Detecção de Ataque DDoS Flash Crowd Realizando Análise Comportamental de Solicitações Web

Samuel Lautert Jardim[1] **Raul Ceretta Nunes**[1,2] **Marcelo Colomé**[1]

[1]Programa de Pós-Graduação em Computação [2]Departamento de Computação Aplicada

Universidade Federal de Santa Maria, Av. Roraima, 1000, Camobi, Santa Maria, RS

samuel@saopedronet.com.br, ceretta@inf.ufsm.br,
marcelocolome@inf.ufsm.br

Abstract. *A Distributed Denial of Service (DDoS) attack is a threat to Internet good functioning. To make undetectable attacks, attackers (botmasters) are exploring the application layer as alternative to get similarity with benign network traffic. A DDoS Flash Crowd attack generates similar traffic to Flash Crowds events (outbreak of unexpected visits). Attack detection tools need to differentiate DDoS attack flows from flash crowd flows. This paper proposes a DDoS detection method based on the analysis of user request interactivity. The method differs a human user from a bot (malware) modeling the interactive behavior by analyzing the number of requests and the time between them (interactivity rate). The experiments demonstrate the detection effectiveness of the method and that the applicability of analysis of expected interactivity pattern as a detection mechanism.*

Resumo. *Um Ataques de Negação de Serviço Distribuído (DDoS) é uma ameaça para o bom funcionamento da Internet. Ataques na camada de aplicação, como DDoS Flash Crowd, vêm se consolidando como alternativa para botmasters tornarem seus ataques ainda mais indetectáveis, dado a similaridade com tráfego de rede benigno do tipo Flash Crowd (surto de visitas inesperadas). Ferramentas de detecção de ataques necessitam diferenciar um tráfego flash crowd de um tráfego com ataque DDoS. Este trabalho propõe um método de detecção baseado na observação do padrão de interatividade nas solicitações dos usuários. O método difere um usuário humano de um bot (programa malicioso) modelando o comportamento através do número de solicitações e do tempo entre elas (taxa de interatividade). Os experimentos demonstram a eficácia do método na detecção, comprovando que o padrão de interatividade esperado pode ser aplicado como mecanismo de detecção.*

1. Introdução

Inúmeras formas de ataques são conduzidas por grupos maliciosos para tornar falho ou inoperante um *website* ou serviço específico, podendo afetar uma cadeia de pessoas que usufruem do serviço de maneira direta ou indireta. Neste sentido, são utilizados muitos

métodos de ataques aos computadores servidores, onde se destaca o ataque de Negação de Serviço Distribuído (DDoS) [Yu et al. 2012][Dantas 2013][Dhingra e Sachdeva 2014].

Para a realização de ataques DDoS são utilizadas redes denominadas *botnets,* formadas por computadores distribuídos geograficamente e comprometidos com softwares maliciosos (*bots*) comandados por criminosos (*botmasters*), sem o consentimento e conhecimento dos proprietários dos computadores (hosts infectados). Segundo [Feily et al. 2009] o ataque possui dois métodos principais. O primeiro se aproveita de uma vulnerabilidade e envia pacotes mal formados ludibriando um protocolo ou aplicação. Já o segundo tem o objetivo de interromper a conectividade do usuário utilizando o esgotamento da largura de banda, capacidade dos roteadores ou recursos de rede, sendo conhecido como ataque por inundação de rede; ou também, esgotamento dos recursos do servidor, este conhecido como ataque de inundação de aplicação. Com o objetivo de dificultar a detecção de *botnets*, os *botmasters* utilizam-se de técnicas anti forenses para disfarçar peculiaridades que possam diferenciar traços de um ataque dos dados legítimos. Ofuscação de código, criptografia, comunicação *peer-to-peer* e imitação de eventos *Flash Crowd* são exemplos destas técnicas [Yu et al. 2012].

Um evento *Flash Crowd* corresponde a um pico de acesso legítimo a dado *website*, tal como pode ocorrer em sites de notícias durante grandes eventos. Este tipo de evento costuma gerar degradação de desempenho ou indisponibilidade com consequente insatisfação de usuários. Um ataque DDoS *Flash Crowd* é um tipo de ataque que procura imitar um comportamento *Flash Crowd*, ou seja, procura tornar o serviço indisponível a partir de rajadas de tráfego legítimo [Xie e Yu 2009][Ke et al. 2009] [Prasad et al. 2013]. Tanto *Flash Crowd* quanto DDoS *Flash Crowd* podem ser detectados analisando as características estatísticas do tráfego [Thapngam et al. 2011]. O desafio é adotar uma abordagem capaz de realizar a diferenciação adequada entre o comportamento de um *Flash Crowd* (gerado por humanos) e o de um ataque (gerado por *bots*) [Xie e Yu 2009][Yu et al. 2012] [Prasad et al. 2013].

O cálculo da correlação de *Pearson* aplicado às séries temporais do número de solicitações HTTP [Xu et al. 2012] tem sido o método mais aplicado para diferir o comportamento humano de um *bot* no caso de ataques DDoS *Flash Crowd*. Este método assume que os traços gerados pelos *bots* possuem entre sí uma correlação maior que os traços gerados por humanos [Thapngam et al. 2011]. Entretanto, cada página web possui um número de solicitações secundárias aleatórias, podendo fazer com que este método apresente muitos falsos positivos. Para minimizar o problema, Thapngam et al. (2011) propuseram um método no qual calculam a correlação de *Pearson* da taxa de chegada de pacotes dos *hosts* ao longo do tempo. Porém, na medida em que um servidor web estiver trabalhando com alta carga o comportamento da taxa de pacotes dos *hosts* pode se assemelhar bastante em função do *throughput* minimizado para cada *host*, o que pode comprometer o método. De maneira semelhante, mas considerando não mais a correlação de Person, Yu et al. (2012) perceberam a existência de maior similaridade nos fluxos de rede dos ataques DDoS *Flash Crowd* do que nos fluxos dos usuários humanos, possibilitando a detecção por análise da similaridade. No entanto, como reconhecido pelos próprios autores, a similaridade pode ser facilmente ludibriada pelos *botmasters* criando vários grupos de padrões de solicitações no código fonte, terminando com a alta similaridade de um grupo grande de *hosts* e tornando a similaridade entre os *hosts* da rede heterogênea.

