Uma Solução para Mitigação de Ataques DDoS Através de Tecnologia NFV

Vinícius F. Garcia[1], Guilherme de F. Gaiardo[1], Leonardo da C. Marcuzzo[1], Thales N. Tavares[1], Nilton C. B. da Silva[1], Anderson Monteiro[1,2], Raul C. Nunes[1], Carlos Raniery P. dos Santos[1]

[1]Universidade Federal de Santa Maria (UFSM)
Pós-graduação em Ciência da Computação (PGCC)

[2]Instituto Federal Farroupilha (IFFar)
Campus São Vicente do Sul

{vfulber, ggaiardo, lmarcuzzo, tntavares}@inf.ufsm.br
{amonteiro, nbatista, ceretta, csantos}@inf.ufsm.br

Abstract. *Distributed Denial of Service (DDoS) attacks are constantly evolving, new and sophisticated infection methods are always being employed by attackers. To deal with such constant change, the research community is always searching for advanced approaches to mitigate, or even eliminate, those threats. One of these new approaches are the use of Network Function Virtualization (NFV). This new paradigm supports the creation of more scalable and flexible, thus resilient, network infrastructures. We, therefore, propose a DDoS mitigation system – called DeMONS – that uses NFV concept together both a dynamic allocation and reputation mechanisms. The results demonstrate that the employed techniques are a feasible solution to reach a better performance related to the system utilization.*

Resumo. *Ataques de Negação de Serviço Distribuído (DDoS) evoluem constantemente, novos e sofisticados métodos de infecção são empregados pelos atacantes. Para lidar com essas mudanças constantes, a comunidade científica busca por abordagens avançadas para mitigar, ou eliminar, tais ameaças. Uma dessas novas abordagens é o uso de Funções Virtualizadas de Rede (Network Functions Virtualization - NFV). Este novo paradigma suporta a criação de infraestruturas de rede escaláveis, flexíveis e resilientes. Neste artigo é proposto um sistema de mitigação de ataques DDoS – chamado DeMONS – que usa os conceitos NFV junto a mecanismos de alocação dinâmica e de reputação. Os resultados demonstraram que as técnicas empregadas são uma solução viável para atingir um desempenho superior em relação a utilização do sistema.*

1. Introdução

Ataques de Negação de Serviço Distribuída (DDoS) são grandes e sofisticadas ameaças que vem se tornando cada vez mais frequentes, podendo causar significativas perdas de dados [Garber 2000]. Suas ocorrências estão entre os maiores fatores de preocupação para empresas de tecnologia da informação que dependem da Internet. Para essas empresas, a indisponibilidade de serviços devido a um ataque DDoS representa severas perdas financeiras (devido, por exemplo, a quebra de políticas ou queda de reputação). Como

um exemplo de quão abrangente um DDoS pode ser, um ataque realizado em 2016 utilizou mais de 100.000 máquinas infectadas para sobrecarregar o ISP *Singapore Star Hub* [StarHub 2016].

Negação de Serviço Distribuída define um ataque que objetiva tornar um serviço indisponível através do emprego de um número massivo de solicitações, tipicamente falsas, advindas de diversas fontes (*i.e.*, ataque distribuído) [Douligeris and Mitrokotsa 2004]. Ataques DDoS podem ser classificados em duas principais categorias [Fung and McCormick 2015]: Falsificação de IP e IP de Origem Real. No primeiro, o endereço IP de origem não representa máquinas reais, mas, em vez disso, são artificialmente gerados. Esse tipo de ataque é a forma mais comum de DDoS e tipicamente são mais facilmente detectados por Sistemas de Detecção de Intrusão (IDS). O segundo, por outro lado, utiliza máquinas reais infectadas (*i.e., botnet*) controladas por um elemento principal (*i.e., botmaster*). Diferentes técnicas são comumente empregadas pelo *botmaster* para fazer deste tipo de ataque ainda mais complexo para ser detectado.

Devido aos problemas em potencial causado pelos ataques DDoS, a comunidade acadêmica está constantemente investigando e propondo novas técnicas de mitigação. Essas técnicas podem ser classificadas em dois principais grupos: capacidade e filtro [Fung and McCormick 2015]. Soluções de capacidades limitam o tráfego de rede usando uma abordagem baseada em prioridade, enquanto métodos baseados em filtros visam bloquear o tráfego malicioso. Na ocorrência de grandes ataques DDoS, sistemas baseados em capacidade usualmente tornam-se sobrecarregados, tornando-se inefetivos. Neste caso, entretanto, devido às características desse tipo de sistema, o tráfego considerado benigno não é totalmente bloqueado durante um ataque mesmo se o sistema de detecção falhar. Por outro lado, soluções baseadas em filtros dependem de um processo de detecção de anomalias muito acurado e preciso, se o sistema falhar, fluxos falsos-positivos são simplesmente bloqueados.

Métodos de mitigação de DDoS baseados em capacidade são apropriados quando a acurácia dos IDSs não foi avaliada ou ataques complexos – como os de IP de Origem Real – são prevalentes. Esses métodos alocam recursos computacionais (*e.g.*, processamento, memória e rede) apenas para fluxos confiáveis. Para fluxos desconhecidos ou suspeitos, o sistema trabalha em modo *best effort*, portanto o sistema pode ficar sobrecarregado e eventualmente não atender algumas requisições. É importante notar que métodos baseados em capacidade requisitam uma quantidade significante de recursos para realizarem a mitigação de ataques DDoS. Nesse contexto, tecnologias que suportem provisionamento elástico de recursos, como virtualização, são especialmente adequadas.

Virtualização de Funções de Rede (NFV) é um paradigma emergente que usa tecnologias de virtualização existentes para desacoplar a função de rede de seu hardware associado (*i.e., middleboxes*) [ETSI 2012]. O paradigma NFV prevê um ambiente flexível e escalável encorajando, portanto, a inovação, redução de *time to market* e de custos de capital e operacionais (CAPEX e OPEX), além de possibilitar o uso de hardware genérico para a execução das funções de rede [John et al. 2013].

