Instituto Federal de Goiás
Campus Formosa
Análise e Desenvolvimento de Sistemas
http://www.ifg.edu.br/formosa

IPV6: SUBFUNÇÕES E ANÁLISE COMPARATIVA DE DESEMPENHO DO DHCP EM REDES IPV4 E IPV6

BRUNO MONTEIRO BISPO

Trabalho de Conclusão de Curso

Formosa
2023

Instituto Federal de Goiás
Campus Formosa
Análise e Desenvolvimento de Sistemas
http://www.ifg.edu.br/formosa

IPV6: SUBFUNÇÕES E ANÁLISE COMPARATIVA DE DESEMPENHO DO DHCP EM REDES IPV4 E IPV6

Bruno Monteiro Bispo

Trabalho de Conclusão de Curso apresentado ao Departamento de Áreas Acadêmicas da Instituto Federal de Goiás campus Formosa, como requisito parcial para obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Me. Mário Teixeira Lemes

Formosa
2023

Bruno Monteiro Bispo
IPv6: subfunções e análise comparativa de desempenho do DHCP em Redes IPv4 e IPv6/ Bruno Monteiro Bispo. – Formosa, 2023-
98 p.; 30 cm.

Orientador Me. Mário Teixeira Lemes

Trabalho de Conclusão de Curso – Instituto Federal de Goiás, 2023.

1. Plano Semestral de Trabalho Docente 2. IFG 3. Desenvolvimento de software 4. Aplicação *Web* I. Orientador: Me. Mário Teixeira Lemes. II. Instituto Federal de Goiás. IV. Título: IPv6: subfunções e análise comparativa de desempenho do DHCP em Redes IPv4 e IPv6

CDU 02:141:005.7

TERMO DE AUTORIZAÇÃO PARA DISPONIBILIZAÇÃO NO REPOSITÓRIO DIGITAL DO IFG - ReDi IFG

Com base no disposto na Lei Federal nº 9.610/98, AUTORIZO o Instituto Federal de Educação, Ciência e Tecnologia de Goiás, a disponibilizar gratuitamente o documento no Repositório Digital (ReDi IFG), sem ressarcimento de direitos autorais, conforme permissão assinada abaixo, em formato digital para fins de leitura, download e impressão, a título de divulgação da produção técnico-científica no IFG.

Identificação da Produção Técnico-Científica

[ ] Tese	[ ] Artigo Científico
[ ] Dissertação	[ ] Capítulo de Livro
[ ] Monografia – Especialização	[ ] Livro
[ ] TCC - Graduação	[ ] Trabalho Apresentado em Evento
[ ] Produto Técnico e Educacional - Tipo: _____	

Nome Completo do Autor:
Matrícula:
Título do Trabalho:

Restrições de Acesso ao Documento

Documento confidencial: [ ] Não [ ] Sim, justifique: _____

_____

Informe a data que poderá ser disponibilizado no ReDi/IFG:__/__/__
O documento está sujeito a registro de patente? [ ] Sim [ ] Não
O documento pode vir a ser publicado como livro? [ ] Sim [ ] Não

DECLARAÇÃO DE DISTRIBUIÇÃO NÃO-EXCLUSIVA

O/A referido/a autor/a declara que:

i. o documento é seu trabalho original, detém os direitos autorais da produção técnico-científica e não infringe os direitos de qualquer outra pessoa ou entidade;
ii. obteve autorização de quaisquer materiais inclusos no documento do qual não detém os direitos de autor/a, para conceder ao Instituto Federal de Educação, Ciência e Tecnologia de Goiás os direitos requeridos e que este material cujos direitos autorais são de terceiros, estão claramente identificados e reconhecidos no texto ou conteúdo do documento entregue;
iii. cumpriu quaisquer obrigações exigidas por contrato ou acordo, caso o documento entregue seja baseado em trabalho financiado ou apoiado por outra instituição que não o Instituto Federal de Educação, Ciência e Tecnologia de Goiás.

_____, ____/____/____.
Local Data

_____
Assinatura do Autor e/ou Detentor dos Direitos Autorais

Dedico este trabalho a Deus, sem ele eu não teria capacidade para desenvolvê-lo. Aos meus pais, João e Maria, pela presença em minha vida.

Agradecimentos

A Deus, pela minha vida, e por me permitir ultrapassar todos os obstáculos encontrados ao longo da realização deste trabalho. Agradeço a todos os amigos que me ajudaram durante essa longa jornada de aprendizado, especialmente minha família, pois acreditaram no meu sonho e não mediram esforços para que eu pudesse concretizá-lo.

Aos professores, pelas correções e ensinamentos que me permitiram evoluir no processo de formação profissional. Em especial ao professor Me. Mário Teixeira Lemes, pelos conhecimentos passados, discussões guiadas, orientação e compreensão.

A Instituição, essencial no meu processo de formação profissional, pela possibilidade de capacitação pública, gratuita e de qualidade. A todos que participaram, direta ou indiretamente no desenvolvimento deste trabalho, enriquecendo o processo de aprendizado.

A todas as pessoas com quem convivi durante os últimos anos, pela troca de experiências que me permitem crescer não somente no campo pessoal, mas também no campo profissional.

Resumo

Com o rápido desenvolvimento da *Internet* com a potencialização da mobilidade e de coisas conectadas, *Internet Protocol Version 4* (IPv4) não garante a escalabilidade devido à escassez e esgotamento de endereços disponíveis. Apesar de ser um protocolo antigo, o *Internet Protocol Version 6* (IPv6) ainda não é amplamente adotado. O objetivo desta pesquisa é realizar o estudo, simulação e análise do servidor de endereçamentos IP, denominado *Dynamic Host Configuration Protoco*l (DHCP), em redes IPv4 e IPv6. Com uso dos emuladores core e eve-ng, criamos um cenário para avaliar o desempenho do DHCP e comparar, através de métricas definidas, a performance deste servidor em diferentes tipos de redes. A metodologia adotada neste trabalho baseia-se na pesquisa teórica-experimental e no uso de um modelo composto pelas fases de planejamento, implementação, verificação e ação. Os testes de desempenho se concentraram em métricas de número de solicitações, tempos de resposta e pacotes perdidos. Os resultados mostraram que o IPv6, com seus novos cabeçalhos e recursos exclusivos, superou em termos de número de solicitações entregues e em termos de tempo de resposta seu predecessor. O protocolo *Dynamic Host Configuration Protocol Version 6* (DHCPv6) destaca-se por sua superioridade no desempenho, sobretudo na capacidade de gerenciar uma quantidade significativa de endereços IP. Além disso, o DHCPv6 apresenta maior eficiência na distribuição de endereços IP e menor taxa de perda de pacotes.

Palavras-chave: DHCP, IPv4, IPv6, Emuladores

Abstract

With the rapid development of the Internet with the enhancement of mobility and connected things, IPv4 does not guarantee scalability due to the scarcity and exhaustion of available addresses. Despite being an old protocol, IPv6 is still not widely adopted. The aim of this research is to study, simulate, and analyze the IP addressing server, called DHCP, in IPv4 and IPv6 networks. Using the core and eve-ng emulators, we created a scenario to evaluate the performance of DHCP and compare, through defined metrics, the performance of this server in distinct types of networks. The methodology adopted in this work is based on theoretical-experimental research and the use of a model composed of planning, implementation, verification, and action phases. Performance tests focused on metrics of the number of requests, response times, and lost packets to show which version of DHCP proved to be more effective. The results showed that IPv6, with its new headers and unique features, outperformed its predecessor in terms of the number of requests delivered and response time. The DHCPv6 protocol stands out for its superiority in performance, particularly in its ability to manage a significant amount of IP addresses. Additionally, DHCPv6 has greater efficiency in distributing IP addresses and a lower packet loss rate.

Keywords: DHCP, IPV4, IPV6, Emulators

Lista de Figuras

Lista de Acrônimos

Sumário

1

Introdução

Antes mesmo da invenção de computadores a humanidade esforçava-se para possibilitar a comunicação de forma rápida e eficiente, sendo o telégrafo a primeira tecnologia a permitir esse feito, através de cabos transatlânticos para transporte da informação. O primeiro uso efetivo dessa tecnologia se deu em 1956, onde Canadá e Escócia foram ligados através de cabos e gerenciados por aparelhos que ocupavam uma sala inteira com pouca ou quase nenhuma interface [Heinisch, 2019].

Segundo Pinho [2018], o evento inicial para a criação da *Internet* foi o lançamento por parte da União das Repúblicas Socialistas Soviéticas (URSS), em 1957 do primeiro satélite de comunicação. Surge a partir desse ponto a necessidade de interligação de centros militares e de pesquisa, tornando real a exigência de um modelo de comunicação em rede. De acordo com Ferrari [2007], em 1969 iniciaram-se pesquisas experimentais, após uma série de testes de conexão executados com o apoio de diferentes estados americanos. O projeto *Arpanet* surge por volta de 1975 e passa a integrar a Agência de Comunicação e Defesa.

Inicialmente o acesso à *Advanced Research Projects Agency Network* (ARPANET) foi disponibilizado somente as empresas ligadas a o setor militar e as Universidades. Redes como a *Unix to Unix Copy Protocol* (UUCP) e a *Unix User Network* (USENET) foram utilizadas por setores acadêmicos e posteriormente, no final dos anos 90 por setores comerciais [Goethals et al., 2000].

A problemática se baseia na necessidade de conexão entre diferentes tipos de enlace de forma transparente. Para atingir esse objetivo a padronização com uso de protocolos objetivos e concisos é necessária. Neste contexto surge o *Transmission Control Protocol* (TCP)/*Internet Protocol* (IP), sucessor do modelo *Open Systems Interconnection* (OSI) criado pela *International Organization for Standardization* (ISO). Segundo Ribeiro [1998], o TCP/IP surge diante da carência de comunicação entre diferentes redes e organizações tendo como objetivo disponibilizar *links* de comunicação com alta largura de banda.

O *Internet Protocol Version 4* (IPv4) é a quarta revisão do IP, um protocolo amplamente utilizado na comunicação de dados sobre diferentes tipos de redes. De acordo com ipv6.br [2012], IPv4 foi projetado com 32 *bits* tendo capacidade de atribuição limitada a 4.294.967.296

endereços IP. Com o rápido crescimento da *Internet* especialmente com o advento de *Internet of Things* (IoT) o esgotamento de endereços IPv4 é inevitável. Em IoT, objetos inteligentes ou "coisas"*(things)* como veículos, eletrodomésticos e outros objetos que não fazem parte da computação, são conectados à *Internet* requerendo endereços IP únicos para efetiva comunicação.

Neste contexto a migração para uso do *Internet Protocol Version 6* (IPv6) garante crescimento e escalabilidade da *Internet*. O protocolo IPv6 possui 128 *bits* de tamanho, aumentando exponencialmente a quantidade de dispositivos que podem ser endereçados unicamente na rede. Com tantos dispositivos que podem se conectar à rede, a atribuição manual de endereços IP é inviabilizada. *Dynamic Host Configuration Protoco*l (DHCP) é usado para automatização do processo de atribuição de endereços IP, reduzindo a carga de trabalho na atribuição de endereços de rede [Bugallo et al., 2007].

1.1 Objetivos

1.1.1 Objetivo Geral

O objetivo deste Trabalho de Conclusão de Curso (TCC) é demonstrar as subfunções do protocolo IPv6 no âmbito do enlace focando em suas funções fundamentais, como a obtenção de informações de dispositivos vizinhos e realizar uma análise comparativa de desempenho do serviço DHCP em redes IPv4 e IPv6 em diferentes condições de uso, através da coleta das métricas de quantidade de solicitações, perda de pacotes e tempo utilizado para finalização dos processos de comunicação cliente-servidor.

1.1.2 Objetivos Específicos

1. Ampliar conhecimento referente ao protocolo IP e ao serviço DHCP em redes IPv4 e IPv6.

2. Demonstrar as subfunções do protocolo IPv6, atraves das mensagens *Neighbor Solicitation* (NS), *Neighbor Advertisement* (NA), *Router Solicitation* (RS) e *Router Advertisement* (RA)

3. Apresentar a operação do *Dynamic Host Configuration Protocol Version 6* (DHCPv6) em redes *stateful* e *stateless*.

4. Comparar desempenho do serviço DHCP nas versões v4 e v6 através de simulação e gerar gráficos que possibilitem visualização comparativa entre os servidores.

1.2 Trabalhos Relacionados

Silveira [2012] avalia o uso de pilha dupla e as técnicas de tradução entre redes IPv4 e IPv6. Para encontrar o protocolo de transição mais adequado para o protocolo de rede atual, a

pesquisa explora tecnologias como NAT64/DNS64 e protocolos de pilha dupla que suportam navegação na *Web*.

Pedrozo [2014] estuda técnicas de tunelamento, propondo o uso de uma rede IPv6 com uso de tunelamento que permite o tráfego IPv6 em uma rede IPv4. Os autores utilizam diferentes técnicas de tunelamento, tais como *Tunnel Broker*, *Tunnel 6over4* e *Tunnel GRE*. No ambiente de simulação proposto, *hosts* atuam como clientes em uma rede IPv6 onde é implementado os serviços *Domain Name System* (DNS) e DHCP.

Araujo et al. [2014] demonstra conceitos básicos relacionados ao IPv6 em uma rede local usando o protocolo DHCP. A simulação realizada pelo autor é suportada pelo *software Cisco Packet Tracer* que define uma topologia com 30 dispositivos conectados a dois *switches* e um roteador com interfaces configuradas para suportar DHCPv6. O autor discute a importância da simulação e que a correta implementação e resolução de problemas de redes simuladas devem ser muito semelhantes às redes reais.

Santos [2016] demonstra e implementa o uso do protocolo IPv6 em provedores, utilizando a tecnologia de pilha dupla para garantir operação simultânea das redes IPv4 e IPv6. Como requisitos de implementação, o autor faz uso do *Border Gateway Protocol* (BGP) para executar dinamicamente o roteamento de rede, planejamento de endereçamento, servidores DNS para suportar solicitações de endereços IPv6 e autenticação de cliente para atribuição de endereço via protocolo *Point-to-Point Protocol over Ethernet* (PPPoE), todos implementados via *Border Gateway Protocol Version 4* (BGPv4) e *Open Shortest Path First Version 3* (OSPFv3) para dar suporte a implementação do IPv6.

1.3 Descrição dos Capítulos

Esse TCC está dividido em 5 capítulos. Nesse Capítulo foi apresentado uma contextualização sobre a *Internet*, endereçamento IPv4 e IPv6 e atribuição de endereçamento com DHCP. Apresentamos também trabalhos relacionados, o objetivo geral e os objetivos específicos que definem a finalidade e a delimitação deste trabalho.

No Capítulo 2 são abordados conceitos teóricos do modelo OSI, TCP/IP, bem como de serviços que circundam a utilização do DHCP. Já no Capítulo 3 destacamos a metodologia e ferramentas utilizadas para realização da simulação dos cenários propostos.

No Capítulo 4 mostramos diferentes cenários para compreender as subfunções do IPv6 bem como sua interação com o serviço DHCP. Também definimos cenários para avaliar o desempenho deste servidor considerando algumas métricas importantes, tais como quantidade de requisições, tempos de resposta e quantidade de pacotes perdidos.

Finalmente no Capítulo 5 apresentamos conclusões obtidas ao longo desenvolvimento deste trabalho, contribuições científicas, dificuldades encontradas e sugestões para possíveis trabalhos futuros.

2

Referencial Teórico

2.1 Modelo de Referência OSI/ISO

O modelo *Open Systems Interconnection* (OSI) é um modelo de rede que define um conjunto de protocolos de comunicação em camadas. Desenvolvido pela *International Organization for Standardization* (ISO) e amplamente utilizado como uma estrutura de referência para a criação de protocolos de rede. O modelo OSI/ISO ilustrado na figura divide a comunicação em sete camadas diferentes cada uma com suas próprias funções e responsabilidades, para que cada camada possa ser desenvolvida e implementada independentemente das outras [Kurose, 2017].

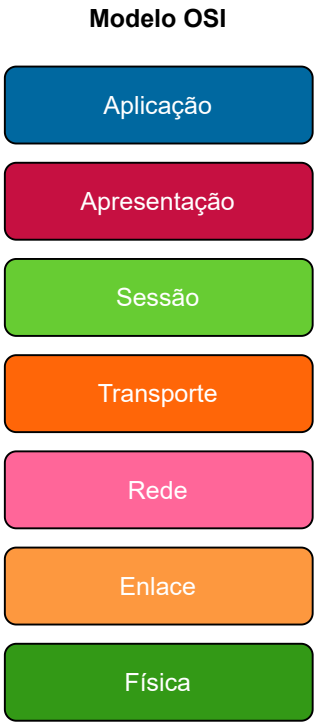

Figura 2.1: Modelo OSI

Segundo Forouzan B.A. [2007] as funcionalidades de cada camada mostrada na Figura 2.1 são destacadas a seguir:

- **Aplicação:** Possibilita comunicação direta entre usuários finais e aplicações. Nesta camada são definidos os protocolos de aplicação, como *Hypertext Transfer Protocol* (HTTP) para transferência de páginas web e *File Transfer Protocol* (FTP) para transferência de arquivos.

- **Apresentação:** Tem a função de representar os dados em um formato que o aplicativo possa entender. Nesta camada, protocolos de compactação, criptografia e formatação de dados como *Joint Photographic Experts Group* (JPEG) e Secure Sockets Layer/-Transport Layer Security (SSL/TLS) são definidos para criptografia de imagens e segurança de transações online.

- **Sessão:** Estabelece, gerencia e encerra sessões entre aplicativos em diferentes dispositivos. Nesta camada é definido um mecanismo de controle de fluxo que gerencia a troca de dados entre as aplicações. o protocolo mais comum dessa camada é o Network Basic Input/Output System (NetBIOS) que gerenciam sessões entre aplicativos em uma rede.

- **Transporte:** Tem como função passar dados entre processos em diferentes dispositivos. Nessa camada, os dados são divididos em segmentos e endereçados para que possam ser transmitidos pelo meio físico. Os protocolos mais comuns nessa camada são o *Transmission Control Protocol* (TCP) e o *User Datagram Protoco*l (UDP), que definem o processo de estabelecimento da conexão, controle de fluxo e confirmação de recebimento de pacotes.

- **Rede:** Encarregado por encaminhar pacotes de dados entre diferentes redes. Nesta camada, os pacotes de dados são endereçados e roteados por diferentes redes utilizando o protocolo IP. A camada de rede define as regras e procedimentos para roteamento e seleção do caminho mais eficiente para transmissão de pacotes de dados.

- **Enlace:** Responsável pelo controle de erros e transmissão confiável de dados entre dispositivos na mesma rede. Nessa camada, os dados são divididos em quadros e endereçados para que possam ser transmitidos pelo meio físico.

- **Física:** Transmite sinais elétricos ou ópticos via transporte físico, como cabos de cobre, fibras ópticas ou ondas de rádio. Define as características elétricas, mecânicas e funcionais dos meios físicos de comunicação, como taxa de transmissão, codificação e modulação do sinal.

No entanto, apesar de sua importância histórica o modelo OSI não é o mais utilizado , sendo substituído pelo modelo TCP/IP por ser a concretização dos conceitos abstratos apresentados.

2.2 Modelo de Referência TCP/IP

O modelo TCP/IP é uma estrutura de protocolo de rede amplamente utilizada em todo o mundo que permite a comunicação entre dispositivos em uma rede de computadores. Pode ser implementado em diferentes tipos de redes, *Local Area Network* (LAN), *Metropolitan Area Network* (MAN) E *Wide Area Network* (WAM). O modelo TCP/IP é um padrão aberto, o que significa que qualquer pessoa ou organização pode usá-lo sem licenciamento ou pagamento de royalties [Fall K.R, 2011]. A Figura 2.2 mostra a arquitetura TCP/IP composta por 4 camadas, acesso a rede, *Internet*, Transporte e Aplicação. As funcionalidades das camadas da arquitetura TCP/IP são semelhantes às definidas no modelo de referência OSI/ISO.

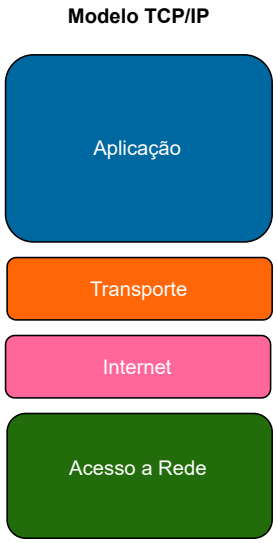

Figura 2.2: Modelo TCP/IP

2.3 Protocolo IPv4

O *Internet Protocol Version 4* (IPv4) é usado para troca de dados entre dispositivos em uma rede de computadores suportando 4.294.967.296 endereços de rede. Por ser um protocolo sem conexão, no qual não há um procedimento formal de estabelecimento de conexão antes do envio dos pacotes e não fornece uma garantia de entrega, seus pacotes são tratados de maneira autônoma. Essa abordagem simplifica o processo de roteamento e confere ao IPv4 uma maior eficiência em sua operação [Forouzan, 2003].

Diante das limitações apresentadas pelo IPv4 e do uso de recursos paliativos discutido na seção 2.4, surge a necessidade de adoção de um novo protocolo chamado, IPv6 possuindo como atrativos o numero maior de endereços, segurança melhorada e melhor qualidade de serviço que é detalhado na seção 2.7. A seguir, mostramos como é a notação de um endereço IPv4.

2.3.1 Notação de Endereço

A notação de endereço do Protocolo IPv4 é uma forma de representação de endereços IP em formato decimal pontuado que é usado para identificar dispositivos em uma rede. Ilustrado na Figura 2.3 o endereço IPv4 é composto por um número de 32 *bits*, que é dividido em quatro octetos de oito *bits* cada, representados em formato decimal variando de 0 a 255 [Perlman, 2000].

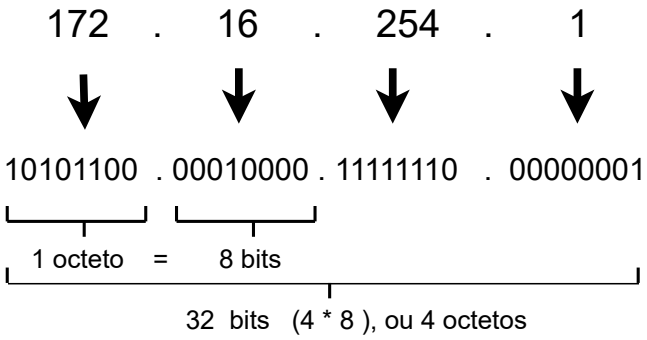

Figura 2.3: Endereçamento IPv4

O endereço IPv4 é dividido em duas partes principais a parte da rede e a parte do *host*. A parte de rede é usada para identificar a rede em que o dispositivo está, e a parte de *host* é usada para identificar dispositivos individuais na rede. A máscara de sub-rede ilustrado na Figura 2.4 é usada para determinar as partes de rede e *host* de um endereço IPv4. Uma máscara de sub-rede é um número de 32 *bits* que identifica a parte de rede de um endereço IPv4. A parte de *host* do endereço IPv4 é determinada pelo que sobra após a aplicação da máscara de sub-rede [Perlman, 2000].

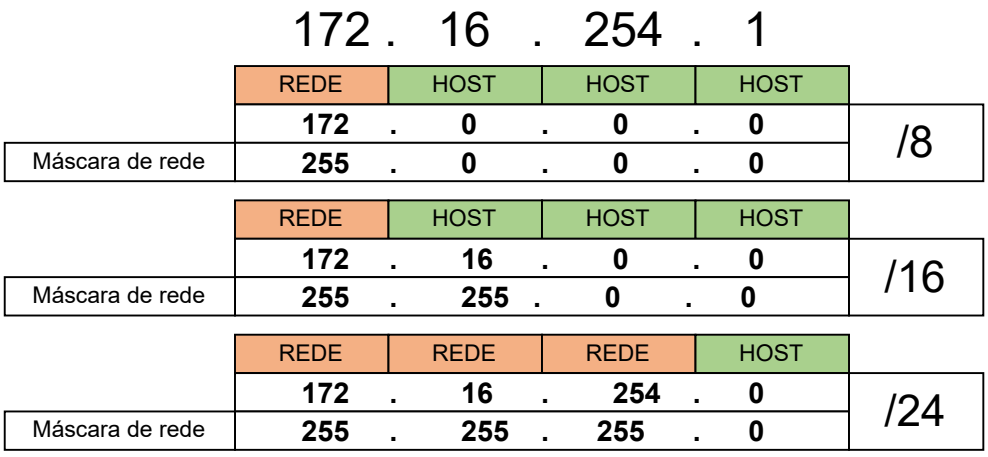

Figura 2.4: Máscara de Rede IPv4

2.3.2 Cabeçalho Básico

O cabeçalho de pacote IPv4 é uma parte essencial do protocolo. Fornecendo informações importantes para roteadores e outros dispositivos que processam pacotes de rede. O cabeçalho do pacote IPv4 tem um comprimento fixo, sendo anexado ao pacote para permitir a transmissão de informações pela *Internet*, embora possa ser estendido através do cabeçalho de extensão. Os campos de cabeçalho são organizados em uma estrutura de 14 campos de comprimento variável [Comer, 2006].