O presente trabalho apresenta um método para diferenciação entre humano e *bot* baseado na análise comportamental da interação nas solicitações HTTP(S) de cada *host* cliente. O comportamento é modelado através do número de solicitações e do tempo entre solicitações, resultando na análise da taxa de interatividade. O método de diferenciação contrasta os comportamentos médios esperados para humanos com os de *bots*. O trabalho inicialmente discute a relação entre ataques DDoS na camada de aplicação e eventos *Flash Crowd* (seção 2), bem como o comportamento interativo nas solicitações HTTP(S) (seção 3). O método proposto e os resultados experimentais são apresentados nas seções 4 e 5, respectivamente. Os trabalhos relacionados são apresentados na seção 6 e as conclusões na seção 7.

2. Flash Crowds e Ataque DDoS na Camada de Aplicação

Tradicionalmente, ataques de negação de serviço exploram vulnerabilidades nos protocolos de rede visando o esgotamento de seus recursos. Diferentemente, ataques DDoS na camada de aplicação tem por objetivo esgotar os recursos do servidor (sockets, CPU, memória, banco de dados, largura de banda de I/O) [Ke et al. 2009][Xie e Yu 2009][Walfish et al. 2010]. A motivação para esta mudança de foco dos atacantes decorre da melhor eficácia dos mecanismos de segurança em redes e da crescente evolução da tecnologia, que permitiu crescimento substancial da largura de banda das conexões de internet (*throughput* das interconexões).

Na camada de aplicação o tráfego anormal pode ser classificado em quatro tipos [Saravanan et al. 2013]: (1) *insistência no pedido* - o atacante concentra o ataque em uma página inicial ou uma página momentaneamente muito acessada, gerando maior fluxo no tráfego que converge para um ou mais pontos; (2) *pedido recursivo* - o tráfego é direcionado para várias páginas web (*urls*), fazendo o endereço fonte dos acessos convergir para um grupo de pontos. Porém as metas de tráfego tornam-se dispersas; (3) *repetidas cargas de trabalho* - tráfego que resulta de um grupo de *bots* que envia repetidas solicitações, por exemplo, acesso a uma imagem grande ou operações que exijam grande carga no banco de dados; e (4) *Flash Crowd* - causa um surto de visitas inesperadas, geralmente em um anúncio de novo serviço ou download de software livre, os endereços fontes são dispersos, as solicitações são legítimas e as metas convergem para um ou dois pontos. O tráfego anormal resultante de comportamento *Flash Crowd* é similar ao de um ataque DDoS na camada de aplicação, tanto na maneira que os dados são transmitidos quanto na troca de mensagens de comunicação entre cliente e servidor. Ambos são volumosos, apresentam rajadas e são instáveis, causando interrupções no serviço [Kandula et al. 2005][Ke et al. 2009][Dhingra and Sachdeva 2014].

Diferenças entre fluxos *Flash Crowd* e ataques DDoS no nível de aplicação (que imitam *flash crowds*) aparecem na análise comportamental dos acessos benignos e malignos, como por exemplo, na distribuição da origem dos endereços IPs e aumento e diminuição da velocidade de tráfego. De maneira geral, o ataque tem a intenção de manter o tráfego intenso de dados com o servidor por um período maior, enquanto que um usuário legítimo realiza a solicitação e encerra ou pausa sua interatividade com o servidor. Observa-se assim que o sucesso na detecção de DDoS *Flash Crowd* pode ser função da interpretação de "intenção" assinada no tráfego e que pode permitir a distinção entre os fenômenos.

A diferença no tipo de tráfego também decorre da característica do *flash crowd*, no qual o aumento na quantidade de usuários e acessos se dá de forma brusca e geralmente em resposta a um evento especial num dado período, tal como na publicação de notícias de grande repercussão, lançamento de filmes, músicas, *softwares*, entre outros. No caso de acesso legítimo, no momento que o usuário recebe a informação desejada do servidor ele tende a prosseguir solicitando outra página ou finaliza seu ciclo no site. Porém em um ataque DDoS *Flash Crowd* o objetivo explícito é sobrecarregar o servidor com solicitações legítimas, mas com más intenções, para causar uma grande lentidão ou, preferencialmente, tornar o servidor inacessível para usuários benignos. Na prática, *botmasters* tendem a forçar seus *bots* a gerar muito mais solicitações web do que um usuário legítimo [Yu et al. 2012]. Em [Dhingra e Sachdeva 2014] observa-se que o tráfego de dados do usuário benigno em um *flash crowd* é variável e tende a formar uma onda em "zigue zague". Esta tendência é em resposta ao comportamento humano de solicitar uma página e após alguns instantes solicitar outra. Outra observação pertinente é que uma eventual lentidão do servidor em função do alto volume de solicitações pode fazer com que o usuário desista e após alguns instantes insista no pedido.

Observa-se, pelo exposto, que eventos *Flash Crowd* e DDoS na camada de aplicação, ou DDoS *Flash Crowd*, compartilham características similares, mas que a diferenciação pode ser realizada via análise comportamental do tráfego.