Devido as suas vantagens, vários esforços disponíveis na literatura [Alharbi et al. 2017] [Jakaria et al. 2016] [Rashidi et al. 2017] [Fung and McCormick 2015] utilizam NFV como abordagem para mitigar ataques

DDoS. Em especial, [Fung and McCormick 2015] propõem uma solução – chamada VGuard – para mitigar ataques DDoS com IPs de Origem Real através de um mecanismo baseado em capacidade. O VGuard emprega túneis com diferentes prioridades (alta e baixa) para separar o tráfego de acordo com suas reputações. O túnel de alta prioridade nunca opera sobrecarregado, garantindo a entrega dos pacotes que o cruzam ao seu destinatário. O segundo túnel, por outro lado, recebe fluxos com baixa reputação e opera no modelo *best effort* (*i.e.,* quando sobrecarregado pacotes são descartados).

Apesar do VGuard ser uma solução viável para mitigar ataques DDoS e garantir certa Qualidade de Serviço (QoS), ele não é totalmente eficiente no descarte de fluxos maliciosos e trata fluxos previamente alocados de forma estática. Por exemplo, uma vez que um fluxo específico é alocado em um certo túnel, o sistema não reavalia sua decisão. Essa abordagem não é adequada para todos os cenários, como quando alguns fluxos maliciosos acessam o túnel de alta prioridade ou fluxos de alta prioridade são alocados no túnel de baixa prioridade.

Nesse contexto, este trabalho apresenta o DeMONS, uma nova solução híbrida (baseada nas abordagens de capacidade e de filtro) para mitigar ataques DDoS através da utilização do paradigma NFV. A arquitetura proposta foi inspirada na arquitetura VGuard, com um IDS provendo as prioridades dos fluxos, um *firewall* bloqueando o tráfego reconhecidamente malicioso e um sistema de alocação atribuindo fluxos a túneis com diferentes prioridades. O DeMONS, entretanto, introduz um novo módulo – o classificador – que constantemente analisa os fluxos da rede para decidir a qual túnel eles devem ser alocados. Além disso, o túnel de baixa prioridade emprega um mecanismo de reputação (implementado como uma Função Virtualizada de Rede (VNF)) responsável por limitar o tráfego de rede de acordo com as prioridades definidas pelo IDS. Com essa abordagem, pacotes com alta prioridade apresentam menores taxas de descarte qual o túnel de baixa prioridade está sobrecarregado.

Este artigo está estruturado como segue: Seção 2 apresenta os principais conceitos relacionados a solução proposta. A Seção 3 examina e conceitua a arquitetura VGuard. Em seguida, a solução DeMONS é introduzida na Seção 4, com uma comparação entre DeMONS e VGuard discutida na Seção 5. Finalmente, a Seção 6 concluí o artigo.

2. Fundamentação e Trabalhos Relacionados

A infraestrutura de rede tradicional baseia-se em equipamentos dedicados e especializados – chamados *middleboxes* – para a execução de funções de rede. Apesar de eficientes, esses equipamentos resultam em altos custos de capital e operacionais (CAPES e OPEX), falta de flexibilidade e alta complexidade de gerenciamento e escala da infraestrutura [Sherry et al. 2012].

Para superar esses problemas, um novo paradigma chamado de Funções Virtualizadas de Rede [ETSI 2012] foi introduzido com o objetivo de desacoplar funções de rede de seu hardware associado através de tecnologias de virtualização. A introdução de uma camada de virtualização implementando funções de rede leva a uma degradação de desempenho (*e.g.,* maior atraso, menor *throughput*). Porém, esforços recentes utilizam aceleradores de pacotes objetivando minimizar esses efeitos colaterais (*e.g.*, OpenNetVM [Zhang et al. 2016] e Click-On-OSv [Marcuzzo et al. 2017]).

Nesse contexto, em [Alharbi et al. 2017] os autores demonstram as vantagens e

problemas em usar técnicas tradicionais de mitigação DDoS em ambientes virtualizados. Os autores, por exemplo, ressaltam como aspectos positivos de NFV sua flexibilidade, elasticidade e redução de CAPEX/OPEX. Entretanto, apesar dessas características, o paradigma NFV ainda apresenta vários desafios, como a falta de suporte para instanciação rápida e eficiente de topologias de defesa.

Em [Alharbi et al. 2017] os autores propõem uma arquitetura de mitigação de ataques DDoS com suporte a análise de dados da camada de rede e de aplicação. O componente central da arquitetura – *i.e.*, *traffic screener* – foi projetado para analisar todo o tráfego de dados procurando por potenciais anomalias. Esse componente classifica o tráfego e rotas de acordo com o seu tipo (*i.e.*, ataque ou serviço). O tráfego de serviço é encaminhado para o sistema final, enquanto, enquanto que o tráfego de ataque é encaminhado para ser processado por uma VNF de segurança. Entretanto, uma lacuna da abordagem proposta é como executar de forma eficiente o provisionamento de recursos de acordo com a demanda dos filtros de tráfego.

O sistema VFence [Jakaria et al. 2016] foi desenvolvido para mitigar ataques do tipo *SYN flood*. Esse sistema é baseado em VNFs escaláveis responsáveis pela filtragem do tráfego malicioso: o expedidor e os agentes. O expedidor é responsável por realizar balanceamento de carga do tráfego entre os agentes disponíveis. Os agentes, por sua vez, são responsáveis por verificar se um fluxo é malicioso ou benigno através da coordenação de um processo de *three-way handshake*. Fluxos autenticados são gravados em uma lista branca, enquanto qualquer outro tráfego é bloqueado.

A solução CoFence [Rashidi et al. 2017] foi criada para dar suporte a colaboração entre múltiplos domínios administrativos para mitigar ataques do tipo *SYN flood*. Cada domínio provê recursos (*e.g.*, uma companhia ou uma agência governamental) que podem ser utilizados para o processo de detecção de DDoS. É importante notar que, apesar das vantagens da utilização de uma abordagem colaborativa, esse método leva a problemas de privacidade uma vez que o tráfego pode ser analisado em diferentes domínios.

