DeMONS++: Utilizando Técnicas de Modelagem de Tráfego no Combate de DDoS via Serviço DeMONS

Vitor Faria Medeiros da Silveira
vitorfaria@ufpr.br
Universidade Federal do Paraná
Curitiba, Paraná, Brasil

João Meyer Muhlmann
joaomeyer@ufpr.br
Universidade Federal do Paraná
Curitiba, Paraná, Brasil

Vinicius Fulber-Garcia
vinicius@inf.ufpr.br
Universidade Federal do Paraná
Curitiba, Paraná, Brasil

Abstract

With the increasing use and popularity of computer networks, data security and online service availability have become priority concerns for academia and industry. One of the significant threats to be prevented, detected, and mitigated is the Distributed Denial of Service (DDoS) attack. With the advancement of new technologies and network paradigms, these attacks have been tackled by innovative solutions such as mitigation services based on Network Function Virtualization (NFV), which was utilized to develop the DeMONS service, a solution for DDoS mitigation. Despite its effectiveness, DeMONS' results can be enhanced by introducing new traffic engineering modules. In this way, this paper presents DeMONS++, an extension of DeMONS that incorporates a strategically positioned traffic shaping module within its service topology. Experiments conducted in simulated environments demonstrated that DeMONS++ improves DDoS mitigation, achieving better performance in metrics such as benign traffic acceptance and user satisfaction in the tested scenarios.

Keywords

REDE, SEGURANÇA, DDOS, NFV, SERVIÇO

1 Introdução

Os ataques do tipo *Distributed Denial of Service* (DDoS) são um dos principais desafios enfrentados por empresas e provedores de serviços, devido ao impacto significativo que podem ter na disponibilidade dos serviços e na experiência do usuário [1]. Esses ataques exploram vulnerabilidades de rede para inundar o tráfego legítimo com requisições falsas, muitas vezes resultando em perdas financeiras e prejuízos de reputação. Um exemplo recente da disseminação dos ataques DDoS foi a ofensiva direcionada à *Cloudflare*, empresa de cibersegurança, que registrou um pico de 3,8 terabits por segundo em outubro de 2024 [2].

Sendo assim, a academia e a indústria têm empregado esforços para criar estratégias e desenvolver soluções que visam prevenir, detectar e mitigar ataques DDoS. Entre as soluções comercialmente mais conhecidas, é possível citar os sistemas de detecção e prevenção de intrusão Snort [3] e Suricata [4]. Esses sistemas analisam o tráfego buscando padrões em pacotes e fluxos que os identifiquem como maliciosos e, potencialmente, como parte de um DDoS.

Também, novos paradigmas de rede vêm sendo explorados como uma forma de estabelecer e suportar serviços de prevenção e mitigação de ataques DDoS. Entre esses paradigmas, destaca-se o de Virtualização de Funções de Rede (*Network Function Virtualization* - NFV) [5], onde diversos serviços do tipo foram propostos, como o VGuard [6], VFence [7] e DeMONS [8].

Particularmente, a solução DeMONS utiliza um serviço virtualizado com uma abordagem híbrida de alocação de capacidade e filtragem de tráfego para enfrentar cenários de ataque DDoS. Essa solução emprega um sistema de reputação que classifica os fluxos como maliciosos (reputação zero), benignos (reputação um) ou em níveis de suspeita (reputação entre zero e um). Essa classificação indica, em cenários de sobrecarga, se um fluxo passará por canais de alta ou baixa prioridade, além de determinar o seu nível de entrega caso esteja em um canal de baixa prioridade — definido pela adoção de elementos de segurança e engenharia de tráfego, como *firewalls* e *traffic policers*.

No entanto, o sistema DeMONS não consegue maximizar a probabilidade de entrega de tráfego benigno ao longo do tempo. Isso ocorre porque ele não possui capacidade de armazenar pacotes para envio tardio em cenários de ataque ou sobrecarga. Uma alternativa para incorporar essa capacidade ao DeMONS é a adição de um módulo de *traffic shaping* (mecanismo de *buffers* para controle da taxa de transmissão do tráfego de rede baseado em prioridade). Nesse módulo, um *buffer* FIFO é adicionado ao sistema e as taxas de armazenamento e esvaziamento dos pacotes são controladas por estratégias de envelhecimento, permitindo um atraso controlado no encaminhamento do tráfego.

Considerando o cenário descrito, este trabalho apresenta o DeMONS++, uma variante do DeMONS que inclui um módulo de *traffic shaping* no conjunto de funções de rede que compõem o serviço de segurança. Essa adição busca, principalmente, melhorias na quantidade e na qualidade das requisições atendidas durante ataques DDoS. Para evidenciar o impacto desse novo módulo, o DeMONS++ foi configurado em diferentes cenários de execução, e os resultados dos testes foram comparados com a solução DeMONS original. Os testes realizados demonstram que o aprimoramento no controle do tráfego de entrada pode aumentar a efetividade do DeMONS++ em relação ao DeMONS em ataques de alta intensidade, preservando a qualidade do serviço e otimizando a alocação de recursos para o tráfego benigno sempre que possível.

O restante deste artigo está organizado como segue. A Seção 2 apresenta conceitos fundamentais relacionados a este trabalho. A Seção 3 descreve brevemente os principais trabalhos relacionados. A Seção 4 detalha a solução DeMONS++, evidenciando características arquiteturais e técnicas. A Seção 5 apresenta cenários de teste, resultados e discussões. Finalmente, a Seção 6 traz considerações finais e indicações de trabalhos futuros.

2 Fundamentação Teórica

Nesta seção, são apresentados os conceitos fundamentais para a compreensão do trabalho. Inicialmente, abordam-se os ataques de negação de serviço na Subseção 2.1; em seguida, o paradigma de

NFV é apresentado na Subseção 2.2; por fim, a solução DeMONS para mitigação de DDoS é detalhada na Subseção 2.3.

