Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Ataque Distribuído de Negação de Serviço por Reflexão Amplificada Explorando o Protocolo Domain Name System

Rodrigo de Sousa Saldanha

Monografia apresentada como requisito parcial
para conclusão do Curso de Engenharia de Computação

Orientador
Prof. Dr. João José Costa Gondim

Brasília
2019

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Ataque Distribuído de Negação de Serviço por Reflexão Amplificada Explorando o Protocolo Domain Name System

Rodrigo de Sousa Saldanha

Monografia apresentada como requisito parcial
para conclusão do Curso de Engenharia de Computação

Prof. Dr. João José Costa Gondim (Orientador)
CIC/UnB

Prof. Dr. Eduardo Adilo Pelinson Alchieri
CIC/UnB

Prof. Dr. Marcos Fagundes Caetano
CIC/UnB

Prof. Dr. José Edil Guimarães de Medeiros
Coordenador do Curso de Engenharia de Computação

Brasília, 11 de julho de 2019

Dedicatória

Dedico este trabalho a todos os que me ajudaram de alguma forma durante a graduação, e especialmente, ao meu filho Miguel Franca Saldanha.

Agradecimentos

Primeiramente, agradeço aos meus pais e a toda minha família por ter me dado todo o apoio necessário durante toda a minha vida para que eu pudesse chegar até aqui, seja ele financeiramente, emocionalmente, incentivando e cobrando pelo meu crescimento como pessoa.

Aos meus colegas e amigos que fiz durante a graduação, em estágio, e também, aos que já conhecia. Agradeço por ter me ajudado de alguma forma a entrar na universidade, enfrentar as matérias e elaborar o trabalho final.

Agradeço a todos meus professores que tive na graduação e colegas de trabalho por ter me passado um pouco do conhecimento de cada um. Também, agradeço ao meu professor João Gondim por ter repassado o máximo possível de seus conhecimentos, dedicado tempo e paciência para me ensinar, e entendendo as dificuldades que enfrentei na universidade.

Finalmente, agradeço à Universidade de Brasília por ter me dado todo apoio e suporte para que eu pudesse adquirir conhecimentos mesmo com todas as dificuldades que ela sempre enfrentou.

Resumo

Ataques cibernéticos do tipo de negação de serviço (DoS) sempre foram uma grande preocupação na área de segurança de redes, e ao longo do tempo, tornaram-se cada vez mais usados e, também, mais robustos. Surgiram-se os distribuídos e os com amplificação. Com isso, faz-se necessário estudar como esses ataques funcionam, falhas, limitações e como se pode evitar.

O protocolo DNS é muito usado em toda a rede de computadores, e sem ele, a *Internet* não funcionaria como é hoje. Um recurso desse protocolo é consultar tudo sobre um certo domínio, podendo gerar uma resposta muito maior. E assim, se faz a amplificação explorada neste trabalho.

Portando, este trabalho estuda ataques de negação de serviço distribuídos com reflexão amplificada explorando o DNS, com o objetivo de descobrir se há alguma saturação nesse tipo de ataque, e também, se houver, aonde ocorre essa saturação.

Palavras-chave: DDoS, DNS, negação de serviço, negação de serviço distribuídos, amplificação, reflexão, segurança de redes

Abstract

Denial-of-service (DoS) cyberattacks always have been a huge concern in network security area and over time it has become increasingly used and also more robust. The distributed and amplified ones appeared. So that, It's necessary to study how these attacks works, failures, limitations and how can be avoided.

DNS protocol is widely used in computer networking and without that Internet wouldn't works like today. A DNS feature is to query everything about a domain which can make a greater response. In case, It's the amplification explored in this work.

Therefore, this work explore distributed reflection denial-of-service attacks with DNS, in order to find out if there is any saturation in this type of attack, and also where this saturation occurs.

Keywords: DDoS, DNS, denial-of-service, distributed denial-of-service, amplification, reflection, network security

Sumário

1 Introdução **1**
1.1 Motivação . . . 1
1.2 Justificativa . . . 1
1.3 Objetivos . . . 2
1.4 Organização do texto . . . 2

2 Fundamentação Teórica **4**
2.1 Negação de Serviço . . . 4
2.1.1 *Denial of Service (DoS)* . . . 4
2.1.2 *Distributed Denial of Service (DDoS)* . . . 5
2.2 *Domain Name System (DNS)* . . . 7
2.2.1 Funcionamento do DNS . . . 7
2.2.2 Registros DNS . . . 8
2.2.3 Mensagens . . . 10
2.2.4 Consulta *ANY* . . . 11
2.3 UDP vs TCP . . . 12

3 Ferramenta **13**
3.1 Arquitetura . . . 13
3.1.1 Oryx . . . 13
3.1.2 Commander . . . 14
3.1.3 Netuno . . . 15
3.2 Espelho DNS . . . 16

4 Resultados **18**
4.1 Configurações dos equipamentos . . . 18
4.2 Servidor DNS . . . 20
4.3 Execução do ataque e Coleta de Dados . . . 20
4.3.1 Amplificação de 15 vezes . . . 21
4.3.2 Amplificação de 30 vezes . . . 26

4.3.3 Amplificação de 45 vezes 31
4.4 Amplificação 36
4.5 Análise 40
4.5.1 Análise dos Testes 40
4.5.2 Análise da Amplificação 40
4.5.3 Análise dos Recursos 40
4.6 Considerações Finais 43

5 Conclusão e Trabalhos Futuros **44**
5.1 Conclusão 44
5.2 Trabalhos Futuros 45

Referências **47**

Lista de Figuras

1.1 Refletores mais usados em ataques DDoS. 2

2.1 DDoS na camada de aplicação. 5
2.2 DDoS abuso de protocolo. 6
2.3 DDoS volumétricos com reflexão amplificada. 6
2.4 Comparação DNS com o sistema de arquivos UNIX. 8
2.5 Método de consulta recursiva. 9
2.6 Cabeçalho DNS. 10
2.7 Comparação entre consultas sem e com DNSSEC. 12

3.1 Arquitetura da ferramenta Linderhof. 14

4.1 Topologia usada no ataque. 19
4.2 Pacotes enviados pelo atacante por segundo na amplificação de 15 no ataque incremental. 22
4.3 Bytes enviados pelo atacante por segundo na amplificação de 15 no ataque incremental. 22
4.4 Pacotes recebidos e enviados pelo refletor por segundo na amplificação de 15 no ataque incremental. 23
4.5 Bytes recebidos e enviados pelo refletor por segundo na amplificação de 15 no ataque incremental. 23
4.6 Pacotes recebidos pelo alvo por segundo na amplificação de 15 no ataque incremental. 24
4.7 Bytes recebidos pelo alvo por segundo na amplificação de 15 no ataque incremental. 25
4.8 Pacotes por segundo na amplificação de 15 em escala logarítmica. 25
4.9 Bytes por segundo na amplificação de 15 em escala logarítmica. 26
4.10 Pacotes enviados pelo atacante por segundo na amplificação de 30 no ataque incremental. 27

4.11 Bytes enviados pelo atacante por segundo na amplificação de 30 no ataque incremental. 27
4.12 Pacotes recebidos e enviados pelo refletor por segundo na amplificação de 30 no ataque incremental. 28
4.13 Bytes recebidos e enviados pelo refletor por segundo na amplificação de 30 no ataque incremental. 28
4.14 Pacotes recebidos pelo alvo por segundo na amplificação de 30 no ataque incremental. 29
4.15 Bytes recebidos pelo alvo por segundo na amplificação de 30 no ataque incremental. 30
4.16 Pacotes por segundo na amplificação de 30 em escala logarítmica. 30
4.17 Bytes por segundo na amplificação de 30 em escala logarítmica. 31
4.18 Pacotes enviados pelo atacante por segundo na amplificação de 45 no ataque incremental. 32
4.19 Bytes enviados pelo atacante por segundo na amplificação de 45 no ataque incremental. 32
4.20 Pacotes recebidos e enviados pelo refletor por segundo na amplificação de 45 no ataque incremental. 33
4.21 Bytes recebidos e enviados pelo refletor por segundo na amplificação de 45 no ataque incremental. 34
4.22 Pacotes recebidos pelo alvo por segundo na amplificação de 45 no ataque incremental. 35
4.23 Bytes recebidos pelo alvo por segundo na amplificação de 45 no ataque incremental. 35
4.24 Pacotes por segundo na amplificação de 45 em escala logarítmica. 36
4.25 Bytes por segundo na amplificação de 45 em escala logarítmica. 36
4.26 Gráfico da amplificação real do ataque em bytes. 38
4.27 Gráfico da amplificação real do ataque em pacotes. 38
4.28 Gráfico da amplificação real do ataque em bytes em escala logarítmica. . . 39
4.29 Gráfico da amplificação real do ataque em pacotes em escala logarítmica. . 39
4.30 Recursos do atacante na amplificação de 45. 41
4.31 Recursos do refletor na amplificação de 45. 42
4.32 Recursos do alvo na amplificação de 45. 42

Lista de Tabelas

3.1 Níveis permitidos pela ferramenta Linderhof. 15
3.2 Campos do cabeçalho DNS usados na ferramenta Linderhof. 17

4.1 Configurações do equipamento do atacante. 18
4.2 Configurações do equipamento do refletor. 19
4.3 Configurações do equipamento do alvo. 19
4.4 Configurações do roteador. 19
4.5 Quantidade de pacotes enviados pelo atacante por segundo na amplificação de 15. 21
4.6 Quantidade de pacotes recebidos e enviados pelo refletor por segundo na amplificação de 15. 23
4.7 Quantidade de pacotes recebidos pelo alvo por segundo na amplificação de 15. 24
4.8 Quantidade de pacotes enviados pelo atacante por segundo na amplificação de 30. 26
4.9 Quantidade de pacotes recebidos e enviados pelo refletor por segundo na amplificação de 30. 28
4.10 Quantidade de pacotes recebidos pelo alvo por segundo na amplificação de 30. 29
4.11 Quantidade de pacotes enviados pelo atacante por segundo na amplificação de 45. 32
4.12 Quantidade de pacotes recebidos e enviados pelo refletor por segundo na amplificação de 45. 33
4.13 Quantidade de pacotes recebidos pelo alvo por segundo na amplificação de 45. 34
4.14 Amplificação real do ataque em bytes. 37
4.15 Amplificação real do ataque em pacotes. 37
4.16 Tráfego da *interface* de rede do atacante na amplificação de 45. 41
4.17 Tráfego da *interface* de rede do refletor na amplificação de 45. 41

4.18 Tráfego da *interface* de rede do alvo na amplificação de 45. 42

Lista de Abreviaturas e Siglas

CLI Command-line Interface.

DDoS Distributed Denial of Service.

DNS Domain Name System.

DNSSEC Domain Name System Security Extensions.

DoS Denial of Service.

HTTP Hypertext Transfer Protocol.

IP Internet Protocol.

MTU Maximum Transmission Unit.

NTP Network Time Protocol.

RFC Request for Comments.

SNMP Simple Network Management Protocol.

SSDP Simple Service Discovery Protocol.

TCP Transmission Control Protocol.

TLD Top-level Domain.

TTL Time-to-Live.

UDP User Datagram Protocol.

UnB Universidade de Brasília.

URL Uniform Resource Locator.

Capítulo 1

Introdução

Apesar da evolução do protocolo DNS e com a adição de algumas características para se aumentar a segurança do protocolo, ainda há algumas vulnerabilidades e certas propriedades para serem exploradas por ataques cibernéticos. Com o DDoS é possível se aproveitar dessas particularidades, e assim, gerar uma amplificação para o ataque.

1.1 Motivação

Os ataques DDoS ainda têm mais para serem explorados, é o que diz um relatório recente da AKAMAI, *State of the Internet / Security: DDoS and Application Attacks* [1], onde mostra que a quantidade de *bots* está, cada vez mais, crescendo, consolidando assim como uma grande preocupação para a segurança da *Internet*, principalmente, para aqueles serviços ou aplicações que são frequentemente usadas pelos próprios *bots*.

Além do mais, com a propagação dos *bots* e a reflexão amplificada, os ataques de negação de serviço estão constantemente aumentando em quantidade de bytes por segundo. Um dos maiores ataques registrados ocorreu fevereiro de 2018 usando *memcached* como refletor, com o pico de aproximadamente 1,3Tbps, também foi reportado pela AKAMAI [2].

1.2 Justificativa

O protocolo DNS é essencial para toda a *Internet*, pois é muito usado para várias aplicações. Existem vários servidores DNS espalhados pelo mundo, tornando o ataque DDoS explorando o DNS uma boa investida para os atacantes, pois podem gerar uma ótima amplificação e ao mesmo tempo há uma ampla opção de refletores.

Um relatório publicado recentemente [3], onde mostra milhões de *bots* sendo usados para ataques DDoS, os servidores DNS ficaram em segundo como os mais usados. A Figura 1.1 ilustra bem essa colocação.

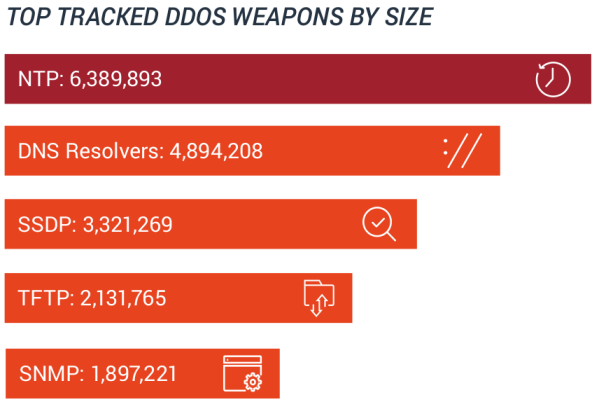

Figura 1.1: Refletores mais usados em ataques DDoS (Fonte: [3]).

1.3 Objetivos

Este trabalho tem o objetivo de desenvolver uma ferramenta capaz de simular uma ambiente de negação de serviço com características de reflexão e amplificação em vários níveis de injeção de pacotes. Além disso, observar qual a dimensão da amplificação que o protocolo DNS apresenta, e também, verificar se há algum tipo de saturação no ataque, e se houver, onde ela ocorre.

O trabalho desenvolvido possui fins acadêmicos para demonstração e análise de um ataque em um ambiente controlado. O autor e seu orientador não se responsabilizam por qualquer uso inapropriado ou ilegal que estão fora do contexto e dos objetivos desse trabalho. O código desenvolvido está sob a custódia do autor e do orientador para a continuidade futura desse estudo.

1.4 Organização do texto

Este trabalho está organizado na seguinte forma:

- No Capítulo 2, estão descritos os fundamentos teóricos necessários para compreensão do desenvolvimento desse trabalho.
- No Capítulo 3, apresenta-se parte da ferramenta desenvolvida para o ataque.
- No Capítulo 4, os resultados obtidos são analisados e discutidos.

- No Capítulo 5, relata a conclusão e os trabalhos futuros.

Capítulo 2

Fundamentação Teórica

Neste capítulo serão apresentados os conceitos fundamentais para se compreender um ataque de negação de serviço, e também, o funcionamento do protocolo *Domain Name System (DNS)*. Da mesma forma, será explicado como ocorre o ataque DDoS com o DNS.

2.1 Negação de Serviço

2.1.1 *Denial of Service (DoS)*

O ataque de negação de serviço tem como objetivo esgotar os recursos ou infraestrutura do alvo, que pode ser um computador, servidor, *switch*, roteador, *firewall* ou qualquer outro dispositivo com conexão à *Internet*, interrompendo o acesso de usuários legítimos a serviços ou recursos compartilhados. O ataque se dá da seguinte maneira: apenas um *host* cria um fluxo muito grande de dados para outro dispositivo, que não suporta processar todas as informações e por consequência acaba exaurindo todos os seus recursos computacionais.