3. Solução VGuard

VGuard [Fung and McCormick 2015] é uma solução para mitigação de DDoS do tipo IP de Origem Real baseado em tecnologias NFV. A arquitetura consiste de: um módulo de classificação (baseado em um IDS) que atribui prioridades para cada fluxo de entrada e marca seus pacotes; um *firewall* para bloquear o tráfego reconhecidamente malicioso; e um módulo de alocação que atribuí tráfego com diferentes prioridades a túneis distintos.

O módulo de classificação (*i.e.*, o IDS) provê a tabela de prioridades acessada pelos outros módulos da solução proposta. Se um fluxo específico é identificado como benigno, ele contará com alta prioridade para acessar o serviço desejado. Se um fluxo for considerado malicioso (*i.e.*, prioridade zero) um *firewall* simplesmente descarta seus pacotes antes de chegarem ao servidor final. Finalmente, se um fluxo está em um estado incerto, ele receberá uma prioridade entre zero e um.

O cenário avaliado consiste de múltiplos usuários acessando um servidor de aplicação (*e.g.*, ProFTP, Apache Server) enquanto um ataque DDoS é iniciado. Durante o ataque, novos usuários também são incluídos no sistema para simular o comportamento de um cenário de serviço de rede real.

É importante evidenciar que ataques DDoS tem características específicas que os distinguem do tráfego benigno. Por exemplo, a alta quantidade de tráfego malicioso, o início abrupto e a presença de fluxos não usuais ao sistema de segurança [Yang et al. 2008]. Essas características podem ser utilizadas para determinar as prioridades para cada fluxo de rede. Entretanto, uma análise simples dos fluxos pode não ser suficiente para determinar se um ataque está ocorrendo. Nesse caso, sistemas adicionais (*e.g.*, IDSs secundários e *Deep Packet Inspection* (DPI)) podem ser usados para atualizar a informação de prioridade dos fluxos.

3.1. Desempenho

Para analisar o desempenho da solução VGuard inicialmente é necessário determinar a taxa de descarte (d) para cada túnel. Essa taxa está no intervalo [0;1], com 0 representando a ausência de descarte de pacotes e 1 quando todo o tráfego é descartado. A taxa de descarte é calculada utilizando a capacidade do túnel (C) e a quantidade de tráfego atravessando o mesmo (r), como apresentado na Equação 1.

Satisfação de serviço (s) é um método para quantificar e capturar a qualidade do serviço. Considerando que todo o tráfego é devidamente processado pelo servidor, a satisfação calculada é 1, de outra forma a satisfação decrescerá de maneira quadrática de acordo com a taxa de descarte (d), assumindo, assim, a forma de uma parábola $s(x) = x^2$ - onde x representa o estado do sistema relacionado ao descarte em determinado tempo. A satisfação do túnel é apresentada na Equação 2.

$$d = 1 - min(1, C/r) \qquad (1) \qquad\qquad s = (1-d)^2 \qquad (2)$$

A eficácia do VGuard é avaliada pela satisfação agregada (S_V) dos túneis de baixa e alta prioridade (S_L e S_H respectivamente), ponderada pela prioridade dos fluxos (p). A S_V corresponde a um número adimensional que varia de zero até o valor da capacidade do túnel. Tal satisfação pode ser expressada na forma da Equação 3.

$$S_V = S_L + S_H = \sum_{i=0}^{n} r_i p_i s_L + \sum_{j=0}^{m} r_j p_j s_L \qquad (3)$$

3.2. Alocação de Fluxos VGuard

A solução VGuard provê duas formas de alocação de fluxos: estática e dinâmica. A alocação de fluxos dinâmica atinge melhores resultados do que a estática. Esse método de alocação usa a informação de prioridade e o estado dos túneis para decidir onde os fluxos devem ser alocados. A alocação dinâmica, demonstrada no Algoritmo 1, apresenta as seguintes características:

- Novos fluxos são alocados no túnel com menor carga se todos os túneis estão subutilizados;
- O túnel de alta prioridade entra em modo seletivo quando o mesmo está próximo de sua carga máxima de tráfego, um algoritmo passa a decidir onde novos fluxos serão alocados;

- Alocação sumária de fluxos no túnel de baixa prioridade quando qualquer sinal de sobrecarga ou de utilização total é constatada no túnel de alta prioridade.

4. Solução DeMONS

DeMONS é uma solução híbrida (baseada em capacidade e filtros) para mitigar ataques DDoS que utiliza o paradigma NFV para garantir um gerenciamento de recursos elástico. A arquitetura DeMONS, ilustrada na Figura 1, consiste em cinco módulos principais (implementados como VNFs): Classificador de Prioridades, *Firewall*, Alocador de Fluxos, Policiamento de Tráfego e Gerente.

Inicialmente, o tráfego de chegada atravessa o Classificador de Prioridades, que analisa os fluxos e define um valor de prioridade no intervalo [0;1], com 0 representando fluxos reconhecidamente maliciosos, bloqueados pelo próximo módulo (*i.e.*, o *Firewall*). Outros fluxos recebem permissão do *Firewall* para serem processados pelo módulo Alocador de Fluxos. O Classificador de Prioridades pode ser um sistema baseado em detecção de anomalias, porém outras possibilidades de análise podem ser consideradas para a definição de prioridade, como políticas e níveis de privilégio da fonte geradora do tráfego.

O módulo Alocador de Fluxos, por sua vez, encaminha os fluxos de acordo com sua prioridade a túneis distintos (*i.e.*, túnel de alta ou de baixa prioridade). O primeiro é destinado a fluxos de alta prioridade quando o sistema está sobrecarregado. É importante ressaltar que esse túnel é desenvolvido para nunca operar acima de sua capacidade. O segundo recebe fluxos com baixas prioridades e pode operar sobrecarregado. Quando este túnel está sobrecarregado, o módulo de Policiamento de Tráfego aplica políticas de descarte de acordo com a prioridade dos fluxos.

O Gerente é responsável pelo provisionamento do sistema DeMONS. Ele monitora a carga da infraestrutura e decide quando as VNFs devem realizar algum tipo de escala (*i.e.*, *in, out, up* ou *down*). Essas ações criam um ambiente elástico para todos os componentes do sistema (*e.g.*, se o classificador está sobrecarregado, é possível adicionar recursos ou instanciar cópias desse classificador junto a um balanceador de carga). Em cenários de subutilização (i.e., quando o total da carga de tráfego é menor que a capacidade total do túnel de alta prioridade), o gerente é capaz de desativar todo o segmento de baixa prioridade, evitando desperdícios de recursos e energia.

