, etc.

Figura 3 - Cabeçalho IPv4



Fonte: Adaptado de Graziani (2017, p. 50).

O tamanho escolhido para os endereços dos *hosts* estava claramente datado no tempo em relação ao restante do protocolo, já que a projeção de 4 bilhões era enorme em vista dos 700 *hosts* conectados a rede em 1977 (COMER, 2014).

Sobre isso, Vinton Gray Cerf, considerado um dos cofundadores da Internet, declarou numa apresentação em 2011 (CERF, 2011):

> Eu fico um pouco envergonhado com isso porque eu fui o cara que decidiu que 32 *bits* era suficiente para o experimento da Internet. Minha única defesa é que a escolha foi feita em 1977, e eu achei que iria ser apenas um experimento. O problema é que o experimento não acabou, e agora estamos aqui (traduzido pelo autor[1]).

[1] "I am a little embarrassed about that because I was the guy who decided that 32-bit was enough for the Internet experiment. My only defense is that that choice was made in 1977, and I thought it was an experiment. The problem is the experiment didn't end, so here we are." - Vinton Gray Cerf, LCA 2011 Keynote Speech.

Ainda nos anos 80, a partir do crescimento em escala da rede, ficou evidente, tanto que o esquema de classes parecia inadequado, como de que a quantidade de endereços total era pouca para a rede. Além disso, políticas de alocação de endereços em grande quantidade para empresas específicas e também de reserva para serviços específicos consumiu muitos endereços (SANTOS *et al.*, 2014; COMER, 2014). Basicamente, tivemos algo em torno de 18% do volume de endereços alocados dessa forma:

- IBM, HP, AT&T, Xerox, Apple, Ford, GE, MIT, DoD, US Army e USPS receberam uma faixa de 16.777.216 endereços cada;
- 268.435.456 endereços foram reservados para aplicações de Multicast;
- 268.435.456 endereços foram reservados para uso futuro/experimental;
- 16.777.216 endereços foram destinados para aplicação em *loopback*.

Para contornar o problema da alocação de endereços, foi permitida a adoção de sub rede de tamanho variável (redes maiores e menores poderiam ser criadas para as diversas aplicações, de acordo com a necessidade) e o correto roteamento para estes casos (COMER, 2014). Basicamente essa definição foi feita na RFC1519, criando tamanhos diferentes de redes derivados das classes A, B e C do endereçamento *classful* com a separação de um determinado número de *bits* de endereço como prefixo, de forma a orientar o roteamento e permitir a adoção de blocos menores de endereços.

A RFC4632 atualiza o padrão CIDR da RFC1519. Nela também é possível conferir a notação para estes casos: uma rede ou endereço IPv4 seria escrito em 4 octetos, como já era utilizado, seguido de “/” (barra) e do número de *bits* que especifica o tamanho da sub-rede (FULLER; LI, 2006).

A RFC1918 foi publicada, criando faixas exclusivas de IP para serem utilizadas em redes privadas, ou seja, aquelas que não teriam acesso direto a rede mundial de computadores e aos serviços nela hospedados (PYLES; CARRELL; TITTEL, 2016). A RFC1631, que descreve a tradução de endereços privados em públicos, permitindo o acesso de *hosts* em uma rede privada o acesso a Internet, propôs o reuso de endereços de rede, era considerada uma solução temporária, mesmo que ela tenha prolongado a duração dos endereços em quase 25 anos. A técnica de tradução, mapeia diferentes conexões em diversas portas TCP/UDP

diferentes, permitindo diversas conexões ativas no mesmo endereço IP público (POPOVICIU, 2006).

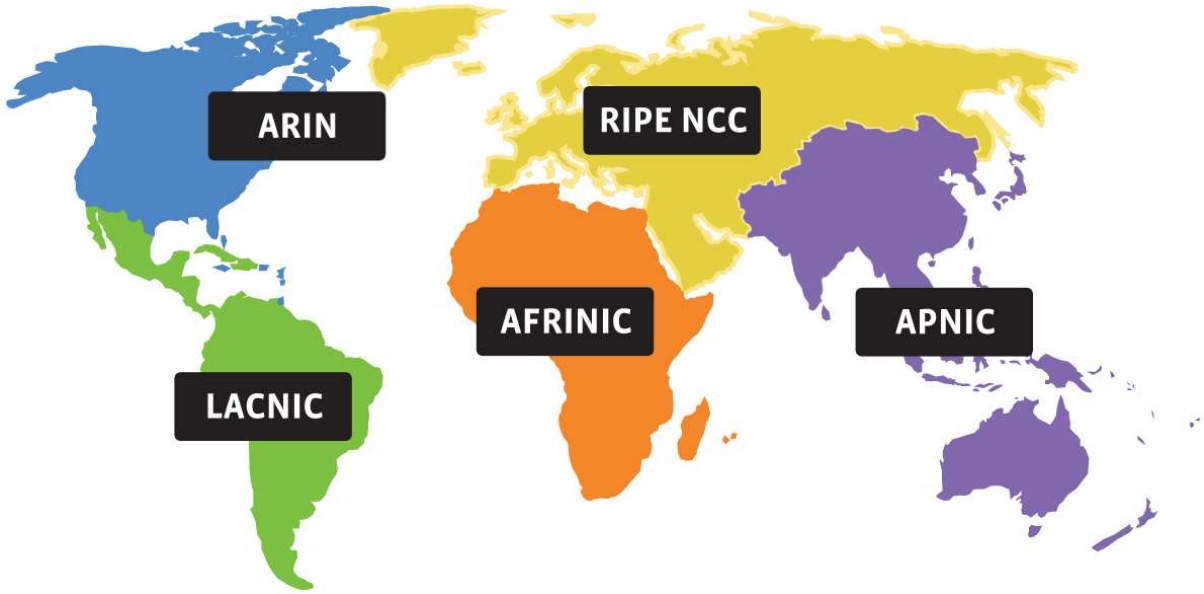
Network Address Translation (NAT) foi um método para economia de endereços amplamente difundido, porém não pode ou deve ser considerado solução definitiva ao problema. A falta de endereços se tornou extremamente crítica, a ponto de não haver endereços disponíveis nem mesmo para ISP, que atualmente fazem uso dos *Carrier Grade NAT* (CGNAT), também chamado de *Large Scale NAT* (LSN), onde o ISP traduz endereços privados alocados a vários clientes em um endereço público (GRAZIANI, 2017).

Segundo Comer (2014), em 1993, antes mesmo da publicação da RFC responsável pela criação do NAT, a Internet foi liberada para acesso comercial, o que aos poucos foi tornando a rede cada vez maior. Entre 1993 e 1997 mais que decuplicou o número de *hosts* conectados a rede. O uso da Internet continua a aumentar por dois principais motivos:

- diversos aparelhos que antes não possuíam a necessidade de um endereço de rede agora estão conectados, o que é denominado *Internet of Things* (IoT) (PYLES; CARRELL; TITTEL, 2016);
- o acesso a Internet por dispositivos como celular ou computador está cada vez mais barato e acessível; no Brasil, por exemplo, a abrangência do acesso chega a 70% da população (GRAZIANI, 2017; CETIC.BR, 2018).

A atribuição de endereços a solicitantes é tarefa dos *Regional Internet Registry* (RIR), que recebem os endereços alocados pela *Internet Corporation for Assigned Names and Numbers* (ICANN). Os cinco RIR existentes gerenciam a distribuição de endereços IP para uma determinada região do mundo (ICANNWIKI, 2015). A abrangência geográfica de cada RIR é representada na Figura 4.

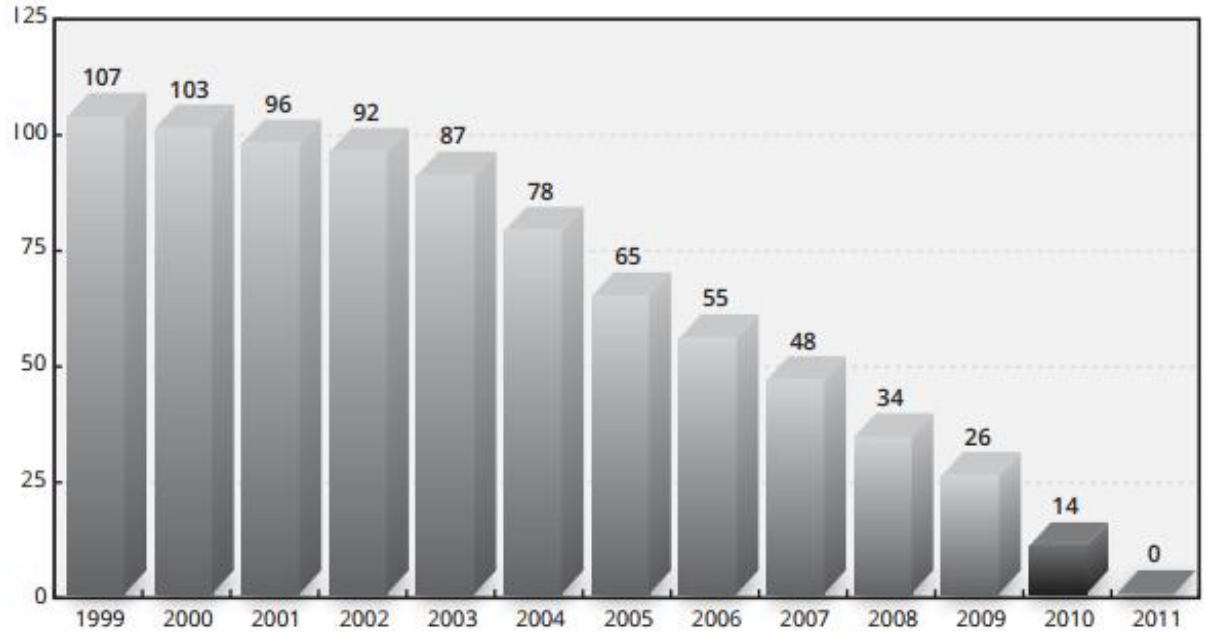
Figura 4 - Mapa de abrangência dos RIR



Fonte: IANA (2013?).

Segundo a documentação da *Internet Assigned Numbers Authority* (IANA), os últimos cinco blocos de IPv4 /8 foram alocados a cada um dos cinco RIR em fevereiro de 2011 e desde então a distribuição de novos endereços IPv4 a novos solicitantes segue restrições (SANTOS *et al.*, 2014). O estoque de blocos IPv4 /8 na IANA está representado no gráfico da Figura 5.