A empresa do ramo de segurança de redes, a *Cloudflare*, classificou os ataques DoS em duas categorias [4]:

- Os *buffer overflow*, onde haveria um esgotamento de recursos da máquina.
- Os *flood*, seria um super saturação do canal de comunicação da vítima.

Esse tipo de ataque é muito fácil de se realizar, basta uma máquina, que seria o atacante ou *bot* enviando informações, dados, consultas ao alvo. E consequentemente, também é simples se proteger, quando detectado bloqueia-se a conexão com o atacante.

2.1.2 *Distributed Denial of Service (DDoS)*

Esse ataque é uma evolução do DoS, assim como o seu antecessor, ele também tem o objetivo de exaurir os recursos ou infraestrutura da vítima. Enquanto no DoS tem apenas um *host*, o DDoS possui várias fontes de ataque, formando uma *botnet*.

Há vários tipos diferentes de ataques DDoS, cada um explorando diferentes vetores. A Cloudflare classificou alguns que serão mostrados a seguir [5].

Camada de Aplicação

Tem como finalidade exaurir os recursos computacionais do alvo. Um exemplo são ataques a servidores *Web*, o atacante gerencia vários *bots* que fazem solicitações HTTP constantemente. Uma única solicitação HTTP é simples de se executar no lado do cliente, mas pode ser custoso para o servidor responder, pois geralmente deve-se carregar vários arquivos e executar várias consultas no banco de dados para se criar uma página da *web*. É trabalhoso se detectar esse tipo de ataque, pois para descobrir se as solicitações são legítimas ou maliciosas é bastante complexo. A Figura 2.1 exemplifica esse tipo de ataque [5].

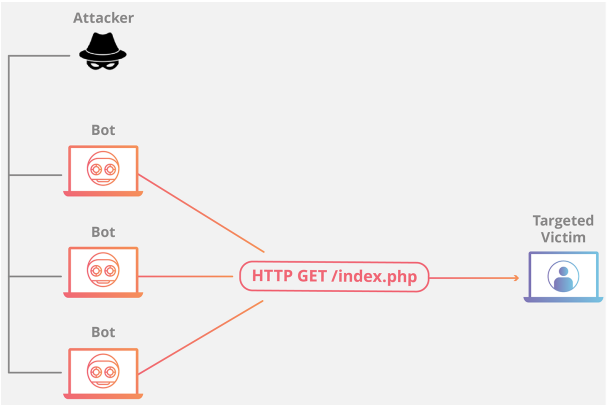

Figura 2.1: DDoS na camada de aplicação.

Abuso de Protocolo

Esses ataques utilizam pontos fracos na camada 3 e 4 para tornar o alvo inacessível. Exemplificando, o SYN *flood* se encaixa nesse tipo de ataque. Explorando o handshake do protocolo TCP, os *bots* enviam vários pacotes com o campo SYN preenchido, e com os IP de origem mascarados para a vítima, que responde cada solicitação de conexão e, em seguida, aguarda a etapa final do *handshake*, o que nunca ocorre, esgotando todos os recursos. A Figura 2.2 demonstra bem essa espécie de DDoS [5].

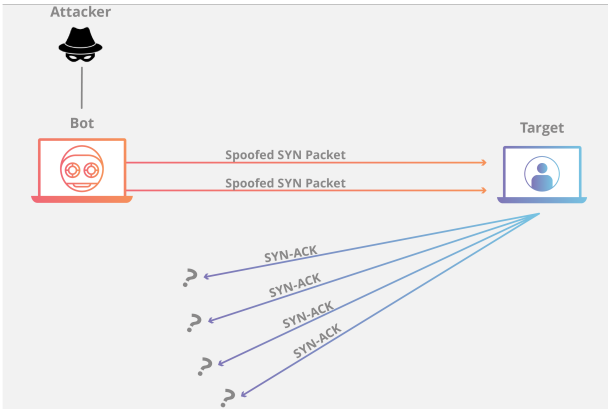

Figura 2.2: DDoS abuso de protocolo.

Volumétricos

Essa é a categoria de DDoS que será explorada neste trabalho. Tem o objetivo de criar um congestionamento consumindo toda a largura de banda disponível entre o alvo e suas conexões externas. Grandes quantidades de dados são enviadas para a vítima, através de uma *botnet*, usando uma forma de amplificação, por meio de um refletor. Essa ampliação no sinal pode ser uma requisição ao refletor que gera vários outros pacotes como resposta, ou uma resposta que gera outro pacote maior que a pergunta. A Figura 2.3 exemplifica essa amplificação por meio de servidores DNS [5], assim como será feito neste trabalho.

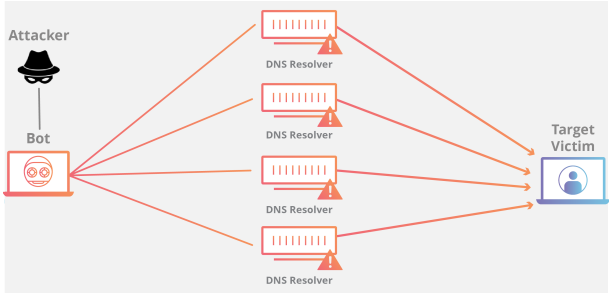

Figura 2.3: DDoS volumétricos com reflexão amplificada.

Todos os tipos de DDoS torna a detecção do ataque muito mais difícil se comparado ao seu precursor, o DoS, pois agora, existem várias fontes atacando a vítima. Torna-se muito complexo definir quais são tráfegos legítimos e quais são falsos, além do mais, quando são requisições maliciosas que o alvo consideram válidas como é o caso dos ataques de camada de aplicação.

Alguns estudos já foram feitos sobre esse tipo de ataque que será explorado nesse trabalho. Designou-se duas possibilidades de reflexão [6], a que será estudada é simplesmente enviar uma requisição ao refletor DNS. Além do mais, também será usado o

DNSSEC, fazendo com o que a amplificação seja maior [7], uma explicação desse recurso será apresentada mais a frente.

2.2 *Domain Name System (DNS)*

O DNS é um protocolo da camada de aplicação que é usado para traduzir IP em nomes, e também, o contrário. Funciona como um banco de dados distribuído executado em uma hierarquia de servidores de DNS [8]. Por exemplo, pesquisando o *site* www.exemplo.com.br no *browser*, tudo ocorre da seguinte maneira:

1. A própria máquina executa no cliente a aplicação DNS.
2. O navegador extrai o nome de hospedeiro, www.exemplo.com.br, da URL e passa para o cliente DNS.
3. O cliente DNS envia uma consulta contendo o nome do hospedeiro para um servidor DNS.
4. O cliente DNS por fim recebe uma resposta, que inclui o endereço IP correspondente ao nome do hospedeiro.
5. Tão logo o navegador receba o endereço do DNS, pode abrir uma conexão TCP com o processo servidor HTTP localizado na porta 80 naquele endereço IP.

O DNS tem algumas outras funções, como apelido dos nomes, apelido para servidor de *email* e distribuição de carga [8]. O conceito e especificações do protocolo está detalhado na RFC 1034 [9], na RFC 1035 [10], e atualizado em diversos outras RFC adicionais.

2.2.1 Funcionamento do DNS

Há vários servidores DNS espalhados pelo mundo, organizados de maneira hierárquica e distribuídos. Nenhum servidor DNS isolado tem todos os mapeamentos para todos os nomes da *Internet*. Em vez disso, os mapeamentos são distribuídos pelos servidores DNS. Há três classes de servidores DNS: *root*, *Top-level Domain (TLD)* e os *authoritative*, organizados em uma hierarquia [8], similar ao sistema de arquivos UNIX como mostra a Figura 2.4.

- Servidores DNS raiz (*root*): na *Internet* há 13 organizações gerenciando mais de 400 servidores DNS raiz e a maior parte deles está localizada na América do Norte. Todos eles formam um conglomerado de servidores replicados, para fins de segurança e confiabilidade [8].

- Servidores DNS de Domínio de Alto Nível (*Top-level Domain (TLD)*): esses servidores são responsáveis por domínios de alto nível como com, org, net, edu e gov, e por todos os domínios de alto nível de países, tais como br, uk, ar, etc.
- Servidores DNS autoritativos (*authoritative*): toda organização que possui domínios a serem acessados publicamente na *Internet* deve fornecer registros DNS também acessíveis publicamente, mapeando os nomes desses domínios para seus respectivos IP. A maioria das universidades e empresas de grande porte executa e mantém seus próprios servidores DNS primário e secundário (*backup*) autoritativos [8].

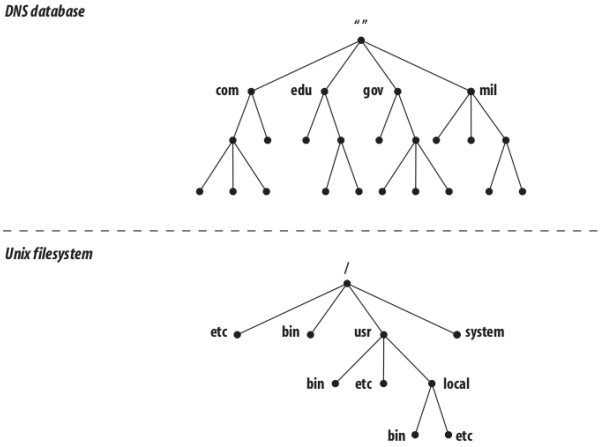

Figura 2.4: Comparação DNS com o sistema de arquivos UNIX (Fonte: [11]).

Para as consultas nos servidores DNS, existem os métodos recursivo e interativo, o que será usado neste trabalho é o recursivo. Como nos algoritmos recursivos em programação, o servidor DNS repete o mesmo processo básico até receber a resposta. Quando um servidor recebe uma solicitação recursiva e não tem a resposta, deve-se consultar outros servidores, podendo fazer outra consulta recursiva ou interativa, a Figura 2.5 ilustra esse método.

2.2.2 Registros DNS

Os servidores DNS que juntos executam o banco de dados distribuídos, armazenam os chamados registros de recursos, que fornecem mapeamentos de nomes de hospedeiros para endereços IP. Cada mensagem de resposta DNS carrega um ou mais registros de recursos. Um registro de recurso é uma tupla de quatro elementos que contém os seguintes campos: *Name*, *Value*, *Type*, TTL [8].

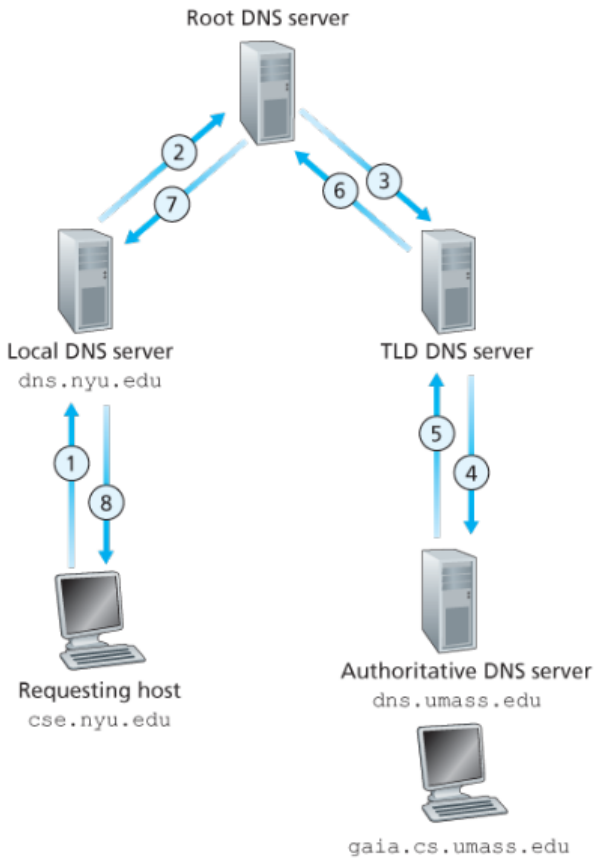

Figura 2.5: Método de consulta recursiva (Fonte: [8]).

- Se *Type* for A, então *Name* é um nome de hospedeiro e *Value* é o endereço IP para o nome de hospedeiro. Assim, um registro *Type* A fornece o mapeamento padrão entre nomes de hospedeiros e endereços IP.
- Se *Type* for NS, então *Name* é um domínio (como exemplo.com) e *Value* é o nome de um servidor DNS autoritativo que sabe como obter os endereços IP para hospedeiros do domínio.
- Se *Type* for CNAME, então *Value* é um nome canônico de hospedeiro para o apelido de hospedeiro contido em *Name*. Esse registro pode fornecer aos hospedeiros consultantes o nome canônico correspondente a um apelido de hospedeiro.
- Se *Type* for MX, então *Value* é o nome canônico de um servidor de correio cujo apelido de hospedeiro está contido em *Name*.

2.2.3 Mensagens

Há somente duas mensagens no protocolo DNS, as de consulta e as de resposta. Elas têm o mesmo formato, ou seja, os mesmos campos no cabeçalho, diferenciando as perguntas de respostas apenas nos valores preenchidos nos campos. A Figura 2.6 apresenta como é o cabeçalho do protocolo DNS com todos os seus campos.

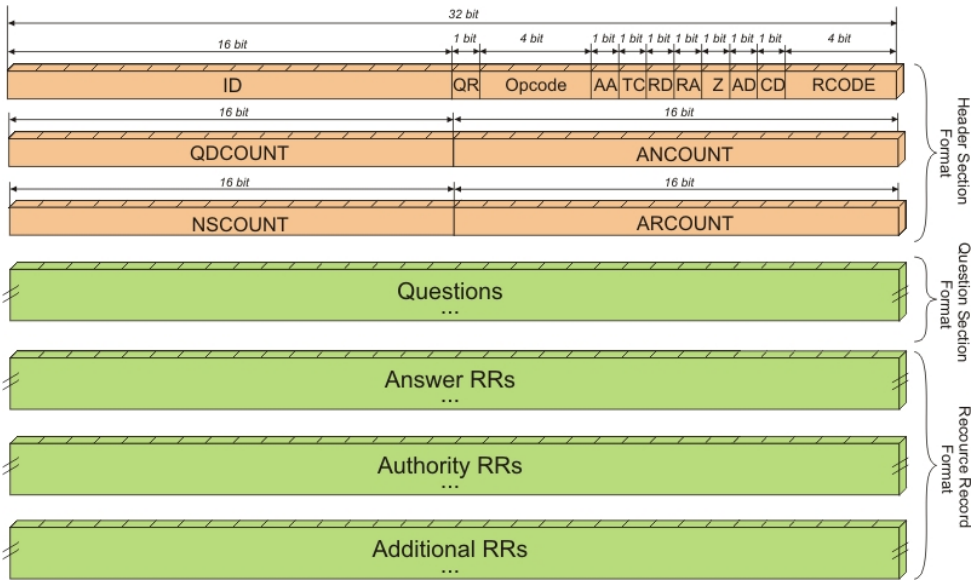

Figura 2.6: Cabeçalho DNS (Fonte: [12]).

Campos

- ID: identificação da consulta.
- QR: se é pergunta ou resposta.
- OPCODE: código de consulta.
- AA: especifica se o servidor é autoritativo para o nome consultado.
- TC: indica que apenas os primeiros 512 bytes da resposta foram retornados.
- RD: define se a consulta deve ser feita recursivamente.
- RA: indica se é possível consultar recursivamente no servidor consultado.
- Z: reservado para uso futuro.
- AD: assinala na resposta que todos os dados incluídos foram autenticados pelo servidor de acordo com as políticas dele.

- CD: indica que os dados não autenticados são admissíveis para quem está fazendo a consulta.
- RCODE: código de resposta.
- QDCOUNT: número de consultas no campo de dados.
- ANCOUNT: número de respostas no campo de dados.
- NSCOUNT: número de autoritativos no campo de dados.
- ARCOUNT: número de adicionais no campo de dados.
- Questions: campo para dados de consultas.
- Answers RRs: campo para dados de respostas.
- Authority RRs: campo para dados de autoritativos.
- Additional RRs: campo para dados de adicionais.