Os métodos do Alocador de Fluxos usado no DeMONS são apresentados no Algoritmo 2. Quando ambos os túneis estão subutilizados, o Alocador de Fluxo balanceia o tráfego entre os mesmos para uma melhor utilização de recursos (linha 10 a 21). Quando o túnel de alta prioridade atinge uma utilização de capacidade pré-definida, este entra em modo seletivo (linha 22) e um método de balanceamento de fluxos é executado para realocar os fluxos de mais alta prioridade para o túnel de alta prioridade. O balanceamento de fluxos consiste na execução de alocações condicionais para cada fluxo que atravessa o túnel de baixa prioridade e que apresente prioridade maior que o fluxo com menor prioridade atravessando o túnel de alta prioridade.

Uma vez que o balanceamento de fluxos garante que os fluxos de mais alta prioridade estão atravessando o túnel de alta prioridade, as próximas decisões de alocação para novos fluxos levam em consideração apenas o fluxo com menor prioridade neste túnel (linha 31). Alocações condicionais (linhas 24 e 36) podem ser realizadas para adicionar

novos fluxos com mais altas prioridades no túnel de alta prioridade através do rebaixamento de fluxos com menores prioridades alocados previamente, o Algoritmo 3 apresenta essa rotina.

```
1  t_H, t_L :
     tuneis de alta e baixa prioridade
2  τ_max :
     utilizacao maxima do tunel de alta prioridade
3  τ_norm :
     marca de entrada em modo seletivo
4  U_H, U_L :
     utilizacao dos tuneis de alta e baixa prioridade
5  begin
6      //Inicializacao
7      t_H = t_L = 0
8      Evento e disparado quando um novo fluxo f chega
9      if U_L < U_H then
10         t_L ← t_L ∪ f
11     end
12     else
13         if U_L < τ_norm then
14             t_H ← t_H ∪ f
15         end
16         else
17             if U_H > τ_max then
18                 t_L ← t_L ∪ f
19             end
20             else
21                 //Modo seletivo trabalhando
22                 if
                     prioridade(f) >
                     prioridadeMedia(t_H)
                    then
23                     t_H ← t_H ∪ f
24                 end
25                 else
26                     t_L ← t_L ∪ f
27                 end
28             end
29         end
30     end
31 end
```

Algorithm 1: ALOCAÇÃO DE FLUXOS DINÂMICA VGUARD

```
1  t_H, t_L :
     tuneis de alta e baixa prioridade
2  τ_Hmax, τ_Lmax :
     utilizacao maxima dos tuneis de alta e baixa
3    prioridade respectivamente
4  τ_Hnorm :
     marca de entrada em modo seletivo
5  U_H, U_L :
     utilizacao dos tuneis de alta e baixa prioridade
6  LU_H :
     ultima utilizacao do tunel de alta prioridade
7  begin
8      //Inicializacao
9      t_H = t_L = 0
10     Evento e disparado quando um novo fluxo f chega
11     if U_L < U_H then
12         t_L ← t_L ∪ f
13     end
14     else
15         if U_L < τ_Hnorm then
16             if U_H +
                 trafego(f) <=
                 τ_Hmax then
17                 t_H ← t_H ∪ f
18             end
19             else
20                 t_L ← t_L ∪ f
21             end
22         end
23         else
24             if U_H > τ_Hmax then
25                 alocacaoCondicional(f)
26             end
27             else
28                 //Modo seletivo trabalhando
29                 if LU_H <
                     τ_Hnorm then
30                     balanceiaFluxos()
31                 end
32                 if
                     prioridade(f) >
                     menorPrioridade(t_H)
                    then
33                     if U_H +
                         trafego(f) <=
                         τ_Hmax
                        then
34                         t_H ←
                             t_H ∪ f
35                     end
36                     else
37                         alocacaoCondicional(f)
38                     end
39                 end
40                 else
41                     t_L ← t_L ∪ f
42                 end
43             end
44         end
45     end
46 end
```

Algorithm 2: ALOCAÇÃO DE FLUXOS DINÂMICA DEMONS

```
1  g_D :
     grupo de fluxos a serem alocados em t_L
2  U_D :
     trafego gerado pelos fluxos de g_D
3  g_D = 0
4  while prioridade(f) >
     menorPrioridade(t_H) do
5  |  g_D ← g_D ∪
   |    menorPrioridade(t_H)
6  |  t_H ← t_H −
   |    menorPrioridade(t_H)
7  |  if U_D >= trafego(f)
   |    then
8  |  |  t_L ← t_L ∪ g_D
9  |  |  t_H ← t_H ∪ f
10 |  |  retorno
11 |  end
12 end
13 t_H ← t_H ∪ g_D
```

Algorithm 3: ROTINA ALOCACAO-CONDICIONAL(F)

```
1  t_Lexc :
     trafego excedente no tunel de baixa prioridade
2  f_drop :
     taxa de descarte para um fluxo
3  l_drop :
     grupo de taxas de descarte por fluxo
4  l_drop = 0
5  if U_L > τ_Lmax then
6  |  t_Lexc = U_L − τ_Lmax
7  |  ordena(t_L)
8  |  for fluxo in t_L do
9  |  |  f_drop = ((1 −
   |  |    prioridade(fluxo))+
   |  |    (prioridade(fluxo) ∗
   |  |    0.1)) ∗
   |  |    traffic(fluxo)
10 |  |  if f_drop < t_Lexc then
11 |  |  |  l_drop = l_drop ∪ f_drop
12 |  |  |  t_Lexc = t_Lexc − f_drop
13 |  |  end
14 |  |  else
15 |  |  |  l_drop = l_drop ∪ t_Ldrop
16 |  |  |  return
17 |  |  end
18 |  end
19 end
```

