Métricas F.O.R para a detecção de ataques Slow DoS

Dulcineia S Sennejunker*, Anderson F P dos Santos
Instituto Militar de Engenharia, Rio de Janeiro, Brasil.
Praça General Tibúrcio, 80, 22290-270, Praia Vermelha
Rio de Janeiro, RJ, Brasil.

RESUMO: *Recentemente, em consequência dos esforços de mitigação aos ataques DoS (Negação de Serviço) tradicionais, os ataques Slow DoS surgem como ameaça à garantia da disponibilização de serviços na Web. Esses ataques são considerados preocupantes devido a furtividade do seu modo de operação, que lentamente e sem alarde, desabilitam a vítima. Esse trabalho apresenta o estado da arte sobre o assunto e algumas estratégias relacionadas à detecção. Com base nestas estratégias, propôs-se as métricas F.O.R. (Flags, Omnia e RTTR), que implementadas em uma ferramenta para data mining, evidenciam a ocorrência de anomalias relacionadas aos ataques Slow DoS. Segundo a metodologia proposta e sob as condições da desigualdade Tchebycheff, discute-se os resultados da detecção. Isto, mediante a análise de um dataset que representa o ambiente de rede real, que mescla tráfego de fundo ao tráfego de ataque.*

PALAVRAS-CHAVE: *Slow DoS. Desigualdade Tchebycheff.*

ABSTRACT: *Recently, as a result of mitigation efforts of traditional DoS attacks, Slow DoS attacks have appeared as a threat to the availability of Web services. These attacks are concerning because of the stealthiness of their operation, which slowly and without flaunt, disable the victim. This work discusses the state of the art on the subject and introduces strategies related to detection. Based on these strategies, the article proposes the F.O.R. metrics, which implemented in a data mining tool evidence the occurrence of anomalies characteristic of Slow DoS attacks. The results of the detection, using the proposed methodology and the Tchebycheff inequality, are discussed. The results are obtained by analyzing a dataset that represents an actual network environment, which was merged background traffic to attack traffic.*

KEYWORDS: *Slow DoS. Tchebycheff Inequality..*

1. Introdução

Os ataques de negação de serviço são preocupantes, porque impedem que usuários legítimos acessem os serviços disponibilizados na Internet. Esses ataques evidenciam a fragilidade dos servidores, o que favorece o sucesso de atividades maliciosas. Em decorrência dos esforços para mitigação dos ataques à camada de transporte, algoritmos de detecção tais como os baseados em ASV (*Adaptative Selection Verification*) [1], para ataques distribuídos de negação de serviço (DDoS), demonstraram resultados positivos, e por isso, a motivação dos atacantes direcionou-os a outras camadas da arquitetura de redes. Assim, os atacantes que, anteriormente, visavam à camada de rede ou transporte, voltaram sua atenção à camada de aplicação.

No quarto trimestre de 2017, o Brasil ocupou o segundo lugar no *ranking* dos países destinos de ataques a aplicações *Web* [2], conforme Tabela 1, que apresenta o quantitativo de ataques por país.

Alguns ataques às aplicações *Web*, tais como HTTP *Flood* e DNS *Flood*, são volumétricos e visam saturar ou exaurir os serviços do alvo, como por exemplo, os servidores HTTP.

Além dos ataques de negação de serviço volumétricos ou tradicionais à camada de aplicação, existem também os ataques lentos e de baixa taxa com relação ao número de conexões HTTP, como por exemplo, o ataque *Slow* DoS.

Os ataques não volumétricos não são menos preocupantes que os ataques de negação de serviço tradicionais, pois conseguem também desabilitar a vítima.

A sofisticação dos atacantes contra os esforços de mitigação, configuram o cenário para mais uma das batalhas da guerra cibernética.

Este artigo propõe métricas a partir de estratégias para a detecção dos ataques Slow DoS. Em função das características dos ataques lentos de negação de serviço, as métricas têm por objetivo a detecção das anomalias relacionadas a deixar estes ataques.

Tabela 1: Países Alvos de Ataques

Posição	País	Nº de Ataques
1	E.U.A.	238.643.360
2	Brasil	21.900.411
3	Reino Unido	19.385.710
4	Canadá	17.459.934
5	Alemanha	13.432.389

Fonte: Com base no Relatório da Akamai Technologies [2], Relatório para o 4º Trimestre de 2017.

Na seção 2 desse artigo, descreve-se alguns conceitos básicos relacionados aos ataques de negação de serviço e o estado da arte dos ataques *Slow DoS* que são o foco deste trabalho. Esses conceitos serão necessários para entendimento da seção 3, que tratará dos atributos do tráfego de rede importantes na proposição das métricas de detecção de ataques *Slow DoS*. Na seção 4, apresenta-se a metodologia empregada para a detecção. E, na seção 5, a conclusão do trabalho.

2. Ataques de Negação de Serviço

A motivação dos atacantes varia muito em função da experiência, ideologia ou intenções. Independentemente do motivo, eles exploram vulnerabilidades nos sistemas ou equipamentos das vítimas para então, executar o ataque.

Algumas taxonomias [3,4] contextualizam os critérios para classificação dos ataques e dos mecanismos de defesa. Em função da quantidade de máquinas empregadas na ação, classifica-se o ataque em DDoS, quando várias máquinas são empregadas e DoS quando apenas uma máquina é empregada [5].

2.1 Ataques Distribuídos de Negação de Serviço

A negação de serviço, ou DoS, é uma técnica pela qual um atacante utiliza um computador para tirar de operação um serviço, um computador ou uma rede conectada à Internet [6].

O Ataque Distribuído de Negação de Serviço (DDoS) ocorre quando utilizado de forma coordenada e distribuída, ou seja, quando um conjunto de computadores é utilizado no ataque [6].