2.2.4 Consulta *ANY*

Para entender o ataque de DDoS com reflexão amplificada é necessário saber como funciona a mensagem *any* do protocolo DNS. Essa consulta retorna todos os tipos de registros de um certo domínio, já mencionados anteriormente, os tipos A, NS, CNAME e MX. Ou seja, uma simples pergunta sobre tudo de um certo nome, responde, geralmente, um pacote muito maior que o pedido. O tamanho da resposta vai depender da quantidade de informação que aquela zona do servidor DNS consultado tem sobre o domínio. Assim, é feita a amplificação no ataque DDoS explorando o protocolo DNS.

Campo *ADDITIONAL*

Além dos registros principais, com o campo *additional* preenchido, é possível receber mais tipos de registros, como AAAA, outros servidores DNS que contém informações sobre o domínio, e até informações no formato de texto.

Domain Name System Security Extensions (DNSSEC)

O DNSSEC foi criado para fornecer uma maior segurança ao protocolo DNS, reduzindo o risco de manipulação de dados e informações. Especificado na RFC 2535 [13] tem o objetivo de fornecer três funções: distribuição de chaves, autenticação da origem dos dados e autenticação das transações e solicitações. Para isso, as chaves públicas e assinaturas devem ser publicadas na zona do servidor DNS, aumentando a quantidade de informações

da própria zona. Consequentemente, a amplificação para o ataque DDoS aumenta com o campo DNSSEC preenchido.

A Figura 2.7 identifica duas consultas ao servidor DNS da UnB de IP 164.41.101.3. Usando o comando 'dig @164.41.101.3 any unb.br', gerou-se como retorno o pacote de número 57, recebendo uma resposta de tamanho 459. Para o pacote 121, foi usado o comando 'dig @164.41.101.3 any unb.br +dnssec', e retornou com um tamanho de 5505, além do mais, é possível se observar outros tipos na resposta, como RRSIG.

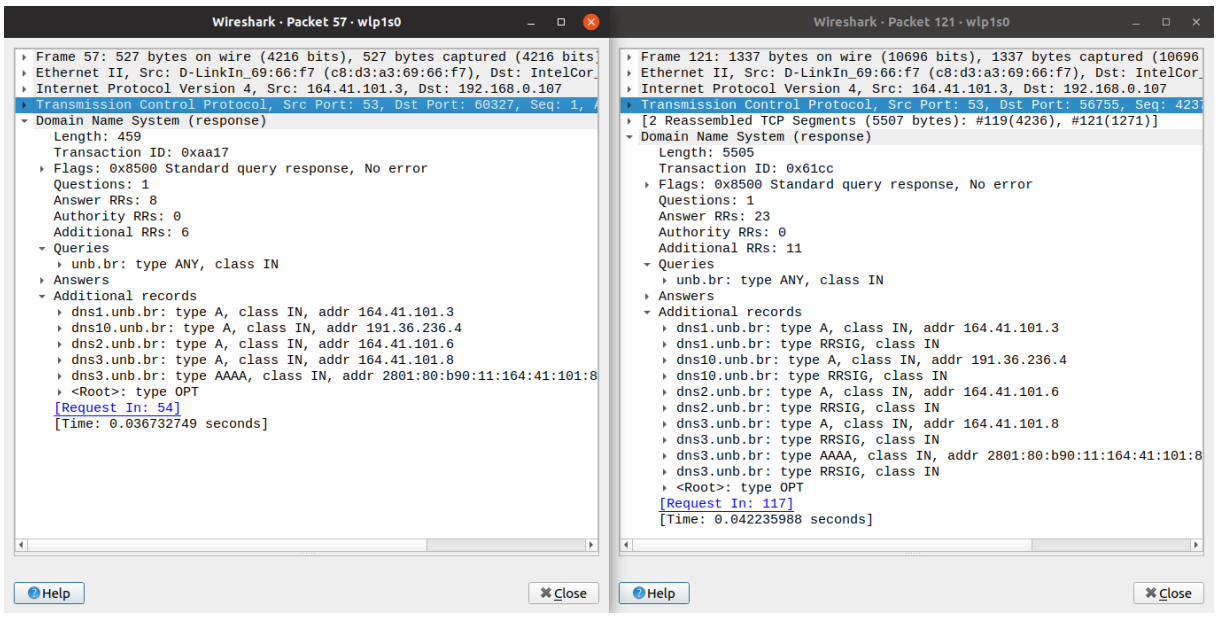

Figura 2.7: Comparação entre consultas sem e com DNSSEC.

2.3 UDP vs TCP

Para se realizar ataques dessa categoria é necessário que o atacante faça um certo tipo de mascaramento no pacote a ser enviado, basicamente, o IP de origem é trocado pelo IP do alvo na geração do pacote, assim, quem receber esse pacote mascarado reponderá ao IP de origem, que no caso é a vítima.

O protocolo *Transmission Control Protocol (TCP)* [14] inviabiliza esse recurso, já que ele é orientado a conexão, isto é, há um tipo de apresentação entre os dois *hosts*, o chamado *handshake*. Logo, o refletor, primeiramente, responderia ao alvo para se estabelecer uma conexão TCP. Por isso, neste trabalho será usado o protocolo *User Datagram Protocol (UDP)* [15] para simular o ataque, que não é orientado a conexão.

Capítulo 3

Ferramenta

Nesse capítulo será tratado sobre parte do funcionamento da ferramenta usada para o ataque DDoS, chamada de Linderhof.

3.1 Arquitetura

A arquitetura do Linderhof foi pensada para ser a mais modular possível. Ela possui três módulos como mostra a Figura 3.1, o Oryx, que seria a *interface*, o Commander, responsável por criar e planejar o ataque, e o Netuno, que é o injetor de pacotes. Os módulos Oryx e Netuno não impactam no Commander, assim, é possível se utilizar qualquer outra *interface* ou injetor de pacotes.

3.1.1 Oryx

Como dito anteriormente, esse módulo é responsável pela *interface* com o usuário, que é uma *Command-line Interface (CLI)*. Além de se comunicar com o usuário, também é encarregado de criar a estrutura do ataque. A seguir são apresentados os argumentos que podem ser usados na ferramenta, e se é obrigatório ou não.

- -m: Obrigatório. Espelho/refletor que será utilizado no ataque.
- -t: Obrigatório. IP do alvo.
- -a: Obrigatório. IP do refletor.
- -h: Opcional. Mostra a ajuda do programa.
- -p: Opcional. Porta dos pacotes enviados ao refletor.
- -g: Opcional. Porta dos pacotes respondidos ao alvo.

- -l: Opcional. Nível do ataque.
- -c: Opcional. Tempo de ataque.
- -f: Opcional. Nome do arquivo para escrita do *log* do ataque.
- -i: Opcional. Incremento no nível do ataque.

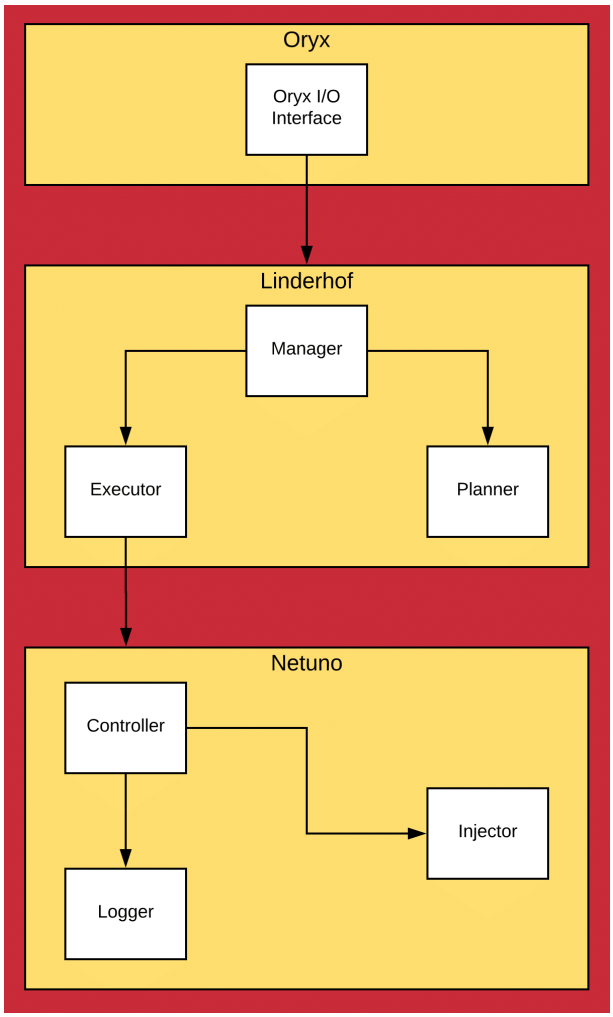

Figura 3.1: Arquitetura da ferramenta Linderhof.

3.1.2 Commander

O módulo Commander é o núcleo da ferramenta. Ele é responsável por criar o planejamento do ataque de acordo com a estrutura passada pelo Oryx e chamar o injetor Netuno com os argumentos necessários para executar o ataque. Esse módulo possui outros quatro submódulos: o *Planner*, responsável por montar a estrutura de plano de ataque de acordo

com o espelho chamado; o Executor, que executa o espelho escolhido; o *Manager*, gerencia o *Planner* e o Executor, funcionando como uma ponte entre eles, além de validar a estrutura do Oryx; e o *Hall of mirrors*, contém as funções de chamada para cada ataque, chamamos essas funções de espelhos, ou *mirror*.

3.1.3 Netuno

Esse módulo é o injetor de pacotes da ferramenta Linderhof. Ele foi implementado para conseguir uma injeção de pacotes constante no ataque. É dividido em três submódulos: o *Controller*, que controla a taxa de injeção; o *Injector*, responsável por criar e destruir as *threads* injetoras; e o Logger, que apenas produz o *log* do ataque em execução.

O nível do ataque corresponde à quantidade de pacotes que o Netuno envia por segundo, esse parâmetro é passado através do argumento chamado no módulo Oryx. A quantidade de pacotes por segundo segue a Equação 3.1.

$$Q = 10^{(L-1)} \tag{3.1}$$

Onde Q é a quantidade de pacotes por segundo e L o nível do ataque. A Tabela 3.1 apresenta os níveis presentes no Linderhof.

Tabela 3.1: Níveis permitidos pela ferramenta Linderhof.

Nível	**Quantidade de Pacotes por Segundo**
1	1
2	10
3	100
4	1000
5	10000
6	100000
7	1000000
8	10000000
9	100000000
10	1000000000
>10	>1000000000

3.2 Espelho DNS

O espelho DNS foi adicionado a ferramenta Linderhof utilizando seus módulos de injeção e planejamento do ataque. A construção dos campos foi inspirada nos pacotes que o comando *dig* gera. O valor preenchido de cada campo usado para se montar o pacote está exibido na Tabela 3.2.

Os campos de 1 ao 15 são espaços reservados do cabeçalho DNS, todo pacote desse protocolo possui esses campos preenchidos. Nessa área é definido que esse pacote é uma consulta com uma pergunta recursiva, e também, que aceita a resposta com os adicionais, se houver. Os campos 16, 17 e 18 são da área de questões, nesse local se define o domínio a ser consultado, e o tipo, que é o *any*. Nos campos restantes, do 19 ao 25, são da área *additional*, nesse local é definido que a consulta usará o DNSSEC, e também, que a maior resposta do UDP será de 4096, o espaço para os campos adicionais foi seguido segundo a RFC 2671 [16].

Tabela 3.2: Campos do cabeçalho DNS usados na ferramenta Linderhof.

Campo	**Nome**	**Valor**
1	ID	Aleatório
2	QR	0
3	OPCODE	0
4	AA	0
5	TC	0
6	RD	1
7	RA	0
8	Z	0
9	AD	1
10	CD	0
11	RCODE	0
12	QDCOUNT	1
13	ANCOUNT	0
14	NSCOUNT	0
15	ARCOUNT	1
16	QUESTION	Domínio consultado
17	TYPE	255
18	NAME	0
19	TYPE	41
20	CLASS	4096
21	RCODE	0
22	EDNS0	0
23	Z	32768
24	RDLEN	12
25	RDATA	Aleatório

Capítulo 4

Resultados

Neste capítulo será apresentado como foram feitos os testes de ataque DDoS com a ferramenta Linderhof, e também será divulgado os resultados obtidos com diferentes amplificações.

4.1 Configurações dos equipamentos

Para a simulação do ataque deste trabalho, todos os testes foram realizados em ambiente controlado. Para tal, foi necessário criar uma rede interna sem conexão à rede *Internet*, onde havia somente 3 computadores e um roteador. O papel do atacante é somente enviar os pacotes com o IP mascarado para o refletor, o refletor tem a função de responder as *queries* enviadas pelo atacante, e por último, o alvo recebe todas as respostas já com a amplificação. As especificações de cada dispositivo estão nas tabelas: Tabela 4.1, as configurações do atacante; Tabela 4.2, do refletor; Tabela 4.3, do alvo; e a Tabela 4.4, do roteador. A Figura 4.1 ilustra a topologia usada na simulação.

Apesar de se utilizar um roteador para os testes, não foi usada a função de roteamento. Assim, o roteador se comporta como um *switch* em toda a simulação, podendo-se usar um próprio *switch* para estudos futuros.

Tabela 4.1: Configurações do equipamento do atacante.

Descrição	**Configuração**
Processador	Intel i3-M370
Memória	4GB DDR3 1333MHz
Interface de rede	100Mbps
Sistema Operacional	Lubuntu 19.04

Tabela 4.2: Configurações do equipamento do refletor.

Descrição	**Configuração**
Processador	Intel i5-8250U
Memória	8GB DDR4 2400MHz
Interface de rede	1Gbps
Sistema Operacional	Lubuntu 19.04

Tabela 4.3: Configurações do equipamento do alvo.

Descrição	**Configuração**
Processador	Intel i5-4210U
Memória	8GB DDR3 1600MHz
Interface de rede	100Mbps
Sistema Operacional	Lubuntu 19.04

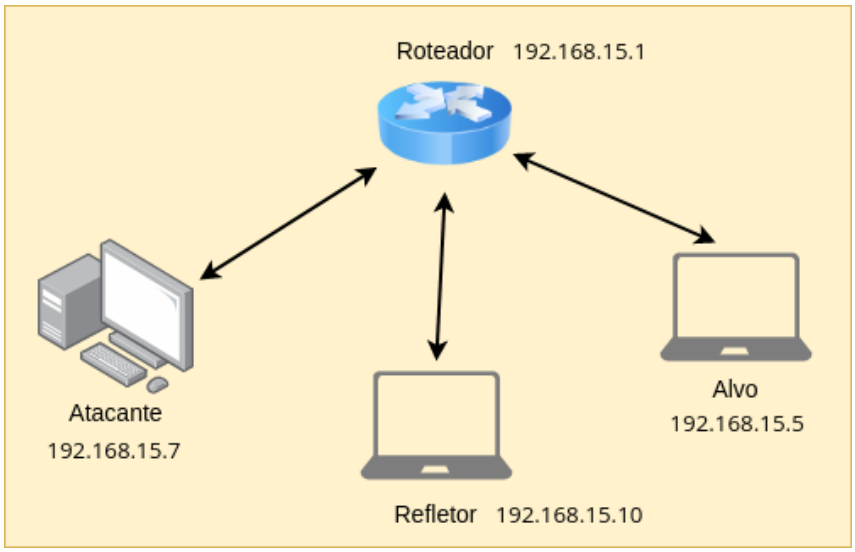

Figura 4.1: Topologia usada no ataque.

Tabela 4.4: Configurações do roteador.