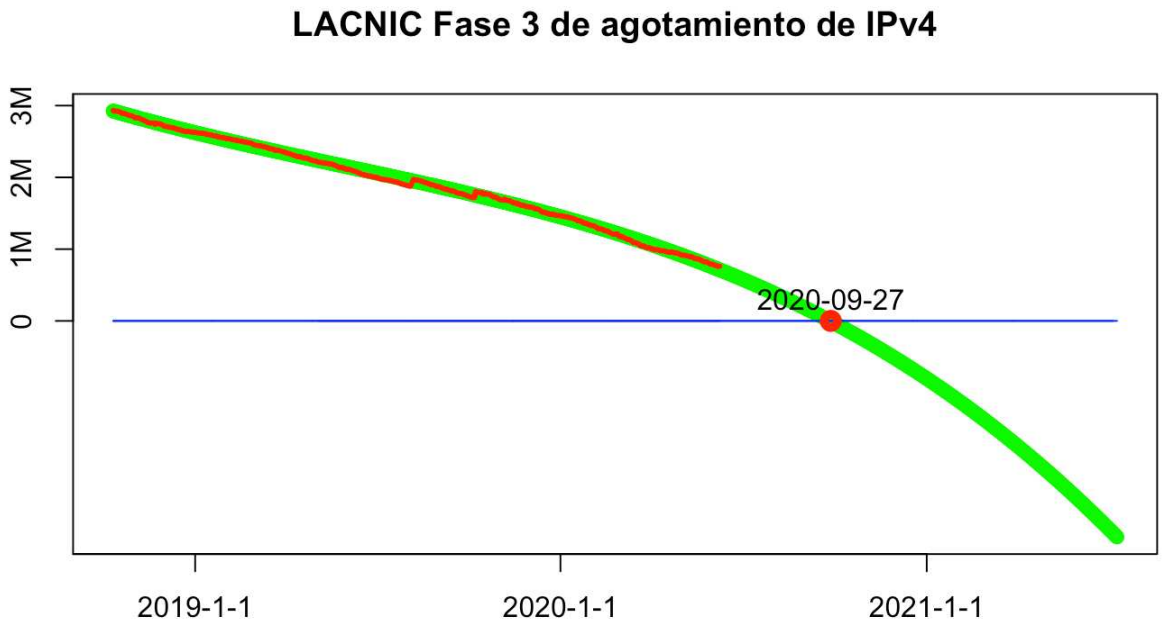
Figura 5 - Evolução do estoque de blocos IPv4 na IANA



Fonte: Santos *et al.* (2014).

No site de acompanhamento do processo de esgotamento do IPv6 na região atendida pelo LACNIC, é possível visualizar o gráfico de projeção de disponibilidade de endereços IPv4 no Brasil, México e restante da América Latina e Caribe em tempo real. Em julho a projeção era como apresentado na Figura 6. O gráfico é apenas um exemplo de como o esgotamento de endereços é crítico, situação que se repete nas outras RIR.

Figura 6 - Previsão do esgotamento de blocos IPv4 no LACNIC



Fonte: LACNIC (2020).

A solução de forma mais definitiva para o problema de esgotamento começou a ser discutida em 1993, que é a criação do protocolo que seria chamado *IP Next Generation* (IPng) e depois IPv6 (COMER, 2014), será abordada no próximo tópico. Tal iniciativa era necessária não só pela necessidade de mais endereços, mas também porque com a adoção do NAT a rede perdeu o princípio da comunicação fim-a-fim, dificultando ou impossibilitando o funcionamento de protocolos que exigem tal conexão direta.

4 O IPV6, ESTRUTURA E TÉCNICAS DE TRANSIÇÃO

Uma visão simplista do IPv6 pode gerar a impressão de que ele é apenas uma expansão do número de endereços em relação ao IPv4, ainda que essa seja sua característica predominante (SANTOS *et al.*, 2014). O IPv6 é na verdade uma evolução do protocolo IPv4, já que o novo protocolo tentou manter características positivas do protocolo antigo, eliminar as não essenciais e criar novas características que se achavam necessárias (TANENBAUM, 2003).

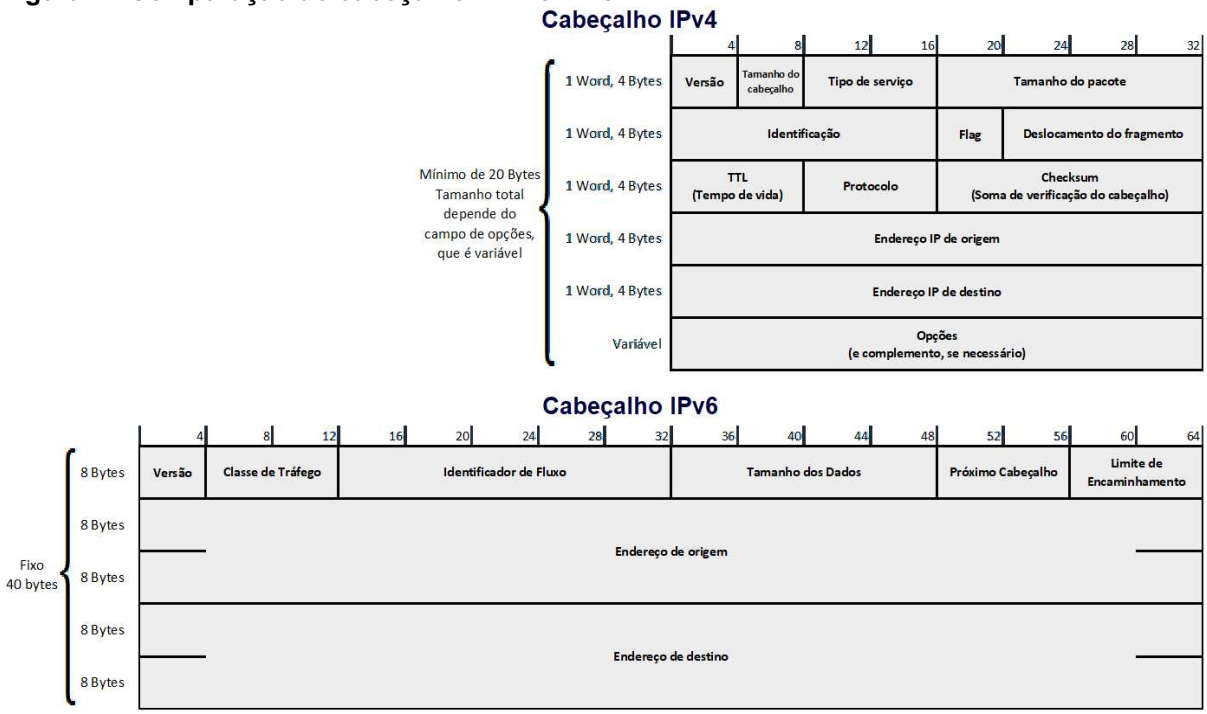
De início podemos ampliar a história da Internet em si com a expectativa de adoção do IPv6. Como a necessidade de um novo esquema de endereçamento culminando na perspectiva para desenvolvimento de uma nova versão do protocolo com a solicitação de propostas para substituição do IPv4 em 1993, várias partes interessadas puderam encaminhar solicitações, bem como evidenciou-se que a Internet até aquele momento estava extremamente concentrada nos Estados Unidos, tanto em acessos como em decisões. Durante os anos de 1990 a 1994, ao mesmo tempo em que a rede mundial crescia, a faixa de endereços mostrava sua insuficiência e o IPv6 era pensado, a governança da Internet evoluiu para uma gestão internacional (GRAZIANI, 2017).

Com a insuficiência de endereços, o novo protocolo foi pensado com endereços de 128 *bits*, ou seja, quatro vezes mais *bits* que o endereço do IPv4. O número de endereços disponíveis é $3,4 * 10^{38}$. Isso quer dizer $8 * 10^{28}$ endereços a mais que o IPv4 (PYLES; CARRELL; TITTEL, 2016).

O blog *We Are Social* publica anualmente uma estimativa anual dos usuários de Internet no mundo. Segundo Kemp (2020), 4,54 bilhões de pessoas usam a Internet, uma penetração de quase 59% da população mundial. Sendo assim, a disponibilidade de endereços é de cerca de $7,5 * 10^{28}$ endereços por usuário hoje conectado. Se projetarmos para a população mundial, 7,75 bilhões de pessoas (KEMP, 2020), seriam $4,3 * 10^{28}$ endereços por pessoa.

A estrutura do pacote IPv6 foi modificada em relação ao IPv4, para melhorar seu processamento. Um representação dos dois cabeçalhos é apresentada na Figura 7. O cabeçalho da nova versão possui 40 *bytes* de tamanho fixo, contra o tamanho variável entre 20 e 60 *bytes* na versão anterior. Além disso, vários campos foram removidos ou se tornaram opcionais (GRAZIANI, 2017).

Figura 7 - Comparação do cabeçalho IPv4 e IPv6



Fonte: Adaptado de Graziani (2017, p. 52).

Segundo Kurose e Ross (2012), o cabeçalho IPv6 possui a seguinte composição:

- Versão (4 *bits*): versão do protocolo IP utilizado, neste caso preenchido com 0110 (6 em binário);
- Classe de Tráfego (8 *bits*): identifica os pacotes por classes de serviços ou por prioridade;
- Identificador de Fluxo (20 *bits*): identifica pacotes permitindo a análise em camada 2 de fluxos de dados, estendendo as funções de QoS;
- Tamanho dos Dados (16 *bits*): tamanho em *bytes* dos dados enviados junto ao cabeçalho no pacote IPv6, somando-se os cabeçalhos de extensão;
- Próximo Cabeçalho (8 *bits*): indica que tipo de tráfego é esperado na porção de dados do pacote e no IPv6 também indica os valores dos cabeçalhos de extensão;
- Limite de Encaminhamento (8 *bits*): indica o número máximo de saltos que o pacote pode dar na rede antes de ser descartado, sendo decrementado a cada passagem por um roteador;
- Endereço de Origem (128 *bits*): endereço IPv6 de origem do pacote;
- Endereço de Destino (128 *bits*): endereço IPv6 de destino do pacote.

Verificados os elementos do cabeçalho, as diferenças em destaque estão dispostas na Tabela 1.

Tabela 1 - Diferenças entre os cabeçalhos IPv4 e IPv6

IPv4	IPv6	Condição	Explicativo
Campo “Tamanho do Cabeçalho”	-	Removido	O cabeçalho IPv6 possui tamanho fixo, inexistindo a necessidade de um campo específico para indicar seu tamanho
Campo “Tipo de Serviço”	Campo “Classe de Tráfego”	Renomeado	O campo tem função idêntica nas duas versões, embora tenha sido renomeado.
-	Campo “Identificador de Fluxo”	Adicionado	Campo preenchido na origem com valor aleatório entre 00001 e FFFFF, permitindo que os roteadores no percurso identifiquem o tráfego com pertencendo a um mesmo fluxo de dados. Pode estender as funções de QoS.
Campo “Tamanho Total”	Campo “Tamanho dos Dados”	Reformulado	O campo apresenta um valor *bytes* nas duas versões. No IPv4, devido ao tamanho do cabeçalho ser variável, era uma soma do tamanho do cabeçalho e do dados, sendo chamado de “Total”. No IPv6, devido ao tamanho do cabeçalho ser fixo, apresenta apenas o tamanho dos dados (sendo que cabeçalhos de extensão são somados no tamanho dos dados).
Fragmentação e Campos “Identificação”, “*Flag*” e “Deslocamento de fragmento”	Sem fragmentação no encaminhamento	Reformulado	Os campos que permitiam a fragmentação de um pacote IPv4 por um roteador intermediário, quando a unidade de transmissão máxima de um *link* era pequena em relação ao tamanho do pacote, foram removidos do IPv6, sendo tal “função” controlada apenas pela origem do tráfego. No IPv6, caso o roteador receba um pacote muito grande, ele o descarta e uma mensagem de “pacote muito grande” é devolvida a origem, que deve gerenciar o reenvio do pacote com o tamanho correto para que a comunicação seja completada.
Campo “Protocolo”	Campo “Próximo cabeçalho”	Renomeado/ Estendido	O campo ainda indica qual é o tipo de dado inserido na porção de dados do pacote (TCP, UDP, ICMP, etc), e recebeu novo nome, pois no IPv6 pode indicar a existência de cabeçalhos de extensão.