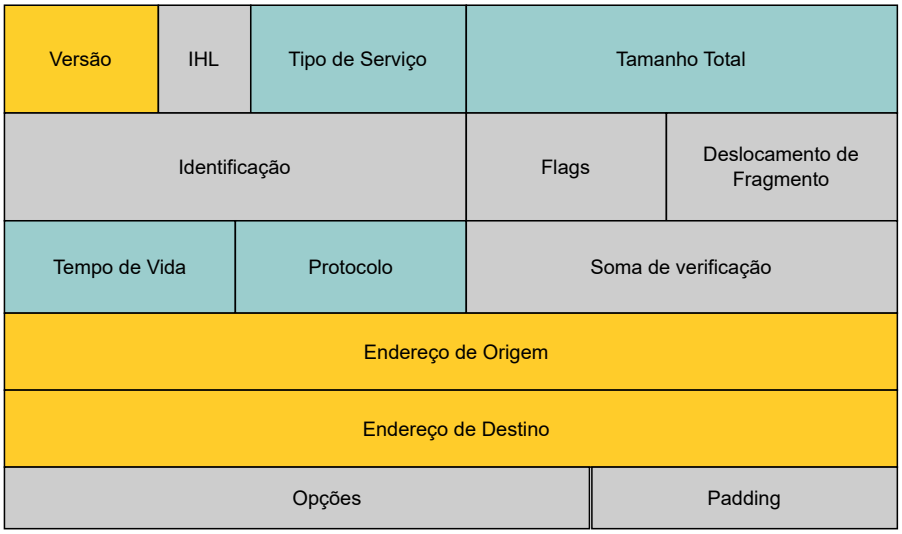

Figura 2.5: Cabeçalho IPv4

Segundo Comer [2006] as função de cada bloco mostrada na Figura 2.5 são destacadas a seguir :

- **Versão:** Indica a versão do protocolo IP sendo que o valor 4 corresponde ao IPv4. Possuindo o tamanho de 4 *bits*.
- **Tamanho do cabeçalho (IHL):** Este campo indica o tamanho do cabeçalho do protocolo IPv4 . O valor mínimo deste campo é 5. O tamanho total deste campo é de 4 *bits*.
- **Tipo de serviço:** Este campo é usado para definir o tipo de serviço que deve ser dado ao pacote, como prioridade de transmissão ou requisitos de largura de banda. O tamanho desse campo é de 8 *bits*.
- **Tamanho Total:** Este campo indica o tamanho total do pacote incluindo o cabeçalho e os dados. Este campo possui tamanho de 16 *bits*.
- **Identificação:** Este campo é usado para identificar o pacote. É geralmente incrementado por um sempre que um novo pacote é enviado. Possui tamanho de 16 *bits*.

- ***Flags*:** Este campo é usado para controlar a fragmentação do pacote em redes com limitações de tamanho de pacote. Possui tamanho de 3 *bits*.
- **Deslocamento do fragmento:** Este campo é usado para identificar a posição do fragmento dentro do pacote original. Possui tamanho de 13 *bits*.
- **Tempo de vida:** Valor atribuído assim que o pacote é gerado sendo decrementado a cada salto. É usado para controlar a quantidade de tempo que um pacote pode permanecer na rede antes de ser descartado. Possuindo 8 *bits* de tamanho.
- **Protocolo:** Indica o tipo de dado a qual o pacote está associado. Possibilitando que a camada de rede repasse os dados ao protocolo de transporte. Este campo possui 8 *bits* de tamanho.
- **Soma de verificação:** Este campo é usado para verificar a integridade do cabeçalho do pacote. Este campo possui 16 *bits*.
- **Endereço de origem:** Este campo é usado para indicar o endereço IP do dispositivo que está enviando o pacote. Tamanho 32 *bits*.
- **Endereço de destino:** Este campo é usado para indicar o endereço IP do dispositivo que deve receber o pacote. Tamanho de 32 *bits*.
- **Opções:** Este campo é opcional e pode ser usado para incluir informações adicionais no cabeçalho do pacote IPv4, como informações de roteamento ou data e hora. Possui tamanho variável.
- ***Padding*:** Este campo é usado para preencher o cabeçalho IPv4 com *bytes* adicionais se necessário, para garantir que o comprimento total do cabeçalho seja um múltiplo de 32 *bits*.

2.3.3 Funções Adicionais

O campo de opções é um campo de comprimento variável localizado entre o cabeçalho IPv4 e a carga útil de dados. Seu propósito é transportar informações adicionais e parâmetros de controle que não são essenciais para o roteamento básico e entrega dos pacotes IP.

As funções do campo de opções incluem descoberta de *Maximum Transmission Unit* (MTU): permitindo determinar o tamanho máximo de pacotes que podem ser transmitidos sem fragmentação. *Timestamp*: esse campo suporta a inclusão de horários nos pacotes IP, auxiliando em diagnósticos de rede. Registro de Rota: registra o caminho percorrido por um pacote IP, útil para análise de rotas e identificação de gargalos. Segurança e Autenticação: inclui opções para garantir a segurança dos pacotes IP e fornecer autenticação, permitindo a implementação de protocolos como IPsec para proteção dos dados transmitidos [Loshin, 2004].

2.4 Tradução de endereços de rede NAT

Segundo Kjeld Borch Egevang [2001] o *Network Address Translation* (NAT) é uma técnica amplamente utilizada em redes de computadores para permitir que múltiplos dispositivos compartilhem um único endereço IP público. O NAT funciona traduzindo os endereços IP privados dos dispositivos locais em um endereço IP público antes de encaminhar os pacotes para a Internet ajudando a mitigar a escassez de endereços IPv4 , já que muitos dispositivos podem se conectar à Internet usando apenas um endereço IP público.

Embora esse recurso estenda a vida do IPv4, o NAT também apresenta algumas limitações. Uma das principais limitações é a dificuldade de estabelecer conexões de entrada para dispositivos internos, uma vez que o NAT permita apenas o encaminhamento de pacotes iniciados a partir da rede interna, causando problemas em cenários onde é necessário que dispositivos externos iniciem a conexão. Introduzindo problemas de desempenho e complexidade em redes de grande porte, com um grande número de conexões simultâneas [Kjeld Borch Egevang, 2001].

2.5 ICMPv4

O *Internet Control Message Protocol Version 4* (ICMPv4) é um protocolo auxiliar do protocolo IPv4 sendo essencial permitindo que os dispositivos de rede comuniquem informações de erro e status entre si, ajudando a manter a conectividade entre os dispositivos [Hunt, 2002].

O ICMPv4 é um protocolo sem conexão o que significa que nenhuma sessão é estabelecida antes que os dados sejam transferidos, operando diretamente na camada de rede e sendo integrado ao cabeçalho do protocolo IPv4 [Postel, 1981].

2.5.1 Mensagens ICMPv4

Independentemente do tipo de mensagem gerada pelo *Internet Control Message Protocol* (ICMP) os três primeiros campos de cabeçalho são padrão. Sendo eles: Tipo (identificação da mensagem), Código (informações sobre o tipo de mensagem) e Soma de verificação (relacionado à integridade da mensagem) a Figura 2.6 mostra o cabeçalho ICMP para o IPv4.

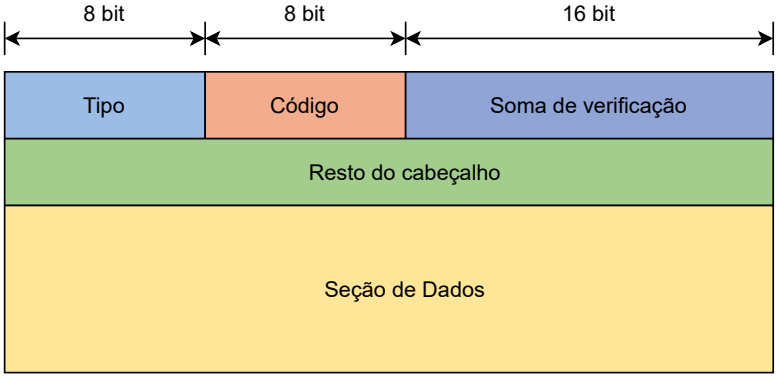

Figura 2.6: Cabeçalho ICMPv4

Segundo Fernando Gont [2013], o ICMPv4 é composto por vários tipos de mensagens. As mensagens mais usadas são:

- ***Echo Request e Echo Reply (ping)*:** Usado para testar a conectividade entre dois dispositivos em uma rede.
- ***Destination Unreachable*:** Enviado por um roteador para indicar que um pacote não pode ser entregue ao destino final.
- ***Time Exceeded*:** Enviado por um roteador para indicar que um pacote foi descartado devido a um tempo de vida expirado.
- ***Redirect*:** Enviado por um roteador para informar a um host que deve usar um roteador diferente para alcançar um determinado destino.
- ***Traceroute*:** Uma ferramenta de diagnóstico que usa o ICMPv4 para rastrear o caminho de um pacote através da rede.

2.6 DHCPv4

O *Dynamic Host Configuration Protocol Version 4* (DHCPv4) é um protocolo de rede projetado para configurar automaticamente dispositivos em uma rede tornando o processo mais eficiente e confiável. Surgiu como uma evolução do *Bootstrap Protocol* (BOOTP) que permitia que endereços IP estáticos fossem atribuídos a dispositivos na rede local [Droms, 1997].

Sendo padronizado em 1997 como uma forma de aumentar a escalabilidade e a flexibilidade da atribuição de endereços IP para redes locais. Antes do DHCPv4 a atribuição de endereços IP era feita manualmente, o que era ineficiente e sujeito a erros especialmente em grandes redes [Bugallo et al., 2007].

2.6.1 Comunicação DHCPv4

A comunicação entre o cliente e o servidor DHCP ocorre através do protocolo UDP que é um protocolo sem conexão e sem garantias de entrega. O UDP é usado para transportar pacotes DHCP entre clientes e servidores, tornando a comunicação mais rápida e eficiente se comparado ao uso de um protocolo orientado a conexão como o TCP, as portas UDP 67 e 68 são utilizadas no processo de comunicação [Droms, 1997].

Segundo Alcott [2001] o DHCPv4 usa um modelo cliente-servidor em que o servidor DHCP é responsável por atribuir configurações de rede aos dispositivos clientes, através da troca das seguintes mensagens:

- ***DHCP DISCOVER:*** Esta mensagem é enviada pelo cliente para localizar um servidor DHCP na rede. O cliente ainda não possui um endereço IP e solicita ao servidor DHCP que lhe atribua um endereço.

- ***DHCP OFFER:*** Quando o servidor DHCP recebe uma mensagem DHCP *Discover*, o mesmo responde com uma mensagem DHCP *offer* para fornecer ao cliente parametros de rede.

- ***DHCP REQUEST:*** O cliente pode receber várias mensagens DHCP *offer* de diferentes servidores DHCP na rede. O cliente escolhe um dos endereços oferecidos e envia uma mensagem DHCP *request* para solicitar ao servidor selecionado a confirmação da oferta.

- ***DHCP ACK:*** O servidor DHCP responde à mensagem DHCP *request* com uma mensagem DHCP ACK para confirmar que o endereço IP solicitado está reservado para o cliente. Os clientes agora podem se comunicar na rede usando o endereços IP adquirido.

2.6.2 Tipos de Endereço

Uma das características do IPv4 é a forma como os endereços são usados para enviar e receber informações. Segundo Forouzan B.A [2009] três tipos principais de endereços são usados no IPv4: *unicast, multicast e broadcast*.

- ***Unicast***: Um endereço *unicast* é um dos tipos de endereço existentes no protocolo IPv4. Sendo usado para identificar um único dispositivo em uma rede, permitindo que os pacotes sejam enviados direta e exclusivamente para esse dispositivo. Os endereços *Unicast* são normalmente usados para comunicações ponto a ponto, onde os pacotes são enviados diretamente de um dispositivo para outro.

- ***Multicast***: Um endereço *multicast* é um endereço atribuído a um grupo de dispositivos em uma rede, permitindo que informações sejam enviadas para vários dispositivos ao mesmo tempo. Quando uma mensagem é enviada para um endereço multicast, a rede identifica todos os dispositivos que pertencem ao grupo e encaminha a mensagem. Isso permite que a mesma informação seja transmitida para vários dispositivos simultaneamente economizando largura de banda e melhorando a eficiência da rede.

- ***Broadcast*** : Um endereço de *broadcast* é um endereço usado para enviar mensagens para todos os dispositivos na rede. Quando uma mensagem é enviada para um endereço de *broadcast* a rede encaminha a mensagem para todos os dispositivos do enlace, independentemente de pertencerem a um grupo ou não. Isso é útil para enviar informações importantes a todos os dispositivos da rede, mas pode causar congestionamento e reduzir a eficiência da rede.

2.7 Protocolo IPv6

O *Internet Protocol Version 6* (IPv6) é a versão mais recente do protocolo da Internet sendo idealizada para resolver os problemas da sua versão anterior. Pelo fato do IPv6 usar endereços de 128 *bits* o mesmo possui aproximadamente 340 undecilhões de endereços exclusivos [Bob Hinden, 1998].

Como vantagem do IPv6 foi adicionado segurança aprimorada, suporte nativo para criptografia, cabeçalhos de pacotes simplificados e opções de extensão incluídas. Permitindo que novos recursos sejam adicionados sem afetar os principais protocolos [Bob Hinden, 1998].

2.7.1 Notação de Endereço

Conforme Dr. Steve E. Deering [2006] a notação de endereço usada no IPv6 é diferente da notação de endereço usada no IPv4. O IPv6 usa endereços de 128 bits divididos em oito blocos de 16 *bits* cada. Cada bloco é representado por quatro números hexadecimais separados por dois pontos. Um endereço IPv6 típico pode ser escrito conforme a figura 2.7 . Observe que os zeros à esquerda em cada bloco podem ser omitidos mas cada bloco deve ter pelo menos um dígito hexadecimal. Uma sequência contínua de blocos de zeros pode ser substituída por um único conjunto de dois pontos conhecido como compressão de zeros.

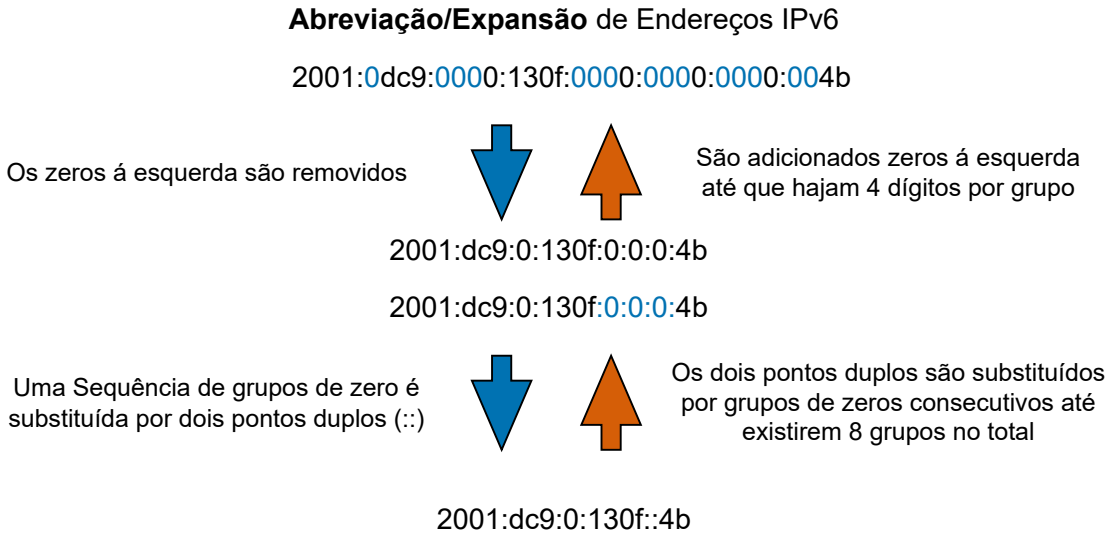

Figura 2.7: Endereçamento IPv6

Na notação de endereço IPv6 um endereço pode ser dividido em três partes: um prefixo, um identificador de sub-rede e um identificador de interface. O prefixo é uma parte fixa do endereço que identifica a rede. O identificador de sub-rede identifica uma sub-rede dentro da rede e o identificador de interface identifica dispositivos específicos dentro da sub-rede [Dr. Steve E. Deering, 2006].

O prefixo é representado por um número fixo de bits no endereço. Conforme a figura 2.8

um prefixo de 48 bits indica que os primeiros 48 bits do endereço são fixos e são usados para identificar a rede. O identificador de sub-rede é formado pelos bits que ficam entre o prefixo e o identificador de interface, enquanto o identificador de interface é composto pelos demais bits.

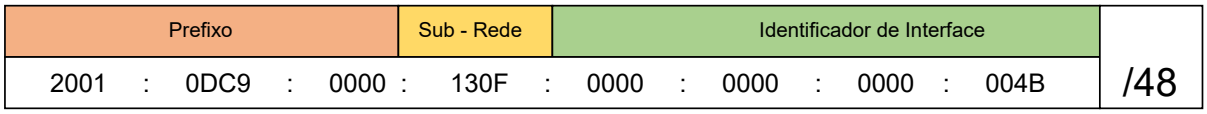

Figura 2.8: Máscara de Rede IPv6

2.7.2 Cabeçalho Básico

O cabeçalho do protocolo IPv6 é uma parte essencial do processo de comunicação de dados na Internet. Comparado ao seu antecessor seção 2.3.2 a arquitetura do cabeçalho IPv6 é mais eficiente. O tamanho do cabeçalho IPv6 é fixo eliminando campos desnecessários e acelerando o processamento de pacotes. Sendo dividido em oito campos cada um com uma finalidade específica para entregar o pacote. Pode ser estendido com mais opções por meio de cabeçalhos de extensão [Dr. Steve E. Deering, 2017].

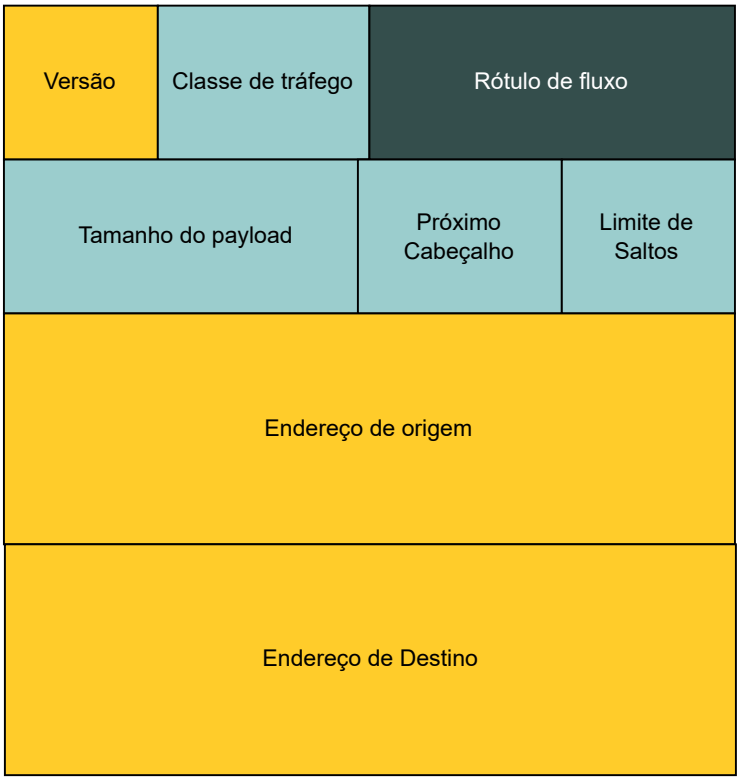

Figura 2.9: Cabeçalho IPv6

Segundo Comer [2016] as função de cada bloco mostrada na Figura 2.9 são destacadas a seguir:

- **Versão (*Version*):** Identifica a versão do protocolo. Para o IPv6 o valor deste campo é sempre 6. Seu tamanho é de 4 *bytes*.

- **Classe do tráfego (*Traffic Class*):** Usado para classificar o tráfego em diferentes níveis de prioridade. O valor deste campo é usado pelos roteadores para determinar a prioridade de entrega de pacotes em situações de congestionamento da rede. Tem um tamanho de 8 *bytes*.

- **Rótulo do fluxo (*Flow Label*):** Usado para rotular os pacotes que são do mesmo fluxo. Util para melhorar o *Quality of Service* (Qos) de aplicações em tempo real. Seu tamanho é de 20 *bytes*.

- **Tamanho do *payload* (*Payload length*):** O campo de tamanho do payload indica o tamanho dos dados úteis em *bytes* incluindo cabeçalhos de extensão, se houver. A carga útil é a parte do pacote que contém as informações que devem ser entregues ao destinatário. Seu tamanho é de 16 *bytes*.

- **Próximo cabeçalho (*Next Header*):** O campo próximo cabeçalho indica o tipo de protocolo que segue o cabeçalho IPv6. Isso permite que o receptor reconheça o tipo de dados por trás do cabeçalho IPv6 e tome as medidas necessárias para processar os dados corretamente. Tem tamanho de 8 *bytes*.

- **Limite de Salto (*Hop Limit*):** O campo de limite de saltos indica o número máximo de saltos permitidos antes que o pacote seja descartado. Isso ajuda a evitar a sobrecarga da rede devido à perda de pacotes ou loops infinitos. Seu tamanho é de 8 *bytes*.

- **Endereço de origem (*Source Address*):** O campo de endereço de origem contém o endereço IP do remetente do pacote. Possui tamanho de 128 *bytes*.

- **Endereço de Destino (*Destination Address*):** O campo de endereço de destino contém o endereço IP do destinatário do pacote. Possui tamanho de 128 *bytes*.

2.7.3 Cabeçalho de Extensão

O cabeçalho de extensão IPv6 permitem que informações adicionais sejam anexadas aos pacotes IPv6 fornecendo mais flexibilidade e funcionalidade ao protocolo. Cada cabeçalho de extensão tem sua própria especificação e propósito e usá-los pode ser útil em muitas situações. No IPv6 a utilização excessiviva de cabeçalhos de extensão pode aumentar o tamanho do pacote e degradar o desempenho da rede diminuindo a interroperabilidade [McFarland et al., 2011].

De acordo com Dr. Steve E. Deering [2017] o uso completo do IPv6 inclui a implementação dos cabeçalhos de extensão sendo os principais deles:

- ***Hop-by-hop options*:** Este cabeçalho permite que o roteador insira as opções de processamento que devem ser aplicadas ao pacote durante seu caminho até o destino

final. Sendo tratado por todos os roteadores no caminho e deve aparecer exatamente uma vez por pacote.

- ***Destination options*:** Este cabeçalho é semelhante ao Cabeçalho de Opções *Hop-by-Hop*, mas as opções contidas são específicas para o destino final. Podendo aparecer várias vezes em um pacote e só pode ser processado por nós finais.

- ***Routing*:** Este cabeçalho é usado para especificar a rota que o pacote deve seguir para chegar ao seu destino final. Pode ser usado para implementar rotas alternativas ou para otimizar a rota que um pacote segue.

- ***Fragmentation*:** Este cabeçalho é usado para quebrar grandes pacotes em pedaços menores que podem ser retransmitidos individualmente. Contém informações sobre como o pacote original foi dividido e como os fragmentos podem ser remontados no destino final.

- ***Authentication*:** Este cabeçalho é usado para garantir a autenticidade e integridade dos pacotes IPv6. Adicionando informações de autenticação e integridade ao pacote, permitindo que o destinatário verifique se o pacote foi realmente enviado pelo remetente correto e não foi alterado durante o trânsito.

- ***Encapsulantion Security Payload*:** Esse cabeçalho é semelhante ao *Authentication*, mas oferece suporte a recursos de privacidade como criptografia. Pode ser usado para proteger o conteúdo do pacote contra terceiros.

2.8 Tradução de endereços de rede NAT64

O *Network Address Translation* (NAT64) é um mecanismo de tradução de endereços de rede que desempenha um papel crucial na interconexão de redes IPv6 e IPv4. Ele permite a comunicação entre dispositivos que utilizam diferentes versões do protocolo IP , facilitando a transição do IPv4 para o IPv6. O NAT64 realiza a tradução dos pacotes IPv6 para o IPv4 e vice-versa, permitindo que dispositivos IPv6 acessem serviços e recursos disponíveis apenas em redes IPv4 [Matthews et al., 2011].

Apesar de suas vantagens, o NAT64 também apresenta algumas limitações. Uma delas é a possível perda de informações de endereçamento original durante o processo de tradução, o que pode dificultar a identificação precisa dos dispositivos envolvidos na comunicação, gerando uma sobrecarga adicional nos dispositivos de rede devido à necessidade de realizar a tradução de pacotes em tempo real Matthews et al. [2011]. Limitações essa que devem ser consideradas na implementação do NAT64, buscando mitigar seus impactos e garantir uma comunicação eficiente entre os diferentes protocolos.

2.9 ICMPv6

O *Internet Control Message Protocol Version 6* (ICMPv6) é um protocolo da camada de rede usado pelo IPv6 para fornecer informações de controle para dispositivos de rede.O ICMPv6 é uma evolução do ICMPv4 usado pelo protocolo IPv4 discutido na seção 2.5. O objetivo principal do ICMPv6 é fornecer informações de erro e controle no contexto do protocolo IPv6. Auxiliando os dispositivos de rede na identificação e correção de problemas de comunicação, abrangendo aspectos como informações de roteamento, erros no roteamento de pacotes e na descoberta de vizinhos. [Mukesh Gupta, 2006].

2.9.1 Mensagens ICMPv6

A Figura 2.10 ilustra o datagrama do protocolo ICMPv6 sendo composto por 3 campos. As funcionalidades de cada campo são semelhantes as definidas no modelo ICMPv4 seção 2.5.1.