Algorithm 4: POLICIAMENTO DE TRÁFEGO UTILIZANDO UM SISTEMA DE REPUTAÇÃO

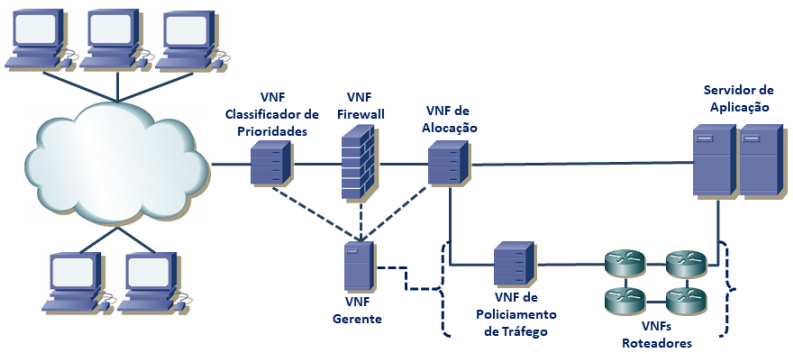

Figura 1. Arquitetura DeMONS

Para beneficiar fluxos com alta prioridades alocados no túnel de baixa prioridade, a solução DeMONS utiliza uma VNF de Policiamento de Tráfego. Essa VNF determina o tráfego máximo que será aceito para cada fluxo considerando as suas prioridades. Esse sistema de reputação objetiva reduzir a sobrecarga de maneira controlada. A rotina verifica os fluxos em ordem crescente de prioridade, para fluxos com a mesma prioridade a ordem decrescente de quantidade de tráfego gerado é considerada para uma ordenação secundária.

O Policiamento de Tráfego determine um fator de descarte (fd) de acordo com a prioridade de determinado fluxo, sendo definido um fator de descarte mínimo de 10% como demonstrado na Equação 4. A solução objetiva computar completamente o maior número possível de fluxos benignos, dessa forma mesmo fluxos com prioridade 1 podem sofrer descartes evitando que apenas um único fluxo consuma uma grande parcela da

capacidade do túnel. A aplicação da política para quando o limite da capacidade do túnel é alcançada ou quando todos os fluxos atravessando o túnel de baixa prioridade são verificados, indicando que, mesmo com o descarte controlado de pacotes, alguns descartes aleatórios ainda podem ocorrem. O Algoritmo 4 mostra como o sistema de reputação foi implementado.

$$fd = ((1 - prioridade(f)) + (prioridade(f) * 0.1)) * trafego(f) \tag{4}$$

5. Validação e Testes

Uma série de simulações foram realizadas para validar a proposta [1]. A metodologia empregada na simulação se baseou na utilizada em [Fung and McCormick 2015], com tanto o túnel de baixa quanto o de alta prioridade com 50 Mbps de capacidade cada e com o túnel de alta prioridade entrando em modo seletivo quando o tráfego alcança 97% de sua capacidade. Fluxos benignos são gerados com uma taxa de 100 Kbps, durando um período de 10 segundos com 10 Kbps de degradação de tráfego por segundo. Fluxos maliciosos também geram 100 Kbps de tráfego cada, entretanto a taxa de transmissão não degrada durante o ataque.

Cada simulação dura 30 segundos, com as informações dos túneis e dos fluxos (*e.g.*, quantidade de tráfego, prioridades) sendo atualizadas a cada segundo. Prioridades são atribuídas aleatoriamente para cada fluxo, fluxos benignos apresentam prioridades entre 0.4 e 1, enquanto fluxos maliciosos tem prioridades entre 0.1 e 0.4. Prioridades 0.4 são consideradas uma zona intermediária, onde alguns fluxos são maliciosos e outros são benignos. Fluxos com prioridade zero (*i.e.*, reconhecidamente maliciosos) não foram gerados uma vez que são bloqueados por um *Firewall*. Três cenários com o objetivo comparar as soluções VGuard e DeMONS são definidos e apresentados nas subseções 5.1, 5.2 e 5.3 e uma análise dos Sistemas de Reputação do Policiamento de Tráfego da solução DeMONS nesses mesmos cenários é apresentada na subseção 5.4.

5.1. Fluxos Benignos e Tráfego Normal

Nesta simulação, todo o tráfego gerado é considerado benigno e apresenta um pico de 99.1 Mbps e distribuição de acordo com a Equação 5 que indica a quantidade de tráfego no tempo t em segundos. Desde que a quantidade de tráfego não exceda a capacidade dos túneis em nenhum momento, todo o tráfego de rede alcança o servidor tanto na solução VGuard e DeMONS. A satisfação de ambos os túneis de baixa e alta prioridade são apresentados respectivamente nas Figuras 2 e 3.

$$f(t) = \frac{66t}{5} - \frac{11t^2}{25} \tag{5}$$

A Figura 3 ilustra o momento quando o túnel de alta prioridade entra em modo seletivo (de $t = 13$ a $t = 16$) e o túnel de baixa prioridade é sobrecarregado em ambas as soluções. Neste caso, o túnel de baixa prioridade do VGuard e do DeMONS opera com um nível de satisfação de, respectivamente, 1,1% e 0,6% abaixo do máximo alcançável.

[1] GitHub: https://github.com/ViniGarcia/DeMONS-PoC

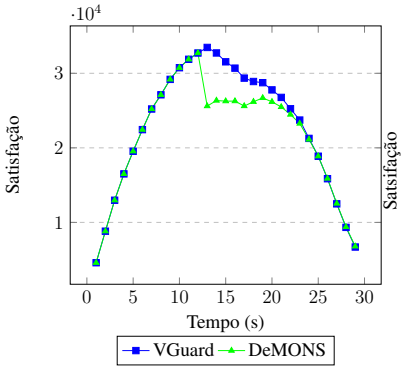

Figura 2. TBP (Eq. 5)

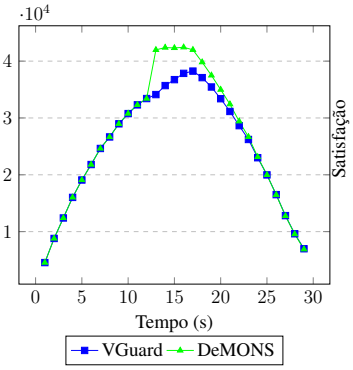

Figura 3. TAP (Eq. 5)

