UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

CURSO DE ESPECIALIZAÇÃO EM CONFIGURAÇÃO E GERENCIAMENTO DE SERVIDORES E EQUIPAMENTOS DE REDES

FRANCIELE HERNANDES

AMBIENTE DE SIMULAÇÃO – IMPLEMENTANDO DNS E SERVIDOR WEB UTILIZANDO A TÉCNICA DE TRANSIÇÃO PILHA DUPLA

MONOGRAFIA

CURITIBA

2014

FRANCIELE HERNANDES

AMBIENTE DE SIMULAÇÃO – IMPLEMENTANDO DNS E SERVIDOR WEB UTILIZANDO A TÉCNICA DE TRANSIÇÃO PILHA DUPLA

Monografia apresentada como requisito parcial para a obtenção do grau de Especialista em Configuração e Gerenciamento de servidores e equipamentos de redes, do Departamento Acadêmico de Eletrônica da Universidade Tecnológica Federal do Paraná – UTFPR

Orientador: Prof. Msc. Lincoln Herbert Teixeira

CURITIBA

2014

RESUMO

HERNANDES, Franciele **Ambiente de simulação – Implementando DNS e Servidor Web utilizando a técnica de transição pilha dupla.** 2014. 89 f. Monografia (Especialização em Configuração e Gerenciamento de Servidores e Equipamentos de Redes). Universidade Tecnológica Federal do Paraná. Curitiba, 2014.

A presente monografia aborda a um ambiente de simulação de uma rede, utilizando o software GNS3, implementando uma das técnicas de transição que atualmente atende a uma rede real, chamada Pilha Dupla. Devido ao aumento do número de usuários de internet a distribuição de IPs da versão 4 está se esgotando. Além de apresentar alguns problemas que foram melhorados em uma nova versão. Surgiu o protocolo IPv6, garantindo o crescimento da internet e corrigindo alguns problemas da versão anterior, com um número de endereços muito maior que o IPv4. A técnica de transição escolhida, Pilha Dupla, permite que os protocolos IPv4 e IPv6 funcionem simultaneamente permitindo assim que a transição para a versão mais recente ocorra gradualmente. Além disso, também foi implementado o servidor DNS, para as duas versões do protocolo, simulando uma rede corporativa com um servidor *web.* Visando mostrar a importância do servidor DNS para a versão do protocolo IPv6, pois com o aumento do tamanho do endereço facilita qualquer configuração utilizada.

Palavras-chave: Redes. Transição. Pilha Dupla. Esgotamento de IPv4. DNS. Servidor *Web*.

ABSTRACT

HERNANDES, Franciele. **Simulation environment - Implementing DNS and Web Server using the Dual Stack transition technique.** 2014. 89 pages. Monograph (Specialization in Configuration and Management of Servers and Network Equipments) - Federal Technological University of Paraná. Curitiba, 2014.

This monograph deals with a network simulation environment, using GNS3 software, implementing one of transition techniques that currently serves a real network, called Dual Stack. Due to the increasing number of internet users the distribution of IP version 4 is running out. Besides presenting some problems that have been improved in the new version. Came the IPv6 protocol, ensuring the growth of the internet and correcting some problems of the previous version, with a number of much larger than IPv4 addresses. The chosen transition technique, Dual Stack, allows IPv4 and IPv6 protocols work simultaneously allowing the transition to the latest version to occur gradually. In addition, we also implemented the DNS server for the two versions of the protocol, simulating a corporate network with a web server. In order to demonstrate the importance of the DNS server to the version of the IPv6 protocol, because with increasing the size of the address facilitates any configuration used.

Keywords: Networks. Transition. Dual stack. IPv4 exhaustion. DNS. Web server. Priority, Mesh Topology. Wireless.

LISTA DE ILUSTRAÇÕES

Figura 1 - Cabeçalho do Protocolo IPv4. 12
Figura 2 - Mapa dos Registros Regionais de Internet. 14
Figura 3 - Cabeçalho do Protocolo IPv6. 17
Figura 4 - Cabeçalho dos Protocolos IPv4 e IPv6 e suas alterações. 21
Figura 5 - Funcionamento da Pilha Dupla. 29
Figura 6 - Topologia da Rede 31
Figura 7 - Topologia da Rede IPv4. 33
Figura 8 - Comandos configuração dos roteadores. 34
Figura 9 - Trecho do arquivo de configuração do roteador. 36
Figura 10 - Configuração do protocolo OSPF. 37
Figura 11 - Rotas. 38
Figura 12 - Base de dados do protocolo OSPF. 39
Figura 13 - Status e estatísticas de cada interface. 40
Figura 14 - Testes de Conexão. 41
Figura 15 - Teste de Conexão com traceroute. 42
Figura 16 - Topologia da Rede – IPv6. 43
Figura 17 - Configuração das Interfaces – IPv6. 44
Figura 18 - Arquivo de configuração – IPv6. 45
Figura 19 - Configuração OSPFv3. 47
Figura 20 - Capturado do pacote ping (ICMP) - IPv4, em vermelho, operando junto com o pacote ping (ICMPv6) – IPv6 em verde. 48
Figura 21 - Capturado do pacote ping (ICMPv6) – IPv6, em verde, operando junto com o pacote ping (ICMP) – IPv4 em vermelho. 49
Figura 22 - Arquivo named.conf.local. 51
Figura 23 - Configuração DNS direto. 52
Figura 24 - Configuração DNS reverso. 54
Figura 25 - Resultados do comando named-checkconf. 55
Figura 26 - Comando dig. 56
Figura 27 - Comando dig. 57
Figura 28 - Comando nslookup. 57
Figura 29 - Arquivo de configuração named.conf.local. 59
Figura 30 - Arquivo efmcorp.com.br.direto. 60
Figura 31 - Arquivo fd00:0000:0000:0001.reverso. 61
Figura 32 - Resultado do comando named-checkzone 1.0.0.0.0.0.0.0.0.0.0.0.0.0.d.f.ip6.arpa. 61
Figura 33 - Resultado comando dig www6.efmcorp.com.br aaaa. 62
Figura 34 - Comando dig –x fd00:0:0:1::2. 63
Figura 35 - Teste no cliente. 64
Figura 36 - Teste no cliente Windows. 65
Figura 37 - Arquivo do servidor Web. 66
Figura 38 - Acesso a página web. 67
Figura 39 - Acesso a página web. 68
Figura 40 - Acesso a página web. 69
Figura 41 - Acesso a página web. 70

SUMÀRIO

1 INTRODUÇÃO 7
1.1 TEMA 7
1.2 PROBLEMAS E PREMISSAS 8
1.3 OBJETIVOS 8
1.3.1 Objetivo Geral 8
1.3.2 Objetivos Específicos 9
1.4 JUSTIFICATIVA 9
1.5 PROCEDIMENTO METODOLÓGICO 9
1.6 EMBASAMENTO TEÓRICO 10
1.7 ESTRUTURA 10
2 REFERENCIAIS TEÓRICOS 11
2.1 INTERNET 11
2.2 PROTOCOLO IP 11
2.3 PROTOCOLO IPV4 12
2.4 PROTOCOLO IPV6 15
2.4.1 Tipos de Endereços 18
2.4.2 Endereços Especiais 19
2.4.2 Comparativo entre IPv4 e IPv6 19
2.6 DNS 22
2.6.1 Implementação do DNS 23
2.6.2 Aplicações para DNS 24
2.6.3 Principais tipos de registros 24
2.6.4 Servidor autoritativo 25
2.6.5 Servidor Recursivo 25
2.6.6 Principais Softwares 26
2.7 OSPFv3 26
2.8 MECANISMOS DE TRANSIÇÃO 27
2.8 PILHA-DUPLA 28
3 IMPLEMENTAÇÃO 31
3.1 CONFIGURAÇÕES DOS PROTOCOLOS DE INTERNET 33
3.1.1 IPv4 33
3.1.1.2 Configuração do Protocolo de Roteamento OSPF 36
3.1.2 IPv6 42
3.1.2.1 Configuração do Protocolo de Roteamento – OSPFv3 46
3.2 PILHA DUPLA 47
3.2 CONFIGURAÇÃO DO SERVIDOR DNS 49
3.2.1 Configuração do Servidor DNS – IPv4 49
3.2.2 Configuração do Servidor DNS – IPv6 58
3.3 CONFIGURAÇÃO DO SERVIDOR WEB 65
4 CONCLUSÃO 71
REFERÊNCIAS 72
APÊNDICE A – ARQUIVO DE CONFIGURAÇÃO DO ROTEADOR RT_A 74
APÊNDICE B – ARQUIVO DE CONFIGURAÇÃO DO ROTEADOR RT_B 78
APÊNDICE C – ARQUIVO DE CONFIGURAÇÃO DO ROTEADOR RT_C 82
APÊNDICE D – ARQUIVO DE CONFIGURAÇÃO DO ROTEADOR RT_D 86

1 INTRODUÇÃO

A internet está cada vez mais popular e acessível por diversos dispositivos eletrônicos. Para que tenham acesso a Internet necessita de um protocolo para acesso a internet chamado IP (*Internet Protocol*). Os IPs são únicos para cada dispositivo, e devido ao aumento do número de dispositivos móveis com acesso a Internet, a distribuição de IPs está praticamente se esgotando. A versão utilizada atualmente é o IPv4, que possibilita entorno de quatro bilhões de IPs, mas apresenta alguns problemas em relação a segurança. Para amenizar o esgotamento dos endereços IP, algumas soluções de curto prazo foram desenvolvidas como: a implantação dos endereços privados e a tradução de endereço de rede (*Network Address Translation* - NAT), mas logo será necessário implantar a nova versão do protocolo IP.

O IPv6, a sexta versão do protocolo IP, é a nova versão a ser implantada e mais segura, pois resolve os problemas relacionados a segurança, identificados no IPv4. Devido a necessidade de substituir a versão do protocolo, vária técnicas de transição surgiram, e a opção mais indicada é a chamada Pilha Dupla ou *Dual Stack*, que possibilita a habilitação dos protocolos IPv4 e IPv6 na mesma interface de rede, possibilitando a transição gradual para a nova versão do protocolo IP.

1.1 TEMA

Esse trabalho irá abordar a questão da transição da versão do protocolo IP implementado o método de transição Pilha Dupla, e também, analisar o funcionamento de um sistema de nomes de domínios, chamado DNS, que faz a tradução de um nome para um IP. Também será implementado um servidor *Web* funcionando nas duas versões do protocolo IP, utilizando um ambiente virtual,

simulador /emulador GNS3.

1.2 PROBLEMAS E PREMISSAS

Segundo o Centro de Estudos e Pesquisas em Tecnologia de Redes e Operações(2014), embora algumas soluções implementadas para resolver a questão do esgotamento de IPs, como o NAT, tenham diminuído a demanda por IPs, elas não foram suficientes para resolver os problemas decorrentes do crescimento da Internet. A adoção dessas técnicas reduziu uma pequena quantidade de blocos de endereços solicitados à IANA e a curva de crescimento da Internet continuava apresentando um aumento exponencial.

Essas medidas ajudaram para que houvesse mais tempo para o desenvolvimento de uma nova versão do IP, baseada nos princípios que fizeram o sucesso do IPv4, mas, que fosse capaz de suprir as falhas apresentadas por ele. Sugiram então os métodos de transição, que ajudaram para que a mudança ocorra gradualmente.

1.3 OBJETIVOS

1.3.1 Objetivo Geral

Criar um ambiente simulando uma rede corporativa, utilizando o método de transição de IPv4 para IPv6 - Pilha Dupla e implementação de um serviço de rede, no caso DNS operando em ambos os protocolos IP.

1.3.2 Objetivos Específicos

- Realizar a configuração dos equipamentos e do roteamento a ser utilizado no ambiente.
- Implementar e configurar os serviços de DNS, utilizando o Bind9 e servidor Web, utilizando Apache 2.2.
- Utilizar o OSPF como protocolo de roteamento nos roteadores.
- Utilizar o software GNS3 para simular o ambiente especificado.

1.4 JUSTIFICATIVA

Devido a possibilidade de alguns serviços não funcionarem e precisarem de modificações na transição do protocolo de acesso a internet de IPv4 para IPv6, pensou-se em realizar esse trabalho que tem como principal fundamento, mostrar que o serviço de gerenciamento de nomes, DNS, pode funcionar em Pilha Dupla assim como o servidor Web, Apache.

1.5 PROCEDIMENTO METODOLÓGICO

Primeiramente, deve ser realizada uma pesquisa bibliográfica para servir como base para dar início ao assunto a ser abordado, buscando vários livros sobre os assuntos específicos utilizados no trabalho. Após deve ser realizada uma pesquisa de caráter exploratório experimental sobre as novas tecnologias e principais tendências utilizadas, para caracterizar o problema o ser solucionado e elaborar o ambiente a ser simulado.

1.6 EMBASAMENTO TEÓRICO

O protocolo IPv6 possui endereços de 128 bits que permite a expansão do número de endereços para permitir o crescimento da internet. O IPv6 possui alguns mecanismos de autoconfiguração robustos, suporte à segurança e mobilidade, entre várias outras vantagens em relação ao IPv4. Apesar de vários benefícios a adoção a essa versão ainda é pouco representativa. Os protocolos IPv4 e IPv6 não são diretamente compatíveis, o que requer mecanismos de transição complexos para viabilizar a comunicação entre eles. Atualmente são poucos profissionais preparados para lidar operacionalmente com IPv6, o que indica que a demanda por esses profissionais tende e a crescer (BRITO, 2013).

1.7 ESTRUTURA

Esse trabalho é divido em 4 capítulos: introdução, referencial teórico, implementação e conclusão. No primeiro capítulo, introdução, será abordado o tema do projeto e os objetivos. No segundo capítulo, referencial teórico, aborda toda a pesquisa realizada sobre os temas abordados no projeto, desde o inicio do surgimento até os principais serviços a serem implementados. No terceiro capítulo, implementação, que apresenta os passos realizados para implementação do ambiente a ser simulado, a configuração dos equipamentos de redes, servidores e *hosts* dos clientes, a configuração do serviço escolhido para ser abrangido DNS e testes realizados. No capítulo de conclusão descreve os resultados obtidos, os problemas encontrados e como foram solucionados.

2 REFERENCIAIS TEÓRICOS

2.1 INTERNET

A Internet foi criada com objetivos militares, seria uma das formas das forças armadas norte-americanas de manter as comunicações em caso de ataques inimigos que destruíssem os meios convencionais de telecomunicações. Nas décadas de 1970 e 1980, a Internet também foi um importante meio de comunicação acadêmico. Somente no ano de 1990 que a Internet começou a alcançar a população em geral. Tim Bernes-Lee desenvolveu a *World Wide Web*, possibilitando a utilização de uma interface gráfica e a criação de sites. Foi então a partir deste momento que a Internet cresceu em ritmo acelerado (SUA PESQUISA, 2014).

Surgiram vários navegadores (*browsers*), provedores de acesso e portais de serviços *on-line* que contribuíram para este crescimento. A Internet passou a ser utilizada por vários segmentos sociais. Mas já nessa época os primeiros problemas estruturais do protocolo IPv4 ficaram evidentes, como escalabilidade em virtude do endereçamento limitado e falta de suporte nativo à segurança de aplicações sigilosas (BRITO, 2013, p.23).

As redes de computadores cresceram em tamanho, complexidade e banda. Com isso os equipamentos e aplicativos de redes tiveram que evoluir rapidamente para suportar o tráfego da rede devido ao aumento do número de usuários.

2.2 PROTOCOLO IP

O protocolo IP, o principal protocolo de comunicação da Internet, é o responsável por endereçar e encaminhar os pacotes que trafegam pela rede mundial de computadores. Os dados numa rede IP que são enviados em blocos, chamados

pacotes. Pacotes são os blocos de informações enviados na Internet e são divididos em duas partes: o cabeçalho, que, como um envelope, possui as informações de endereçamento da correspondência, e dados, que é a mensagem a ser transmitida. Cada pacote recebe um cabeçalho IP contendo uma série de campos de controle, dentre eles, o endereço IP de origem e destino e, a cada roteador no caminho, o endereço de destino é verificado e o pacote encaminhado para o próximo salto no caminho (PISA, 2014).

2.3 PROTOCOLO IPV4

Segundo o Centro de Estudos e Pesquisas em Tecnologia de Redes de Operações (CETPRO.BR) (2014) o cabeçalho IPv4 é composto por 12 campos fixos, como pode ser visualizado na figura 1, que podem ou não conter opções responsáveis por fazer com que o tamanho varie de 20 a 60 Bytes.

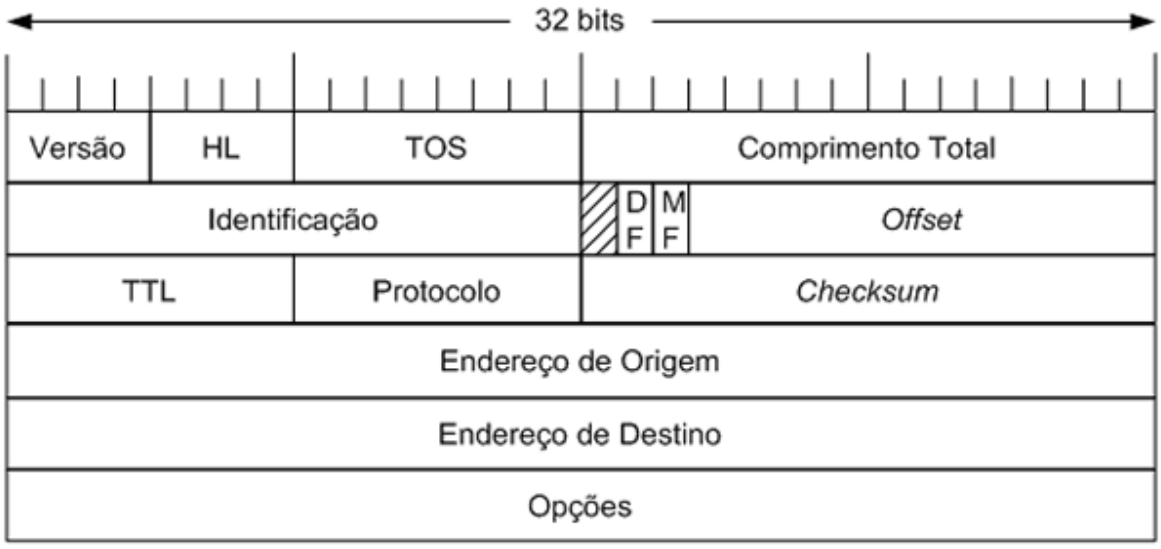


Figura 1 - Cabeçalho do Protocolo IPv4.
Fonte: http://s.glbimg.com/po/tt/f/original/2012/05/07/7011-cabecalho-ip.png

Segundo Centro de Estudos e Pesquisas em Tecnologia de Redes de Operações (CETPRO.BR) (2014), os campos são destinados para transmitir informações sobre:

- Versão do Protocolo;
- Tamanho do Cabeçalho e dos Dados;
- Fragmentação dos Pacotes;
- Tipo dos Dados Enviados;
- Tempo de Vida do Pacote;
- Protocolo da Camada Seguinte (TCP, UDP, ICMP);
- Integridade dos Dados;
- Origem e Destino do Pacote.

O IPv4 é o atual protocolo de comunicação que permite a comunicação entre os diversos dispositivos conectados à Internet, sejam eles computadores, *tablets* ou celulares. São reservados 32 bits para um endereço numérico (IP), gerando mais de quatro bilhões de endereços para máquinas. As faixas de endereços começadas com "10", com "192.168" ou com de "172.16" até "172.31" são reservadas para uso em redes locais e por isso não são usados na internet.

Na época de seu desenvolvimento esta quantidade de endereços era considerada suficiente para identificar todos os computadores na rede e suportar novas sub-redes. No entanto, com o rápido crescimento da Internet, ocasionou a escassez dos endereços IPv4, motivando a criação de uma nova geração do protocolo IP. No entanto, o protocolo IP em sua versão atual (a versão quatro, rotulada como IPv4) apresenta muitos problemas. Os mais graves são falhas de segurança, que periodicamente são descobertas e não têm solução. Além disso, tem um problema ainda mais premente do que sua inerente insegurança: já esgotou sua capacidade de expansão.

Segundo o Centro de Estudos e Pesquisas em Tecnologia de Redes de Operações (CETPRO.BR) (2014), o IANA (*Internet Assigned Numbers Authority)* é

responsável pela distribuição de todos os números IPs e, atualmente, ele realiza suas operações através da ICANN (*Internet Corporation for Assigned Names and Numbers*). A responsabilidade sobre uma parte dos endereços é delegada pela IANA para cada um dos Registros Regionais de Internet, que os gerenciam e distribuem dentro de suas respectivas regiões geográficas, figura 2. Em nossa região, o responsável é o LACNIC (*Latin America and Caribbean Network Information Centre*).

Figura 2 - Mapa dos Registros Regionais de Internet.
Fonte: IANA.

A IANA fez um padrão de divisão dos IP's em três classes principais para evitar ao máximo o desperdício de endereços, mas foi esse motivo que contribuiu para o esgotamento.