Descrição	**Configuração**
Fabricante	Askey
Modelo	RTF3505VW-N2
Velocidade das *Interfaces*	1Gbps

4.2 Servidor DNS

Para simular um resolvedor DNS foi utilizado o *software Bind* [17], na sua versão 9. Foram feitas várias configurações para deixar o servidor acessível a todos os dispositivos da rede em simulação. Seguindo a RFC 1035 [10], em */etc/bind* nas configurações do arquivo *named.conf.option*, os parâmetros *listen-on*, *allow-transfer*, *allow-query*, *allow-recursion* foram abertos para os endereços da rede 192.168.0.0/16, e em *recursion* foi colocada a opção *yes*. Ou seja, o servidor está aceitando consultas recursivas de todo os IP que estão dentro dessa rede. Já as zonas DNS estão em */var/lib/bind*, foram criadas três zonas com diferentes tamanhos, a menor é a zona ddos.dns.com, a segunda é ddos2.dns.com e a maior ddos3.dns.com. Isso, para gerar diferentes amplificações e avaliar a saturação em cada.

4.3 Execução do ataque e Coleta de Dados

O ataque consiste em duas etapas para cada a amplificação diferente. A primeira etapa foi realizada uma espécie de *benchmark*, gerou-se o ataque somente por 5 segundos para cada nível e amplificação, com isso, pode-se analisar quantos pacotes cada máquina consegue enviar e/ou receber, e assim, observar se houve alguma saturação no atacante, refletor ou alvo, resultando em um total de 30 simulações. A segunda etapa foi um ataque incremental de 50 segundos para cada amplificação, passando em cada nível por 5 segundos, dessa forma, é possível se analisar um teste mais próximo de um ataque feito em ambiente real, e também, os recursos de cada máquina, gerando em um total de 3 simulações. Todos os dados foram capturados com o analisador de rede *tshark* [18], e a análise das capturas foram feitas na ferramenta *Wireshark* [19].

Como dito anteriormente, foram criadas três zonas diferentes no servidor DNS com diferentes quantidade de registros, sendo assim, as diferentes amplificações se dá pelo número de pacotes recebidos a uma única consulta, pois a *Maximum Transmission Unit (MTU)* máxima é de 1500 bytes. Sabendo que a *query* tem um pouco menos de 100 bytes, na amplificação mais baixa, consultando o domínio ddos.dns.com, o servidor consegue responder tudo em somente um pacote, resultando em um amplificação de 15 vezes, na segunda amplificação, o domínio ddos2.dns.com é consultado, a resposta é fragmentada em dois pacotes obtendo um ampliação de 30 vezes, e a maior, cujo domínio é o ddos3.dns.com, consegue-se no máximo uma amplificação de 45 vezes, respondendo tudo em três pacotes.

4.3.1 Amplificação de 15 vezes

Nessa configuração o domínio a ser consultado, ddos.dns.com, foi construído com a quantidade de dados para que a resposta fosse dada com somente um pacote, sem fragmentação. Ou seja, a resposta é próxima do limite da MTU, que foi de 1408 bytes.

Atacante

Como o atacante mascara o IP de origem, logo, ele somente envia pacotes no ataque, não recebe pacotes de nenhuma fonte. Na Tabela 4.5 mostra a quantidade de pacotes enviados em cada nível, e também, a quantidade em bytes por segundo.

No ataque incremental, foram 50 segundos passando por todos os níveis constantemente. Na Figura 4.2 ilustra a saída de pacotes por segundo do atacante, e na Figura 4.3, a quantidade em bytes.

Tabela 4.5: Quantidade de pacotes enviados pelo atacante por segundo na amplificação de 15.

Nível	**Pacotes Ferramenta**	**Pacotes Real**	**Bytes**
1	1	1	96
2	10	10	960
3	100	100	9600
4	1000	1000	96000
5	10000	9999	959904
6	100000	31324	3007104
7	1000000	103867	9971232
8	10000000	104155	9998880
9	100000000	103712	9956352
10	1000000000	103681	9953376

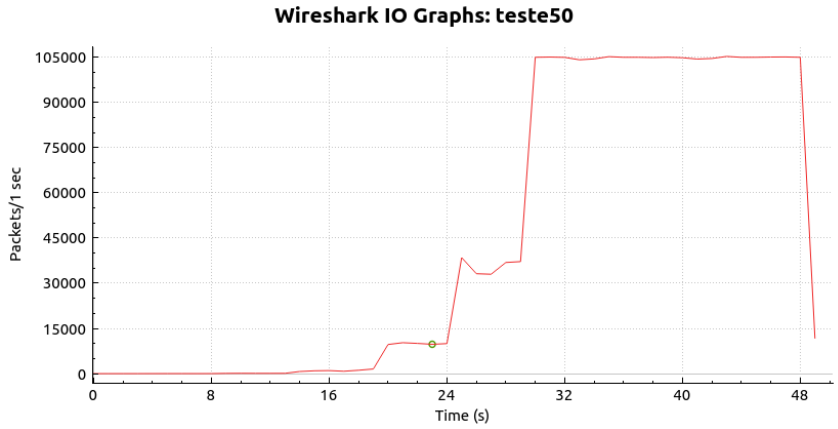

Figura 4.2: Pacotes enviados pelo atacante por segundo na amplificação de 15 no ataque incremental.

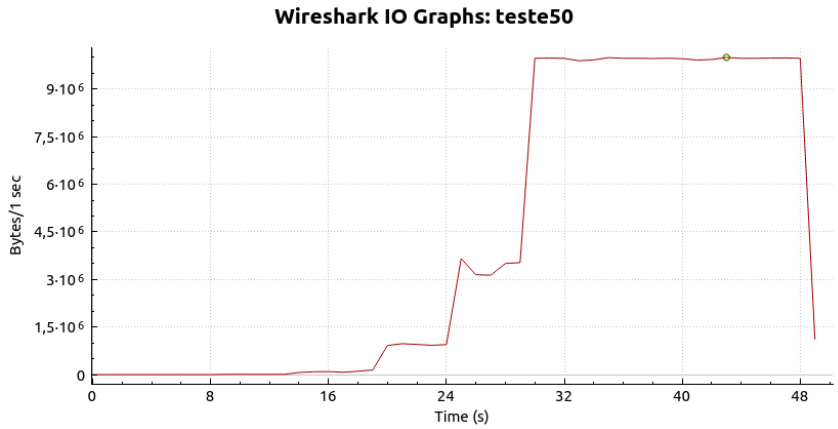

Figura 4.3: Bytes enviados pelo atacante por segundo na amplificação de 15 no ataque incremental.

Refletor

Já no refletor é possível analisar os pacotes recebidos com os enviados. Nessa amplificação o refletor amplia a resposta em termos de tamanho mas não em número de pacotes, ou seja, quando ele recebe uma *query*, responde com somente um pacote em tamanho maior. Na Tabela 4.6 é possível observar a quantidade de pacotes que saem e entram no refletor por segundo. E na Figura 4.4 exemplifica a entrada e saída de pacotes por segundo do refletor no processo contínuo, na Figura 4.5 em bytes, onde a curva verde é a entrada e a vermelha a saída.

Tabela 4.6: Quantidade de pacotes recebidos e enviados pelo refletor por segundo na amplificação de 15.

Nível	Pacotes Recebidos	Bytes Recebidos	Pacotes Enviados	Bytes Enviados
1	1	96	1	1408
2	10	960	10	14080
3	100	9600	100	140800
4	993	95328	797	1122176
5	9679	929184	3778	5319424
6	30006	2880576	13487	18989696
7	99196	9522816	42500	59840000
8	98928	9497088	43835	61719680
9	99108	9514368	43405	61114240
10	99079	9511584	44128	62132224

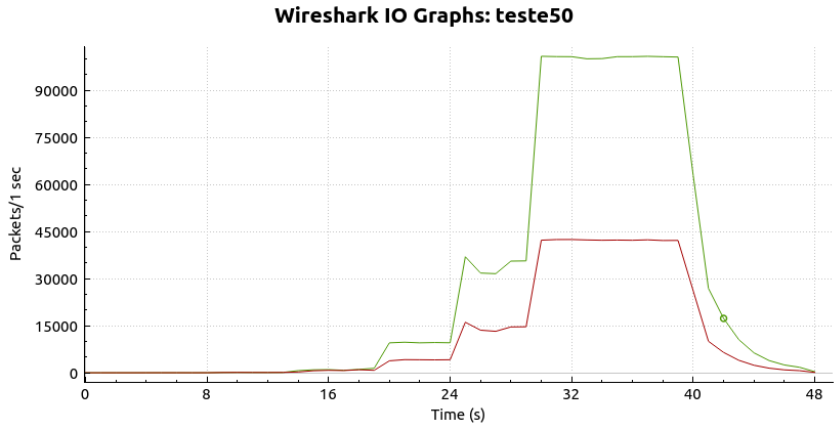

Figura 4.4: Pacotes recebidos e enviados pelo refletor por segundo na amplificação de 15 no ataque incremental.

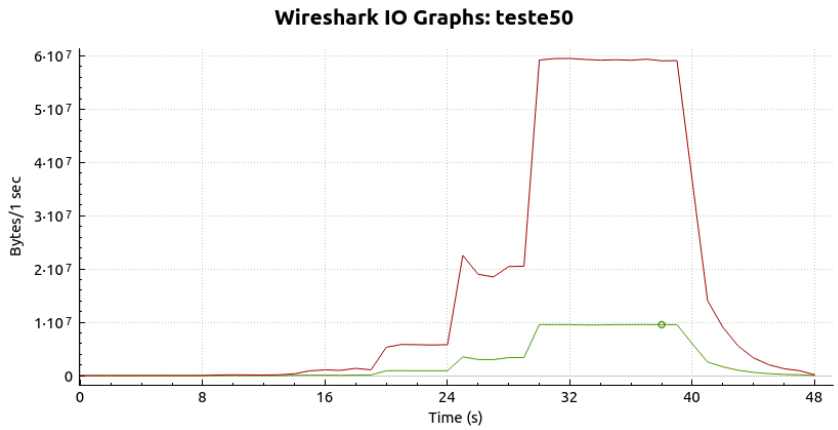

Figura 4.5: Bytes recebidos e enviados pelo refletor por segundo na amplificação de 15 no ataque incremental.

Alvo

Na vítima, somente há pacotes recebidos, pois quem envia os pacotes para o alvo receber é o atacante. Os pacotes recebidos no alvo já está amplificado em 15 vezes. Na Tabela 4.7 pode-se reparar a quantidade de pacotes que chegam na vítima por segundo. Na Figura 4.6 mostra entrada de pacotes por segundo do alvo no processo incremental, Na Figura 4.7 em bytes, onde verde é a entrada.

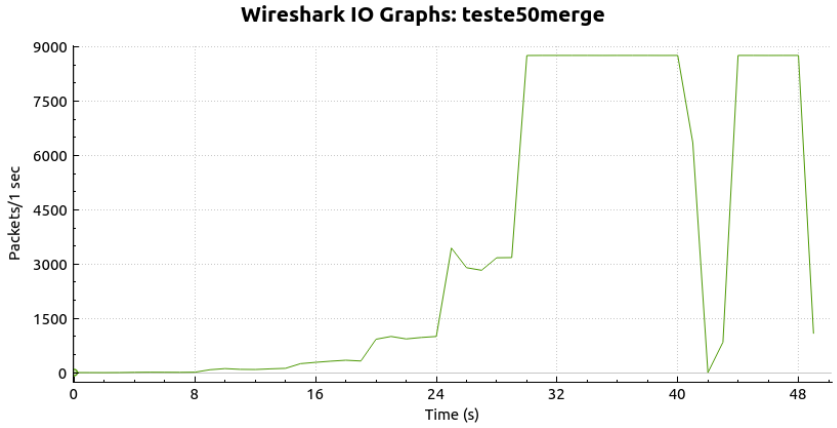

Figura 4.6: Pacotes recebidos pelo alvo por segundo na amplificação de 15 no ataque incremental.

Tabela 4.7: Quantidade de pacotes recebidos pelo alvo por segundo na amplificação de 15.

Nível	Pacotes Recebidos	Bytes Recebidos
1	1	1408
2	10	14080
3	100	140800
4	403	567424
5	1024	1441792
6	2832	3987456
7	8726	12286208
8	8731	12293248
9	8724	12283392
10	8731	12293248

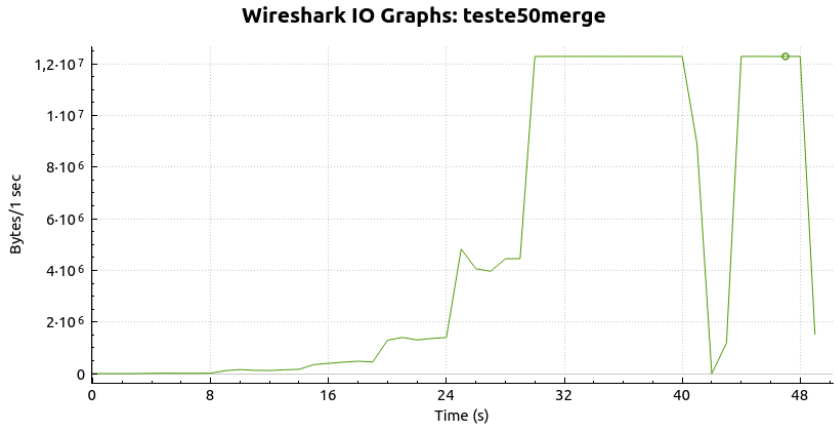

Figura 4.7: Bytes recebidos pelo alvo por segundo na amplificação de 15 no ataque incremental.

A Figura 4.8 demonstra a quantidade de pacotes recebidos e/ou enviados pelo atacante, refletor e alvo na amplificação de 15 em escala logarítmica e a Figura 4.9, em bytes.

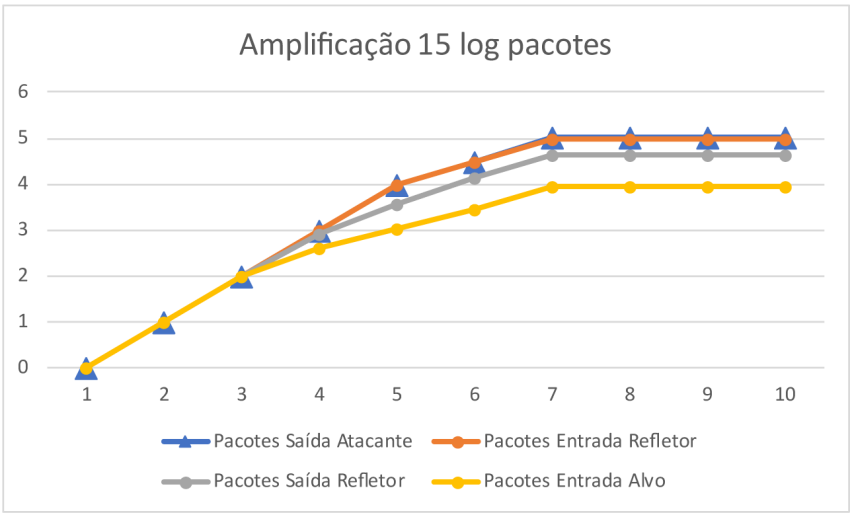

Figura 4.8: Pacotes por segundo na amplificação de 15 em escala logarítmica.

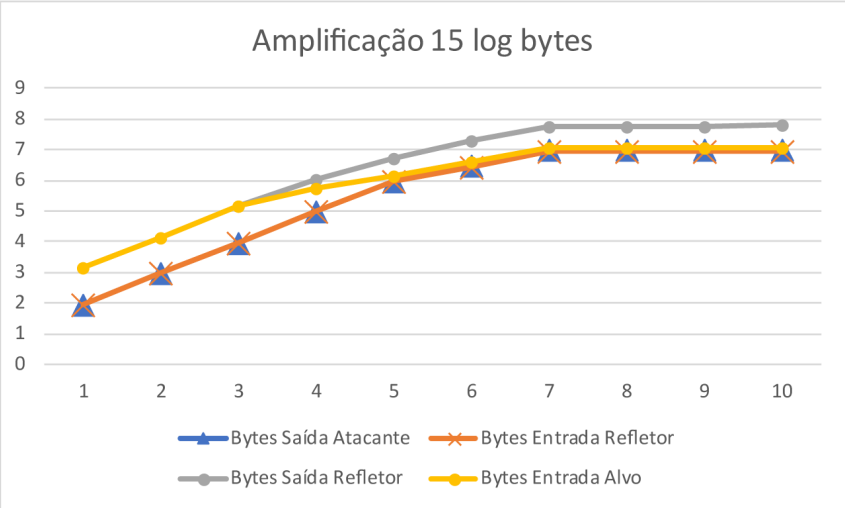

Figura 4.9: Bytes por segundo na amplificação de 15 em escala logarítmica.