2.1 Ataques de Negação de Serviço

Os Ataques de Negação de Serviço têm como principal objetivo sobrecarregar sistemas e redes, tornando-os indisponíveis para uso legítimo. Em particular, os Ataques de Negação de Serviço Distribuídos (*Distributed Denial of Service* - DDoS) utilizam múltiplos dispositivos comprometidos (as chamadas *botnets*) para gerar requisições maliciosas, impedindo que os servidores alvo atendam às solicitações legítimas de usuários benignos [9]. Os DDoS podem ser classificados em duas categorias principais, com base em suas características técnicas: *low-rate* e *high-rate*. Os DDoS *low-rate* se referem a ataques direcionados, onde o atacante envia pacotes estrategicamente elaborados para explorar vulnerabilidades e sobrecarregar sistemas específicos, buscando, também, evitar a detecção por mecanismos de segurança tradicionais [10]. Já os ataques DDoS *high-rate* caracterizam-se por sua abordagem ruidosa, envolvendo o envio de um volume massivo de pacotes para saturar servidores, infraestruturas de rede e até mesmo sistemas de segurança.

Para detectar e mitigar ataques DDoS, soluções como IDS (*Intrusion Detection Systems*) e IPS (*Intrusion Prevention Systems*) são amplamente utilizadas. IDS são sistemas que, posicionados estrategicamente em uma rede, monitoram cópias do tráfego, buscando padrões de comportamento e assinaturas em pacotes e fluxos. Ao identificar atividades suspeitas, eles geram alertas para que ações possam ser tomadas. Esses sistemas não interferem diretamente no tráfego, funcionando como uma ferramenta de monitoramento. Já os IPS, diferentemente dos IDS, operam de forma *inline* com o tráfego, analisando pacotes em tempo real. Eles têm a capacidade de bloquear pacotes maliciosos ou encerrar fluxos considerados perigosos. No entanto, devido ao impacto direto que têm na comunicação de rede, uma configuração inadequada pode levar ao bloqueio de pacotes e fluxos legítimos. Isso torna essencial a validação rigorosa das estratégias de detecção implementadas [11].

Com base nos conceitos apresentados, inúmeras soluções e sistemas foram desenvolvidos ao longo dos anos para detectar e mitigar ataques DDoS. Algumas dessas abordagens fazem uso de tecnologias de rede inovadoras, destacando-se sistemas como VGuard [6], Vfence [7] e DeMONS [8], que estão alinhados ao paradigma de NFV. Essas soluções se beneficiam da flexibilidade e escalabilidade oferecidas pela virtualização de funções de rede, permitindo uma adaptação dinâmica e eficiente para lidar com os desafios impostos por ataques DDoS modernos.

2.2 Virtualização de Funções de Rede

Em redes convencionais, funções essenciais, como roteamento, tradução de endereços e filtragem de tráfego, são geralmente executadas por meio de hardware especializado. O paradigma de Virtualização de Funções de Rede (*Network Function Virtualization* - NFV), no entanto, permite que essas funções sejam implementadas em ambientes virtualizados, eliminando a dependência de dispositivos físicos dedicados e reduzindo significativamente os custos de infraestrutura [12]. Além disso, NFV oferece vantagens como alta flexibilidade, escalabilidade e mobilidade das funções. A capacidade de ajustar e realocar recursos torna dinamicamente o paradigma NFV uma alternativa promissora para responder de forma eficaz a ataques DDoS em redes modernas [6, 13].

Além disso, o paradigma NFV possibilita a criação de serviços virtualizados de rede por meio das Cadeias de Função de Serviço (*Service Function Chains* - SFC) [14]. Esses serviços podem ser configurados especificamente como Cadeias de Serviço de Segurança (*Security Service Chains* - SSC), que integram funções como IDS, IPS e *firewalls* para detectar, prevenir e mitigar ameaças de forma coordenada [15]. A virtualização dessas funções não apenas reduz a necessidade de dispositivos físicos, mas também facilita sua implementação em ambientes de nuvem elástica, simplificando a administração. Isso resulta em uma solução escalável e eficiente para combater ataques, especialmente em cenários dinâmicos e de grande escala [16].

2.3 A Solução de Mitigação de DDoS DeMONS

O DeMONS [8] é uma solução híbrida que combina técnicas de capacidade e filtragem para mitigar ataques DDoS, utilizando o paradigma de NFV para proporcionar um gerenciamento flexível e dinâmico de recursos. Sua arquitetura é composta por cinco módulos principais, implementados como Funções Virtualizadas de Rede (*Virtual Network Functions* - VNFs): o Classificador de Prioridade, responsável por priorizar fluxos com base em critérios predefinidos; o *Firewall*, que realiza a filtragem inicial de pacotes; o Alocador de Fluxos, que gerencia a alocação dinâmica de recursos para fluxos; o *Traffic Policer*, que regula o tráfego conforme uma série de políticas; e o Gerenciador, que orquestra a operação integrada dos demais módulos.

No DeMONS, o tráfego de rede que ingressa no sistema é primeiramente processado pelo módulo Classificador de Prioridade. Esse módulo atribui um valor de prioridade ao fluxo analisado, variando entre 0 e 1. Um valor de 0 indica que o fluxo foi identificado como malicioso, enquanto um valor de 1 representa um fluxo reconhecidamente benigno. Valores intermediários (entre 0 e 1) indicam fluxos pertencentes a uma "zona cinza", cuja reputação é incerta e aplicam-se níveis de suspeita. Fluxos classificados como maliciosos são imediatamente bloqueados pelo próximo módulo, o *Firewall*. Já os fluxos com valores diferentes de 0 passam pelo *Firewall*, que os valida e encaminha ao módulo seguinte, o Alocador de Fluxos, para um processamento mais detalhado.

O módulo Alocador de Fluxos, por sua vez, direciona o tráfego para túneis distintos com base nas prioridades atribuídas aos fluxos ingressantes. São definidos dois túneis principais: um de alta prioridade, destinado a fluxos prioritários, e outro de baixa prioridade. O túnel de alta prioridade é configurado para nunca operar acima de sua capacidade, assegurando um processamento eficiente para fluxos críticos. Já o túnel de baixa prioridade pode operar com sobrecarga. Em situações de sobrecarga nesse túnel, o módulo de Controle de Tráfego (*Traffic Policer*) intervém para alocar a largura de banda disponível de forma a considerar a prioridade dos fluxos, garantindo que fluxos de maior prioridade recebam uma parcela maior dos recursos disponíveis.

O módulo Gerenciador é encarregado pelo provisionamento do sistema DeMONS. Ele monitora continuamente a carga da infraestrutura para determinar quando e como escalar as funções de rede do serviço. Essa escalabilidade pode envolver tanto o aumento

quanto a redução do número de instâncias das funções ou da quantidade de recursos computacionais alocados a cada uma delas. Em cenários de subutilização, o Gerenciador pode optar por desativar o túnel de baixa prioridade, reduzindo o consumo desnecessário de recursos e energia, promovendo eficiência operacional.