(continua)

Tabela 1 - Diferenças entre os cabeçalhos IPv4 e IPv6

(conclusão)

IPv4	IPv6	Condição	Explicativo
Campo “TTL”	Campo “Limite de encaminhamento”	Renomeado	Foi renomeado devido a implementação desse controle, que prevê o decréscimo de 1 unidade do valor a cada roteador percorrido. Tal implementação era feita também no IPv4, ou seja, a contagem era de saltos e não de tempo.
Campo “*Checksum*”	-	Removido	O campo foi removido, pois foi considerado uma redundância desnecessária. A camada 2 já possui um mecanismo de checagem e controle de erros, além dos mecanismos com o mesmo propósito presentes nos protocolos de camadas superiores a de rede (como o Checksum do cabeçalho TCP e o Checksum do cabeçalho UDP).
Endereços de origem e destino	Endereços de origem e destino	Alteração do tamanho	O tamanho dos campos no cabeçalho foi alterado para acomodar o novo tamanho dos endereços: 64 *bits*, utilizados por dois endereços IPv4; 256 *bits*, utilizados por dois endereços IPv6.
Campo “Options”	-	Removido	O campo “Options” é opcional no IPv4 e podia conter informações em implementações específicas em segurança ou rotas, por exemplo. Por possuir tamanho variável, devia ser completado com zeros (0) até o tamanho de 32 *bits*. O IPv6 pode acrescentar essas informações nos cabeçalhos de extensão.

Fonte: Adaptado de Graziani (2017).

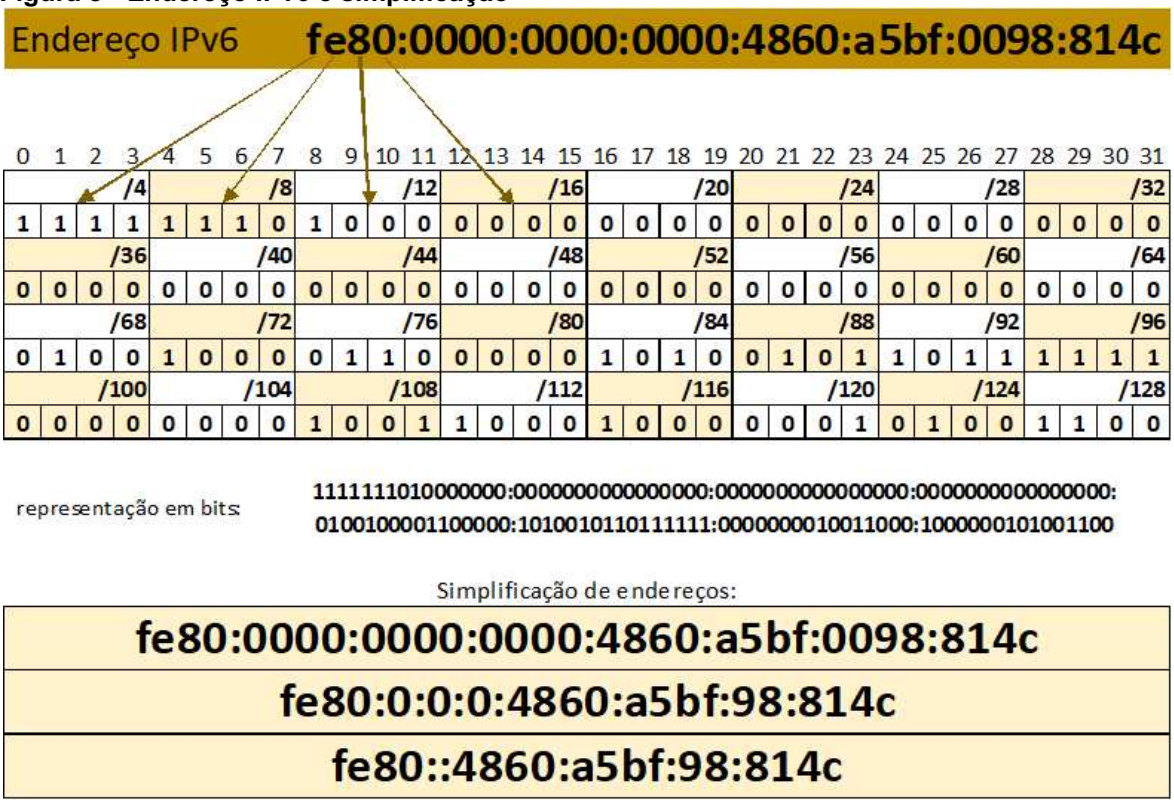
Com as mudanças e pensando principalmente no número de *bits* necessários para encaminhamento apenas dos endereços de origem e destino, um pacote IPv6 tem o tamanho de 40 *bytes*, enquanto um pacote IPv4 base, ou seja, sem opções adicionais, teria no mínimo 20 *bytes* de tamanho. Com uma reorganização do cabeçalho e apenas dobrando o tamanho do pacote foi possível aumentar em $7{,}9 * 10^{27}$ vezes o número de endereços disponíveis para conexão a Internet (KUROSE; ROSS, 2012).

Com 128 *bits*, a notação do endereço IPv6 com 8 grupos de 16 *bits*, separados por dois pontos (**:**), onde, portanto, cada grupo pode assumir um valor entre 0000 e FFFF, podendo usar letras maiúsculas ou minúsculas para o dígitos hexadecimais (COMER, 2014). O exemplo de notação é mostrado na Figura 8.

Para melhorar a representação dos endereços, é adotada uma simplificação de notação quando há grupos de zeros. Nos grupos de valores, os

zeros a esquerda não necessitam ser representados. Além disso, pode ser utilizado um par de dois pontos (**::**) para representar grupos de zeros consecutivos, atentando-se que essa simplificação somente pode ser realizada uma vez no endereço (COMER, 2014). Diferentes exemplos de simplificação da representação são mostrados na Figura 8.

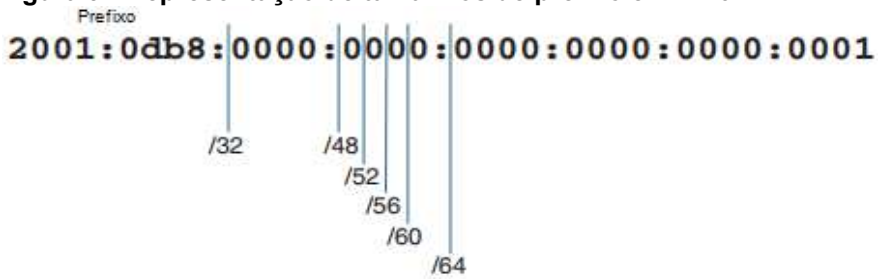
Figura 8 - Endereço IPv6 e simplificação



Fonte: Autoria própria.

O IPv6 não possui máscara, mas utiliza uma notação similar a CIDR, delimitando o tamanho das sub redes IPv6, como descrito na RFC 3513. Após a barra (/), é denotado o tamanho do prefixo, um valor decimal que especifica a quantidade de *bits* contíguos à esquerda que distingue o prefixo de um *range* de endereços IPv6 específico (GRAZIANI, 2017). Exemplos de diferentes tamanhos de prefixo são ilustrados na Figura 9.

Figura 9 - Representação de tamanhos de prefixo em IPv6



Fonte: Adaptado de Graziani (2017, p. 99).

Segundo Graziani (2017), outra diferença de funcionamento para o IPv4, é a inexistência de endereço *broadcast*, que é o responsável por encaminhar uma mensagem a todos os endereços de um mesmo domínio no IPv4. No IPv6, existem os seguintes tipos de endereços:

- Endereço *Unicast*: Endereço específico de uma interface de rede num dispositivo com suporte a IPv6. Esses endereços ainda podem ser subdivididos:
  - Endereços *unicast* globais: endereços roteáveis globalmente, sendo acessíveis de qualquer ponto da Internet IPv6. Inicialmente a IANA reservou 2000::/3 para alocação de endereços globais;
  - Endereços de *link-local*: endereço por meio do qual executa-se a comunicação entre nós pertencentes à mesma rede local, sendo essa faixa de endereçamento confinada no escopo de enlace dessa rede, ou seja, esses pacotes não serão encaminhados pelo roteador. Todo equipamento com IPv6 possui um endereço de *link-local* automaticamente configurado, mas que pode ser alterado manualmente. Os dez primeiros *bits* desse endereço são FE80;
  - Endereço de *loopback*: Definido como ::1 é reservado para teste de *loopback*, ou seja, os pacotes não são transmitidos, mas sim processados localmente como se fossem pacotes de entrada. O funcionamento é equivalente ao 127.0.0.1 em IPv4;
  - Endereço não-especificado: É um endereço com todos os *bits* zero, e não pode ser utilizado por uma interface. O endereço não-especificado pode ser utilizado como “endereço de origem” do pacote para representar ausência de endereço;
  - Endereço *unique-local*: Similar a endereços privados IPv4, estão na faixa fc00::/7. Endereços *unique-local* não são roteáveis na Internet,

mas podem ser utilizados em redes em áreas limitadas.

  - Endereços IPv4 compatíveis com IPv6: Consistiam em endereços em que os primeiros 96 *bits* seriam zero e os 32 *bits* finais um endereço IPv4. Esta faixa seria utilizada durante a transição do IPv4 para IPv6, porém sua utilização dessa forma foi descontinuada.

- Endereço *Multicast*: Endereços utilizados quando é necessária comunicação de um *host* encaminhando um pacote para vários destinos em simultâneo.
- Endereço *Anycast*: Não existente no IPv4, o endereço *anycast* é atribuído a mais de uma interface, desta forma, se vários equipamentos possuírem o mesmo endereço, um pacote encaminhado será entregue ao equipamento mais próximo com aquele endereço, considerando o roteamento.

Os aspectos de funcionamento do IPv6 promoveram mudanças no *Internet Control Message Protocol* (ICMP) que era utilizado no IPv4. O protocolo modificado passou a ser chamado ICMPv6 (COMER, 2014). No IPv6 o uso do ICMPv6 é necessária para garantir o funcionamento das funções básicas do protocolo, notificando erros e apresentando mensagens informativas sobre a rede. O ICMPv6 incorporou as funcionalidades do *Address Resolution Protocol* (ARP), que mapeia endereços na camada 2, e do *Internet Group Management Protocol* (IGMP), que gerencia grupos de *multicast*, do IPv4 (GRAZIANI, 2017).

O cabeçalho ICMPv6 é inserido no pacote logo após ao cabeçalho principal do IPv6 ou cabeçalhos de extensão, se existirem, e possuem os seguintes campos (SANTOS *et al.*, 2014):

- Tipo (8 *bits*): determina o tipo da mensagem, influenciando a leitura do restante;
- Código (8 *bits*): fornece informações adicionais para alguns tipos mensagens;
- Soma de verificação - *Checksum* (16 *bits*): dá possibilidade a detecção de dados corrompidos no cabeçalho ICMPv6 e em parte do cabeçalho IPv6;
- Dados (tamanho variável): apresenta as informações de diagnóstico e erro de acordo com o tipo. No caso de uma mensagem de erro, o pacote que originou a mensagem pode estar contido na mensagem, desde que o

tamanho total do pacote ICMPv6 não exceda o *Maximum Transmission Unit* (MTU) mínimo do IPv6 de 1280 *bytes*.

O *Network Discover Protocol* (NDP), é uma das funcionalidades do ICMPv6, para encontrar roteadores, descobrir *hosts* vizinhos conectados na mesma sub rede e seus endereços na camada de enlace e manter informações sobre o roteamento (SANTOS *et al.*, 2014).

A auto configuração, uma das funcionalidades do IPv6, permite aos dispositivos adquirir informações da rede, do enlace e de endereçamento de forma autônoma para um dispositivo que ingressa em uma sub rede. A auto configuração pode acontecer de forma *stateless* ou *stateful* (PYLES; CARRELL; TITTEL, 2016).