4.3.2 Amplificação de 30 vezes

Nessa amplificação, foi consultado o domínio ddos2.dns.com, que foi construído para responder tudo em dois pacotes, sendo um fragmentado e o outro a resposta em si, chegando próximo de 3000 bytes, e assim, resultando em uma ampliação de 30 vezes.

Atacante

O atacante se mantém como a simulação anterior, dado que, ele continua enviando o mesmo único pacote. A Tabela 4.8 apresenta a quantidade de pacotes por segundo que o atacante consegue enviar ao refletor. Na Figura 4.10 mostra saída de pacotes por segundo do atacante no processo incremental, na Figura 4.11 em bytes, a curva vermelha é a saída.

Tabela 4.8: Quantidade de pacotes enviados pelo atacante por segundo na amplificação de 30.

Nível	Pacotes Ferramenta	Pacotes Real	Bytes
1	1	1	96
2	10	10	960
3	100	100	9600
4	1000	1000	96000
5	10000	10000	960000
6	100000	34140	3277440
7	1000000	104549	10036704
8	10000000	103560	9941760
9	100000000	103743	9959328
10	1000000000	103737	9958752

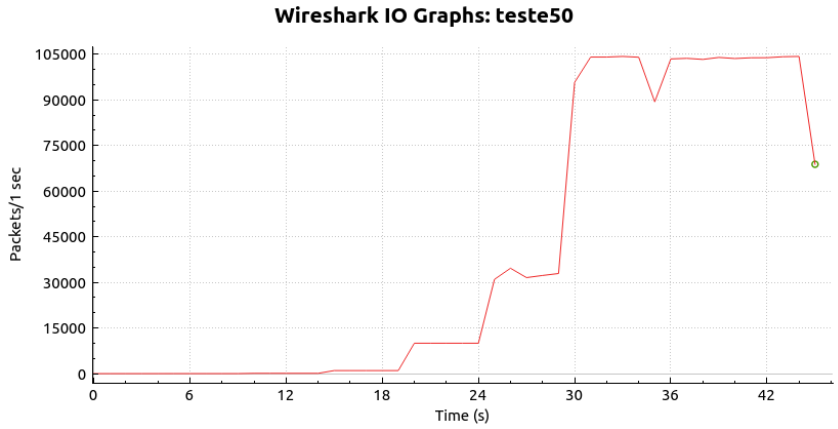

Figura 4.10: Pacotes enviados pelo atacante por segundo na amplificação de 30 no ataque incremental.

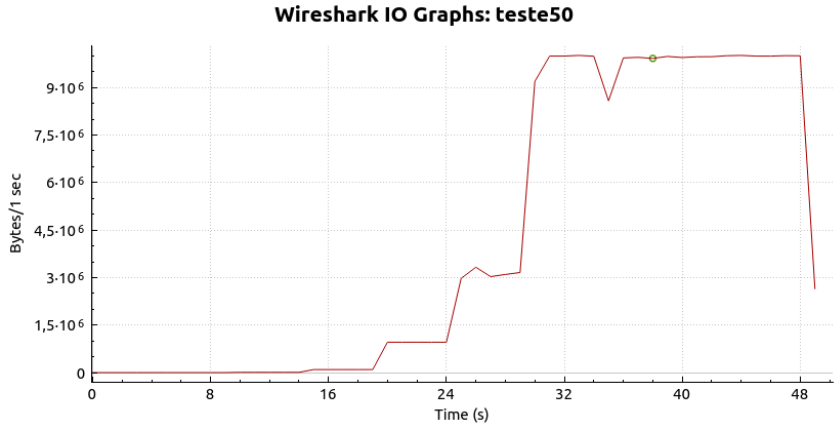

Figura 4.11: Bytes enviados pelo atacante por segundo na amplificação de 30 no ataque incremental.

Refletor

Novamente no refletor encontra-se o maior número de pacotes, pois ele responde ao atacante com dois novos pacotes. A Tabela 4.9 demonstra a quantidade de pacotes que saem e entram no refletor por segundo. E na Figura 4.12 exemplifica a entrada e saída de pacotes por segundo do refletor no processo contínuo, na Figura 4.13, onde a curva verde é a entrada e a vermelha a saída.

Tabela 4.9: Quantidade de pacotes recebidos e enviados pelo refletor por segundo na amplificação de 30.

Nível	Pacotes Recebidos	Bytes Recebidos	Pacotes Enviados	Bytes Enviados
1	1	96	2	2976
2	10	960	20	29760
3	100	9600	200	297600
4	966	92736	1366	2032608
5	9487	910752	5876	8743488
6	32224	3093504	18336	27283968
7	98482	9454272	54020	80381760
8	97475	9357600	54782	81515616
9	97767	9385632	54700	81097500
10	97659	9375264	54500	81096000

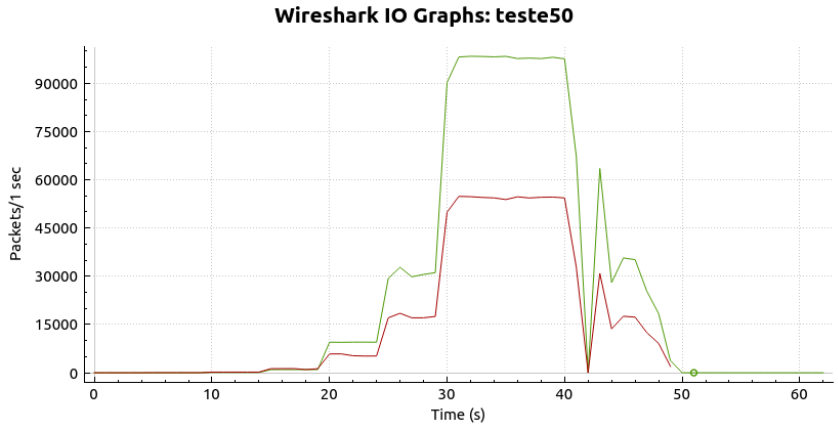

Figura 4.12: Pacotes recebidos e enviados pelo refletor por segundo na amplificação de 30 no ataque incremental.

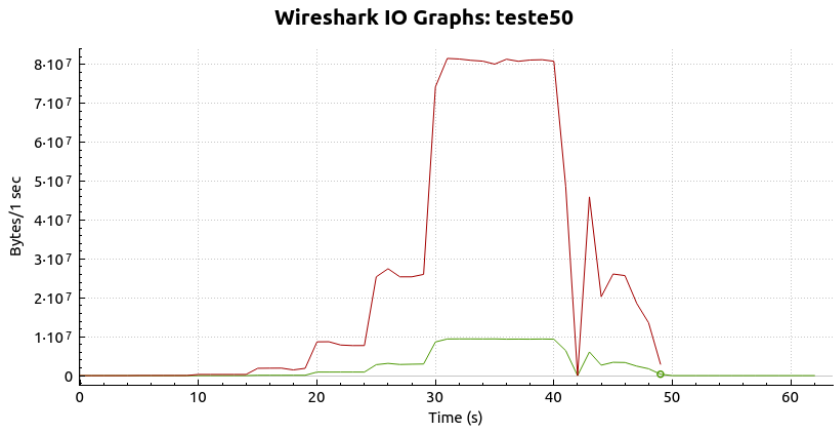

Figura 4.13: Bytes recebidos e enviados pelo refletor por segundo na amplificação de 30 no ataque incremental.

Alvo

Nesse evento, a vítima deverá receber dois pacotes do refletor para cada resposta. Na Tabela 4.10 mostra-se a quantidade de pacotes que chegam ao alvo por segundo. Na Figura 4.14 mostra entrada de pacotes por segundo no alvo no processo incremental, e na Figura 4.15 em bytes, a curva verde é a entrada.

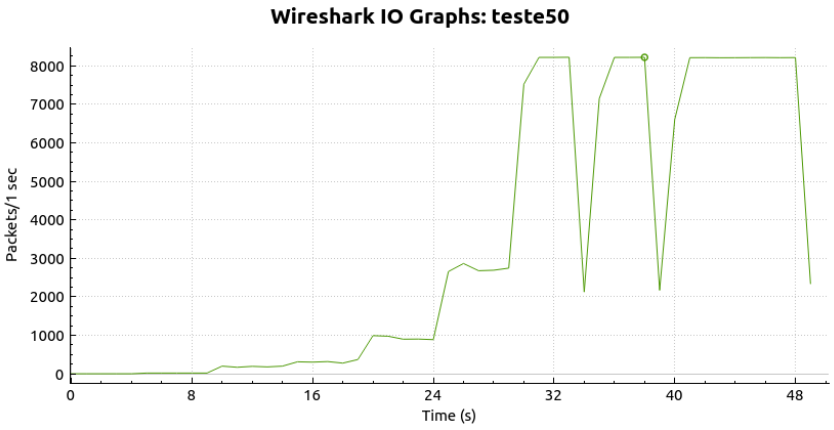

Figura 4.14: Pacotes recebidos pelo alvo por segundo na amplificação de 30 no ataque incremental.

Tabela 4.10: Quantidade de pacotes recebidos pelo alvo por segundo na amplificação de 30.

Nível	Pacotes Recebidos	Bytes Recebidos
1	2	2976
2	20	29760
3	161	239594
4	359	536454
5	886	1326532
6	2867	4288066
7	8226	12305132
8	8211	12283722
9	8225	12304762
10	8223	12301006

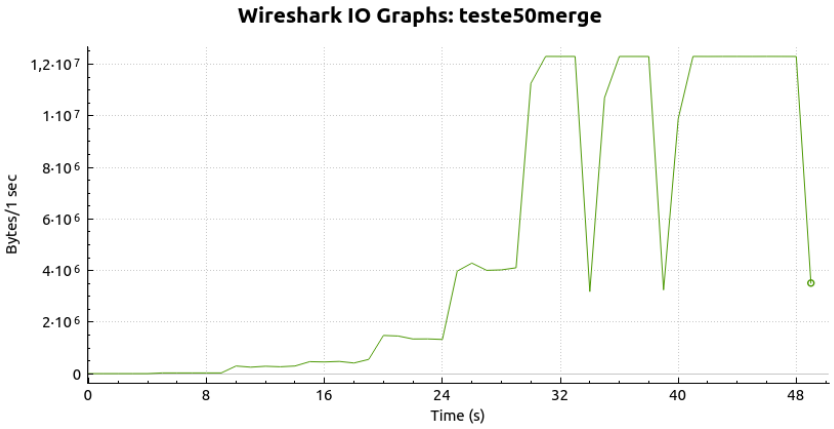

Figura 4.15: Bytes recebidos pelo alvo por segundo na amplificação de 30 no ataque incremental.

A Figura 4.16 demonstra a quantidade de pacotes recebidos e/ou enviados pelo atacante, refletor e alvo na amplificação de 30 em escala logarítmica e a Figura 4.17, em bytes.

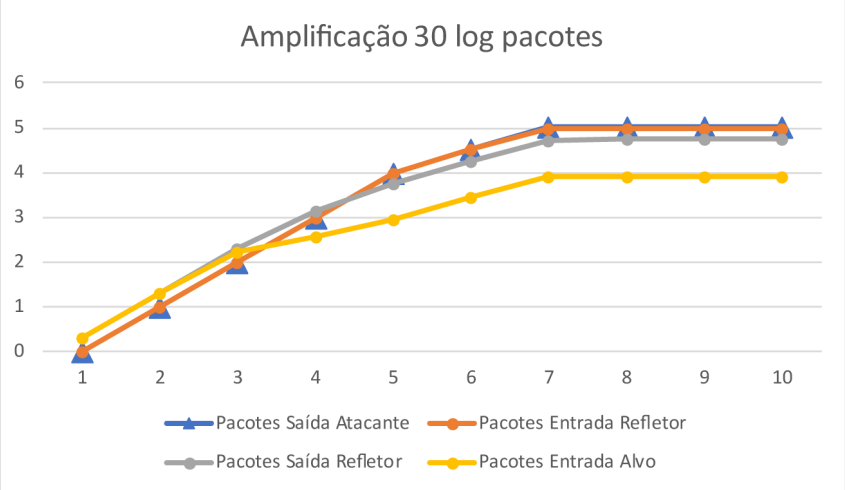

Figura 4.16: Pacotes por segundo na amplificação de 30 em escala logarítmica.

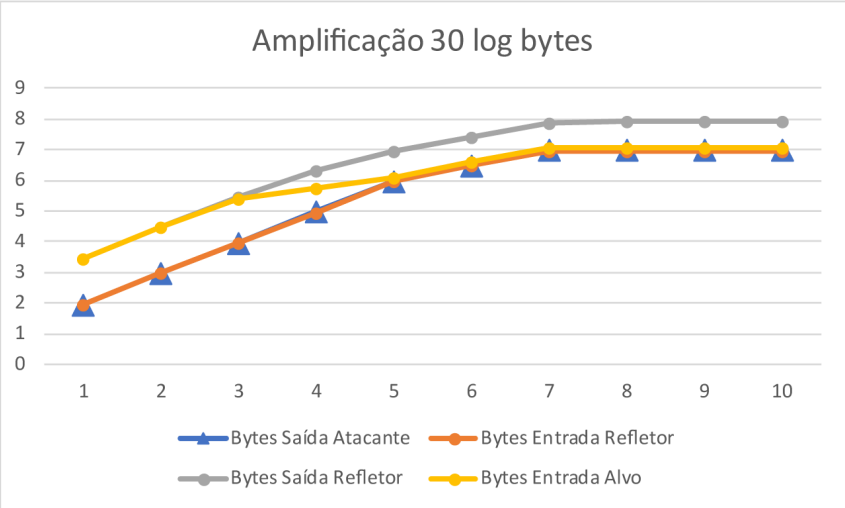

Figura 4.17: Bytes por segundo na amplificação de 30 em escala logarítmica.

4.3.3 Amplificação de 45 vezes

Essa amplificação é a máxima que se conseguiu com o servidor DNS implementado, próximo de 45 vezes, a resposta são dois pacotes fragmentados, e um terceiro pacote que é a resposta no protocolo DNS, resultando em um total de 4206 bytes.

Atacante

O comportamento do atacante nessa amplificação é parecida com a das outras amplificações, pois o trabalho dele é o mesmo independente da ampliação do sinal. A Tabela 4.11 demonstra a quantidade de pacotes por segundo que o atacante consegue enviar ao refletor. Na Figura 4.18 mostra saída de pacotes por segundo do alvo no processo incremental, e na Figura 4.19 em bytes, a curva vermelha é a saída.

Tabela 4.11: Quantidade de pacotes enviados pelo atacante por segundo na amplificação de 45.