Existem duas categorias de ataques. Na primeira, o atacante lança o ataque mediante a exploração de vulnerabilidades, sendo, portanto, classificado como ataque de exploração de vulnerabilidades, utilizando *exploits* que são mensagens que exploram brechas de segurança nos sistemas das vítimas desguarnecidas. Assim, sistemas que não são atualizados frequentemente ou máquinas desprovidas de sistemas de proteção como *firewalls,* têm alta chance de serem cooptadas para empreender um ataque. A segunda categoria ocorre mediante grande volume de ataques (inundação), comumente chamada de ataque distribuído de negação de serviço (DDoS). Nessa forma, observa-se o envio brutal de mensagens, aparentemente legítimas, a um sistema alvo e que consomem recursos importantes da vítima, tais como memória, CPU e largura de banda do equipamento alvo [3]. Assim, o objetivo do ataque é atingido através do envio de grandes volumes de pacotes, que ocupam uma proporção significativa de largura de banda disponível, consumindo este recurso considerado crítico num serviço de rede [7].

2.1.1 Ataques de Negação de Serviço do Tipo Slow DoS

Dentre as diversas modalidades de negação de serviço, observa-se que a maioria explora as características dos protocolos no nível da camada de transporte tais como UDP, TCP e ICMP [8]. Mas, com os recentes esforços à mitigação dos ataques DDoS tradicionais, os atacantes concentram-se na realização de ataques à camada de aplicação.

O *Slow DoS* não se enquadra na categoria ataque DDoS tradicional, ou seja, ataque volumétrico, porque estes ataques beneficiam-se de vulnerabilidades na segurança, comandando várias máquinas infectadas, normalmente mal-intencionadas. Nestas máquinas, encontram-se instaladas aplicações maliciosas, as *bots,* que são comandadas a enviarem uma grande quantidade de tráfego de ataque [9]. O ataque *Slow DoS* é um ataque DDoS não tradicional, que faz juz ao termo DDoS, principalmente, porque realiza a abertura de inúmeras conexões para concretizar a negação do serviço de maneira furtiva. É um ataque que, diferentemente dos ataques DDoS tradicionais, atua sob o radar de detecção.

Face a diversidade dos tipos de técnicas de ataques de negação de serviço, surgiram diferentes nomenclaturas para os ataques *Slow DoS*. Para melhor entendimento dos processos de análise de tráfego de rede, serão apresentadas algumas definições importantes.

As duas vertentes mais conhecidas desse tipo de ataque são os de alto volume de requisições, semelhantes aos ataques DDoS tradicionais (*High Volume* ou *Flood*) e o baixo volume de requisições (*Low Volume*) com tráfego malicioso enviado à vítima em pequenas porções. Os ataques de baixa taxa classificam-se nas seguintes modalidades: Ataques *Low Rate*, *Slow DoS* e *One Shot* [10].

A maneira pela qual o cliente malicioso atua, ocorre basicamente em três variações: ataques de cabeçalho (*Slowloris* ou *Slow Headers*), ataques de conteúdo (*Slow Body* ou *R.U.D.Y.*), sendo esses também conhecidos como ataques *Slow Send* e o ataque *Slow Read* [10].

Segundo [10], o *Slow Send* é um ataque com foco no cabeçalho ou corpo da requisição (às vezes chamadas como ataques lentos de cabeçalhos ou ataques lentos de conteúdo (*body*)). O ataque lento de cabeçalho pode ser executado com uma variação popular implementada pela ferramenta *Slowloris* [11] que oferece solicitações HTTP parciais (apenas com métodos GET ou POST) enviadas em intervalos regulares para manter a conexão; e com uma implementação mais geral que permite o uso de vários métodos que podem ser encontrados em *Slowhttptest* [12]. O ataque lento de conteúdo começa com um cabeçalho de mensagem HTTP legítimo e, em seguida, continua enviando uma carga HTTP GET ou POST a um ritmo lento (por exemplo, 1byte/1 min) [13, 14].

Com o *Slow Read*, o atacante envia as mensagens de requisição legítimas e lê as respostas do servidor lentamente. Segundo [15], o ataque explora o controle de fluxo do TCP, ou seja, o atacante envia um pedido legítimo após o *3-way handshake* e, depois o atacante anuncia o tamanho da janela menor que o de costume, fazendo reduzir a operação de resposta HTTP. Assim, o servidor envia dados lentamente para o cliente, mantendo seus sockets abertos. A fim de verificar seu tamanho de janela de recepção o servidor continua investigando o cliente, enquanto o cliente sempre o adverte sobre o pequeno tamanho de janela, diminuindo assim a taxa de transferência. Quanto maior o tamanho do arquivo, mais tempo levará para concluir essas conexões.

2.3 Estado da Arte da Detecção dos Ataques Slow DoS

Com a evolução das novas ferramentas de ataque DDoS, novos mecanismos de defesa têm sido propostos. Para que a arquitetura de defesa seja eficiente, o mecanismo de defesa deverá atuar em função do local onde o atacante executa o seu ataque [4], ou seja, onde o ataque ocorrer o mecanismo de defesa deverá atuar.

Segundo [16] os métodos de detecção de anomalia classificam-se em: *machine learning*, *Data Mining*, Inteligência Artificial, métodos estatísticos e baseados em classificadores. A principal vantagem desses métodos é o aprendizado do comportamento esperado mediante observações sem conhecimento prévio das atividades normais do sistema alvo [5], como por exemplo demanda de serviço entre outras.

Na literatura, os trabalhos sobre SDN (*Software-Defined Networking*) mostram-se promissores mediante um conceito de gerenciamento de rede, que pode oferecer defesa eficaz contra ataques DDoS [17]. Segundo [18], é possível utilizar o mecanismo de defesa SDN contra ataques volumétricos à camada de aplicação, porque ataques de inundação, como o HTTP POST, têm o mesmo impacto nos servidores. Este autor ressalta o uso da arquitetura SDN somente para os ataques por inundação.