O método de alocação de fluxos usado no DeMONS é projetado para maximizar a eficiência no uso dos túneis. Em condições de baixa carga em ambos os túneis, o tráfego é distribuído de forma balanceada entre eles. No entanto, quando o túnel de alta prioridade atinge uma utilização acima de uma porcentagem predefinida, ele passa a operar em "modo seletivo". Nesse modo, um mecanismo de balanceamento é ativado para realocar fluxos de maior prioridade para o túnel de alta prioridade, garantindo que os fluxos mais importantes tenham acesso preferencial aos recursos disponíveis.

Além disso, o DeMONS adota, no módulo *Traffic Policer*, políticas específicas para beneficiar fluxos de maior prioridade no túnel de baixa prioridade. Essas políticas definem a quantidade máxima de tráfego permitida para cada fluxo. Contudo, o sistema de reputação integrado busca reduzir a sobrecarga de forma controlada, aplicando uma taxa mínima de descarte de pacotes, mesmo para fluxos de mais alta prioridade trafegando pelo túnel de baixa prioridade. Essa abordagem evita que poucos fluxos monopolizem a capacidade do túnel, assegurando que fluxos de baixa prioridade tenham a chance de demonstrar comportamento benigno ao longo do tempo, melhorando a sua reputação.

3 Trabalhos Relacionados

O uso do paradigma NFV para implementar serviços de segurança tem atraído crescente atenção tanto da academia quanto da indústria. A seguir, são apresentados e discutidos brevemente alguns dos trabalhos mais relevantes publicados nos últimos anos.

O paradigma NFV ainda enfrenta desafios significativos, como a necessidade de suporte robusto para a implantação rápida e eficiente de topologias de serviço voltadas para segurança. Nesse contexto, [13] propõe uma arquitetura de mitigação de DDoS baseada em NFV, que analisa dados de aplicação e de rede. O elemento central dessa arquitetura é um componente de mapeamento de tráfego, que inspeciona os fluxos para detectar anomalias, classificando e redirecionando pacotes com base na sua natureza (ataque ou serviço). O tráfego legítimo é encaminhado ao sistema de destino, enquanto o tráfego suspeito é redirecionado para uma função de rede especializada em tratamento. Entretanto, a principal limitação dessa abordagem está na dificuldade de estabelecer um processo elástico para alocar recursos de filtragem de tráfego.

Outra solução apresentada na literatura é o VFence, desenvolvido para mitigar ataques de inundação SYN [7]. Baseado em funções virtualizadas escaláveis, o sistema consiste em um despachante e agentes especializados na filtragem de tráfego malicioso. O despachante é responsável por distribuir a carga de tráfego entre os agentes disponíveis, enquanto os agentes inspecionam os fluxos de rede em busca de ameaças. Além disso, os agentes realizam a coordenação do processo de *three-way handshake*. Fluxos autenticados são adicionados a uma lista de permissões, liberando o acesso ao sistema, enquanto os fluxos não autenticados são bloqueados.

O CoFence [17] foi projetado para viabilizar a colaboração entre diferentes domínios administrativos no combate a ataques de inundação SYN. Por meio dessa abordagem, cada domínio — seja pessoal, corporativo ou governamental — pode contribuir com recursos computacionais ou de rede utilizados no processo de detecção e mitigação de ataques DDoS. Apesar da evidente vantagem de um enfoque colaborativo, essa abordagem levanta preocupações significativas em relação à privacidade, pois o tráfego pode ser analisado por domínios externos, expondo potencialmente dados confidenciais durante o processo de detecção e mitigação cooperado.

O VGuard [6] é uma solução para mitigação de ataques DDoS que utiliza tecnologia de virtualização de funções de rede e se baseia na identificação do IP de origem real. Sua arquitetura é composta por um módulo de classificação responsável por atribuir prioridades aos fluxos de entrada e marcar seus pacotes; um firewall que bloqueia o tráfego identificado como malicioso; e um módulo de alocação que direciona os fluxos para túneis com diferentes níveis de prioridade. O túnel de alta prioridade é configurado para suportar todos os fluxos atribuídos a ele, enquanto o túnel de baixa prioridade opera para oferecer o atendimento sob melhor esforço. O VGuard foi a solução inspiradora do DeMONS original, alterou seus modelos operacionais e adicionou novos módulos em sua topologia de serviço.

O trabalho de [18, 19] apresenta o NFV-TE, um *framework* desenvolvido para lidar com desafios de engenharia de tráfego de rede utilizando o paradigma NFV. O NFV-TE permite que operadores configurem funções de rede virtualizadas focadas em mecanismos de *policing* e *shaping* de tráfego, oferecendo uma solução extensível e parametrizável para controle de pacotes na rede. No contexto de engenharia de tráfego, o *framework* destaca-se por sua capacidade de adaptação a diferentes perfis de tráfego, possibilitando a manutenção de níveis satisfatórios de Qualidade de Serviço (*Quality of Service* - QoS), mesmo sob alta demanda. Isso é particularmente útil para redes que enfrentam variações significativas de tráfego ou situações de congestionamento. Assim, a abordagem baseada em NFV para engenharia de tráfego contribui para uma gestão mais eficiente dos recursos de rede, viabilizando o ajuste dinâmico da distribuição de tráfego conforme as condições e as políticas aplicáveis. Estratégias como as implementadas no NFV-TE podem ser integradas a serviços virtualizados de segurança, como o DeMONS, descrito na Seção 2.3.

Apesar das diversas soluções disponíveis para a mitigação de ataques DDoS, nenhuma incorpora o uso de estratégias e funções de *traffic shaping* em seus serviços. Essas estratégias têm o potencial de otimizar a alocação de recursos de rede durante ataques, melhorando a capacidade de resposta às ameaças e reduzindo a sobrecarga nos sistemas de defesa, tornando-os mais eficientes e resilientes.