ENDEREÇOS IP'S PRIVADOS			
Classes	Número de End. por Rede	Intervalos de endereçamentos	Total de *Hosts*
Classe A	Até 256	0.0.0.0 até 127.0.0.0	Até 16.777.216
Classe B	Até 65.536	128.0.0.0 até 191.255.0.0	Até 65.536
Classe C	Até 16.777.216	192.0.0.0 até 223.255.255.0	Até 256

Tabela 1: Tabela de resumo das Classes de Endereço IP
Fonte: MSDN - Microsoft

Os endereços IP da classe A são usados onde é necessária uma rede apenas, mas uma grande quantidade de máquinas ligadas a ela. Os endereços IP

da classe B são usados nos casos onde a quantidade de redes é equivalente à quantidade de computadores. Os endereços IP da classe C são usados em locais com grande quantidade de redes, mas com poucas máquinas em cada uma (ALECRIM, 2014).

O mecanismo de tradução de endereços, NAT - *Network Address Translation*, - foi criado para responder à escassez de endereços IP com o protocolo IPv4, pois o número de endereços IP roteáveis não é suficiente para atender a todas as máquinas que conectadas à Internet. O princípio do NAT consiste em utilizar uma ponte estreita de conexão à Internet, com uma interface de rede ligada a rede interna e outra interface de rede ligada à Internet para possibilitar a conexão do conjunto de máquinas da rede. É responsável por realizar uma tradução dos pacotes que provêm da rede interna para a rede externa. Assim, cada máquina da rede que necessita de acesso à Internet é configurada para utilizar o NAT. Quando uma máquina da rede efetua um pedido à internet, o NAT realiza o pedido em seu lugar, recebe a resposta e transmite à máquina que fez o pedido (PILLOU, 2014).

2.4 PROTOCOLO IPV6

Segundo o Centro de Estudos e Pesquisas em Tecnologia de Redes e Operações (CEPTRO.BR) o IPv6, a sexta versão do protocolo IP, é a nova versão a ser implantada e mais segura, pois resolve os problemas relacionados a segurança identificados no IPv4, tornando a comunicação mais segura. Começou a ser desenvolvido no início da década de 1990, com o objetivo de ser a solução definitiva tendo como o principal objetivo o esgotamento de endereços IPs na Internet. O IPv6 possui endereçamento de 128 bits, sendo possível obter

340.282.366.920.938.463.463.374.607.431.768.211.456 endereços, que representa aproximadamente 79 octilhões (7,9×10E+28) de vezes a quantidade de endereços IPv4.

Segundo Teleco(2014), o protocolo IPv6 não foi criado somente para resolver o problema de esgotamento de endereços, mas também para disponibilizar novos serviços e benefícios que não existiam no IPv4 ou que não eram utilizados de forma otimizada. Podemos citar alguns desses benefícios:

- Espaço de endereçamento (128 *bits*);
- Formato de cabeçalho simplificado;
- Arquitetura hierárquica de rede para um roteamento eficiente;
- Suporte aos atuais protocolos de roteamento;
- Serviços de autoconfiguração;
- Implementação de IPSec (*IP Security Protocol)* de forma nativa;
- Crescimento do número de endereços *multicast*;
- Implantações para qualidade de serviço;
- Suporte a serviços de tempo real.

O cabeçalho IPv6 é descrito na figura 3, e apresenta um formato otimizado com apena 8 campos, totalizando um tamanho fixo de 40 bytes, o que otimiza o desempenho da rede, pois os roteadores não necessitam mais analisar o campo no IPv4 denominado: “tamanho de cabeçalho” que cujo o nome diz, indicava o tamanho do cabeçalho, antes de analisar as demais informações do pacote (BRITO, 2013, p.42).

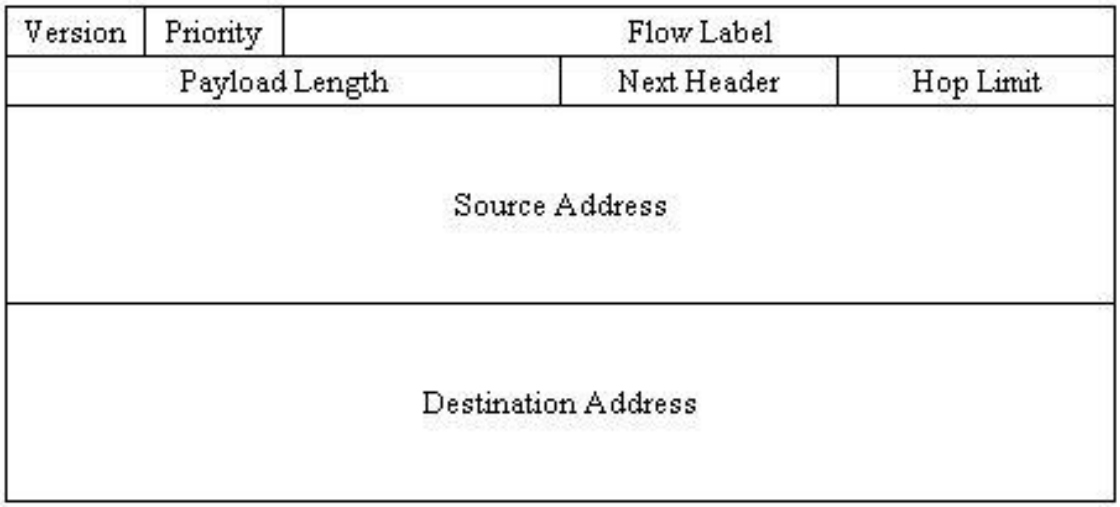


Figura 3 - Cabeçalho do Protocolo IPv6.
Fonte: http://www.teleco.com.br/tutoriais/tutorialredeip1/pagina_4.asp.

Os campos que constituem o cabeçalho IPv6 são:

- ***Version* (4 bits) -** Versão do IP utilizada. No caso no IPv6, este campo vale 0110.
- ***Priority* (4 bits) -** Indica a prioridade com a qual o pacote deve ser tratado.
- ***Flow Label* (24 bits) -** Identifica, juntamente com os campos Source Address e Destination Address, o fluxo ao qual o pacote pertence.
- ***Payload Length* (16 bits)** - Tamanho, em octetos, do restante do pacote, após o cabeçalho.
- ***Next Header* (8 bits) -** Indica o tipo do possível cabeçalho de extensão que segue o cabaçalho IPv6. Caso não esteja se utilizando cabeçalho de extensão, este campo indica a qual protocolo de transporte o pacote deve ser repassado.
- ***Hop Limit* (8 bits) -** Número máximo de roteamentos que o pacote pode sofrer. O valor deste campo é decrementado a cada roteamento. Quando seu

valor chega a zero o pacote é descartado. Similar ao campo Time to live do IPv4.

- ***Source Address* (128 bits) -** Endereço de origem.
- ***Destination Address* (128 bits) -** Endereço de destino.

2.4.1 Tipos de Endereços

Segundo o Centro de Estudos e Pesquisas em Tecnologia de Redes e Operações (CEPTRO.BR) (2014) no IPv6 existem três tipos de endereços definidos, são eles:

- ***Unicast*** – tipo de endereço que identifica uma única interface, de modo que um pacote enviado a um endereço *unicast* é entregue a uma única interface;
- ***Anycast*** – tipo de endereço que identifica um conjunto de interfaces. Um pacote encaminhado a um endereço *anycast* o mesmo é entregue a interface pertencente a este conjunto mais próxima da origem. É utilizado em comunicações de um-para-um-de-muitos.
- ***Multicast*** – tipo de endereço que um conjunto de interfaces, entretanto, um pacote enviado a um endereço *multicast* é entregue a todas as interfaces associadas a esse endereço. É utilizado em comunicações de um-para-muitos.

No IPv6 não existe endereço *broadcast*, responsável por direcionar um pacote para todos os nós de um mesmo domínio. No IPv6, essa função foi atribuída à um tipo específico de endereço *multicast*, denominado: *multicast-all-nodes*, no qual todos os nós fazem parte quando a interface é ativada e é identificado pelo endereço ff02::1 (BRITO, 2013, p. 57).

2.4.2 Endereços Especiais

Alguns endereços IPv6 são utilizados para fins específicos, são eles:

- **Endereço Não-Especificado (*Unspecified*):** representado pelo endereço **0:0:0:0:0:0:0:0** ou **::0** , equivalente ao endereço IPv4 *unspecified* **0.0.0.0**. Não deve ser atribuído a nenhum nó, serve para indicar apenas a ausência de um endereço. Pode ser utilizado, por exemplo, no campo Endereço de Origem de um pacote IPv6 enviado por um *host* durante o processo de inicialização, antes que este tenha seu endereço exclusivo determinado, mas não deve ser utilizado como endereço de destino de pacotes IPv6;
- **Endereço *Loopback*:** representado pelo endereço *unicast* **0:0:0:0:0:0:0:1** ou **::1**, equivalente ao endereço IPv4 *loopback* **127.0.0.1**. É utilizado para referenciar a própria máquina, sendo muito utilizado para testes internos. Este tipo de endereço não deve ser atribuído a nenhuma interface física, nem usado como endereço de origem em pacotes IPv6 enviados para outros nós.

Algumas faixas de endereços são reservadas para usos específicos:

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

2.4.2 Comparativo entre IPv4 e IPv6

O IPv6 se diferencia em endereços quase ilimitados, aumento da mobilidade, melhor desempenho e características de segurança superiores. Abaixo podemos analisar um o comparativo entre os protocolos, na tabela 2, (TELECO, 2014).

IPV4	IPV6
Endereço de 32bits	Endereço de 128bits
Suporte opcional de IPSec	Suporte nativo de IPSec
Nenhuma referência a capacidade de QoS (*Quality of Service)*	Introduz capacidades de QoS utilizando para isso o campo *Flow Label*
Processo de fragmentação realizada pelo *router*	A fragmentação deixa de ser realizada pelos *routers* e passa a ser processada pelos *hosts* emissores
O cabeçalho inclui os campos de opção	Todos os campos de opção foram mudados para dentro do campo*extension header*
O *Adress Resolution Protocol* (ARP), utiliza requisitos do tipo *Broadcast*	O ARP foi abandonado, sendo substituídos pelas mensagens *Neibhbor Discovery*
Internet Resolution Management Protocol (IGMP) é utilizado para gerir relações locais de sub-redes	O IGMP fio substituído por mensagens *Multcast Listner Discovery*
Os Endereços de *Broadcast* são utilizados para enviar tráfego para todos os *hosts* de uma rede	Deixa de existir o endereço de *Broadcast,* para utilizar endereços *multicast*
O endereço tem de ser configurado manualmente	Adição de funcionalidades de autoconfiguração
Suporta pacotes de 576 bytes, passiveis de serem fragmentados	Suporta pacotes de 1280 bytes, sem fragmentação

Tabela 2: Comparativo entre IPv4 e IPv6.
Fonte: http://www.teleco.com.br/tutoriais/tutorialredeip1/pagina_4.asp

O IPv6 introduz um novo formato de cabeçalho, em oposição ao anterior IPv4, todos os campos deste possuem tamanho fixo, totalizando 64 bytes. O fato de possuir um tamanho fixo torna mais ágil o processamento dos pacotes pelos roteadores, visto que não há necessidade de calcular a extensão de certos campos, e nem o tamanho do cabeçalho como um todo. Além disso, também houve uma redução dos números de campos utilizados, por meio da exclusão de alguns campos que contribui para a diminuição do tempo gasto em processamento pelos roteadores, conforme descrito na figura 4 (TELECO, 2014).

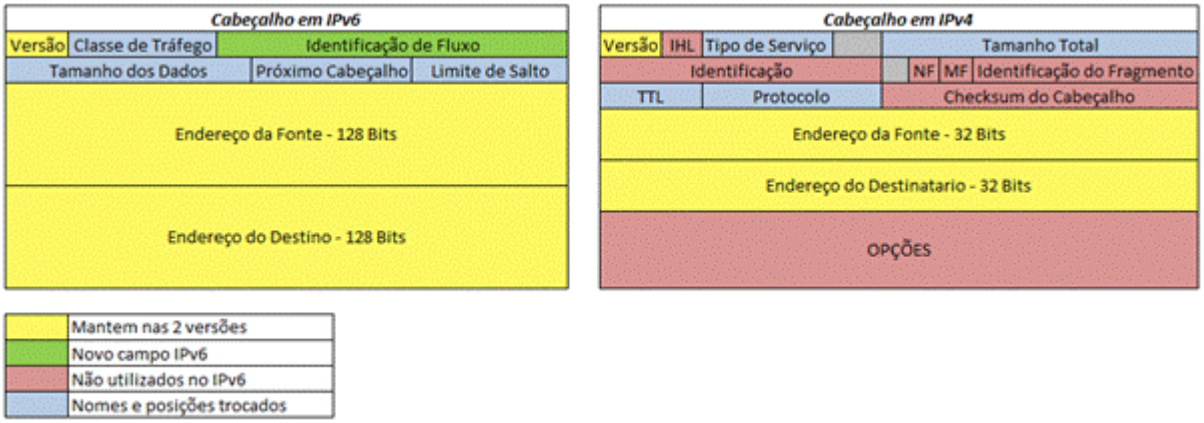


Figura 4 - Cabeçalho dos Protocolos IPv4 e IPv6 e suas alterações.
Fonte: http://www.teleco.com.br/tutoriais/tutorialredeip1/pagina_4.asp.

Dentre os campos que foram eliminados, merecem destaque o de *Checksum* e o de fragmentação. A função do campo de *Checksum* era detectar erros que afetassem ao cabeçalho IP, não detectando no entanto erros no restante do pacote. A maioria dos erros não é de transmissão, visto que os mecanismos de detecção de erros *Ethernet* e PPP são eficientes, mas sim nos roteadores. Como os roteadores alteram somente o campo *Hop Limit* (*Time-to-live* no IPv4), estes então terminam por recalcular o *Checksum* antes de retransmitir o pacote, o que pode causar a não detecção de possíveis erros (TELECO, 2014).

Quanto ao campo fragmentação foi excluído, pois se decidiu que pacotes não serão mais fragmentados por roteadores. Caso um roteador receba um pacote com tamanho maior que o permitido, será descartado e enviará uma mensagem ao *host* que o enviou, comunicando o ocorrido. Este *host* então deverá retransmitir o pacote transformando em pacotes menores. Desta forma há um ganho de performance no roteamento, pois é eliminada a necessidade de um roteador fragmentar vários pacotes.

Outra alteração realizada com o intuito de agilizar o processamento foi a renomeação e reposicionamento de quatro campos conforme a tabela 3:

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

Tabela 3: Comparativo dos campos do cabeçalho IPv4 renomeados no cabeçalho IPv6. Fonte: http://ipv6.br/entenda/cabecalho/.

2.6 DNS

Segundo Couto(2014), quando um computador se comunica com outro na internet, para acessar uma página web, enviar um e-mail ou mandar uma mensagem instantânea, ele precisa saber o endereço do computador com o qual irá se comunicar. Esse endereço, chamado de endereço IP, é uma sequência de 4 números decimais separados por ponto da seguinte forma: 201.154.37.98 na versão 4. Quando digitamos um endereço em um *Browser*, ele irá fazer uso do DNS para transformar o nome do *host* em um endereço IP. Essa não é a única função desse protocolo, mas é um exemplo da importância desse serviço.

Na década de 70, a *Advanced Research Projects Agency Network* (ARPANet) era uma rede pequena que possuia todos os nomes das máquinas em um único arquivo chamado hosts.txt, como consta na RFC 952. Nesse ambiente, se o endereço IP quando havia mudança em um servidor, não havia a garantia que o arquivo hosts.txt dos outros servidores também seriam alterados. Com o aumento da rede cada vez mais ficou inviável o uso do arquivo estático em cada ponto da rede. Foi na década de 80 que ocorreu o desenvolvimento do protocolo e sua primeira implementação do DNS, nas RFCs 882 e 883, que foram substituídas pelas RFCs 1034 e 1035. A primeira implementação de um servidor DNS para

Unix aconteceu em 1984, com o *Berkeley Internet Name Domain* (BIND), que é amplamente utilizado até os dias de hoje (COUTO, 2014).

Atualmente o controle de registros de domínio é realizado pelo IANA(Internet Assigned Numbers Authority) que é um órgão vinculado ao Governo dos Estados Unidos. No Brasil, o órgão responsável pelo controle dos domínios é a FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo) (ANGÉLICA, 2014).

2.6.1 Implementação do DNS

Segundo Angélica(2014) o DNS é implementado por meio de uma aplicação cliente-servidor. O cliente é o resolver, que contém um conjunto de rotinas em uma implementação de TCP/IP, que permite a consulta a um servidor, e um servidor geralmente é o programa bind ou uma implementação específica de um servidor de DNS, no caso da utilização do Windows NT.

O servidor de DNS pode ser responsável pela resolução de vários nomes de domínios. Seu escopo de atuação define a zona de atuação de um servidor DNS e para resolver um domínio e seus sub-domínios existem três zonas: a primeira resolve o próprio domínio principal e os subdomínios; a segunda resolve os domínios; e a terceira resolve o domínio. Cada zona possui um servidor de nomes principal ou primário, que mantêm em tabelas o mapeamento dos nomes em endereços IP daquele domínio (ANGÉLICA, 2014).

A resolução de um nome é realizada de forma recursiva, consultando diversos servidores até chegar ao responsável pelo domínio consultado.

2.6.2 Aplicações para DNS

Segundo Angélica(2014), as utilidades mais comuns para consulta de servidores de nome que são providos com a implementação do DNS:

Aplicação	Função
nslookup	Permite localizar informações sobre ligacões na rede, examina os conteúdos de um servidor de nome de base de dados e estabiliza a acessibilidade dos servidores de nomes
dig	Permite exercer servidores de nomes, reuni grandes volumes de informações de domínios de nome e executa uma consulta de domínio de nome simples. DIG quer dizer Domain Internet Groper

Tabela 4: DNS
Fonte: **http://www.m8.com.br/antonio/redes/dns.htm**

2.6.3 Principais tipos de registros

Segundo Angélica(2014) o servidor DNS Autoritativo guarda os dados referentes aos domínios separados em diversos tipos de registros, cada um uso específico. Os principais tipos de registros utilizados na atualidade são:

Tipo de Registro	**Detalhamento**
SOA	É o registro que guarda o cabeçalho daquela Zona, disponibilizando dados importantes, seu formato é:
	· Name: O nome da zona (ex. ibm.com)
	· TTL: *Time-To-Live*
	· Class: Classe do registro, IN = Internet. Existem outras classes históricas HS (Hesiod) e CH (Chaos), que hoje em dia não são mais usadas
	· Name server: Indica o servidor DNS autoritativo daquela zona
	· Email: Indica o email do responsável por aquela zona. O símbolo de '@' é trocado por ponto '.'
	· SN (Serial Number): Número serial, usado por outros servidores DNS (secundários ou recursivos) para saberem que aquela zona foi alterada
	· Refresh: Indica de quanto em quanto tempo um servidor DNS *Slave* deverá atualizar os dados referentes àquela zona

	· Retry: Indica ao servidor DNS *Slave* de quanto em quanto tempo ele deverá tentar obter os dados daquela zona quando o Refresh falhar
	· Expiry: Indica quando o servidor DNS *Slave* não será mais considerado autoritativo daquela zona quando não for possível fazer o Refresh
	· Minimum / TTL: Define quanto tempo o registro dessa zona deverá permanecer no cache de um servidor DNS antes que seja feito um Refresh
A	Associa um *hostname* à um endereço IPv4.
AAAA	Associa um *hostname* à um endereço IPv6.
PTR	Associa um endereço IP a um *hostname* para a resolução de DNS reverso.
NS	Informa os IPs dos servidores DNS autoritativos de um domínio.
MX	Informa os IPs dos servidores SMTP de um domínio. Esse tipo de registro tem como particularidade um campo a mais, que informa a prioridade do servidor SMTP. Quanto mais baixo o valor, maior a prioridade.
CNAME	Usado para criarmos um alias de um *host* para outro, facilitando no caso de uma mudança de IP.
TXT	Pode armazenar qualquer informação em formato texto. Inicialmente criado para armazenar comentários ou informações sobre o domínio, hoje é muito usado por ferramentas anti-spam como o SPF (Sender Policy Framework) e o DomainKeys/DKIM.

Tabela 5: Tipos de Registros

Fonte: http://www.ibm.com/developerworks/br/local/opensource/dns_protocol/

2.6.4 Servidor autoritativo

O servidor autoritativo possui autoridade sobre qualquer domínio. Ele é capaz de responder as requisições DNS sobre o domínio com autoridade, informando que o servidor possui os arquivos de zonas com os registros de recursos solicitado do domínio em questão.

2.6.5 Servidor Recursivo

O Servidor Recursivo realiza um consulta recursiva, isto é, solicita informações sobre determinada requisição DNS em outros servidores, conhecidos como servidores autoritativos, até obter uma reposta satisfatória.

Ao receber requisições de resoluções de nomes, o servidor recursivo faz requisições para os servidores autoritativos e conforme a resposta recebida dos mesmos continua a realizar requisições para outros servidores autoritativos, até obter a resposta satisfatória. O servidor recursivo é obrigado a retornar uma resposta para o cliente DNS, seja positiva ou negativa.