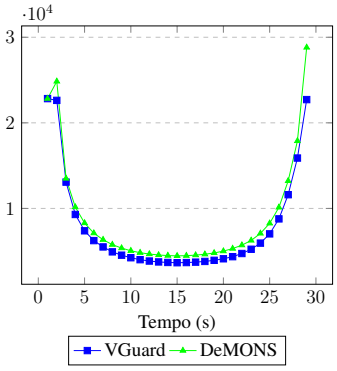

Figura 4. TBP (Eq. 6)

Quando o túnel de alta prioridade entra em modo seletivo, o balanceamento de fluxos realizado no DeMONS ($t = 13$) resulta em uma mudança abrupta nas medidas dos níveis de satisfação. Uma queda instantânea ocorre no túnel de baixa prioridade (uma vez que ele recebe os fluxos com menores prioridades) e uma elevação instantânea é notada no túnel de alta prioridade (onde os fluxos de mais alta prioridade passam).

5.2. Fluxos Benignos e Sobrecarga de Tráfego

Com uma geração de tráfego benigno, distribuído de acordo com a Equação 6 onde t é o tempo em segundos, cinco vezes maior que a capacidade agregada dos túneis (pico de 506 Mbps), esta simulação impõem uma sobrecarga significante a solução. Uma vez que o túnel de alta prioridade não aceita mais tráfego do que sua capacidade, muito mais tráfego é forçado a passar através do túnel de baixa prioridade. A satisfação do túnel de baixa prioridade – representado na Figura 4 – decresce de acordo com que o número de fluxos aumentam além de sua capacidade. A Figura 5 mostra um zoom dos momentos de maior tráfego no túnel de baixa prioridade, enquanto a satisfação do túnel de alta prioridade é demonstrado na Figura 6.

$$f(t) = \frac{135t}{2} - \frac{9t^2}{4} \tag{6}$$

O túnel de alta prioridade entra em modo seletivo em $t = 2$, e continua nele até $t = 27$ como ilustrado na Figura 6. A satisfação do túnel de baixa prioridade apresenta uma abrupta e prematura queda e mantém esses níveis até os momentos finais da simulação. O uso do sistema de reputação no DeMONS leva em um aumento de satisfação de 21,3% em comparação ao VGuard.

Devido ao rápido balanceamento do túnel de alta prioridade do DeMONS, a satisfação aumenta rapidamente no início da simulação. A satisfação no túnel de alta prioridade do VGuard leva mais tempo para aumentar já que depende de fluxos com baixa prioridade terminem e fluxos com mais alta prioridade assumam esses lugares e, dessa forma, o VGuard alcança maiores taxas de satisfação, como visto nos momentos 16 e 22 na Figura 6.

5.3. Ataque DDoS

O cenário de ataque DDoS foi simulado gerando fluxos benignos (com uma taxa máxima de 99.1 Mbps representada em 5) e um tráfego malicioso contínuo de 500 Mbps

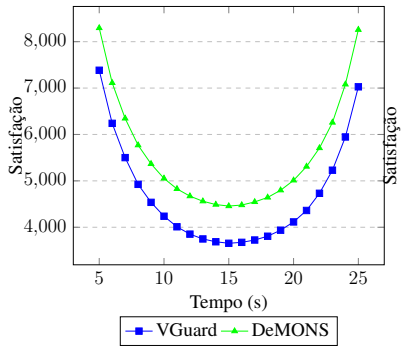

Figura 5. Zoom TBP (Eq. 6)

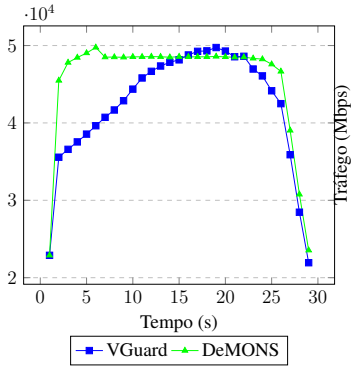

Figura 6. TAP (Eq. 6)

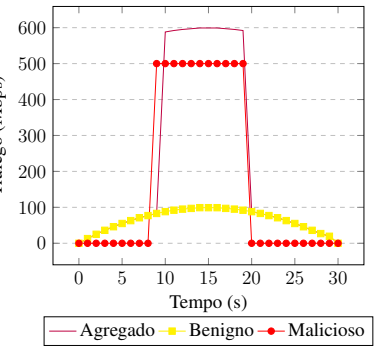

Figura 7. Cenário de Ataque DDoS

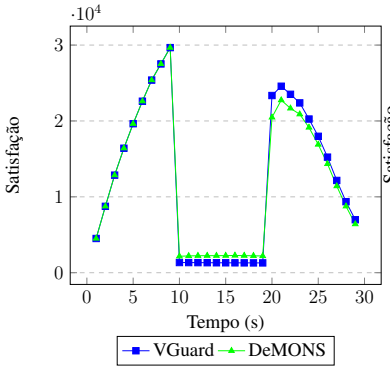

Figura 8. TBP (DDoS)

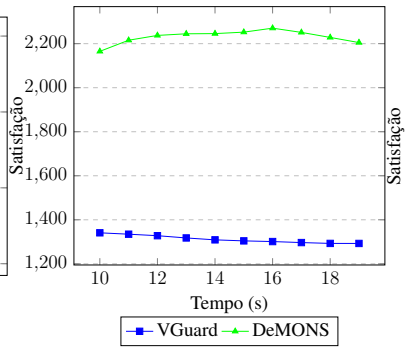

Figura 9. Zoom TBP (DDoS)

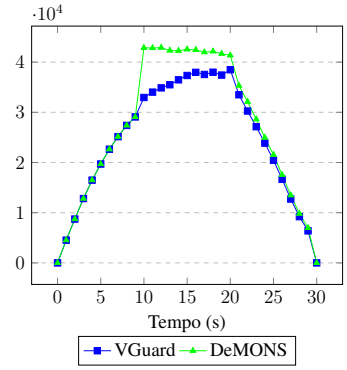

Figura 10. TAP (DDoS)

começando em $t = 10$ e terminando em $t = 20$. A Figura 7 ilustra os fluxos benignos e maliciosos, além da distribuição do tráfego submetido ao sistema.