O método *stateless*, também chamado pela sigla SLAAC (*Stateless Address Autoconfiguration*) decorre do equipamento que fornece as informações de configuração não manter registro do estado e das características do nó do destinatário. O pacote de informações de auto configuração, chamado de *Router Advertisement* (RA), engloba propriedades do enlace, da rede, de prefixos, de DNS, MTU e outros que ao serem recebidas e usadas pelos dispositivos para se configurarem, sendo o endereço de *unicast* global gerado e depois confirmado na rede para que não haja duplicidade (SANTOS *et al.*, 2014).

O método *stateful*, utiliza um DHCPv6 fornecendo todas as informações de configuração para um dispositivo, como ocorre no DHCP para IPv4. Nesse tipo de configuração, o servidor DHCPv6 poderá manter registro de todos os *hosts* por ele configurados, e por isso são usados em ambientes de rede que a controle dos dispositivos e seu endereçamento. Neste método ainda ocorre disseminação pelo roteador de pacote *Router Advertisement*, porém este conterá uma instrução para que o dispositivo solicite a um servidor DHCPv6 as informações de conexão (GRAZIANI, 2017).

Métodos de configuração podem utilizar servidor *stateless* e *stateful* em paralelo, assim as informações para configuração do endereço *unicast* global do *host* pode ser feito com informações do roteador e informações de *Network Time Protocol* (NTP), DNS e outros serviços podem ser fornecidos pelo DHCPv6. Isso é feito com fornecimento no *Router Advertisement* de instrução para que informações extras sejam solicitadas ao servidor *stateful* (SANTOS *et al.*, 2014).

4.1 TÉCNICAS DE TRANSIÇÃO (IPV4 PARA IPV6)

O desenvolvimento do IPv6 foi feito de forma que sua interoperação com IPv4 não é possível diretamente, tanto pela estrutura de endereços como pelas demais características que fizeram ele diferente da versão anterior. Ainda assim, com o encapsulamento, uma alteração nos pacotes em alguma camada superior não mudaria totalmente o funcionamento, o que permite que equipamentos funcionem em paralelo com IPv4 e IPv6 quando consideramos a camada 2, e apenas alterações para reconhecimento do novo pacote quando na camada 3 (SANTOS *et al.*, 2014; GRAZIANI, 2017).

Isso permite, por exemplo, que um *switch* que opera apenas encaminhando tráfego na camada 2 do modelo OSI, encaminhe pacotes IPv4 e IPv6 sem que seja necessárias alterações. Roteadores e equipamentos finais (como um computador, *notebook* ou impressora), por exemplo, só poderão funcionar com os dois protocolos em paralelo caso tenham suas aplicações estejam preparadas para tal (GRAZIANI, 2017). Essas considerações ajudam a entender a evolução dos métodos de tunelamento e tradução que surgiram para apoiar o método principal de pilha dupla.

A coexistência dos protocolos IPv4 e IPv6 é indispensável durante o período de transição, já a substituição imediata da rede completa seria inviável por sua proporção e tamanho. Acrescenta-se a isso o fato de a coexistência dos protocolos vai ocorrer sem prazo estipulado, já que a migração da rede para o novo protocolo pode demorar muito tempo e também a aproximação do esgotamento de endereços no protocolo antigo (SANTOS *et al.*, 2014).

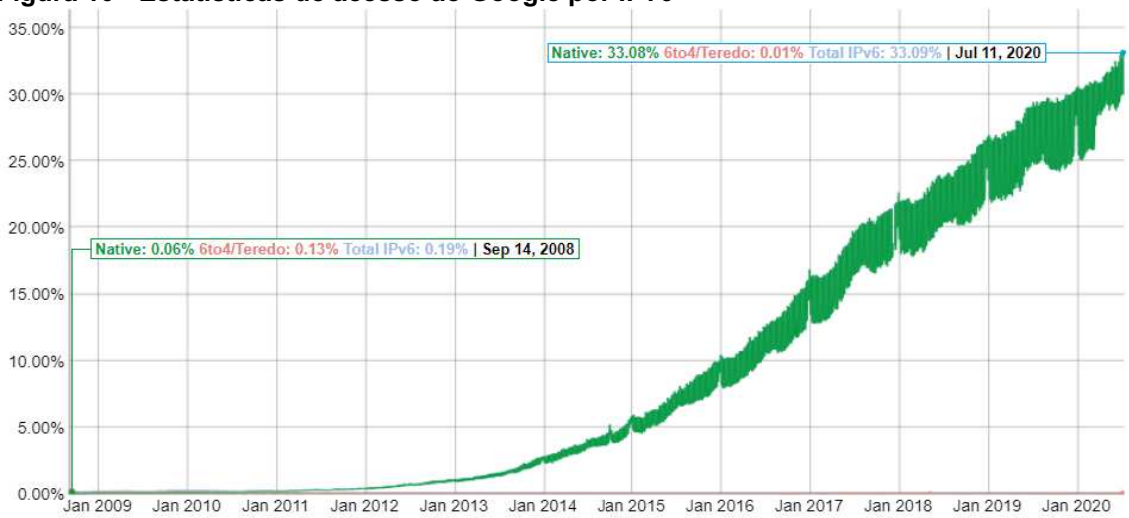
Devido a existência de diversas técnicas de transição, cada responsável pela migração de sua rede deve avaliar qual técnica será melhor implantada naquela realidade, mas algumas recomendações são preferenciais (SANTOS *et al.*, 2014), como:

- Utilizar nativamente IPv6 pelos usuários (túneis IPv6 dentro de IPv4 são preteridos em relação a túneis IPv4 dentro de IPv6, por exemplo);
- Utilizar técnica *stateless* em vez de *stateful*;
- Evitar que se prolongue a utilização do IPv4, sem adoção concomitante do IPv6;

- Analisar o nível de maturidade e opções de implementação da técnica de transição.

Com esgotamento do IPv4 aliado aos variados métodos de transição, a adoção gradual do IPv6 cresce, como pode ser visto no gráfico de comparação de acessos IPv6 no Google, reproduzido na Figura 10.

Figura 10 - Estatísticas de acesso ao Google por IPv6



Fonte: Adaptado de Google (2020).

4.1.1 Pilha Dupla

A técnica de pilha dupla, permite que os roteadores e dispositivos possam encaminhar, enviar e receber pacotes de ambas as versões do protocolo, IPv4 e IPv6. Um dispositivo receberá configuração, automaticamente ou manualmente, para as duas redes, como os endereços, por exemplo, e assim fará as comunicações em IPv4 quando em contato com outro dispositivo IPv4 ou em IPv6 quando em contato com dispositivo que já suporta também a comunicação IPv6 (SANTOS *et al.*, 2014).

Esse método, permite que no futuro, caso seja desativada a comunicação usando IPv4, o protocolo possa simplesmente ser desativado nos segmentos de rede que deixarem de operar nele (SANTOS *et al.*, 2014).

Como ilustrado por Santos *et al.* (2014), embora possam utilizar a mesma interface na pilha dupla, o IPv4 e IPv6 funcionam como redes separadas, então as configurações de pilha dupla também devem ser realizadas nos outros serviços e equipamentos que darão suporte ao funcionamento da rede. Alguns exemplo, são:

- Um roteador terá de receber configuração de rota estática para que encaminhe o tráfego IPv6 ou mudança no protocolo de roteamento para um que seja compatível com IPv4 e IPv6 simultaneamente;
- Um servidor DNS deverá receber registros do tipo AAAA, para que resolva domínios em endereços IPv6;
- Listas de controle de acesso e/ou regras de firewall têm de receber regras específicas para controlar o tráfego IPv6.

Na pilha dupla a decisão pela comunicação por IPv4 ou IPv6 ficará por conta das aplicações clientes e servidoras. Mas se as duas estão disponíveis, tem de haver uma forma de selecionar a qual versão será dada preferência na comunicação. Algumas implementações são usadas para que o tráfego seja direcionada pela versão do IP mais recente ou mais antiga, uma delas é a RFC6555, denominada *Happy Eyeballs* (IPV6.BR, 2012a).

A implementação do *Happy Eyeballs* é descrita de forma sucinta no site do IPV6.BR (2012a):

> Seu funcionamento consiste em tentar se conectar às duas conexões simultaneamente e utilizar aquela que é estabelecida mais rapidamente, dando uma leve preferência para a conexões IPv6. *Browsers* como Google Chrome e Mozilla Firefox em suas versões atuais já implementam o *Happy Eyeballs* e o utilizam por padrão.

A sugestão do algoritmo *Happy Eyeballs*, prevê que o código da aplicação deve se comportar fazendo as duas conexões com uma pequena margem de diferença na ordem de milisegundos entre a conexão IPv6, realizada primeiro, e a tentativa com a conexão IPv4. Assim, se a conexão IPv6 falha, o intervalo de tempo da falha é praticamente imperceptível ao usuário. E se a conexão IPv6 funciona, a segunda conexão, IPv4, não será efetuada, evitando um excesso de conexões que duplicaria o número de *sockets* utilizados. Por fim, e para que esse teste duplo não seja feito continuamente, o algoritmo prevê um cache de no máximo 10 minutos, em que o status de falha da conexão IPv6 é guardado, assim as conexões neste tempo serão feitas preferencialmente em IPv4. Findo o prazo do cache, o teste com a conexão dupla é realizado novamente, para evitar vícios de escolha do protocolo (IPV6.BR, 2012b).

4.1.2 Tunelamento

Os métodos de tunelamento estão disponíveis para conexões que não possuem acesso à rede IPv6 em pilha dupla. As técnicas de tunelamento fazem o encapsulamento de pacotes IPv6 dentro de pacotes IPv4 para que seja possível oferecer o novo protocolo em locais onde ele ainda não está acessível diretamente (IPV6.BR, 2012b). Como outras técnicas de transição, os túneis devem ser aplicados como um mecanismo provisório até que o IPv6 possa ser oferecido de forma nativa (GRAZIANI, 2017). A Tabela 2 descreve as técnicas de tunelamento mais comuns.

Tabela 2 - Técnicas de tunelamento mais comuns

Tunnel Broker	O *Tunnel Broker* permite que uma pequena rede ou um *host* isolado obtenha conexão IPv6 através da rede IPv4 e é descrito na RFC3053. Para realizar a conexão através desse tipo de túnel, é necessário se cadastrar em um provedor que ofereça essa conexão. Normalmente este provedor possui um cliente de conexão para o túnel ou *script* de configuração. Os túneis podem utilizar tecnologias diferentes para prover o acesso, como encapsulamento em UDP, protocolo *Anything In Anything* (AYIYA) ou *Tunnel Setup Protocol* (TSP). Provedores de *tunnel broker* costumam oferecer faixas de IPv6 variando entre /48 e /64.
ISATAP	Sigla para *Intra-Site Automatic Tunnel Addressing Protocol* e é definida pela RFC5214. Não é oferecido por serviços externos, como o *Tunnel Broker*. Sua utilização é feita por organizações que já possuem IPv6 disponível e entregue no roteador de perímetro pelo ISP, mas a infraestrutura interna limita sua implementação na rede interna. Com ISATAP, *hosts* podem se conectar ao roteador por túnel na rede IPv4. Os pacotes são levados encapsulados até o roteador, que os desempacota e trafega-os em IPv6 na rede externa.
Teredo	A técnica Teredo foi proposta pela Microsoft e definida na RFC4380. A técnica utiliza encapsulamento UDP dos pacotes IPv6 que são transmitidos pela 3544 até um servidor Teredo. O destaque para esta técnica está na possibilidade de atravessar um ou mais NAT. A técnica tem diversas considerações quanto a segurança, já que o tráfego pode passar despercebido por *firewalls* e ser interceptado em servidores Teredo inseguros, por exemplo. Além disso, o funcionamento é complexo e tem alta taxa de falhas e atrasos.
Túnel GRE	GRE (*Generic Routing Encapsulation*) é uma técnica de transição proposta pela Cisco, e definida na RFC2784. Um túnel GRE consiste numa conexão segura ponto a ponto e tem como diferencial a possibilidade de transportar outros protocolos além do IP. Para encaminhar o pacote original, o túnel GRE adiciona o cabeçalho IPv4 e um cabeçalho GRE. O pacote IPv4 recebe no campo “Protocolo” o valor “47” e no cabeçalho GRE o transporte de um pacote IPv6 é denotado pela *word* “86DD” no campo “Protocolo”, já que o protocolo não é tunelamento apenas para IPv6.