4 DeMONS++: Uma Abordagem Repaginada para a Mitigação de DDoS

O DeMONS++ é uma solução inspirada no DeMONS, apresentado na Seção 2.3, que utiliza o paradigma NFV para detectar e mitigar ataques DDoS. Assim como o DeMONS em sua versão original, o

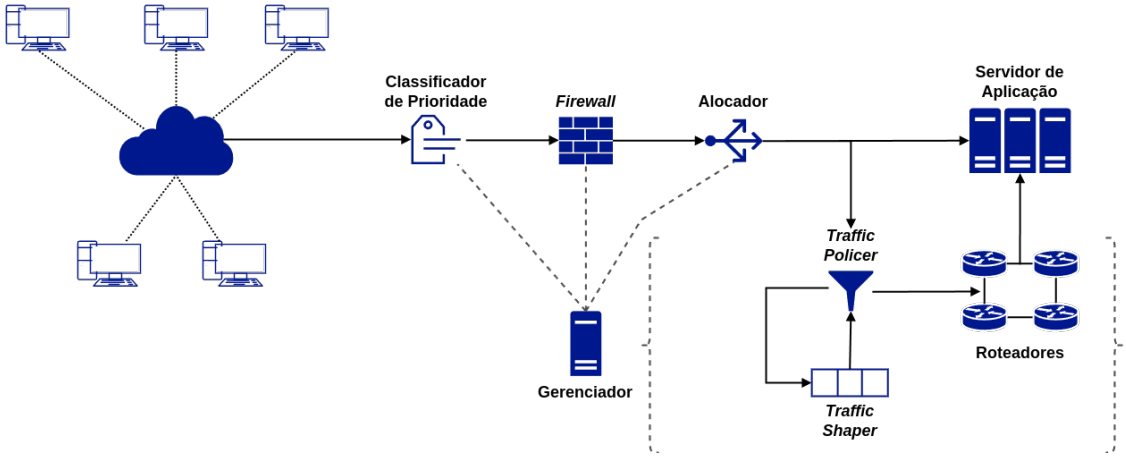

Figura 1: Topologia de Serviço do DeMONS++

DeMONS++ combina abordagens baseadas em capacidade e filtragem; no entanto, distingue-se por introduzir um novo módulo de engenharia de tráfego: o *Traffic Shaper*.

O *Traffic Shaping* consiste em mecanismos projetados para suavizar perfis heterogêneos de tráfego, regulando o fluxo de pacotes para otimizar o uso da largura de banda disponível. Diferentemente dos *Traffic Policers*, que frequentemente descartam pacotes quando as políticas de uso de banda são excedidas, o *Traffic Shaper* prioriza o armazenamento temporário dos pacotes excedentes em filas, minimizando perdas e proporcionando maior estabilidade no tráfego da rede [19].

No contexto do DeMONS++, o *Traffic Shaper* opera no túnel de baixa prioridade da topologia de serviço, posicionado antes do módulo de *Traffic Policer*. Assim, o serviço é composto por seis módulos principais: Classificador de Prioridade, *Firewall*, Alocador de Fluxos, *Traffic Shaper*, *Traffic Policer* e Gerenciador. A topologia de serviço do DeMONS++ é ilustrada na Figura 1.

O fluxo de execução do DeMONS++ inicia com o recebimento do tráfego pelo Classificador de Prioridade, que atribui um valor entre 0 e 1 para os pacotes de cada fluxo. Esse valor representa a prioridade dos pacotes, utilizando um mecanismo de *in-band control* para sinalizar essa classificação. Fluxos identificados como maliciosos, com prioridade igual a 0, são descartados pelo módulo seguinte, o *Firewall*. Pacotes cujos fluxos apresentam valores de prioridade superiores a 0 são encaminhados ao Alocador de Fluxos.

O Alocador de Fluxos decide o túnel de encaminhamento dos pacotes, podendo ser de alta ou baixa prioridade. Fluxos com maior prioridade possuem maior reputação e, consequentemente, maior probabilidade de serem alocados no túnel de alta prioridade. Entretanto, impedindo sobrecarga, o tráfego que excede a capacidade do túnel de alta prioridade – a indicação de entrada em modo seletivo é dada por uma porcentagem de utilização deste túnel – é redirecionado para o túnel de baixa prioridade. O modelo de operação geral dos módulos descritos neste parágrafo, assim como as atribuições do Gerenciador (de manter o desempenho do serviço via migrações e escala dos elementos virtualizados), é idêntico ao do serviço original DeMONS.

Por outro lado, a inserção do módulo de *Traffic Shaping* no DeMONS++ altera as características operacionais do túnel de baixa prioridade, introduzindo um mecanismo adicional de gerenciamento de tráfego. Especificamente, esse módulo implementa um *buffer* que armazena, total ou parcialmente, o tráfego que seria descartado pelo módulo de *Traffic Policer*. Em vez de descartar pacotes imediatamente, o *Traffic Policer* redireciona o tráfego excedente do túnel de baixa prioridade para o *Traffic Shaper*. Este, por sua vez, utiliza a mesma política de controle do *Traffic Policer*, além do tamanho do seu *buffer* e da expectativa de recebimento de tráfego, para determinar se um pacote será armazenado no final do *buffer* ou descartado – como exibido no Algoritmo 1.

O *Traffic Shaper*, por sua vez, reenvia os pacotes armazenados ao *Traffic Policer* em um modelo FIFO, com base em uma taxa de transmissão predefinida. Para evitar a retenção prolongada de pacotes no *buffer*, um limite máximo de reinserções é estipulado. Com isso, caso um pacote não seja transmitido ou descartado após um determinado número de tentativas, ele é removido do *buffer*, pois é provável que já tenha sido retransmitido pela origem ou que o remetente tenha desistido de sua resposta devido a *timeout*. As dinâmicas para encaminhamento são demonstradas no Algoritmo 2. Cabe ressaltar que as características operacionais descritas tornam o *Traffic Shaper* do DeMONS++ uma versão adaptada de um *Leaky Bucket Shaper* [19].

A inclusão do módulo de *Traffic Shaping* no DeMONS++ é motivada principalmente pela necessidade de gerenciar de forma mais refinada o tráfego excedente no túnel de baixa prioridade. Essa abordagem busca minimizar descartes desnecessários, especialmente de pacotes oriundos de usuários potencialmente benignos, promovendo uma utilização mais eficiente dos recursos da rede e contribuindo para uma maior qualidade do serviço para esses usuários.