3. Comportamento Interativo de um Humano e de um Bot

Assumindo como hipótese que o comportamento do evento *Flash Crowd* e do ataque DDoS *Flash Crowd* pode ser analisado e classificado em função do padrão de interatividade em requisições web e do número de requisições, esta seção detalha os comportamentos básicos destes eventos.

A interatividade nas solicitações de um humano junto ao servidor web apresenta um comportamento que varia de acordo com cada ação tomada pelo usuário em seu computador. Inicialmente, o usuário solicita o acesso a um *website* pelo seu *browser*, o que gera a abertura de conexão entre os *hosts* cliente/servidor utilizando o protocolo TCP. Via protocolo HTTP ou HTTPS, neste texto representados por HTTP(S), após a conexão, é realizada a solicitação da *url* primária e, posteriormente, das secundárias que estão incorporadas no corpo do código. Como observado em [Oikonomou e Mirkovic 2009][Xu et al. 2012][Thapngam et al. 2011][Pan et al. 2014], durante esta fase de solicitações legítimas o comportamento pode ser definido como de *interação ativa*, ou seja, quando estão sendo enviadas solicitações HTTP(S) legítimas para o servidor. Dependendo do tamanho dos arquivos a serem carregados e da largura de banda do cliente/servidor, os dados começam a ser transmitidos logo após as solicitações e a transmissão pode perdurar por vários segundos até sua conclusão. Entretanto, este tempo de transmissão até o término do *download* não deve comprometer a observação do comportamento interativo do usuário, pois a troca de dados, mesmo que ocorra de maneira intercalada com as solicitações HTTP(S) do usuário, não é realizada via protocolo HTTP(S). Logo, a troca de dados não interfere na frequência de mensagens das solicitações web. Numa situação de normalidade, após o recebimento dos dados pelo *browser* é esperado que o usuário realize a apreciação das informações apresentadas e leve um tempo variável até a próxima solicitação. Em outras palavras, numa *interação ativa* é esperado que o usuário não realize solicitações em elevada frequência [Hwang et al. 2005][Xie and Yu 2009][Oikonomou and Mirkovic 2009][Thapngam et al. 2011]. Por outro lado, o comportamento de um

usuário maligno, realizando um ataque genérico DDoS *Flash Crowd* de acesso ao conteúdo do *website*, corresponde às solicitações praticamente interruptas.

Ao modelar a interatividade como uma variável aleatória discreta, com valor 0 indicando períodos sem interatividade e 1 indicando período com interatividade, é observado que o comportamento esperado do usuário humano apresenta tempo variável entre novas solicitações, dado a existência de momentos de apreciação das informações (sem interatividade). Por outro lado, o comportamento esperado de um atacante (representado por um *bot*) apresenta maior interatividade. A Figura 1 ilustra os comportamentos típicos e esperados de um humano e de um atacante na notação discreta. Note que a interatividade, mensurada de forma binária (com interatividade e sem interatividade), o número de solicitações (representados pelos traços horizontais) e o tempo entre os estados com interatividade (t), quando analisados por unidade de tempo descrevem um comportamento observável que também permite diferir um usuário humano de um robô de ataque (*bot*).

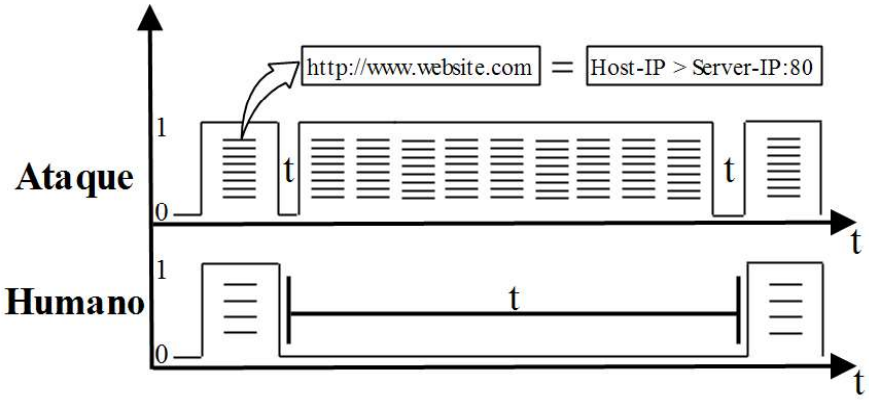

Figura 1. Padrão de interatividade de humano e de atacante.

4. Método para Detecção de Ataques DDoS *Flash Crowd*

Esta seção explora as características apresentadas na seção 3 e apresenta um novo método para detecção de ataques DDoS *Flash Crowd*. O método assume como hipótese básica que a análise do comportamento de um usuário web humano pode se distinguir de comportamentos de usuários web automatizados (*bots*), possibilitando a diferenciação entre *Flash Crowds* e ataques DDoS *Flash Crowd*.

Esta seção está organizada como segue. A seção 4.1 descreve as variáveis utilizadas para modelar o comportamento de humanos e de *bot*s no método proposto e a seção 4.2 apresenta o novo método proposto para diferenciar o comportamento humano do de um *bot* e distinguir entre um *Flash Crowd* e um DDoS *Flash Crowd*.

4.1 Modelagem Comportamental

Para distinguir um comportamento malicioso de um comportamento benigno três variáveis de interesse são modeladas: o número de solicitações HTTP(S) (*NumS*); a taxa de interatividade do usuário (*TxIn*); e o tempo entre estados de interatividade (*Teei*). A seguir, o comportamento de cada uma destas variáveis é discutido e modelado com vistas à diferenciação entre um ataque e um tráfego legítimo.