Nível	Pacotes Ferramenta	Pacotes Real	Bytes
1	1	1	96
2	10	10	960
3	100	100	9600
4	1000	1000	96000
5	10000	10000	960000
6	100000	35612	3383140
7	1000000	104462	9923890
8	10000000	105427	10015565
9	100000000	105094	9983930
10	1000000000	104604	9937380

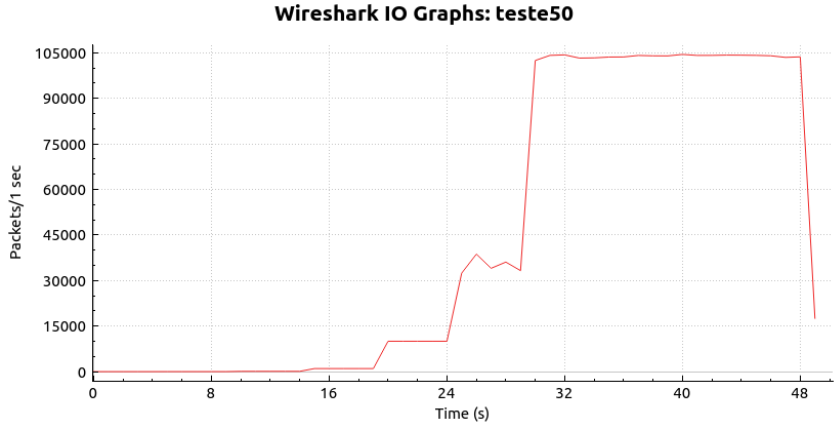

Figura 4.18: Pacotes enviados pelo atacante por segundo na amplificação de 45 no ataque incremental.

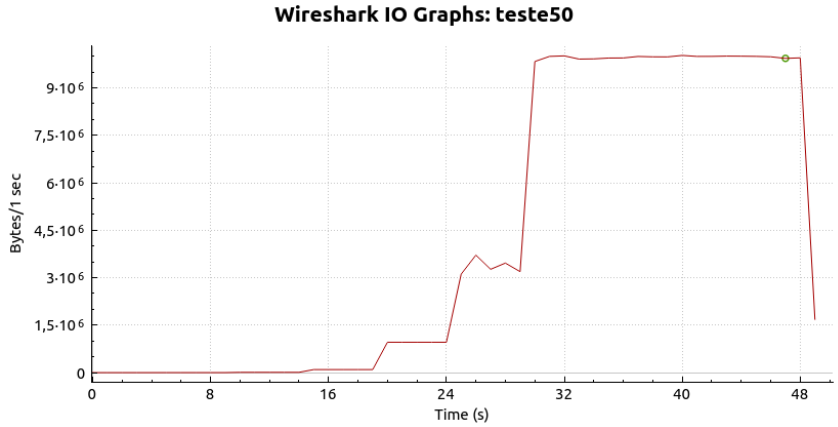

Figura 4.19: Bytes enviados pelo atacante por segundo na amplificação de 45 no ataque incremental.

Refletor

No refletor é onde há mais tráfego, pois ele recebe os pacotes do atacante, processa as informações e é responsável por responder três pacotes ao alvo. A Tabela 4.12 demonstra a quantidade de pacotes que saem e entram no refletor por segundo. E na Figura 4.20 exemplifica a entrada e saída de pacotes por segundo do refletor no processo contínuo, e na Figura 4.21 em bytes, onde a curva verde é a entrada e a vermelha a saída.

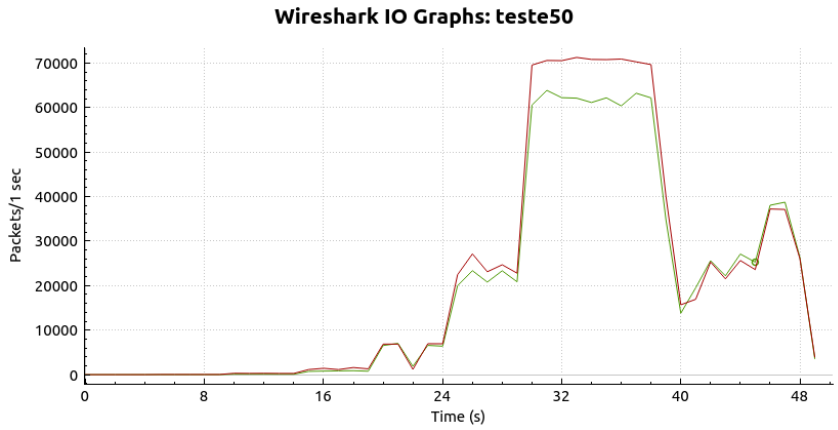

Figura 4.20: Pacotes recebidos e enviados pelo refletor por segundo na amplificação de 45 no ataque incremental.

Tabela 4.12: Quantidade de pacotes recebidos e enviados pelo refletor por segundo na amplificação de 45.

Nível	Pacotes Recebidos	Bytes Recebidos	Pacotes Enviados	Bytes Enviados
1	1	96	3	4206
2	10	960	30	42060
3	100	9600	300	420600
4	798	75810	1683	2359566
5	6770	643150	7254	10170108
6	21695	2061025	25077	35157954
7	62377	5925815	73467	103000734
8	62408	5928760	72552	101717904
9	59063	5610985	72762	102012324
10	60255	5724225	72852	102138504

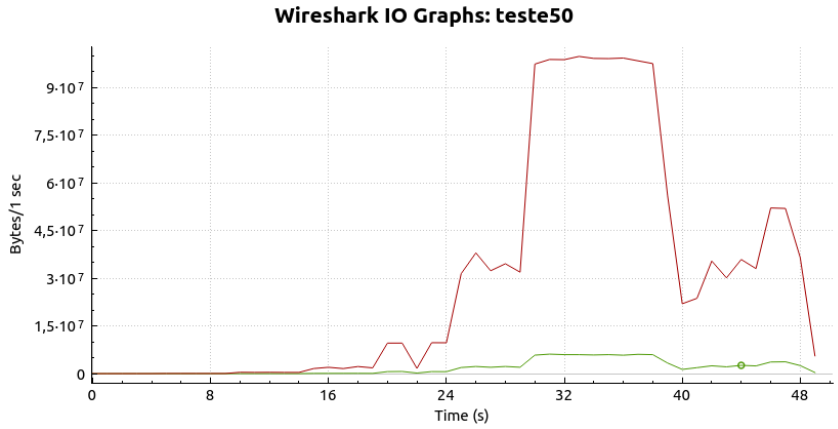

Figura 4.21: Bytes recebidos e enviados pelo refletor por segundo na amplificação de 45 no ataque incremental.

Alvo

Nesse caso, a vítima deverá receber três pacotes do refletor para cada resposta, e também, montar esses pacotes para tornar a resposta válida. Na Tabela 4.13 observa-se a quantidade de pacotes que chegam no alvo por segundo. Na Figura 4.22 mostra a chegada de pacotes por segundo no alvo no processo incremental, e na Figura 4.23 em bytes, a curva verde é a entrada.

Tabela 4.13: Quantidade de pacotes recebidos pelo alvo por segundo na amplificação de 45.

Nível	Pacotes Recebidos	Bytes Recebidos
1	3	4206
2	30	42060
3	196	274904
4	354	496644
5	1033	1432922
6	3205	4341314
7	9073	12282650
8	9042	12253188
9	9050	12282436
10	9076	12281816

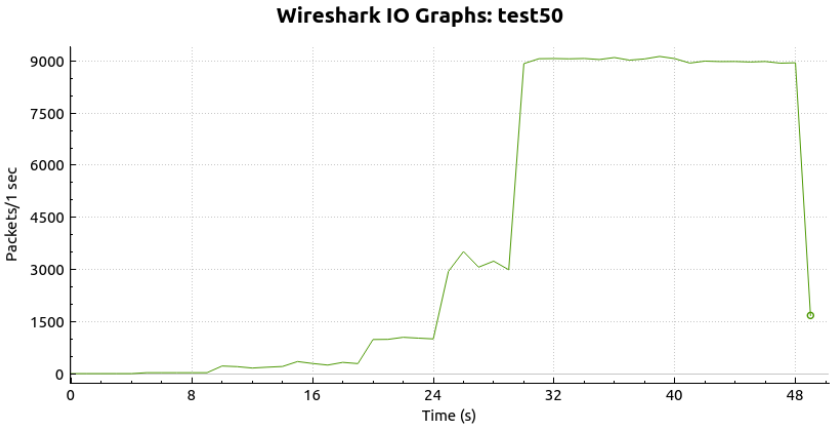

Figura 4.22: Pacotes recebidos pelo alvo por segundo na amplificação de 45 no ataque incremental.

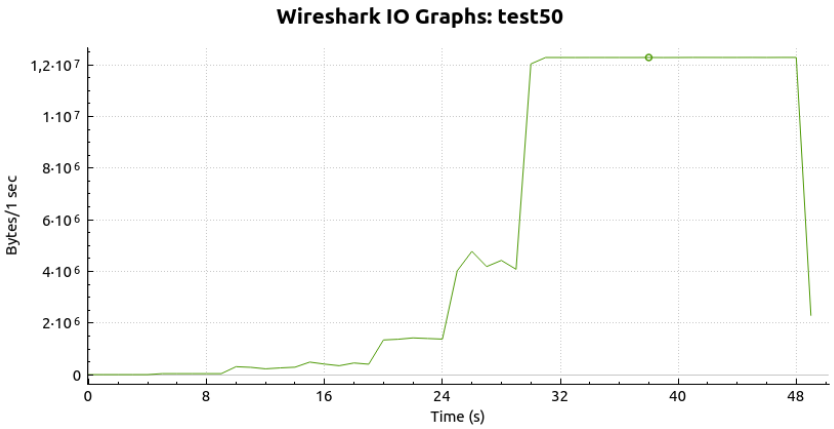

Figura 4.23: Bytes recebidos pelo alvo por segundo na amplificação de 45 no ataque incremental.

A Figura 4.24 demonstra a quantidade de pacotes recebidos e/ou enviados pelo atacante, refletor e alvo na amplificação de 45 em escala logarítmica e a Figura 4.25, em bytes.

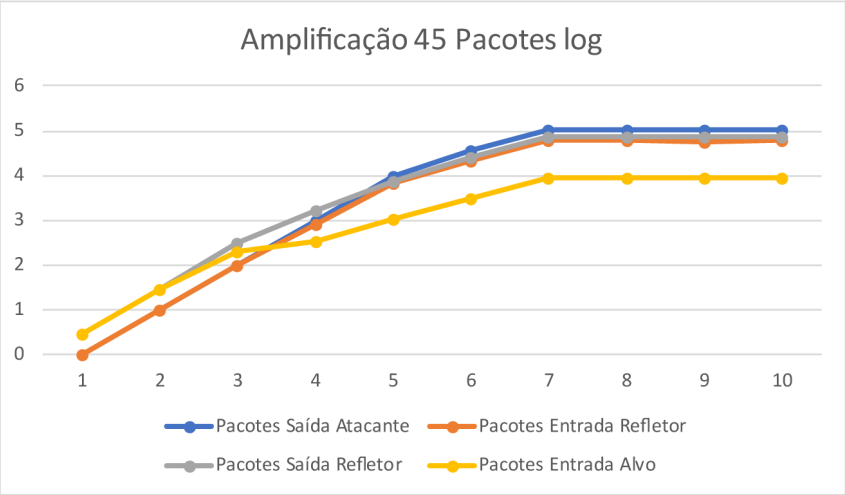

Figura 4.24: Pacotes por segundo na amplificação de 45 em escala logarítmica.

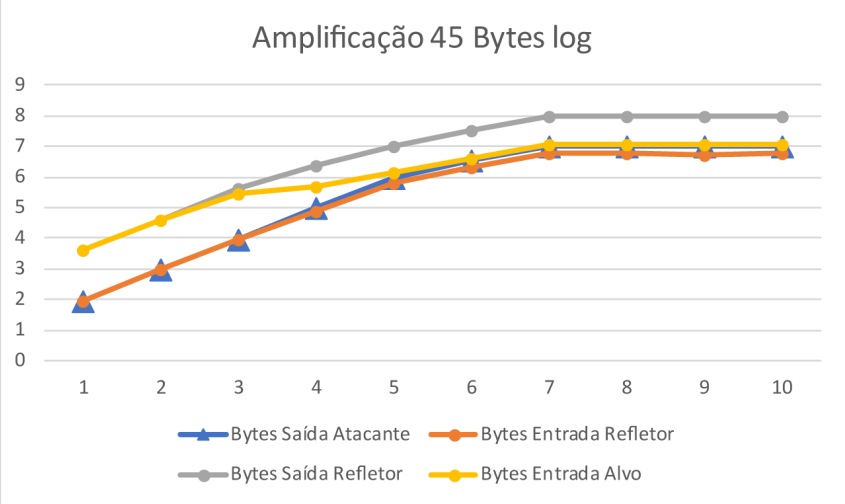

Figura 4.25: Bytes por segundo na amplificação de 45 em escala logarítmica.

4.4 Amplificação

Nessa seção será calculada a amplificação real do ataque, ou seja, a diferença de bytes e pacotes entre o alvo e o atacante. Foi pego a quantidade de bytes e pacotes por segundo para cada nível e amplificação recebida pelo alvo e dividida pela quantidade enviada pelo atacante. A Equação 4.1 mostra como foi feito o cálculo da amplificação em bytes e em pacotes.

$$A = Q(vítima)/Q(atacante) \tag{4.1}$$

Onde A é amplificação, Q(vítima) é a quantidade recebida pela vítima em bytes ou

Tabela 4.14: Amplificação real do ataque em bytes.

Nível	**Amplificação de 15**	**Amplificação de 30**	**Amplificação de 45**
1	14,67	31	43,81
2	14,67	31	43,81
3	14,67	24,95	28,63
4	5,91	5,58	5,17
5	1,5	1,38	1,52
6	1,32	1,30	1,28
7	1,23	1,22	1,23
8	1,22	1,23	1,22
9	1,23	1,23	1,23
10	1,23	1,23	1,23

Tabela 4.15: Amplificação real do ataque em pacotes.

Nível	**Amplificação de 15**	**Amplificação de 30**	**Amplificação de 45**
1	1	2	3
2	1	2	3
3	1	1,61	1,96
4	0,4	0,35	0,35
5	0,1	0,08	0,08
6	0,09	0,08	0,08
7	0,08	0,07	0,08
8	0,08	0,07	0,08
9	0,08	0,07	0,08
10	0,08	0,07	0,08

pacotes por segundo e Q(atacante) é a quantidade enviada pelo atacante em bytes ou pacotes por segundo.

A tabela Tabela 4.14 apresenta a amplificação em bytes do ataque em números, para uma melhor visualização a Figura 4.26 exibe graficamente. Já a Tabela 4.15 indica a amplificação de pacotes numericamente e a Figura 4.27 graficamente.

A Figura 4.28 apresenta a amplificação em bytes do ataque em escala logarítmica e a Figura 4.29 exibe em pacotes.

