Lógica Paraconsistente Anotada Aplicada na Detecção de Ataques DDoS em Redes SDN*

Euclides P. Farias Junior[1,2], **Allainn C. Jacinto Tavares**[2], **Michele Nogueira**[1]

[1]Centro de Ciência de Segurança Computacional (CCSC) – UFPR

[2]Grupo de Estudos em Redes e Segurança Computacional (GENETSEC) – UTFPR

{epfjunior,michele}@inf.ufpr.br, allainn@alunos.utfpr.edu.br

Abstract. *This short paper evaluates the use of the Annotated Paraconsistent Logic (LPA) for detecting Distributed Denial of Service (DDoS) attacks in Software Defined Networks (SDN). A system called LPAprog was developed, and it performs the detection through the Cartesian Plane Unitary Square (CPUS) in a emulated network by Mininet, with POX controller. LPAprog indicates the degrees of disbelief in grid frames and calculates weights using the total number of packets, number of SYN and FIN packets. LPAprog results show efficiency in the detection of DDoS SYN-Flooding attacks on packet cycles and windows of size above 200 and 10 seconds.*

Resumo. *Este artigo curto avalia o uso da Lógica Paraconsistente Anotada (LPA) para detecção de ataques de negação de serviços distribuídos (DDoS) em Redes Definidas por Software (SDN). Foi desenvolvido um sistema intitulado LPAprog que faz a detecção dos ataques através do Quadrado Unitário de Plano Cartesiano (QUPC) em uma rede emulada pelo Mininet, com controlador POX. O LPAprog indica os graus de descrenças nos quadros reticulados e calcula os pesos através da quantidade total de pacotes, quantidade de pacotes SYN e FIN. Os resultados do sistema LPAprog mostram a eficiência na detecção dos ataques DDoS SYN-Flooding em ciclos e janelas de tamanho acima de 200 pacotes e a partir de 10 segundos.*

1. Introdução

As Redes Definidas por *Software* (do inglês, *Software Defined Networks* – SDNs) desacoplam o plano de dados do plano de controle oferecendo maior dinamicidade comparado às redes tradicionais. As principais características das SDNs estão no plano de controle, que é responsável pela inteligência da rede, pelo aprendizado do encaminhamento dos pacotes através dos códigos dos protocolos de roteamento. O plano de dados é responsável por encaminhar pacotes com base em regras simples, associadas a cada entrada da tabela de encaminhamento do dispositivo [Kreutz et al. 2015].

A SDN é considerada um alvo potencial dos ataques DDoS, pois muitas vezes segue uma topologia centralizada, sendo facilmente comprometida por ataques desta natureza. Esta afirmação é apoiada na definição sobre controladores, como é o caso do POX com o protocolo OpenFlow, que tem a responsabilidade de configurar todos os dispositivos da rede. O controlador é responsável por manter as informações da topologia e monitorar o estado global da rede. Além disso, o OpenFlow provê interfaces para a criação,

*Este artigo teve apoio financeiro do CNPq #432204/2018-0 e projeto RNP/GT-Periscope.

modificação e controle de uma tabela de fluxos na rede, com a função de comutação (*switch*) [Kreutz et al. 2015].

Os ataques DDoS fazem uso de máquinas infectadas com *software* maliciosos (*bots*) a fim de sobrecarregar um alvo com uma grande quantidade de requisições e tráfego. Este tipo de ataque é agressivo em função da possibilidade de um grupo massivo de *hosts* se direcionar como atacantes a um único servidor. A vulnerabilidade que a SDN apresenta com relação a ataques DDoS é um assunto de destaque na comunidade científica [Liang and Znati 2019]. Por exemplo, o sistema RADAR (Reforçar as Ações Anti-DDoS em Tempo Real) detecta os ataques DDoS por meio de análise de correlação adaptativa, aplicados em comutadores SDN comerciais [Dayal and Srivastava 2018]. Outro trabalho apresentou uma abordagem baseada em algoritmos de inteligência artificial, o qual construiu um módulo de detecção baseado na Rede de Função Radial e também fez uso de redes neurais, para detecção precoce de ataques DDoS na SDN. O trabalho apresentado por [Rai et al. 2019] aborda o conceito de entropia para observar a sua variação durante um ataque ao controlador.