Alguns trabalhos não tratam dos ataques *Slow DoS* com maior amplitude tratando apenas determinados tipos desse ataque [14, 17, 19]. Nesse sentido, o presente trabalho destina-se à detecção das anomalias referentes a todos os tipos de ataques *Slow DoS*: ataques de cabeçalho, ataques de conteúdo e ataques *Slow Read*. Em [17], é proposto o SDN-*Assisted Slow* HTTP DDoS *Attack* Defense *Method* para detectar ape-

nas os ataques *Slowloris* e *Slow* HTTP POST. Neste método, o servidor solicita ao controlador SDN, a verificação do tráfego suspeito. Como por exemplo, quando um servidor *Web* recebe uma requisição HTTP incompleta em uma situação onde o número de conexões abertas no servidor *Web* excede o limiar pré-determinado de conexões concorrentes sendo processadas, o tráfego é considerado tráfego de ataque [17]. Foi utilizado o simulador NS-3.

Outra abordagem que avaliou a detecção para apenas um tipo de ataque *Slow DoS* foi proposta por [14], que utilizou aprendizado de máquina para a determinação das *features* ou atributos mais importantes para a detecção do ataque de conteúdo denominado R.U.D.Y. ("aRe yoU Dead Yet?"), sendo este o nome dado à implementação do ataque. Foi utilizado um comitê de 10 métodos para selecionar as *features* mais importantes para a classe rotulada, dentre eles, a estatística Kolmogorov-Smirnov, Curva ROC, Qui-Quadrado, ganho de informação entre outras. As *features* selecionadas (sete *features*) foram submetidas a três classificadores (KNN, C4.5N e C4.5D) e um outro conjunto de dados com o tamanho de *features* original também foi submetido ao mesmo grupo de classificadores. Não houve diferença significativa na aplicação dos métodos para redução de dimensionalidade. O *dataset* utilizado foi o produzido por [20] denominado SANTA *dataset*. Este trabalho demonstrou a preocupação na seleção das *features* mais importantes a serem utilizadas por um classificador. [21] propôs limiares para o número de conexões por IP com base no número de conexões. E, [19] desenvolveu um sistema detector de ataques DDoS contra servidores utilizando um classificador Bayesiano.

Recentemente surgiram mais trabalhos com foco na nuvem [5, 18]. Segundo [22], a tecnologia SDN trouxe novas perspectivas para a mitigação de ataques DDoS na nuvem.

3. Métricas para a detecção Slow DoS

A principal particularidade dos ataques lentos à camada de aplicação é a furtividade. Por exemplo, o ataque *Slowloris* caracteriza-se por picos de tráfego em intervalos regulares [23]. Tais picos de tráfego podem passar despercebidos, pois assemelham-se ao tráfego normal. Esta peculiaridade, potencializa o ataque, fazendo-o muito difícil de ser detectado se comparado a um ataque tradicional de inundação, onde os picos de tráfego são percebidos significativamente.

Independentemente da quantidade de máquinas necessárias para empreender um ataque, [10] afirma que apenas uma máquina não surte o efeito desejado para o ataque *Slow Send*.

Dentre as diversas técnicas para a mitigação de ataques lentos à camada de aplicação pode-se destacar: limitação de conexões que um servidor pode manipular ou uma limitação de várias conexões simultâneas que um cliente pode ter ao mesmo tempo [24].

Uma grande quantidade de endereços IP envolvidos em um ataque não é a principal característica dos *Slow DoS*, diferentemente do que acontece durante os ataques por inundação. Sites de consultas ou notícias podem experimentar esse fenômeno, como por exemplo, em função de divulgação conteúdo ruidoso por parte dos provedores.

Nos ataques *Slow DoS*, é difícil prever os pontos no tempo em que ocorrerão picos de uma demanda imprevisível, que surge aleatoriamente em certos momentos e consome toda a capacidade do sistema [25].

Por isso, é importante que o sistema de detecção possa estabelecer ou buscar outros padrões característicos de anomalias com base em analogias históricas, para oferecimento de avaliações mais acuradas do tráfego de rede.

Tratando-se especificamente dos ataques lentos ao protocolo HTTP 1.1, pode-se observar características, ou padrões bem definidos, nos traces maliciosos [10], conforme lista a seguir:

1. Súbito aumento no número de conexões.
2. Cabeçalhos das requisições HTTP incompletos.
3. Manipulação do Corpo ou conteúdo das requisições HTTP.
4. Leitura extremamente lenta das respostas do servidor atacado.

As métricas propostas neste artigo buscam detectar tais características mediante análise do comportamento dos atributos do tráfego de rede.

3.1 Flags

No estabelecimento das conexões, durante o 3-*way handshake* do protocolo TCP, estão envolvidos os atributos das *Flags* que orientam o estágio das conexões, dentre elas, as *flags* SYN e ACK, onde o servidor denota estar em contato com o cliente durante o estabelecimento da conexão e as *flags* FIN e ACK ao final da conexão.

Refletindo sobre a primeira questão da lista das principais características de ataques lentos, no início desta seção, a métrica *Flags* foi projetada para investigar ocorrências das *flags* SYN e ACK e das flags FIN e ACK das conexões HTTP 1.1, presentes durante um período de tempo.

Embora tenham sido introduzidos novos recursos ao protocolo HTTP 2.0, tais como mecanismo de controle de fluxo e algoritmo de compactação de cabeçalho, para mitigação dos transtornos do HTTP 1.1, segundo [26], tais recursos também podem ser explorados por atacantes para empreender um ataque DDoS.

O protocolo HTTP 1.1 da camada de aplicação utiliza o TCP como protocolo da camada de transporte para transmissões confiáveis. No entanto, segundo [27] o protocolo HTTP 2.0, apresentou mais deficiências do que seu predecessor no que tange aos ataques *Slow DoS*.

O escopo desse trabalho está relacionado ao protocolo HTTP 1.1 e aos ataques DDoS com baixo volume de requisições *Slow DoS*.