4.1.1 Número de Solicitações HTTP(S)

Conforme seção 3, e também observado em [Xu et al. 2012][Singh et al. 2015], em ataques DDoS na camada de aplicação o número de solicitações HTTP(S) (*NumS*), aquelas com porta de destino 80 ou 443, tende ser maior para um *host* fonte do tipo *bot* do que para um *host* fonte tradicional (não *bot*), dado as intenções distintas de cada um. Com isso o número de solicitações de um *host* atacante (*NumSH_bot*) tende a ser maior que o de um *host* operado por humano (*NumSH_humano*) em um mesmo período Δt, conforme já ilustrado na Figura 1. Desta forma, para um dado período Δt assume-se o que segue.

$$NumSH\_bot > NumSH\_humano$$

A observação do número de solicitações dos *hosts* permite assim a análise parcial da distinção entre comportamento humano e de um *bot*. Esta análise pode ser realizada em função da intensidade de *NumS*. Deste modo, o método de detecção proposto realiza duas medidas:

(*i*) a média de solicitações do traço analisado no período Δt, dado por $\overline{NumSG}$; e
(*ii*) a média de solicitações por *host* no mesmo período, dado por $\overline{NumSH}$.

O intervalo de tempo Δt é denominado período de atividade observado (*Pat*). Desta forma, em termos do número de solicitações, em um *Pat* é possível avaliar o quão distante um *host* sob análise ($\overline{NumSH}$) encontra-se da média do traço ($\overline{NumSG}$). Salienta-se que para um traço livre de ataque, as duas médias tendem ser próximas.

4.1.2 Taxa de Interatividade

A taxa de interatividade (*TxIn*) de um dado usuário, ou seja, de sua interação ativa, pode ser obtida através da análise da ocorrência ou não de solicitações por período de tempo. Dado o período de atividade observado (*Pat*), correspondente a uma janela de observação de tamanho Δt, discretizada em períodos $\Delta t'$, onde $\Delta t >= 2.\Delta t'$, a taxa de interatividade corresponde a frequência de ocorrência de solicitações nos períodos de tempo $\Delta t'$ dentro período Δt.

A verificação de interatividade de um dado usuário (*In*) é realizada observando em *Pat* se há ou não solicitações do *host* (*NumSH*) para o servidor a cada período $\Delta t'$. O resultado é um vetor de interatividade do host (*VIn*) contendo valores 0s e 1s, onde 0 indica que não existiu solicitação/interação e 1 indica que existiu. O tamanho do vetor depende da granularidade desejada para o *Pat*, ou seja, da janela temporal de observação Δt e da janela temporal considerada no computo da amostra $\Delta t'$.

Desta forma, a taxa de interatividade de um usuário (*TxIn*) deriva da quantidade de períodos com interatividade em um *Pat*. De acordo com a seção 3, a interatividade do *bot* (*In_bot*) tenderá a ocupar todo o período *Pat* e ser maior que a média observada dos comportamentos humanos $\overline{TxIn\_humano}$, ou seja,

$$TxIn\_bot > \overline{TxIn\_humano}.$$

De maneira similar, a interatividade do humano tenderá a um comportamento de baixa interatividade no período observado, tendendo ao padrão médio dos traços observados em período sem ataques $\overline{TxIn\_humano} \cong \overline{TxIn\_traço}$. Salienta-se que em períodos de *Flash crowd* o comportamento humano não é alterado.

4.1.3 Tempo entre Estados Interativos

Entre dois estados interativos existe um intervalo de tempo de não interatividade do usuário. O Tempo entre Estados Interativos (*Teei*), correspondente ao intervalo de tempo de não interatividade, inicia no final de um estado interativo (última solicitação observada de uma interação) e finda na próxima solicitação (início de um estado interativo). Dado a característica de navegação humana apresentadas na seção 3, num traço legítimo o *Teei* médio para um dado host, ou $\overline{Teei}$, tende ser maior do que para *bots*. Como observado em [Xu et al. 2012][Dhingra and Sachdeva 2014][Singh et al. 2015], os períodos de não interatividade para *bots* tendem a zero segundos, e, de acordo com [Hwang 2004][Hwang et al. 2005], para humanos tendem a valores maiores do que 10 segundos. Logo tem-se que

$$\overline{Teeı\_humano} \gg \overline{Teeı\_bot}.$$

4.2 O Método de Diferenciação

Para que a distinção comportamental entre humanos e *bot*s possa ser avaliada, possibilitando diferenciar um *Flash Crowd* de um DDoS *Flash Crowd*, o método proposto, chamado DVM, explora uma nova técnica baseada na análise das diferenças aos valores médios (*DVM*).

Conforme apresentado na seção 4.1, três variáveis de interesse são modeladas (*NumS*, *TxIn* e *Teei*). Cada uma delas pode expressar o comportamento médio de um dado *host* ($\overline{NumSH}$, $\overline{TxInH}$ e $\overline{TeeıH}$) e, juntas, o comportamento médio do traço sob análise ($\overline{NumSG}$, $\overline{TxInG}$ e $\overline{TeeıG}$) . Para um dado período de atividade amostrado (*Pat*), o cálculo das diferenças aos valores médios tem o intuito de explicitar a distância comportamental de um host frente aos comportamentos esperados. À distância, expressa por host (*DVMH*) é função dos percentuais diferenciais relativos ao número de solicitações, taxa de interatividade e tempo entre estados interativos, tal como segue.

$$(DVMH)=\left[\left(\frac{\text{NumSH-NumSG}}{\text{NumSG}}.100\right)+\left(\frac{\text{TxInH-TxInG}}{\text{TxInG}}.100\right)+\left(\frac{\text{TeeiH-TeeiG}}{\text{TeeiG}}.100\right).(\text{-}1)\right]$$

O valor *DVMH* assume magnitude positiva e negativa, e quando zero indica equivalência comportamental do host com a do traço envolvendo todos os hosts. Quanto mais negativo, mais distante o host está de um comportamento maligno. Por outro lado, quando positivo, quanto maior sua magnitude maior é a chance de representar um comportamento malicioso. Na formulação, o cálculo percentual do $\overline{TeeıH}$ é subtraído pelo fato do seu valor desejado ser o maior possível. Quanto maior mais próximo das características humanas, conforme discutido na seção 4.1.3, diferentemente de $\overline{NumSH}$ e $\overline{TxInH}$ que quanto menor mais próximo do humano.