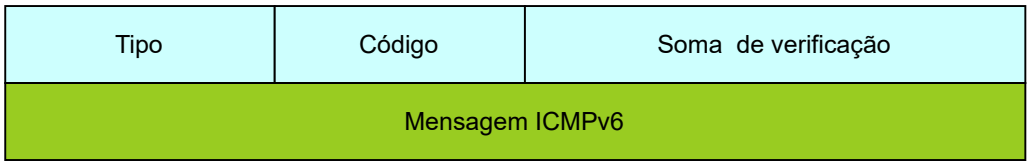

Figura 2.10: Cabeçalho ICMPv6

Segundo Narten et al. [2007b] dentre as funcionalidades do ICMPv6 as que mais se sobressaem são as apresentadas pelo *Neighbor Discovery Protocol* (NDP) sendo um protocolo essencial para comunicação de rede IPv6 utilizado por todos os hosts da rede, garantindo uma comunicação eficiente entre dispositivos além de descobrir o estado geral do enlace. Através de 5 mensagens distintas o NDP é operado sendo elas:

- ***Router Solicitation* (RS):** Usada para solicitar informações sobre os roteadores na rede. Quando um dispositivo ingressa em uma rede é enviado uma mensagem de solicitação de roteador para descobrir os roteadores na rede. Esta mensagem é identificada pelo código 133.
- ***Router Advertisement* (RA):** Os roteadores o utilizam para notificar os dispositivos na rede sobre sua presença e disponibilidade. Podendo também incluir informações sobre prefixos de rede, tempo de vida de endereços e outras informações importantes. Esta mensagem é identificada pelo código 134.
- ***Neighbor Advertisement* (NA):** Usada para responder à mensagens de solicitação e fornecer informações sobre a disponibilidade do dispositivo na rede. Esta mensagem é identificada pelo código 136.
- ***Neighbor Solicitation* (NS):** Usada para localizar o endereço *Media Access Control* (MAC) associado a um endereço IPv6. Um dispositivo envia uma mensagem

de solicitação de endereço IPv6 e o dispositivo com esse endereço responde com uma mensagem de resposta NA contendo seu endereço MAC. Esta mensagem é identificada pelo código 135.

- ***Redirect*:** Uma mensagem enviada de um roteador para um *host* informando a melhor rota disponível para o destino. Esta mensagem é identificada pelo código 137.

2.10 DHCPv6

Idealizado por Carney et al. [2003] o DHCPv6 da continuidade à proposta original do seu predecessor discutido na seção 2.6. O DHCPv6 surgi da necessidade de se possuir um mecanismo de atribuição de endereços IPv6 dinâmicos, que permitisse a configuração centralizada e automatizada dos clientes da rede sem a necessidade de configuração manual individual em cada host.

No âmbito operacional sofreu mudanças importantes como configuração de múltiplas interface com apenas uma requisição, reformulação de algumas estruturas e consequentemente não sendo compatível com seu antecessor [Davies, 2003].

2.10.1 Comunicação DHCPv6

Segundo Carney et al. [2003] o DHCPv6 utiliza o protocolo de transporte UDP. Sendo que clientes escutam mensagens DHCP na porta UDP 546. Já Servidores e agentes de retransmissão escutam mensagens DHCP na porta UDP 547. O DHCPv6 é baseado em uma arquitetura cliente-servidor, onde os clientes solicitam a configuração de um servidor DHCPv6 e o servidor responde com as informações de configuração solicitadas. A comunicação entre o dispositivo e o servidor DHCP ocorre através da troca de 4 mensagens:

- ***Solicit*:** O processo de comunicação começa quando um dispositivo cliente envia uma mensagem de solicitação (*solicit*) para o servidor DHCPv6.

- ***Advertise:*** O servidor DHCPv6 recebe essa mensagem e responde com uma mensagem de oferta (*advertise*) . Essa mensagem contém as informações de configuração que o servidor está oferecendo ao cliente.

- ***Request:*** O cliente então envia uma mensagem de solicitação *(request*) para confirmar a oferta do servidor.

- ***Reply*:** Uma vez que o servidor DHCPv6 recebe a confirmação do cliente, é enviado uma mensagem de resposta (*reply*) para confirmar a atribuição do endereço IPv6 e outras informações de configuração. A partir desse momento o dispositivo cliente está configurado e pronto para usar o endereço IPv6 atribuído.

2.10.2 Tipos de Endereços

De acordo com Brito [2018], ambos os protocolos IPv4 e IPv6 usam conceitualmente estratégias semelhantes para dividir seus endereços. O IPv4 foi dividido em endereços *unicast*, *multicast* e *broadcast* discutido na seção 2.6.2. Entretanto com a adição de alguns recursos essa estrutura não se enquadra no IPv6 sendo melhor descrita assim:

1. **Endereços *unicast* :** Um endereço unicast é um endereço IPv6 que identifica exclusivamente uma única interface de rede. Sendo usado para enviar pacotes de um nó para outro em uma rede IPv6. Os endereços Unicast podem ser globalmente ou localmente roteáveis, incluindo:

    - Endereço *unicast* Globalmente Roteável: Identifica exclusivamente uma interface de rede roteável globalmente na Internet, semelhante a um endereço IPv4 de rede pública.
    - Endereço *unicast* de Link-local: Identifica exclusivamente uma interface na rede local, semelhante a um endereço IPv4 privado.

2. **Endereços *Anycast*:** Um endereço *Anycast* é um endereço IPv6 que identifica um grupo de interfaces de rede, mas o pacote é encaminhado apenas para a interface mais próxima em termos de roteamento. Adicionando redundância e alta disponibilidade em redes IPv6. Os endereços *Anycast* incluem:

    - Endereço *Anycast* Globalmente Roteável: Identifica um conjunto de interfaces de rede roteáveis globalmente e permite que os pacotes sejam encaminhados para a interface mais próxima em termos de roteamento.
    - Endereço *Anycast* de Link-local: Identifica um conjunto de interfaces de rede em um *link* e permite que os pacotes sejam encaminhados para a interface mais próxima em termos de roteamento.

3. **Endereços *Multicast:*** Os endereços *multicast* são endereços IPv6 que identificam um grupo de interfaces de rede semelhante ao que ocorre com o *Anycast*. Entretanto os pacotes enviados para um endereço *multicast* são entregues a todas as interfaces pertencentes ao grupo identificado pelo endereço . Os endereços *multicast* incluem:

    - Endereço *Multicast* Globalmente Roteável: Identifica um grupo de interfaces de rede globalmente roteáveis.
    - Endereço *Multicast* de Link-local: Identifica um grupo de interfaces de rede em um link ou segmento de rede local.

2.10.3 Tipos de Configuração automática

Segundo Droms R [2003] o *Stateful* é a forma mais tradicional de usar o protocolo DHCP, nesse modo o servidor DHCP é responsável por atribuir endereço exclusivo a cada dispositivo conectado à rede. O servidor DHCP mantém armazenado o registro dos endereços atribuídos aos dispositivos .

No modo *stateless* o servidor DHCP fornece configurações de rede complementares como endereço de DNS, *Network Time Protocol* (NTP). Neste modo, em redes IPv6 os clientes utilizam um endereço IP configurado automaticamente através do mecanismo de autoconfiguração IPv6 [Droms R, 2003].

3

Planejar-Fazer-Verificar-Agir

3.1 Metodologia

Esse projeto utiliza como base a Metodologia experimental, tornando possível produzir os cenários para o estudo proposto permitindo que sejam observados individualmente. Para isso será empregado o uso de emuladores, onde os emuladores buscam reconstruir o sistema por meio do entendimento detalhado do funcionamento do mesmo, de forma que o resultado final seja fidedigno se comparado ao uso de equipamentos reais.

A pesquisa bibliográfica aprofundou os conhecimentos fundamentais ao qual foi utilizado no desenvolvimento dos cenários. Portanto, bases eletrônicas, sites de busca, biblioteca virtual e repositórios de Universidades foram utilizadas na construção do acervo literário essencial à realização da pesquisa. A modalidade de pesquisa bibliográfica utilizada leva em consideração as contribuições científicas ao tema central deste trabalho, Sendo consideradas as contribuições dos últimos 20 anos. Um longo período de tempo pois alguns dos *Request for Comments* (RFC) mencionados são do início da *Internet*.

No processo de desenvolvimento como método central foi utilizado o *Plan, Do, Check, Act* (PDCA) ilustrado na figura 3.1. Tratando-se de um projeto cíclico de melhoria contínua com foco intenso no planejamento e observação dos resultados do projeto tendo como pilares os pontos mencionados a seguir.

Planejamento: Nesta etapa identifica-se os objetivos e metas a serem alcançados onde se avalia os problemas ou oportunidade de melhoria para definir as metas a serem atingidas. Fazer: envolve colocar o plano em ação conforme planejado e coletar dados para possibilitar o monitoramento do processo e a medição dos resultados. Verificação: Analisar os resultados obtidos de médio a longo prazo e comparar com o plano inicial definido na primeira fase, para observar problemas ou falhas que podem ser ajustadas no próximo ciclo. Agir: com base na análise dos resultados devem ser definidas as ações necessárias para corrigir possíveis desvios e melhorar continuamente o processo.

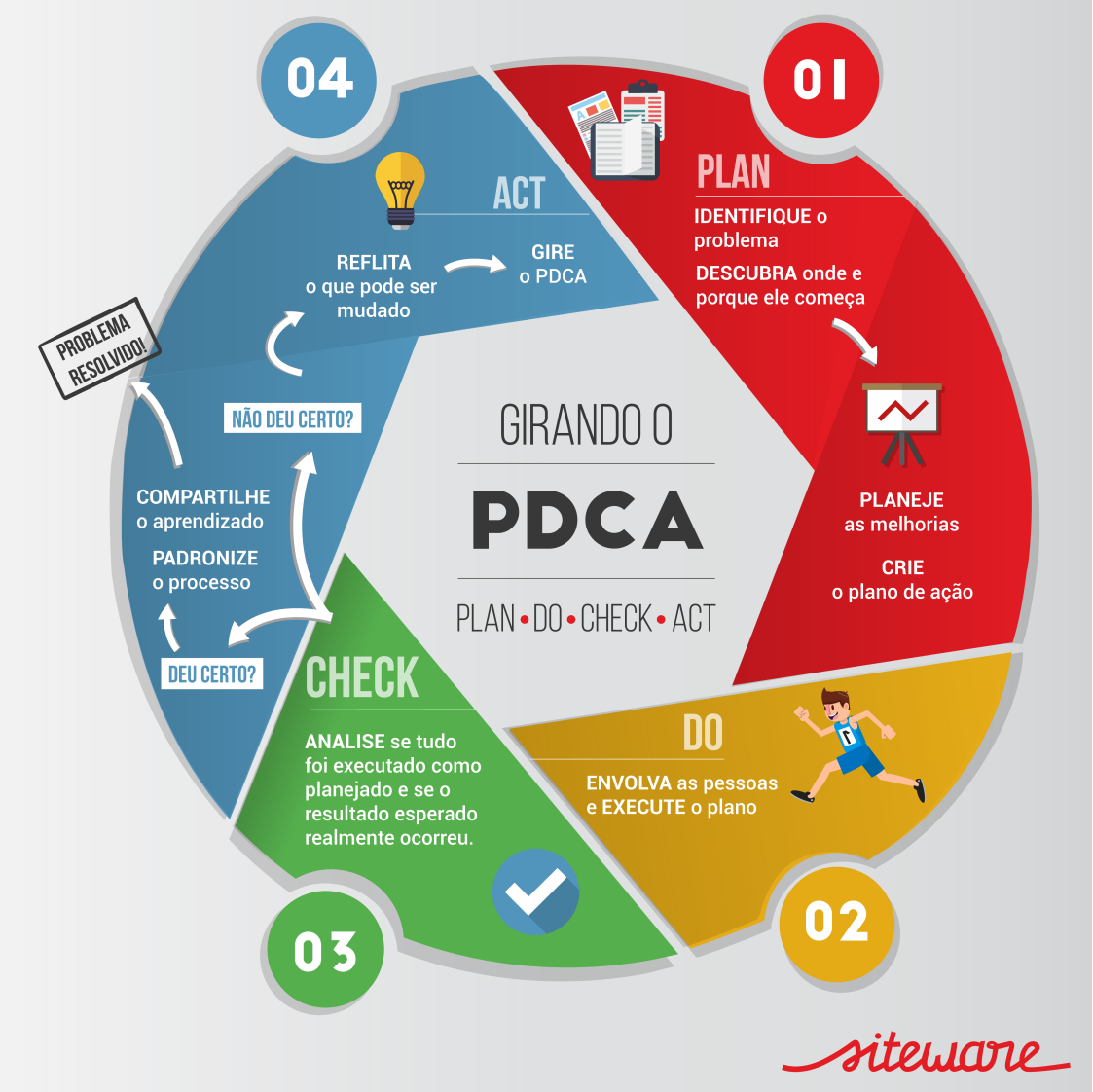

Figura 3.1: PDCA - Fonte: siteware.com

Os objetivos deste estudo foram apresentados na seção 1.1 do objetivo geral, visando uma delimitação mais precisa. Como resultado, as descobertas deste estudo serão apresentados em duas seções distintas, em conformidade com os pontos específicos estabelecidos.

No capítulo de Resultados, a seção 4.1, denominada Comportamento das Subfunções do IPV6, tem como finalidade demonstrar as subfunções mencionadas nos itens 2 e 3 do objetivo. Para alcançar esses pontos, a seção 3.2, intitulada Contexto Simulação Comportamental, apresenta a metodologia adotada para investigar o comportamento da obtenção de informações dos vizinhos. Por sua vez, a seção 4.2, denominada Análise de Desempenho, localizada no capítulo de Resultados, tem o propósito de atender ao objetivo descrito no item 3, fornecendo os resultados e a análise dos dados obtidos. Para obter tais dados, foi utilizada a metodologia apresentada na seção 3.3, intitulada Contexto Análise de Desempenho.

3.2 Contexto Simulação Comportamental

Nesta seção é apresentado os *software* utilizados no desenvolvimento das simulações. As simulações dos cenários 1 ao 6 foram realizadas exclusivamente no contexto de redes IPv6. Para construção dos cenários que visam demonstrar as subfunções do IPv6 foi utilizado o *software* core, que é um emulador de rede capaz de fornecer um ambiente para executar aplicativos e protocolos reais, aproveitando as ferramentas fornecidas pelo sistema operacional linux, disponivel em https://coreemu.github.io/core/.

O uso dos emuladores foi realizado através do *software virtual box*, disponível em virtualbox.org. Todas as capturas de pacotes foram realizadas através do *software wireshark*, disponível em *wireshark.org*. a ilustração das topologias de rede foram geradas através do draw.io. Todas as ferramentas mencionadas foram utilizadas em um computador com processador i3-2120, 12gb RAM, SSD 240 GB com sistema operacional *windows* 11 x64.

3.2.1 Estrutura

Foram adotadas algumas convenções para melhor compreensão do que será apresentado em cada cenário de simulação. A topologia utilizada nos cenários busca representar apenas o mínimo necessário para verificar as mensagens trocadas, podendo incluir hosts conectados diretamente, com ou sem a presença de *switches* ou roteadores intermediários. os cenários de 1 a 6 utilizaram-se de todos os passos do método PDCA.

A utilização da ferramenta *Wireshark* variou de acordo com o filtro escolhido, buscando facilitar o encontro e a análise das mensagens. A troca de pacotes, foi analisada por meio de figuras que destacam os campos mais relevantes para uma melhor compreensão. Todas os pacotes capturados podem ser encontradas no apêndice C.

3.3 Contexto analise de desempenho

Esta Seção possui objetivo de mostrar como foram conduzido os testes de desempenho do servidor DHCP em redes IPv4 e IPv6. Todos os testes de performance foram executados usando o *software* EVE-NG, sendo um emulador que permite criar e simular uma rede virtual cujo comportamento é o mais fidedigno possível se comparado a equipamentos reais, disponível em eve-ng.net. Avaliar o desempenho do DHCP em redes de computadores é crucial para determinar sua capacidade de lidar com condições operacionais distintas.

Para gerar os gráficos de visualização devido a diferença de performance e a impossibilidade de se obter valores idênticos, os pontos de referência para efeitos de comparação em redes IPv4 e IPv6 foram os mais próximos possíveis. Todos os códigos e dados obtidos dos testes estão disponíveis nos Apêndices A e B. Ressalta-se que não foram consideradas questões externas à simulação como a segurança, portanto não foram utilizados mecanismos de segurança comumente implementados em redes de computadores para evitar ataques ao serviço DHCP.

Nos testes de performance discutidos na Seção 4.2, utilizamos o método de comunicação padrão (*default*) de um servidor DHCP. Para redes IPv4 empregamos o modelo D.O.R.A (Descoberta, Oferta, Requisição, Confirmação). Por outro lado para os testes realizados usando IPv6, usamos o modelo S.A.R.R (Solicitação, Anúncio, Requisição, Resposta). Nas próximas subseções, serão descritos os cenários de teste definidos e como foram conduzidos os testes de desempenho.

3.3.1 Cenários

Para avaliar o desempenho do servidor DHCP foram considerados dois testes distintos: carga e estresse. O teste de carga simula condições normais de uso, enquanto o teste de estresse testa o comportamento do servidor sob condições severas. No teste de estresse o número de requisições será incrementado progressivamente até que o serviço não seja capaz de responder a nenhuma solicitação adicional.

Entendemos que testes de carga de um servidor DHCP demanda análise de múltiplas variáveis para que seja possível identificação de valores referenciais das métricas consideradas como condições normais de uso. Neste sentido, para mensurar o desempenho do servidor DHCP através dos testes de carga, submetemos quantidades variáveis de requisições de endereços IP com objetivo de estabelecer limites referenciais para o funcionamento do serviço em condições de uso típicas.

Os testes de estresse têm como finalidade avaliar a estabilidade e confiabilidade do serviço em condições extremas, simulando altas taxas de dados e tráfego que ultrapassem os limites de uso normal. Durante o teste, o sistema é submetido a condições severas para verificar como ele responde a essas situações. A seguir, mostramos detalhes da topologia de rede utilizada para aplicação dos testes de carga e estresse.

3.3.2 Métricas

Este trabalho realizou a coleta e análise das seguintes métricas de interesse: quantidade de solicitações, quantidade de pacotes descartados e tempos mínimo, médio e máximo para entrega de endereçamento IP.

A métrica quantidade de solicitações tem como objetivo mensurar o comportamento do serviço DHCP com o aumento gradual do número de requisições enviadas pelos clientes. A métrica quantidade de pacotes descartados registra os pacotes com mensagens DHCP perdidos não entregues pelo servidor ou pelo cliente. As métricas de tempo mínimo, médio e máximo são importantes para avaliar a eficiência e velocidade da entrega das configurações IP providas pelo servidor DHCP.

3.3.3 Topologia

A Figura 3.2 apresenta a topologia adotada para avaliar o desempenho do servidor DHCP por meio de testes. A configuração da rede consistiu na utilização do simulador EVE-NG, no qual uma máquina GNU/Linux Ubuntu foi conectada diretamente ao roteador 1905 que executou o sistema operacional Cisco IOS versão 15. Na máquina Linux, foi instalada a ferramenta de performance perfdhcp, enquanto que no roteador habilitamos e configuramos o serviço DHCP para redes IPv4 e IPv6.

Figura 3.2: Topologia para Análise de Desempenho do DHCP

3.3.4 Perfdhcp - Definição dos Parâmetros dos Testes

Como ferramenta de desempenho, o Perfdhcp foi utilizado por ser um meio de *benchmarking* capaz de mensurar o desempenho de servidores DHCP através da simulação de grandes quantidades de requisições, disponível em isc.org. A ferramenta perfdhcp oferece diversos parâmetros configuráveis para a realização de testes e coleta de informações sobre o desempenho do servidor DHCP. Na Figura 3.3, é possível verificar a quantidade de opções disponíveis para personalização da ferramenta. Para facilitar a compreensão, os parâmetros foram agrupados em cinco categorias distintas.

- **Modo de operação:** Permite a escolha da versão do protocolo IP e o número de vias a serem utilizadas na comunicação cliente-servidor DHCP.
- **Opções de camada IP:** Especificam as interfaces e portas a serem usadas nas comunicações.
- **Controle de taxa:** Permite definir a quantidade de mensagens a serem enviadas em um determinado período de tempo.
- **Conclusão de teste:** Estabelece as condições para a finalização dos testes, como quantidade de requisições, período de tempo e número máximo de pacotes descartados.
- **Relatório:** Permite a visualização dos dados produzidos pelos testes para análise de desempenho do servidor DHCP.

```
perfdhcp [-1] [-4 | -6] [-A encapsulation-level] [-b base] [-B] [-c] [-C separator] [-d drop-time]
[-D max-drop] [-e lease-type] [-E time-offset] [-f renew-rate] [-F release-rate] [-g thread-mode]
[-h] [-i] [-I ip-offset] [-J remote-address-list-file] [-l local-address|interface] [-L local-port]
[-M mac-list-file] [-n num-request] [-N remote-port] [-O random-offset] [-o code,hexstring]
[-p test-period] [-P preload][-r rate] [-R num-clients] [-s seed] [-S srvid-offset][scenario name]
[-t report] [-T template-file] [-u] [-v] [-W exit-wait-time] [-w script_name][-x diagnostic-selector]
[-X xid-offset] [server]
```

Figura 3.3: Parâmetros de Configuração da Ferramenta Perfdhcp

3.3.5 Protocolo de Execução dos Testes

O protocolo do teste elucida como os testes foram executados. De acordo com a RFC 3315 Carney et al. [2003] o tempo limite inicial para a resposta de um servidor DHCP é de 1 segundo. Já a RFC 2131 Droms [1997] não é tão especifica mas é limitado a sugerir um tempo limite de 3 segundos para conexões *Ethernet* com 10MB/s. Deste modo, definimos no protocolo de testes o tempo de 1 segundo como fator decisivo para considerar a perda de um pacote.

A prática de dividir os testes de desempenho em categorias é comum, como discutido na seção 3.3.1 dividimos os testes em dois cenários distintos. Com o objetivo de delimitar o escopo de cada um deles, no teste de carga, será estabelecido um limite de 70% para a porcentagem de pacotes perdidos.

No Teste de Estresse, não haverá a imposição desse limite, e a rodada de requisições terá início com a quantidade de solicitações em que o teste de carga ultrapassou o referido limite de perda de pacotes.

A ferramenta perfdhcp emula uma quantidade X de solicitações de clientes para obtenção de endereços IP. Para realização dos testes utilizamos o modelo de comunicação padrão. A cada nova execução dos testes alteramos a quantidade de solicitações a fim de mensurar o desempenho do servidor DHCP sob diferentes condições de carga. Dessa forma executamos o seguinte protocolo de testes:

1. **Executamos 30 (trinta) iterações com as seguintes configurações:**

    - **-p 60 (Valor Fixo):** Parâmetro que estabelece a duração do teste em segundos. A quantidade de requisições é calculada ao se multiplicar a duração do teste pela quantidade de solicitações por segundo.
    - **-r <> (Valor Variável):** Estabelece a quantidade de solicitações por segundo.
    - **-d 1 (Valor Fixo):** Estabelece o tempo limite para recebimento de resposta do servidor DHCP.

- **-D <> (Valor Variável):** Estabelece a quantidade máxima de pacotes que podem ser descartados de acordo com o número de solicitações. Assim como o parâmetro de tempo esse também é uma das condições de encerramento do teste.

2. **Os dados obtidos de cada iteração são registrados.**

3. **Cálculo da média:** ignorando valores discrepantes (máximo e mínimo) a média dos valores obtidos é calculada.

4. **Reinício do servidor DHCP:** Antes da execução de um novo teste com um número maior de requisições, reiniciamos o serviço DHCP. A fim de garantir que todos os *caches* fossem eliminados de modo a minimizar interferência em testes posteriores.

4

Resultados: Fazer-Verificar-Agir

4.1 Comportamento Subfunções IPv6

Os cenários subsequentes foram desenvolvidas com base nos ciclos completos do método PDCA. Inicialmente, foi realizado o planejamento utilizando as RFCs que abordam o uso do IPv6 e do DHCP, servindo como ponto de partida para definir os objetivos dos cenários e os recursos necessários, como a escolha do emulador e do software de captura de pacotes.

No ciclo fazer, foi configurado o ambiente de simulação e implementados os cenários para geração das mensagens de estudo. Em seguida, na ciclo de verificação, os resultados da simulação foram analisados, garantindo que as mensagens transmitidas estivessem em conformidade com as informações obtidas nas RFCs.

Na última etapa do método, caso alguma inconsistência fosse identificada, eram realizados os ajustes e correções necessários. Os resultados finais da simulação foram devidamente documentados, e uma análise dos resultados em relação aos objetivos estabelecidos foi realizada.