A estratégia de detecção para desenvolvimento da métrica *Flags* foi a seguinte:

- Se no período considerado, observa-se o súbito aumento de *flags* SYN e ACK, pode-se, no mínimo, considerar um comportamento anormal. A anormalidade pode ser confirmada caso haja, também, a ausência das *flags* FIN e ACK no período considerado dado que existem mais conexões abertas que fechadas. Para verificar esse tipo de anomalia, estabeleceu-se a métrica *Flags* para auxiliar a análise do comportamento da rede, no que tange ao súbito aumento do número de conexões por período de tempo, conforme a Equação 1.

$$Flags = \sum_{i=t_{inicial}}^{t_{final}} SYNACK_i - \sum_{i=t_{inicial}}^{t_{final}} FINACK_i \qquad (1)$$

A Equação 1 consiste na diferença entre o somatório das ocorrências das flags SYN ACK e FIN ACK por janela de tempo. Onde i representa o instante t (pacote) por janela de

tempo.

Para confirmação da condição de anormalidade, calcula-se a média e respectivos desvios padrão das janelas de tempo mais recentes, para a variável *Flags*, conforme Equações 2 e 3.

$$\overline{Flags} = \frac{\sum_{i=i-1}^{i-(n-1)} Flags_i}{n} \quad (2)$$

$$S_{Flags} = \frac{\sum_{i=i-1}^{i-(n-1)} \left(Flags - \overline{Flags}\right)^2}{n-1} \quad (3)$$

Para a detecção de anomalia, duas condições deverão ser satisfeitas. A primeira, compara a dispersão da variável *Sum_SYNACK* dada pela Equação 4.

$$Sum\_SYNACK = \sum_{i=t_{inicial}}^{t_{final}} SYNACK_i \quad (4)$$

A segunda compara o valor da diferença das *flags* da janela atual com relação às janelas mais recentes.

Na estatística, em análise de séries temporais, os modelos atribuem peso maior aos dados mais recentes. Então, para a confirmação do estado anormal, ambas as condições citadas deverão atender ao intervalo de confiança estimado pela desigualdade Tchebycheff.

O poder da métrica Flags concentra-se na investigação das anomalias comuns aos ataques *Slow DoS*: *Slow Headers*, *Slowloris*, *Slowbody*, *R.U.D.Y.* e *Slow Read*.

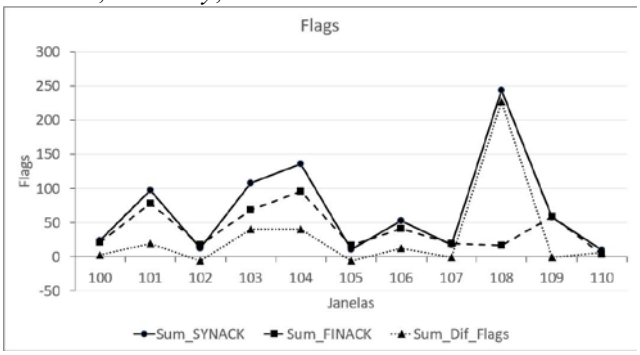

Fig 1 - Gráfico da distribuição dos valores das métricas Flags por janela de tempo de 30 segundos. Fonte: com base nos dados da Akamai [2].

Pode-se observar na **Figura 1**, que as métricas Flags detectam o ataque ocorrido na janela 108, pois os valores das mesmas, com relação ao histórico do tráfego, encontram-se fora do limite considerado pela desigualdade Tchebycheff.

3.2 Omnia

Segundo [24], a avaliação da capacidade de resposta do servidor é representada pela percentagem de requisições de clientes HTTP deixadas incompletas durante determinado período. Foram comparadas as taxas de responsividade entre diversos servidores Web a fim de medir a resiliência de cada um dos servidores aos ataques *Low Rate*, *Slow Send* e *Slow Read*.

A métrica *Omnia* avalia a quantidade de requisições efetivamente incompletas. Defini-se por requisição incompleta, as requisições HTTP que não apresentam o CRLF (*Carriage Return Line Feed*) ao final do cabeçalho das requisições com o método GET.

Como estratégia de detecção, para a definição da métrica Omnia, considerou-se que uma requisição gerada com um formulário não utiliza necessariamente o método POST [24]. Ao contrário, formulários HTML costumam empregar o método GET e incluem os dados informados (nos campos do formulário) no endereço requisitado. Neste caso, é possível observar os parâmetros próprios desse tipo de solicitação, tais como "?" e "\&". Não foram encontrados esses parâmetros nas solicitações com o método GET, fato que poderia configurar um ataque da categoria *Slow Body*.

A métrica *Omnia* destina-se à investigação dos ataques com essa característica, tais como o *Slow Headers* e *Slow Body*, de acordo com a literatura atual. Esta métrica oferece o cálculo da taxa de requisições incompletas por janela de tempo sendo representa pela Equação 5.

$$Omnia = \frac{\sum_{i=t_i}^{t_f} \widetilde{R}_i}{t} \quad (5)$$

Onde *i* é o i-ésimo pacote de dados, é o tempo inicial do intervalo *t*, t_f é o tempo final do intervalo *t*, $\widetilde{R}$ é definida por requisição incompleta do i-ésimo pacote e *t* é o intervalo da janela de tempo estática.

3.3 RTTR

O RTT (*Round Trip Time*) é o tempo decorrido entre o envio de um pacote e o recebimento do respectivo reconhecimento [28]. Os atrasos percebidos por este atributo da camada de transporte (RTT) podem ser causados pelo envio dos cabeçalhos das requisições HTTP incompletos ao servidor e pela manipulação do conteúdo dos atributos do tráfego de rede, tais como: janela de recepção, *content-length*, entre outros, durante o envio dessas mensagens ao servidor e vice-versa. Pode-se exemplificar tal estratégia de detecção da seguinte forma:

- Se um atacante envia diversos pacotes incompletos, o servidor irá respondê-lo, porque este entende que a conexão pode estar lenta. Neste caso, o cliente mal-intencionado demorará mais tempo que o normal para enviar o reconhecimento (ACK) às mensagens do servidor, pois o atacante realmente deseja manter a atenção do servidor pelo maior tempo possível fazendo-o consumir recursos importantes, como a memória.