5 Avaliação e Resultados

Um conjunto de casos de teste foi construído e realizado com o objetivo de medir o impacto do módulo de *traffic shaping* incluído no DeMONS++. Em particular, as análises focaram em determinar como a introdução desse novo componente influencia métricas fundamentais, como a Taxa de Aceitação de Tráfego Benigno e a

Algoritmo 1: Pseudocódigo da Inserção no *Traffic Shaper*

Input:
– `P`: pacote;
– `F`: fila de pacotes;
– `FM`: tamanho máximo da fila de pacotes;
– `EM`: envelhecimento máximo de um pacote na fila;
Data:
– `tamanho(e)`: função que retorna a quantidade de Bytes ocupados por/em um elemento e;
– `evelhecimento(p)`: função que retorna a quantidade de vezes em que o pacote p foi inserido na fila;
– `analisa(p)`: função que retorna VERDADEIRO se, baseado na prioridade do pacote p e na taxa esperada de tráfego, p deve ser enfileirado – caso contrério, retorna FALSO;
– `insere(p, F)`: função que insere o pacote p no final da fila F, além de o envelhece;.
– `descarta(p)`: função que descarta o pacote p.

```
1  begin
2      if tamanho(F) + tamanho(p) ≤ FM then
3          if envelhecimento(p) ≤ EM then
4              if analisa(p) then
5                  insere(p, F);
6                  return VERDADEIRO;
7      descarta(p);
8      return FALSO;
9  end
```

Algoritmo 2: Pseudocódigo do Encaminhamento do *Traffic Shaper*

Input:
- `TA`: taxa atual de transmissão de pacotes;
- `TM`: taxa máxima de transmissão de pacotes;
- `F`: fila de pacotes.

Data:
- `próximo(F)`: função bloqueante que retorna o primeiro pacote em F;
- `atualiza(p)`: função que atualiza e retorna a taxa atual de transmissão de pacotes do *Traffic Shaper*;
- `encaminha(p)`: função que encaminha o pacote p para o *Traffic Policer*;
- `reinsere(p, F)`: função que reinsere o pacote p no início da fila F.

```
1  begin
2      p ← próximo(F);
3      TA ← atualiza(NULL);
4      if TA < TM then
5          encaminha(p);
6          TA ← atualiza(p);
7      else
8          reinsere(p, F);
9      end
10 end
```

Satisfação do Usuário, em cenários sob ataque DDoS. Assim, esta seção apresenta os detalhes técnicos dos testes realizados, descrevendo as configurações do ambiente experimental, além de discutir os resultados obtidos.

Todos os experimentos foram conduzidos em um ambiente simulado, utilizando uma máquina equipada com 8 GB de memória RAM DDR4, processador Intel I7 G11, e executando o sistema operacional PopOS. Para a simulação, foi utilizado um simulador desenvolvido em Python 3, projetado para gerar diferentes perfis de tráfego de rede e executar o serviço DeMONS++ com base nesses perfis (o simulador está disponível em https://github.com/ViniGarcia/DeMONS-PoC). Esse simulador foi integrado ao da solução original DeMONS, estendendo suas funcionalidades.

O tráfego benigno foi modelado como fluxos com distribuição normal, variando entre 10 e 100 Kbps por fluxo, enquanto os ataques DDoS foram representados por fluxos com distribuição constante de 100 Kbps por fluxo. O volume total de tráfego benigno foi configurado para variar, seguindo uma distribuição normal, entre 0 e 1,5 Gbps durante os testes. Por outro lado, o ataque DDoS foi fixado em um volume de tráfego constante de 5 Gbps ao longo de seu ciclo de vida.

Os testes foram configurados com duração de 30 segundos para o tráfego benigno, enquanto o ataque DDoS, com duração de 10 segundos, foi programado para iniciar no décimo segundo e encerrar no vigésimo segundo do tráfego benigno. Fluxos benignos tem prioridades entre 0,4 e 1,0, enquanto fluxos maliciosos, mas ainda suspeitos, apresentam prioridade entre 0,1 e 0,4, sendo todos os intervalos fechados. A Figura 2 ilustra os perfis de tráfego e a configuração utilizada nos experimentos. Após a geração dos perfis, o simulador submeteu esses dados a diferentes configurações do serviço DeMONS++, comparando os melhores resultados com aqueles obtidos a partir das simulações do serviço DeMONS original.

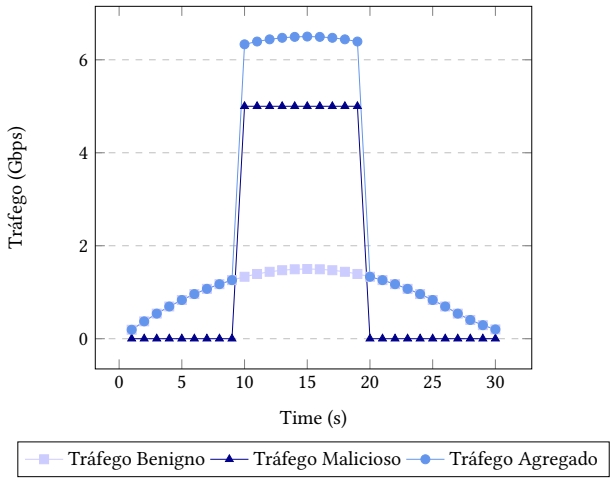

Figura 2: Perfis de Tráfego e Cenário de Testes

No simulador, os túneis de alta e baixa prioridade foram configurados para suportar 500 Mbps cada. O modo seletivo, tanto para o DeMONS++ quanto para o DeMONS original, foi ajustado para operar até 97% da capacidade do túnel de alta prioridade. Adicionalmente, no DeMONS++, o fator de envelhecimento dos pacotes

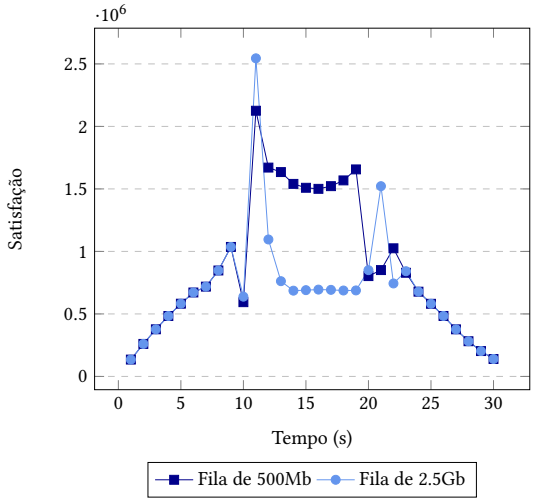

Figura 3: Satisfação do DeMONS++ (Política Restritiva)