Fonte: Adaptado de IPV6.BR (2012b) e Graziani (2017).

4.1.3 Tradução

Alguns métodos de transição envolvem a tradução de um pacote de uma versão do IP para outra. Os métodos de tradução têm como objetivo oferecer acesso transparente a parte da rede não disponível na versão do protocolo adotado na porção de *hosts* da rede (GRAZIANI, 2017).

Uma das técnicas mais características do tipo tradução é o NAT64, a RFC6146. Com o NAT64 é possível oferecer uma rede conectada apenas com IPv6 e, através da tradução de endereço, conexão à rede IPv4, num processo parecido com o NAT para IPv4 da RFC1918. Como numa consulta para obtenção do acesso IPv6 não haverá resposta dos servidores DNS com um registro AAAA, é necessário utilizar uma segunda técnica para auxiliar a tradução de registros DNS, chamado de DNS64, a RFC6147 (SANTOS *et al.*, 2014).

A comunicação com um serviço disponível apenas em IPv4 para um *host* IPv6 que está em uma rede com NAT64 começa com o envio da requisição de DNS pelo dispositivo. O servidor DNS64 consultará outro serviço de DNS autoritativo para descobrir qual o endereço IPv4 do domínio consultado, assim poderá fazer a conversão do endereço para IPv6 adicionando o prefixo adotado pela rede para tradução e encaminhando o endereço IPv6 composto como resposta para o *host*. O *host* encaminhará o pacote IPv6, tendo como endereço de destino a resposta recebida pelo DNS, para o roteador/NAT64 (SANTOS *et al.*, 2014).

O roteador, conhecendo o prefixo configurado para NAT64, fará a conversão do pacote para IPv4, adaptando o cabeçalho. Em seguida, encaminha o pacote traduzido através da interface IPv4 a sua disposição, guardando a informação de porta de serviço. Ao receber a resposta na mesma porta, o roteador converterá o pacote IPv4 para um pacote IPv6 e devolverá a resposta ao *host* de origem (SANTOS *et al.*, 2014).

A vantagem do NAT64 é o acesso transparente para os usuários finais, além do fato da implementação já ser muito madura e contar com diversas opções de software e hardware que a oferecem. Uma das desvantagens do NAT64 é o fato da técnica ser *stateful* e a possibilidade de haver problemas na tradução de endereço, que podem comprometer o funcionamento da rede (SANTOS *et al.*, 2014).

4.2 SOBRE A IMPLANTAÇÃO DE IPV6 EM REDES JÁ EM OPERAÇÃO COM IPV4

Graziani (2017), lista diversas considerações a serem realizadas por um administrador de redes quando da necessidade ou opção pela implementação de IPv6 em uma rede. Como o escopo de implementação neste trabalho está voltado a rede interna de uma organização, entre as considerações avaliadas por Graziani (2017) foram separados:

- Esquema de endereçamento;
- VLANs com IPv6;
- Pilha dupla;
- IPv6 no *data center*;
- DNS.

A distribuição de blocos IPv6 a organizações normalmente é feita destinando uma faixa /48 de IPv6. Uma faixa dessa contém 65.536 redes IPv6 /64 que podem ser utilizadas. Uma disponibilidade tão grande de endereços de sub-redes, permite a adoção de esquemas de endereçamento onde se pode utilizar a numeração completa de *VLANs* na notação do endereço ou adotar o endereço IPv4 inteiro dentro do endereço IPv6. Estas facilidades e a possibilidade de organizar a rede de forma hierárquica dada pela estrutura de endereço, podem ajudar a manter o esquema de endereçamento simples ou fazê-lo complexo, de acordo com a necessidade da rede (GRAZIANI, 2017).

Num ambiente com diversas *VLANs*, a configuração pode ser executada de forma que facilite diagnósticos de problemas no futuro, definindo um endereço de *link-local* personalizado para a interface da *VLAN* configurada no *switch layer 3*, além disso, a interface também tem o papel de enviar as mensagens de *Router Advertisement* para aquela *VLAN*, fornecendo os dados de configuração para os *hosts* conectados (GRAZIANI, 2017).

O funcionamento da rede IPv6 em pilha dupla não depende apenas da configuração dos ativos de rede (*switches*, roteadores, pontos de acesso), mas também de haver suporte nos equipamentos finais dos usuários, com sistemas operacionais compatíveis e aplicações preparadas para utilização de IPv6 (GRAZIANI, 2017). Neste sentido, uma tabela compilada por diversos usuários

fazendo a comparação do suporte ao IPv6 em diferentes sistemas operacionais mostra que os diversos sistemas passaram a ter suporte a IPv6 com qualidade de produção, mesmo que com algum grau de limitação, em meados do ano de 2011 (WIKIPEDIA, 2020).

A disponibilidade de IPv6 num *data center* vai além de simplesmente ter a rede configurada nos *switches* e roteadores, o assunto permeia a disponibilização dos serviços ali hospedados também em IPv6. Ainda que as redes funcionem paralelamente, questões de desempenho podem aparecer quando em interação na utilização de *Storage Access Networks* (SAN), serviços em nuvem, *cache* e outras aplicações específicas (GRAZIANI, 2017).

Os serviços de DNS passam a ter um papel ainda mais importante com a adoção do IPv6, considerando o tamanho do endereço e seu formato hexadecimal. Um nome de domínio pode ser mapeado para um mais endereços, sejam eles IPv4 ou IPv6. Da mesma forma, a resposta de um servidor DNS pode ser encaminhada com ambos os endereços ainda que este servidor responda apenas na interface IPv4 (GRAZIANI, 2017).

5 A IMPLEMENTAÇÃO DE IPV6 NA REDE DO CÂMPUS CURITIBA DA UTFPR

Este capítulo está divido em três partes, sendo a primeira apenas complementar para conceitos que não abrangem os protocolos IP mas são essenciais para entendimento do relato de implementação. As segunda parte descreve as motivações para início da implementação e a terceira parte apresenta o relato da implementação e estado atual do uso do IPv6 no Câmpus Curitiba da UTFPR.

5.1 FIREWALL, DMZ E PORTAL DE AUTENTICAÇÃO

A rede do Câmpus Curitiba da UTFPR faz uso de roteador de perímetro que aplica diversos recursos de *firewall*, aplica o conceito de DMZ e, por força de regulamento, restringe o acesso a Internet com base na autenticação dos usuários utilizadores. Os serviços são melhores conceituados abaixo.

Segundo Moraes (2015), *firewall* é um conjunto de recursos de hardware e software que atua entre a rede pública e privada ou entre sub-redes privadas controlando o tráfego de entrada e saída, fazendo com que o tráfego passe livremente ou seja bloqueado, podendo ainda registrar este tráfego. Esse controle é feito a partir de regras pré-determinadas ou aprendendo padrões observados da própria rede.

Com aplicação de regras de *firewall* no controle da rede interna e externa é possível criar vários níveis de segurança, sendo um deles a denominada Zona Desmilitarizada (DMZ). A DMZ é uma sub-rede ou porção da rede física, na qual estão conectados os serviços de rede que serão disponibilizados para a acesso através da Internet e normalmente possui um nível de segurança diferente da rede interna na qual estão conectados *hosts* de usuários (Moraes, 2015), evitando expor a rede de usuários da mesma forma que a rede com os serviços. A DMZ pode possuir regras diferenciadas para tratar o tráfego de rede proveniente da rede interna e da rede externa (SHRIMALI, 2017).

Uma dos recursos de um *firewall* pode ser o controle dos equipamentos e usuários conectados a rede de acesso. Existem diversas formas de realizar esse controle, uma delas é com um portal de autenticação (*Captive Portal*). Numa configuração com portal de autenticação, o *firewall* pode encaminhar um comando

ao *host* (geralmente em conexões HTTP) redirecionando o tráfego original do usuário para um página na qual ele deve fornecer credenciais de autenticação para que, a partir de então, o tráfego que foi redirecionado e os demais acessos posteriores sejam permitidos ou bloqueados, dependendo do nível de permissão daquela credencial (CISCO SYSTEMS, 2018). Esse recurso pode ser usado em implementação própria ou em interagindo com outros protocolos que realizam autenticação de usuários, como o IEEE 802.1x, *Microsoft Active Directory*, *Lightweight Directory Access Protocol* (LDAP) ou *Remote Authentication Dial In User Service* (RADIUS).

5.2 MOTIVAÇÕES PARA IMPLEMENTAÇÃO

Embora a maior motivação para implementação do protocolo IPv6 seja o esgotamento de endereços, a realidade na UTFPR quanto a escassez de endereços IPv4 não se assemelha a realidade mundial. Embora não existam endereços para que todos os *host* conectados a Internet possuam IPv4 público, todos os *hosts* que eventualmente precisam estar visíveis e acessíveis através da Internet fora da rede interna, possuem endereço IPv4 atribuído. Além disso, existem blocos /24 inteiros de endereços IPv4 não utilizados atribuídos a instituição.

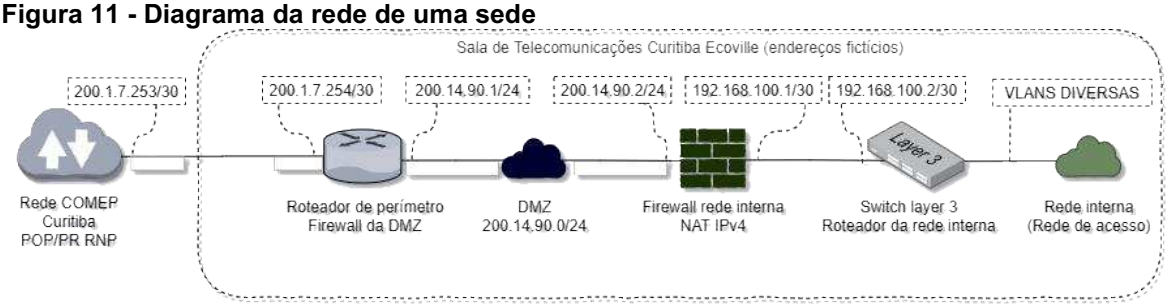
Desta forma, o principal motivo para a implantação do IPv6 no Câmpus Curitiba da UTFPR é a solicitação dos docentes e pesquisadores interessados na utilização do novo protocolo nas salas de aula e laboratórios. Além disso, providenciar acesso às novas redes IPv6 implementadas pelo mundo é necessário como preparo antecipado, já que é esperado o esgotamento de endereços IPv4 no mundo e com isso a migração para o protocolo IPv6 seja acelerado nos próximos anos.

Outra motivação é a disponibilização dos serviços hospedados também em IPv6, pois eles estão apenas configurados para IPv4. Servidores Web, de e-mail, de bancos de dados, de VoIP, etc. passarão estar disponíveis para todas as conexões que também já estiverem preparadas para o IPv6 na Internet.