2.6.6 Principais Softwares

Alguns dos principais softwares (*opensource* ou comerciais) utilizados para instalar e configurar um Servidor DNS:

- BIND
- NSD
- djbdns
- Unbound
- Microsoft DNS
- Dnsmasq
- MaraDNS

2.7 OSPFv3

O OSPFv3 é um protocolo de roteamento para IPv4 e IPv6. É um protocolo de *link-state*, em oposição a um protocolo de vetor de distância. Um protocolo link-state toma as suas decisões de roteamento com base nos estados dos links que conectam máquinas de origem e de destino. O estado de um link é uma descrição do que a interface e sua relação com os seus dispositivos de rede vizinhos. A informação de interface inclui o prefixo IPv6 da interface, a máscara de rede, do tipo

de rede que está ligado a, os dispositivos ligados a essa rede, e assim por diante. Esta informação é propagada em vários tipos de anúncios de link-state (LSAs) (CISCO, 2014).

2.8 MECANISMOS DE TRANSIÇÃO

Segundo CEPTRO (2014) o IPv4 e o IPv6 não são compatíveis entre si. O IPv6 não foi projetado para ser uma extensão, ou complemento, do IPv4, mas sim, um substituto para resolver o problema do esgotamento de endereços. Ambos protocolos podem funcionar simultaneamente nos mesmos equipamentos e com base nisto pensou-se em realizar a transição de forma gradual.

Inicialmente no projeto do IPv6, sua implantação começaria a ser realizada gradualmente na Internet, de forma que funcionasse simultaneamente ao IPv4. Esse mecanismo de transição chamamos de pilha dupla, ou *dual stack*. Quando o IPv6 estivesse implantado em todos os dispositivos, o IPv4 deixaria de ser realmente útil e poderia ser abandonado lentamente.

Durante o período de implantação do IPv6 seriam necessárias técnicas auxiliares de transição, para interconectar ilhas IPv6 em uma Internet com predominância de IPv4 e, depois de algum tempo, para fazer o contrário.

Dessa maneira a transição seria muito simples de ser executada tecnicamente. Contudo, por várias razões, não foi o que aconteceu. Atualmente o IPv6 ainda não está sendo amplamente utilizado na Internet e o esgotamento do IPv4 já se tornou uma realidade. Há necessidade de se implantar o IPv6 numa Internet sempre crescente, onde os novos usuários ainda precisam de conectividade IPv4, mas não há mais endereços IPv4 livres para atendê-los. Assim, novas técnicas auxiliares foram e continuam sendo, desenvolvidas para essa nova realidade.

O período de transição e de coexistência dos dois protocolos exigiu o desenvolvimento de técnicas auxiliares. Um dos problemas era como conectar redes IPv6 a outras redes IPv6 por meio de equipamentos ou de uma Internet que suportassem somente o IPv4. Surgiram então vários tipos de túneis IPv6 sobre IPv4 para atender a essa necessidade, usando diferentes técnicas, estabelecidos manualmente ou automaticamente. Outras técnicas foram criadas para permitir que redes IPv6 e IPv4 interoperassem, por meio da tradução dos pacotes.

Mais recentemente, o problema principal a ser resolvido pela técnicas de transição passou a ser a implantação do IPv6 num ambiente em que o IPv4 não está mais disponível, mas ainda é necessário para os novos usuários da Internet. Foram, e continuam sendo, desenvolvidos então diversos tipos de túneis IPv4 sobre IPv6 para, aliados a técnicas de tradução, solucionar esse problema.

Pode-se, então, classificar as técnicas de transição segundo sua funcionalidade, em:

- **Pilha dupla:** consiste na convivência do IPv4 e do IPv6 nos mesmos equipamentos, de forma nativa, simultâneamente. Essa técnica é a técnica padrão escolhida para a transição para IPv6 na Internet e deve ser usada sempre que possível.
- **Túneis:** Permitem que diferentes redes IPv4 comuniquem-se através de uma rede IPv6, ou vice-versa.
- **Tradução:** Permitem que equipamentos usando IPv6 comuniquem-se com outros que usam IPv4, por meio da conversão dos pacotes.

2.8 PILHA-DUPLA

Segundo CEPTRO (2014), a técnica conhecida como pilha dupla (Dual Stack ou DS) mantem o IPv4 funcionando de forma estável e permite a implantação

do IPv6 nativamente, é a forma básica escolhida para a transição na Internet. A utilização deste método permite que dispositivos e roteadores estejam equipados com pilhas para ambos os protocolos, tendo a capacidade de enviar e receber os dois tipos de pacotes, IPv4 e IPv6. Com isso, um nó Pilha Dupla, ou nó IPv6/IPv4, se comportará como um nó IPv6 na comunicação com outro nó IPv6 e se comportará como um nó IPv4 na comunicação com outro nó IPv4. Cada nó IPv6/IPv4 é configurado com ambos endereços, utilizando mecanismos IPv4 para adquirir seu endereço IPv4 e mecanismos IPv6 para adquirir seu endereço IPv6. Este método de transição permita uma implantação gradual, com a configuração de pequenas seções do ambiente de rede de cada vez. Quando o IPv4 não for mais usado, basta simplesmente desabilitar a pilha IPv4 em cada nó. O funcionamento da pilha dupla está ilustrado na figura 12.

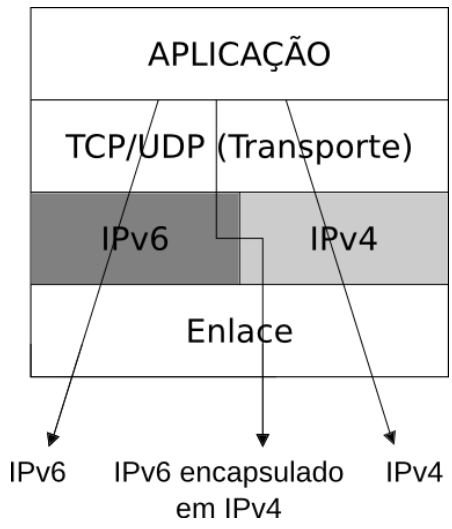


Figura 5 - Funcionamento da Pilha Dupla.
Fonte: http://ipv6.br/entenda/transicao/.

Alguns aspectos referentes à infra-estrutura da rede devem ser configurados para implementar a técnica de pilha dupla: a estruturação do serviço de DNS e a configuração dos protocolos de roteamento e de firewalls.

Pode não ser possível utilizar pilha dupla em todas as ocasiões. Por exemplo, em situações quando não há mais IPv4 disponíveis e o provedor necessita atender a usuários novos com IPv6 e IPv4. Para redes corporativas que já utilizam o IPv6 nativo pode ser utilizado em conjunto com o IPv4 compartilhado. Outra situação que dificulta a implantação do IPv6 usando pilha dupla quando há equipamentos que não o suportam e que não podem ser facilmente substituídos. Para contornar essas situações existem diversas técnicas disponíveis.

3 IMPLEMENTAÇÃO

A proposta desse trabalho é implementar um ambiente de simulação de uma rede local, utilizando o *software* GNS3 versão 0.8.7, com a configuração de Pilha Dupla, método de transição escolhido, e a configuração de um serviço DNS nas versões IPv4 e IPv6, procurando analisar seu funcionamento e importância. Para analisar os serviços implementados serão feitos testes entre cliente e servidor, procurando demonstrar o funcionamento e identificar possíveis problemas. Com essa análise será observado o comportamento dos protocolos de internet versão 4 (IPv4) e versão 6 (IPv6) funcionando simultaneamente.

Para montar o ambiente foram utilizados alguns equipamentos essenciais para a simulação da topologia de rede escolhida, figura 6. Foram necessários 5 roteadores, 2 servidores (DNS e *Web*), 4 switches e 4 *hosts,* que simulam os clientes da rede.

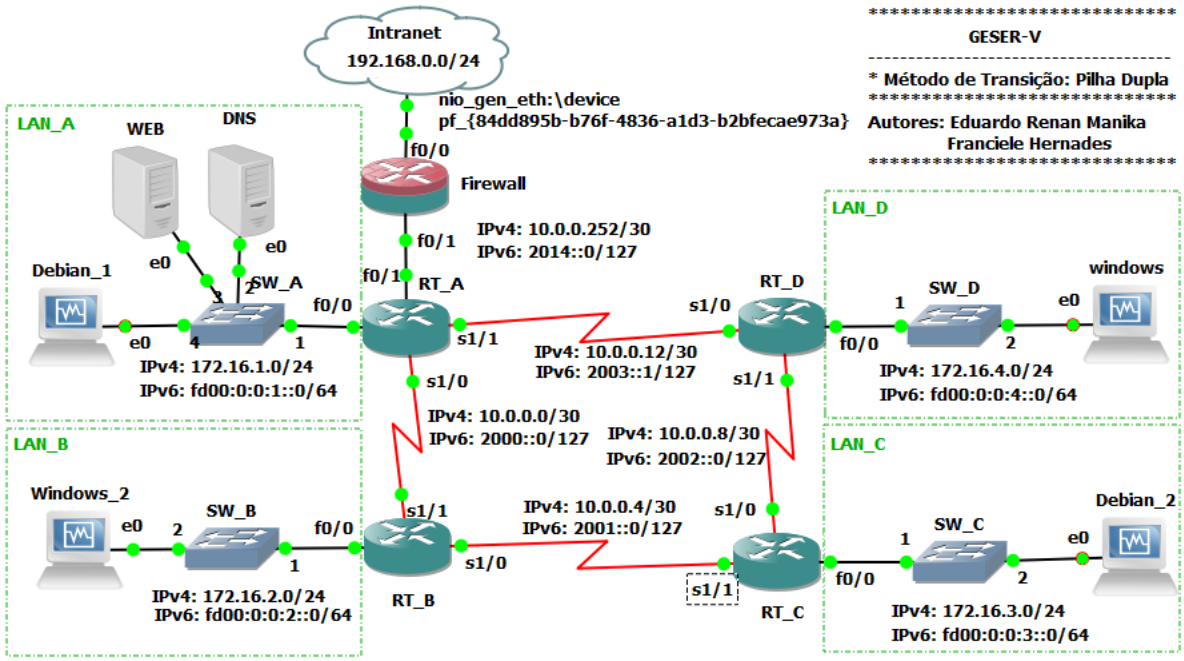


Figura 6 - Topologia da Rede
Fonte: Próprio Autor

O modelo do roteador escolhido foi o Cisco 7206VXR NPE-400 com 256 MB RAM, com suporte aos protocolos IPv4 e IPv6. Para os *hosts* e servidores utilizou-se o software que permite virtualizar computadores de diferentes sistemas operacionais, *Linux* e *Windows, o VirtualBox* . Na tabela 6 constam as versões dos sistemas operacionais utilizados.

Sistema Operacional	**Equipamento**
Debian 7.0 – Server	Servidores(DNS e WEB)
Debian 7.7 – Desktop	*Hosts* da rede
Windows XP	*Hosts* da rede

Tabela 6: Sistemas operacionais utilizados

Fonte: Próprio autor

Os equipamentos utilizados foram configurados com o endereçamento IPv4 e IPv6, conforme tabela 7.

Equipamento	**Protocolo**	**Interface**	**Endereço**	**Máscara**	**Gateway**
Firewall	IPv4	f0/0	DHCP - Intranet	24	N/A
		f0/1	10.0.0.252	30	N/A
	IPv6	f0/0	-	-	-
		f0/1	2014::0	127	N/A
RT_A	IPv4	f0/0	172.16.1.1	24	N/A
		f0/1	10.0.0.253	30	N/A
		s1/0	10.0.0.1	30	N/A
		s1/1	10.0.0.14	30	N/A
	IPv6	f0/0	FD00:0:0:1::0	64	N/A
		f0/1	2014::1	127	N/A
		s1/0	2000::0	127	N/A
		s1/1	2003::1	127	N/A
RT_B	IPv4	f0/0	172.16.2.1	24	N/A
		s1/0	10.0.0.5	30	N/A
		s1/1	10.0.0.2	30	N/A
	IPv6	f0/0	FD00:0:0:2::0	64	N/A
		s1/0	2001::0	127	N/A
		s1/1	2000::1	127	N/A
RT_C	IPv4	f0/0	172.16.3.1	24	N/A
		s1/0	10.0.0.9	30	N/A
		s1/1	10.0.0.6	30	N/A
	IPv6	f0/0	FD00:0:0:3::0	64	N/A
		s1/0	2002::0	127	N/A
		s1/1	2001::1	127	N/A

RT_D	IPv4	f0/0	172.16.4.1	24	N/A
		s1/0	10.0.0.13	30	N/A
		s1/1	10.0.0.10	30	N/A
	IPv6	f0/0	FD00:0:0:4::0	64	N/A
		s1/0	2003::0	127	N/A
		s1/1	2002::1	127	N/A
Servidor DNS	IPv4	f0/0	172.16.1.2	24	172.16.1.1
	IPv6	f0/0	FD00:0:0:1::1	64	FD00:0:0:1::0
Servidor WEB	IPv4	f0/0	172.16.1.3	24	172.16.1.1
	IPv6	f0/0	FD00:0:0:1::2	64	FD00:0:0:1::0

Tabela 6: Configurações

Fonte: Próprio autor

3.1 CONFIGURAÇÕES DOS PROTOCOLOS DE INTERNET

3.1.1 IPv4

Para realizar a configuração do protocolo IPv4, baseou-se na topologia mostrada na figura 7.

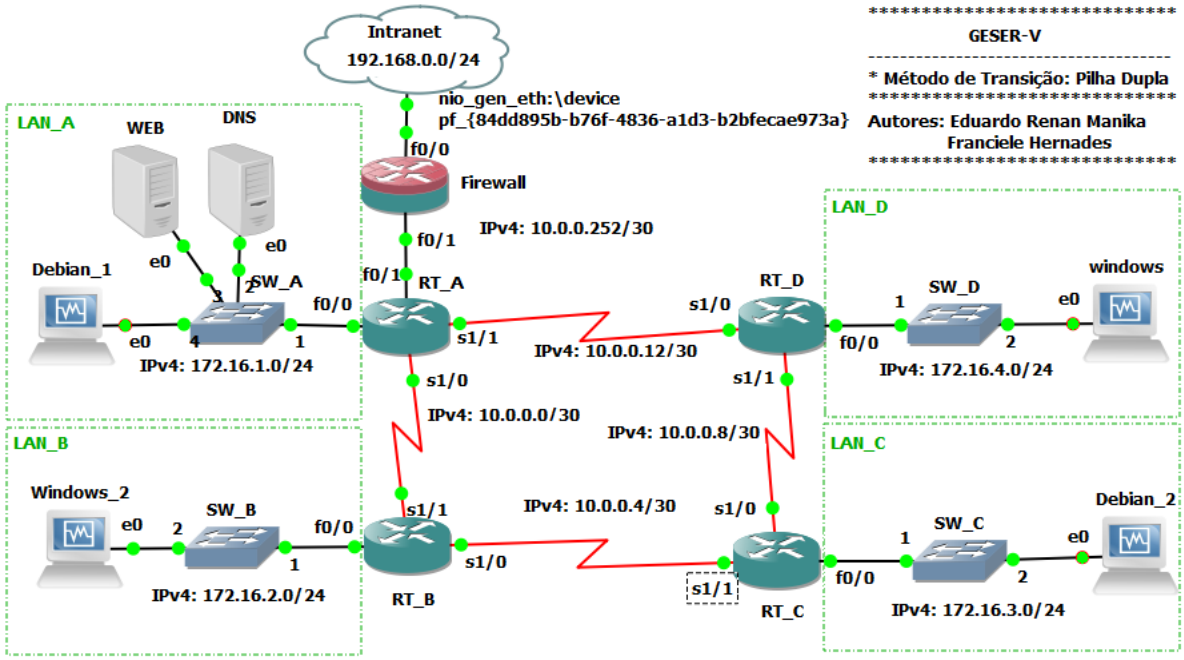


Figura 7 - Topologia da Rede IPv4.
Fonte: Próprio autor

Para configurar a topologia conforme especificação são necessários os seguintes passos:

- Configuração das interfaces dos roteadores, com endereço IPv4;
- Configuração do protocolo de roteamento, OSPF, versão IPv4;
- Configuração dos hosts, utilizando DHCP;
- Configuração dos servidores, DNS e Web.

Para configurar as interfaces dos roteadores foram utilizados os comandos, conforme figura 8 e as configurações utilizadas para cada equipamento da rede foram realizadas de acordo com a tabela 8.

```
-----------------------------------------------------------------------
                Configuração das Interfaces – IPv4
-----------------------------------------------------------------------
RT_A>enable                        //Entra no modo exec privilegiado
RT_A#configure terminal            //Entra no modo de configuração
-----------------------------------------------------------------------
Configurar a Interface f0/0:
-----------------------------------------------------------------------
RT_A(config)#interface fastEthernet 0/0      //Entra na interface
RT_A(config-if)#ip address 172.16.1.1 255.255.255.0 //Seta o endereço
RT_A(config-if)#no shutdown     //Habilita a interface
RT_A(config-if)#exit           //Volta para o modo de conf. global
-----------------------------------------------------------------------
Configurar a Interface f0/1:
-----------------------------------------------------------------------
RT_A(config)#interface fastEthernet 0/1      //Entra na interface
RT_A(config-if)#ip address 10.0.0.253 255.255.255.252 //Seta o end.
RT_A(config-if)#no shutdown  //Habilita a interface
RT_A(config-if)#exit         // volta para o modo de conf. Global
-----------------------------------------------------------------------
Configurar a Interface s1/0:
-----------------------------------------------------------------------
RT_A(config)#interface serial 1/0             //Entra na interface
RT_A(config-if)#ip address 10.0.0.1 255.255.255.252 //Seta o end.
RT_A(config-if)#clock rate  64000 //(valor em bps) Configura a taxa
                                  clock, usado somente para as
                                  interfaces DCE
RT_A(config-if)#no shutdown  //Habilita a interface
RT_A(config-if)#exit         // volta para o modo de conf. Global
-----------------------------------------------------------------------
Configurar a Interface s1/1:
-----------------------------------------------------------------------
RT_A(config)#interface serial 1/1 //Entra na interface
RT_A(config-if)#ip address 10.0.0.14 255.255.255.252 //Seta o end.
RT_A(config-if)#no shutdown  //Habilita a interface
RT_A(config-if)#exit         // volta para o modo de conf. Global
-----------------------------------------------------------------------
```

Figura 8 - Comandos configuração dos roteadores.
Fonte: Próprio autor

Dispositivo	Interface	Endereço IPv4	Máscara IPv4	Gateway IPv4
RT_A	f0/0	172.16.1.1	24	N/A
	f0/1	10.0.0.253	30	N/A
	s1/0	10.0.0.1	30	N/A
	s1/1	10.0.0.14	30	N/A
RT_B	f0/0	172.16.2.1	24	N/A
	s1/0	10.0.0.5	30	N/A
	s1/1	10.0.0.2	30	N/A
RT_C	f0/0	172.16.3.1	24	N/A
	s1/0	10.0.0.9	30	N/A
	s1/1	10.0.0.6	30	N/A
RT_D	f0/0	172.16.4.1	24	N/A
	s1/0	10.0.0.13	30	N/A
	s1/1	10.0.0.10	30	N/A
Firewall	f0/0	DHCP - Intranet	24	N/A
	f0/1	10.0.0.252	30	N/A
Servidor DNS	f0/0	172.16.1.2	24	172.16.1.1
Servidor WEB	f0/0	172.16.1.3	24	172.16.1.1

Tabela 7: Configurações

Fonte: Próprio autor

Para configurar os endereços das interfaces é necessário entrar na interface através do comando: # *interface <Nome _da_Interface>*, aparecendo a sinalização que encontra-se no modo de configuração da interface, exemplo: *RT_A(config-if)*. Para adicionar o endereço de rede IPv4 utiliza-se o comando: # *ip address <endereço_IPv4> <máscara_de_rede>* e o comando: # *no shutdown* habilita a interface.

Nas interfaces Serial – DCE, que no caso for especificadas as interfaces *Serial 0/0*, é necessário seta o clock rate para setar a taxa de transmissão do enlace, através do comando: # *clock rate 64000*, que equivale a 64 Kbps.

Na figura 9 é apresentado o trecho do arquivo de configuração do roteador RT_A, após a configuração dos endereços em todas as interfaces, através do comando: #*show running-config*.

```
interface FastEthernet0/0
 ip address 172.16.1.1 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet0/1
 ip address 10.0.0.253 255.255.255.252
 duplex auto
 speed auto
!
interface Serial1/0
 ip address 10.0.0.1 255.255.255.252
 serial restart-delay 0
 clock rate 64000
!
interface Serial1/1
 ip address 10.0.0.14 255.255.255.252
 serial restart-delay 0
```

Figura 9 - Trecho do arquivo de configuração do roteador.
Fonte: Próprio autor

Os arquivos de configuração de todos os roteadores da rede encontram-se nos Apêndices A, B, C, D.