Neste trabalho, avalia-se o uso da Lógica Paraconsistente Anotada (LPA) para detectar ataques DDoS do tipo *SYN Flooding*. Este ataque explora o mecanismo de aperto de mão em três etapas (*Three-Way Handshake*) do protocolo TCP entre a origem e o destino, o qual produz várias conexões TCP abertas sem que estas conexões finalizem o processo das etapas de aceitação para que se estabeleça uma comunicação entre ambos, quando aplicados à SDN, o ataque TCP *SYN flooding* gera saturação do plano de controle da rede [Mohammadi et al. 2017]. Este trabalho apresenta o sistema LPAprog, cuja função é capturar os pacotes e analisá-los seguindo a LPA. A cada janela de captura é feita a avaliação baseado no total geral dos pacotes, na quantidade de pacotes FIN e na quantidade de pacotes SYN de forma a indicar a existência ou não de um ataque.

Para a avaliação do LPAprog, foi utilizado o emulador de redes Mininet em SDN com um controlador POX. A topologia da rede foi do tipo árvore (com 13 *switches* e 27 *hosts*), onde o *host* 13 foi determinado como a vítima dos ataques. A partir desta infraestrutura, foram gerados ataques com a ferramenta T50 para DDoS, a fim de avaliar a eficiência do LPAprog diante da inundação de pacotes SYN_flood. Através dos testes, foi possível averiguar o ataque DDoS, bem como avaliar a contagem de pacotes na fila e os intervalos de tempos dos ataques reproduzidos na rede.

2. Lógica Paraconsistente Anotada (LPA)

A Lógica Paraconsistente Anotada é uma lógica não clássica fundamentada na revogação do princípio da *Não Contradição* [da Silva Filho 2010]. Esta lógica admite o tratamento de informações contraditórias na sua estrutura teórica, sem trivialização. A LPA cria um quadro reticulado de quatro vértices, onde intuitivamente as constantes de anotações são representadas nos vértices e dão conotações de estados lógicos extremos às proposições.

O Quadrado Unitário de Plano Cartesiano (QUPC) apresenta valores X e Y variando num intervalo real fechado [0,1], de modo que estes valores representam respectivamente os graus de crença, μ_1 e de descrença, μ_2. O QUPC é dividido em doze regiões (Figura 1), e suas respostas são apresentadas em quatro estados lógicos, onde:

- **(1;0)** - representa **Verdade** (**V**), a crença total e nenhuma descrença;
- **(0;1)** - representa **Falso** (**F**), nenhuma crença e descrença total;

- **(1;1)** - representa **Inconsistência** ($\top$), ao mesmo tempo a crença e descrença total;
- **(0;0)** - representa **Paracompleteza** ou de **Indeterminação** ($\bot$), ausência total de crença e de descrença.

A linha de limite de paracompleteza e de inconsistência são representadas por z_1, enquanto a linha de limite de falsidade e de verdade são representadas por z_2. Convencionalmente, é adotada a notação $z_1 = z_2 = z$ de forma a gerar uma simetria ao gráfico, como mostra a Figura 1, sendo $z_1 = z_2 = z = 0,60$. Desta forma, o valor de z_2 é chamado de nível de exigência, considera z_2 sendo o nível de exigência por ser limitante de regiões de decisões. Por este motivo, são destacadas na Figura 1 as quatro regiões extremas e uma região central. As regiões CPQ e DTU são chamadas de regiões de decisões, sendo CPQ uma decisão favorável e o DTU uma decisão desfavorável. Partindo deste princípio, os pontos no plano cartesiano MNTUSRQP representam uma única região que não permite tomada de decisão, ou seja, quando o ponto que traduz o resultado da análise pertence a essa região diz-se que a análise é não conclusiva. É importante ressaltar que, se o resultado estiver na região BRS, a análise é não conclusiva quanto à viabilidade do ataque, mas acusa um alto grau de inconsistência dos dados. Analogamente, se estiver na região AMN, significa que os dados apresentam um alto grau de indeterminação. A Tabela 1 apresenta o intervalo de cada região dentro do quadrante QUPC. Além destas indicações dentro do quadro reticulado faz-se uso do indicador denominado baricentro. O baricentro é a média dos graus resultantes, sendo obtido através da soma dos graus de crença resultantes e dividido pela quantidade de graus resultantes, realizando o mesmo processo para os graus de descrença resultantes, assim se obtém um valor médio de grau de crença e descrença resultantes. O baricentro tem como função a tomada de decisão final, sendo ele a influência conjunta de todas as análises realizadas. Desta forma, este indicador é representado por um círculo maior, onde no gráfico tem a função de orientação de como se comportam os dados a partir de seu fluxo dentro da rede.