Em outra situação, caso a janela de recepção seja manipulada, o servidor, por não ter um limite para a duração das conexões com o cliente, enviará o solicitado em pacotes de tamanho mínimo. Assim, as mensagens de resposta terão tamanho extremamente reduzido e com isso o cliente ilegítimo vai conseguindo manter a atenção do servidor, pois demora a dar o reconhecimento de tudo o que foi enviado pelo servidor. A concretização do RTT lento ocorrerá no servidor. Assim, as mensagens de resposta terão tamanho extremamente reduzido e, dessa forma, o objetivo do atacante, que é a manutenção da conexão com o servidor pelo maior período possível, é atingido.

Portanto, a métrica *RTTR* (RTT Real) buscará por esses tipos de anormalidade. Para averiguação dos comportamentos do RTT fora do padrão, assume-se que o servidor será a

fonte e o cliente o destino, pois o valor RTT se concretizará no servidor.

A métrica *RTTR* foi projetada em função do RTT Real e não o valor de RTT suavizado, que é utilizado para gerenciamento da temporização de transmissão. Para evitar que o remetente sature o *buffer* do destinatário rapidamente, o protocolo TCP utiliza o controle de fluxo, fazendo com que o remetente mantenha a variável janela de recepção sob qualquer suspeita [28].

Calcula-se a métrica *RTTR* pelo somatório dos valores de RTT por janela de tempo de acordo com Equação 6.

$$RTTR = \sum_{i=t_{inicial}}^{t_{final}} RTT_i \qquad (6)$$

O poder da métrica RTTR concentra-se na investigação das características dos ataques *Slow DoS* (*Slow Headers, Slowloris, Slowbody* e *Slow Read*).

4. Metodologia para Sistema de Detecção

Os sistemas de detecção baseiam-se na assinatura dos ataques ou buscam padrões de comportamentos anômalos no tráfego de rede, principalmente, para realimentar os métodos empregados na mitigação dos problemas ocasionados pelos ataques à rede.

Outra classificação importante em um sistema de detecção é o local onde o sistema deverá atuar: se próximo à fonte, próximo à vítima ou em roteadores intermediários. Segundo a taxonomia proposta por [5], o tráfego é mais disperso próximo à fonte e, portanto, torna-se muito difícil encontrar pacotes maliciosos, pois parecem pacotes inofensivos.

A metodologia proposta por [29] detecta anomalias causadas pelos ataques Slow DoS, com base nas métricas concebidas no presente trabalho, derivadas das estratégias de detecção descritas na seção anterior. O mecanismo desse sistema baseia-se na detecção de anomalias com base no comportamento dos atributos de rede e o local onde o mecanismo atua é próximo à vítima.

Os ataques Slow DoS aproveitam-se da leniência de alguns servidores Web [30, 31] com relação aos atrasos na rede de computadores. Por este motivo, a detecção deve atentar para as mudanças de comportamento na rede.

Utiliza-se a metodologia de detecção de ataques Slow DoS [29], conforme **Figura 2**. Essas métricas fornecerão o cálculo dos atributos de rede necessários para o previsor Média Móvel Simples (MMS) realizar a previsão do instante seguinte, com base na avaliação do histórico de rede. Foi utilizado o período de 30 janelas de tempo para o previsor MMS.

A metodologia tem início com o recebimento do tráfego de rede, que sofrerá um processo de estimação das janelas de tempo. Este processo consiste em dividir o atributo do tráfego, *time*, em intervalos fixos, ou seja, as janelas de tempo estáticas de 30 segundos. Em seguida os pacotes de uma determinada janela são encaminhados à fase de processamento pelas métricas. Adotou-se a janela de tempo de 30 segundos, com base em experimentos realizados em outros trabalhos de detecção [10,21].

As métricas compõem um comitê de votação com peso uniforme e processam seus resultados com execução paralela. Assim, todas as métricas participam do processo de verificação de anomalia e fornecem o seu resultado ao decisor, após o processo de classificação, onde ocorrerá a rotulação do tráfego

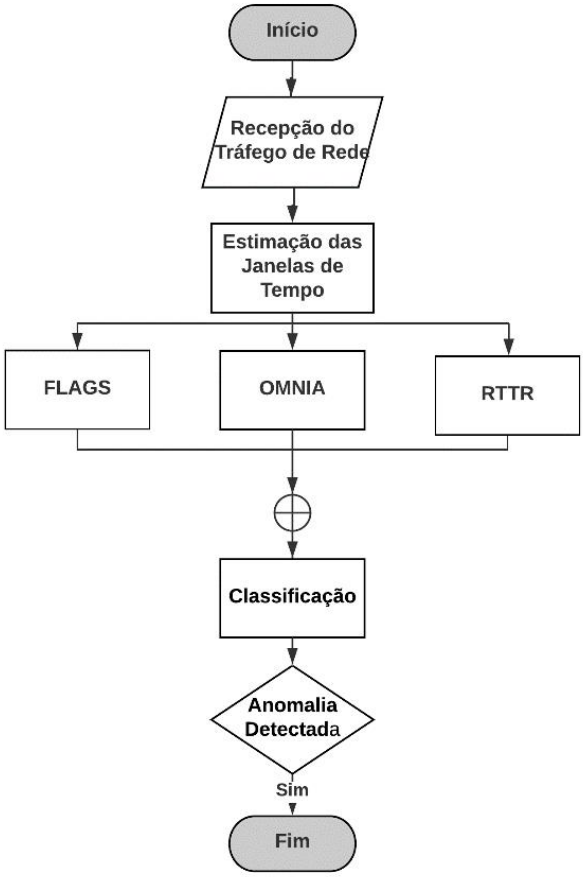

Figura 2 - Arquitetura para detecção de anomalias para ataques Slow DoS. Fonte: [29].

em normal ou anormal, segundo a desigualdade Tchebycheff.

Assim, dentro do espaço hipótese, dado que uma das métricas detectou anormalidade, o classificador rotula o período analisado como anormal.