3.1.1.2 Configuração do Protocolo de Roteamento OSPF

A configuração do protocolo de roteamento OSPF é realizar após a configuração de todas as interfaces dos roteadores. Na figura 10 é apresentado um exemplo de configuração do protocolo, no caso o roteador RT_A. Para configurar o OSPF é necessário informar quais são as redes conectadas nas interfaces do roteador, é utilizada a máscara curinga, e informar à área que a interface esta conecta.

```
-----------------------------------------------------------------
                  Configuração do OSPF – IPv4
-----------------------------------------------------------------
RT_A>enable                      //Entra no modo exec privilegiado
RT_A#configure terminal          //Entra no modo de configuração
-----------------------------------------------------------------
Configurar o OSPF – id do processo:
-----------------------------------------------------------------
RT_A(config)#router ospf 1       //id do processo – (1 até 65535)
-----------------------------------------------------------------
Configurar o OSPF – redes diretamente conectadas:
#network <endereço_da_rede> <máscara_curinga> área <id_da_area(0 até
65535)>
-----------------------------------------------------------------
RT_A(config-router)#network 10.0.0.0 0.0.0.3 area 0
RT_A(config-router)#network 10.0.0.12 0.0.0.3 area 0
RT_A(config-router)#network 10.0.0.252 0.0.0.3 area 0
RT_A(config-router)#network 172.16.1.0 0.0.0.3 area 0
RT_A(config-router)#default-information originate //Adic. a rota
                                                  padrão ao proc. OSPF
RT_A(config-router)#exit        // Volta para o modo de conf. Global
-----------------------------------------------------------------
Configurar a Rota Padrão – que será disponibilizada aos demais
roteadores pelo processo OSPF:
-----------------------------------------------------------------
RT_A(config)#ip route 0.0.0.0 0.0.0.0 10.0.0.254 //Configura uma rota
                                                   Padrão
```

Figura 10 - Configuração do protocolo OSPF.
Fonte: Próprio autor

O roteador RT_A foi configurado com a rota default, que é a rota de saída para fora da rede, no caso apontando para o Firewall. Isso é realizado através do comando: *#ip route 0.0.0.0 0.0.0.0 10.0.0.254.* Essa rota default é disponibilizada para os demais roteadores da rede, através do protocolo OSPF, realizado através do comando: *# default-information originate.*

Após a configuração do OSPF em todos os roteadores e a rede convergida é possível realizar os testes. Na figura 11 é apresentada as rotas aprendidas pelo roteador RT_A, através do comando: # *show ip route*. As rotas descritas com “C” no inicio da descrição, são as rotas diretamente conectadas as interfaces do roteador; as rotas iniciadas com “O” são as rotas aprendidas pelo protocolo de roteamento OSPF; e a rota iniciada com “S*” representa a rota default que é informada via OSPF.

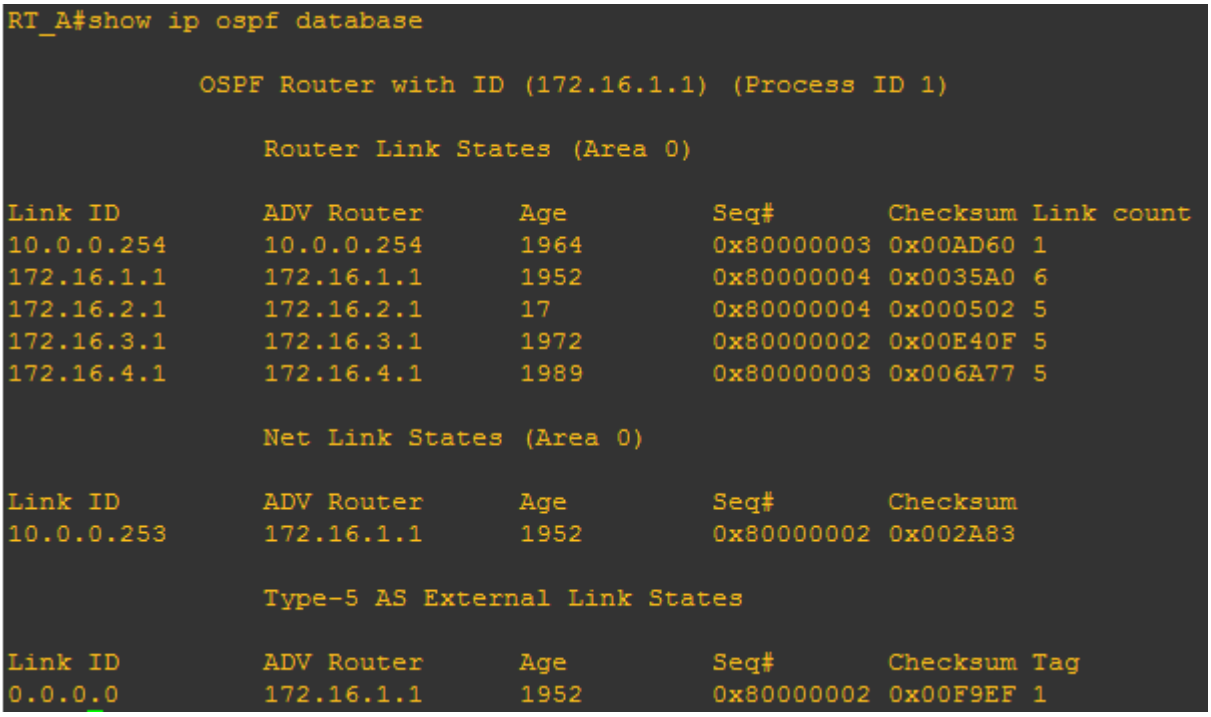
```
RT_A#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 10.0.0.254 to network 0.0.0.0

     172.16.0.0/24 is subnetted, 4 subnets
O       172.16.4.0 [110/65] via 10.0.0.13, 00:10:51, Serial1/1
C       172.16.1.0 is directly connected, FastEthernet0/0
O       172.16.2.0 [110/65] via 10.0.0.2, 00:10:51, Serial1/0
O       172.16.3.0 [110/129] via 10.0.0.13, 00:10:51, Serial1/1
                   [110/129] via 10.0.0.2, 00:10:51, Serial1/0
     10.0.0.0/30 is subnetted, 5 subnets
O       10.0.0.8 [110/128] via 10.0.0.13, 00:10:51, Serial1/1
C       10.0.0.12 is directly connected, Serial1/1
C       10.0.0.0 is directly connected, Serial1/0
O       10.0.0.4 [110/128] via 10.0.0.2, 00:10:51, Serial1/0
C       10.0.0.252 is directly connected, FastEthernet0/1
S*   0.0.0.0/0 [1/0] via 10.0.0.254
```

Figura 11 - Rotas.
Fonte: Próprio autor

Na figura 12 é apresentada base de dados do protocolo OSPF, que contem os roteadores que fazem parte de cada área especificada.



```
RT_A#show ip ospf database

            OSPF Router with ID (172.16.1.1) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.0.0.254      10.0.0.254      1964        0x80000003 0x00AD60 1
172.16.1.1      172.16.1.1      1952        0x80000004 0x0035A0 6
172.16.2.1      172.16.2.1      17          0x80000004 0x000502 5
172.16.3.1      172.16.3.1      1972        0x80000002 0x00E40F 5
172.16.4.1      172.16.4.1      1989        0x80000003 0x006A77 5

                Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
10.0.0.253      172.16.1.1      1952        0x80000002 0x002A83

                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
0.0.0.0         172.16.1.1      1952        0x80000002 0x00F9EF 1
```

Figura 12 - Base de dados do protocolo OSPF.
Fonte: Próprio autor

Na figura 13 apresenta o comando: *#show ip ospf interface*, que descreve os status e as estatísticas de cada interface que faz parte do protocolo OSPF, do referido roteador, no caso o roteador RT_A.

```
RT_A#show ip ospf interface
FastEthernet0/0 is up, line protocol is up
  Internet Address 172.16.1.1/24, Area 0
  Process ID 1, Router ID 172.16.1.1, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 172.16.1.1, Interface address 172.16.1.1
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:01
  Supports Link-local Signaling (LLS)
  Index 4/4, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 0
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
FastEthernet0/1 is up, line protocol is up
  Internet Address 10.0.0.253/30, Area 0
  Process ID 1, Router ID 172.16.1.1, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 172.16.1.1, Interface address 10.0.0.253
  Backup Designated router (ID) 10.0.0.254, Interface address 10.0.0.254
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:09
  Supports Link-local Signaling (LLS)
  Index 3/3, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 10.0.0.254  (Backup Designated Router)
  Suppress hello for 0 neighbor(s)
Serial1/1 is up, line protocol is up
  Internet Address 10.0.0.14/30, Area 0
  Process ID 1, Router ID 172.16.1.1, Network Type POINT_TO_POINT, Cost: 64
  Transmit Delay is 1 sec, State POINT_TO_POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:02
  Supports Link-local Signaling (LLS)
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 172.16.4.1
  Suppress hello for 0 neighbor(s)
Serial1/0 is up, line protocol is up
  Internet Address 10.0.0.1/30, Area 0
  Process ID 1, Router ID 172.16.1.1, Network Type POINT_TO_POINT, Cost: 64
  Transmit Delay is 1 sec, State POINT_TO_POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:02
  Supports Link-local Signaling (LLS)
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 172.16.2.1
  Suppress hello for 0 neighbor(s)
```

Figura 13 - Status e estatísticas de cada interface.
Fonte: Próprio autor.

Na figura 14 é apresentado o comando: # ping, que é uma ferramenta que é utilizada para realizar testes de conexão e utiliza o protocolo ICMP para realizar esses testes. No caso é realizado um teste de conexão entre o roteador RT_C e os demais roteadores, que não estão diretamente conectados, demonstrando assim o funcionamento do protocolo OSPF.

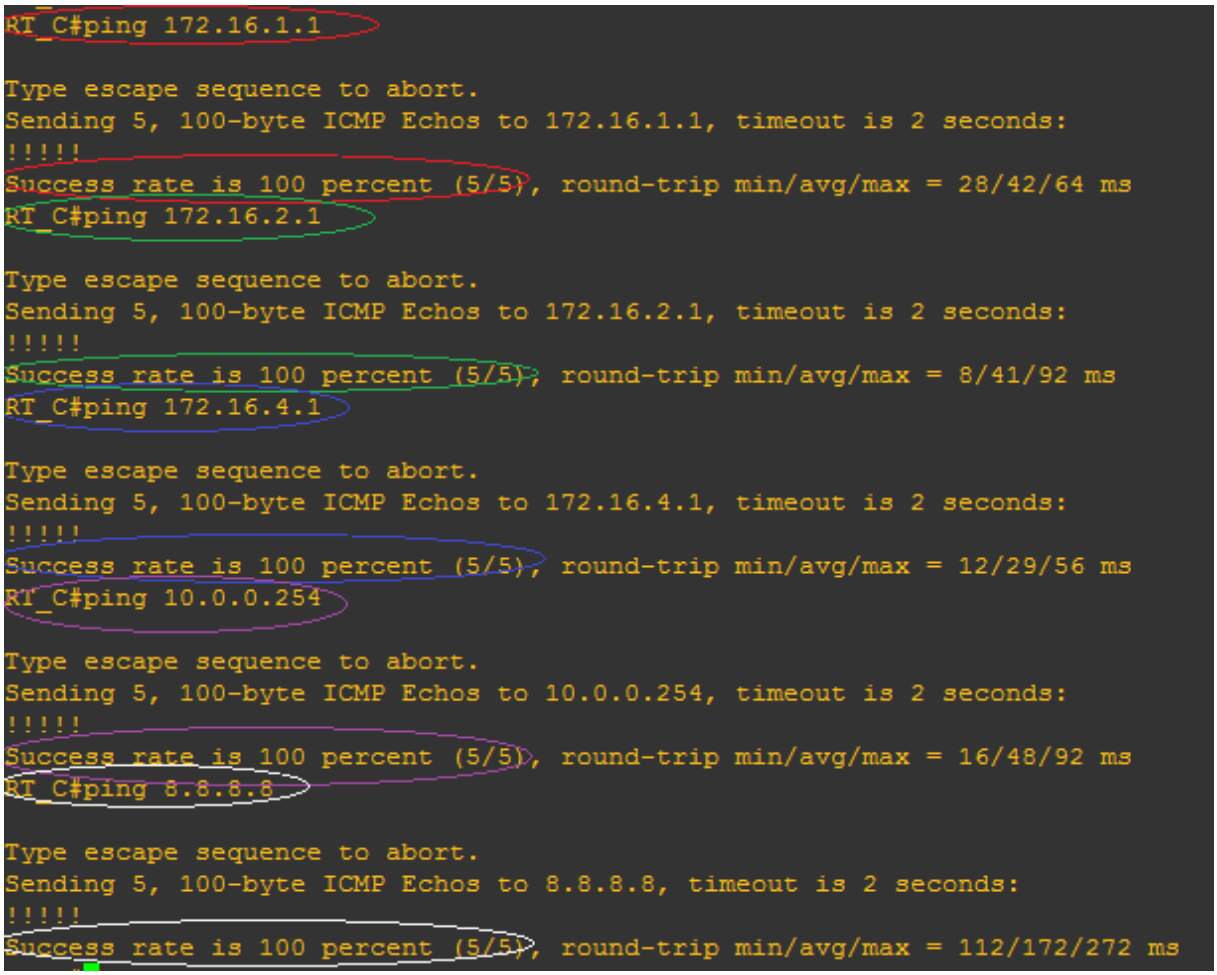


Figura 14 - Testes de Conexão.
Fonte: Próprio autor

Comando ping do roteador RT_C para as interfaces LAN dos roteadores RT_A, RT_B, RT_C e para Fora da rede (DNS – Google), respectivamente.

Outro teste possível é através do comando: # *tracert*, que é uma ferramenta de teste de conexão, que mostra todos os roteadores que o pacote IPv4 percorre até

atingir seu destino. A figura 15 mostra os resultados do comando, testando a conexão do roteador RT_C com as interfaces LAN dos roteadores RT_A, RT_B, RT_D e o *Firewall* da rede, respectivamente.

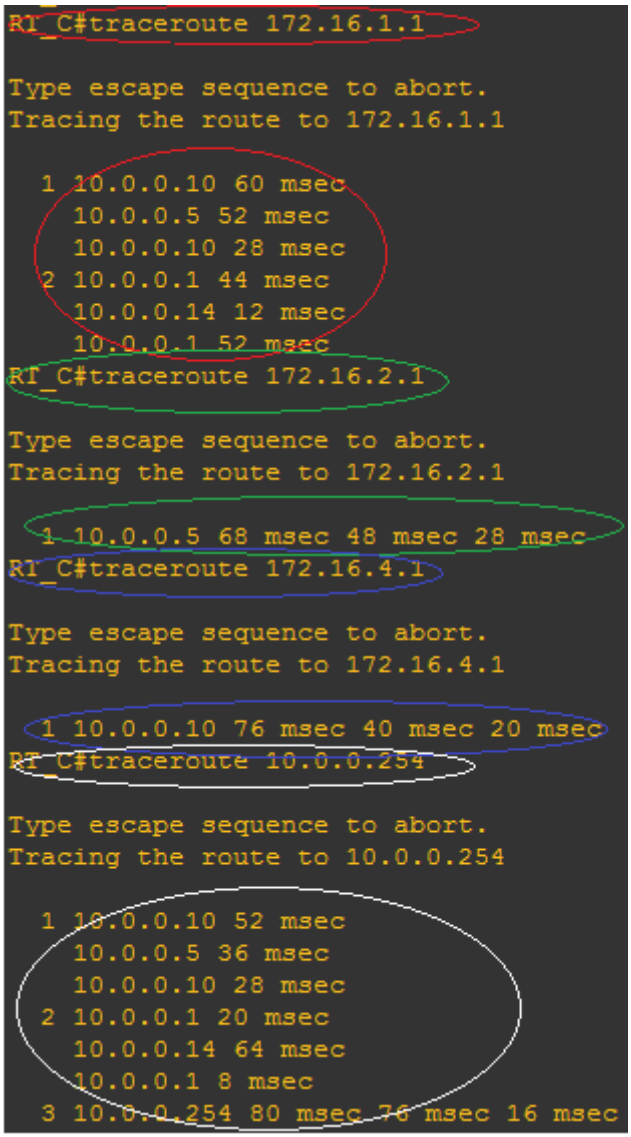


Figura 15 - Teste de Conexão com traceroute.
Fonte: Próprio autor

Comando: *traceroute* do roteador RT_C para as interfaces LAN dos roteadores: RT_A, RT_B, RT_D e o Firewall da rede, respectivamente.

3.1.2 IPv6

Para realizar a configuração do protocolo IPv6, baseou-se na topologia

mostrada na figura 16.

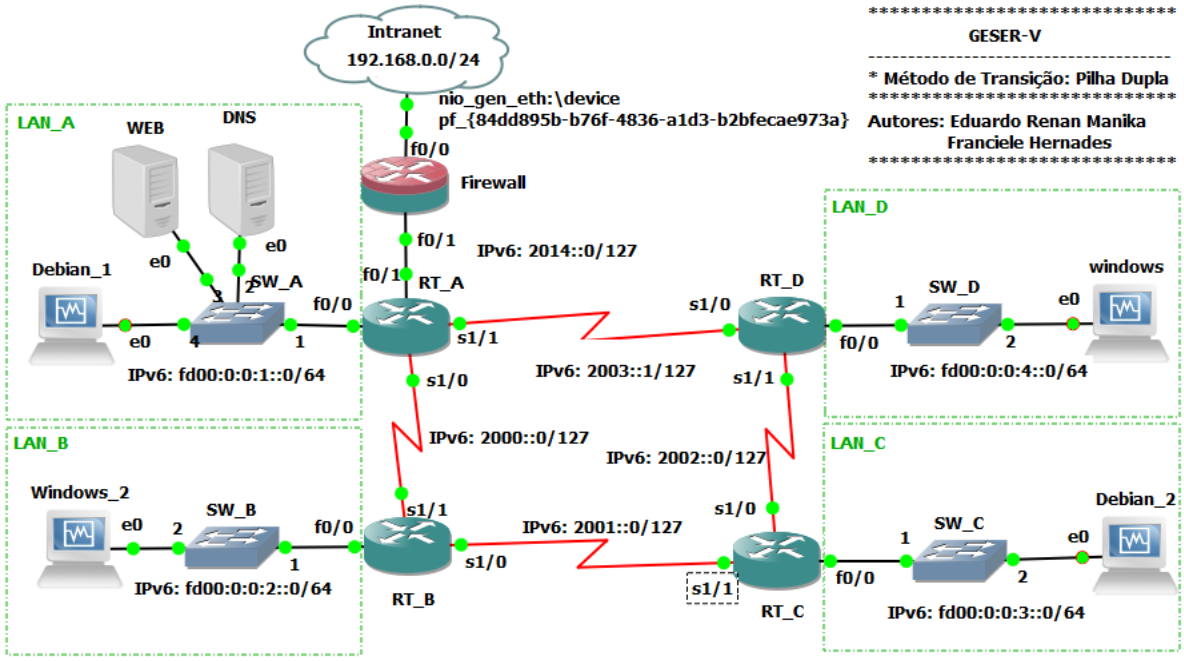


Figura 16 - Topologia da Rede – IPv6.
Fonte: Próprio autor

Para configurar a topologia conforme a especificação é necessário os seguintes passos:

- Configuração das interfaces dos roteadores, com endereço IPv6;
- Configuração do protocolo de roteamento OSPF, com suporte a versão IPv6 – OSPFv3;
- Configuração dos hosts, utilizando DHCPv3;
- Configuração dos servidores, DNS e Web.

Para configurar as interfaces dos roteadores foram utilizados os comandos, conforme figura 17 e as configurações utilizadas para cada equipamento da rede foram realizadas de acordo com a tabela 9.

```
-----------------------------------------------------------------------
                Configuração das Interfaces – IPv6
-----------------------------------------------------------------------
RT_A>enable                        //Entra no modo exec privilegiado
RT_A#configure terminal            //Entra no modo de configuração
-----------------------------------------------------------------------
Configurar a Interface f0/0:
-----------------------------------------------------------------------
RT_A(config)#interface fastEthernet 0/0       //Entra na interface
RT_A(config-if)#ipv6 address fd00:0:0:1::0/64 //Seta o endereço
RT_A(config-if)#exit               //Volta para o modo de conf. global
-----------------------------------------------------------------------
Configurar a Interface f0/1:
-----------------------------------------------------------------------
RT_A(config)#interface fastEthernet 0/1       //Entra na interface
RT_A(config-if)#ipv6 address 2014::1/127      //Seta o endereço
RT_A(config-if)#exit            // Volta para o modo de conf. global
-----------------------------------------------------------------------
Configurar a Interface s1/0:
-----------------------------------------------------------------------
RT_A(config)#interface serial 1/0             //Entra na interface
RT_A(config-if)#ipv6 address 2000::0/127      //Seta o endereço
RT_A(config-if)#exit            // Volta para o modo de conf. global
-----------------------------------------------------------------------
Configurar a Interface s1/1:
-----------------------------------------------------------------------
RT_A(config)#interface serial 1/1             //Entra na interface
RT_A(config-if)#ipv6 address 2003::1/127      //Seta o endereço
RT_A(config-if)#exit            // Volta para o modo de conf. Global
-----------------------------------------------------------------------
```

Figura 17 - Configuração das Interfaces – IPv6.
Fonte: Próprio autor

Equipamento	Protocolo	Interface	Endereço	Máscara	Gateway
		f0/0	-	-	-
Firewall	IPv6	f0/1	2014::0	127	N/A
		f0/0	FD00:0:0:1::0	64	N/A
		f0/1	2014::1	127	N/A
		s1/0	2000::0	127	N/A
RT_A	IPv6	s1/1	2003::1	127	N/A
		f0/0	FD00:0:0:2::0	64	N/A
		s1/0	2001::0	127	N/A
RT_B	IPv6	s1/1	2000::1	127	N/A
		f0/0	FD00:0:0:3::0	64	N/A
		s1/0	2002::0	127	N/A
RT_C	IPv6	s1/1	2001::1	127	N/A
		f0/0	FD00:0:0:4::0	64	N/A
		s1/0	2003::0	127	N/A
RT_D	IPv6	s1/1	2002::1	127	N/A
Servidor DNS	IPv6	f0/0	FD00:0:0:1::1	64	FD00:0:0:1::0
Servidor Web	IPv6	f0/0	FD00:0:0:1::2	64	FD00:0:0:1::0

Tabela 9: Configurações IPv6

Fonte: Próprio autor

Para configurar as interfaces com o protocolo IPv6 é necessário entrar em cada interface do roteador, já configurado com o endereço IPv4 e adicionar o endereço IPv6. Para adicionar o endereço IPv6 é utilizado o comando: *#ipv6 address <Endereço_IPv6>/<Máscara_de_Rede>.*

Após configura o endereço IPv6 em todas as interfaces, é possível verificar o arquivo de configuração do roteador, com o comando: *#show running-config*, para validar os endereços configurados. Na figura 18 é apresentado o trecho do arquivo de configuração com os endereços IPv6 configurados.