4.1.1 Cenário 1: NDP - Solicitação e Anúncio de Vizinhos

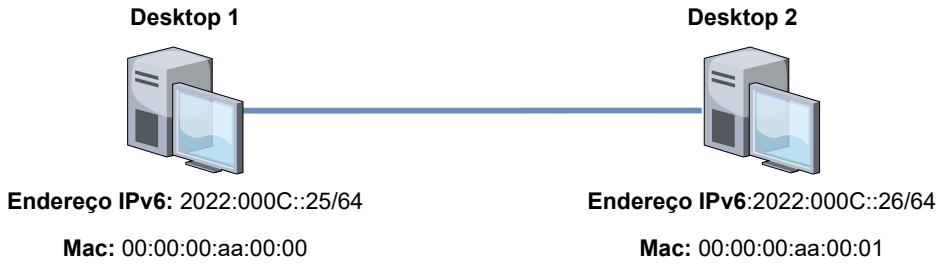

Figura 4.1: Cenário 1 - Envio de Mensagens NS e NA

Nesse primeiro cenário a funcionalidade do Protocolo NDP do IPv6 será demonstrada, com destaque para as funções das mensagens NS e NA. A finalidade da experiência é ilustrar como esses protocolos são empregados para descobrir o endereço físico dos dispositivos na rede IPv6, permitindo assim o encaminhamento eficiente de pacotes. A função das mensagens NS e

NA é semelhante há desempenhada pelo *Address Resolution Protocol* (ARP) no protocolo IPv4 [Simpson et al., 2007]. A topologia utilizada nesse cenário é ilustrada na Figura 4.1.

A Solicitação de Vizinho é iniciada quando um dispositivo precisa alcançar outro dispositivo em uma rede local. Antes de enviar o pacote o dispositivo emissor envia um mensagem NS através do grupo multicast solicited node (ff02::1:ffXX:XXXX). Na mensagens NS o endereço IPv6 a ser resolvido é indicado no campo *Target*. O campo denominado *Source link layer address* informa ao *host* destinatário o endereço físico do *host* de origem.

Ao receber uma mensagem NS, o dispositivo verifica se é o destinatário do endereço. Se for o caso, responde com uma mensagem NA enviada como uma transmissão *unicast*, direcionada ao dispositivo solicitante contendo as informações de endereçamento. Na mensagem NA o endereço físico do dispositivo é inserido no campo *Target link-layer address*.

As informações que mapeiam a relação entre os endereço físico e lógico são armazenadas na tabela chamada *neighbor cache* "cache de vizinhos". As mensagens capturadas através da ferramenta *wireshark*, encontram-se no apêndice C figuras C.1 e figuras C.2.

4.1.2 Cenário 2: NDP - Solicitação de Roteador

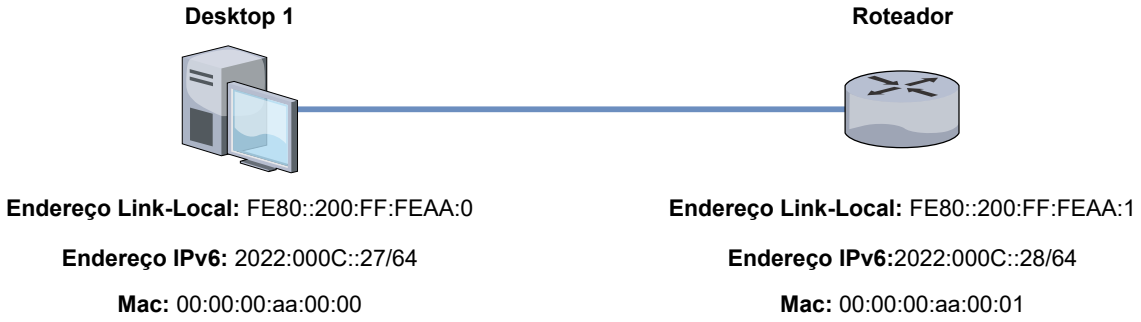

Figura 4.2: Cenário 2 - Solicitação de Roteador

A ação da descoberta de roteadores pode ser provocada pelos *hosts* de uma rede ou pelo próprio roteador. Nesta subseção mostramos como um *host* envia mensagens RS após inicialização de sua interface de rede, para obter como resposta um RA emitida pelo roteador. No IPv4, esta função é realizada através das mensagens ARP *Request*. Na subseção 4.1.3 mostramos como essa ação de descoberta pode ser iniciada pelo dispositivo de roteamento da rede.

Conforme Simpson et al. [2007], uma solicitação de roteador se inicia no momento que um *host* ingressa na rede pela primeira vez, e/ou se reconecta a mesma com o intuito de verificar quais roteadores estão presente na rede e quais parâmetros de endereçamento esta sendo anunciados. A Figura 4.2 ilustra como um *host* realiza o processo de descoberta de roteadores na rede, através do uso de mensagens de solicitação de roteador.

O *host* ao ingressarem na rede enviam uma mensagem RS aos roteadores disponíveis no enlace através do endereço de grupo *multicast all routers* (ff02::2) . O *host* solicitante também

envia seu endereço MAC o que evita que os roteadores tenham que fazer uso de uma descoberta de vizinhança, como foi demonstrado na simulação do Cenário 4.1.1.

O roteador ao receber uma mensagem RS responde com uma mensagem RA tendo como origem o seu endereço de link-local e como destino o endereço *multicast all-nodes* (ff02::1). Dentro do pacote RA é enviado parâmetros para configuração de endereçamento. Ressaltando que dependendo das configurações realizadas no roteador pode-se ativar um mecanismo periódico de envio automático de mensagens RA.

Ao receber as mensagens de anúncio de roteador o host da rede pode atualizar suas tabelas de roteamento, e configurar adequadamente seus parâmetros de rede com base nas informações fornecidas pelos roteadores.

Para possibilitar a captura dos pacotes enviados o *Desktop1* da Figura 4.2 teve sua interface de rede desabilitada e habilitada em sequência. As mensagens trocadas durante esse cenário encontram-se no apêndice C figuras C.3 e figuras C.4.

4.1.3 Cenário 3: NDP - Anúncio de Roteador

Na subseção 4.1.2 visualizamos como um *host* pode enviar uma mensagem do tipo RS ao conectar-se em uma rede obtendo uma mensagem RA do roteador como resposta. O cenário desta subseção utiliza a mesma topologia da Figura 4.2. Conforme discutido por Simpson et al. [2007] essa subseção demonstra como os roteadores podem habilitar o envio periódico de mensagens RA.

O seguinte trecho de código contem a estrutura utilizada para demonstrar o mecanismo de envio periódico de mensagens RA através do software Quagga.

```
1 interface eth0
2     # habilita o envio de mensagens RA
3     no ipv6 nd suppress-ra
4     # Intervalo para divulgar a mensagens RA
5     ipv6 nd ra-interval 5
6     # Especifica o IP do roteador
7     ipv6 address 2022:c::28/64
8 !
```

Listing 4.1: Arquivo de configuração Quagga.conf

A Figura C.5 presente no apêndice C exibe a mensagem RA capturada através do *wireshark*. Enviado pelo roteador apos ser configurado para divulgar de forma periódica as mensagens de anúncio, permitindo que os *hosts* o localizem sem precisar enviar a mensagens RS.

4.1.4 Cenário 4: NDP - Detecção de endereços duplicados

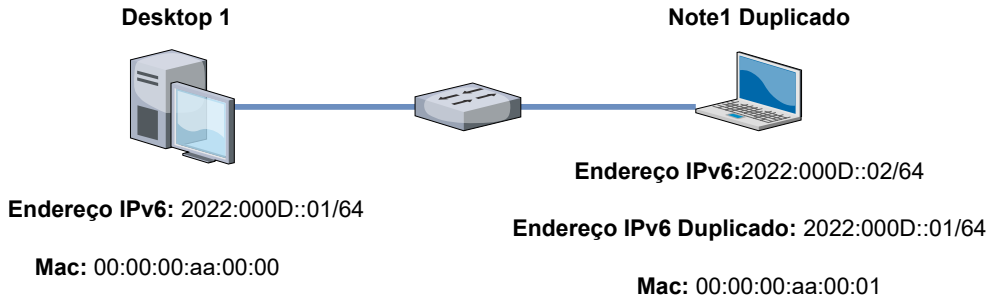

Figura 4.3: Cenário 4 - Verificação de Endereços Duplicados

Conforme discutido pelos autores Narten et al. [2007a] e Moore [2006] nesta subseção vamos mostrar como o *Duplicate Address Detection* (DAD) funciona em redes IPv6, no ambiente IPv4 é utilizado a mensagem ARP *request* e o método *gratuitous* ARP para executar a mesma função. Para simular endereços IP duplicados configuramos dois hosts com o mesmo endereço IP de forma proposital. A topologia usada para verificar a detecção de endereços duplicados é mostrada na Figura 4.3.

O DAD deve ser realizado antes do dispositivo IPv6 atribuir o endereço a uma interface, seja através de autoconfiguração stateless , DHCPv6 ou configuração manual. O DAD é realizado seguindo o seguinte processo. O host envia uma mensagem NS para o endereço *multicast solicited-node* (ff02::1), com o campo *target address* sinalizado com o endereço IPv6 que sera utilizado. É semelhante ao que ocorre no cenário 4.1.1, mas o campo *source* da mensagem NS fica marcado como "::"(não especificado) porque o dispositivo ainda não possui um endereço de rede válido e isso evita conflitos.

Caso uma mensagem NA seja recebido como resposta, o endereço IPv6 já está em uso, o dispositivo que realizou o DAD assume que existe um conflito de endereços na rede. O tempo padrão para aguardar uma resposta NA é de 1 segundo. Caso não haja resposta o endereço IPv6 alvo está disponível para ser utilizado. A Figura C.6 presente no apêndice C mostra a mensagem NS capturada através do *wireshark*.

A Figura 4.4 mostra o retorno do comando para verificação do endereço IP do dispositivo Note1Duplicado presente na Figura 4.3. Note que, como destacado na Figura houve uma mensagem de erro DAD para informar o conflito e impedir o uso do endereço duplicado.

```
root@Note1Duplicado:/tmp/pycore.35288/Note1Duplicado.conf# ip addr show dev eth0
20: eth0: <BROADCAST ,MULTICAST ,UP ,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:00:00:aa:00:01 brd ff:ff:ff:ff:ff:ff
    inet6 2022:d::1/64 scope global tentative dadfailed
       valid_lft forever preferref_lft forever
    inet6 fe80::200:ff:feaa:1/64 scope link
       valid_lft forever preferref_lft forever
root@Note1Duplicado:/tmp/pycore.35288/Note1Duplicado.conf#
```

Figura 4.4: Verificação de Atribuição de Endereço

4.1.5 Cenário 5: DHCPv6 *Stateful*

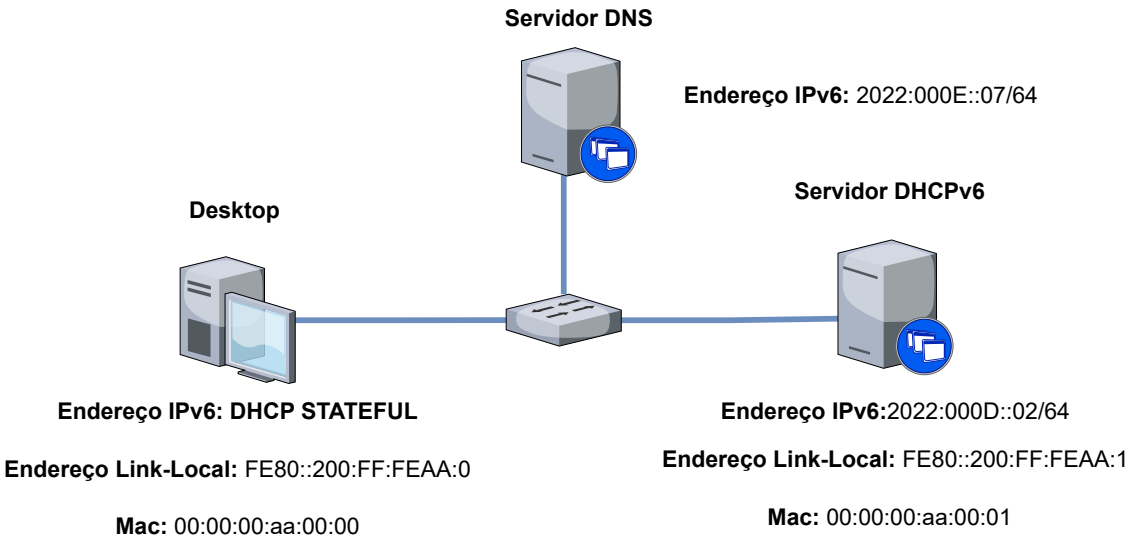

Figura 4.5: Cenário 5 - Topologia DHCPv6 *Stateful*

Nesta subseção vamos simular a operação do DHCPv6 no modo stateful. O Servidor DHCPv6 atribui endereço IPv6 para os dispositivos da rede, mantém o registro do status dos clientes e fornece informações complementares. O DHCPv6 possui dois modos de operação *stateful* discutido nessa subseção e *stateless* discutido subseção 4.1.6.

Conforme a topologia apresentada na Figura 4.5 o *Desktop* atuará como cliente da rede sendo responsável por realizar a requisição de endereços, e o servidor DNS sera utilizado para validar se os parâmetros divulgado pelo servidor DHCPv6 foram obtidos. Observe que a topologia não possui um roteador e o serviço DHCPv6 não informa o roteador padrão para os clientes. Como resultado os dispositivos do enlace não possuem uma rota padrão, e consequentemente não conseguiriam alcançar novas redes.

Conforme descrita por Carney et al. [2003] o processo de comunicação DHCPv6 envolve quatro mensagens principais. Primeiro o host envia uma mensagem de solicitação "*Solicit*"para o grupo *multicast all-dhcp-agents* (ff02::1:2). O servidor DHCPv6 responde com uma mensagem de anúncio "*Advertise*"contendo parâmetros de rede disponíveis para uso, essa mensagem é enviada por *unicast* direcionada ao endereço link-local do cliente.

Se o host aceitar a oferta é feita uma solicitação "*Request*"para confirmar a escolha dos parâmetros de rede ofertados , destinada ao endereço de *multicast all-dhcp-agents* (ff02::1:2). O servidor DHCPv6 responde com uma mensagem "*Reply*"confirmando a atribuição do endereço IPv6 e enviando a mensagem através do endereço link-local do cliente. As Figuras C.7, C.8, C.9 e C.10 presente no apêndice C ilustram as mensagens DHCP trocadas durante o cenário.

O trecho de código a seguir aborda o arquivo de configuração DHCP utilizados no servidor DHCPv6 para operar no modo *Stateful*.

```
1 # tempos de emprestimos
```

```
default-lease-time 600;
max-lease-time 7200;
# pool dos IPs fornecidos
subnet6 2022:e::64{
    range6 2022:e:1111 2022:e::abcd;
    option Dhcp6.name-servers 2022:e::7;
}
```

Listing 4.2: Arquivo de configuração DHCPv6.conf

Ao iniciar a interface de rede do dispositivo Desktop, foi obtido um endereço IP através do servidor DHCP *stateful*. A Figura 4.6 apresenta a verificação realizada no dispositivo Desktop para confirmar o funcionamento do servidor DHCP.

```
root@Desktop:/tmp/pycore.47979/Desktop.conf# ip addr show
7: eth0: <BROADCAST ,MULTICAST ,UP ,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:00:00:aa:00:00 brd ff:ff:ff:ff:ff:ff
    inet6 2022:e::abcc/64 scope global
       valid_lft forever preferref_lft forever
    inet6 fe80::200:ff:feaa:0/64 scope link
       valid_lft forever preferref_lft forever
root@Desktop:/tmp/pycore.47979/Desktop.conf#  cat /etc/resolv.conf
nameserver 2022:e::7
root@Desktop:/tmp/pycore.47979/Desktop.conf#
```

Figura 4.6: Endereço Obtido Através do DHCPv6 *stateful*

4.1.6 Cenário 6 - DHCPv6 *Stateless*

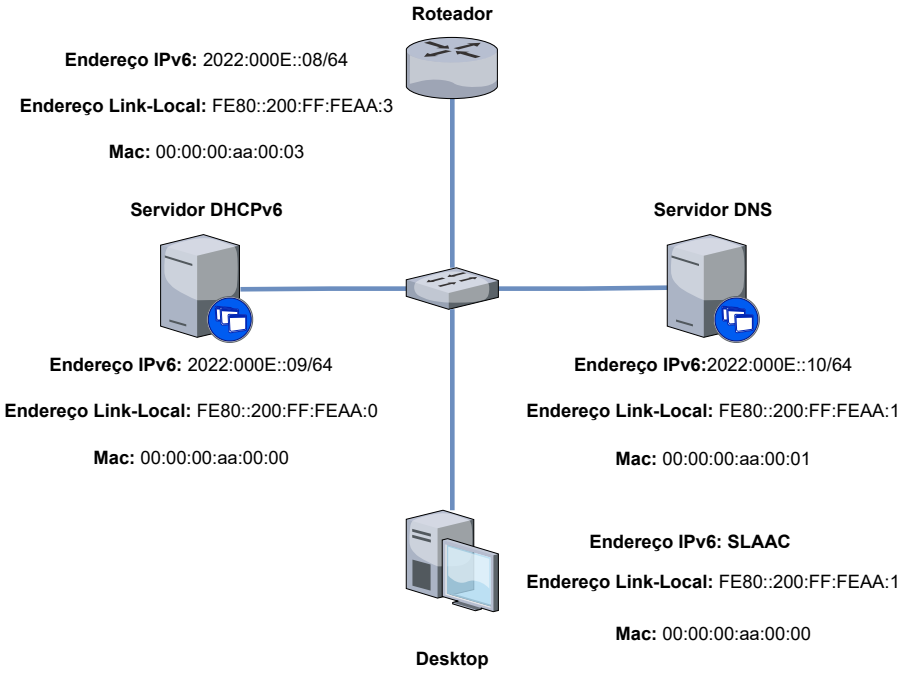

Figura 4.7: Cenário 6 - Topologia DHCPv6 *Stateless*

Nesta subseção o DHCPv6 opera em seu modo stateless e o processo de autoconfiguração *Stateless Address Autoconfiguration* (SLAAC) é demonstrado. Com o uso desses recursos os servidores DHCPv6 distribuem apenas informações de rede complementares sem armazenar informações de status, os hosts por sua vez geram seus próprios endereços de rede.

Na topologia apresentada na Figura 4.7 o roteador tem a função de encaminhar mensagens RA que divulgam o prefixo global IPv6 e a *flag Other Configuration*. Cabendo ao host *Desktop* gerar o seu próprio endereço utilizando o recurso SLAAC e solicitar informações complementares ao servidor DHCPv6, que por sua vez indicará o endereço do servidor DNS da rede.

O DHCPv6 não oferece a opção de roteador padrão, o que significa que os clientes precisam utilizar o serviço DHCPv6 em conjunto com o protocolo NDP para acessar outras redes, ou configurar manualmente o roteador padrão em cada dispositivo.

Para que o NDP funcione em conjunto com o DHCPv6, é necessário habilitar o envio de mensagens RA nos roteadores, permitindo que o roteador se anuncie como roteador padrão e possa manipular duas *flags* na mensagem RA: a *flag Managed addres configuration* , que permite que os dispositivos obtenham endereços de rede através do DHCPv6, e a *flag Other configuration* , que habilita o recebimento de configurações complementares do servidor DHCPv6.

O trecho de código a seguir mostra o arquivo de configuração utilizado no roteador para enviar mensagens RA e permitir que os hosts da rede façam a autoconfiguração stateless. A Figura C.11 presente no apêndice C apresenta a captura da mensagem RA enviada pelo roteador.

```
1 interface eth0
2 # habilita o envio do RA e prefixo da rede
3     no ipv6 nd suppress-ra
4     ipv6 nd ra-interval 5
5     ipv6 nd prefix 2022:e::/64
6 # Especifica o uso das flags e indica o DHCP da rede
7     no ipv6 nd managed-config-flag
8     ipv6 nd other-config-flag
9     ipv6 address 2002:e::8/64
10 !
```

Listing 4.3: Arquivo de configuração Quagga.conf

Neste cenário, o pacote RA enviado pelo roteador não inclui informações sobre o servidor DNS a ser utilizado. Essa função é desempenhada pelo servidor DHCPv6 no modo *stateless*. O trecho de código a seguir apresenta os parâmetros utilizados no servidor DHCPv6 para operar no modo stateless e indicar o servidor DNS da rede.

```
1 # especifica a rede e servidor dns a ser utilizado
2 interface eth0
3 subnet6 2022:e::/64 {
4 option dhcp6.name-servers 2022:e::10;
```

```
5 }
```

Listing 4.4: Arquivo de configuração DHCP.conf

O SLAAC é um recurso que permite aos hosts configurar seu próprio endereço exclusivo na rede. Esse processo é realizado com base nas informações contidas na mensagem RA enviada pelos roteadores. O endereço é formado pela combinação do prefixo da rede obtido na mensagem RA com os primeiros 64 bits do endereço, enquanto os últimos 64 bits são gerados usando o endereço MAC como base. Segundo Narten et al. [2007a] o SLAAC abrange a obtenção do endereço link-local e a detecção de endereços duplicados , conforme especificado por .

Para simular o uso do DHCP no modo stateless, foram feitas algumas alterações na configuração do *Desktop*, a fim de garantir o correto uso dos recursos na topologia. O código a seguir apresenta os comandos utilizados para realizar essas alterações.

```
1 interface eth0 {
2 # indica o uso do DHCPv6 no modo Stateless
3 information-only;
4 request domain-name-servers;
5 script "/etc/wide-dhcpv6/dhcp6c-script";
6 };
```

Listing 4.5: Arquivo de configuração DHCP.conf

Conforme discutido por Droms [2004] , para que os dispositivos do enlace possam obter informações adicionais, é necessário enviar uma mensagem de requisição chamada *Information Request* para o servidor DHCPv6 *stateless*. Em resposta, o servidor enviará uma mensagem *Reply* contendo as informações solicitadas, como o endereço DNS ou outras informações complementares. As Figura C.12 e C.13 presente no apêndice C mostra as mensagens *Information Request* e *Reply* capturadas por meio do *Wireshark*.

O endereço de rede autoconfigurado através da mensagem RA, e os parâmetros complementares solicitados ao DHCP *stateless* são apresentados na Figura 4.8.

```
root@Desktop:/tmp/pycore.33181/Desktop.conf# ip addr show eth0
8: eth0: <BROADCAST ,MULTICAST ,UP ,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:00:00:aa:00:00 brd ff:ff:ff:ff:ff:ff
    inet6 2022:e::200:ff:feaa:0/64 scope global
       valid_lft forever preferref_lft forever
    inet6 fe80::200:ff:feaa:0/64 scope link
       valid_lft forever preferref_lft forever
root@Desktop:/tmp/pycore.33181/Desktop.conf#  cat /etc/resolv.conf
nameserver 2022:e::10
root@Desktop:/tmp/pycore.33181/Desktop.conf#
```

Figura 4.8: Endereço Obtido Através do DHCPv6 *stateless*

4.2 Análise de Desempenho do Servidor DHCP em Redes IPv4 e IPv6

Os resultados obtidos por meio da metodologia descrita na seção de análise de desempenho 3.3 são apresentados nas subseções a seguir. Da mesma forma que na seção de comportamento 4.1, os princípios fundamentais do método PDCA também foram aplicados para a obtenção desses resultados.

4.2.1 Teste de Carga - Resultados e Análise

Para análise das métricas definidas no cenário foram realizados 58 conjuntos de testes com os protocolos IPv4 e IPv6. Cada conjunto de teste abrange 30 iterações resultando em um total de 1.740 testes. A fim de proporcionar uma melhor compreensão do comportamento dos protocolos e coletar o máximo de dados possíveis, as cargas de requisições foram variadas dentro do limite estabelecido para delimitar o escopo do teste, que variaram entre 1.000 a 190.128 solicitações. Tal amostra ampla possibilitou um entendimento mais abrangente dos resultados obtidos nos testes.

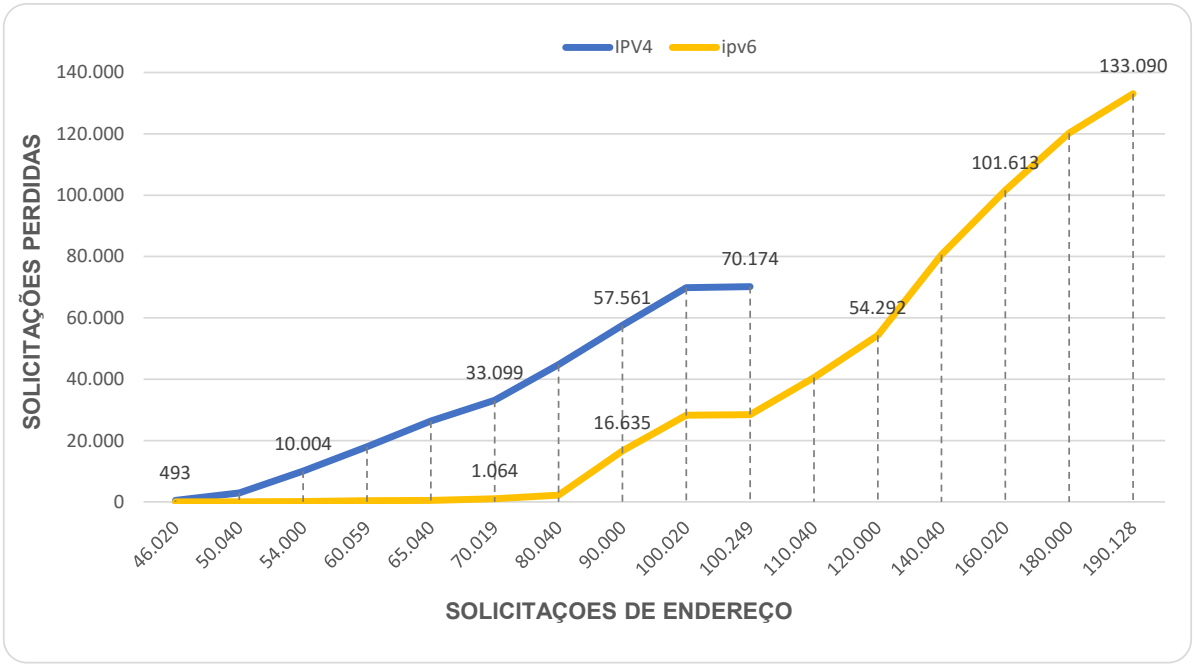

Figura 4.9: Gráfico Comparativo de Quantidade Requisições x Quantidade de Pacotes Perdidos em redes IPv4 e IPv6

Como atestado pela Figura 4.9, os resultados obtidos revelaram uma notável disparidade na taxa de solicitações bem-sucedidas entre os protocolos IPv4 e IPv6. Notavelmente, o protocolo IPv6 demonstrou um desempenho superior, evidenciado por um número substancialmente menor de solicitações perdidas em todas as faixas de requisições testadas.