A classificação do tráfego será determinada pela desigualdade Tchebycheff, que computará o resultado da média móvel analisando-o segundo o intervalo estipulado. Segundo [32], tais desigualdades são úteis para o trabalho com distribuições, cuja forma da função de distribuição é desconhecida. Por esse motivo, dado que não se conhece a distribuição das variáveis de tráfego envolvidas nesse trabalho, tais como aquelas calculadas pela métrica Flags e pela métrica RTTR, emprega-se a desigualdade Tchebycheff, dada pela Equação 7.

$$P(X \geq \lambda\sigma + \mu \ \ ou \ \ X \leq \mu - \lambda\sigma) \leq \frac{1}{\lambda^2} \qquad (7)$$

Pela desigualdade Tchebycheff, estima-se que pelo menos 89% do tráfego normal (sem ataque) ficará dentro dos limiares de 3S a partir da média . Há uma chance de 89% do intervalo conter a média da população e, consequentemente, o que estiver fora dos limites será considerado anormal. Adotou-se o valor de igual à 3, com base no controle estatístico de qualidade, cuja convenção dos limites de controle dos pontos do processo posiciona-se no mesmo limiar a partir da média.

4.1 Análise da Metodologia

As métricas foram implementadas em um Workflow Científico para análise de big data chamado Alteryx e testa-

das no CIC (Canadian Institute for Cybersecurity) DoS Dataset da Universidade de New Brunswick no Canadá.

4.1.1 O Dataset

Há grande dificuldade na obtenção de *dataset* privado para a pesquisa de detecção de ataques, pois há grande receio de danificar ativos importantes durante a aplicação do experimento.

Sobre o tema *Slow* DoS, existem poucos *datasets* que possam ser explorados. O CIC DoS *Dataset* (*Application-Layer*) foi escolhido porque o CIC [33] possui alguns dos *datasets benchmarks* mais utilizados por universidades, indústria privada e pesquisadores independentes para detecção de anomalias. Segundo CIC, as características do dataset são as seguintes: Apache Linux v.2.2.22; Traces do ISCX-IDS Dataset (livre de ataques); Produção de 4 tipos de ataques à camada de aplicação com diferentes ferramentas; 8 traces de ataque DoS à camada de aplicação com 24 horas de tráfego de rede totalizando 4.6 GB; e Ataques direcionados aos 10 Servidores Web do ISCX-IDS Dataset [33].

4.1.2 Resultados Obtidos

Os resultados foram analisados segundo a matriz confusão da **Figura 3**.

Matriz Confusão		Predita +	Predita -
Classe Verdadeira	+	VP	FN
	-	FP	VN

Fig 3 – Matriz Confusão com base em [34].

A matriz confusão traz informações entre a classe verdadeira e a classe predita. A classe pode ser positiva (+), quando há ataque, e negativa (-), caso contrário. VP é a quantidade de verdadeiros positivos, ou seja, janelas detectadas corretamente com ataque; FP (falsos positivos) são as janelas que foram detectadas com ataque, mas na realidade não contém ataque; VN (verdadeiros negativos) é a quantidade de janelas que não são ataques e também não foram detectados quaisquer ataques e FN é a quantidade de janelas que deveriam ter sido detectadas com ataque, mas não foram detectadas. Logo, o somatório de VP, VN, FP e FN corresponde à n (número de janelas de tempo após divisão do tráfego em janelas estáticas).

Assim, a regra comum aplicada para classificação dos resultados decorrentes da execução paralela das métricas, segundo a metodologia proposta, classificou as janelas em verdadeiros positivos ou falsos positivos de acordo com as seguintes regras:

- Se a anomalia foi detectada na janela que coincidia ou pertenciam à janela indicada pelo gabarito da Figura 4, a janela foi rotulada anormal. Portanto esse tráfego seria VP; e
- quando a anomalia foi detectada em janelas imediatamente posteriores à janela indicada pelo gabarito da Figura 4, foi realizada uma apuração do tráfego dentro da janela. Verificou-se que o comportamento suspeito (ex. demandas com alto RTTR, envio de pacotes incompletos, aumento expressivo de conexões) era atribuído ao endereço IP listado como alvo. Conforme gabarito, a anomalia foi classificada como VP e FP, caso contrário.

Janelas Detectadas pelas Métricas F.O.R.				
Ataque	Alvo	Início	Janela	Detecções
Slow Body	75.127.97.72	após 00:53 min	106	108, 109
Slow Read	75.127.97.72	após 01:58 min	236	237, 238, 239, 240
Slow Headers	74.63.40.21	após 02:57 (177 min)	354	371
R.U.D.Y.	75.127.97.72	após 03:08 (188 min)	376	∅
R.U.D.Y.	208.113.162.153	após 03:29 (209 min)	418	419
Slow Headers	67.220.214.50	após 06:00 (360 min)	720	725, 737
Slow Body	69.192.24.88	após 08:13 (493 min)	986	988, 989
Slow Body	97.74.144.108	após 09:03 (543 min)	1086	1088, 1089, 1093
Slow Body	203.73.24.75	após 09:09 (549 min)	1098	1106
R.U.D.Y.	97.74.144.108	após 09:20 (560 min)	1120	∅
Slow Read	74.55.1.4	após 11:02 (662 min)	1324	1325, 1326, 1328
Slow Headers	97.74.104.201	após 11:27 (687 min)	1374	1379, 1380, 1384, 1385, 1391
Slowloris	97.74.144.108	após 15:20 (920 min)	1840	1841, 1842, 1843, 1844
Slow Headers	97.74.144.108	após 15:47 (947 min)	1894	∅
Slowloris	75.127.97.72	após 16:33 (993 min)	1986	1987, 1989, 1990
Slow Headers	75.127.97.72	após 17:13 (1033 min)	2066	2072, 2083
R.U.D.Y.	74.55.1.4	após 20:59 (1259 min)	2518	2518, 2519

Fig 4 – Gabarito e Resultados Obtidos na Implementação das Métricas F.O.R.

Os resultados da implementação das métricas estão dispostos na Figura 5. O processamento das métricas F.O.R., executadas paralelamente resultou em acurácia de 95%, taxa de detecção ou sensibilidade de 71%, taxa de erro na classe positiva de 29% e taxa de erro na classe negativa de 5%.

A precisão alcançada foi de 22%. Já a média harmônica, com peso igual a um, para as medidas de precisão e sensibilidade, foi de 34%.