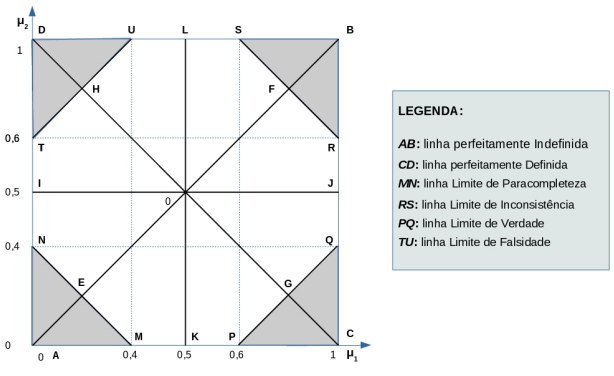

Figura 1. Quadro QUPC com $|G_{contr}| = 0,60$ e $|H_{cert}| = 0,60$. Adaptado de [Carvalho 2003].

Região	Condicional	Resultante
Região AMN	-1 $\leq$ G_{contr} $\leq$ -0,60	$\Rightarrow$ região de paracompleteza
Região BRS	0,60 $\leq$ G_{contr} $\leq$ 1	$\Rightarrow$ região de inconsistência
Região CPQ	0,60 $\leq$ H_{cert} $\leq$ 1	$\Rightarrow$ região de verdade
Região DTU	$-1\leq$ H_{cert} $\leq$ 0,60	$\Rightarrow$ região de falsidade

Tabela 1. Tabela de Regiões, adaptado de [da Silva Filho 2010]

A motivação em aplicar a LPA na detecção de DDoS em SDN se dá pela possibilidade de explorar a sua funcionalidade principal, ou seja o uso dos operadores lógicos (1;0) que representa a crença total; (0;1) nenhuma crença e descrença total; (1;1) a crença e descrença efetivamente total; e por fim, (0;0) a ausência total de crença e descrença. A avaliação feita de forma quadrática mostra-se eficiente e com um potencial de descoberta

do conhecimento para auxiliar na tomada de decisão. Então, ao a aplicar a LPA em SDN, uma vez que esta rede possui a característica de programabilidade, torna-se justificável a exploração desta lógica na busca por novas soluções para detectar e explorar vulnerabilidades DDoS em SDN.

3. Sistema LPAprog

O sistema LPAprog neste trabalho é composto por um conjunto de programas os quais implementam a LPA e os *scripts* para apoio na coleta e tratamento das informações geradas para plotagem dos gráficos. O sistema aciona um programa para calcular pesos e recebe deste módulo um vetor com os valores de graus de crença e descrença. Na sequência, é atribuída em uma variável o valor do nível de exigência. Trata-se de um valor aleatório e, a partir deste momento, o cálculo da LPA é realizado atribuindo um *AND* entre os graus de crenças e descrenças. Este *AND* recebe o valor mínimo entre ambos e adiciona nos vetores de graus de crença e descrença resultantes. A partir daí é realizado o cálculo do grau de certeza através da expressão $(u1 - u2)$ e do grau de descrença $(u1 + u2 - 1)$.

Para se obter os graus de crenças e descrenças são realizados os seguintes cálculos: $\mu_{11} = QS/QT$, $\mu_{21} = 1 - \mu_{11}$, $\mu_{22} = QF/QS$ (caso a divisão seja maior que 1, $\mu_{22} = 1,00$), $\mu_{12} = 1 - \mu_{22}$, onde μ_{11} e μ_{12} são os graus de crenças; μ_{21} e μ_{22} são os graus de descrenças; QT é a quantidade total de pacotes; QS a quantidade de pacotes SYN; e QF a quantidade de pacotes FIN. Caso QT seja igual a 0 (zero), são atribuídos os seguintes valores: $\mu_{11} = 0,00$, $\mu_{21} = 1,00$, $\mu_{12} = 0,00$, $\mu_{22} = 1,00$, sendo que não é possível que seja um ataque quando não se há pacotes. Caso QS seja igual a 0 (zero) são atribuídos os seguintes valores: $\mu_{12} = 0,00$, $\mu_{22} = 1,00$, sendo que não é possível existir um ataque do tipo SYN *Flooding* quanto não existam pacotes SYN na rede.