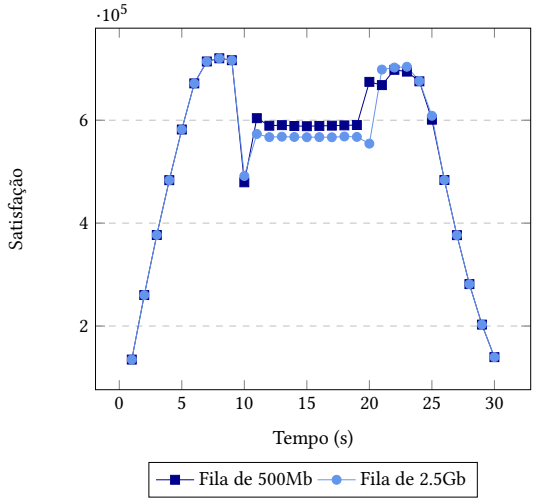

Figura 4: Satisfação do DeMONS++ (Política Intermediária)

na fila foi configurado como 1, ou seja, pacotes reincidentes não são permitidos na fila. A taxa de transmissão do *Traffic Shaper* foi configurada igual ao seu tamanho – ou seja, a fila pode ser, se nenhum pacote for inserido na mesma, esvaziada a cada segundo. As demais configurações foram adaptadas conforme as especificações do caso de teste em execução.

O primeiro caso de teste avalia o impacto do módulo de *Traffic Shaping* do DeMONS++ considerando diferentes combinações de políticas aplicadas (no *Traffic Policer* e no *Traffic Shaper*) e tamanhos de fila. As políticas restritiva e intermediária, previamente definidas para o DeMONS original [8], foram utilizadas. Além disso, foram avaliados dois tamanhos de fila: 500 Mb e 2,5 Gb. Os resultados de satisfação dos usuários (número adimensional calculado a partir da quantidade e prioridade dos fluxos atendidos, busca-se sua maximização [6]) obtidos para as políticas restritiva e intermediária de aceitação de tráfego são apresentados, respectivamente, nas Figuras 3 e 4.

Em relação aos resultados obtidos com a variação de políticas e tamanhos de filas no DeMONS++, dois pontos principais devem ser destacados:

- Considerando que a capacidade do túnel de baixa prioridade é fixa e que existe tráfego benigno (de alta prioridade) que excede essa capacidade, juntamente com tráfego atacante de baixa prioridade, é esperado que políticas mais restritivas de encaminhamento resultem em maior satisfação do sistema. Esse comportamento pode ser observado nos testes apresentados nas Figuras 3 e 4;
- Quando a taxa de transmissão é equivalente ao tamanho do *buffer* do *traffic shaper* por segundo, um *buffer* maior aumenta a sobrecarga do sistema e, consequentemente, a quantidade de tráfego descartado por segundo. Isso leva a uma redução na satisfação, especialmente evidente na comparação das curvas sob políticas mais restritivas (Figura 3). Nesses casos, além da maior taxa de descarte geral, existe uma maior quantidade de pacotes descartados que pertencem a fluxos de maior prioridade; mesmo que, proporcionalmente, não exista diferença.

Utilizando políticas restritivas nos módulos de *Traffic Policing* e *Traffic Shaping* (quando pertinente), as soluções DeMONS e DeMONS++ foram comparadas no mesmo cenário de testes de DDoS. Inicialmente, a Figura 5 apresenta os resultados observados para a satisfação no túnel de alta prioridade. Com a mesma política e o mesmo modo operacional aplicado fora do túnel de baixa prioridade, há pouca variação nos resultados nesse caso, sendo as flutuações atribuídas a diferenças no balanceamento dos fluxos em situações de sobrecarga. Todos os fluxos alocados no túnel de alta prioridade são benignos, o que garante uma Taxa de Aceitação de Tráfego Benigno de 100%.

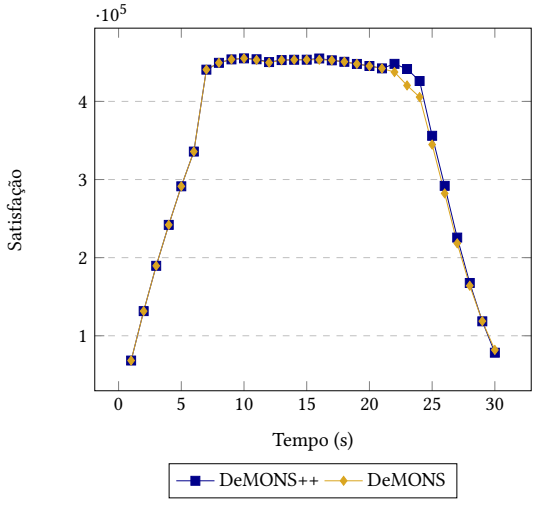

Figura 5: Túnel de Alta Prioridade - DeMONS++ e DeMONS

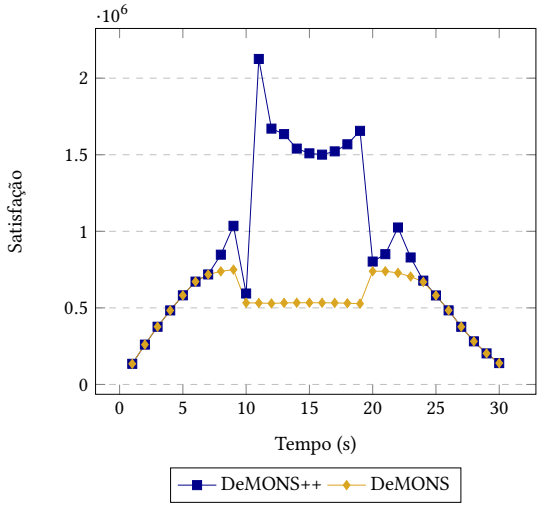

Figura 6: Satisfação: DeMONS++ e DeMONS (Fila de 500Mb e Política Restritiva)

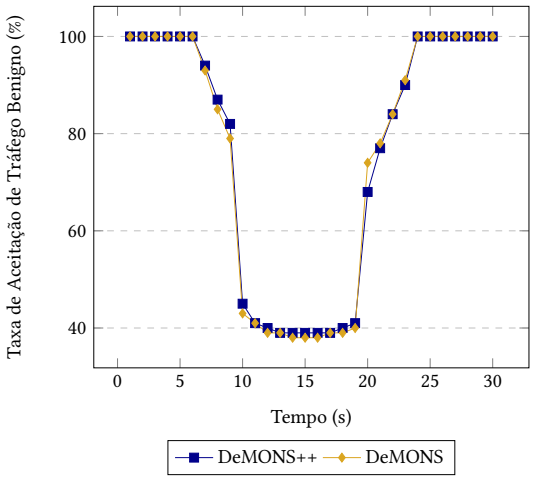

Figura 7: Aceitação de Tráfego Benigno: DeMONS++ e DeMONS (Fila de 500Mb e Política Restritiva)