Matriz Confusão		Classe Predita +	Classe Predita -
Classe Verdadeira	+	VP = 36	FN = 15
	-	FP = 128	VN = 2702

Fig 5 – Resultados de Desempenho do Processamento das Métricas

Segundo [5], nenhum mecanismo de defesa pode alcançar 100% de detecção dos pacotes de ataque, mas deve alcançar taxas de verdadeiros positivos (VP) e verdadeiros negativos (VN), com o mínimo possível de falsos positivos (FP) e falsos negativos (FN).

É importante destacar que, caso a métrica se baseie apenas no padrão do tráfego de rede, qualquer *outlier* poderia ser detectado como ataque. Desse fato, observa-se que, dependendo do ambiente de rede, que pode ser bastante instável, este pode contribuir para o aumento de falsos positivos. Apesar do planejamento das métricas preocupar-se com essa possibilidade, o fato é que os ambientes de rede são instáveis por natureza e a ocorrência nula de falsos positivos, em tese, não faz parte da realidade.

Com essa abordagem, o processamento das métricas F.O.R. obteve uma taxa de erro na classe positiva alta. Esta medida de desempenho demonstra que o algoritmo deixou de classificar ataques de fato, com erro de 29% sobre o total de ataques. Esse resultado trouxe uma quantidade alta de FN, o que afetou o cálculo da taxa de detecção (sensibilidade ou abrangência).

A acurácia obteve ótimo resultado com 95% de acertos no total da classificação do trafego sob análise.

A anomalia referente ao ataque *R.U.D.Y.* foi a mais difícil de ser detectada, conforme Tabela 2. Este fato também foi citado por [10].

Tab 2 - Ocorrências Detectadas – Métricas F.O.R.

Tipo Ataque	Nome Ataque	# Ocorrências Ataques	#Detecções	# FN
Cabeçalho	Slowloris	2	2	0
	Slow Headers	*5*	4	1
Conteúdo	*Slow Body*	4	4	0
	R.U.D.Y	4	2	2
Leitura Lenta	*Slow Read*	2	2	0

Fonte: Com base nos dados do *CIC DoS Dataset* [33].

5. Conclusões

Recentemente, como consequência dos esforços de mitigação aos ataques DoS tradicionais, os ataques *Slow DoS* surgem como ameaça à garantia da disponibilização de serviços na web. Esse ataque é considerado preocupante devido a furtividade do seu modo de operação, que lentamente e sem alarde, desabilita a vítima.

Uma das vantagens da metodologia proposta é que não há um *threshold* previamente definido para os atributos do tráfego de rede. A detecção de anomalias na distribuição de pacotes de uma janela é feita mediante a observação dos valores prévios ou analogia histórica do próprio tráfego de rede.

A contribuição desse trabalho foi a proposição de estratégias de detecção e métricas, sem complicações de cálculo, que tem papel fundamental na detecção desses ataques, pois a detecção de ataques de negação de serviço possui vários desafios dentre os quais pode-se citar a obtenção de mecanismos de detecção mais rápidos, porque esquemas de detecção consomem recursos da vítima e isso afeta a capacidade de acurácia.

Como trabalho futuro, pretende-se estender as discussões deste artigo propondo uma metodologia que possa não apenas detectar anomalias, mas também efetuar a identificação do ataque *Slow DoS*.

Agradecimentos

Os autores gostariam de agradecer ao Sr. Arash Habibi Lashkari pela disponibilização do CIC DoS Dataset, ao Sr. Hugo Gonzalez pelas respostas referentes aos ataques Slow DoS e também à equipe do Grupo Alteryx Analytics, responsável pela disponibilização da ferramenta Alteryx para as análises de big data realizadas nesse trabalho de pesquisa.

Referências Realizadas

[1] Khanna, S., Venkatesh, S. S., Fatemieh, O., Khan, F., & Gunter, C. A. (2012). Adaptive selective verification: An efficient adaptive countermeasure to thwart dos attacks. IEEE/ACM Transactions on Networking (TON), 20(3), 715-728.

[2] Akamai Technologies. Q1 report 2017 akamai state of the internet. Q1 Report 2017. Akamai State of the Internet, v. Q1 2017, 2017. Disponível em:<https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/q1-2017-state-of-the-internet-connectivity-report.pdf/>. Acesso em: 21 jun. de 2017.

[3] J. Mirkovic, P. Reiher, A taxonomy of DDoS attack and DDoS defense mechanisms. ACM SIGCOMM Computer Communication Review, v. 34, n. 2, p. 39–53,2004.

[4] Bhuyan, M. H., Kashyap, H. J., Bhattacharyya, D. K., & Kalita, J. K. (2013). Detecting distributed denial of service attacks: methods, tools and future directions. The Computer Journal, 57(4), 537-556.

[5] Gupta, B. B., Badve, O. P. (2017). Taxonomy of DoS and DDoS attacks and desirable defense mechanism in a cloud computing environment. Neural Computing and Applications, 28(12), 3655-3682.

[6] CGI.BR. Cartilha de segurança para internet. CGI. br (Comitê Gestor da Internet no Brasil), v. 2012, 2012. Disponível em: <https://www.cgi.br/>. Acesso em: nov.de 2016

[7] T. Peng, G. Leckie, K. Ramamohanarao, Survey of network-based defense mechanisms countering the dos and DDoS problems. ACM Computing Surveys.

[8] Y. Ohsita, S. Ata, M. Murata, Detecting distributed denial-of-service attacks by analyzing tcp syn packets statistically. IEICE transactions on communications, v. 89, n. 10, p. 2868–2877, 2006.

[9] S. S Silva, R. M Silva, R. C Pinto, R. M Salles,. (2013). Botnets: A survey. Computer Networks, 57(2), 378-403.

[10] H. Jazi, H. Gonzalez, N. Stakhanova, A. A. Ghorbani, Detecting http-based application layer dos attacks on web servers in the presence of sampling. Computer Networks, v. 121, p. 25–36, 2017.