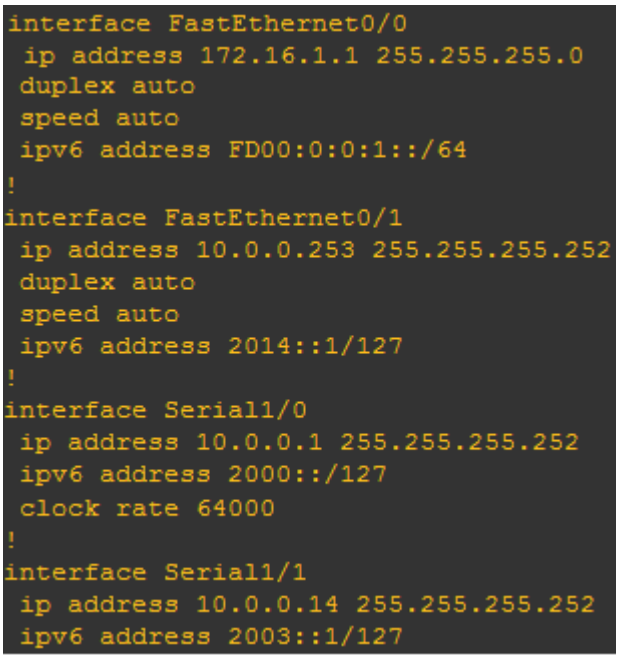


Figura 18 - Arquivo de configuração – IPv6.
Fonte: Próprio autor

3.1.2.1 Configuração do Protocolo de Roteamento – OSPFv3

Para configurar o OSPF com suporte ao protocolo IPv6 – OSPFv3 é necessário habilitar o protocolo de roteamento IPv6, através do comando: #*ipv6 unicast-routing*. É necessário também configurar o identificador com roteador, que é utilizado nos processos do protocolo de roteamento, através do comando: #*router-id* <*numero_identificador*>. Este número identificador é representado igual ao endereço IPv4, no OSPF com suporte IPv4 esse comando não era necessário, pois o menor endereço setar na interface era utilizado para esse identificador. Já no IPv6, como não apresenta esse endereço, tornou-se um comando obrigatório na configuração do OSPFv3.

Para setar as redes que o roteador esta configurado, na configuração IPv4 essa configuração é realizada separadamente. No IPv6 essa configuração é realizada na própria interface, através do comando: # *ipv6 ospf* <*identificador_processo_OSPF*> *area* <*identificador_da_área*>. No caso o processo do OSPF foi definido com 1 e a área 0. Esses procedimentos é descrito na figura 19, apresentando o passo a passo da configuração, no exemplo o roteador RT_A.

```
-----------------------------------------------------------------
                Configuração do OSPFv3 – IPv6
-----------------------------------------------------------------
RT_A>enable                      //Entra no modo exec privilegiado
RT_A#configure terminal          //Entra no modo de configuração
-----------------------------------------------------------------
Configurar o OSPFv3:
-----------------------------------------------------------------
RT_A(config)#ipv6 unicast-routing
RT_A(config)#ipv6 router ospf 1          //id do processo OSPF
RT_A(config-rtr)#router-id 1.1.1.1       //identificador do roteador
RT_A(config-rtr)#exit            //Volta para o modo de conf. Global
-----------------------------------------------------------------
Configurar o OSPFv3 –
Necessário entrar em cada interfaces e adicionar o comando:
#ipv6 ospf <id_processo> área <id_area>
-----------------------------------------------------------------
RT_A(config)#interface fastEthernet 0/0  //Entra na interface
RT_A(config-if)#ipv6 ospf 1 area 0
RT_A(config-router)#exit       // Volta para o modo de conf. Global
RT_A(config)#interface fastEthernet 0/1  //Entra na interface
RT_A(config-if)#ipv6 ospf 1 area 0
RT_A(config-router)#exit       // Volta para o modo de conf. Global
RT_A(config)#interface serial 1/0  //Entra na interface
RT_A(config-if)#ipv6 ospf 1 area 0
RT_A(config-router)#exit       // Volta para o modo de conf. Global
RT_A(config)#interface serial 1/1  //Entra na interface
RT_A(config-if)#ipv6 ospf 1 area 0
RT_A(config-router)#exit       // Volta para o modo de conf. Global
-----------------------------------------------------------------
```

Figura 19 - Configuração OSPFv3.
Fonte: Próprio autor

3.2 PILHA DUPLA

Com as interfaces configuradas com ambos os protocolos IPs, versão 4 e versão 6, é possível verificar a aplicação do método de transição do IPv4-IPv6: Pilha Dupla. No qual, ambos os protocolos operam simultaneamente na rede, sem um interferir na operação do outro.

Para verificar os pacotes trafegando na rede é utilizado o *Software Wireshark*, que permite identificar o tipo de pacote, a versão do protocolo e permite analisar o cabeçalho do mesmo. Para demonstrar o funcionamento a Pilha Dupla, no host “*Debian 2*”, host *Linux* da Lan_C, foi realizado um solicitação de *ping* na versão IPv4 ao servidor DNS – 172.16.1.2 e um *ping* na versão IPv6 ao servidor Web –

fd00:0:0:1::2, simultaneamente e capturado e apresentado os referido pacotes utilizando o *Wireshark*. Na figura 20 é apresentado um pacote *ping* IPv4 capturado, sinalizado com a cor vermelha e apresentado o seu cabeçalho logo abaixo, e na cor verde é apresentado um pacote *ping* IPv6 trafegando simultaneamente.

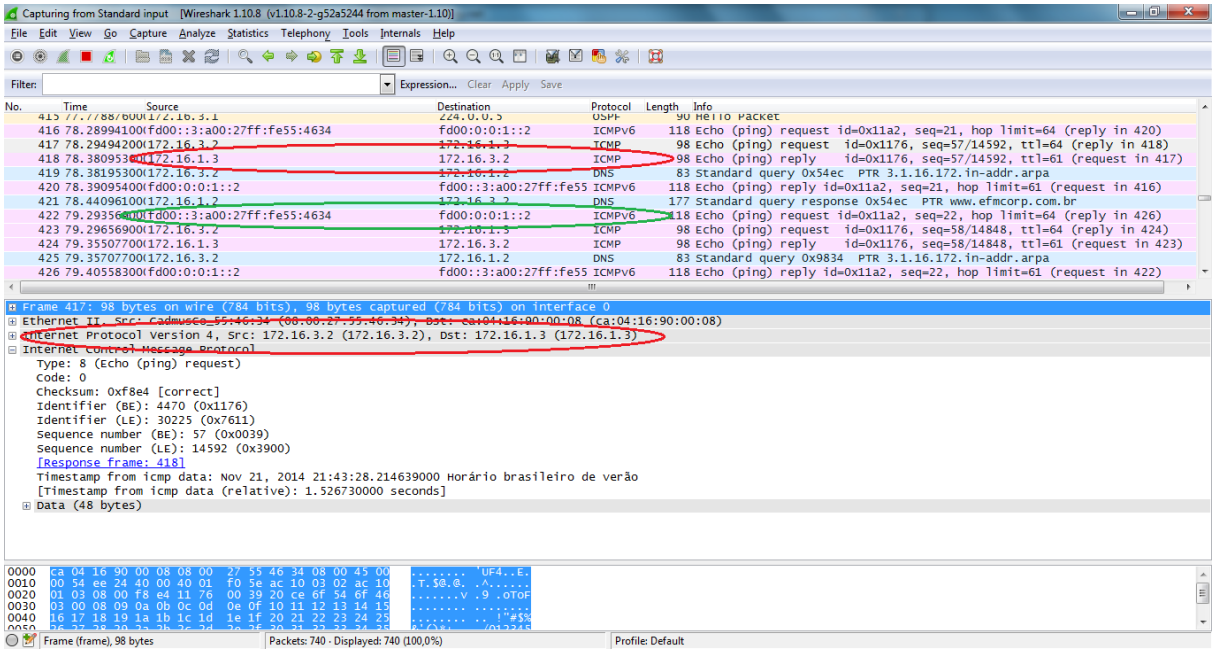


Figura 20 - Capturado do pacote ping (ICMP) - IPv4, em vermelho, operando junto com o pacote ping (ICMPv6) – IPv6 em verde.
Fonte: Próprio autor

Na figura 21 é apresentado um pacote *ping* IPv6 capturado e apresentado o seu cabeçalho, representado na cor verde e em vermelho mostra um pacote *ping* IPv4. Com isso é possível validarmos os dois protocolos operando simultaneamente, sem um interferir na operação do outro.

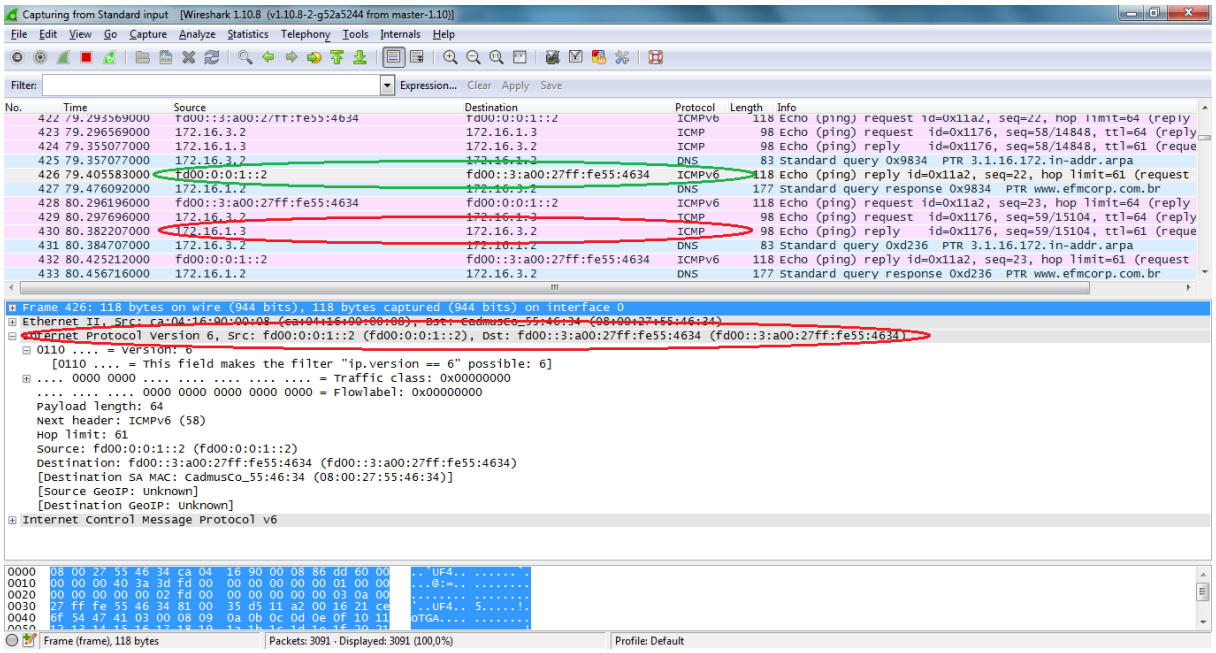


Figura 21 - Capturado do pacote ping (ICMPv6) – IPv6, em verde, operando junto com o pacote ping (ICMP) – IPv4 em vermelho.
Fonte: Próprio autor

3.2 CONFIGURAÇÃO DO SERVIDOR DNS

Outro objetivo desse trabalho, além de demonstrar o funcionamento do mecanismo de transição IPv4-IPv6, Pilha Dupla, é apresentar a configuração do serviço de DNS em ambos os protocolos IP, demonstrando as diferenças na configuração de ambos.

3.2.1 Configuração do Servidor DNS – IPv4

Para configurar o serviço de DNS no ambiente de simulação foi utilizado um servidor específico, utilizando a plataforma *Linux*, distribuição Debian Server 7.0, ou seja, distribuição que não apresenta a interface gráfica, toda a configuração é realizada via linha de comando. O software utilizado é o *Bind9*, que apresenta o suporte a ambos os protocolos IP. Para instalar o *Bind9* via linha de comando é executado os seguintes comandos, lembrando que os comandos apresentados

devem ser executados como permissão de super usuário: *root*:

```
#apt-get update

#apt-get install bind9
```

Após a instalação do serviço, os arquivos de configuração padrão encontram-se na pasta: */etc/bind*. O nome do domínio utilizado na configuração do DNS foi: "*efmcorp.com.br*". Para iniciar a configuração do DNS com suporte ao IPv4 é necessário adicionar no arquivo: */etc/hosts* – que é uma arquivo local da máquina responsável por realizar a tradução de endereços IP por nomes locais – os seguintes comandos:

```
127.0.0.1    localhost.localdomain    localhost
172.16.1.2   dns.efmcorp.com.br       dns
```

Dessa forma caso seja enviado qualquer requisição para o endereço "*dns.efmcorp.com.br*" ou somente "*dns"*, esse pacote é enviado para o IPv4 172.16.1.2, que no caso é o próprio servidor.

Para configurar o DNS é necessário configurar duas zonas, ou seja, arquivos onde serão verificados a tradução do endereço direto – DNS Direto, de um nome para um endereço IP e a tradução de endereço reverso – DNS Reverso, de um endereço IP em um nome. Para o DNS direto foi especificado o arquivo: "*efmcorp.com.br.direto*" e para o DNS reverso foi especificado o arquivo: "*efmcorp.com.br.reverso*".

Inicialmente esse dois arquivos citados devem ser configurados no arquivo: */etc/bind/named.conf.local*, conforme apresentado na figura 22, para que o serviço de DNS tenho especificado os locais de busca para a tradução dos nomes.

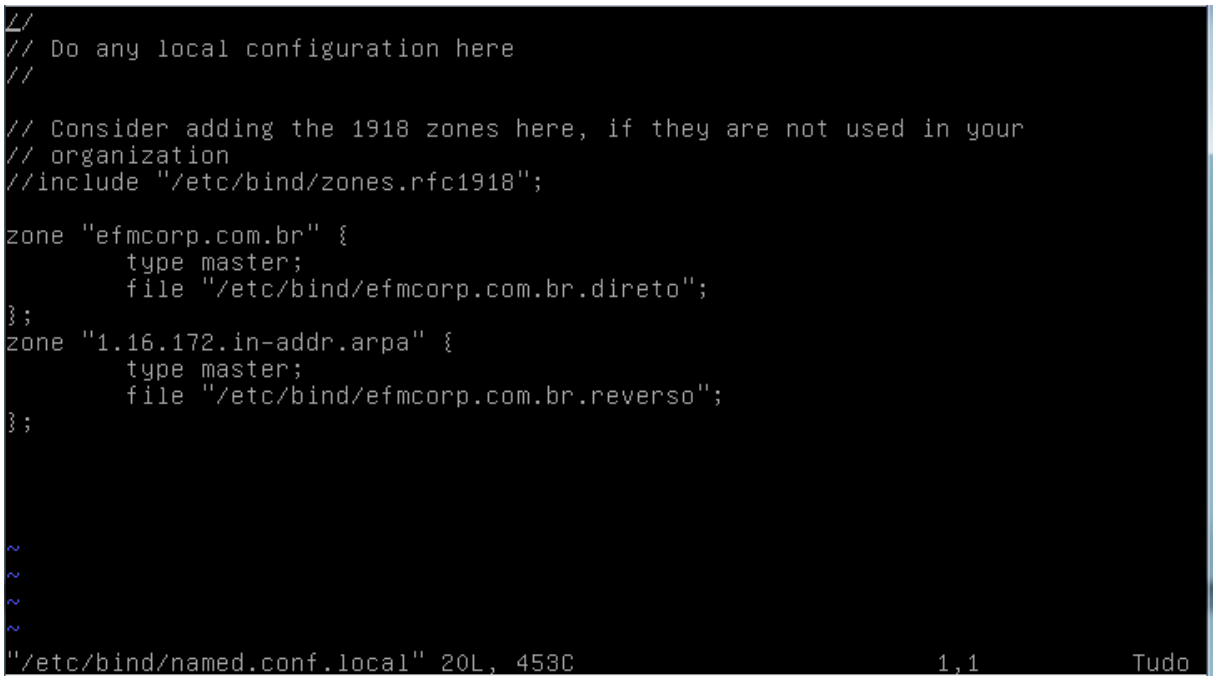


Figura 22 - Arquivo named.conf.local.
Fonte: Próprio autor

Para criar o arquivo de configuração DNS direto, “efmcorp.com.br.direto”, pode ser criado através da cópia do arquivo: */etc/bind/db.local*, servindo como modelo de configuração. Essa cópia é realizada através do comando:

```
# copy /etc/bind/db.local /etc/bind/efmcorp.com.br.direto
```

Após isso é necessário realizar a configuração do DNS direto conforme os nomes especificados anteriormente, na figura 23 é apresentado o arquivo de configuração do DNS direto. A opção denominada “*Serial*” foi padronizado uma codificação representada por <Ano><Mes><Dia><Número_Alterações_no_Mesmo_Dia>. Esse campo deve ser modificado a cada alteração no arquivo de configuração para que essa alteração possa ser aplicada o mais rápido possível.

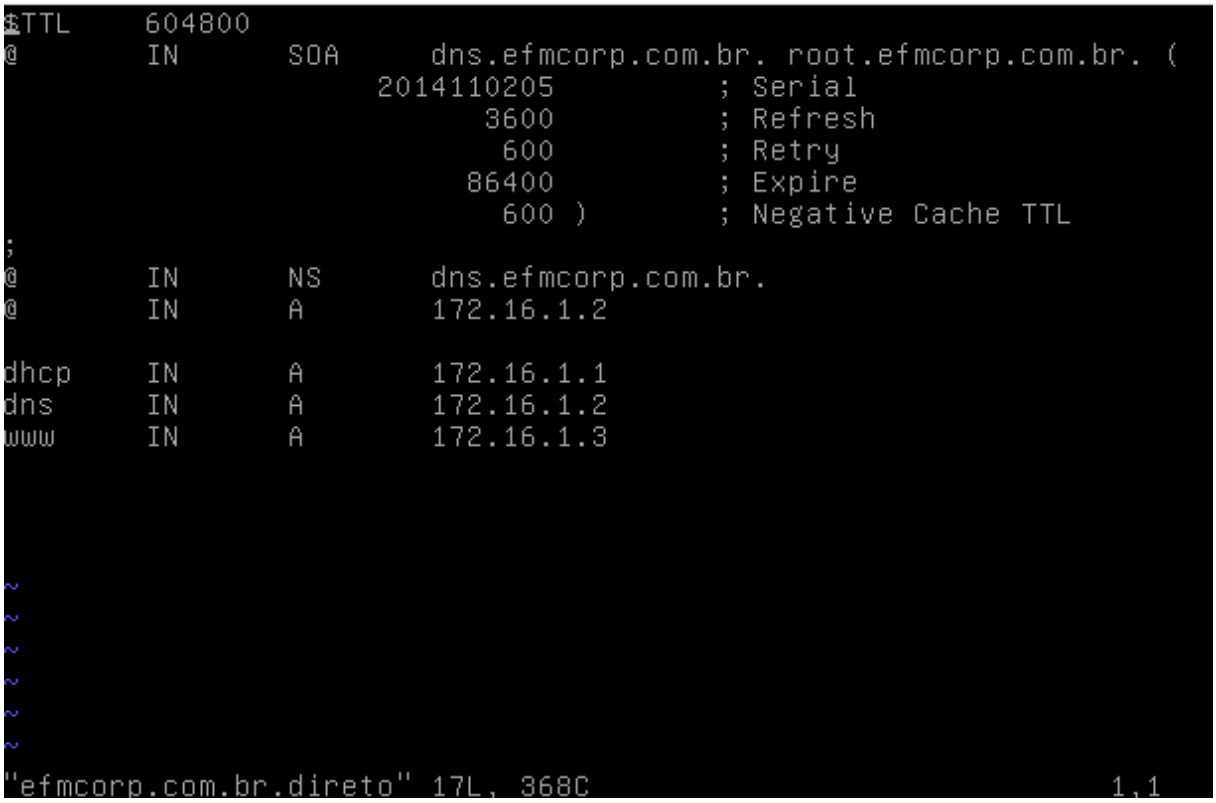


Figura 23 - Configuração DNS direto.
Fonte: Próprio autor

Nesse arquivo foi configurado três nome que apontaram seus respectivos IPs, “dhcp”, “dns”, “www”. Quando solicitado qualquer resolução de um desses nomes cadastrados, seguido do nome do domínio que o servidor DNS é responsável, o servidor retornará seu endereço IPv4 cadastrado. Por exemplo: uma requisição de resolução do endereço: “www.efmcorp.com.br”, o servidor DNS retornará o endereço IPv4: 172.16.1.3. Lembrando que pode ser configurado mais um nome para um determinado endereço IP, ficando a critério do administrador do serviço.