Desta forma, para realizar o cálculo do grau de crença e descrença do baricentro, deve-se realizar a soma dos graus resultantes e dividir pela sua quantidade. O grau de certeza do baricentro é dado através da expressão $(w1 - w2)$ e o grau de contradição do baricentro, é dado por $(w1 + w2 - 1)$. A partir destes dados imputados e processados, o LPAprog mostra na tela vetores e os resultados obtidos de cada um, bem como chama um programa para gerar os gráficos para informar os graus de crença e descrença resultantes, as cores dos pontos inclusive do baricentro, o nível de exigência e os limites dos gráficos a serem plotados. Desta forma, o LPAprog possui variáveis indicadas como: n = Nível de exigência; $u1$ = Grau de crença; $u2$ = Grau de descrença; $u1r$ = Grau de crença resultante; $u2r$ = Grau de descrença resultante; $w1$ = Grau de crença do Baricentro, $w2$ = Grau de descrença do Baricentro; h = Grau de certeza; g = Grau de contradição, hw = Grau de certeza do Baricentro; gw = Grau de contradição do Baricentro. Estas variáveis são definidas como o ponto focal do sistema LPAprog, o qual realiza todo o processo de análise de avaliação dos pacotes em função de janelas pré-estabelecidas de tempo.

4. Metodologia

A metodologia de avaliação empregada neste trabalho tomou como base a implementação do sistema LPAprog, os *scripts shell* implementados e os programas para calcular os pesos, escritos na linguagem de programação Python. O ambiente de avaliação foi implementado no emulador de redes Mininet, executado sobre o sistema operacional Linux Ubuntu Server 18.04.02 LTS, em ambiente virtual (VirtualBox) e instalado em um computador Intel i5, com 8Gb de memória RAM. Configurou-se o cenário de avaliação no

emulador e optou-se por construir uma topologia de rede tipo árvore, com controlador POX L2_learning, com dimensões 3x3, cuja infraestrutura desta configuração gera uma rede contendo 13 *switches*, onde cada *switch* de borda possui três *hosts* sequenciais de 1 a 27. Baseado nesta infraestrutura, foi escolhido o *host* 13 para servir como servidor HTTP e é a vítima a ser atacada pelos demais *hosts* da rede. Os demais *hosts* são os principais veículos de ataque ao servidor HTTP. Para o ataque, foi utilizada a ferramenta T50, a qual executa testes de *stress* em infraestruturas de redes. Os ataques foram efetuados gradativamente, onde todos os *hosts* inicialmente foram iniciados, como ilustra a Figura 2, a partir do gráfico 1 até o 4, de forma que no momento em que os ataques ocorrem, o *host* 13 ficou cada vez mais comprometido.

O sistema LPAprog consegue representar a evolução da detecção. Ele gera gráficos baseado no QUPC, onde cada intervalo de tempo que gera a matriz é conhecido por $T(n)$. Desta forma, neste trabalho, construiu-se o QUPC com intervalos de 0,0 até 1,0. Os dados são extraídos da rede através da ferramenta tcpdump e todos os dados formam um arquivo de *log*. Cada interação é adicionada no final da linha os valores, sendo que a cada 10 interações, este arquivo é reinicializado. O *shell script* chama o módulo responsável por realizar o cálculo da LPA. A validação dos graus de certeza e contradição foi realizada de forma a gerar os resultados que mostram a existência das possíveis condições: ataque; não ataque; ou não conclusivo. Assim, o sistema finaliza com a geração dos gráficos e salva as imagens de forma sequencial, a contar do processo 1 até 10 como ciclo de geração de evolução do sistema.