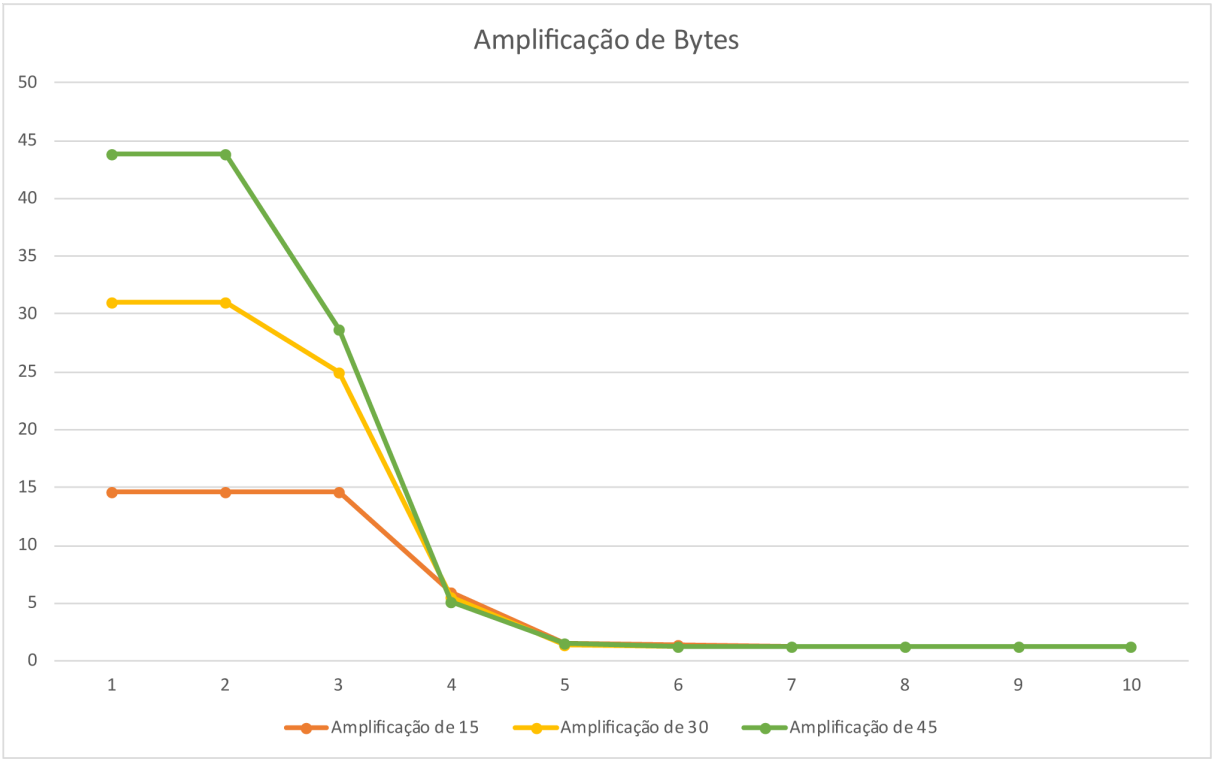

Figura 4.26: Gráfico da amplificação real do ataque em bytes.

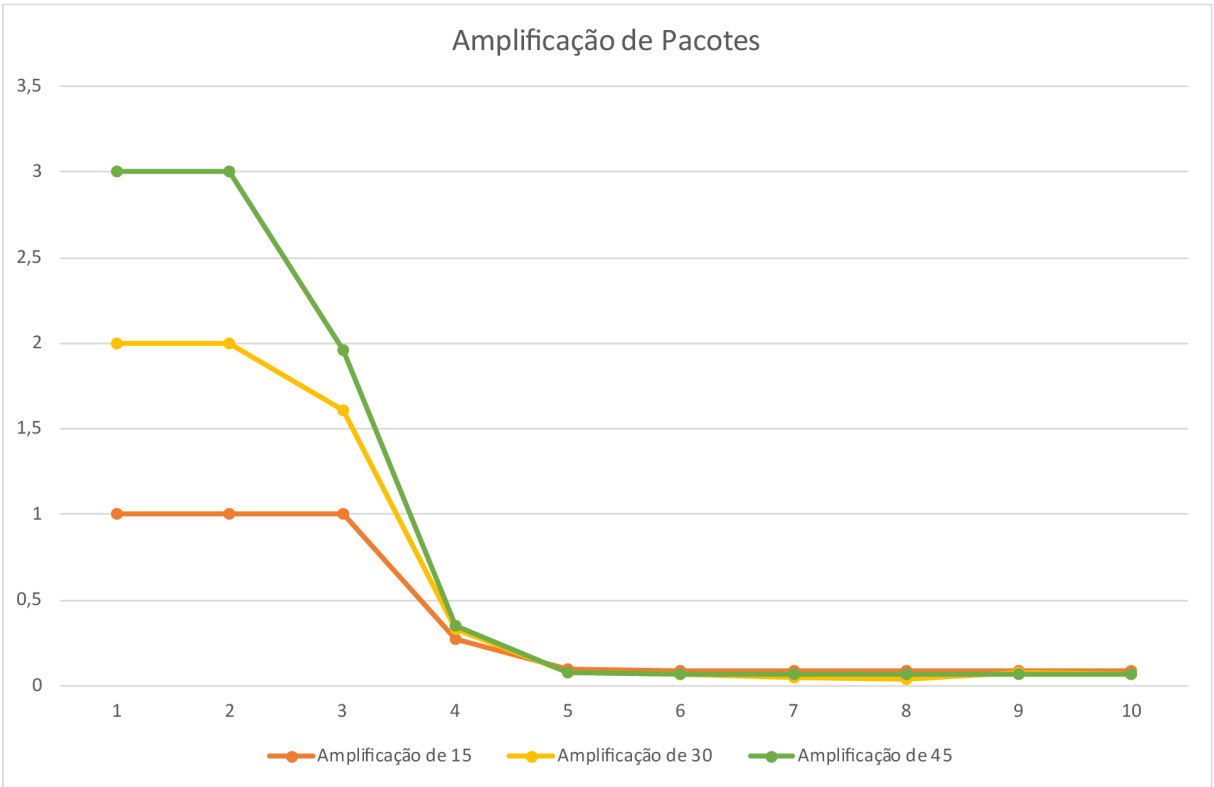

Figura 4.27: Gráfico da amplificação real do ataque em pacotes.

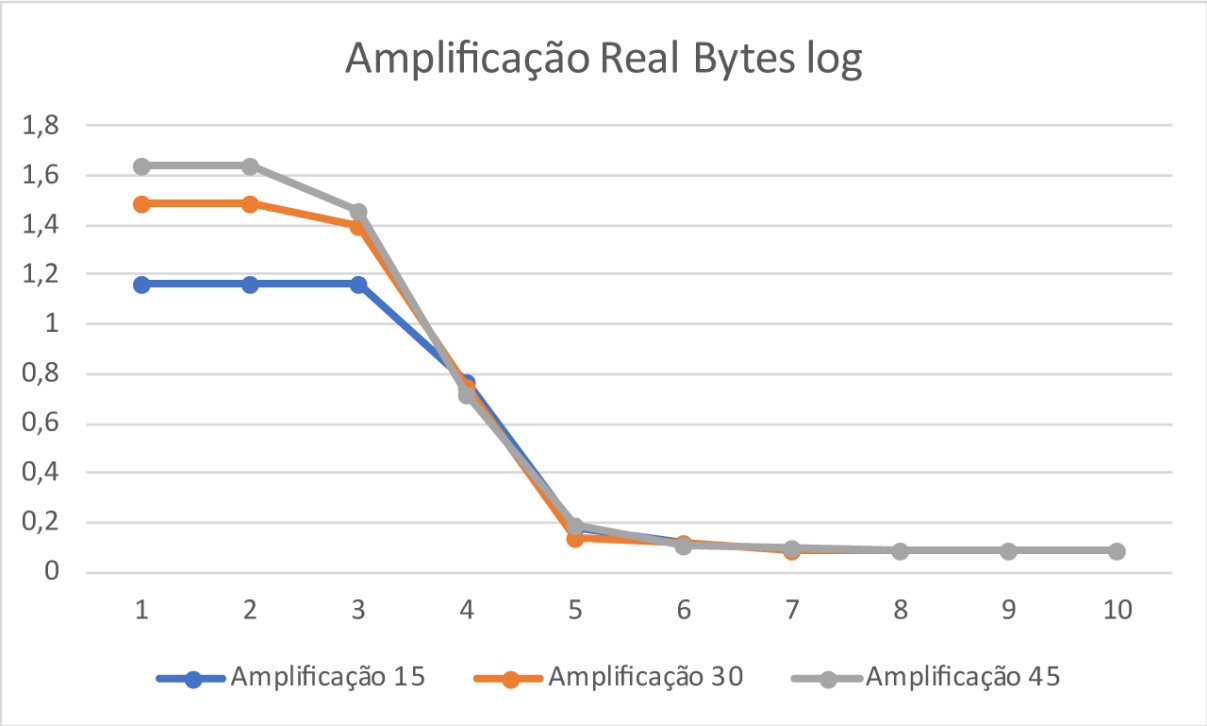

Figura 4.28: Gráfico da amplificação real do ataque em bytes em escala logarítmica.

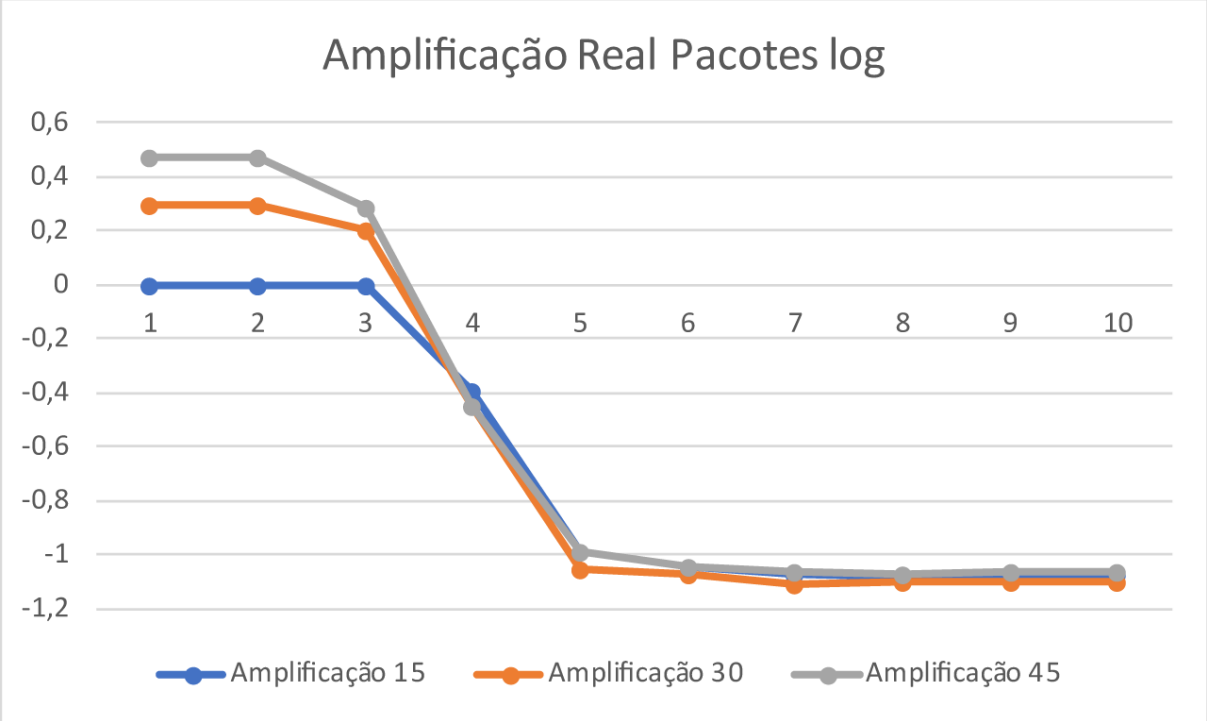

Figura 4.29: Gráfico da amplificação real do ataque em pacotes em escala logarítmica.

4.5 Análise

Nessa seção será tratada sobre as análises dos dados coletados no trabalho, analisaremos a diferença do ataque incremental para o teste de *benchmark*, a amplificação ganha no ataque de forma geral, e também, dos recursos de cada máquina.

4.5.1 Análise dos Testes

Primeiramente, houve uma espécie de teste do número de pacotes e bytes recebidos e/ou enviados por cada máquina para cada nível, e depois, um teste mais real onde houve um ataque de 50 segundos passando por todos os níveis.

Essas duas simulações não tiveram tanta discrepância entre elas, inclusive, o refletor nos testes executados mantém sua anormalidade em receber mais pacotes do que envia nos níveis mais altos, entretanto, há uma grande amplificação no número em bytes. Além disso, pode-se observar uma pequena elevação do número de pacotes e bytes por segundo no alvo nas amplificações 30 e 45.

4.5.2 Análise da Amplificação

Observa-se que a amplificação do DDoS usando o protocolo DNS não gera muitos pacotes comparando-se com outros protocolos, como é o caso do NTP, mas os pacotes gerados são muito maiores que a pergunta em tamanho de bytes. A maior ampliação de pacotes no ataque foi de 3. Já na amplificação medida em bytes chegou-se em próximo de 45, mas em níveis mais baixos. Além disso, é notável a saturação do ataque começando no nível 3, e praticamente não existir amplificação do nível 5 em diante.

4.5.3 Análise dos Recursos

Também foi feita uma análise dos recursos de cada máquina, usando a ferramenta HTOP [20], com o comando *echo q | htop | aha –black –line-fix > teste.html*, foram salvos em vários momentos do ataque contínuo esses dados, para cada amplificação. Similarmente, foi monitorado a *interface* de rede de cada máquina usando o *software* BMON [21], com o comando *bmon -o ascii | tee teste.txt*, foram gravados a taxa de RX, entrada de dados, e TX, saída de dados, das *interfaces* de cada dispositivo. A seguir serão divulgados os dados coletados no maior tráfego do ataque, que é na amplificação de 45 nos níveis de 7 para acima.

Tabela 4.16: Tráfego da *interface* de rede do atacante na amplificação de 45.

Taxa	Pacotes por segundo	Bytes por segundo
RX	0	0
TX	104.26K	9.55M

Atacante

Na Figura 4.30 apresenta os recursos da máquina do atacante nos níveis mais altos. E a Tabela 4.16 mostra o tráfego de rede da *interface*.

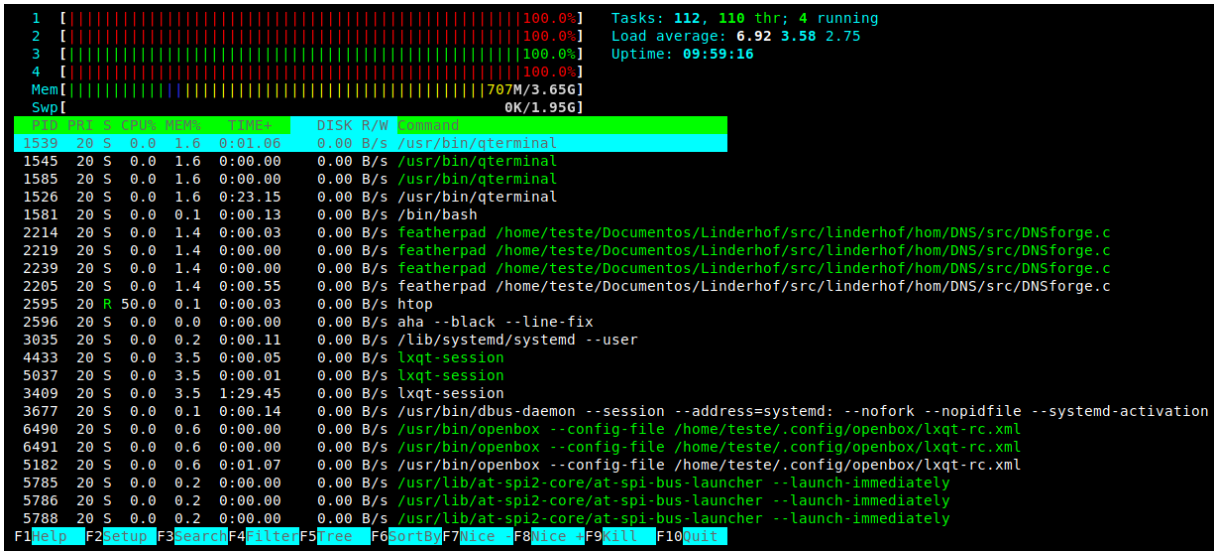

Figura 4.30: Recursos do atacante na amplificação de 45.

Refletor

Na Figura 4.31 apresenta os recursos da máquina do refletor nos níveis mais altos. E a Tabela 4.17 mostra o tráfego de rede da *interface*.

Tabela 4.17: Tráfego da *interface* de rede do refletor na amplificação de 45.