A menor taxa de solicitações perdidas no IPv6 em comparação com o IPv4 indica uma melhor qualidade de serviço, o que é especialmente valioso para ambientes que demandam de uma quantidade considerável de dispositivos conectados.

Demonstrando uma diferença significativa no número de solicitações requisitadas alcançado, o protocolo IPv6 se destaca como uma solução altamente escalável para redes de computadores. Essa característica é de suma importância para o seu emprego em redes de grande porte, onde a capacidade de lidar com um volume crescente de requisições de forma uniforme e eficiente é essencial. Essa diferença significativa entre os protocolos reforça a eficiência do IPv6 em relação ao IPv4 no contexto de alcance e qualidade de serviço.

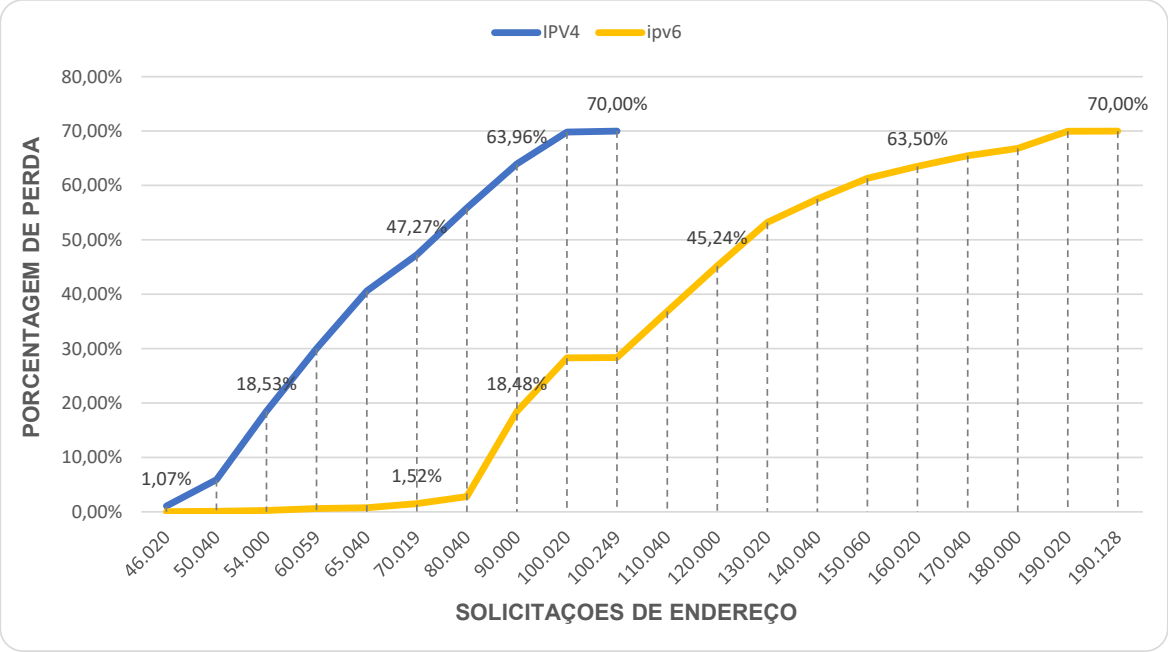

Figura 4.10: Gráfico Comparativo de Quantidade Requisições x Percentual de Pacotes Perdidos em redes IPv4 e IPv6

Ao comparar o limite estabelecido para delimitação do escopo do teste com a quantidade de solicitações requisitadas representada na Figura 4.10, observa-se um padrão distinto. Em ambos protocolos as perdas de pacotes permanecem próximas a 0% quando o número de requisições é inferior a 45 mil. Entretanto, ao incrementar suavemente a quantidade de requisições, verifica-se que para o protocolo IPv4 as perdas de pacotes chegam a cerca de 1% quando atingidas 46 mil requisições, enquanto para o protocolo IPv6 esse mesmo percentual de perda só é observado quando se aproxima das 67 mil requisições.

Ao aumentar a carga na rede, torna-se evidente um notável aumento no número de pacotes perdidos em ambos os protocolos. No entanto pode-se notar a maior robustez do protocolo IPv6 ao longo de todo o teste. Essa superioridade torna-se aparente quando se alcança o limiar estabelecido de 70% de perda de pacotes: no caso do IPv4, tal limite é atingido na marca das 100.020 requisições, enquanto que no IPv6 somente é observado por volta das 190.128 requisições. Esses resultados ressaltam a capacidade do IPv6 em lidar com cargas elevadas de solicitações de endereço, conferindo um serviço mais estável em comparação com o IPv4. Tal constatação é de grande importância para ambientes que exigem alta disponibilidade e desempenho contínuo de seus serviços de DHCP.

A diferença de desempenho entre os protocolos IPv4 e IPv6 pode ser atribuída à reformulação do cabeçalho IPv6, que possibilita um gerenciamento mais eficiente da fragmentação de pacotes. O IPv6 apresenta vantagens adicionais, como suporte nativo Qos, priorizando os pacotes de acordo com as necessidades específicas de largura de banda, latência e *jitter*. Além disso, a escolha de utilizar grupos *multicast* em vez de *broadcast* contribui para a redução da perda de pacotes devido a colisões e aumenta a confiabilidade dos pacotes transmitidos. Essas melhorias no IPv6 resultam em um desempenho aprimorado e uma maior capacidade de garantir a integridade e a entrega eficiente dos dados na rede.

Os gráficos de tempo de resposta exibidos nas Figuras 4.11 e 4.12 revelam que tanto o IPv4 quanto o IPv6 demonstraram desempenho satisfatório em termos de tempo de resposta. Ambos os protocolos apresentaram valores médios, mínimos e máximos de tempo semelhantes, com diferenças mínimas variando de 0,04 a 5,35 milissegundos nas faixas de requisições semelhantes. No entanto, apesar dessa equivalência em relação aos tempos de resposta, o IPv6 apresentou uma latência menor, conferindo-lhe uma vantagem sobre o IPv4.

Essa menor latência no IPv6 pode ser atribuída a uma série de melhorias e otimizações implementadas nesse protocolo. O IPv6 possui um cabeçalho mais simplificado, com menos campos e opções, o que resulta em um processamento mais ágil e eficiente em comparação com o IPv4. Essa maior velocidade de processamento tem impacto direto na usabilidade do serviço DHCPv6 , possibilitando uma atribuição de endereços IP mais confiável e previsível.

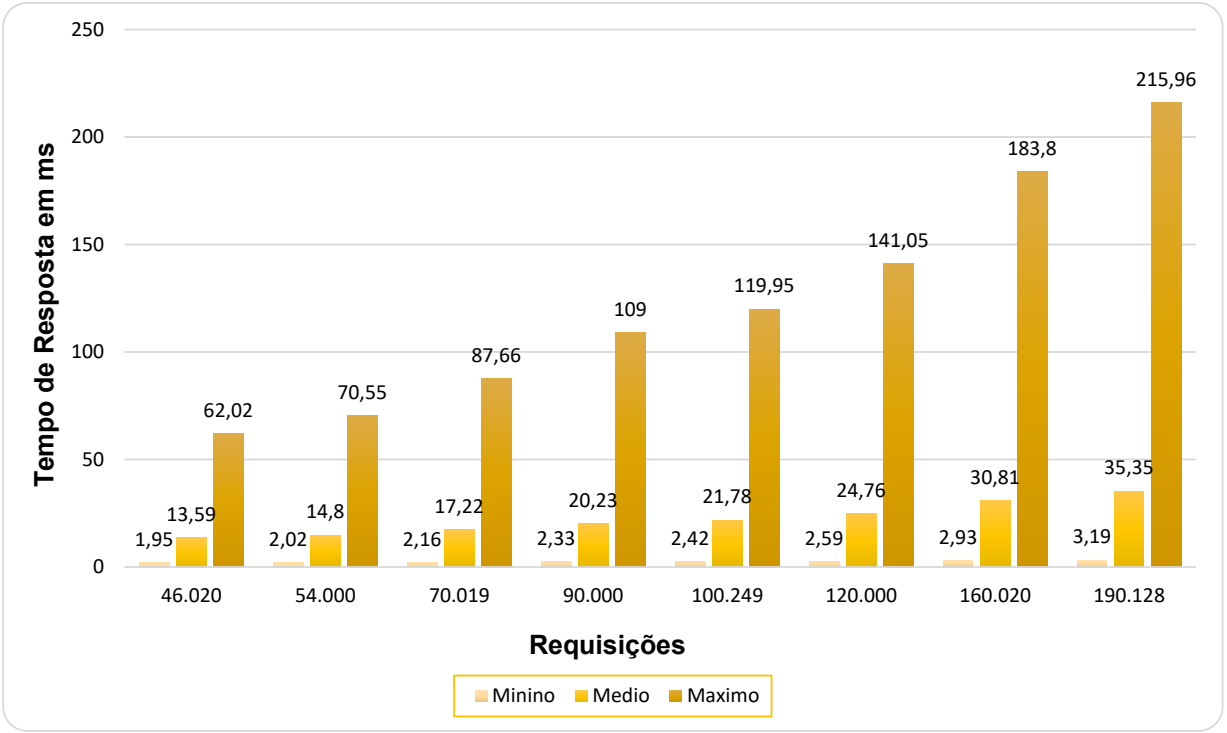

Figura 4.11: Gráfico de Tempos de Resposta em Rede IPv6

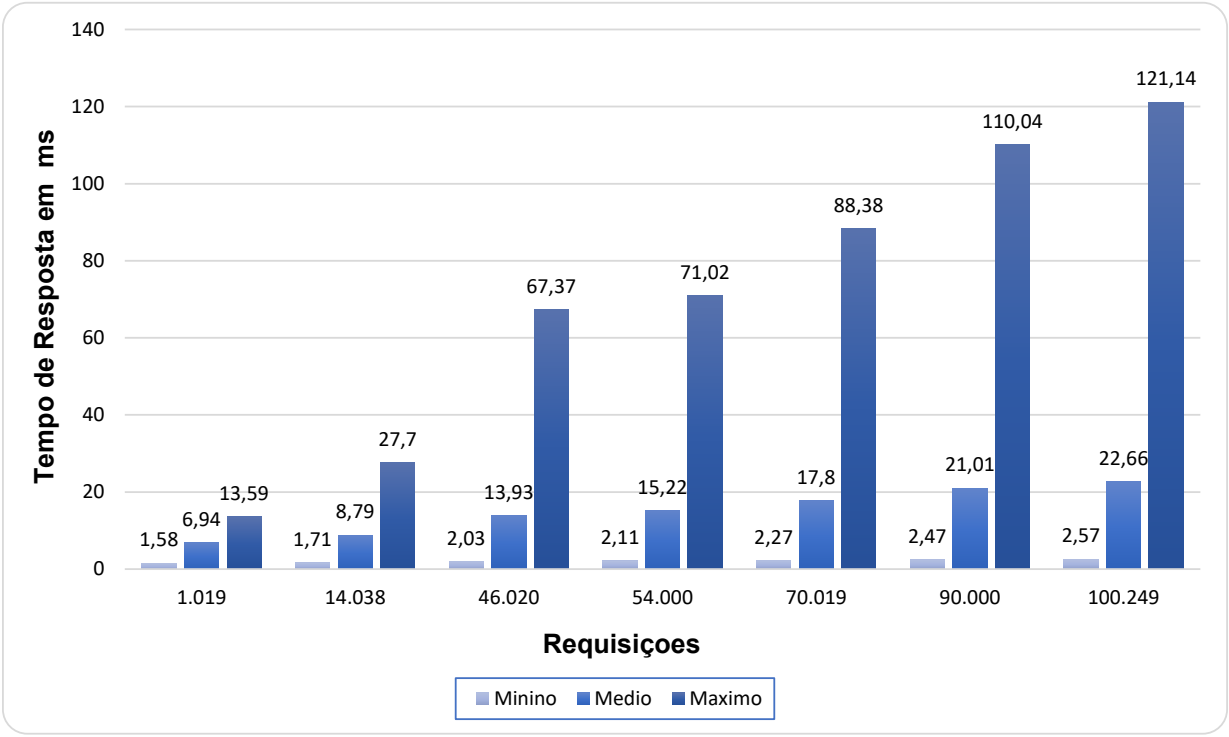

Figura 4.12: Gráfico de Tempos de Resposta em Rede IPv4

4.2.2 Teste de Estresse - Resultados e Análise

Nesta fase foram conduzidos um total de 630 testes de estresse, divididos em 21 grupos, contemplando os protocolos IPv4 e IPv6, cada grupo consistiu em 30 iterações. É importante ressaltar que, na execução do protocolo conforme descrito na seção 3.3.5, não houve imposição de um limite para o descarte de pacotes.

No teste de estresse, foram empregados valores de solicitação de endereço que ultrapassaram os limites estabelecidos no teste anterior. Levando em consideração os resultados superiores obtidos no teste de carga, o protocolo IPv6 iniciou esta fase com um total de 200.040 solicitações de endereço, enquanto o protocolo IPv4 iniciou com 110.040 solicitações, devido ao seu desempenho inferior em relação ao IPv6. No entanto, devido às restrições impostas pelo hardware utilizado, o número máximo de solicitações de endereço foi fixado em 250 mil para ambos os protocolos.

A Figura 4.13 exibe a relação entre quantidade de requisições e o número de pacotes perdidos utilizando IPv4 e IPv6. Para facilitar o entendimento, mostramos na Figura 4.14 os valores percentuais de pacotes perdidos. As figuras ressaltam a marcante disparidade entre os protocolos IPv4 e IPv6. Mesmo durante a fase inicial do teste, na qual o IPv6 é submetido a um número significativamente maior de solicitações em comparação ao IPv4, seus números de descarte de pacotes se mantém abaixo dos apresentados pelo IPv4, mantendo-se assim ate chegar no final do teste onde a diferença entre ambos é ligeiramente aumentada para um diferencial de 10,7% em favor do IPv6.

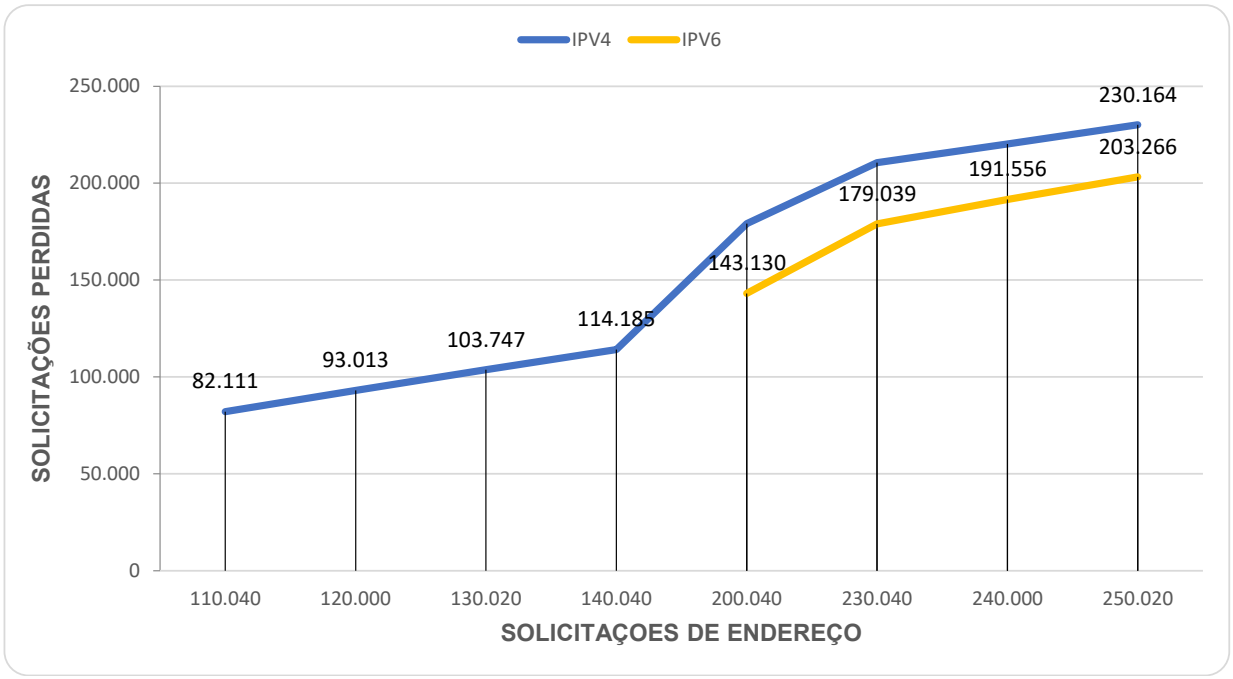

Figura 4.13: Gráfico Comparativo de Quantidade Requisições x Quantidade de Pacotes Perdidos em redes IPv4 e IPv6

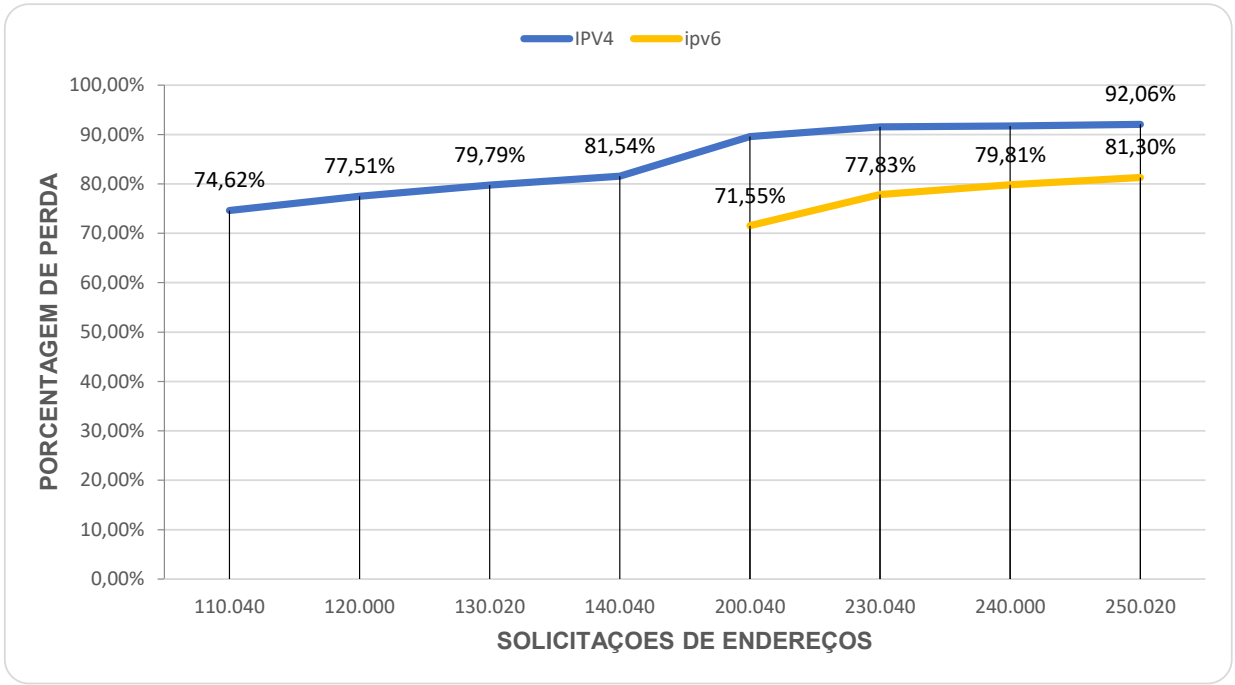

Figura 4.14: Gráfico Comparativo de Quantidade Requisições x Percentual de Pacotes Perdidos em redes IPv4 e IPv6

As Figuras 4.15 e 4.16 fornecem uma visualização dos tempos de resposta dos protocolos durante o teste de estresse. Observa-se que a diferença entre os tempos se manteve relativamente similar, variando de 0,29 a 3,54 milissegundos, com uma leve vantagem para o protocolo IPv6.

Embora essa diferença não impacte de maneira significativa a usabilidade do serviço

DHCP para usuários humanos, uma vez que o tempo médio de resposta para ambos os protocolos foi rápido e dentro dos limites recomendados pela RFC 3315 Carney et al. [2003], ela pode ser considerada relevante para as máquinas que dependem do serviço DHCP.

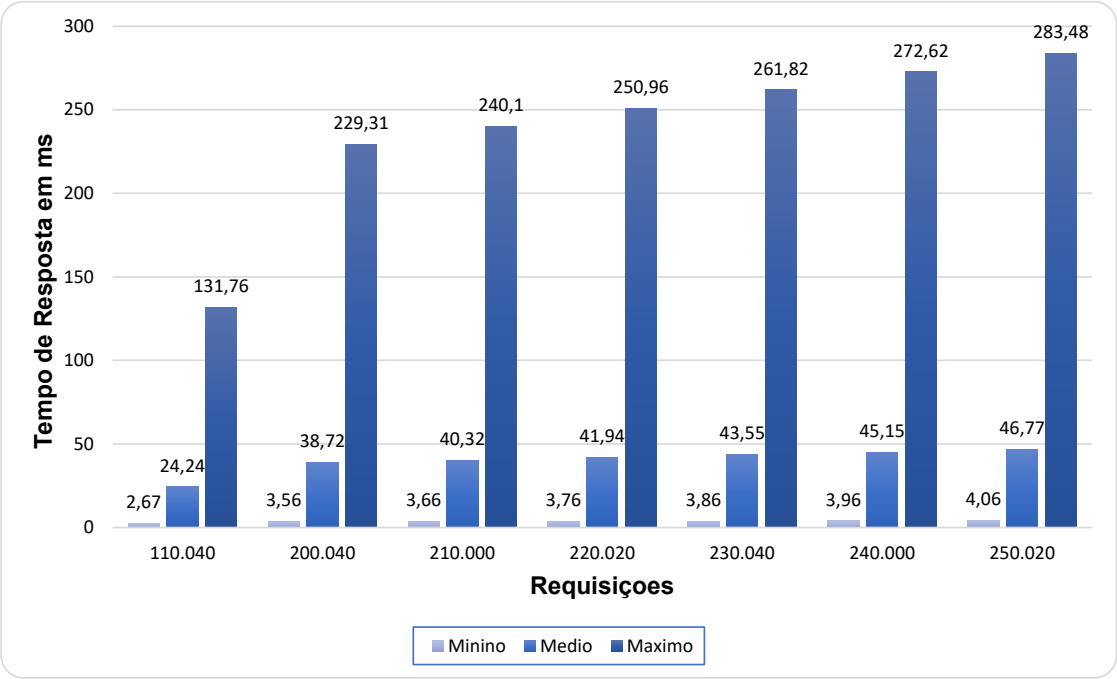

Figura 4.15: Gráfico de Tempos de Resposta em Rede IPv4

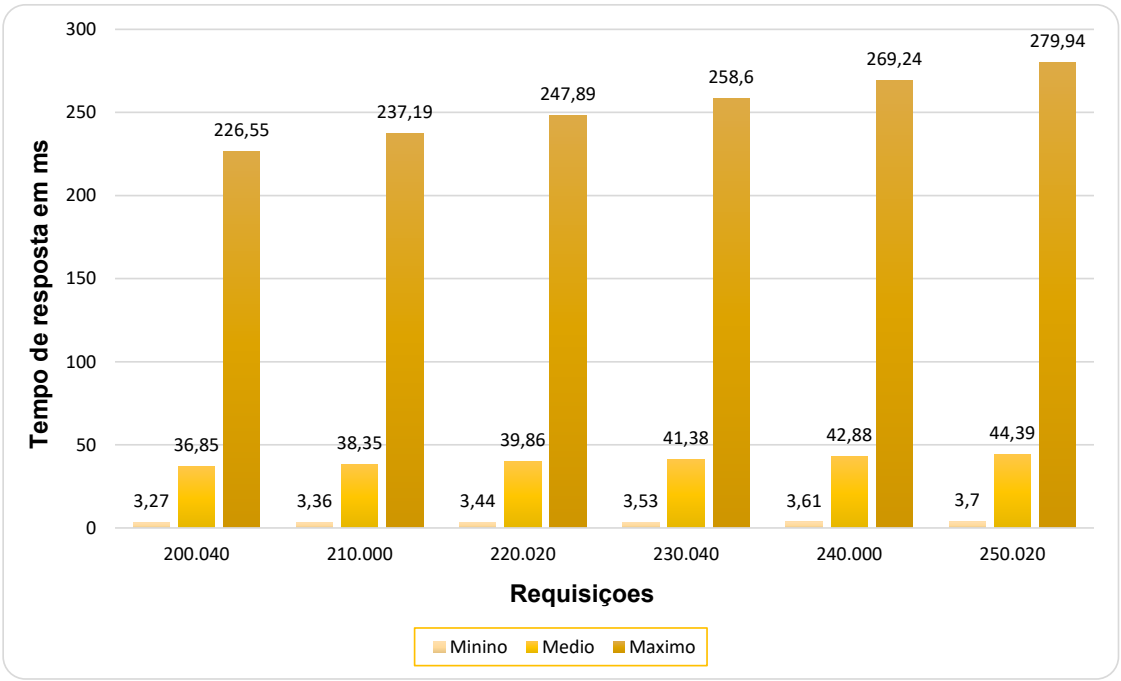

Figura 4.16: Gráfico de Tempos de Resposta em Rede IPv6

Apesar dos benefícios inegáveis oferecidos pelo IPv6 observados nos resultados apresentados, o IPv4 ainda permanece em uso devido a uma série de razões pertinentes.