Para configurar o DNS Reverso, ou seja, de um endereço IPv4 descobrir qual ou quais nomes o endereço IP corresponde, é necessário configurar o arquivo: “efmcorp.com.br.reverso”, conforme especificado no arquivo de configuração “name.conf.local”. Para criar esse arquivo é possível criar um cópia do arquivo: “/etc/bind/db.127” que serve como modelo desse arquivo e é um arquivo padrão da

instalação do serviço de Bind9. Para realizar essa cópia é necessário executar o seguinte comando:

```
# copy /etc/bind/dB.127 /etc/bind/efmcorp.com.br.reverso
```

Com o arquivo criado é necessário configurá-lo conforme descrito na figura 24, no qual deve ser configurado os domínios e o campo “Serial”, seguindo o mesmo padrão descrito no arquivo DNS – Direto. No arquivo de configuração: “named.conf.local” foram configurados duas zonas, uma que representa o DNS – Direto e outra do DNS – Reverso. No DNS – Reverso contem a seguinte configuração: “1.16.172.in-addr.arpa”, observamos que os números representados equivalem a porção de rede do endereço IPv4, descrito de forma inversa, da LAN_A, onde encontram-se os servidores do ambiente simulado. Dessa forma no arquivo de configuração do DNS – Reverso é necessário configurar a porção de hosts do endereço IPv4, onde o número, por exemplo “3”, ou seja, o endereço 172.16.1.3 equivale ao endereço IPv4 do servidor Web e será resolvido pelo DNS para o endereço: “*www.efmcorp.com.br*”.

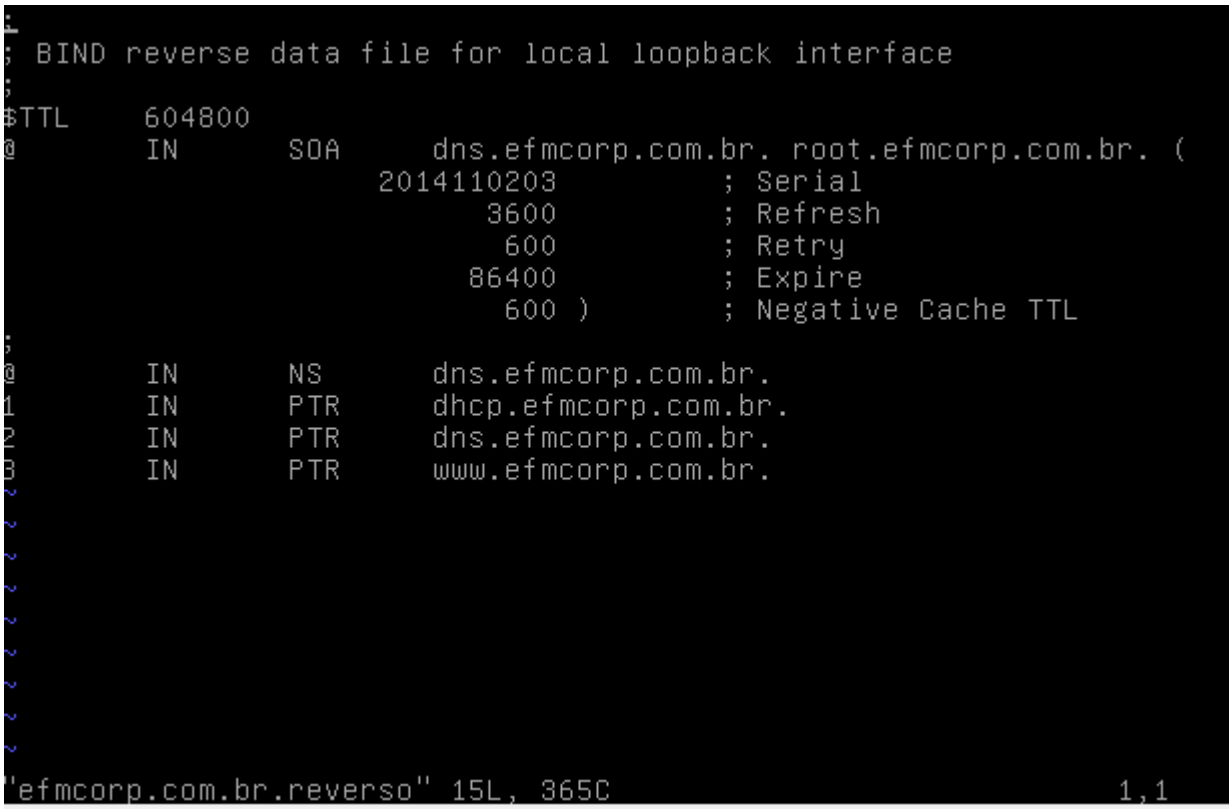


Figura 24 - Configuração DNS reverso.
Fonte: Próprio autor

Após a configuração dos arquivos é possível verificar se a configuração esta correta, realizando uma validação no arquivo: "*named.conf*" e nas zonas configuradas: ""efmcorp.com.br.direto" e "efmcorp.com.br.reverso", através dos comandos:

```
# named-checkconf /etc/bind/named.conf

# named-checkzone efmcorp.com.br efmcorp.com.br.direto

# named-checkzone efmcorp.com.br efmcorp.com.br.reverso

# named-checkzone 1.16.172 efmcorp.com.br.reverso
```

O resultado desses comandos é apresentado na figura 25.

```
root@dns:/etc/bind# named-checkconf /etc/bind/named.conf
root@dns:/etc/bind#
root@dns:/etc/bind#
root@dns:/etc/bind# named-checkzone efmcorp.com.br efmcorp.com.br.direto
zone efmcorp.com.br/IN: loaded serial 2014110206
OK
root@dns:/etc/bind#
root@dns:/etc/bind#
root@dns:/etc/bind# named-checkzone 1.16.172  efmcorp.com.br.reverso
zone 1.16.172/IN: loaded serial 2014110203
OK
root@dns:/etc/bind#
```

Figura 25 - Resultados do comando named-checkconf.
Fonte: Próprio autor

Com as validações realizadas pode iniciar o servidor de DNS através do comando:

```
# /etc/init.d/bind9 restart
```

Com o serviço de DNS iniciado pode ser realizado os testes de resolução de nomes. Para isso existem algumas ferramentas quer realizaram esse teste, um deles é o “dig”. Permite realizar o teste direto, ou seja, de nome para um endereço IP, conforme apresentado na figura 26, no qual foi realizado o seguinte comando:

```
# dig www.efmcorp.com.br a
```

No qual é solicitado ao servidor DNS a resolução do endereço: “www.efmcorp.com.br” e retornar o endereço IPv4 correspondente, através do “a” adicionado no final do comando. Posteriormente será apresentado que esse comando mudará quando solicitado o retorno do endereço IPv6.

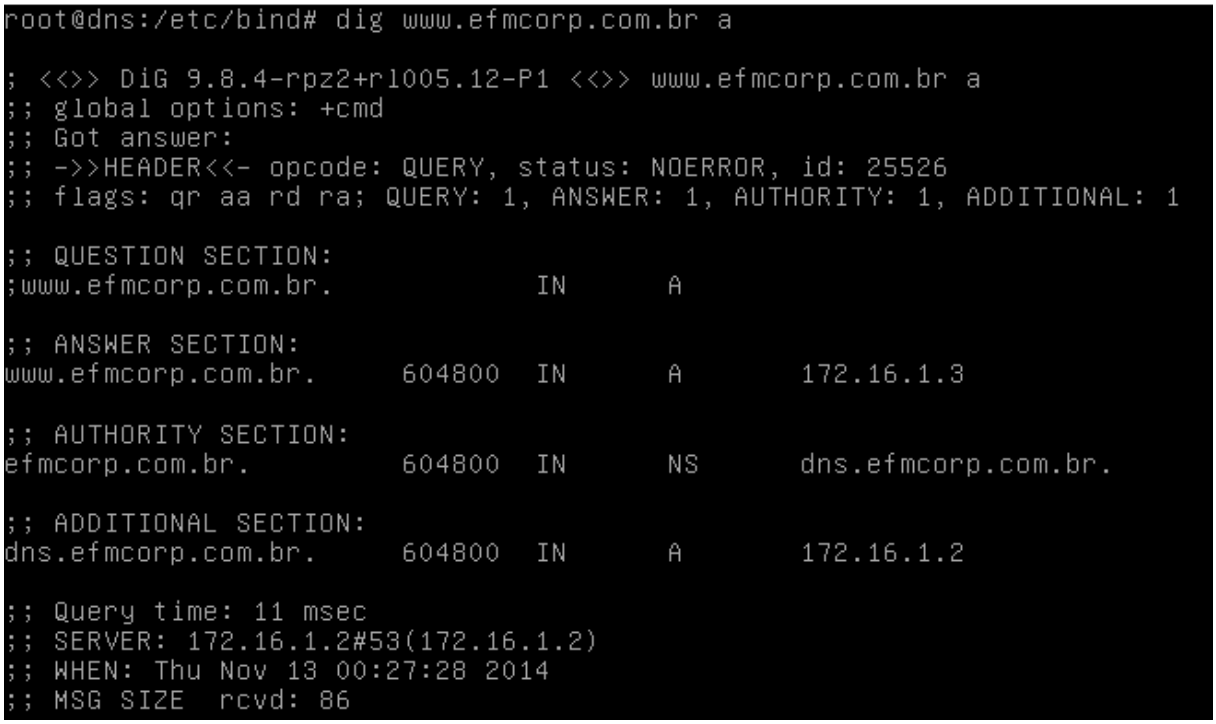


Figura 26 - Comando dig.
Fonte: Próprio autor

Como resultado do comando observa-se que o endereço retornado é “172.16.1.3”, que equivale ao servidor Web que corresponde ao endereço: “www.efmcorp.com.br”.

Para solicitar a resolução do DNS – Reverso, de um endereço IPv4 descobrir qual o endereço corresponde, pode ser utilizado o mesmo comando: “dig”, porém com a opções “–x” antes do endereço. Na figura 27 é apresentado o resultado da resolução do endereço 172.16.1.3, servidor Web, através do comando:

```
# dig –x 172.16.1.3
```

```
root@dns:/etc/bind# dig -x 172.16.1.3

; <<>> DiG 9.8.4-rpz2+rl005.12-P1 <<>> -x 172.16.1.3
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 44296
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;3.1.16.172.in-addr.arpa.       IN      PTR

;; ANSWER SECTION:
3.1.16.172.in-addr.arpa. 604800 IN      PTR     www.efmcorp.com.br.

;; AUTHORITY SECTION:
1.16.172.in-addr.arpa.  604800  IN      NS      dns.efmcorp.com.br.

;; ADDITIONAL SECTION:
dns.efmcorp.com.br.     604800  IN      A       172.16.1.2

;; Query time: 19 msec
;; SERVER: 172.16.1.2#53(172.16.1.2)
;; WHEN: Thu Nov 13 00:24:06 2014
;; MSG SIZE  rcvd: 107
```

Figura 27 - Comando dig.
Fonte: Próprio autor

Outro comando possível para verificar o funcionamento do DNS é o comando: "*nslookup*", que apresenta um resposta mais simplificada da consulta, conforme mostrado na figura 28. No qual foi realizado uma requisição reversa e direta para o servidor Web da rede.

```
root@dns:/etc/bind# nslookup 172.16.1.3
Server:         172.16.1.2
Address:        172.16.1.2#53

3.1.16.172.in-addr.arpa name = www.efmcorp.com.br.

root@dns:/etc/bind#
root@dns:/etc/bind#
root@dns:/etc/bind# nslookup www.efmcorp.com.br
Server:         172.16.1.2
Address:        172.16.1.2#53

Name:   www.efmcorp.com.br
Address: 172.16.1.3
```

Figura 28 - Comando nslookup.
Fonte: Próprio autor

No arquivo de log: “*syslog*”, localizado na pasta: “*/var/log*” é possível verificar possíveis erros e problemas na configuração do serviço de DNS, através do comando:

```
# tail –f /var/log/syslog
```

3.2.2 Configuração do Servidor DNS – IPv6

Para configurar o DNS com suporte ao protocolo IPv6 é necessário configurar as zonas necessárias para o novo protocolo. Para o DNS – Direto pode ser utilizado a mesma zona configurada no IPv4, porém é necessário adicionar os endereços IPv6 que correspondem aos endereços. Para o DNS – Reverso é necessário configura um zona especifica, pois a codificação dos endereços é diferente. Para isso deve ser criar um novo arquivo reverso para o IPv6 como o nome: “fd00:0000:0000:0001.reverso” que também pode ser copiado do modelo: “db.127”, conforme realizado no IPv4. Após isso é necessário adicionar essa nova zona configurada no arquivo de configuração: “named.conf.local”, conforme descrito na figura 29, para que o DNS possa resolver o DNS – Reverso do endereço IPv6.

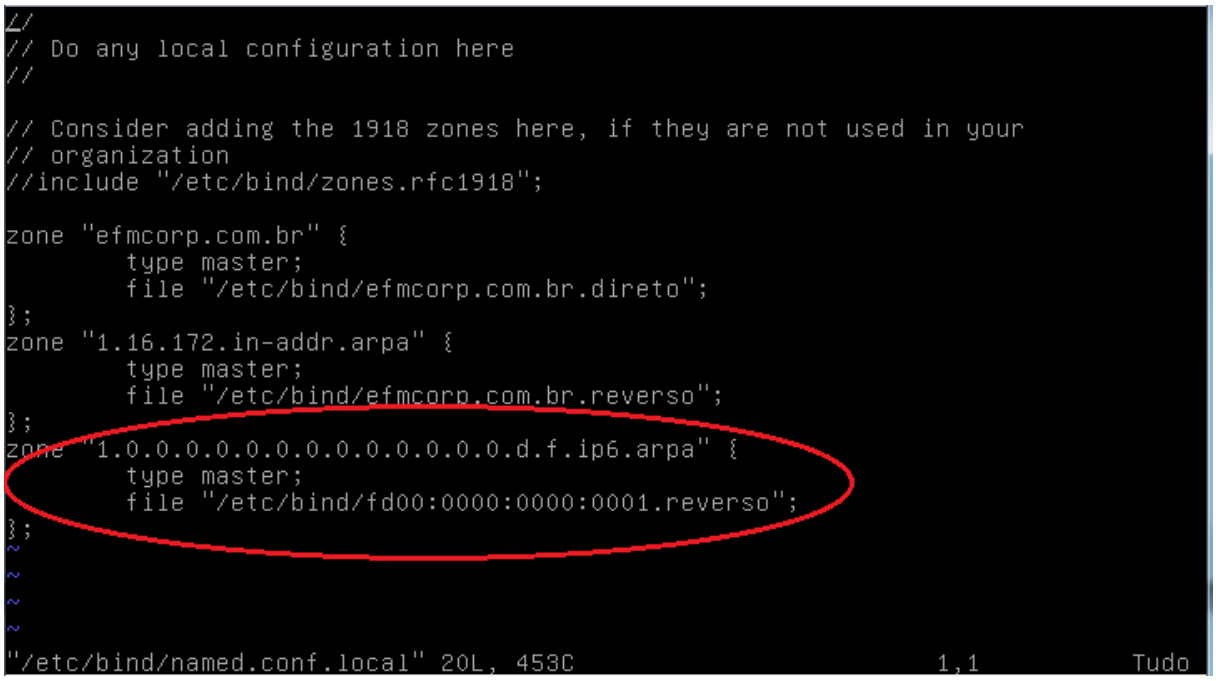


Figura 29 - Arquivo de configuração named.conf.local.
Fonte: Próprio autor

Para configurar o DNS – Direto para resolver os endereço no protocolo IPv6 é necessário adicionar os endereços IPv6 com a opções “AAAA”, no qual sinalizar para o serviço DNS que os endereços são configurados para essa versão. E no IPv4 essa opções é configurada com “A”. O arquivo: “efmcorp.com.br.direto” modificado é apresentado na figura 30. Os nomes “www” e “dns” estão configurados para responderem tanto com o protocolo IPv4 quanto o protocolo IPv6. Já os nomes: “www6” e “dns6” só estão configurados para responder o protocolo IPv6.

Figura 30 - Arquivo efmcorp.com.br.direto.
Fonte: Próprio autor

O novo arquivo de configuração do DNS – Reverso do protocolo IPv6: “fd00:0000:0000:0001.reverso” é descrito na figura 31. A zona reversa IPv6 configurada é “1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.d.f.ip6.arpa”, observa-se que os números representados equivalem a porção de rede do endereço IPv6 (/64), descrito de forma inversa, da LAN_A. Dessa forma no arquivo de configuração do DNS – Reverso do IPv6 é necessário configurar a porção de *hosts* do endereço IPv6, no caso os demais 64 bits, onde o número, por exemplo “2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0”, ou seja, o endereço: “fd00:0000:0000:0001:0000:0000:0000:0002” equivale ao endereço IPv6 do servidor Web e será resolvido pelo DNS para o endereço: “*www6.efmcorp.com.br*”.

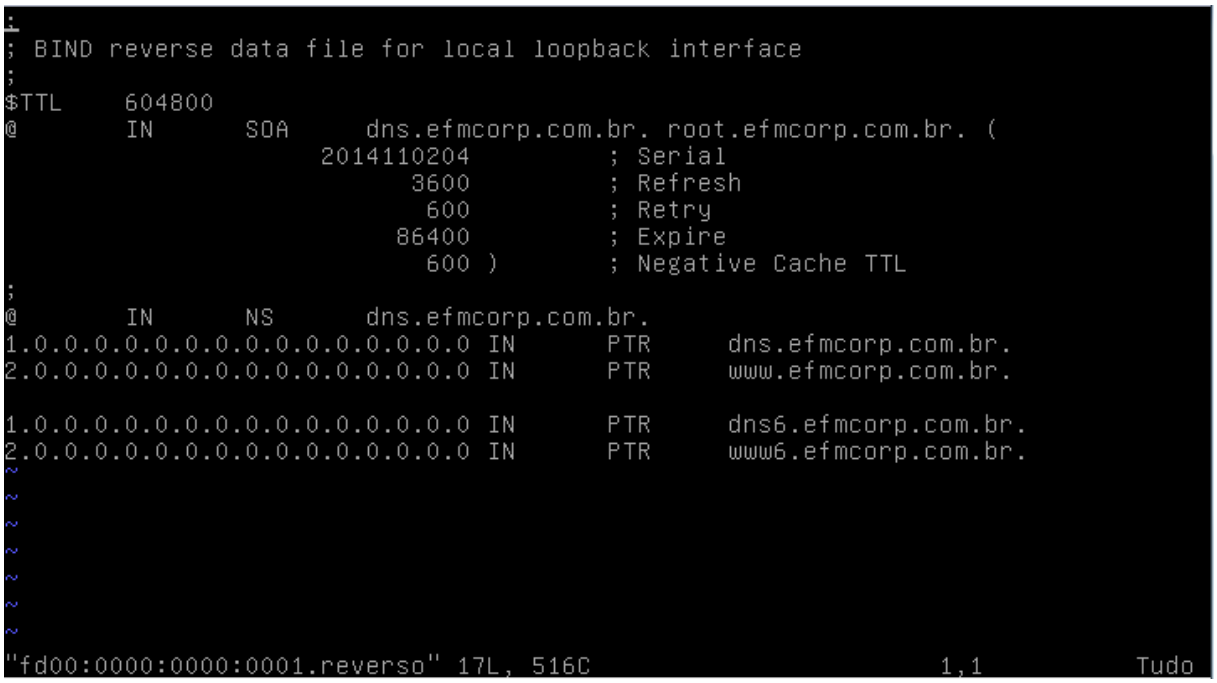


Figura 31 - Arquivo fd00:0000:0000:0001.reverso.
Fonte: Próprio autor

Para validar o arquivo de configuração reverso do protocolo IPv6 é possível utilizar o seguinte comando:

#named-checkzone 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.d.f.ip6.arpa fd00:0000:0000:0001.reverso

O resultado desse comando é apresentado na figura 32.