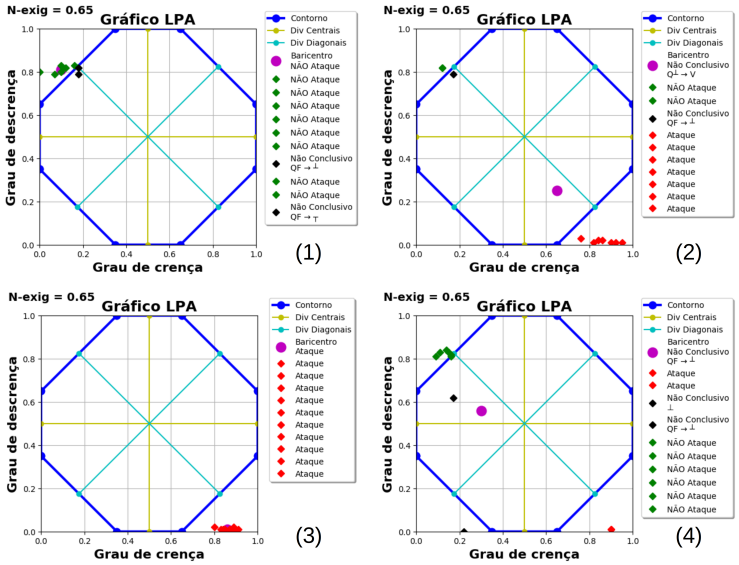

Figura 2. Evolução do QUPC durante o processo de detecção DDoS

5. Resultados

O sistema apresenta a evolução da detecção de ataques através de um QUPC, cujo propósito é analisar os graus de crenças e descrenças nos quadros reticulados, como mostra a Figura 2. Esta apresenta evoluções do ataque passo a passo. A Figura 2 apresenta um conjunto de quatro LPA (gráficos), onde no eixo X representa o Grau de crença e no eixo Y o grau de descrença. Desta forma, no gráfico LPA (1), o baricentro e a maioria dos indicadores estão localizados dentro da região de inviabilidade, o que significa não

existir ataque. No gráfico LPA (2), o baricentro encontra-se em uma região não conclusiva, porém a maioria dos indicadores, estão na região de viabilidade, representando que o sistema está em estado de ataque, porém não conclusivo. No gráfico LPA (3), o baricentro e todos os indicadores encontram-se dentro da região de viabilidade, representando agora que o sistema está em ataque iminente. Por fim, no gráfico LPA (4) após intervenção para encerrar os ataques, o baricentro se desloca novamente para a região não conclusiva, indicando que o baricentro caminha de volta para a região de não ataque.

6. Conclusão

Este trabalho apresentou LPAprog, um sistema que faz uso da Lógica Paraconsistente Anotada (LPA) para a detecção de ataques *DDoS SYN-Flooding* em SDN. LPA estende da lógica clássica e obtém resultados além da verdade e falsidade, sendo possível obter respostas como a inconsistência e a indeterminação. O sistema gera gráficos indicadores através de um baricentro. Nestes gráficos foram exploradas informações apresentadas pela diagonal principal da matriz, o que significa estado total ou parcial de um ataque. Na matriz, quanto maior a aproximação dos pontos distribuídos no eixo $Y = (0, 6)$ significa a proximidade da certeza de não ataques. Quanto maior a aproximação dos pontos distribuídos do eixo $X = (0, 6)$ significa a proximidade da certeza de ataque. Os resultados mostraram a evolução do ataque em um cenário de avaliação controlado. Toda a evolução do processo de análise de pacotes, desenvolvido no sistema LPAprog, mostra a eficiência e a eficácia na detecção de ataques DDoS. Como trabalho futuro, pretende-se explorar esta lógica em *datasets* de ataques reais e comparar com outras técnicas de detecção.

Referências

Carvalho, F. R. d. (2003). Um estudo de tomada de decisão baseado em lógica paraconsistente anotada: avaliação do projeto de uma fábrica. *Revista Pesquisa e Desenvolvimento Engenharia de Produção*, (1):47–62.

da Silva Filho, J. I. (2010). Lógica Paraquântica LPQ (parte I): introdução aos conceitos fundamentais. *Seleçao Documental: Inteligência Artificial e novas Tecnologias*, (18):17–26.

Dayal, N. and Srivastava, S. (2018). An RBF-PSO based approach for early detection of DDoS attacks in SDN. In *International Conference on Communication Systems & Networks (COMSNETS)*, pages 17–24. IEEE.

Kreutz, D., Ramos, F. M. V., Veríssimo, P. E., Rothenberg, C. E., Azodolmolky, S., and Uhlig, S. (2015). Software-Defined Networking: A Comprehensive Survey. *Proceedings of the IEEE*, 103(1):14–76.

Liang, X. and Znati, T. (2019). An empirical study of intelligent approaches to DDoS detection in large scale networks. In *International Conference on Computing, Networking and Communications (ICNC)*, pages 821–827.

Mohammadi, R., Javidan, R., and Conti, M. (2017). Slicots: An SDN-based lightweight countermeasure for TCP SYN flooding attacks. *IEEE Transactions on Network and Service Management*, 14(2):487–497.

Rai, A., D Vyavahare, P., and Jain, A. (2019). Distributed dos attack detection and mitigation in software defined network (sdn). *Anjana, Distributed DoS Attack Detection and Mitigation in Software Defined Network (SDN)(April 1, 2019)*.