Em primeiro lugar, a ampla implantação do IPv4 ao longo dos anos resultou em uma infraestrutura de rede consolidada e abrangente, com suporte a diversos dispositivos e sistemas operacionais. A transição completa para o IPv6 exigiria uma complexa e custosa migração, envolvendo atualizações de *hardware*, *software* e amplo treinamento para os profissionais de TI.

Embora o IPv4 apresente limitações em termos de espaço de endereçamento e segurança, soluções como o NAT têm sido adotadas para prolongar a vida útil do IPv4. Mesmo com o compartilhamento de um único endereço IP público entre vários dispositivos na rede local, embora isso reduza a conectividade direta entre os dispositivos, tal abordagem fornece um nível básico de proteção contra ataques cibernéticos.

Com todos esses pontos levantados , o IPv4 continua a ser utilizado devido à sua ampla compatibilidade, infraestrutura consolidada e a implementação de soluções paliativas para superar suas limitações iniciais. Embora o IPv6 seja considerado o futuro promissor da *Internet*, sua transição completa requer um planejamento cuidadoso e complexo, que está sendo realizado gradualmente.

5
Conclusão

Com base na escassez de endereços IPv4 e no contínuo crescimento acelerado da Internet impulsionado pela expansão da conectividade móvel, surgiu a necessidade de um novo protocolo, o IPv6, para atender às demandas da era do IoT. O objetivo deste estudo consistiu em analisar o protocolo IPv6, explorando seus recursos essenciais a fim de avaliar o desempenho da rede e a eficiência do processo de alocação dinâmica de endereços IP.

Com o intuito de alcançar os objetivos propostos, foram adotados softwares de emulação de redes, que além de possuírem baixo custo, destacam-se pela facilidade de utilização. A simulação empreendida por meio do emulador Core buscou reproduzir fielmente o comportamento dos dispositivos em cenários isolados, proporcionando resultados equiparados ou equivalentes àqueles que seriam obtidos mediante o uso de dispositivos físicos reais.

A divisão da simulação em ambientes isolados permitiu a observação e verificação do comportamento do protocolo. Essa abordagem possibilitou uma compreensão mais aprofundada da interação e funcionamento das subfunções intrínsecas ao protocolo IPv6. Os resultados obtidos nos cenários isolados desenvolvidos, em conformidade com as RFCs, demonstraram um comportamento em consonância com as informações fornecidas pelos respectivos autores.

O emulador EVE-NG desempenhou um papel fundamental na avaliação da performance dos protocolos em questão. No entanto, é válido ressaltar que as métricas empregadas não devem ser interpretadas como uma medida absoluta de desempenho. A fim de obter uma avaliação abrangente e precisa da eficiência do serviço DHCP, é crucial empregar um protocolo de teste criteriosamente projetado, que combine diversos indicadores relevantes. Essa abordagem multidimensional possibilitará uma análise mais completa e aprofundada, fornecendo uma visão holística do desempenho dos protocolos IPv4 e IPv6 e suas capacidades de gerenciamento de alocação de endereços IP.

Com base nas métricas estabelecidas e na extensa quantidade de requisições aplicadas nos dois cenários de teste, fica evidente que o IPv6 superou o IPv4 em termos de escalabilidade, o que é a principal justificativa para a substituição dos protocolos. O IPv6 conseguiu lidar com quase o dobro do volume de requisições durante os testes, com uma perda menor de solicitações, o que confere uma melhor qualidade de serviço. Em relação à latência, ambos os protocolos

apresentaram tempos de resposta satisfatórios, não ultrapassando o tempo de resposta imposto. Embora haja uma diferença mínima nos tempos de resposta, o IPv6 mostrou-se superior nesse aspecto.

O desempenho do DHCP é influenciado por uma série de variáveis que devem ser consideradas para uma análise abrangente. No contexto deste estudo, é importante ressaltar que os testes realizados não levaram em conta fatores como a utilização de DNS dinâmicos, mecanismos de prevenção de ataques de rede ou gerenciadores de carga. A ativação desses recursos, bem como outros adicionais, pode potencialmente impactar os resultados apresentados, alterando a dinâmica da alocação de endereços IP e o desempenho geral do serviço DHCP. Portanto, é fundamental reconhecer que o ambiente de implementação do DHCP, incluindo suas configurações e recursos adicionais, pode ter um efeito significativo na análise e nos resultados obtidos.

5.1 Dificuldades encontradas e sugestões para trabalhos futuros

Este estudo é caracterizado por algumas limitações que merecem ser destacadas, entre elas a não utilização de equipamentos reais de rede. No entanto, essas limitações foram superadas por meio da utilização de emuladores, que proporcionaram a condução dos experimentos com a mesma confiabilidade e precisão que seriam obtidos caso equipamentos reais fossem utilizados. Dessa forma, os resultados e conclusões alcançados neste trabalho são respaldados pela eficácia dos emuladores empregados, que garantiram a validade dos experimentos e a fidedignidade dos resultados obtidos.

As restrições impostas pelos recursos computacionais limitaram a capacidade de realizar testes com um volume de requisições superior a 250 mil, assim como de alcançar uma taxa de perda de pacotes de 100% durante o teste de estresse. Além disso, a execução dos testes de desempenho demandou um considerável período de tempo, uma vez que parte do processo ainda depende de intervenção manual. Nesse sentido, uma possível solução para otimizar a eficiência do procedimento seria a automação, proporcionando uma maior agilidade na execução dos testes.

Referências

N. Alcott. *DHCP for Windows 2000: Managing the Dynamic Host Configuration Protocol*. O'Reilly Media, 2001. ISBN 9781491931851. URL https://books.google.com.br/books?id=PL0bCAAAQBAJ.

Araujo, Ediney, Fernando, Teixeira, and Everton. Redes de computadores usando ipv6 com protocolo dhcpv6. B.S. thesis, Universidade Tecnológica Federal do Paraná, 2014. URL https://repositorio.utfpr.edu.br/jspui/bitstream/1/9788/3/CT_COTEL_2014_2_03.pdf.

Dr. Steve E. Deering Bob Hinden. Internet Protocol, Version 6 (IPv6) Specification. RFC 2460, December 1998. URL https://www.rfc-editor.org/info/rfc2460.

Samuel Henrique Bucke Brito. *IPv6-O novo protocolo da Internet*. Novatec Editora, 2018. URL https://books.google.com.br/books/about/IPv6_O_Novo_Protocolo_da_Internet.html?id=1hlQAwAAQBAJ&redir_esc=y.

Angela Maria Duran Bugallo, Marcio Almeida BARROS, and Waldeck Ribeiro TORRES. Introdução ao dhcp. *Rev. Consultada*, 3(6):13, 2007. URL https://memoria.rnp.br/newsgen/9911/dhcp.html.

Michael Carney, Charles E. Perkins, Bernie Volz, Ted Lemon, and Jim Bound. Dynamic Host Configuration Protocol for IPv6 (DHCPv6). RFC 3315, July 2003. URL https://www.rfc-editor.org/info/rfc3315.

D. Comer. *Internetworking with TCP/IP: Principles, protocols, and architecture*. Internetworking with TCP/IP. Pearson Prentice Hall, 2006. ISBN 9780131876712. URL https://books.google.com.br/books?id=jonyuTASbWAC.

D. Comer. *Interligação de Redes com TCP/IP –: Princípios, Protocolos e Arquitetura*. Number v. 1. Elsevier Editora Ltda., 2016. ISBN 9788535278644. URL https://books.google.com.br/books?id=F1_jBwAAQBAJ.

J. Davies. *Understanding IPv6*. Pro-One-Offs Series. Microsoft Press, 2003. ISBN 9780735612457. URL https://books.google.com.br/books?id=2P9vPgAACAAJ.

Bob Hinden Dr. Steve E. Deering. IP Version 6 Addressing Architecture. RFC 4291, February 2006. URL https://www.rfc-editor.org/info/rfc4291.

Bob Hinden Dr. Steve E. Deering. Internet Protocol, Version 6 (IPv6) Specification. RFC 8200, July 2017. URL https://www.rfc-editor.org/info/rfc8200.

Ralph Droms. Dynamic Host Configuration Protocol. RFC 2131, March 1997. URL https://www.rfc-editor.org/info/rfc2131.

Ralph Droms. Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6. RFC 3736, April 2004. URL https://www.rfc-editor.org/info/rfc3736.

Lemon T Droms R. *The DHCP Handbook*. Kaleidoscope Series. Sams, 2003. ISBN 9780672323270. URL https://books.google.com.br/books?id=8acohIuwp6QC.

Stevens W.R Fall K.R. *TCP/IP Illustrated, Volume 1: The Protocols*. Addison-Wesley Professional Computing Series. Pearson Education, 2011. ISBN 9780132808187. URL https://books.google.com.br/books?id=a23OAn5i8R0C.

Carlos Pignataro Fernando Gont. Formally Deprecating Some ICMPv4 Message Types. RFC 6918, April 2013. URL https://www.rfc-editor.org/info/rfc6918.

Pollyana Ferrari. *Jornalismo digital*. Editora Contexto, 2007. URL https://books.google.com.br/books/about/Jornalismo_digital.html?id=GthnAwAAQBAJ&redir_esc=y.

B.A. Forouzan. *TCP/IP Protocol Suite*. McGraw-Hill Forouzan Sreries. McGraw-Hill, 2003. ISBN 9780072460605. URL https://books.google.com.br/books?id=HsCjH_V04tUC.

Fegan S.C. Forouzan B.A. *Data Communications and Networking*. Data Communications and Networking. McGraw-Hill Higher Education, 2007. ISBN 9780072967753. URL https://books.google.com.br/books?id=bwUNZvJbEeQC.

Fegan S.C. Forouzan B.A. *Protocolo TCP/IP - 3.ed.* McGraw Hill Brasil, 2009. ISBN 9788563308689. URL https://books.google.com.br/books?id=fNvIgp3kkyQC.

Karen Goethals, Antónia Aguiar, and Eugénia Almeida. História da internet. *Faculdade de Engenharia da Universidade do Porto, Mestrado em Gestão da Informação*, 2000. URL https://web.fe.up.pt/~mgi99022/goii/M1/final.doc.

Carsten Heinisch. 1956: Primeiro cabo telefônico através do atlântico, September 2019. URL https://www.dw.com/pt-br/1956-primeiro-cabo-telef%C3%B4nico-atrav%C3%A9s-do-atl%C3%A2ntico/a-268244. cabo telefônico.

C. Hunt. *TCP/IP Network Administration*. Nutshell handbook. O'Reilly Media, 2002. ISBN 9780596002978. URL https://books.google.com.br/books?id=wqabAgAAQBAJ.

ipv6.br. Endereçamento, may 2012. URL https://ipv6.br/post/enderecamento/. Endereçamento ipv6.

Pyda Srisuresh Kjeld Borch Egevang. Traditional IP Network Address Translator (Traditional NAT). RFC 3022, January 2001. URL https://www.rfc-editor.org/info/rfc3022.

K.W. Kurose, J.F.and Ross. *Computer Networking: A Top-down Approach*. Pearson, 2017. ISBN 9780133594140. URL https://books.google.com.br/books?id=OljpOAAACAAJ.

Pete Loshin. Ipv6: Theory, protocol, and practice. 2004. URL https://books.google.com.br/books/about/IPv6.html?id=6JDuPUzMU4AC&redir_esc=y.

Philip Matthews, Iljitsch van Beijnum, and Marcelo Bagnulo. Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers. RFC 6146, April 2011. URL https://www.rfc-editor.org/info/rfc6146.

S. McFarland, M. Sambi, N. Sharma, and S. Hooda. *IPv6 for Enterprise Networks*. Networking Technology. Pearson Education, 2011. ISBN 9781587142314. URL https://books.google.com.br/books?id=e0_9osAeYgEC.

Nick Moore. Optimistic Duplicate Address Detection (DAD) for IPv6. RFC 4429, April 2006. URL https://www.rfc-editor.org/info/rfc4429.

Alex Conta Mukesh Gupta. Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification. RFC 4443, March 2006. URL https://www.rfc-editor.org/info/rfc4443.

Dr. Thomas Narten, Tatsuya Jinmei, and Dr. Susan Thomson. IPv6 Stateless Address Autoconfiguration. RFC 4862, September 2007a. URL https://www.rfc-editor.org/info/rfc4862.

Thomas Narten, Erik Nordmark, William Simpson, and Hesham Soliman. Neighbor discovery for ip version 6 (ipv6). Technical report, 2007b. URL https://www.rfc-editor.org/info/rfc1970.

Raissa Monego Pedrozo. Implantação de uma rede utilizando os padrões do protocolo ipv6. *Universidade Federal de Santa Maria, Santa Maria, RS*, 2014. URL https://www.ufsm.br/app/uploads/sites/495/2019/05/2014-Raissa_Monego.pdf.

R. Perlman. *Interconnections: Bridges, Routers, Switches, and Internetworking Protocols*. Addison-Wesley professional computing series. Addison Wesley, 2000. ISBN 9780201634488. URL https://books.google.com.br/books?id=AIRitf5C-QQC.

JB Pinho. Jornalismo na internet: planejamento e produção da informação on-line. são paulo: Summus, 2003. _. *Comunicação organizacional*, 2018. URL https://books.google.com.br/books?id=TtRD6VjRWBEC&redir_esc=y.

J. Postel. Internet Control Message Protocol. RFC 792, September 1981. URL https://www.rfc-editor.org/info/rfc792.

Ligia Maria Ribeiro. A historia da internet, jun 1998. URL https://paginas.fe.up.pt/~mgi97018/historia.html.

Luciano Santana dos Santos. Implementação de ipv6 em um provedor de internet. B.S. thesis, Universidade Tecnológica Federal do Paraná, 2016. URL https://riut.utfpr.edu.br/jspui/handle/1/16854.

André Manoel Silveira. Rede ipv6 com integração ipv4. *Centro Federal de Educação Tecnologia, São José, SC*, 2012. URL https://wiki.sj.ifsc.edu.br/images/e/e3/TCC_AndreManoeldaSilveira.pdf.

William A. Simpson, Dr. Thomas Narten, Erik Nordmark, and Hesham Soliman. Neighbor Discovery for IP version 6 (IPv6). RFC 4861, September 2007. URL https://www.rfc-editor.org/info/rfc4861.

Apêndice

A

Comandos Cisco IOS

```
1 !Configuração da interface do Roteador
2 !Configuração do Roteador - DHCPv4
3 !Modo EXEC Privilegiado
4 enable
5 !Modo de Configuração Global
6 configure terminal
7 !Configurando a Interface Física do roteador
8 interface fastEthernet 0/0
9 !Descrição da Interface Física do roteador
10 description Interface dhcpv4
11 !Endereçamento IPv4 da Interface Física do roteador
12 ip address 10.1.1.1 255.255.255.0
13 !Habilitando Interface Física do Router
14 no shutdown
15 !voltando ao configure terminal
16 exit
17 !criando o pool DHCPv4
18 ip dhcp pool redeipv4
19 !indicando a rede a ser usada pelo DHCPV4
20 network 10.1.1.1 255.255.255.0
21 !Gateway da rede
22 default-router 10.1.1.1
23 !servidor dns da rede
24 dns-server 8.8.8.8
25 !saindo do configure terminal
26 end
27 !Salvando as configurações
28 copy running-config startup-config
```

```
29 !Configuração da interface do Roteador
30 !Configuração do Roteador - DHCPv6
31 !Modo EXEC Privilegiado
32 enable
33 !habilitando Roteamento ipv6
34 ipv6 unicast-routing
35 !Modo de Configuração Global
36 configure terminal
37 !criando o pool DHCPv6
38 ipv6 dhcp pool redeipv6
39 !Atribuindo prefixo a o pool DHCPv6
40 address prefix 2001:db8:acad:1::/64
41 !servidor dns da rede
42 dns-server 2001:4860:4860::8888
43 !determinando nome de Domínio
44 domain-name simulacao.cisco
45 !voltando um nível
46 exit
47 !Configurando a Interface Física do roteador
48 interface fastEthernet 0/0
49 !Descrição da Interface Física do roteador
50 description Interface dhcpv6
51 !Endereçamento IPv6 de link-local
52 ipv6 address fe80::1 link-local
53 !Endereçamento IPv6 de Global unicast
54 ipv6 address 2001:db8:acad:1::1/64
55 !Alterando o sinalizador M de 0 para 1
56 ipv6 nd managed-confg-flag
57 !Alterando o sinalizador A de 0 para 1
58 ipv6 nd prefix default no-autoconfig
59 !vinculando o pool DHCPV6 a interface
60 ipv6 dhcp server redeipv6
61 !Habilitando Interface Física do Router
62 no shutdown
63 !saindo do configure terminal
64 end
65 !Salvando as configurações
66 copy running-config startup-config
```

B

Dados dos testes

Teste de Carga - IPV4

Requisições	1019		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	0	0,00	0,00
Tempo Mínimo	1,58	0,00	0,05
Tempo Médio	6,94	0,14	0,38
Tempo Máximo	13,59	0,75	0,86
Porcentagem de Perda		0,00%	

Requisições	2039		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	0	0,00	0,00
Tempo Mínimo	1,59	0,32	0,57
Tempo Médio	6,97	0,49	0,70
Tempo Máximo	14,70	0,79	0,89
Porcentagem de Perda		0,00%	

Requisições	3059		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	0	0,18	0,42
Tempo Mínimo	1,60	0,44	0,67
Tempo Médio	7,02	0,73	0,86
Tempo Máximo	15,78	0,40	0,63
Porcentagem de Perda		0,01%	

Requisições	4019		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	1	0,23	0,48
Tempo Mínimo	1,61	0,22	0,47
Tempo Médio	7,17	0,49	0,70
Tempo Máximo	16,84	1,17	1,08
Porcentagem de Perda		0,01%	

Requisições	8040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	1	0,14	0,38
Tempo Mínimo	1,65	0,00	0,02
Tempo Médio	7,82	0,00	0,03
Tempo Máximo	21,20	2,64	1,62
Porcentagem de Perda		0,02%	

Requisições	14038		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	4	0,07	0,26
Tempo Mínimo	1,71	0,00	0,02
Tempo Médio	8,79	0,01	0,09
Tempo Máximo	27,70	0,13	0,36
Porcentagem de Perda		0,03%	

Requisições	28019		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	8	2,74	1,65
Tempo Mínimo	1,85	0,00	0,00
Tempo Médio	11,04	0,00	0,04
Tempo Máximo	42,86	0,48	0,69
Porcentagem de Perda		0,03%	

Requisições	41999		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	21	10,07	3,17
Tempo Mínimo	1,99	0,00	0,01
Tempo Médio	13,29	0,06	0,24
Tempo Máximo	58,01	0,62	0,79
Porcentagem de Perda		0,05%	

Requisições	44038		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	25	4,16	2,04
Tempo Mínimo	2,01	0,00	0,01
Tempo Médio	13,61	0,11	0,32
Tempo Máximo	60,22	0,25	0,50
Porcentagem de Perda		0,06%	

Requisições	46020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	493	22034	148,44
Tempo Mínimo	2,03	0,01	0,10
Tempo Médio	13,93	0,60	0,77
Tempo Máximo	62,37	116	10,77
Porcentagem de Perda		1,07%	

Requisições	48000		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	2650	60751	246,48
Tempo Mínimo	2,05	0,10	0,31
Tempo Médio	14,25	0,11	0,33
Tempo Máximo	64,51	212	14,55
Porcentagem de Perda		5,52%	

Requisições	50040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	2916	124501	352,85
Tempo Mínimo	2,06	0,03	0,18
Tempo Médio	14,49	0,01	0,10
Tempo Máximo	66,14	5,76	2,40
Porcentagem de Perda		5,89%	

Requisições	52020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	7039	1621434	1273,36
Tempo Mínimo	2,09	0,00	0,04
Tempo Médio	14,90	0,04	0,19
Tempo Máximo	68,87	14,90	3,86
Porcentagem de Perda		13,53%	

Requisições	54000		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	10004	2258574	1502,86
Tempo Mínimo	2,11	0,05	0,22
Tempo Médio	15,22	0,04	0,21
Tempo Máximo	71,02	11,58	3,40
Porcentagem de Perda		18,53%	

Requisições	56039		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	13871	2930334	1711,82
Tempo Mínimo	2,13	0,10	0,32
Tempo Médio	15,55	0,67	0,82
Tempo Máximo	73,23	63,80	7,99
Porcentagem de Perda		24,75%	

Requisições	58020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	15955	2930334	1711,82
Tempo Mínimo	2,15	0,10	0,32
Tempo Médio	15,86	0,67	0,82
Tempo Máximo	75,37	63,80	7,99
Porcentagem de Perda		27,50%	

Requisições	60059		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	18017	2698022	1642,57
Tempo Mínimo	2,17	0,50	0,71
Tempo Médio	16,19	0,01	0,10
Tempo Máximo	77,58	24,31	4,93
Porcentagem de Perda		30,00%	

Requisições	65040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	26409	3691890	1921,43
Tempo Mínimo	2,22	0,08	0,28
Tempo Médio	16,99	0,22	0,47
Tempo Máximo	82,98	39,21	6,26
Porcentagem de Perda		40,60%	

Requisições	70019		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	33099	9778851	3127,12
Tempo Mínimo	2,27	0,01	0,10
Tempo Médio	17,80	0,01	0,08
Tempo Máximo	88,38	30,58	5,53
Porcentagem de Perda		47,27%	

Requisições	80040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	44736	19354697	4399,40
Tempo Mínimo	2,37	1,41	1,19
Tempo Médio	19,41	0,00	0,02
Tempo Máximo	99,24	2,04	1,43
Porcentagem de Perda		55,89%	

Requisições	90000		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	57561	42039873	6483,82
Tempo Mínimo	2,47	0,01	0,10
Tempo Médio	21,01	0,00	0,01
Tempo Máximo	110,04	0,71	0,85
Porcentagem de Perda		63,96%	

Requisições	100020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	69854	72189644	8496,45
Tempo Mínimo	2,57	0,21	0,46
Tempo Médio	22,62	0,01	0,09
Tempo Máximo	120,90	0,10	0,31
Porcentagem de Perda		69,84%	

Requisições	100249		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	70174	2930334	1711,82
Tempo Mínimo	2,57	0,10	0,32
Tempo Médio	22,66	0,67	0,82
Tempo Máximo	121,14	63,80	7,99
Porcentagem de Perda		70,00%	

Teste de Stress - IPV4

Requisições	110040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	82111	118174023	10870,79
Tempo Mínimo	2,67	0,01	0,12
Tempo Médio	24,24	0,03	0,18
Tempo Máximo	131,76	2,09	1,45
Porcentagem de Perda		74,62%	

Requisições	120000		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	93013	216083499	14699,78
Tempo Mínimo	2,77	0,12	0,34
Tempo Médio	25,84	0,03	0,17
Tempo Máximo	142,55	1,57	1,25
Porcentagem de Perda		77,51%	

Requisições	130020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	103747	542442392	23290,39
Tempo Mínimo	2,87	0,76	0,87
Tempo Médio	27,45	0,04	0,20
Tempo Máximo	153,41	659	25,66
Porcentagem de Perda		79,79%	

Requisições	140040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	114185	438663118	20944,29
Tempo Mínimo	2,97	3,35	1,83
Tempo Médio	29,06	0,19	0,44
Tempo Máximo	164,27	8,81	2,97
Porcentagem de Perda		81,54%	

Requisições	150060		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	125953	710042494	26646,62
Tempo Mínimo	3,07	0,57	0,75
Tempo Médio	30,68	0,00	0,06
Tempo Máximo	175,13	10,95	3,31
Porcentagem de Perda		83,94%	

Requisições	160020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	136401	593224328	24356,20
Tempo Mínimo	3,17	0,17	0,41
Tempo Médio	32,28	24,78	4,98
Tempo Máximo	185,93	1,57	1,25
Porcentagem de Perda		85,24%	

Requisições	170040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	147147	1142188114	33796,27
Tempo Mínimo	3,27	0,39	0,62
Tempo Médio	33,89	0,01	0,12
Tempo Máximo	196,79	0,75	0,86
Porcentagem de Perda		86,54%	

Requisições	180000		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	157865	1457931017	38182,86
Tempo Mínimo	3,36	0,10	0,31
Tempo Médio	35,50	0,01	0,10
Tempo Máximo	207,59	87,12	9,33
Porcentagem de Perda		87,70%	

Requisições	190020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	168632	1938753557	44031,28
Tempo Mínimo	3,46	0,00	0,05
Tempo Médio	37,11	0,04	0,21
Tempo Máximo	218,45	2,61	1,61
Porcentagem de Perda		88,74%	