Para avaliar se os valores de *DMVH* indicam ataque é aplicado um limiar (*threshold*). O método considera um limiar adaptativo calculado com base no *threshold* universal por nível [Donoho 1995], tal como aplicado em [Dalmazo et al. 2009]. O limiar adaptativo é dado por

$$L = \sqrt{2.logn(\sigma^2)}$$

na respectiva *Pat*. A Figura 2 ilustra o método de diferenciação.

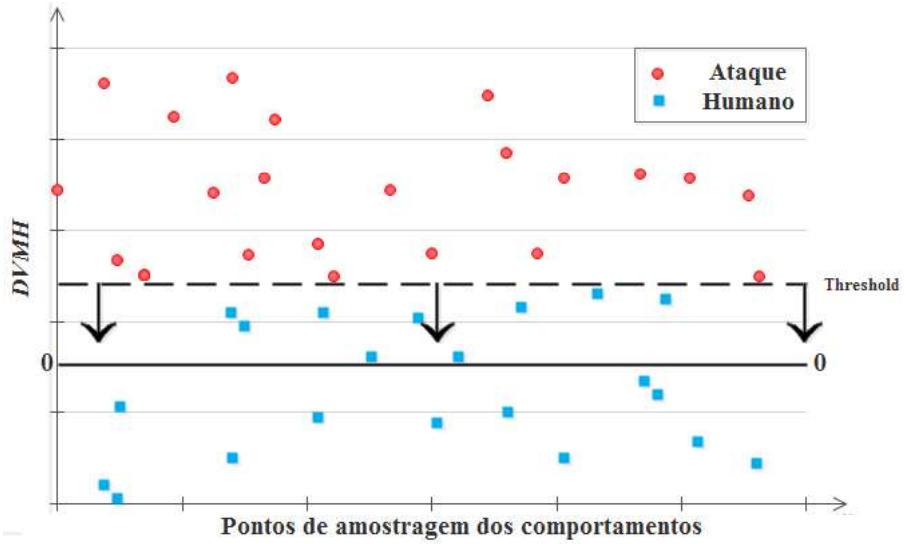

Figura 2. Diferenciação do comportamento humano e de ataque

De acordo com a Figura 2, o método de diferenciação do comportamento humano e de atacantes *bot* adota como parâmetro uma função de corte nas diferenças aos valores médios, que calculadas por *host* permite distinguir bots e identificar ataques DDoS mesmo em momentos de ampliação de tráfego *Flash Crowd*.

5. Resultados Experimentais

A obtenção de amostras reais de tráfego malicioso do tipo DDoS *Flash Crowd* é um desafio. Entretanto, a validação experimental pode também trabalhar com geração de ataques artificiais [Lucena and Moura 2010][Xie and Yu 2009] inseridos ao tráfego de fundo sem ataque. Neste trabalho, para tráfego de fundo são utilizados dados das bases de tráfego público CAIDA [CAIDA 2016] e WITS [Waikato 2016]. A ferramenta utilizada para geração de tráfego anômalo foi a DDoSIM v0.2 [DDoSIM 2016], que também foi utilizada nos trabalhos de [Ye and Zheng 2011][Bekeneva et al. 2015], e permite a simulação de diferentes rajadas de acessos HTTP válidos provenientes de distintos IPs. O tráfego de fundo WITS já foi utilizado em [Lu et al. 2007][Behal and Kumar 2016] e o CAIDA em [Hwang et al. 2005][Xie and Yu 2009][Babu et al. 2011][Righi and Nunes 2015][Bhuyan et al. 2015][David and Thomas 2015].

Inicialmente demonstra-se que o modelo proposto na seção 4 pode modelar o comportamento sem ataques (de humanos), medindo valores médios (*NumSG, TxInG, TeeiG*) de traços reais. Para tal, foram medidos valores médios para os tráfegos de fundo WITS ISPDSL II 2010 06/Jan 15:30h-15:40h e CAIDA 2015 19/Fev 13:00h-13:10h com *Pat* definido em 60s. Salienta-se que os tráfegos das bases são bastante distintos, ou seja, possuem diferença de *throughput,* número de pacotes, tipo de aplicação, dentre outros. Por exemplo, o número de *hosts* analisados nos trechos citados foi de 7.343 na base WITS e 1.989.553 na base CAIDA. Os valores médios obtidos (média para todos os *Pats*) nos dois traços foi: CAIDA apresentou 13,98 segundos e WITS 18,37 segundos para *TeeiG* e para *NumSG* e *TxInG* CAIDA resultou em 32,10 e 3,80 e WITS 33,71 e 4,55, respectivamente. A regularidade e proximidade dos resultados demonstra capacidade do método para modelar o comportamento humano, permitindo a discrimição para não humanos (*bots*). Além disto, este método de análise contribui para a formação do perfil

humano na navegabilidade web, encontrado anteriormente somente por [Jena et al. 2003], que afirma que o humano possui o tempo médio entre interações (*TeeiG*) de 15 segundos.