5.3 PADRONIZAÇÕES DE CONFIGURAÇÃO E ESQUEMA DE ENDEREÇAMENTO

Dentro da rede do Câmpus Curitiba, a DMZ é composta pela faixa de rede IPv4 público /24 e tem escopo de sede, já que o Câmpus possui três sedes. Há um roteador de perímetro/*firewall* entre a DMZ da sede e a WAN. A rede de acesso de cada sede possui endereços IPv4 privados e há um roteador/*firewall* entre ela e a DMZ, uma das funções deste segundo roteador é a de fazer a tradução de endereços (NAT). A Figura 11 ilustra a rede de uma sede, as redes das demais sedes seguem esquemas lógicos idênticos.

Figura 11 - Diagrama da rede de uma sede



Fonte: Autoria própria.

Uma das padronizações adotadas consiste na *tag* das *VLANs* e distribuição de endereços nas sedes, blocos e racks.

A *tag* das *VLANs* de acesso a Internet segue a seguinte padronização:

- Três ou quatro dígitos;
- O primeiro dígito pode ser 0 (ou desconsiderado), caso a *VLAN* seja de serviços de TI gerenciados pela Coordenadoria de Gestão de Tecnologia da Informação (COGETI), nos demais casos, o primeiro dígito é o número da sede:
  - 1 para Sede Centro;
  - 2 para Sede Ecoville;
  - 3 para Sede Neoville.
- O segundo e terceiro dígitos são utilizados para o número do bloco:
  - 01 para Bloco A;
  - 02 para Bloco B;
  - 03 para Bloco C;

  - …;
  - 11 para Bloco K;
  - 12 para Bloco L;
  - etc.
- O quarto e último dígito é utilizado para representar o *rack* de conexão (com um dígito podendo variar de 0 a 9, os blocos possuem até 10 *VLANs* disponíveis para segmentar a distribuição de conexões).

As *VLANs* de acesso por rede sem fio, as *VLANs* de serviço e a *VLAN* de administração da rede seguem uma numeração própria definida pela COGETI, desvinculada do padrão, com três dígitos e semelhante nas três sedes. As *tags* neste caso são pensadas para que fiquem fora da faixa das *VLANs* de acesso e não haja confusão com blocos e racks.

Os endereços IPv4 seguem uma padronização baseada no número das *VLANs*:

- Adotou-se, para a rede de acesso a Internet, a faixa 10.0.0.0/8, variando de acordo com a *VLAN*:
  - O primeiro octeto é 10;
  - O segundo octeto adota o número do bloco;
  - O terceiro octeto adota o número do *rack*;
  - O quarto octeto é reservado aos *hosts*, sendo:
    - Final 1 reservado ao roteador;
    - Finais 2 a 9 reservado para necessidades da COGETI;
    - Finais 11 a 29 reservado para solicitações de endereço IP fixo;
    - Finais 30 a 254 distribuído dinamicamente pelo servidor DHCP.
  - Não há variação de acordo com a sede, já que as redes de acesso não se comunicam diretamente de uma sede a outra via Rede privada virtual (VPN).
- As redes de serviço possuem endereços dentro da faixa 172.16.0.0/12;
  - As redes variam o segundo octeto conforme o serviço (16, 17, 18, etc.);
  - O terceiro octeto varia de acordo com a sede, visto da necessidade dessas faixas de rede se comunicarem via VPN e, para tal

funcionalidade, ser necessário endereços diferentes na criação das rotas.

- A administração da rede possui endereço 192.168.1.0/24, 192.168.2.0/24 e 192.168.3.0/24, de acordo com a sede.

Essa distribuição padronizada de *VLANs* e endereços de rede tem como justificativa principal a facilitação da compreensão do funcionamento pelos técnicos responsáveis pela administração e operação da rede.

Seguindo o modelo de padronização adotado no IPv4, distribuir as redes da faixa IPv6 recebida para cada sede (/48) foi fácil, pois foi possível adotar o número completo da *VLAN* no endereço, mantendo a mesma segmentação de rede e a lógica de distribuição. Manteve-se assim o princípio que facilita a compreensão da rede.

O padrão de endereçamento IPv6 adotado foi:

- Os primeiros 48 *bits* foram aqueles recebidos pelo Câmpus Curitiba quando as faixas foram distribuídas pela Diretoria de Gestão de Tecnologia da Informação (DIRGTI) da Reitoria;
- Os 16 *bits* seguintes correspondem ao número da *VLAN* (como definido anteriormente);
- Os demais 64 *bits* são destinados aos *hosts*, sendo:
  - O primeiro endereço, final ::1, é reservado ao roteador;
  - Os endereços com final de ::2 até ::fff reservados para necessidades da COGETI;
  - Os endereços de ::1000 até ::ffff reservados para solicitações de IP fixo;
  - Os demais endereços são usados pelos *hosts* de forma dinâmica para conexão.

Dois exemplos aplicados do esquema de padronização, com *tags* de *VLAN*, faixa de endereçamento IPv4 e IPv6 são representados na Figura 12, com base na sede, bloco e *rack* daquela rede.

Figura 12 - Exemplo de padronização: VLAN e endereços IPv4 e IPv6

Exemplos:

Local:	Sede Centro (1)
Bloco:	N (14)
Rack:	3

Tag da VLAN:	
1143	
Faixa de endereço IPv4:	
10.14.3.0/24	
Faixa de endereço IPv6:	
2001:db8:4:1143::/64	

Conforme a padronização:

Local:	Sede Neoville (3)
Bloco:	A (1)
Rack:	1

Tag da VLAN:	
3011	
Faixa de endereço IPv4:	
10.1.1.0/24	
Faixa de endereço IPv6:	
2001:db8:f:3011::/64	

Conforme a padronização:

Fonte: Autoria própria.

Ainda que as faixas de endereços verdadeiras possam ser facilmente encontradas com utilização de ICMP, preferiu-se omitir neste descritivo os endereços IPv4 públicos e IPv6 utilizados no Câmpus, adotando-se endereços fictícios para os exemplos, quando necessários.

5.4 RELATO DE IMPLEMENTAÇÃO

Com a definição dos endereços e com a facilidade de ter a rede IPv6 entregue em pilha dupla na conexão principal de cada sede do Câmpus Curitiba, foi decido pela utilização em pilha dupla da conexão também na DMZ e LAN. Com isso, também não é necessário preocupar-se com interferência direta no funcionamento da rede IPv4 enquanto se disponibiliza a rede IPv6.

A disponibilização em pilha dupla, que se mostrava a melhor escolha, somente passou a ser possível no Câmpus Curitiba após a aquisição de um novo *switch* principal de distribuição para cada sede, para trabalhar o roteamento (*layer* 3) entre as *VLANs*. Após a aquisição do novo equipamento, já com suporte a IPv6, as funcionalidades *layer* 3 para IPv6 foram completamente contempladas, o que não acontecia no equipamento antigo.

A implementação em fase experimental foi iniciada fazendo a configuração das interfaces e rotas no roteador de perímetro (Apêndice A), sempre observando se a rede em ambiente de produção permanecia funcional da mesma forma de antes de qualquer modificação.

A interface de cada roteador de perímetro recebeu endereço IPv6 de um segmento /126 e rota estática para saída (Apêndice A), sendo essa rota para o

roteador da Reitoria, para a conexão a WAN na sede Centro, e do Ponto de presença no Paraná (POP-PR) da RNP, no caso da conexão a WAN nas sedes Ecoville e Neoville. Nas sedes Ecoville e Neoville, após a configuração, o roteador principal teve conectividade direta com o gateway padrão, que era o roteador do POP-PR RNP. No caso do roteador da Reitoria, houve sucesso apenas após as configurações realizadas pela equipe de TI do órgão, pois ainda não havia disponibilidade completa de IPv6 naquela infraestrutura.

O próximo passo foi realizar a implementação na rede interna (Apêndice B), visto que a conexão com a WAN já estava funcional. Para evitar a necessidade de mudar conexões fisicamente, decidiu-se transpor as duas camadas de *firewall* existentes na rede (na saída da DMZ para a WAN e na saída da rede acesso para a DMZ) usando rotas. Sendo assim duas redes IPv6 /126 foram criadas fazendo a ligação entre o roteador de perímetro e o roteador da rede de acesso e entre o roteador da rede de acesso e *switch layer 3* da rede de acesso. As rotas necessárias para aplicação desta forma de conexão foram criadas.

Com as conexões nas três sedes, partindo da WAN, funcionais até o *switch layer 3* da rede de acesso, o equipamento recebeu a configuração de cada uma das interfaces das *VLANs* do qual é gateway (Apêndice B). A configuração seguiu a padronização de número de VLAN, conforme descrito no tópico anterior e é ilustrada na Figura 13.

Figura 13 - Exemplo de configuração de uma interface

```
RouterCentro#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
RouterCentro(config)#interface Vlan1143
RouterCentro(config-if)#description BlocoN_Rack3
RouterCentro(config-if)#ipv6 address 2001:db8:4:1143::1/64
RouterCentro(config-if)#ipv6 enable
RouterCentro(config-if)#ipv6 nd other-config-flag
RouterCentro(config-if)#ipv6 dhcp server default-dns-domain
RouterCentro(config-if)#exit
RouterCentro(config)#
```

Fonte: Autoria própria.

Com as interfaces configuradas, era necessário definir o método de distribuição de endereços dentro da rede no caso do protocolo IPv6, já que a rede IPv4 já contava com DHCP configurado para todas as redes de acesso. A opção por um DHCPv6 *stateful* chegou a ser cogitada, pois assim o switch poderia registrar os logs de todas as respostas de requisição retornadas a cada dispositivo

da rede, como já faz com o DHCP no IPv4, armazenando assim o log. Porém, com a facilidade da configuração *stateless* usando o *Router Advertisement* (RA) do NDP a preferência foi dada a auto-configuração. Acrescentou-se o anúncio via DHCPv6 das configurações de DNS e domínio. Para manter o registro dos endereços de cada nó, optou-se por utilizar um servidor de log em arquivo já disponível na infraestrutura, acrescentando um script para que com frequência de 30 minutos seja despejado em arquivo o conteúdo da tabela de *neighbors* descobertos pelo NDP armazenados pelo roteador das *VLANs*. A Figura 13 contém também os comandos para configuração do DHCP.

Com as configurações de interfaces, rotas e endereçamento realizada para todos os equipamentos de rede principais, e estando a estrutura de *VLANs* já difusa na infraestrutura e funcionando com IPv4, procedeu-se os testes pontuais, ainda com o tráfego geral bloqueado. Os testes começaram com verificação da comunicação entre as redes, utilizando apenas o roteamento das redes internas em uma das sede, permitindo verificar inclusive as configurações de Listas de Controle de Acesso (ACL) IPv6, ajustando redes que não deveriam manter comunicação direta entre si através do *switch layer 3*.

O teste com sucesso permitiu avançar para a análise do funcionamento do tráfego entre as redes internas e a rede externa. Para isso, foi inserido regra de firewall que libera a faixa usada pela administração de rede para que se pudesse chegar até *hosts* externos. O acesso foi satisfatório e não foi notado nenhuma queda de desempenho registrada nos gráficos e estatísticas do roteador/*firewall* que gerencia o tráfego.