Devido a sobrecarga de tráfego malicioso e a baixa prioridade apresentada pelos mesmos, uma alocação massiva no túnel de baixa prioridade ocorre. Isso gera uma taxa de satisfação baixa abrupta e contínua. O túnel de alta prioridade mantém a entrega dos fluxos normalmente, mantendo o tráfego abaixo da capacidade máxima. A satisfação dos túneis de baixa e alta prioridade são apresentadas respectivamente nas Figuras 8 e 10, enquanto 9 ilustra um zoom da satisfação túnel de baixa prioridade quando o ataque DDoS está ocorrendo.

O túnel de alta prioridade entra em modo seletivo no inicio do ataque DDoS e mantém esse estado até o tráfego malicioso cessar. Os fluxos maliciosos são alocados no túnel de baixa prioridade devido suas baixas prioridades. Dessa forma, o túnel de baixa prioridade apresenta baixos níveis de satisfação e uma vez que está sobrecarregado altas taxas de descarte são verificadas. Entretanto, o sistema de reputação empregado no DeMONS é capaz de aumentar a satisfação do túnel de baixa prioridade através do descarte de boa parte do tráfego malicioso.

O balanceamento de carga executado pelo DeMONS (quando o túnel de alta prioridade entra em modo seletivo), além de manter os fluxos de mais alta prioridade no túnel de alta prioridade, também remove todos os fluxos maliciosos que cruzam por este

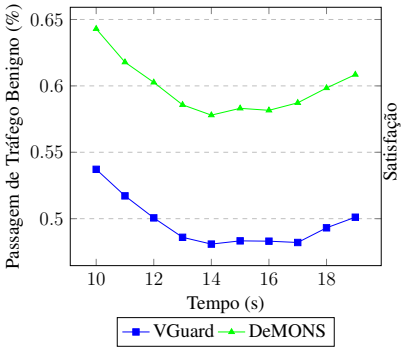

Figura 11. TPTB (DDoS)

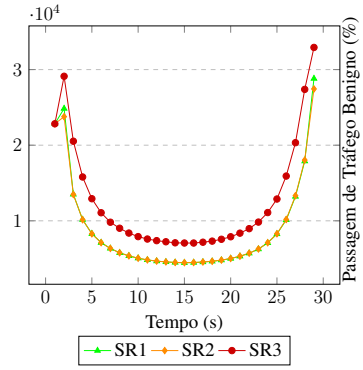

Figura 12. Filtros em TBP (Eq. 6)

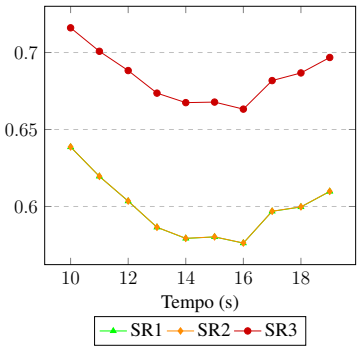

Figura 13. TPTB entre Filtros (DDoS)

túnel antes da entrada em modo seletivo, uma vez que eventuais fluxos maliciosos podem acessar este canal enquanto o ataque não é detectado. Esse fluxos maliciosos geram 6.9 Mbps de tráfego no túnel de alta prioridade. Uma vez que esse fluxos não degradam nem terminam antes do fim do ataque, a presença deles no túnel de alta prioridade representa um eminente desperdício de recursos.

A Figura 11 apresenta a Taxa de Passagem de Tráfego Benigno (TPTB), ou seja, porcentagem de dados que alcançou o servidor para cada solução testada. O VGuard apresentou os piores resultados devido aos fluxos maliciosos atravessando o túnel de alta prioridade e o alto número de fluxos benignos no túnel de baixa prioridade, onde altas taxas de descartes são constatadas.

A solução DeMONS obteve melhores resultados para todos os cenários testados. O sistema de reputação no túnel de baixa prioridade obteve sucesso no descarte de boa parte dos fluxos maliciosos devido suas baixas prioridades. A utilização eficiente do túnel de alta prioridade é devida ao processo de balanceamento dos fluxos de rede, aliado a isso, o sistema de reputação no túnel de baixa prioridade garante boas taxas de entrega do tráfego benigno.

5.4. Sistemas de Reputação

O módulo de Policiamento de Tráfego pode utilizar diferentes Sistemas de Reputação. Essas implementações implicam em uma maior ou menor influência na quantidade e prioridade do tráfego que atravessa o túnel de baixa prioridade. Sistemas de Reputação extremamente restritivos levam a eventuais descartes totais de fluxos com baixa prioridade, dificultando que o tráfego de fluxos ainda desconhecidos, porém benignos, sejam entregues ao servidor. Por outro lado, um sistema demasiadamente brando permite a passagem de grande parte do tráfego de baixa prioridade e, eventualmente, de ataques, atingindo assim baixos níveis de satisfação e TPTB no túnel.

Além do Sistema de Reputação integrado ao DeMONS (SR1 - Equação 4), outras duas possíveis implementações foram consideradas para a aplicação em dois cenários de teste: um menos restritivo, inferindo menores taxas de descarte a fluxos com baixa prioridade e não descartando totalmente nenhum fluxo (SR2 - Equação 7) e outro mais restritivo, relativo a quantidade de tráfego excedente no túnel de baixa prioridade e capaz de descartar a totalidade do tráfego de um fluxo (SR3 - Equação 8).

$$fd = (1 - prioridade(f)) * trafego(f) \tag{7}$$

$$fd = ((1 - prioridade(f)) + (1 - prioridade(f) + prioridade(f) * 0.1) * excedente(t_L)) * trafego(f) \tag{8}$$

Considerando o cenário de sobrecarga de fluxos benignos (Equação 6), a satisfação no túnel de baixa prioridade, ilustrada na Figura 12, utilizando os Sistemas de Reputação 1 e 2 são virtualmente iguais entre os momentos $t = 3$ e $t = 28$, apresentando uma variação máxima de 0,8%, enquanto que no $t = 2$ e $t = 29$ uma variação de, respectivamente, 4,3% e 4,8% a favor do SR1 é verificado devido ao maior descarte de fluxos de baixa prioridade em um cenário pouco sobrecarregado. O SR3 obteve maiores níveis de satisfação durante todo o período de avaliação, porém este descarta a totalidade do tráfego gerado por fluxos de baixa prioridade que, dessa forma, nunca são atendidos.