Taxa	Pacotes por segundo	Bytes por segundo
RX	62.33K	5.70M
TX	69.86K	93.32M

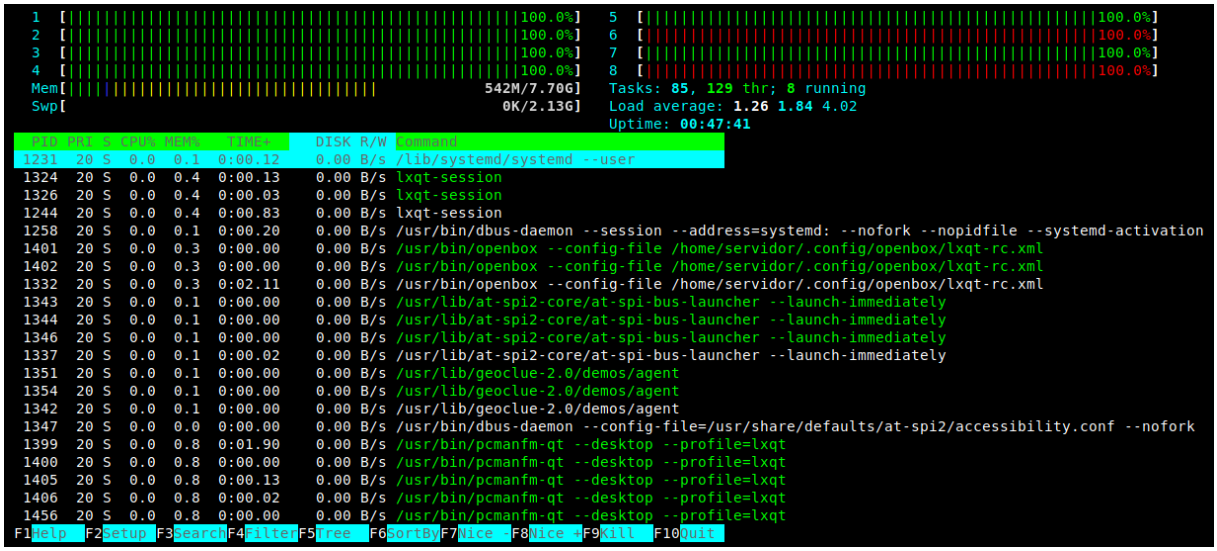

Figura 4.31: Recursos do refletor na amplificação de 45.

Alvo

Na Figura 4.32 apresenta os recursos da máquina do alvo nos níveis mais altos. E a Tabela 4.18 mostra o tráfego de rede da *interface*.

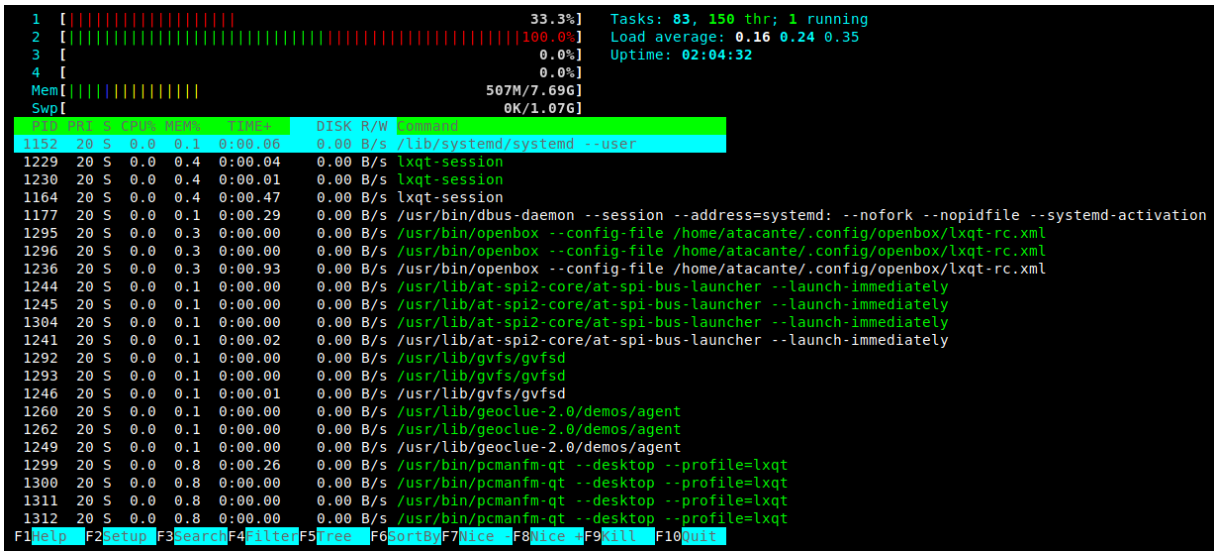

Figura 4.32: Recursos do alvo na amplificação de 45.

Tabela 4.18: Tráfego da *interface* de rede do alvo na amplificação de 45.

Taxa	Pacotes por segundo	Bytes por segundo
RX	8.98K	11.72M
TX	0	0

Na *software* HTOP foi classificado para mostrar os processos que mais consomem entradas e saídas de disco, então, é possível verificar o consumo de disco, do processador e memória nas barras.

Primeiramente, nota-se que no atacante, os pacotes analisados no *wireshark* condiz com as taxas na *interface* registradas pelo BMON. No refletor há uma pequena diferença entre as duas análises, no BMON registrou-se um pouco mais de pacotes entrando e um pouco menos saindo. Já no alvo, observa-se um pouco mais de pacotes e bytes entrando registrados pelo BMON do que pelo analisador de pacotes.

Com isso, nota-se que a memória e o disco são pouco usados em todos os dispositivos durante o ataque. O processador chega em 100% no atacante e no refletor, já no alvo não se aplica. A *interface* de rede chega ao seu máximo de transferência no caso do refletor e no alvo, mas não no atacante.

4.6 Considerações Finais

Não se encontrou um *software* capaz de registrar os dados de todos os recursos da máquina durante o ataque, em texto e em tempo real, para se construir uma tabela mais precisa de recursos por cada nível. Então, com o HTOP os dados foram registrados em alguns momentos durante o ataque, já com o BMON conseguiu-se armazenar as taxas a cada segundo.

A amplificação chegou-se ao máximo de 45 vezes pois foi a maior resposta conseguida no protocolo UDP, que é de no máximo 4096 bytes. Mesmo enviando o pacote com o campo *class*, onde se diz o tamanho do *payload*, maior que 4096, e também, com as configurações *max-udp-size* e *edns-udp-size* maiores que 4096 no servidor BIND. Quando fez-se a consulta com essas configurações o servidor forçou responder no protocolo TCP, tanto com a ferramenta criada, quanto com o comando *dig*.

Para a melhor *performance* do Linderhof, monta-se um pacote e salva-se na memória, assim, é possível enviar mais de 100 mil pacotes por segundo. A ferramenta não monta cada pacote todas as vezes que os envia. Logo, todos os pacotes enviados recebem o mesmo valor no campo ID. Mas isso não influenciou as simulações do ataque, pois o DNS é um protocolo *stateless*, ou seja, não se guarda o estado das informações. Independente da identificação da *query* recebida, o resolvedor de nomes a responderá.

Capítulo 5

Conclusão e Trabalhos Futuros

5.1 Conclusão

Neste trabalho foi possível analisar a viabilidade do protocolo DNS para ataques distribuídos de negação de serviço por reflexão amplificada, onde faz-se necessário estudar como eles funcionam para uma melhor defesa.

Inicialmente, foi apresentado como os ataques DDoS funcionam e quais os tipos existentes. Depois foi explicado como o protocolo *Domain Name System* é implementado, e também, os campos do cabeçalho, assim como a consulta *any*, essencial para a compreensão do ataque. Da mesma maneira, houve uma breve explicação do motivo de ser usado o protocolo UDP.

Também foi descrito como a ferramenta Linderhof opera, descrevendo seus módulos e os comandos necessários para ela funcionar. Do mesmo modo, foi detalhado como o espelho DNS foi construído e quais *flags* foram usadas. Assim, é possível concluir que essa ferramenta é bastante versátil e robusta, sendo viável adicionar mais espelhos a ela.

Nos resultados, é possível se analisar como a ferramenta criada se comporta, se realmente ela gera alguma amplificação e se há alguma saturação, temos as seguintes conclusões:

Nos primeiros testes, é notável a similaridade das duas simulações, as resposta obtidas no *benchmark* são bem próximas as do ataque incremental, ou seja, se o atacante tiver condições pode testar qual será a amplificação do refletor antes do ataque em si, e assim, escolher a melhor ampliação do sinal para o melhor efeito do ataque.

A amplificação do DDoS mostrou ser alta, chegando a aproximadamente 45 vezes. Mas isso no nível 2, onde a taxa de pacotes é baixa. Nos níveis onde se esperava maior amplificação com maiores taxa de pacotes se mostrou o contrário, praticamente não houve ampliação. Isso demonstra que para a melhor *performance* do ataque não exige tanto esforço dos *bots*, ou seja, pode ser feito com dispositivos mais simples.

Além disso, foi possível se verificar os recursos de cada dispositivo, foi notado que a memória e o disco não sofrem com os efeitos do ataque, agora o processador e a *interface* de rede são bastante exigidos. Logo, conclui-se que provavelmente a limitação de envio de pacotes do atacante é definida pelo processador, já que a taxa de bytes enviados chega próximo de 10, e o máximo seria 12,5M devido a sua *interface* de 100Mbps. Por ser aonde há mais tráfego, no refletor o processador e a *interface* são altamente solicitados, fazendo com que ele receba menos pacotes em comparação ao envio do atacante nos níveis mais altos e envie menos respostas do que realmente foi solicitado, chegando a próximo de 100M o tráfego de rede capturado pelo monitor de rede. Por isso, a saturação do refletor está no processador e na *interface* de rede, que chega próximo de 125M pois ela permite conexões de até 1000Mbps. Por último, o alvo chega a usar uma quantidade considerável do processador, mas com poder de processamento ainda sobrando, e a *interface* fica no seu limite com quase 12M de bytes por segundo, bem próximo do máximo suportado, que é de até 100Mbps.

O DDoS com reflexão amplificada simulado mostrou o que outros estudos similares mostram, que há uma saturação nesse tipo de ataque. Assim como nos outros protocolos já estudados, Network Time Protocol (NTP) [22], Simple Service Discovery Protocol (SSDP)[23], Simple Network Management Protocol (SNMP)[24], e Memcached[25], também houve uma saturação a partir dos níveis 3 ou 4, e praticamente sem ampliação do ataque nos níveis mais altos.

Pensando no lado do atacante, para se ter o melhor efeito possível do ataque, deve-se buscar servidores com bastante dados nas zonas para serem consultados, e realizar-se o ataque no nível 2, onde há a maior ampliação. Mas para isso é fundamental ter muitos *bots*, pois o a taxa nesse nível é baixa. Se o ataque for realizado nos últimos níveis, a vítima também ficará com a sua banda sobrecarregada mas seria praticamente nula a amplificação, tornando o ataque em um DDoS comum.

Finalmente, pensando no lado do defensor, as técnicas de mitigação são mais complexas, um alternativa é verificar se tá chegando muitas respostas sem que ele mesmo tenha perguntado a um resolvedor de nomes. Outra maneira é bloquear de alguma maneira as respostas do tipo *any* do protocolo DNS. Para o refletor, pode-se permitir consultas do tipo *any* somente para IP confiáveis, ou de redes internas, na configuração *allow-query*, e também, limitar o *payload* do UDP para um valor baixo.

5.2 Trabalhos Futuros

Para se estudar mais a fundo no futuro os ataques distribuídos de negação de serviço por reflexão amplificada, alguns pontos são sugeridos a seguir:

- **Mais dispositivos na simulação:** Realizar um estudo com mais *bots*, e também, refletores, para que haja uma maior similaridade com os ataques DDoS.
- **Mais recursos nos dispositivos:** Usar servidores, pois têm maior poder computacional para se processarem os dados, e principalmente mais *interfaces* de rede nos refletores, e com velocidades maiores.
- **Melhor monitoramento dos recursos:** Usar alguma ferramenta com uma melhor verificação dos recursos, assim como o *sofware* Wireshark funciona, registrando os recursos a cada segundo, tornando possível se analisar em cada nível. Além do mais, monitorar os recursos também do roteador ou switch usado.
- **Aperfeiçoamento do Linderhof:** Um bom estudo seria manipular a ferramenta Linderhof para usar a cada determinado tempo um protocolo diferente. E também, detectar possíveis refletores em uma rede.

Referências

[1] AKAMAI: *State of the internet / security: Ddos and application attacks*, 2019. 1

[2] AKAMAI: *Pmemcached-fueled 1.3 tbps attacks*, Fevereiro 2018. `https://blogs.akamai.com/2018/03/memcached-fueled-13-tbps-attacks.html`. 1

[3] A10: *The state of ddos weapons*, Junho 2019. 2

[4] Cloudflare: *Denial-of-service (dos).* `https://www.cloudflare.com/learning/ddos/glossary/denial-of-service/`. 4

[5] Cloudflare: *What is a ddos attack?* `https://www.cloudflare.com/pt-br/learning/ddos/what-is-a-ddos-attack/`. 5, 6

[6] Paxson, Vern: *An analysis of using reflectors for distributed denial-of-service attacks*. ACM SIGCOMM Computer Commun. Rev., 31, 2001. 6

[7] Rossow, Christian: *Amplification hell: Revisiting network protocols for ddos abuse*. Proceedings of the 21st Annual Network and Distributed System Security Symposium, NDSS 2014, San Diego, CA, USA, 2014. 7

[8] Kurose, James e Keith Ross: *Computer Networking: A Top-Down Approach*. Pearson, 7ª edição, 2016. 7, 8, 9

[9] IETF: *Domain names - concepts and facilities*, Novembro 1987. `https://www.ietf.org/rfc/rfc1034.txt`. 7

[10] IETF: *Domain names - implementation and specification*, Novembro 1987. `https://www.ietf.org/rfc/rfc1035.txt`. 7, 20

[11] Liu, Cricket e Paul Albitz: *DNS and BIND*. O'Reilly Media, 5ª edição, 2006. 8

[12] INACON: *Dns message format.* `https://www.inacon.de/ph/data/DNS/DNS-Message-Format_OS_RFC-1035.htm`. 10

[13] IETF: *Domain name system security extensions*, Março 1999. `https://tools.ietf.org/html/rfc2535`. 11

[14] IETF: *Transmission control protocol*, Setembro 1981. `https://www.rfc-editor.org/rfc/rfc793.txt`. 12

[15] IETF: *User datagram protocol*, Agosto 1980. `https://tools.ietf.org/html/rfc768`. 12

[16] IETF: *Extension mechanisms for dns (edns0)*, Agosto 1999. http://www.networksorcery.com/enp/rfc/rfc2671.txt. 16

[17] Consortium, Internet Systems: *Bind 9*. https://www.isc.org/bind/. 20

[18] TShark: *Tshark*. https://www.wireshark.org/docs/man-pages/tshark.html. 20

[19] Wireshark: *Wireshark*. https://www.wireshark.org. 20

[20] *Manual htop*. http://manpages.ubuntu.com/manpages/disco/en/man1/htop.1.html. 40

[21] *Manual bmon*. http://manpages.ubuntu.com/manpages/disco/en/man8/bmon.8.html. 40

[22] Souza Vieira, Alexander André de: *Ataque distribuído de negação de serviço por reflexão amplificada usando network time protocol*. Monografia apresentada como requisito parcial para conclusão do Curso de Engenharia da Computação, 2019. 45

[23] Duarte, Eduardo S.: *Ataque distribuído de negação de serviço por reflexão amplificada usando o protocolo simple service discovery protocol*. Monografia apresentada como requisito parcial para conclusão do Curso de Engenharia da Computação, 2018. 45

[24] Medeiros, Tiago Fonseca: *Ataque distribuído de negação de serviço por reflexão amplificada usando simple network management protocol*. Monografia apresentada como requisito parcial para conclusão do Curso de Engenharia da Computação, 2015. 45

[25] Miranda, Igor F.: *Ataque de negação de serviço por reflexão amplificada explorando memcached*. Monografia apresentada como requisito parcial para conclusão do Curso de Engenharia da Computação, 2019. 45