Requisições	200040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	179195	2342426792	48398,62
Tempo Mínimo	3,56	0,15	0,39
Tempo Médio	38,72	0,00	0,03
Tempo Máximo	229,31	5,04	2,24
Porcentagem de Perda		89,58%	

Requisições	210000		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	188580	2119441854	46037,40
Tempo Mínimo	3,66	0,91	0,95
Tempo Médio	40,32	0,00	0,05
Tempo Máximo	240,10	83,70	9,15
Porcentagem de Perda		89,80%	

Requisições	220020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	198954	3360405386	57969,00
Tempo Mínimo	3,76	1,92	1,39
Tempo Médio	41,94	0,09	0,29
Tempo Máximo	250,96	3,66	1,91
Porcentagem de Perda		90,43%	

Requisições	230040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	210587	4100621673	64036,10
Tempo Mínimo	3,86	1,20	1,09
Tempo Médio	43,55	0,12	0,35
Tempo Máximo	261,82	17,19	4,15
Porcentagem de Perda		91,54%	

Requisições	240000		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	220196	4726912444	68753
Tempo Mínimo	3,96	1,28	1,13
Tempo Médio	45,15	0,01	0,12
Tempo Máximo	272,62	61,23	7,83
Porcentagem de Perda		91,75%	

Requisições	250020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	230164	5534009848	74390,93
Tempo Mínimo	4,06	1,33	1,15
Tempo Médio	46,77	0,15	0,39
Tempo Máximo	283,48	1,19	1,09
Porcentagem de Perda		92,06%	

Teste de Carga - IPV6

Requisições	1019		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	0	0,21	0,46
Tempo Mínimo	1,48	0,00	0,06
Tempo Médio	6,80	0,01	0,08
Tempo Máximo	13,95	0,71	0,84
Porcentagem de Perda		0,00%	

Requisições	2039		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	0	0,00	0,00
Tempo Mínimo	1,58	0,01	0,11
Tempo Médio	6,95	0,01	0,10
Tempo Máximo	15,04	1,24	1,11
Porcentagem de Perda		0,00%	

Requisições	3059		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	0	0,00	0,00
Tempo Mínimo	1,59	0,02	0,15
Tempo Médio	7,11	0,01	0,09
Tempo Máximo	16,13	1,46	1,21
Porcentagem de Perda		0,00%	

Requisições	4019		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	0	0,00	0,00
Tempo Mínimo	1,60	0,01	0,11
Tempo Médio	7,25	0,00	0,07
Tempo Máximo	17,15	0,67	0,82
Porcentagem de Perda		0,00%	

Requisições	8040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	2	0,71	0,84
Tempo Mínimo	1,63	0,00	0,04
Tempo Médio	7,86	0,17	0,41
Tempo Máximo	21,45	0,73	0,85
Porcentagem de Perda		0,02%	

Requisições	14038		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	4	0,45	0,67
Tempo Mínimo	1,68	0,00	0,04
Tempo Médio	8,77	0,02	0,14
Tempo Máximo	27,86	0,27	0,52
Porcentagem de Perda		0,03%	

Requisições	28019		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	8	0,35	0,59
Tempo Mínimo	1,80	0,00	0,05
Tempo Médio	10,88	0,13	0,36
Tempo Máximo	42,79	1,41	1,19
Porcentagem de Perda		0,03%	

Requisições	41999		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	12	3,79	1,95
Tempo Mínimo	1,92	0,00	0,04
Tempo Médio	12,99	0,00	0,06
Tempo Máximo	57,73	0,40	0,63
Porcentagem de Perda		0,03%	

Requisições	44038		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	15	1,90	1,38
Tempo Mínimo	1,94	0,01	0,08
Tempo Médio	13,29	0,05	0,21
Tempo Máximo	59,90	0,48	0,69
Porcentagem de Perda		0,03%	

Requisições	46020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	15	3,96	1,99
Tempo Mínimo	1,95	0,01	0,10
Tempo Médio	13,59	0,08	0,29
Tempo Máximo	62,02	0,91	0,95
Porcentagem de Perda		0,03%	

Requisições	48000		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	29	75,95	8,71
Tempo Mínimo	1,97	0,02	0,13
Tempo Médio	13,89	1,08	1,04
Tempo Máximo	64,14	0,95	0,97
Porcentagem de Perda		0,06%	

Requisições	50040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	50	48,00	6,93
Tempo Mínimo	1,98	0,05	0,22
Tempo Médio	14,12	0,01	0,10
Tempo Máximo	65,74	1,70	1,30
Porcentagem de Perda		0,10%	

Requisições	52020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	79	10,00	3,16
Tempo Mínimo	2,01	0,00	0,05
Tempo Médio	14,50	0,01	0,12
Tempo Máximo	68,43	0,92	0,96
Porcentagem de Perda		0,14%	

Requisições	54000		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	135	193,54	13,91
Tempo Mínimo	2,02	0,00	0,05
Tempo Médio	14,80	1,81	1,34
Tempo Máximo	70,55	306,66	17,51
Porcentagem de Perda		0,25%	

Requisições	56039		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	172	215,74	14,69
Tempo Mínimo	2,04	0,01	0,10
Tempo Médio	15,11	0,01	0,10
Tempo Máximo	72,72	62,55	7,91
Porcentagem de Perda		0,32%	

Requisições	58020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	250	215,74	14,69
Tempo Mínimo	2,06	0,01	0,10
Tempo Médio	15,41	0,01	0,10
Tempo Máximo	74,84	62,55	7,91
Porcentagem de Perda		0,44%	

Requisições	60059		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	354	2,48	1,58
Tempo Mínimo	2,07	0,03	0,17
Tempo Médio	15,71	0,07	0,27
Tempo Máximo	77,02	78,21	8,84
Porcentagem de Perda		0,59%	

Requisições	65040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	488	2350,79	48,48
Tempo Mínimo	2,12	0,19	0,44
Tempo Médio	16,47	0,09	0,31
Tempo Máximo	82,34	26,44	5,14
Porcentagem de Perda		0,75%	

Requisições	70019		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	1064	208,33	14,43
Tempo Mínimo	2,16	0,03	0,16
Tempo Médio	17,22	0,18	0,42
Tempo Máximo	87,66	58,75	7,67
Porcentagem de Perda		1,52%	

Requisições	80040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	2236	67729,62	260,25
Tempo Mínimo	2,25	0,04	0,20
Tempo Médio	18,73	0,17	0,41
Tempo Máximo	98,36	9,94	3,15
Porcentagem de Perda		2,79%	

Requisições	90000		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	16635	3677203	1917,60
Tempo Mínimo	2,33	0,01	0,09
Tempo Médio	20,23	0,14	0,37
Tempo Máximo	109,00	8,08	2,84
Porcentagem de Perda		18,48%	

Requisições	100020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	28295	13198179	3632,93
Tempo Mínimo	2,42	0,09	0,30
Tempo Médio	21,75	0,09	0,29
Tempo Máximo	119,71	3,30	1,82
Porcentagem de Perda		28,29%	

Requisições	100249		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	28421	27000351	5196,19
Tempo Mínimo	2,42	0,87	0,93
Tempo Médio	21,78	0,25	0,50
Tempo Máximo	119,95	13,79	3,71
Porcentagem de Perda		28,35%	

Requisições	110040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	40566	27000351	5196,19
Tempo Mínimo	2,50	0,08	0,28
Tempo Médio	23,26	0,24	0,49
Tempo Máximo	130,41	10,18	3,19
Porcentagem de Perda		36,86%	

Requisições	120000		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	54292	63507238	7969,14
Tempo Mínimo	2,59	1,13	1,06
Tempo Médio	24,76	0,06	0,24
Tempo Máximo	141,05	2,78	1,67
Porcentagem de Perda		45,24%	

Requisições	130020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	69213	189212264	13755,44
Tempo Mínimo	2,67	0,13	0,35
Tempo Médio	26,28	0,07	0,27
Tempo Máximo	151,75	1,17	1,08
Porcentagem de Perda		53,23%	

Requisições	140040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	80534	127386426	11286,56
Tempo Mínimo	2,76	0,56	0,75
Tempo Médio	27,79	0,03	0,18
Tempo Máximo	162,46	4,41	2,10
Porcentagem de Perda		57,51%	

Requisições	150060		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	92038	196384523	14013,73
Tempo Mínimo	2,84	0,09	0,31
Tempo Médio	29,30	0,28	0,53
Tempo Máximo	173,16	5,19	2,28
Porcentagem de Perda		61,33%	

Requisições	160020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	101613	392090361	19801,27
Tempo Mínimo	2,93	1,97	1,40
Tempo Médio	30,81	0,50	0,71
Tempo Máximo	183,80	3,71	1,93
Porcentagem de Perda		63,50%	

Requisições	170040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	111314	427337803	20672,15
Tempo Mínimo	3,01	0,75	0,86
Tempo Médio	32,32	0,10	0,31
Tempo Máximo	194,50	11,05	3,32
Porcentagem de Perda		65,46%	

Requisições	180000		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	120297	587458235	24237,54
Tempo Mínimo	3,10	0,33	0,58
Tempo Médio	33,82	0,25	0,50
Tempo Máximo	205,14	20,08	4,48
Porcentagem de Perda		66,83%	

Requisições	190020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	132929	913861456	30230,14
Tempo Mínimo	3,19	0,87	0,93
Tempo Médio	35,33	0,25	0,50
Tempo Máximo	215,85	13,79	3,71
Porcentagem de Perda		69,96%	

Requisições	190128		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	133090	1141991973	33793,37
Tempo Mínimo	3,19	0,11	0,33
Tempo Médio	35,35	0,31	0,56
Tempo Máximo	215,96	7,78	2,79
Porcentagem de Perda		70,00%	

Teste de Stress - IPV6

Requisições	200040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	143130	1141991973	33793,37
Tempo Mínimo	3,27	0,11	0,33
Tempo Médio	36,85	0,31	0,56
Tempo Máximo	226,55	7,78	2,79
Porcentagem de Perda		71,55%	

Requisições	210000		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	155237	1410824364	37560,94
Tempo Mínimo	3,36	0,07	0,27
Tempo Médio	38,35	0,54	0,73
Tempo Máximo	237,19	2,48	1,57
Porcentagem de Perda		73,92%	

Requisições	220020		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	169142	1806421856	42502,02
Tempo Mínimo	3,44	0,26	0,51
Tempo Médio	39,86	0,61	0,78
Tempo Máximo	247,89	13,37	3,66
Porcentagem de Perda		76,88%	

Requisições	230040		
	Média	Variância	Desvio Padrão
Pacotes Perdidos	179039	1927227178	43900,20
Tempo Mínimo	3,53	0,12	0,34
Tempo Médio	41,38	0,21	0,46
Tempo Máximo	258,60	5,09	2,26
Porcentagem de Perda		77,83%	

Requisições	240000			Requisições	250020		
	Média	Variância	Desvio Padrão		Média	Variância	Desvio Padrão
Pacotes Perdidos	191556	2302823113	47987,74	Pacotes Perdidos	203266	2777877450	52705,57
Tempo Mínimo	3,61	0,33	0,58	Tempo Mínimo	3,70	0,05	0,23
Tempo Médio	42,88	0,44	0,66	Tempo Médio	44,39	0,86	0,93
Tempo Máximo	269,24	1,44	1,20	Tempo Máximo	279,94	7,80	2,79
Porcentagem de Perda		79,81%		Porcentagem de Perda		81,30%	

C

Captura de Pacotes

C.1 Pacotes capturados Cenário 1

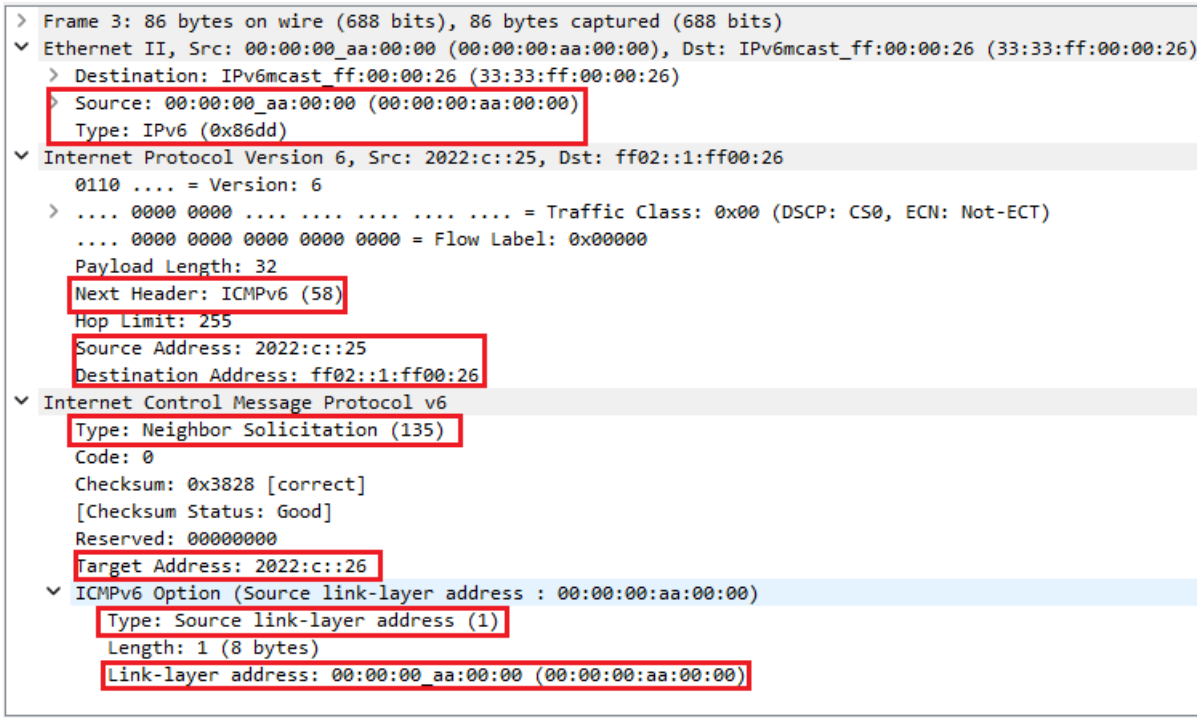

Figura C.1: Cenário 1 - Mensagem NS

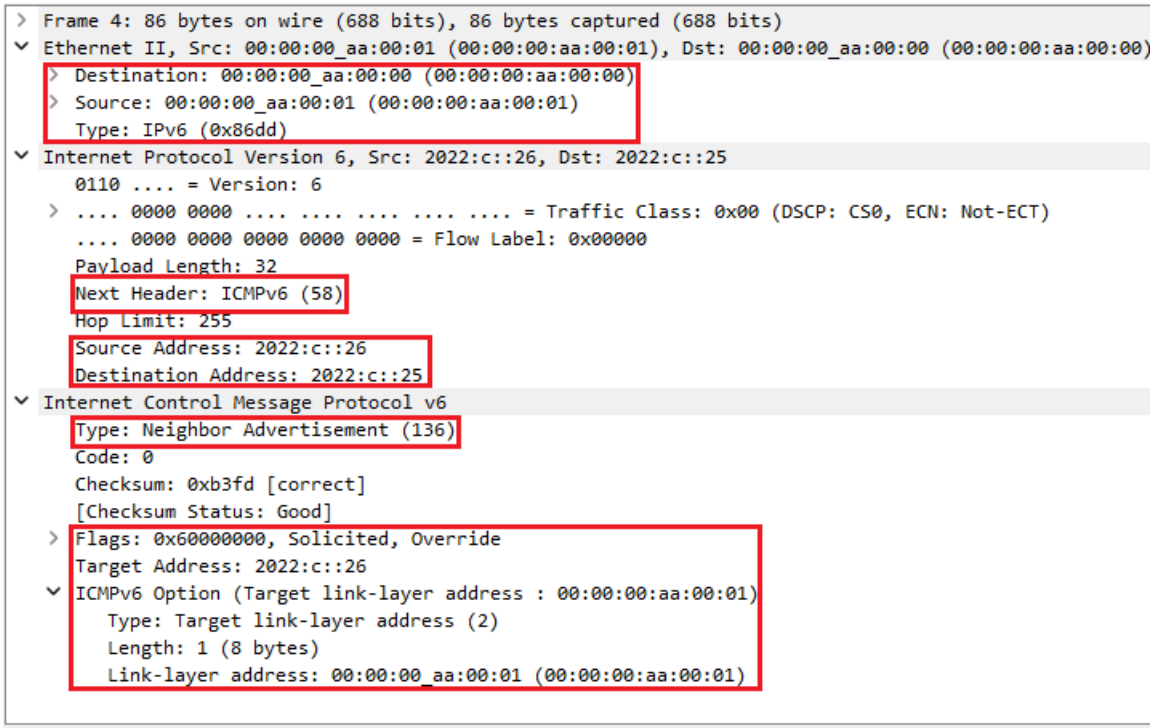

Figura C.2: Cenário 1 - Mensagem NA

C.2 Pacotes capturados Cenário 2

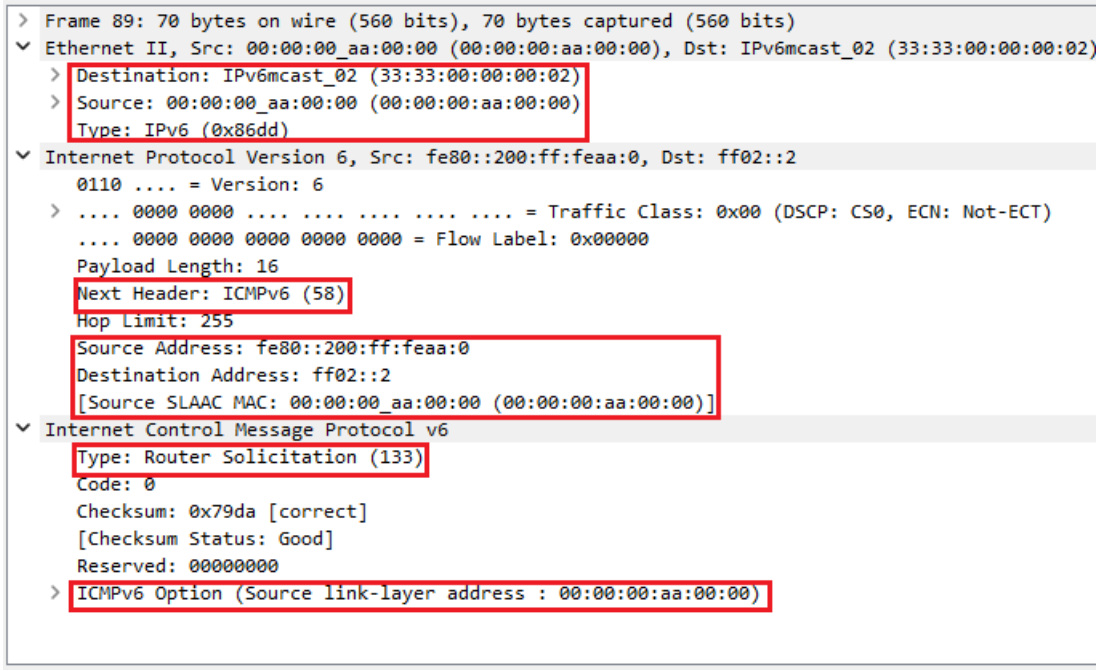

Figura C.3: Cenário 2 - Mensagem RS

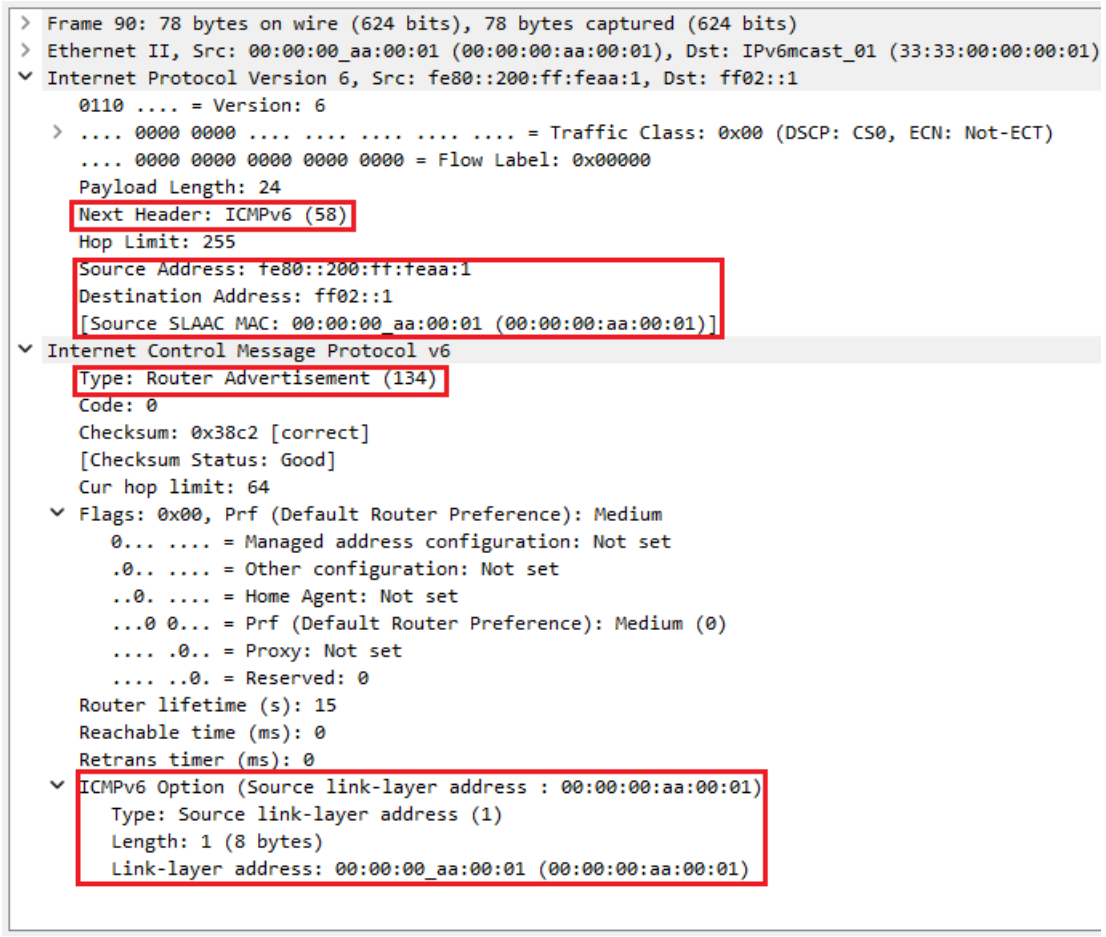

Figura C.4: Cenário 2 - Mensagem RA

C.3 Pacotes capturados Cenário 3

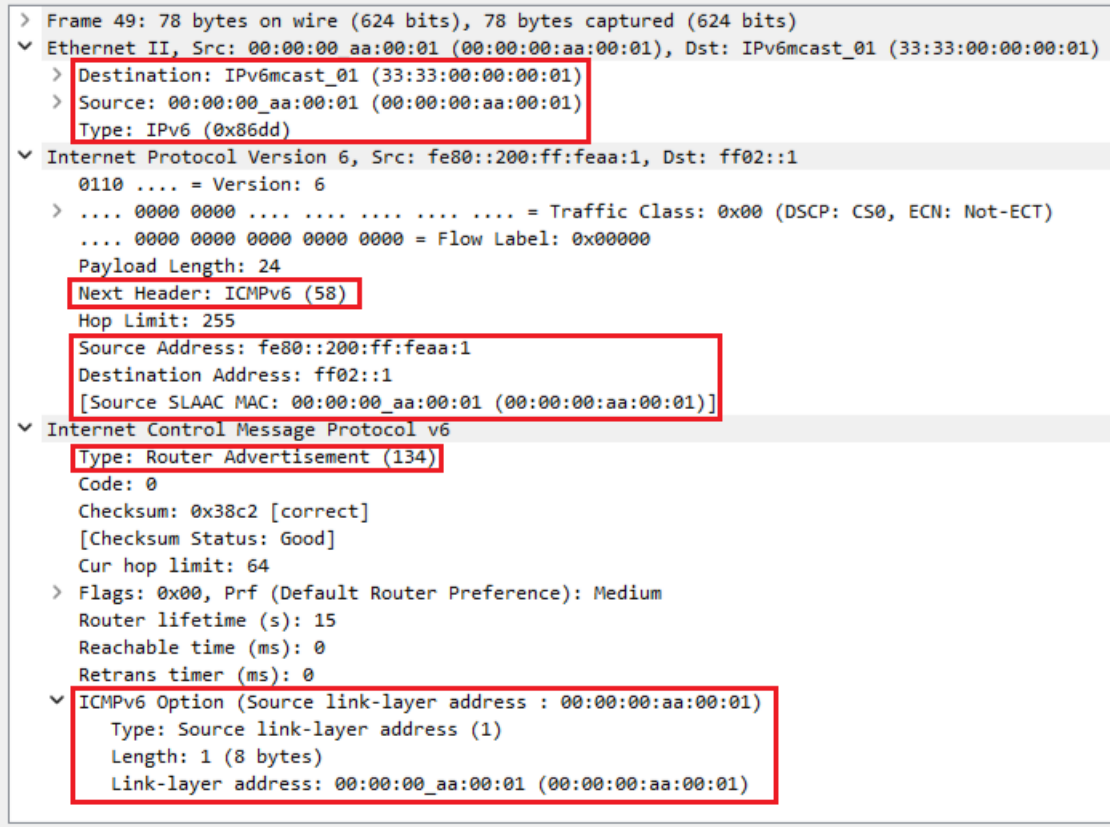

Figura C.5: Cenário 3 - Mensagem RA

C.4 Pacotes capturados Cenário 4

```
> Frame 24: 78 bytes on wire (624 bits), 78 bytes captured (624 bits)
v Ethernet II, Src: 00:00:00_aa:00:01 (00:00:00:aa:00:01), Dst: IPv6mcast_ff:00:00:01 (33:33:ff:00:00:01)
  > Destination: IPv6mcast_ff:00:00:01 (33:33:ff:00:00:01)
  > Source: 00:00:00_aa:00:01 (00:00:00:aa:00:01)
    Type: IPv6 (0x86dd)
v Internet Protocol Version 6, Src: ::, Dst: ff02::1:ff00:1
    0110 .... = Version: 6
  > .... 0000 0000 .... .... .... .... .... = Traffic Class: 0x00 (DSCP: CS0, ECN: Not-ECT)
    .... 0000 0000 0000 0000 0000 = Flow Label: 0x00000
    Payload Length: 24
    Next Header: ICMPv6 (58)
    Hop Limit: 255
    Source Address: ::
    Destination Address: ff02::1:ff00:1
v Internet Control Message Protocol v6
    Type: Neighbor Solicitation (135)
    Code: 0
    Checksum: 0x5a77 [correct]
    [Checksum Status: Good]
    Reserved: 00000000
    Target Address: 2022:d::1
```