Já no cenário de ataque DDoS (Figura 7), a TPTB entre os Sistemas de Reputação, apresentada na Figura 13, mostra que, da mesma forma que a satisfação no teste de estresse do túnel de baixa prioridade, a diferença entre o SR1 e SR2 é mínima, nesse com vantagem entre 0,005% e 0,015% para o SR2. O SR3 garante o melhor TPTB já que é capaz de realizar o descarte total de fluxos maliciosos com baixa prioridade, obtendo resultados superiores entre 10,80% e 13,21% quando comparados ao SR1 e SR2.

6. Conclusão

Ataques DDoS são ameaças sofisticadas que geram uma quantidade massiva de tráfego de rede. A ascensão de novas tecnologias (*i.e.*, *Internet of Things*, *Fog Computing*) conduz a um aumento de equipamentos conectados a Internet. Esses equipamentos, entretanto, podem ser infectados e usados para realizar ataques significativos. Ao mesmo tempo, um novo paradigma – chamado NFV – apresenta características interessantes que suportam o desenvolvimento de novas soluções de mitigação de ataques DDoS.

Este artigo propõe uma nova solução híbrida para mitigar DDoS chamada DeMONS. A solução é composta de cinco módulos principais (implementados como VNFs): Classificador de Prioridades, *Firewall*, Alocador de Fluxos, Policiamento de Tráfego e Gerente. No DeMONS, fluxos de rede são analisados pelo Classificador de Prioridades e marcados com uma reputação entre 0 e 1. Fluxos com prioridade zero são bloqueados no *Firewall*, enquanto que os demais fluxos são atribuídos a túneis distintos (alta e baixa prioridade) pelo módulo Alocador de Fluxos. Quando o túnel de baixa prioridade está sobrecarregado, o módulo de Policiamento de Tráfego executa um algoritmo para limitar o tráfego de cada fluxo baseado em suas prioridades. O módulo Gerente é responsável pelo provisionamento e controle do ciclo de vida de todos os demais módulos.

Os resultados obtidos demonstram a viabilidade do DeMONS para eficientemente mitigar ataques DDoS. O uso do sistema de reputação, através do módulo de Policiamento de Tráfego, e da efetividade do método de alocação, possibilitou o alcance de melhores resultados quando comparados ao VGuard nos cenários testados. É importante ressaltar que, apesar dos aprimoramentos alcançados na mitigação de DDoS, o DeMONS também provê uma melhor Qualidade de Serviço (QoS) na perspectiva do usuário final.

Como trabalhos futuros, o impacto da desativação e reativação de módulos será profundamente avaliada. Também, diferentes políticas serão simuladas para aplicação no

módulo de Policiamento de Tráfego. Finalmente, uma avaliação mais abrangente, considerando diferentes cenários com distribuições fluxos e prioridades diversas, será realizada para uma maior compreensão das capacidades da solução.

Referências

Alharbi, T., Aljuhani, A., and Liu, H. (2017). Holistic ddos mitigation using nfv. In *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 1–4.

Douligeris, C. and Mitrokotsa, A. (2004). Ddos attacks and defense mechanisms: classification and state-of-the-art. *Computer Networks*, 44(5):643–666.

ETSI, N. (2012). Network functions virtualization, white paper.

Fung, C. J. and McCormick, B. (2015). Vguard: A distributed denial of service attack mitigation method using network function virtualization. In *2015 11th International Conference on Network and Service Management (CNSM)*, pages 64–70.

Garber, L. (2000). Denial-of-service attacks rip the internet. *Computer*, 33(4):12–17.

Jakaria, A. H. M., Yang, W., Rashidi, B., Fung, C., and Rahman, M. A. (2016). Vfence: A defense against distributed denial of service attacks using network function virtualization. In *2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC)*, volume 2, pages 431–436.

John, W., Pentikousis, K., Agapiou, G., Jacob, E., Kind, M., Manzalini, A., Risso, F., Staessens, D., Steinert, R., and Meirosu, C. (2013). Research directions in network service chaining. In *2013 IEEE SDN for Future Networks and Services (SDN4FNS)*, pages 1–7.

Marcuzzo, L. d. C., Garcia, V. F., Cunha, V., Corujo, D., Barraca, J. P., Aguiar, R. L., Schaeffer-Filho, A. E., Granville, L. Z., and Santos, C. R. P. d. (2017). Click-on-osv: A platform for running click-based middleboxes. In *2017 IFIP/IEEE International Symposium on Integrated Network Management*. IEEE.

Rashidi, B., Fung, C., and Bertino, E. (2017). A collaborative ddos defence framework using network function virtualization. *IEEE Transactions on Information Forensics and Security*, PP(99):1–1.

Sherry, J., Hasan, S., Scott, C., Krishnamurthy, A., Ratnasamy, S., and Sekar, V. (2012). Making middleboxes someone else's problem: Network processing as a cloud service. *SIGCOMM Comput. Commun. Rev.*, 42(4):13–24.

StarHub (2016). Starhub confirms cause of home broadband incidents on 22 october and 24 october 2016. `http://bit.ly/2i8Uef0`. Accessed: 2017-05-26.

Yang, X., Wetherall, D., and Anderson, T. (2008). Tva: a dos-limiting network architecture. *IEEE/ACM Transactions on Networking (ToN)*, 16(6):1267–1280.

Zhang, W., Liu, G., Zhang, W., Shah, N., Lopreiato, P., Todeschi, G., Ramakrishnan, K., and Wood, T. (2016). Opennetvm: A platform for high performance network service chains. In *Proceedings of the 2016 Workshop on Hot Topics in Middleboxes and Network Function Virtualization*, HotMIddlebox '16, pages 26–31, New York, NY, USA. ACM.