Para a obtenção dos traços com *Flash Crowd* foram utilizados traços das bases CAIDA e WITS. Os traços foram modificados utilizando como base o resultado do trabalho de [Pan et al. 2014], no qual, apresenta a diferença comportamental do *host* em momento de normalidade e eventos *Flash Crowd.* Os autores apontam que estes traços são 2,35 mais intensos do que sem *Flash Crowd* e que no evento sob análise somente 62,3% dos *hosts* mudam seu comportamento. Desta forma, para a elaboração dos traços com *Flash Crowd* foi realizado incremento de 2,35 em *NumSH* e *TxInH* e decremento em *TeeiH* em 62,3% *hosts*.

Os traços com DDoS *Flash Crowd* usaram o mesmo tráfego de fundo, sendo os ataques elaborados de forma sintética com a ferramenta DDoSIM. A geração do traço anômalo considerou *delay* de 4000 milisegundos entre pacotes *SYN* e 254 *hosts* atacantes, resultando em um comportamento médio de 2000 solicitações em um Δt de 300 segundos. Tal configuração está alinhada ao que foi utilizado no experimento de [Xu et al. 2012].

Para ilustrar que o DVMH é uma variável que permite distinguir *Flash Crowd* de DDoS *Flash Crowd*, plota-se na Figura 3 o DVMH para uma amostra de trecho (a) sem anomalias e com *Flash Crowd* e (b) sem anomalias e com DDoS *Flash Crowd*, onde metade da amostra é sem anomalia e a outra com anomalias. O traço usado é do projeto WITS denominado ISPDLS II 2010 do dia 06/Jan com 60 minutos. Os comportamentos anormais são indicados nos retângulos pontilhados. A análise comparativa da Figura 3(a) e 3(b) permite identificar visualmente o efeito dos cálculos do método proposto, demonstrando que a diferença aos valores médios dos *host*s (DVMH) apresenta variações significativas entre traços com *Flash Crowd* e com DDoS *Flash Crowd*. A diferença no número de cálculos decorre no maior número de *hosts* solicitantes em momentos de ataques DDoS *Flash Crowd*.

Para demonstrar a eficiência do método DVM, foi implementado o algoritmo de [Xu et al. 2012] que visa detectar ataques DDoS *Flash Crowd* realizando a contagem de solicitações HTTP. O método de Xu foi então aplicado aos mesmos traços, sendo calculado o valor de Correlação do Coeficiente de *Pearson* (PCC) a cada Δt de 5 minutos, conforme os autores sugerem, e definido um R de 280, que é o número de *hosts* com maior número de solicitações dentro do Δt, no qual, são utilizados para determinar o PCC do Δt.

Os resultados obtidos com os dados CAIDA e WITS modificados, para conter *flash crowd* de acordo com os indicadores de [Pan et al. 2014] (envolvendo 62,3% dos hosts) e para conter ataques DDoS *flawh crowd* sintéticos gerados pela ferramenta DDoSIM (254 ataques), são sintetizados na Tabela 1. Para tráfegos sem ataques, mas contendo comportamentos anômalos do tipo *flash crowd*, o método DVM, quando comparado com o método de Xu et al., reduziu significativamente a taxa de falsos positivos (de 22,05% para 14,01%). Redução ainda mais significativa foi observada para tráfegos com ataques DDoS (de 31,07% para 8,36¨%). O método também apresentou melhor acurácia, reduzindo significativamente o número de falsos positivos (de 23,90% para 1,97%). Salienta-se que embora o método ainda apresente taxa de falsos positivos relativamente alta (14%), claramente demonstra eficiência na diferenciação entre *Flash Crowd* e DDoS *Flash Crowd* e demonstrando-se uma importante ferramenta para detecção efetiva de ataques DDoS *Flash Crowds* (98,03% de efetividade).