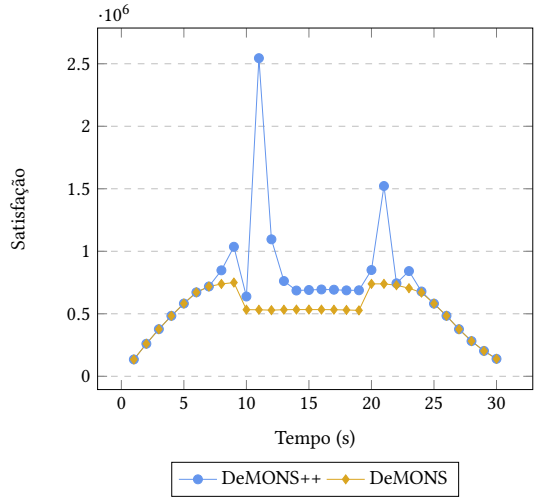

Figura 8: Satisfação: DeMONS++ e DeMONS (Fila de 2.5Gb e Política Restritiva)

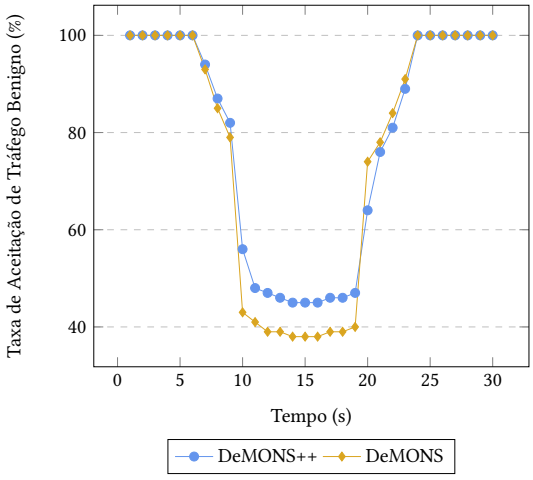

Figura 9: Aceitação de Tráfego Benigno: DeMONS++ e DeMONS (Fila de 2.5Gb e Política Restritiva)

Como a operação do módulo de *Traffic Shaper* introduz diferenças significativas nos resultados do túnel de baixa prioridade dependendo da política de aceitação de tráfego e do tamanho do *buffer* utilizado, a comparação com a solução original DeMONS foi conduzida utilizando as configurações que apresentaram os melhores resultados para o DeMONS++ em termos de métricas de satisfação e taxa de aceitação de tráfego benigno nos testes anteriores — Política Restritiva com Fila de 500 Mb e Política Restritiva com Fila de 2,5 Gb, respectivamente.

Nesse contexto, a Figura 6 apresenta a satisfação obtida no túnel de baixa prioridade pelas soluções DeMONS original e DeMONS++ com política restritiva e fila de 500 Mb. Observa-se que a manutenção parcial do tráfego benigno na fila e com um leve aumento na quantidade de tráfego descartado por segundo, devido ao esvaziamento da fila, geram um impacto positivo na satisfação. Isso permite que usuários com alta reputação, que já eram parcialmente atendidos, passem a ter mais requisições respondidas. No entanto, o aumento agregado da Taxa de Aceitação de Tráfego Benigno, ilustrado na Figura 7, é marginal em comparação ao DeMONS original. Isso ocorre porque, embora individualmente os pacotes de fluxos benignos acessem a fila em boa proporção, esta também é utilizada para armazenar parte do tráfego de baixa reputação e malicioso que ainda consegue ingressar no sistema (ainda não foi reconhecido como de prioridade 0), que é significativamente mais volumoso. Assim, o tamanho da fila não é suficiente para armazenar uma quantidade expressiva de tráfego benigno que resulte em um grande aumento na sua taxa de aceitação geral durante a ocorrência

de um ataque: em média, 39,4% para o DeMONS original e 40,2% para o DeMONS++.

Por outro lado, a comparação entre a solução DeMONS++ com o *traffic shaper* configurado com uma fila de 2,5 Gb e a solução DeMONS original, ambas adotando a política restritiva para aceitação de pacotes, apresenta resultados distintos. Nesse cenário, embora haja um ganho na métrica de satisfação para o DeMONS++ em relação ao DeMONS, como ilustrado na Figura 8, esse ganho não é disruptivo. Ele ocorre pelos mesmos motivos apresentados anteriormente no caso de teste da Figura 6; contudo, é reduzido devido a uma quantidade substancialmente maior de descarte extra de pacotes por segundo, na ordem de cinco vezes. Isso se deve ao fato de que a capacidade dos túneis é estática, e a vazão do *traffic shaper* é suficiente para esvaziar a fila a cada segundo (*i.e.*, até 2,5 Gbps).

Por outro lado, mantendo o tráfego de teste, o aumento do tamanho da fila implementada permite armazenar um maior volume de tráfego benigno e de alta reputação. Com a adoção dos filtros restritivos, isso aumenta significativamente as chances de os pacotes de interesse serem entregues ao destino. Como resultado, durante a execução do ataque, as taxas de aceitação de tráfego benigno no DeMONS++ variaram entre 45% e 48%, enquanto no DeMONS original oscilaram entre 38% e 43%.

De maneira geral, os testes realizados demonstraram que o DeMONS++ apresenta comportamentos desejáveis e resultados promissores na mitigação de ataques de negação de serviço, com efeitos benéficos e superiores em todas as métricas analisadas em relação à solução DeMONS original. Todos os arquivos de teste, assim como o simulador do VGuard, DeMONS e DeMONS++, estão disponíveis em https://github.com/ViniGarcia/DeMONS-PoC/tree/master/Paper/COTB.

6 Conclusão

Diante do crescente desafio imposto pelos ataques distribuídos de negação de serviço, soluções baseadas em virtualização, como o DeMONS, têm se mostrado efetivas no combate a essas ameaças, ao oferecer uma abordagem híbrida de filtragem e priorização de tráfego. No entanto, as soluções disponíveis nesse contexto não empregam esforços relacionados à manutenção de tráfego potencialmente benigno através do tempo, abrindo novas janelas de oportunidade para a sua transmissão.