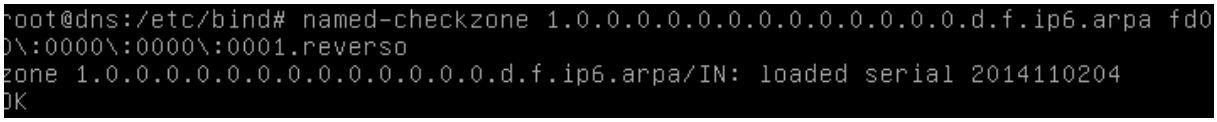


Figura 32 - Resultado do comando named-checkzone 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.d.f.ip6.arpa.
Fonte: Próprio autor

Com a validação ok, pode ser reiniciado o servidor DNS, através do comando:

/etc/init.d/bind9 restart

E realizado os testes de resolução do DNS para o novo protocolo. Para o teste do DNS – Direto foi utilizado o comando:

“# dig www6.efmcorp.com.br aaaa”

Deve-se destacar que no IPv6 após os endereço é adicionado a opção: “aaaa” o que representa ao DNS que retorne o endereço IPv6, conforme descrito na figura 33. Já no IPv4 esse opção era : “a”.

```
; <<>> DiG 9.8.4-rpz2+rl005.12-P1 <<>> www6.efmcorp.com.br aaaa
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 3285
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;www6.efmcorp.com.br.           IN      AAAA

;; ANSWER SECTION:
www6.efmcorp.com.br.    604800  IN      AAAA    fd00:0:0:1::2

;; AUTHORITY SECTION:
efmcorp.com.br.         604800  IN      NS      dns.efmcorp.com.br.

;; ADDITIONAL SECTION:
dns.efmcorp.com.br.     604800  IN      A       172.16.1.2

;; Query time: 13 msec
;; SERVER: 172.16.1.2#53(172.16.1.2)
;; WHEN: Thu Nov 13 00:30:46 2014
;; MSG SIZE  rcvd: 99
```

Figura 33 - Resultado comando dig www6.efmcorp.com.br aaaa.
Fonte: Próprio autor

Na figura 34 é apresentado o teste do DNS – Reverso utilizando o comando:

dig –x fd00:0:0:1::2

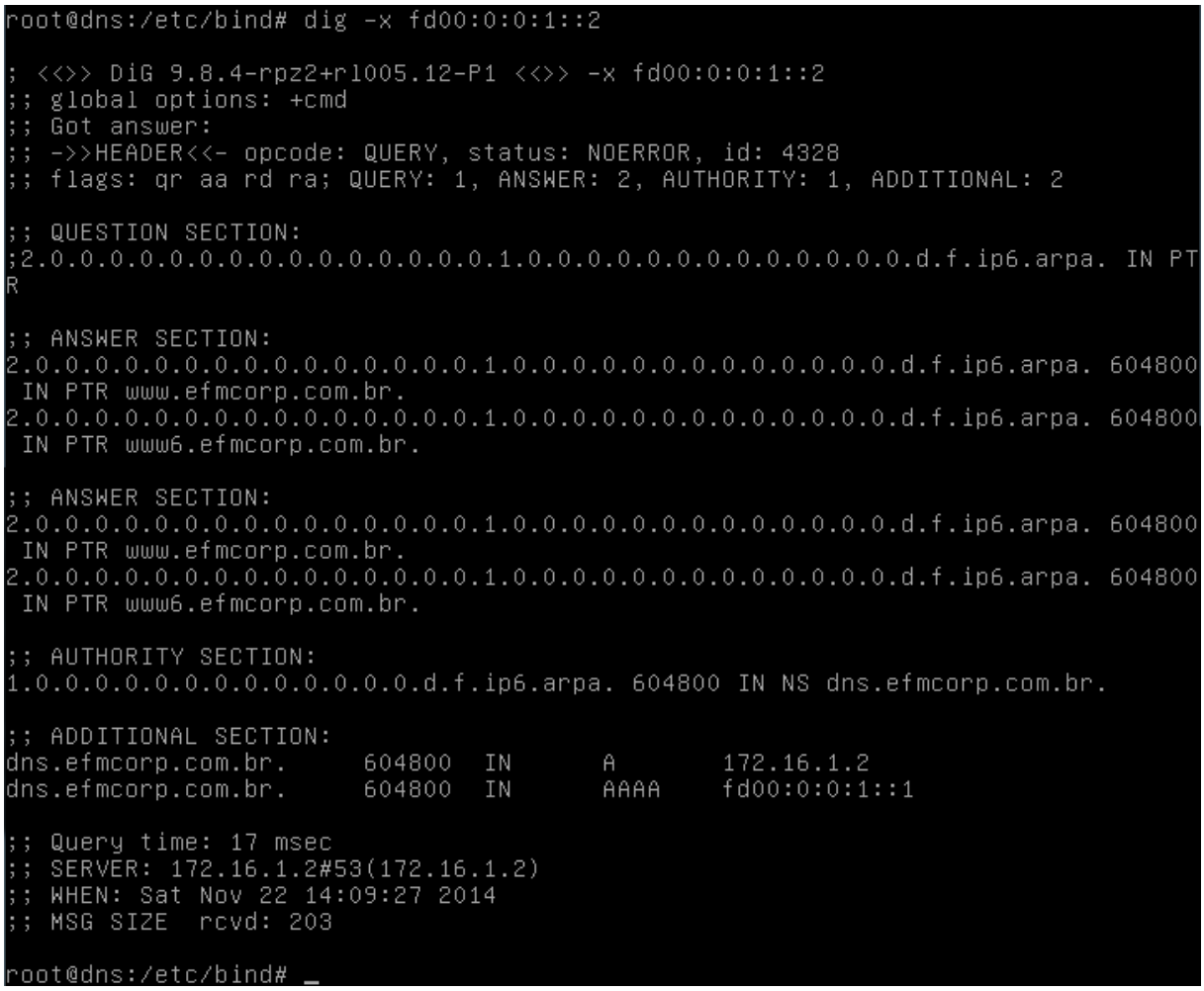


Figura 34 - Comando dig –x fd00:0:0:1::2.
Fonte: Próprio autor

Deve-se destacar que na configuração do DNS – Reverso do IPv6 foi configurado dois endereços para o mesmo endereço IPv6, quando consultado o DNS retorna os endereços: “www.efmcorp.com.br” e “www6.efmcorp.com.br”.

Como foi configurado no DNS – Direto o mesmo endereço para os dois protocolos IP, no caso o endereço: “www.efmcorp.com.br”, o DNS retorna em sua consulta o endereço IP de ambos os protocolos conforme descrito na figura 35. No qual o cliente: “Debian” da rede LAN_C realiza essa consulta para o endereço: “www.efmcorp.com.br” e é retornado os endereços: 172.16.1.3 (IPv4) e fd00:0:0:1::2 (IPv6)

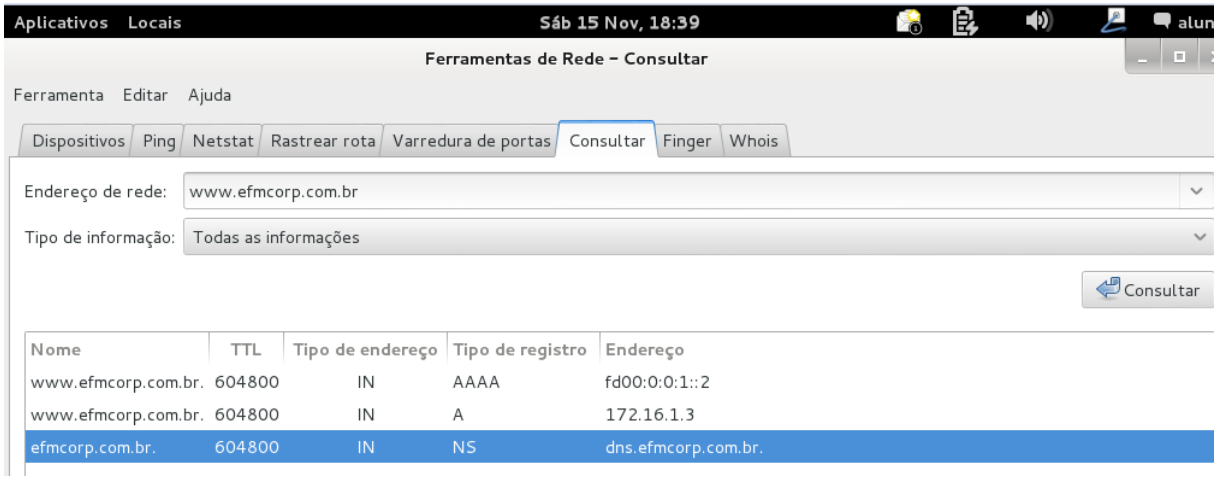


Figura 35 - Teste no cliente.
Fonte: Próprio autor

No cliente “Windows” da LAN_B foi realizado um teste de *ping* para o endereço: “www.efmcorp.com.br”, como o cliente apresenta ambos os endereços IP, recebidos via DHCP; serviço esse de distribuição de endereços não abordados nesse trabalho; o endereço resolvido foi o endereço IPv6 e realizado a consulta do ping utilizando o novo protocolo conforme descrito na figura 36.

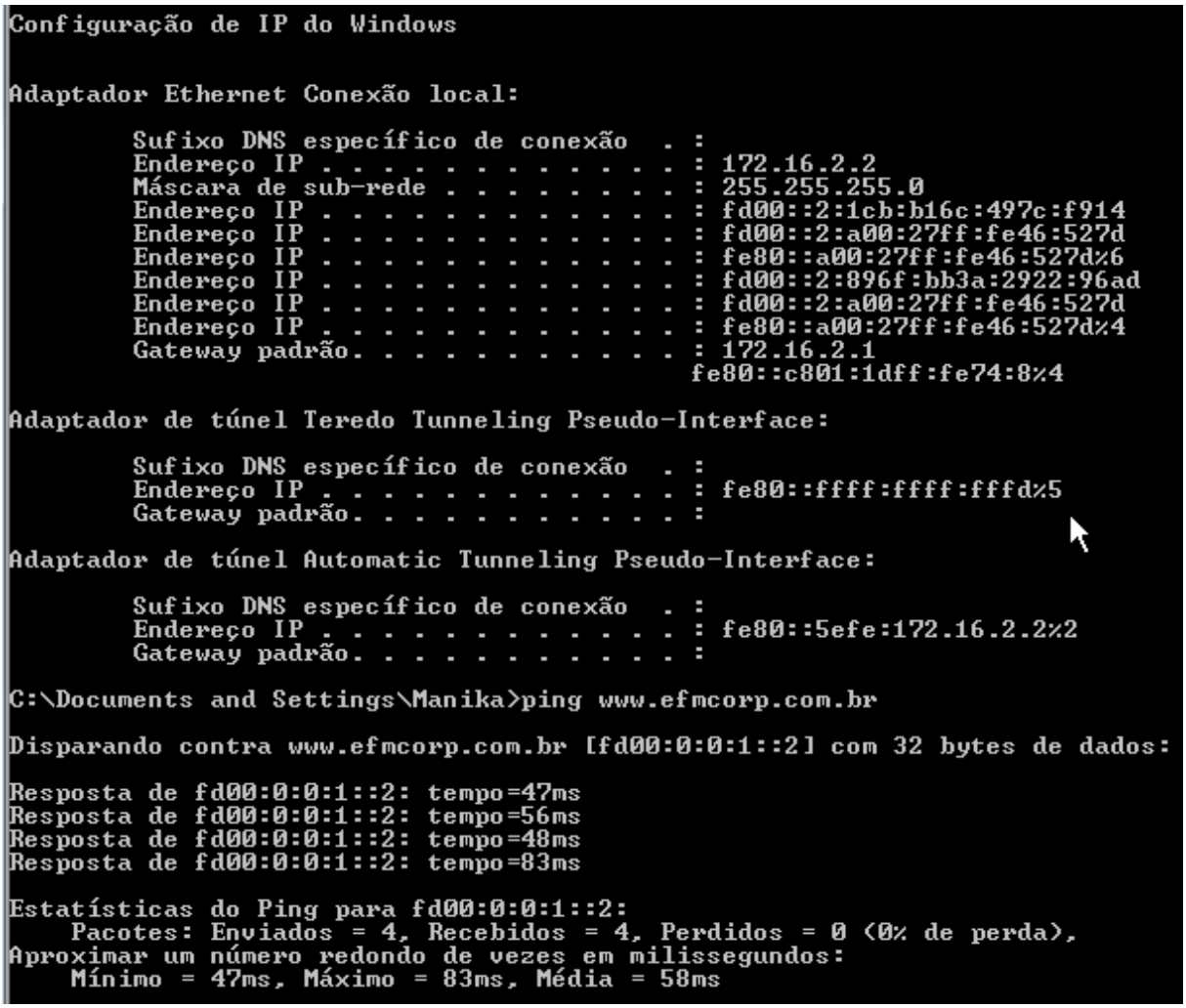


Figura 36 - Teste no cliente Windows.
Fonte: Próprio autor

3.3 CONFIGURAÇÃO DO SERVIDOR WEB

Para configurar um servidor Web foi utilizado o servidor a plataforma Linux distribuição Debian Server 7.0. Para o serviço de servidor web foi utilizado o pacote: "*apache2.2*", os passos de instalação são os seguintes comandos:

apt-get update

#apt-get install apache2.2

Os arquivos de configuração instalados ficam localizados na pasta: "*/etc/apache2*". A pasta com o arquivos publicados no servidor encontram-se no caminho: "*/var/www*". Nesta pasta encontra-se o arquivo: "*index.html*", como sendo o

primeiro arquivo carregando quando solicitado o serviço ao servidor.

Para realizar testes no ambiente de simulação foi modificada a página inicial, arquivo: “índex.html”, conforme descrito na figura 37.

Figura 37 - Arquivo do servidor Web.
Fonte: Próprio autor

Em sua configuração básica não é necessário configurações especificas para o servidor Web operar em ambos os protocolos IP. Somente necessário que a rede e o serviço de DNS operem em ambos os protocolos IP. Como o ambiente de simulação esta operando com ambos os protocolo e o serviço de DNS esta configurado para resolver nomes tanto para o protocolo IPv4 e IPv6 pode ser realizados os testes, para isso utilizado o navegador do cliente Linux da LAN_D, requisitando o servidor Web, utilizando o endereço IPv4, conforme demonstrado na figura 38.

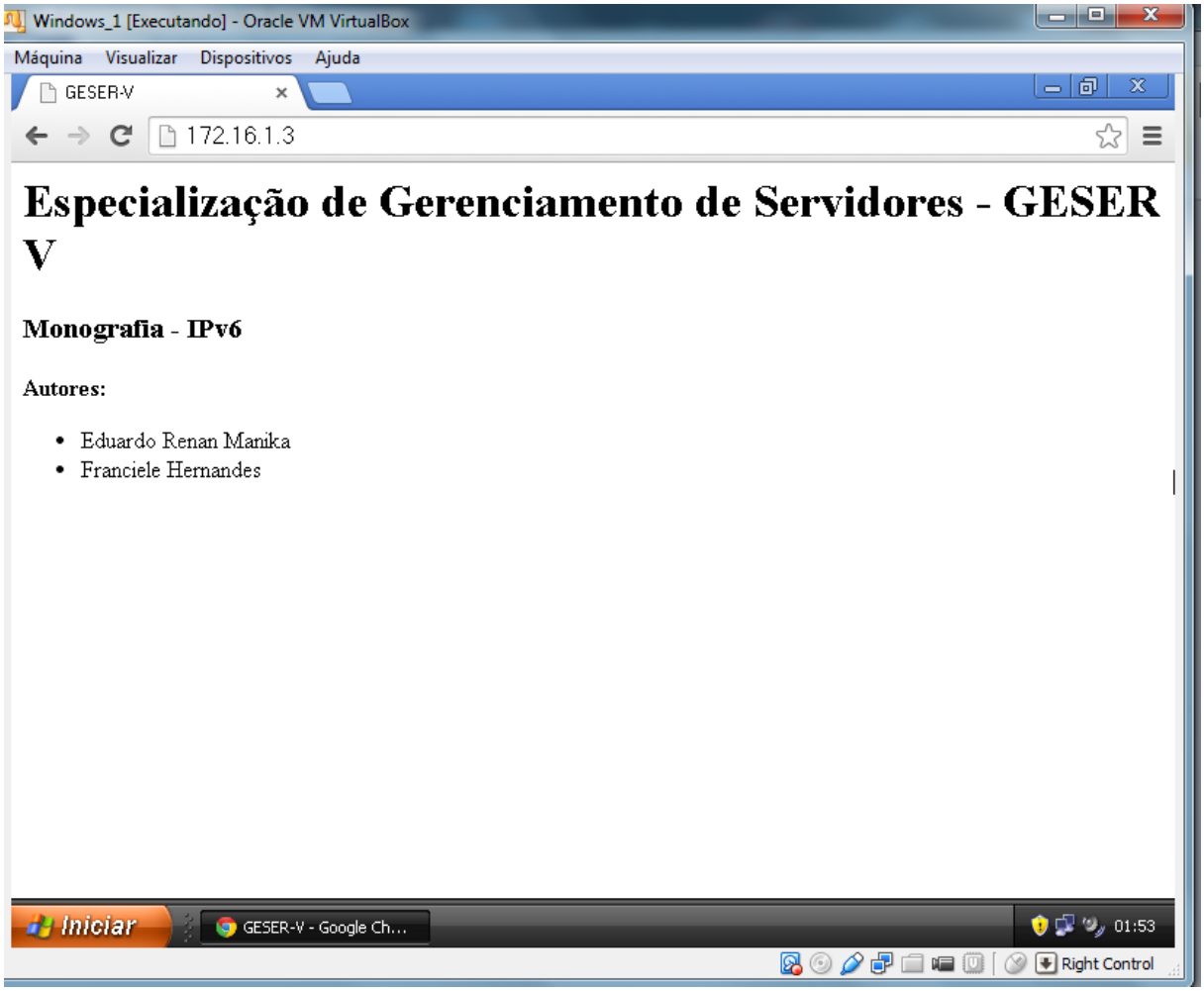


Figura 38 - Acesso a página web.
Fonte: Próprio autor

Na figura 39 apresenta a requisição utilizando o nome: “www.efmcorp.com.br”.

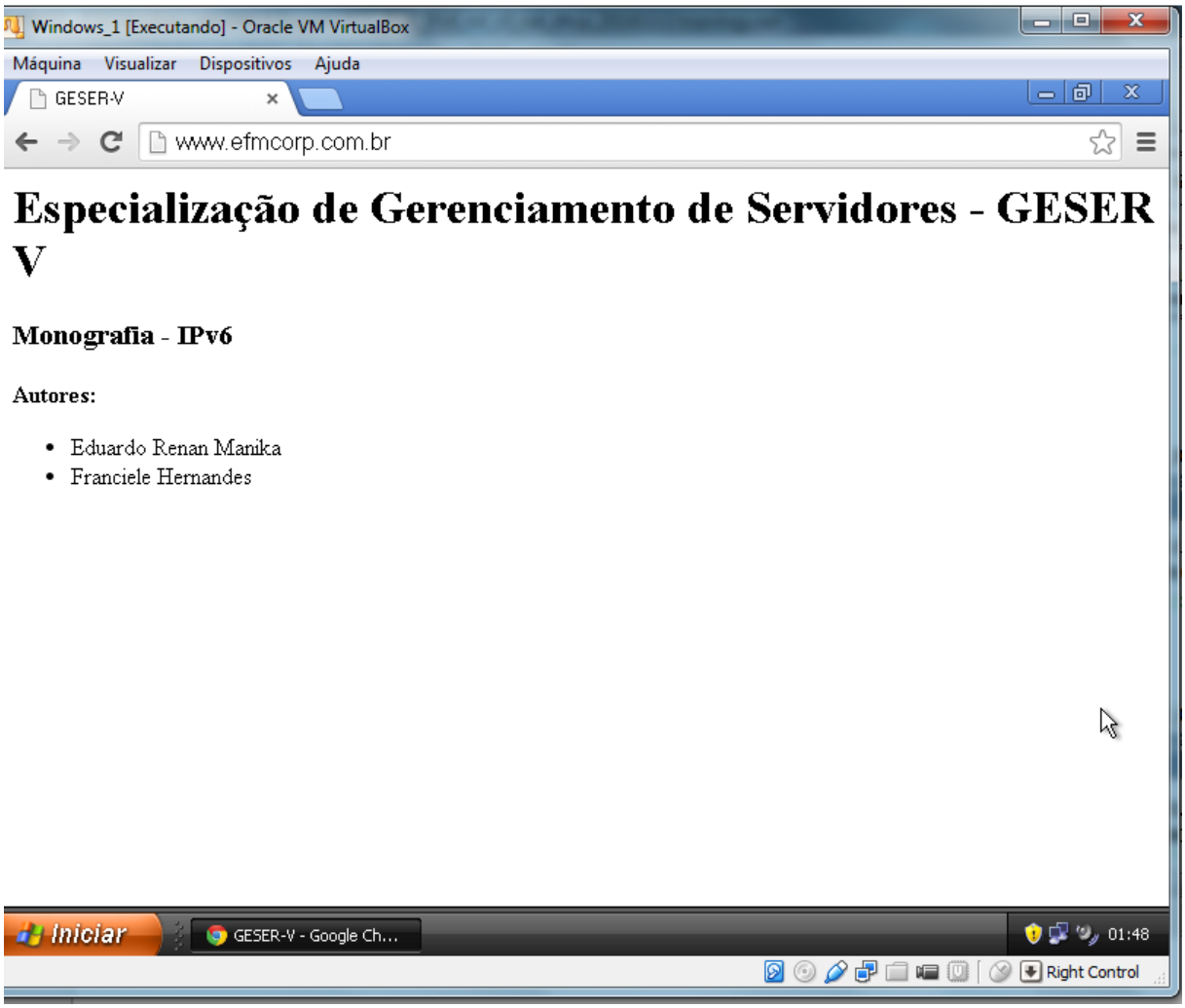


Figura 39 - Acesso a página web.
Fonte: Próprio autor

Na figura 40 é apresentado a consulta utilizando o endereço IPv6. Como esse endereço utiliza o caractere: “:” (dois- pontos) na composição de seu endereço e esse mesmo caractere é utilizado para designar a porta de comunicação. Devido a isso a RFC 2732 define que os endereços IPv6 quando utilizado diretamente na URL dos navegadores deve ser descrito entre colchetes.