Figura C.6: Cenário 4 - Mensagem NS

C.5 Pacotes capturados Cenário 5

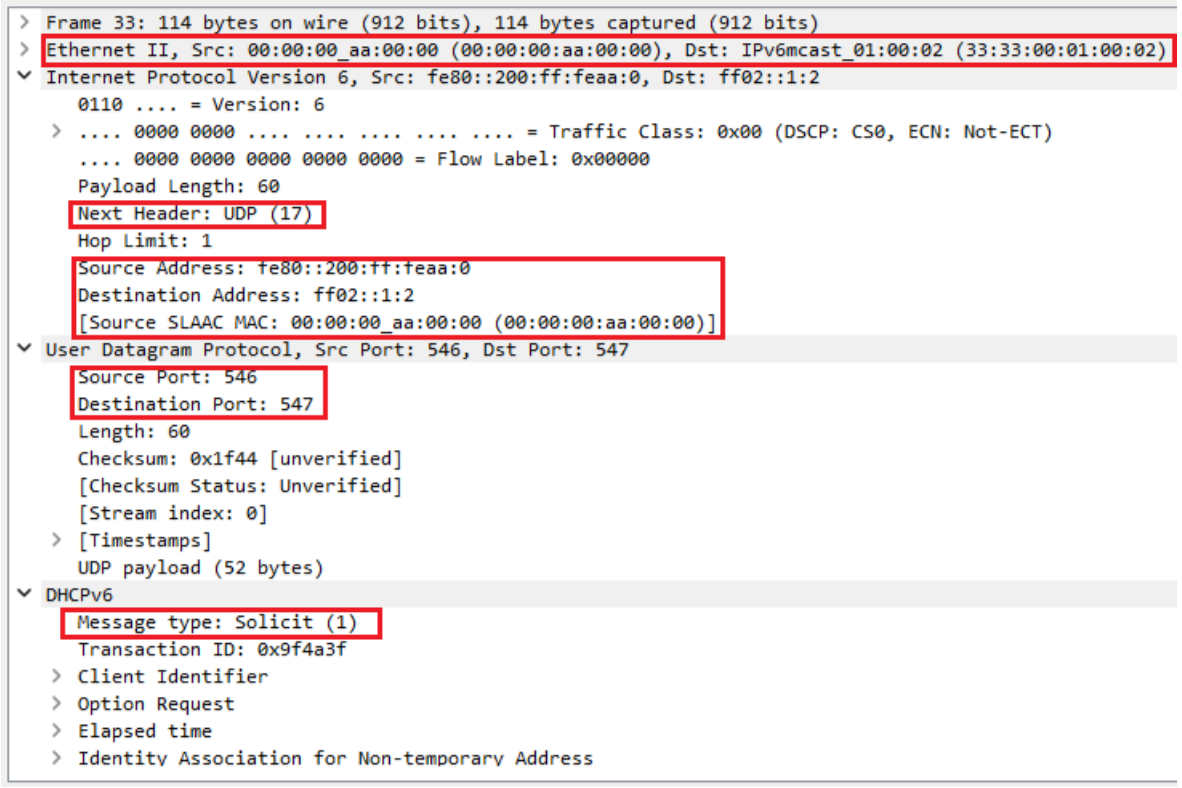

Figura C.7: Cenário 5 - Mensagem Solicit

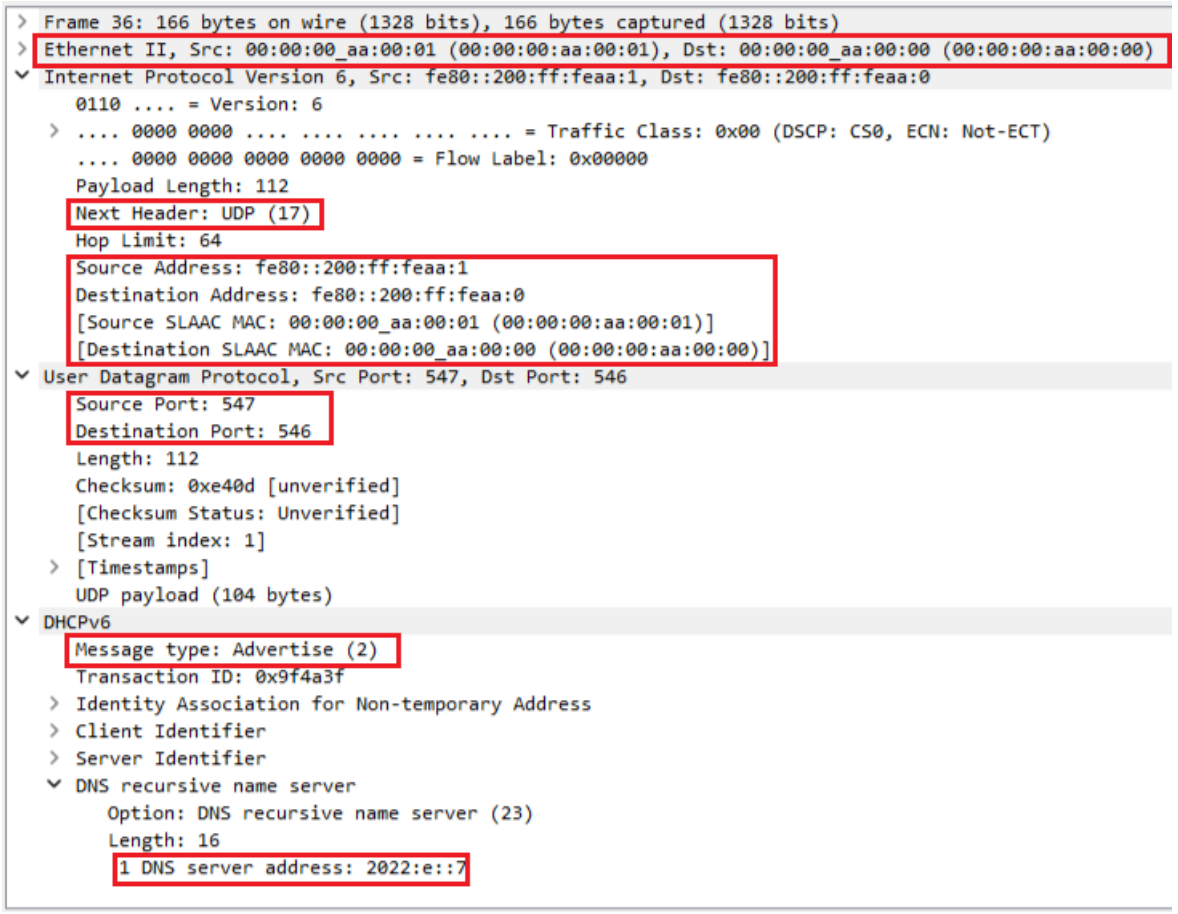

Figura C.8: Cenário 5 - Mensagem Advertise

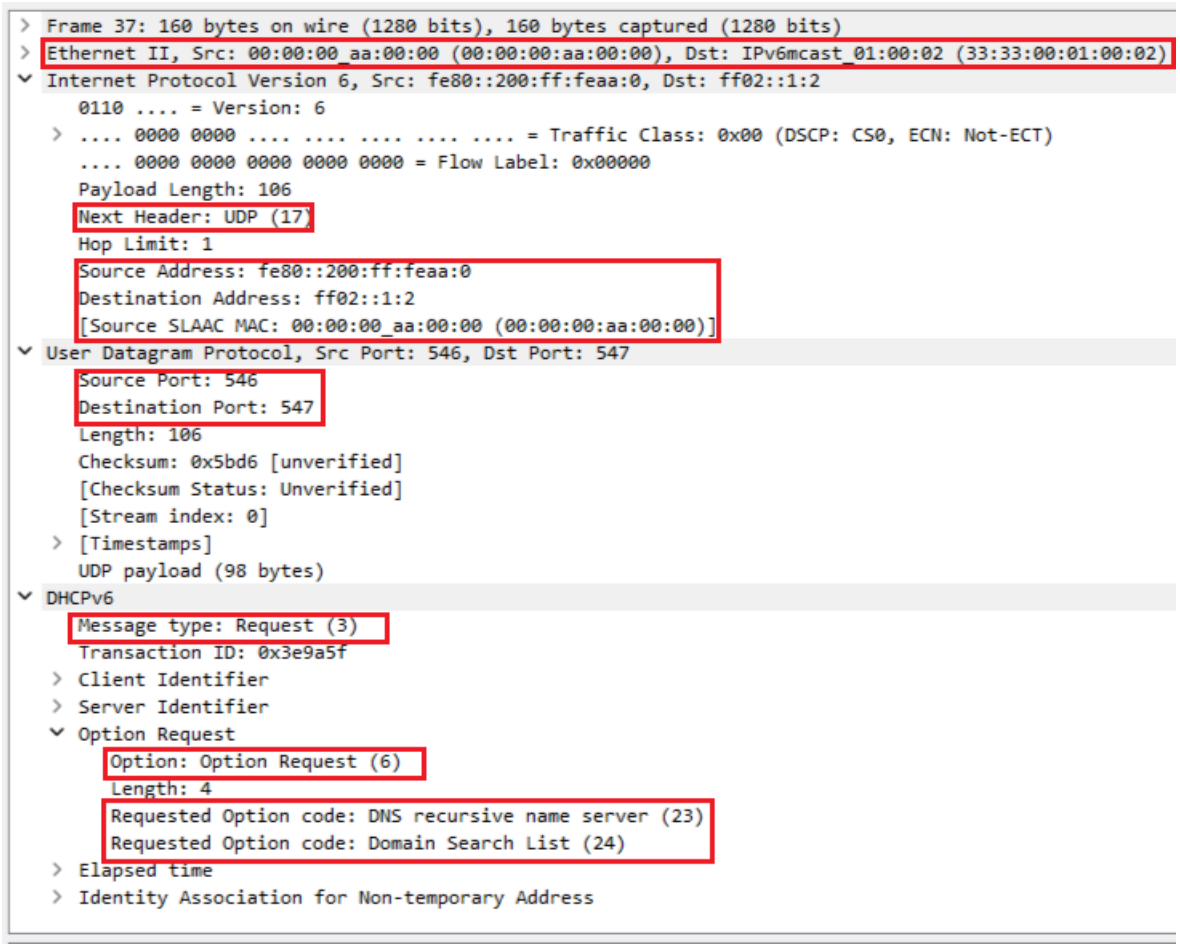

Figura C.9: Cenário 5 - Mensagem Request

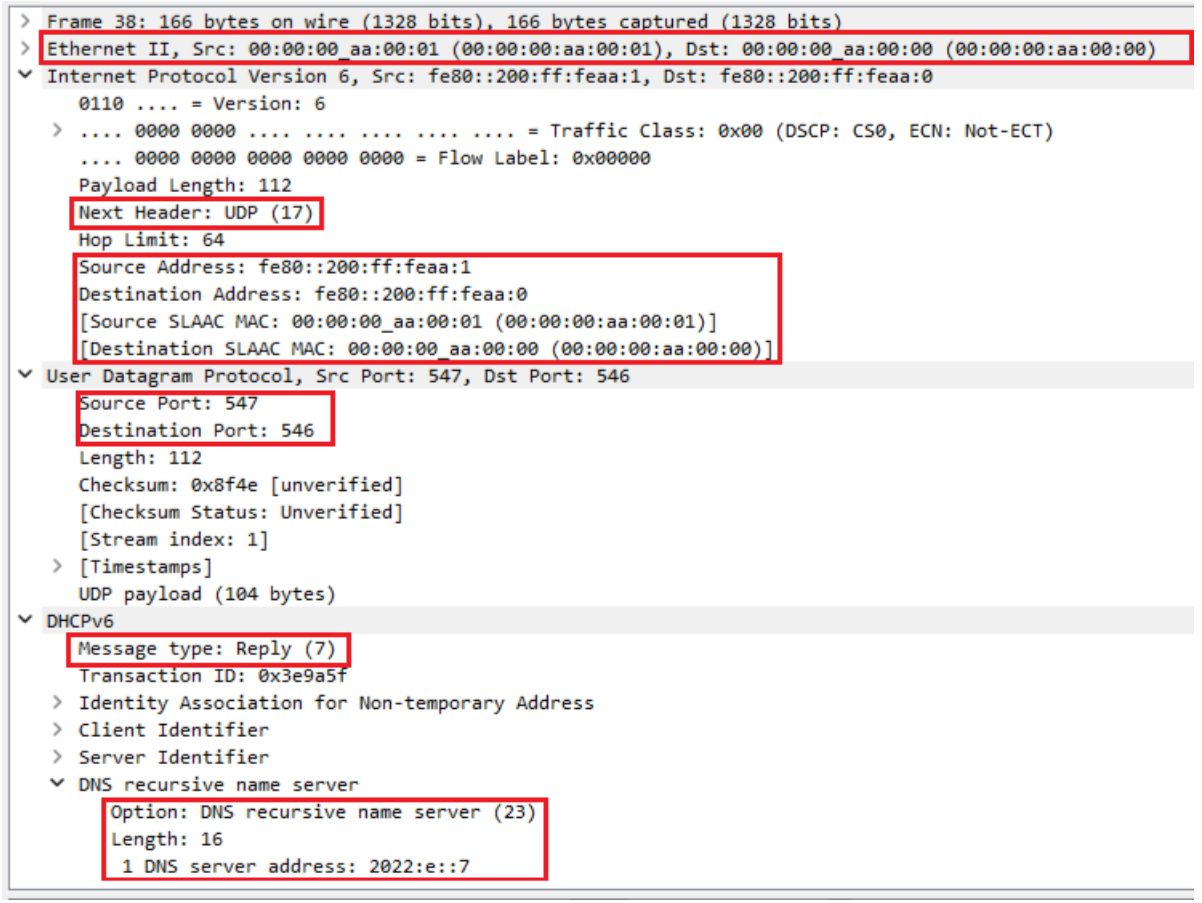

Figura C.10: Cenário 5 - Mensagem Reply

C.6 Pacotes capturados Cenário 6

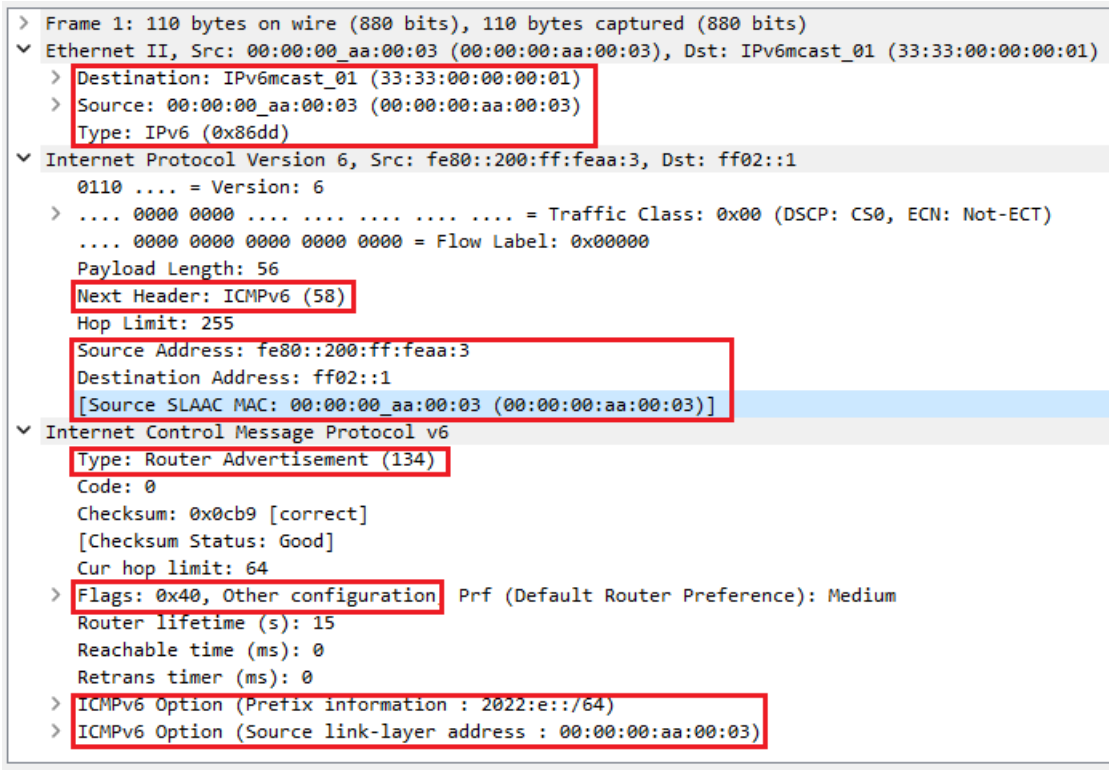

Figura C.11: Cenário 6 - Mensagem RA

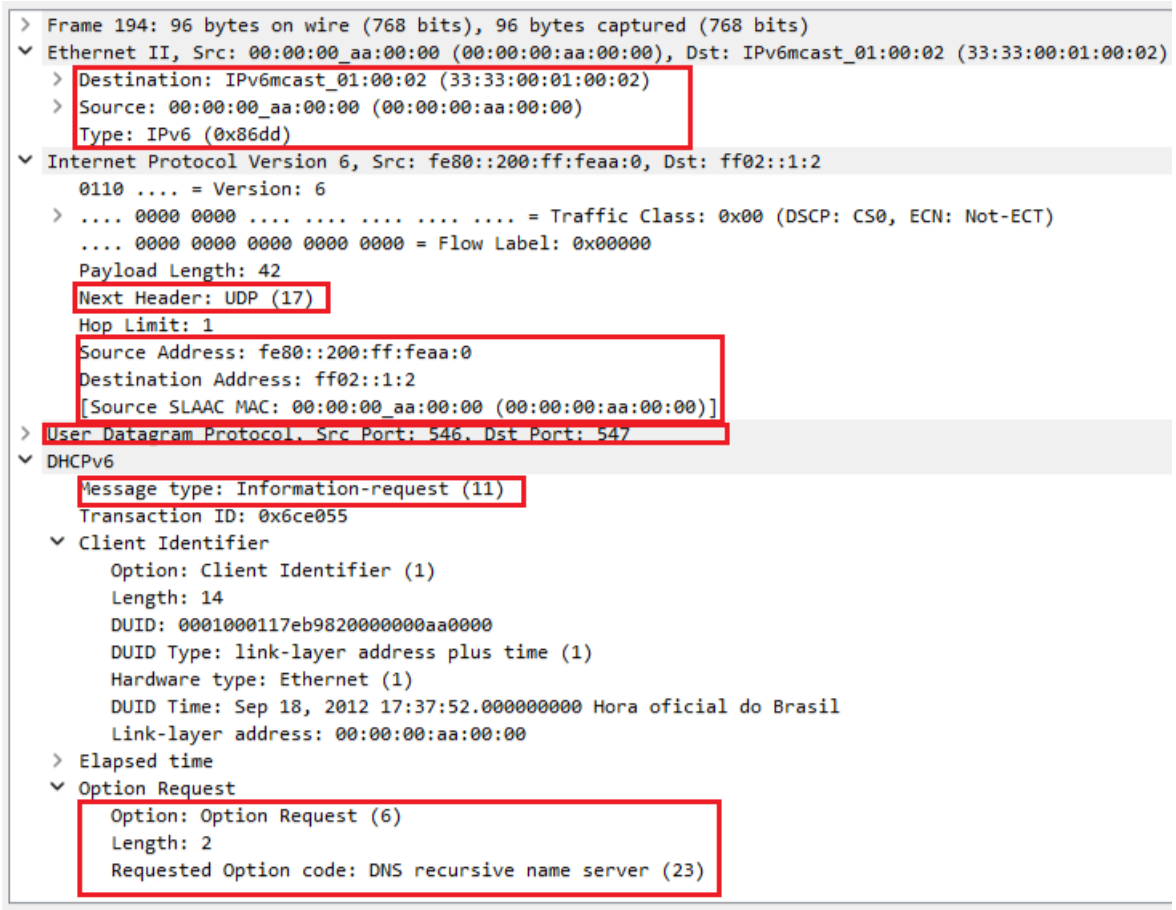

Figura C.12: Cenário 6 - Mensagem Information Request

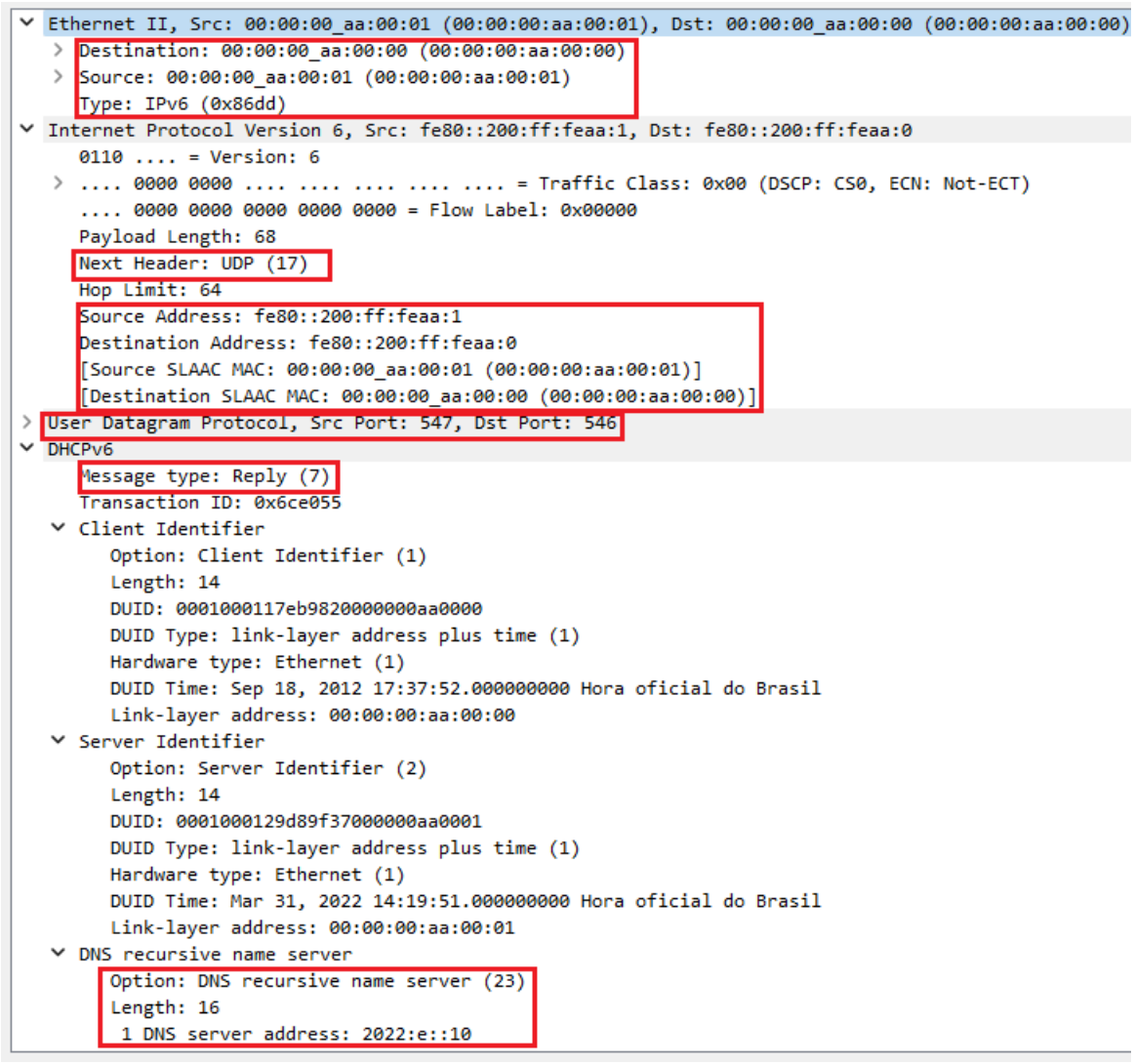

Figura C.13: Cenário 6 - Mensagem Reply