Sendo assim, este trabalho apresentou o DeMONS++, uma solução baseada no serviço DeMONS que incorpora novas mecânicas de gerenciamento de tráfego, além de um módulo de *traffic shaping*, permitindo que técnicas de engenharia de tráfego de rede sejam exploradas no contexto de segurança. Para avaliar a solução proposta, utilizou-se um ambiente simulado em diferentes casos de teste, que demonstraram a superioridade do DeMONS++ em relação ao DeMONS original nas métricas de satisfação e taxa de aceitação de tráfego benigno em todos os cenários analisados.

Como trabalhos futuros, planeja-se explorar a implementação de diversas técnicas de *traffic shaping* investigar o impacto de diferentes políticas de priorização de tráfego em ambientes de rede dinâmicos e submeter as soluções a testes variados, considerando tipos e perfis distintos de ataques de negação de serviço. Além disso, o objetivo é implementar as funções de rede em cenários reais, eliminando as rotinas de simulação de tráfego de rede para experimentação e explorando algoritmos inteligentes para a atribuição de reputação aos fluxos.

Agradecimentos

Este projeto foi parcialmente financiado pelo Ministério da Saúde através de uma TED para PD&I entre SAPS/MS e C3SL/UFPR.

Referências

[1] Ajeet Kumar Sharma and Rakesh Kumar. A comprehensive survey of ddos attacks: Evolution, mitigation and emerging trend. In *International conference on Power Electronics and IoT Applications in Renewable Energy and its Control*, pages 185–188, Vrindavan, India, 2024. IEEE.

[2] The Hacker News. Cloudflare thwarts largest-ever 3.8 tbps ddos attack targeting global sectors. https://thehackernews.com/2024/10/cloudflare-thwarts-largest-ever-38-tbps.html, 2024. Acessado em 21/10/2024.

[3] Marty Roesch, Amy Henderson, et al. Snort - open source intrusion prevention system. https://www.snort.org, 2024. Acessado em 21/10/2024.

[4] Open Information Security Foundation. Suricata - observe. protect. adapt. https://suricata.io/, 2024. Acessado em 21/10/2024.

[5] Xuhui Cai, Hui Deng, AE Lingli Deng, S Gao, AMD Nicolas, Y Nakajima, J Pieczerak, J Triay, X Wang, B Xie, et al. Evolving nfv towards the next decade. *ETSI White Paper*, 1(54):1–22, 2023.

[6] Carol J Fung and Bill McCormick. Vguard: A distributed denial of service attack mitigation method using network function virtualization. In *International Conference on Network and Service Management*, pages 64–70, Barcelona, Spain, 2015. IEEE.

[7] AHM Jakaria, Wei Yang, Bahman Rashidi, Carol Fung, and M Ashiqur Rahman. Vfence: A defense against distributed denial of service attacks using network function virtualization. In *Annual Computer Software and Applications Conference*, volume 2, pages 431–436, Atlanta, USA, 2016. IEEE.

[8] Vinícius Fülber Garcia, Guilherme de Freitas Gaiardo, Leonardo da Cruz Marcuzzo, Raul Ceretta Nunes, and Carlos Raniery Paula dos Santos. Demons: A ddos mitigation nfv solution. In *International Conference on Advanced Information Networking and Applications*, pages 769–776, Kraków, Poland, 2018. IEEE.

[9] Christos Douligeris et al. Ddos attacks and defense mechanisms: classification and state-of-the-art. *Computer networks*, 44(5):643–666, 2004.

[10] Monowar H Bhuyan, Abhishek Kalwar, A Goswami, DK Bhattacharyya, and JK Kalita. Low-rate and high-rate distributed dos attack detection using partial rank correlation. In *International Conference on Communication Systems and Network Technologies*, pages 706–710, Gwalior, India, 2015. IEEE.

[11] Lee Garber. Denial-of-service attacks rip the internet. *Computer*, 33(04):12–17, 2000.

[12] Vinicius Fulber-Garcia, Giovanni Venâncio De Souza, Elias Procopio Duarte Jr, Thales Nicolai Tavares, Leonardo Da Cruz Marcuzzo, Carlos RP Dos Santos, Muriel Figueredo Franco, Lucas Bondan, Lisandro Zambenedetti Granville, Alberto Egon Schaeffer-Filho, et al. On the design and development of emulation platforms for nfv-based infrastructures. *International Journal of Grid and Utility Computing*, 11(2):230–242, 2020.

[13] Talal Alharbi, Ahamed Aljuhani, and Hang Liu. Holistic ddos mitigation using nfv. In *Annual Computing and Communication Workshop and Conference*, pages 1–4, Las Vegas, USA, 2017. IEEE.

[14] Vinicius Fulber-Garcia, Alexandre Huff, Carlos R P dos Santos, and Elias P Duarte Jr. Network service topology: Formalization, taxonomy and the custom specification model. *Computer Networks*, 178:107337, 2020.

[15] Woosik Lee, Yoon-Ho Choi, and Namgi Kim. Study on virtual service chain for secure software-defined networking. *Advanced Science and Technology Letters*, 29(13):177–180, 2013.

[16] Leonardo da Cruz Marcuzzo, Vinicius Fulber-Garcia, Vitor Cunha, Daniel Corujo, Joao P Barraca, Rui L Aguiar, Alberto E Schaeffer-Filho, Lisandro Z Granville, and Carlos RP dos Santos. Click-on-osv: A platform for running click-based middleboxes. In *Symposium on Integrated Network and Service Management*, pages 885–886, Lisbon, Portugal, 2017. IEEE.

[17] Bahman Rashidi and Carol Fung. Cofence: A collaborative ddos defence using network function virtualization. In *International Conference on Network and Service Management*, pages 160–166, Montreal, Canada, 2016. IEEE.

[18] Felipe Quiles, Joaão Moreira, Vinicius Fulber-Garcia, and Elias Duarte Jr. Nfv-te: Uma ferramenta para a geração automática de funções virtuais rede para engenharia de tráfego. In *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 1–8, Fortaleza, Brazil, 2022. SBC.

[19] Felipe Ribeiro Quiles, João Vitor Moreira, Vinicius Fulber-Garcia, and Elias Procópio Duarte Jr. Nfv-te: Geração automática de funções virtualizadas para engenharia de tráfego de rede. *Revista Eletrônica de Iniciação Científica*, 21(1), 2023.