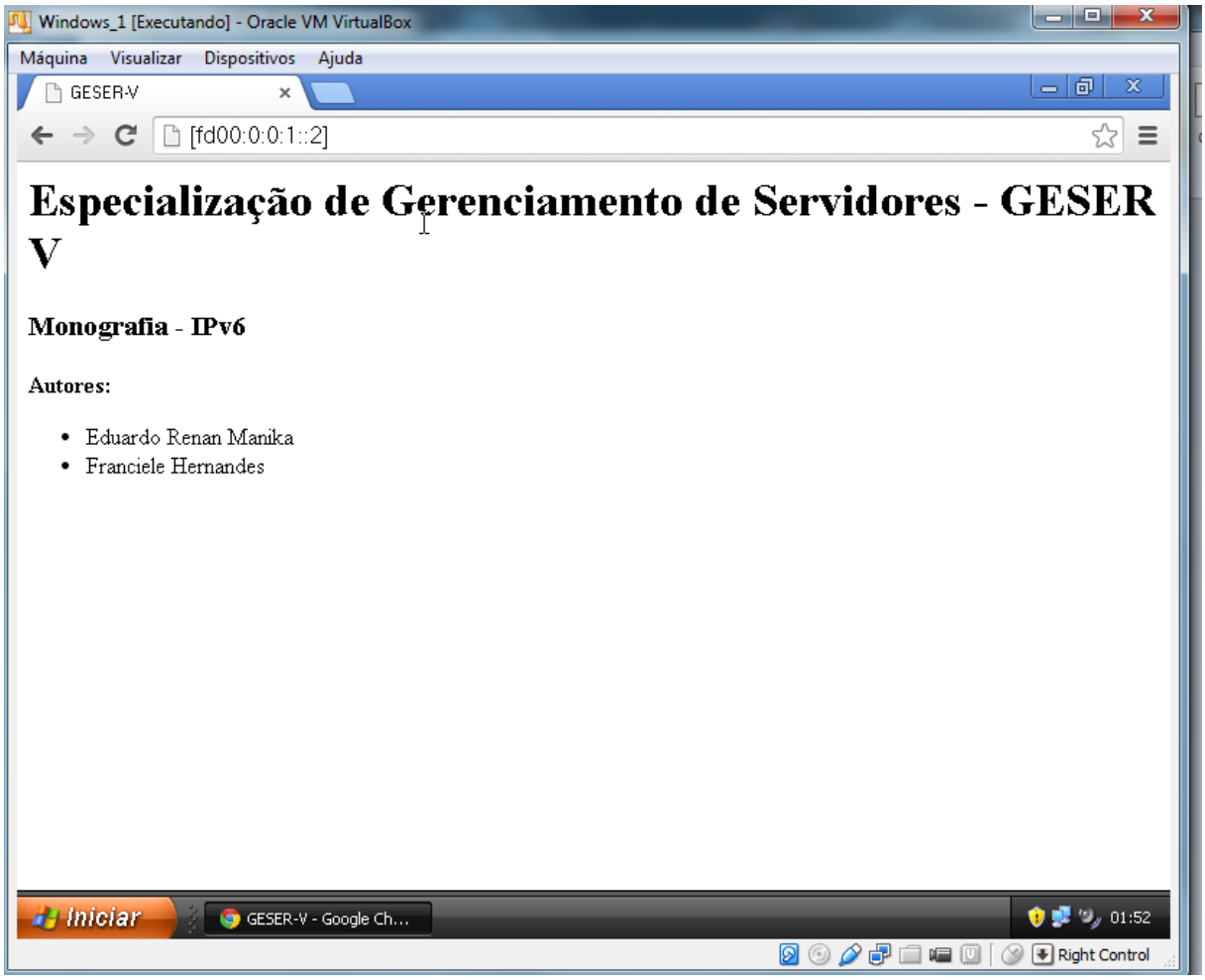


Figura 40 - Acesso a página web.
Fonte: Próprio autor

Na figura 42 apresenta uma requisição IPv6 utilizado endereço: “www6.efmcorp.com.br”.

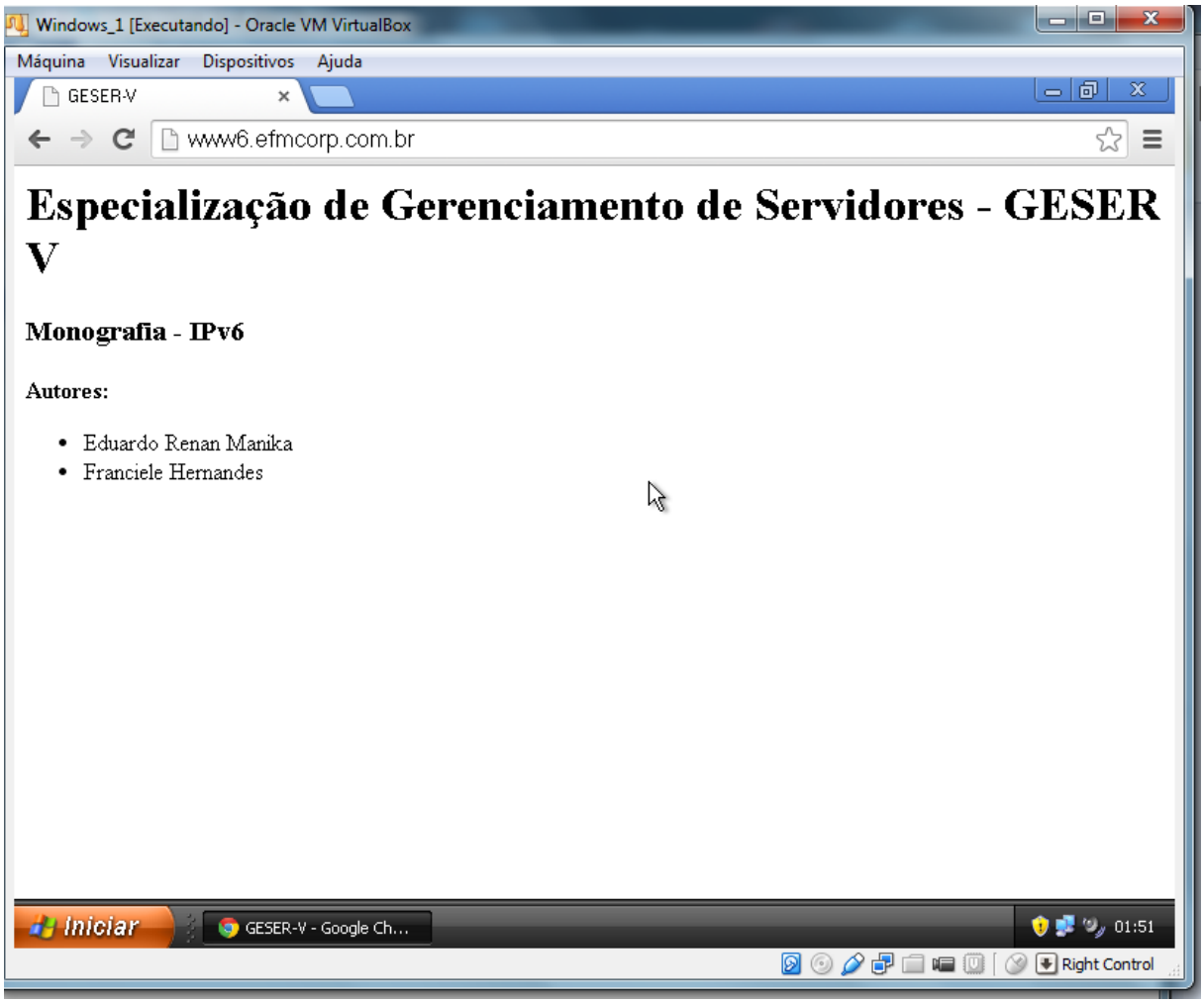


Figura 41 - Acesso a página web.
Fonte: Próprio autor

Após esses testes é possível verificarmos que o serviço Web pode operar simultaneamente em ambos protocolos de forma transparente para o usuário final.

4 CONCLUSÃO

Os resultados obtidos foram positivos, foi possível observar a importância do funcionamento do servidor DNS, implementado tanto para IPv4 quanto para IPv6 e a configuração dos equipamentos em pilha dupla permitiu isso. Observou-se que alguns comandos de configuração são diferentes para as versões do protocolo IP e que alguns equipamentos não apresentam suporte ao IPv6. Além disso, mostrou-se a importância do servidor DNS para a versão do protocolo IPv6, pois com o aumento do tamanho do endereço facilita qualquer configuração utilizada. Com isso os administradores de rede terão tempo para ajustar as configurações dos servidores para IPv6.

REFERÊNCIAS

ALECRIM, Emerson. **Endereço IP (Internet Protocol)**. Disponível em: <http://www.infowester.com/ip.php>. Acesso em 27/out/2014.

ANGÉLICA, Antonio dos Santos**. Introdução a Redes de Computadores.** Protocolo DNS. Disponível em <http://www.m8.com.br/antonio/redes/dns.htm>. Acesso em 27/out/2014.

BRITO, Samuel H. B. **IPv6.** O novo protocolo da internet. 1ª. ed, São Paulo: Novatec, 2013.

CENTRO DE ESTUDOS E PESQUISAS EM TECNOLOGIA DE REDES E OPERAÇÕES (CEPTRO.BR). **Introdução**. Equipe IPv6.br. Disponível em <http://ipv6.br/entenda/cabecalho/>. Acesso em 27/out/2014.

CENTRO DE ESTUDOS E PESQUISAS EM TECNOLOGIA DE REDES E OPERAÇÕES (CEPTRO.BR). **Cabeçalho**. Equipe IPv6.br. Disponível em <http://ipv6.br/entenda/cabecalho/>. Acesso em 27/out/2014.

CENTRO DE ESTUDOS E PESQUISAS EM TECNOLOGIA DE REDES E OPERAÇÕES (CEPTRO.BR). **Endereçamento**. Equipe IPv6.br. Disponível em <http://ipv6.br/entenda/enderecamento/>. Acesso em 27/out/2014.

CISCO. **IPv6 Routing: OSPFv3.** Disponível em <http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_ospf/configuration/15-sy/iro-15-sy-book/ip6-route-ospfv3.html#GUID-CCF5FD08-6199-4088-A202-DA7BCC61E830>. Acesso em 27/out/2014.

COMER, Douglas E. **Interligação de redes com TCP/IP**. Vol. 1 princípios, protocolo e arquitetura. 5ª. ed. Rio de Janeiro: Campus, 2006.

COMER, Douglas E. **Redes de computadores e internet.** Abrange transmissão de dados, ligações inter-redes, web e aplicações. 4ª. ed. São Paulo: Artmed, 2007.

COSTA, Daniel G. **DNS: Um guia para administradores de Redes.** Rio de Janeiro: Brasport,2006.

COUTO, Renato Botelho do. **O protocolo DNS.** Entendendo como funciona a

resolução de nomes de domínio. Disponível em <http://www.ibm.com/developerworks/br/local/opensource/dns_protocol/>. Acesso em 27/out/2014.

FILIPPETTI, Marco A. **CCNA 5.0.** Guia completo de estudo. Florianópolis: Visual Books, 2014.

GNS3. **What is GNS3**. Disponível em <http://www.gns3.net/>. Acesso em 13/jun/2014.

Kurose, James F. **Redes de computadores e a internet.** 5ª. ed. São Paulo: Pearson, 2010.

PILLOU,Jean-François. **NAT - Network Address Translation, porta e encaminhamento porta.** Disponível em: <http://pt.kioskea.net/contents/273-nat-network-address-translation-porta-e-encaminhamento-porta>. Acesso em 27/out/2014.

PISA, Pedro. **O que é IP.** Disponível em <http://www.techtudo.com.br/artigos/noticia/2012/05/o-que-e-ip.html/>. Acesso em 27/out/2014.

SUA PESQUISA. Historia **da internet: Acesso a Internet, provedores, Internet no Brasil, avanço da Informática, computadores, História da Internet, as redes sociais.** Disponível em <http://www.suapesquisa.com/internet/>. Acesso em 27/out/2014.

TELECO. **Redes IP I: IPv6**. Disponível em: <http://www.teleco.com.br/tutoriais/tutorialredeip1/pagina_3.asp>. Acesso em 27/out/2014.

APÊNDICE A – ARQUIVO DE CONFIGURAÇÃO DO ROTEADOR RT_A

```
RT_A#show running-config
Building configuration...

Current configuration : 2531 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RT_A
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
no ip icmp rate-limit unreachable
!
!
ip cef
no ip domain lookup
no ip dhcp use vrf connected
ip dhcp excluded-address 172.16.1.1 172.16.1.10
ip dhcp excluded-address 172.16.2.1
ip dhcp excluded-address 172.16.3.1
ip dhcp excluded-address 172.16.4.1
!
ip dhcp pool LAN_A
   network 172.16.1.0 255.255.255.0
   default-router 172.16.1.1
   dns-server 172.16.1.2 8.8.8.8
!
ip dhcp pool LAN_B
   network 172.16.2.0 255.255.255.0
   default-router 172.16.2.1
   dns-server 172.16.1.2 8.8.8.8
!
ip dhcp pool LAN_C
```

```
   network 172.16.3.0 255.255.255.0
   default-router 172.16.3.1
   dns-server 172.16.1.2 8.8.8.8
!
ip dhcp pool LAN_D
   network 172.16.4.0 255.255.255.0
   default-router 172.16.4.1
   dns-server 172.16.1.2 8.8.8.8
!
!
ipv6 unicast-routing
ipv6 dhcp pool LAN6_A
 prefix-delegation pool LAN6_A
 dns-server FD00:0:0:1::1
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
ip tcp synwait-time 5
!
!
!
!
!
interface FastEthernet0/0
 ip address 172.16.1.1 255.255.255.0
 ip helper-address 172.16.1.1
 duplex auto
 speed auto
 ipv6 address FD00:0:0:1::/64
```

```
 ipv6 nd managed-config-flag
 ipv6 dhcp server LAN6_A
 ipv6 ospf 1 area 0
!
interface FastEthernet0/1
 ip address 10.0.0.253 255.255.255.252
 duplex auto
 speed auto
 ipv6 address 2014::1/127
 ipv6 ospf 1 area 0
!
interface Serial1/0
 ip address 10.0.0.1 255.255.255.252
 ipv6 address 2000::/127
 ipv6 ospf 1 area 0
 serial restart-delay 0
 clock rate 64000
!
interface Serial1/1
 ip address 10.0.0.14 255.255.255.252
 ipv6 address 2003::1/127
 ipv6 ospf 1 area 0
 serial restart-delay 0
!
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
router ospf 1
 log-adjacency-changes
 network 10.0.0.0 0.0.0.3 area 0
 network 10.0.0.12 0.0.0.3 area 0
 network 10.0.0.252 0.0.0.3 area 0
 network 172.16.1.0 0.0.0.255 area 0
 default-information originate
!
ip route 0.0.0.0 0.0.0.0 10.0.0.254
```

```
!
no ip http server
no ip http secure-server
!
!
ipv6 local pool LAN6_A FD00:0:0:1::/64 64
ipv6 router ospf 1
 router-id 1.1.1.1
 log-adjacency-changes
!
!
!
!
!
control-plane
!
!
!
!
!
!
gatekeeper
 shutdown
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
 stopbits 1
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
 stopbits 1
line vty 0 4
 login
!
!
End
```

APÊNDICE B – ARQUIVO DE CONFIGURAÇÃO DO ROTEADOR RT_B

```
RT_B#show running-config
Building configuration...

Current configuration : 1667 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RT_B
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
no ip icmp rate-limit unreachable
!
!
ip cef
no ip domain lookup
!
!
ipv6 unicast-routing
ipv6 dhcp pool LAN6_B
prefix-delegation pool LAN6_B
dns-server FD00:0:0:1::1
!
!
!
!
!
!
!
!
!
!
!
!
!
```

```
!
!
!
!
!
ip tcp synwait-time 5
!
!
!
!
!
interface FastEthernet0/0
ip address 172.16.2.1 255.255.255.0
ip helper-address 10.0.0.1
duplex auto
speed auto
ipv6 address FD00:0:0:2::/64
ipv6 dhcp server LAN6_B
ipv6 ospf 1 area 0
!
interface FastEthernet0/1
no ip address
shutdown
duplex auto
speed auto
!
interface Serial1/0
ip address 10.0.0.5 255.255.255.252
ipv6 address 2001::/127
ipv6 ospf 1 area 0
serial restart-delay 0
clock rate 64000
!
interface Serial1/1
ip address 10.0.0.2 255.255.255.252
ipv6 address 2000::1/127
ipv6 ospf 1 area 0
serial restart-delay 0
!
interface Serial1/2
no ip address
shutdown
serial restart-delay 0
```

```
!
interface Serial1/3
no ip address
shutdown
serial restart-delay 0
!
router ospf 1
log-adjacency-changes
network 10.0.0.0 0.0.0.3 area 0
network 10.0.0.4 0.0.0.3 area 0
network 172.16.2.0 0.0.0.255 area 0
!
!
no ip http server
no ip http secure-server
!
!
ipv6 local pool LAN6_B FD00:0:0:2::/64 64
ipv6 router ospf 1
router-id 2.2.2.2
log-adjacency-changes
!
!
!
!
!
control-plane
!
!
!
!
!
!
gatekeeper
shutdown
!
!
line con 0
exec-timeout 0 0
privilege level 15
logging synchronous
stopbits 1
line aux 0
```

```
exec-timeout 0 0
privilege level 15
logging synchronous
stopbits 1
line vty 0 4
login
!
!
End
```

APÊNDICE C – ARQUIVO DE CONFIGURAÇÃO DO ROTEADOR RT_C

```
RT_C#show running-config
Building configuration...

Current configuration : 1696 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RT_C
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
no ip icmp rate-limit unreachable
!
!
ip cef
no ip domain lookup
!
!
ipv6 unicast-routing
ipv6 dhcp pool LAN6_C
 prefix-delegation pool LAN6_C
 dns-server FD00:0:0:1::1
!
!
!
!
!
!
!
!
!
!
!
!
!
```

```
!
!
!
!
!
ip tcp synwait-time 5
!
!
!
!
!
interface FastEthernet0/0
 ip address 172.16.3.1 255.255.255.0
 ip helper-address 10.0.0.1
 duplex auto
 speed auto
 ipv6 address FD00:0:0:3::/64
 ipv6 nd managed-config-flag
 ipv6 dhcp server LAN6_C
 ipv6 ospf 1 area 0
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 ip address 10.0.0.9 255.255.255.252
 ipv6 address 2002::/127
 ipv6 ospf 1 area 0
 serial restart-delay 0
 clock rate 64000
!
interface Serial1/1
 ip address 10.0.0.6 255.255.255.252
 ipv6 address 2001::1/127
 ipv6 ospf 1 area 0
 serial restart-delay 0
!
interface Serial1/2
 no ip address
 shutdown
```

```
 serial restart-delay 0
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
router ospf 1
 log-adjacency-changes
 network 10.0.0.4 0.0.0.3 area 0
 network 10.0.0.8 0.0.0.3 area 0
 network 172.16.3.0 0.0.0.255 area 0
!
!
no ip http server
no ip http secure-server
!
!
ipv6 local pool LAN6_C FD00:0:0:3::/64 64
ipv6 router ospf 1
 router-id 3.3.3.3
 log-adjacency-changes
!
!
!
!
!
control-plane
!
!
!
!
!
!
gatekeeper
 shutdown
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
 stopbits 1
```

```
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
 stopbits 1
line vty 0 4
 login
!
!
End
```

APÊNDICE D – ARQUIVO DE CONFIGURAÇÃO DO ROTEADOR RT_D

```
RT_D#show running-config
Building configuration...

Current configuration : 1670 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RT_D
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
no ip icmp rate-limit unreachable
!
!
ip cef
no ip domain lookup
!
!
ipv6 unicast-routing
ipv6 dhcp pool LAN6_D
 prefix-delegation pool LAN6_D
 dns-server FD00:0:0:1::1
!
!
!
!
!
!
!
!
!
!
!
!
!
```

```
!
!
!
!
!
ip tcp synwait-time 5
!
!
!
!
!
interface FastEthernet0/0
 ip address 172.16.4.1 255.255.255.0
 ip helper-address 10.0.0.1
 duplex auto
 speed auto
 ipv6 address FD00:0:0:4::/64
 ipv6 dhcp server LAN6_D
 ipv6 ospf 1 area 0
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 ip address 10.0.0.13 255.255.255.252
 ipv6 address 2003::/127
 ipv6 ospf 1 area 0
 serial restart-delay 0
 clock rate 64000
!
interface Serial1/1
 ip address 10.0.0.10 255.255.255.252
 ipv6 address 2002::1/127
 ipv6 ospf 1 area 0
 serial restart-delay 0
!
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
```

```
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
router ospf 1
 log-adjacency-changes
 network 10.0.0.8 0.0.0.3 area 0
 network 10.0.0.12 0.0.0.3 area 0
 network 172.16.4.0 0.0.0.255 area 0
!
!
no ip http server
no ip http secure-server
!
!
ipv6 local pool LAN6_D FD00:0:0:4::/64 64
ipv6 router ospf 1
 router-id 4.4.4.4
 log-adjacency-changes
!
!
!
!
!
control-plane
!
!
!
!
!
!
gatekeeper
 shutdown
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
 stopbits 1
line aux 0
```

```
 exec-timeout 0 0
 privilege level 15
 logging synchronous
 stopbits 1
line vty 0 4
 login
!
!
end
```