Para completar a implementação, seria necessária a ativação do portal de autenticação (*captive portal*) para a rede IPv6, como já ocorre na rede IPv4. A utilização de *captive portal* no Câmpus é determinada por regulamentos internos da instituição, ou seja não pode deixar de estar ativo. Este método é limitado às redes que não possuem outra forma de identificação dos usuários; a wireless é um exemplo de exceção, pois possui identificação dos usuários pelo protocolo IEEE 802.1x e armazenamento de logs no servidor RADIUS. Uma das medidas utilizadas para manter a segurança das credenciais do usuário a utilizar o portal de autenticação, é a disponibilidade deste portal apenas em HTTPS e com certificado SSL válido e para isso é configurado um nome de domínio para as páginas de autenticação.

O primeiro teste para ativação do portal de autenticação foi realizado com entradas de DNS do tipo A e AAAA iguais, fwauth.ct.utfpr.edu.br, resultando em um problema no qual o navegador sempre direcionava a conexão para o protocolo IPv6, sendo assim nunca era possível estabelecer autenticação do endereço IPv4, devido a atuação do algoritmo *Happy Eyeballs*.

A solução desse problema foi a adoção de nomes de domínio distintos para evitar que o algoritmo *Happy Eyeballs* selecione automaticamente qual versão do IP seria autenticada. Os domínios utilizados foram fwauth.ct.utfpr.edu.br e fwauth6.ct.utfpr.edu.br. Assim caso usuário chegue a página de autenticação redirecionado através de uma conexão IPv4, seria enviado ao portal de autenticação respondendo no endereço IPv4; com o usuário chegando a página de autenticação redirecionado através de uma conexão IPv6, seria enviado ao portal de autenticação no endereço IPv6.

Com nomes de domínios distintos, o funcionamento da página de autenticação nas duas versões do IP mostrou-se ainda problemática, pois o serviço de autenticação da solução de *firewall* existentes no Câmpus permitia a seleção de apenas um certificado SSL, sendo assim, os navegadores apresentavam página informando que o certificado era inválido, pois não correspondia a um dos domínios, IPv4 ou IPv6, dependendo do acesso e do certificado carregado no momento de configuração do serviço de autenticação.

Para a solução deste problema foi necessário solicitar a emissão de um certificado SSL *wildcard*, que pode ser válido para diversos subdomínios diferentes de um mesmo domínio (HOUSLEY *et al.*, 1999). Carregando o serviço de autenticação com este certificado, os dois domínio (fwauth.ct.utfpr.edu.br e fwauthct6.ct.utfpr.edu.br) possuíam certificado SSL válido e a premissa de utilização de HTTPS estava atendida.

Uma das configurações utilizadas no serviço de autenticação, exigia que o usuário mantivesse aberta em seu navegador uma página de manutenção de autenticação (*keep alive*) que com frequência de 200 segundos, era recarregada, enviando um comando ao serviço para que mantivesse a sessão daquele usuário ativa. Caso em 600 segundos, esse comando não fosse recebido pelo serviço, isso poderia ser interpretado como se usuário tivesse deixado de usar a conexão que havia estabelecido, propositalmente ou acidentalmente, e, por segurança, essa conexão deveria ser encerrada. A página ainda conta com um botão de

desconexão deliberada, que o usuário pode acionar a qualquer momento, sendo sua sessão descontinuada.

A decisão por uso desta página, aconteceu principalmente devido a grande troca de usuários em laboratórios de informática, que acontece devido a forma como as aulas são distribuídas na universidade. Com página de *keep alive*, uma autenticação feita num laboratório poderia ser encerrada em pouco tempo, caso o usuário deixasse de fazer a utilização de um computador fechando a página sem um clique no botão de desconexão e ao mesmo tempo, um funcionário de setor administrativo que não compartilha um computador poderia manter uma conexão ativa por várias horas, somente tendo o cuidado de manter a página de *keep alive* aberta.

A autenticação ficou disponível na rede de administração após a implementação do certificado *wildcard* em IPv4 e IPv6, sem ocorrência de novas falhas. Os usuários eram identificados em ambos os endereços em momentos diferentes, dependendo da disponibilidade de versão do IP do *website* que se desejava acessar. Nesta fase foi percebida a dificuldade com a página de *keep alive*. Na utilização de pilha dupla, embora os protocolos estejam disponíveis em paralelo, a conexão IPv4 e IPv6 são distintas entre si e tratadas em paralelo pelo serviço de autenticação. Sendo assim, o usuário deveria manter duas abas abertas em seu navegador, cada qual com a página de *keep alive* de uma das sessões autenticadas em uma das versões do protocolo. Os próprios usuários da rede de administração, que são membros da equipe de TI do Câmpus estavam confundindo as páginas de *keep alive* em duplicidade, fechando e derrubando suas próprias sessões.

Uma abordagem diferente teve de ser utilizada modificando a configuração para que a página de *keep alive* não fosse necessária. Para isso, foi modificado o modo de controle da sessão no serviço de autenticação para que expirasse após um tempo em que a sessão não produz tráfego. Agora, após a autenticação via navegador, uma tela confirmando o sucesso é devolvida ao usuário e nessa tela ainda está um botão para desconexão, porém ela pode ser fechada sem que a sessão expire.

Com todos os problemas encontrados durante os testes sanados, era o momento de disponibilizar o acesso em toda a rede, pois embora todos os enlaces entre a WAN, DMZ e rede de acesso e os *hosts* pudessem se configurar

automaticamente com as funções do NDP, o tráfego IPv6 estava totalmente bloqueado pelo *firewall* entre a rede de acesso e a DMZ. Desta forma a conexão IPv4 era sempre preferida pelas aplicações dos equipamentos conectados.

Foram replicadas as regras no roteador/*firewall* da rede interna de IPv4 para IPv6, ou seja, como a solução de *firewall* separa as regras entre as versões do protocolo, cada regra para IPv4 foi copiada, recebendo uma regra IPv6 equivalente. Com isso o tráfego de todas as redes em IPv6 foi liberado. Foi constatado pelos gráficos e logs que o tráfego estava fluindo pelos dois protocolos. Apenas para complementação da próxima análise, destacamos que nessa fase a liberação foi feita para todas as redes, independente da existência de autenticação por *captive portal* ou por IEEE 802.1x.

Infelizmente, a exigência de autenticação por *captive portal* na rede cabeada se mostrou ruim para o funcionamento da implementação de pilha dupla em questões de usabilidade. Durante o primeiro dia de testes, foram registrados sete chamados falando sobre falta de funcionamento da rede. Os chamados descreviam situações diversas em que não era possível navegar de forma satisfatória envolvendo o procedimento de autenticação.

Considerando estes chamados até a desativação das regras, pudemos identificar importantes situações:

- em três chamados era mencionado o fato da tela solicitando informações de usuário e senha ser mostrada repetidas vezes;
- em dois chamados os usuários, encontrando a tela de autenticação pela segunda vez, finalizaram a primeira sessão antes de autenticar na outra versão do protocolo, achando que deveria ser feito dessa forma e assim encerrando a sessão no primeiro protocolo;
- em um dos chamados um docente escreveu que em seu computador funcionava, mas os estudantes reclamavam da autenticação em suas estações e por isso não conseguiu ministrar sua aula com sucesso;
- no último chamado o usuário afirmava acreditar estar sendo atacado por um vírus e não preencheria usuário e senha uma segunda vez, pois já havia autenticado.

Nos testes e atendimentos com esses chamados, foi identificado a dificuldade dos usuários em assimilar o fato de haverem duas redes distintas, cada uma com sua autenticação.

Com sucesso técnico da implementação seguido dos problemas de usabilidade que geraram reclamações, foi optado pela desativação das regras que liberavam o tráfego IPv6 sainte com destino a Internet gerados pela faixa de rede cabeada, onde o *captive portal* está ativado. Mantiveram-se as regras para liberação do tráfego IPv6 sainte com destino a Internet provenientes da *rede sem fio*, pois, com validação do usuário realizada pelo 802.1x no momento do acesso à rede, ambos os IP, v4 e v6, já estão registrados para aquele usuário autenticado na controladora dos pontos de acesso sem fio.

Mesmo com a implementação parcial do IPv6 na rede, foram adaptadas as configurações de alguns servidores que estavam na DMZ para que funcionassem em pilha dupla. Os serviços foram: DNS, servidor do sistema de atendimento da equipe de TI e o servidor de arquivos compartilhados.

Por fim, ainda está sendo testado uma possível solução para o problema da necessidade de dupla autenticação, utilizando a técnica de transição NAT64 já que a implementação utilizando esse método está disponível para ativação na solução de *firewall* utilizada atualmente no Câmpus Curitiba, sendo o procedimento de ativação e configuração nesta solução relativamente simples. Os testes foram interrompidos durante a dispensa das atividades presenciais ocorrida em virtude da pandemia de COVID-19.

6 CONSIDERAÇÕES FINAIS

O protocolo IPv6 traz diversas vantagens para todos os usuários da Internet no mundo, embora a transição exija treinamento para os profissionais de rede e custos com configuração e substituição de equipamentos. Isso permitirá a continuidade da evolução e expansão da rede de redes que é a Internet.

As técnicas de transição estão bem maduras e a difusão do protocolo nos equipamentos que são utilizados por ISP já saem de fábrica com o suporte ao novo protocolo. Assim também nos dispositivos dos usuários, já que todos os sistemas operacionais atuais e com grande penetração de mercado possuem suporte ao protocolo IPv6.

A configuração para uso do novo protocolo no Câmpus Curitiba da UTFPR se tornou viável com a modernização dos equipamentos na borda da rede, como o roteador de perímetro, *firewall* e *switch layer 3*. A implementação das configurações foi satisfatória e como esperado, não interferiu no funcionamento da rede IPv4. Com isso, foi executada a demanda de conexão IPv6, como era solicitada pela RNP as diversas instituições de ensino, incluindo a UTFPR. Como demonstrado durante o trabalho, foi feita utilizando a prática de pilha-dupla, melhor opção de implementação do protocolo.

Por fim, o uso contínuo do protocolo na rede da UTFPR Curitiba ainda não atingiu seu potencial total, pois a necessidade de autenticar os usuários em duas redes trouxe dificuldades de usabilidade. Ainda assim, nas redes possíveis, o tráfego IPv6 está liberado e adequadamente roteado.

Uma alternativa para aprimorar a usabilidade da autenticação com os dois protocolos usando NAT64, está em teste e pode ser alvo de novo estudo de caso. Também poderá ser abordado em novo trabalho uma medição de desempenho da rede comparativa entre IPv4 e IPv6 nos ambientes da UTFPR Curitiba.

Indo além dos objetivos, a adequação de todos os serviços/servidores hospedados na DMZ para que trabalhem em pilha dupla ainda é necessária, já que os serviços que receberam a configuração tinham como objetivo dar um mínimo suporte a rede IPv6 de acesso que foi objeto deste estudo.

REFERÊNCIAS

CERF, V. G. **Keynot**. 2011. Disponível em: <http://www.archive.org/download/Linux.conf.au2011-KeynoteVirtCerf/LCA2011-Vint_Cerf.webm>. Acesso em: 14 abr. 2020.

CETIC.BR. **TIC domicílios**. 2018. Disponível em: <https://cetic.br/pt/pesquisa/domicilios/>. Acesso em: 15 fev. 2020.

CISCO SYSTEMS. **Configurando o portal prisioneiro nos accesspoint WAP351 e WAP371**. 2018. Disponível em: <https://www.cisco.com/c/pt_br/support/docs/smb/wireless/cisco-small-business-300-series-wireless-access-points/smb5044-configuring-captive-portal-on-the-wap351-and-wap371-access-p.pdf>. Acesso em: 08 jul. 2020.

COMER, D. E. **Internetworking with TCP/IP**. 6. ed. Pearson, 2014.