[11] Rsnake, J. K.; LEE, R. Slowloris http DoS. URL: http://ha.ckers.org/slowloris/ (June 2009), v.1, p. s/n, 2009. Disponível em: <http://ha.ckers.org/slowloris/(June 2009)>. Acesso em: 21 jan. de 2018.

[12] Shekyan, S. Slowhttptest-application layer DoS attack simulator. Available: http://code. google. com/p/slowhttptest, v. s/n, p. 1, 2013. Disponível em: <http://code.google.com/p/slowhttptest>. Acesso em: 21 jan. de 2018.

[13] Chee, W. O.; Brennan, T. HTTP POST slide show. In: [S.N.], [s.n.]., 2010. Anais eletrônicos. [S.l.: s.n.], 2010, p. 1. Disponível em: <https://www.owasp.org/index.php/OWASPHTTPPostTool>. Acesso em: 21 jan. de 2018.

[14] M. M. Najafabadi, T. M. Khoshgoftaar, A. Napolitano, C. Wheelus, Rudy attack: Detection at the network level and its important features. In: Flais Conference, 29., Florida Artifical Intelligence, 4., Florida.Proceedings... Florida: SciTePress, 2016, p. 288–293.

[15] Park, J.; Iwai, K.; Tanaka, H.; Kurokawa, T. Analysis of slow read DoS attack. In: Information Theory and its Applications (ISITA), 2014 International Symposium ON, [s.n]., 2014. Anais eletrônicos. [S.l.: s.n.], 2014, p.9560 – 64. Disponível em: https://ieeexplore.ieee.org/abstract/document/6979803/>. Acesso em: 14 set. de 2017.

[16] Osanaiye, O., Choo, K. K. R., Dlodlo, M. (2016). Distributed denial of service (DDoS) resilience in cloud: review and conceptual cloud DDoS mitigation framework. Journal of Network and Computer Applications, 67, 147-165.

[17] K. Hong, Y. Kim, H. Choi, J. Park, SDN-Assisted Slow HTTP DDoS Attack Defense Method, 2017. IEEE Communications Letters.

[18] Yan, Q., Yu, F. R., Gong, Q., & Li, J. (2016). Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and challenges. IEEE Communications Surveys & Tutorials, 18(1), 602-622.

[19] Katkar, V., Zinjade, A., Dalvi, S., Bafna, T., & Mahajan, R. (2015, February). Detection of DoS/DDoS Attack against HTTP Servers Using Naive Bayesian. In Computing Communication Control and Automation (ICCUBEA), 2015 International Conference on (pp. 280-285). IEEE.

[20] Wheelus, C., Khoshgoftaar, T. M., Zuech, R., & Najafabadi, M. M. (2014, November). A Session Based Approach for Aggregating Network Traffic Data--The SANTA Dataset. In Bioinformatics and Bioengineering (BIBE), 2014 IEEE International Conference on (pp. 369-378). IEEE.

[21] T. Hirakawa, K. Ogura, B.B. Bista, T. Takata, T., A Defense Method against Distributed Slow HTTP DoS Attack. In Network-Based Information Systems (NBiS), 2016, September 19th International Conference on (pp. 152-158). IEEE.

[22] Shameli-Sendi, A., Pourzandi, M., Fekih-Ahmed, M., & Cheriet, M. (2015). Taxonomy of distributed denial of service mitigation approaches for cloud computing. Journal of Network and Computer Applications, 58, 165-179.

[23] Aqil, A., Atya, A. O., Jaeger, T., Krishnamurthy, S. V., Levitt, K., McDaniel, P. D., & Swami, A. (2015, October). Detection of stealthy tcp-based dos attacks. In Military Communications Conference, MILCOM 2015-2015 IEEE (pp. 348-353). IEEE.

[24] Gonzalez, H., Gosselin-Lavigne, M. A., Stakhanova, N., & Ghorbani, A. A. (2014). The Impact of Application-Layer Denial-of-Service Attacks. Case Studies in Secure Computing: Achievements and Trends, 261.

[25] Menascé, D. A., & Almeida, V. A. (2003). Planejamento de Ca-

pacidade para Serviços na Web: Métricas, modelos e métodos. Rio de Janeiro: Campus.
[26] Jiang, M., Wang, C., Luo, X., Miu, M., & Chen, T. (2017, June). Characterizing the impacts of application layer DDoS attacks. In Web Services (ICWS), 2017 IEEE International Conference on (pp. 500-507). IEEE.
[27] Tripathi, N., & Hubballi, N. (2018). Slow rate denial of service attacks against HTTP/2 and detection. Computers & Security, 72, 255-272.
[28] Kurose, J. F., & Ross, K. W. (2014). Redes de Computadores e a Internet. Uma Abordagem Top Down.
[29] Sennejunker, D. S. T-Slow: Algoritmo para a Detecção de Ataques Slow DoS. 2018. 105p. Dissertação (Mestrado em Sistemas e Computação) – Instituto Militar de Engenharia, Rio de Janeiro, 2018.
[30] Damon, E., Dale, J., Laron, E., Mache, J., Land, N., & Weiss, R. (2012, October). Hands-on denial of service lab exercises using slowloris and rudy. In proceedings of the 2012 information security curriculum development conference (pp. 21-29). ACM.
[31] Helalat, S. M. (2017). An Investigation of the Impact of the Slow HTTP DOS and DDOS attacks on the Cloud environment.
[32] Savage, I. R. (1961). Probability inequalities of the Tchebycheff type. Journal of Research of the National Bureau of Standards-B. Mathematics and Mathematical Physics B, 65(3), 211-222.
[33] CIC-DOS.University of New Brunswick-Canadian Institute for Cybersecurity (CIC)-DoS Dataset. [S.l.: s.n.], 2017. (Relatório Técnico). Acesso http://www.unb.ca/cic/datasets/index.html no dia 04/03/2018.
[34] K. Faceli, A. C. Lorena, J. Gama, A. C. Carvalho. Inteligência Artificial: Uma abordagem de aprendizagem de máquina. [S.l.]: Grupo Gen-LTC, 2015