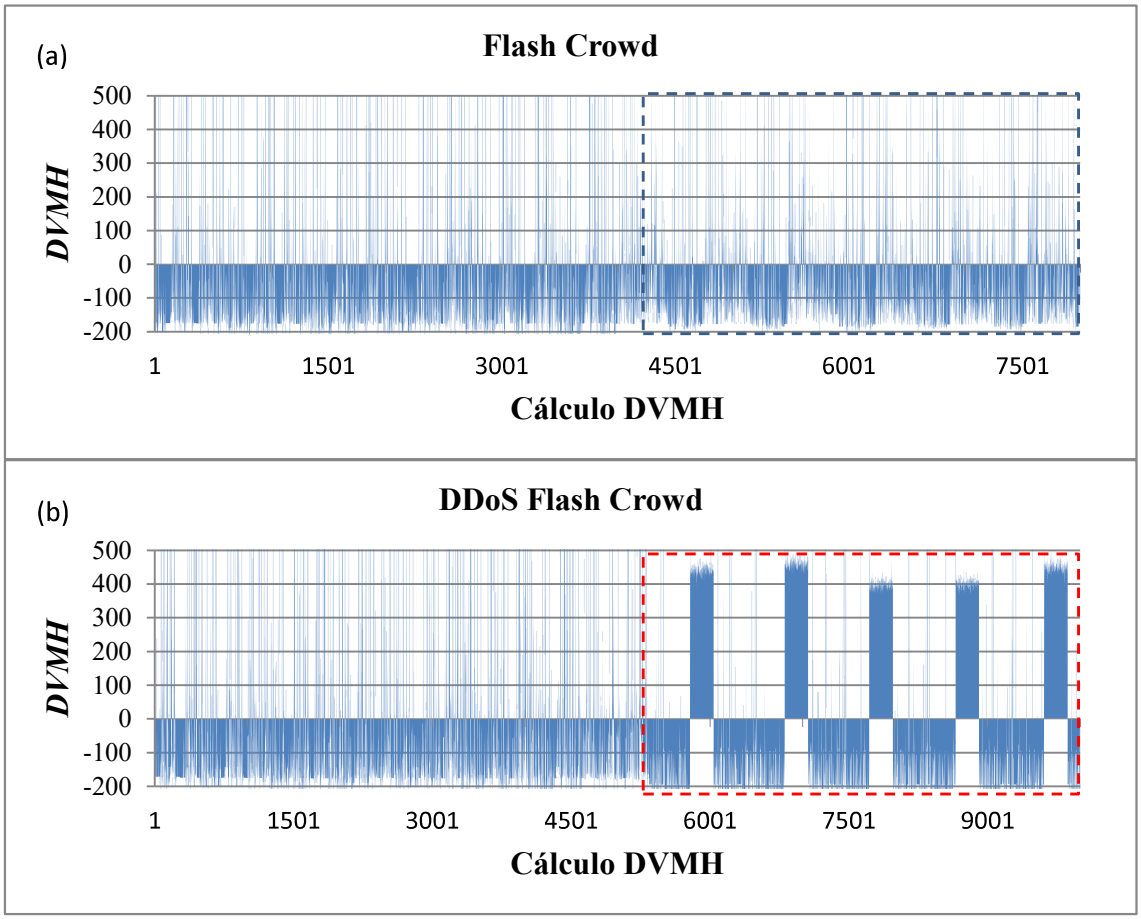

Figura 3. DVMH de tráfego (a) com Flash Crowd e (b) com DDoS Flash Crowd. Traço do projeto WITS denominado ISPDLS II 2010 do dia 06/Jan.

Tabela 1. Resultados da análise de eficiência do método DVM

Método	Base de dados CAIDA e WITS	Ataques	Detecção	Falso Positivo	Falso Negativo
DVM	Tráfegos com *Flash Crowd*	0	-	14,01%	-
Xu et al.		0	-	22,05%	-
DVM	Tráfegos com DDoS *Flash Crowd*	254	98,03%	8,36%	1,97%
Xu et al.		254	76,10%	31,07%	23,90%

Como observado em [Valcy et al. 2011], a técnica NAT (*Network Address Translator*) ainda é bastante utilizada, ou seja, vários usuários podem utilizar um mesmo endereço IP para se comunicar com o servidor. Embora o uso de NAT não tenha sido alvo dos experimentos, ressalta-se que eventual aumento no número de falsos positivos causado pelo uso de NAT pode ser facilmente contornado. Nos experimentos um usuário

foi identificado pelos atributos *host_fonte*, *host_destino* e *porta_destino* (80/443). A simples adoção de mais um parâmetro de identificação dos hosts de usuários, tal como a porta fonte (diferentes hosts atrás do NAT usam diferentes portas para retorno de informações do servidor), pode descriminar os hosts usando NAT.

6. Trabalhos Relacionados

Alguns métodos foram propostos para possibilitar a detecção de ataques DDoS *Flash Crowd*. [Xu et al. 2012] apresenta um algoritmo que utiliza o Coeficiente de Correlação de *Pearson* para analisar o grau de atividade dos usuários contabilizado pelo número de solicitações HTTP. Os autores assumem que após um início de ataque os *bots* não param de enviar solicitações com o propósito de negar o serviço, gerando alto volume de solicitações HTTP e que os humanos terão um comportamento de altos e baixos, oscilando ao longo da análise. Os autores assumem assim que a correlação dos *bots* deve ser mais alta que a dos humanos, permitindo a descriminação dos comportamentos. Entretanto, analisar somente a correlação entre os graus de atividades (número de requisições) pode, ao longo do tempo, acarretar em muitos falsos positivos, principalmente pelo fato do número de solicitações na requisição de uma página web variar muito. Cada requisição possuirá uma quantidade aleatória de *urls* incorporada no seu corpo (*body*), fazendo com que o comportamento do grau de atividade não seja influenciado somente pelo usuário, mas sim pela quantidade de solicitações de cada página, comprometendo a análise discriminatória.

O trabalho de Thapngam et al. (2011) assume que os humanos não possuem uma previsibilidade no comportamento, enquanto os *bots*, por realizarem as solicitações de forma automatizada a partir de um algoritmo, possuem características que se repetem em um período curto de tempo. Desta forma, os autores apresentam um método baseado na análise da repetição da taxa de transmissão de pacotes. A análise avalia a Correlação de *Pearson* entre a taxa de chegada de pacotes x tempo e a autocorrelação da taxa calculada. A limitação desta técnica é seu foco apenas na taxa de chegada de pacotes. Os autores omitiram a quantidade de falsos positivos.

Segundo [Yu et al. 2012] os fluxos de ataques são mais similares entre si do que os fluxos de acessos benignos. Desta forma, os autores apresentam uma técnica que consiste na discriminação utilizando o coeficiente de correlação de fluxos e o cálculo de similaridade entre os fluxos suspeitos. A descoberta que os fluxos em um ataque são bastante semelhantes é um grande avanço, porém, como também discutido no trabalho, o atacante tendo conhecimento da técnica pode customizar o ataque para imitar um acesso benigno, deixando a semelhança entre os fluxos muito equivalentes. Logo, a detecção utilizando a similaridade dos fluxos pode não ser suficiente para construção de um método eficiente.

Neste trabalho a distinção entre comportamento humano e de um *bot* explora uma análise conjunta de outras características, não só da quantidade de solicitações e similaridade entre fluxos. Diferencia-se dos demais por explorar a característica de interatividade e tempo entre estados interativos por método distinto da similaridade entre fluxos.