FULLER, V.; LI, T. **RFC4632**: Classless Inter-Domain Routing (CIDR): The internet address assignment and aggregation plan. Internet Engineering Task Force (IETF), 2006. Disponível em: <https://tools.ietf.org/html/rfc4632>. Acesso em: 07 jul. 2020.

GOOGLE. **IPv6 statistics**. 2020. Acesso: <https://www.google.com/intl/en/ipv6/statistics.html#tab=ipv6-adoption>. Acesso: 12 jul. 2020.

GRAZIANI, R. **IPv6 fundamentals**: A straightforward approach to understanding IPv6. Cisco Press, 2017.

HOUSLEY, R. *et al.* **RFC2459**: Internet X.509 public key infrastructure certificate and CRL Profile. 1999. Disponível em: <https://tools.ietf.org/html/rfc2459>. Acesso em: 09 jul. 2020.

ICANNWIKI. **Regional internet registry**. 2015. Disponível em: <https://icannwiki.org/Regional_Internet_Registry>. Acesso em: 10 jun. 2020.

IANA. **Number resources**. [2013?]. Internet Assigned Numbers Authority (IANA). Disponível em: <https://www.iana.org/numbers>. Acesso em: 10 jun. 2020.

IPV6.BR (Brasil). **Happy eyeballs**. 2012a. Disponível em: <http://ipv6.br/post/happy-eyeballs/>. Acesso em: 07 jul. 2020.

IPV6.BR (Brasil). **Transição**. 2012b. Disponível em: <http://ipv6.br/post/transicao/>. Acesso em: 15 fev. 2020.

KEMP, S. **Digital 2020**: 3.8 billion people use social media. Special reports. We Are Social, 2020. Disponível em: <https://wearesocial.com/blog/2020/01/digital-2020-3-8-billion-people-use-social-media>. Acesso em: 10 jul. 2020.

KUROSE, J. F; ROSS, K. W. **Computer networking**: a top-down approach. 6. ed. Pearson, 2012.

LACNIC. **Fases de Esgotamento do IPv4**. 2020. Disponível em: <https://www.lacnic.net/1077/3/lacnic/fases-de-esgotamento-do-ipv4#tabs-2>. Acesso em: 06 jul. 2020.

MORAES, A. F. **Firewalls: Segurança no controle de acesso**. 1. ed. São Paulo: Érica, 2015. 120 p.

POPOVICIU, C. **Deploying IPv6 networks**. Cisco Press, 2006.

PYLES; CARRELL; TITTEL, J.; CARRELL, J. L.; TITTEL, E. **Guide to TCP/IP**: IPv6 and IPv4. Nelson Education, 2016.

SANTOS, R. R. dos; *et al.* **Curso IPv6 básico**. 3. ed. Rio de Janeiro: RNP/ESR, 2014.

SHRIMALI, S. **DeMilitarized Zone**: Network Architecture for Information Security. Int. J. Comput. Appl, v. 174, n. 5, p. 16-19, 2017.

TANENBAUM, A. S. **Redes de computadores**. 4. ed. Rio de Janeiro, RJ: Campus, 2003. XXI, 923 p.

WIKIPEDIA. **Comparison of IPv6 support in operating systems**. Wikipedia, The Free Encyclopedia. 2020. Disponível em: <https://en.wikipedia.org/wiki/Comparison_of_IPv6_support_in_operating_system>. Acesso em: 08 jul. 2020.

APÊNDICE A: COMANDOS DE CONFIGURAÇÃO PARA O ROTEADOR DE PERÍMETRO E FIREWALL DE BORDA DA REDE INTERNA

```
//INÍCIO COM CONFIGURAÇÃO DAS INTERFACES JÁ ATIVAS COM IPV4, PARA ENDEREÇAMENTO
//IPV6 - Endereços são sempre fictícios

config system interface
  edit "port2"
    set vdom "PerimetroDMZ"
    config ipv6
      set ip6-address 2001:db8::5/64    //ENDEREÇANDO INTERFACE GATEWAY DA DMZ
    end
  next
  edit "VLAN9"
    set vdom "PerimetroDMZ"
    config ipv6
      set ip6-address 2001:db8:4:a000::1/126  //ENDEREÇANDO INTERFACE DA VLAN COM
                                              //INTERFACE COM IPV4 PUBLICO DA DMZ
                                              //USADA PARA O NAT
    end
    set interface "x1"
    set vlanid 9
  next
  edit "VLAN9-Gw"
    set vdom "FirewallNAT"
    set alias "Gw_NAT"
    set role wan
    config ipv6
      set ip6-address 2001:db8:4:a000::2/126  //ENDEREÇANDO INTERFACE DA VLAN COM
                                              //INTERFACE COM IPV4 PUBLICO USADA
                                              //PELO NAT IPV4. NO IPV6 TEM FUNÇÃO
                                              //APENAS DE FIREWALL
    end
    set interface "x2"
    set vlanid 9
  next
edit "VLAN4"
    set vdom "FirewallNAT"
    set alias "CORE_cisco"
    set role lan
    config ipv6
      set ip6-address 2001:db8:4:b000::1/126  //ENDEREÇANDO INTERFACE DA VLAN DE
                                              //LIGAÇÃO DO FIREWALL DA REDE
                                              //PRIVADA COM O SWITCH LAYER 3
    end
    set interface "x2"
    set vlanid 4
  next
end
```

```
//CONFIGURAÇÃO DAS ROTAS DO ROTEADOR DE PERÍMETRO/FIREWALL DA DMZ

config vdom
edit PerimetroDMZ
config router static6
  edit 1
    set gateway 2001:db8::1            //ROTA PARA GATEWAY PADRÃO DA DMZ
    set device "port2"
    set distance 1
  next
  edit 2
    set dst 2001:db8:4:b000::0/126
    set gateway 2001:db8:4:a000::2     //ROTA PARA A REDE INTERNA IPV6 DE ACESSO
    set device "VLAN9"
    set distance 1
  next
  edit 3
    set dst 2001:db8:4:1000::/52
    set gateway 2001:db8:4:a000::2     //ROTA PARA A REDE INTERNA IPV6 DE ACESSO
    set device "VLAN9"
    set distance 1
  next
end


//CONFIGURAÇÃO DAS ROTAS DO ROTEADOR DE PERÍMETRO/FIREWALL DA DMZ

config vdom
edit FirewallNAT
config router static6
  edit 1
    set gateway 2001:db8:4:a000::1     //ROTA PARA GATEWAY PADRÃO DA REDE INTERNA
    set device "VLAN9-Gw"
    set distance 1
  next
  edit 2
    set dst 2001:DB8:4:1000::/52
    set gateway 2001:db8:4:b000::2     //ROTA PARA A REDE INTERNA, DIRECIONADA
                                       //AO SWITCH LAYER 3
    set device "VLAN4"
    set distance 1
  next
end
```

APÊNDICE B: COMANDOS DE CONFIGURAÇÃO PARA O SWITCH LAYER 3 DA REDE INTERNA

```
//CONFIGURAÇÃO DO SWITCH LAYER 3, COM INTERFACES E DHCP (AS ACLS FORAM OMITIDAS)
//OS COMANDOS NO CISCO SEGUEM COM ATIVAÇÃO DE IPV6, CONFIGURAÇÃO DE POOL DHCPV6
//CONFIGURAÇÃO INDIVIDUAL DAS INTERFACES E, POR FIM, DEFINIÇÃO DA ROTA PADRÃO
//Endereços são sempre fictícios

RedeInternaL3>enable
Password:
RedeInternaL3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
RedeInternaL3(config)#ipv6 unicast-routing
RedeInternaL3(config)#ipv6 dhcp pool default-dns-domain
RedeInternaL3(config-dhcpv6)#dns-server 2001:DB8:4:B000::1
RedeInternaL3(config-dhcpv6)#dns-server 2001:4860:4860::8888
RedeInternaL3(config-dhcpv6)#domain-name ct.utfpr.edu.br
RedeInternaL3(config-dhcpv6)#exit
RedeInternaL3(config)#interface Vlan4
RedeInternaL3(config-if)#description Gw_RedeInterna
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:B000::2/126
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd ra suppress all
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan990
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1990::1/64
RedeInternaL3(config-if)#ipv6 address FE80::990:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan991
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1991::1/64
RedeInternaL3(config-if)#ipv6 address FE80::991:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd ra suppress all
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan992
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1992::1/64
RedeInternaL3(config-if)#ipv6 address FE80::992:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd ra suppress all
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan993
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1993::1/64
RedeInternaL3(config-if)#ipv6 address FE80::993:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan994
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1994::1/64
RedeInternaL3(config-if)#ipv6 address FE80::994:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd ra suppress all
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan995
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1995::1/64
RedeInternaL3(config-if)#ipv6 address FE80::995:1 link-local
RedeInternaL3(config-if)#ipv6 enable
```

```
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan996
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1996::1/64
RedeInternaL3(config-if)#ipv6 address FE80::996:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd ra suppress all
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan997
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1997::1/64
RedeInternaL3(config-if)#ipv6 address FE80::997:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd ra suppress all
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan998
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1998::1/64
RedeInternaL3(config-if)#ipv6 address FE80::998:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan999
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1999::1/64
RedeInternaL3(config-if)#ipv6 address FE80::999:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd ra suppress all
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1010
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1010::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1010:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1011
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1011::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1011:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1012
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1012::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1012:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1013
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1013::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1013:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1020
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1020::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1020:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
```

```
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1021
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1021::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1021:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1022
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1022::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1022:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1023
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1023::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1023:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1030
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1030::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1030:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1031
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1031::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1031:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1032
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1032::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1032:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1033
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1033::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1033:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1040
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1040::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1040:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1041
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1041::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1041:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
```

```
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1042
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1042::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1042:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1043
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1043::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1043:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1044
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1044::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1044:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1045
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1045::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1045:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1046
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1046::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1046:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1047
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1047::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1047:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1048
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1048::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1048:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1049
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1049::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1049:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1050
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1050::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1050:1 link-local
RedeInternaL3(config-if)#ipv6 enable
```

```
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1051
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1051::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1051:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1052
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1052::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1052:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1060
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1060::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1060:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1070
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1070::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1070:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1090
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1090::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1090:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1091
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1091::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1091:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1110
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1110::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1110:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1111
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1111::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1111:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1120
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1120::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1120:1 link-local
```

```
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1121
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1121::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1121:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1122
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1122::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1122:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1123
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1123::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1123:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1124
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1124::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1124:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1140
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1140::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1140:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1141
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1141::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1141:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1142
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1142::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1142:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1143
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1143::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1143:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1144
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1144::1/64
```

```
RedeInternaL3(config-if)#ipv6 address FE80::1144:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1160
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1160::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1160:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1170
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1170::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1170:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1171
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1171::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1171:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1172
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1172::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1172:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1173
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1173::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1173:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1200
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1200::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1200:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1220
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1220::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1220:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1221
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1221::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1221:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1230
```

```
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1230::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1230:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1231
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1231::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1231:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1232
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1232::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1232:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1900
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1900::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1900:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#interface Vlan1980
RedeInternaL3(config-if)#description Wireless
RedeInternaL3(config-if)#ipv6 address 2001:DB8:4:1980::1/64
RedeInternaL3(config-if)#ipv6 address FE80::1980:1 link-local
RedeInternaL3(config-if)#ipv6 enable
RedeInternaL3(config-if)#ipv6 nd other-config-flag
RedeInternaL3(config-if)#ipv6 dhcp server default-dns-domain
RedeInternaL3(config-if)#exit
RedeInternaL3(config)#ipv6 route ::/0 2001:DB8:4:B000::1
RedeInternaL3(config)#exit
RedeInternaL3#wr
Building configuration...
[OK]
RedeInternaL3#exit
```