7. Conclusão

Para distinguir um ataque DDoS *Flash Crowd* de um *Flash Crowd* legítimo este artigo apresenta um método que modela o comportamento de humanos e *bot*s por três características (número de solicitações, tempo entre estados ativos e taxa de interatividade) e realiza uma análise de suas diferenças aos valores médios. Os experimentos demonstram que o método gera resultados satisfatórios na diferenciação entre o comportamento de humanos e de atacantes automatizados (*bots*). O resultado é um novo método para elaborar modelos comportamentais para detecção de ataques DDoS *Flash Crowd*.

Por detectar comportamentos anômalos proveniente de *bot*s, o método pode ser usado também para fornecimento de uma lista de *hosts potencialmente maliciosos* ou para apoiar técnicas de Web QoS [Fang et al. 2015], que permitem priorizar o tráfego para *hosts* que não estejam incluídos em listas de suspeitos. O método também pode ser aplicado em conjunto com métodos de redirecionamento para uma página de *CAPTCHA* [Singh and De 2015], para verificação que não se trata de um *bot*. Diante disto, além da diferenciação entre *Flash Crowd* e DDoS *Flash Crowd*, o método proposto pode ser usado para compor soluções que visam manter a operabilidade de sistemas web minimizando a exploração por *bots*. Trabalhos futuros irão avaliar o desempenho do método em termos de consumo de recursos e tempo de resposta, bem como variantes que permitam contornar falsos positivos decorrentes de um grande número de usuários legítimos operando por meio de NAT/Proxy.

Referências

Babu, G. P.; Jayavani, V.; Mohan Rao, C. P. V. N. J. (2011). Anomaly Detection On User Browsing Behaviors Using Hidden Semi-Markov Model. *Int. Journal of Computer Science and Information Technologies*, v. 2, n. 3, p. 1197-1201

Behal, S. and Kumar, K. (2016). Trends in Validation of DDoS Research. *In: Int. Conf. On Computational Modeling and Security*. Procedia Computer Science n. 85, Elsevier, p. 7-15. doi: 10.1016/j.procs.2016.05.170

Bekeneva, Y., Shipilov, N., Borisenko, K. and Shorov, A. (2015). Simulation of DDoS-attacks and protection mechanisms against them. In: *IEEE NW Russia Young Researchers in Electrical and Electronic Engineering Conference (EIConRusNW)*.

Bhuyan, M. H.; Kalwar, A.; Goswami, A.;Bhattacharyya, D. K.; Kalita, J. K. (2015) Low-Rate and High-Rate Distributed DoS Attack Detection Using Partial Rank Correlation. In: *Int. Conf. On Communication Systems and Network Technologies*, IEEE, doi:10.1109/CSNT.2015.24

CAIDA Anonymized Internet Traces (2015). http://www.caida.org/data/overview/, [acessado em Junho de 2016].

Dalmazo, B. L.; Perlin, T.; Nunes, R. C.; Kozakevicius, A. J. (2009) Filtros de Alarmes de Anomalias através de Wavelets. In: *Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais* – SBSeg, Campinas/SP. Porto Alegre: Sociedade Brasileira de Computação, 2009. v. 1. p. 85-98.

Dantas, R. S. A. (2013). Diferenciação de Ataques DDoS e Flash Crowds. Rio de Janeiro: Instituto Militar de Engenharia. 75p. (Dissertação).

David, D. and Thomas, C. (2015). DDoS Attack Detection using Fast Entropy Approach on Flow-Based Network Traffic. *In: Int. Conf. On Big Data and Cloud Computing*. Procedia Computer Science n. 50, Elsevier, p. 30-36. doi: 10.1016/j.procs.2015.04.007

DDoSIM (2016). https://stormsecurity.wordpress.com/2009/03/03/application-layer-ddos-simulator/, [acessado em Junho de 2016].

Dhingra, A. and Sachdeva, M. (2014). Recent Flash Events : A Study. *International Conference on Communication, Computing & Systems (ICCCS–2014)*, p. 94–99.

Donoho, D. L. (1995). De-noising by soft-thresholding. *IEEE Transactions on Information Theory*, *41*, 613–627.

Fang, W.; Sun, J.; Wu, X.; Palade, V. (2015). Adaptive Web QoS controller based on online system identification using quantum-behaved particle swarm optimization. *Soft Computing*. v.19, n. 6, p. 1715-1725. doi:10.1007/s00500-014-1359-9

Feily, M., Shahrestani, A. and Ramadass, S. (2009). A survey of botnet and botnet detection. *Proceedings - 2009 3rd International Conference on Emerging Security Information, Systems and Technologies, SECURARE 2009*, p. 268–273.

Hwang, F. (2004). Análise dos Efeitos Gerados pelo Comportamento das Aplicações e pelo Perfil das Redes na Característica Auto-Similar do Tráfego Internet. São Paulo: Universidade Estadual de Campinas. 167p. (Dissertação)

Hwang, F., Bianchi, G. R. and Lee Luan Ling (2005). Impacto Gerado pelo Comportamento das Aplicações (Web, FTP e E-mail) e pelo Perfil das Redes na Característica Auto-Similar. *IEEE Latin America Transactions*, v.3, n.4, p.356–361.

Jena, A. K., Popescu, A. and Nilsson, A. A. (2003). Modeling and evaluation of internet applications. *2014 IEEE International Conference on Internet of Things(iThings), and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom)*. IEEE. p. 531–540.

Kandula, S., Katabi, D., Jacob, M. and Berger, A. (2005). Botz-4-sale: Surviving organized DDoS attacks that mimic flash crowds. *2nd Symposium on Networked Systems Design and Implementation (NSDI)*, p.287–300.

Ke, L., Wanlei, Z., Ping, L., Jing, H. and Jianwen, L. (2009). Distinguishing DDoS Attacks from Flash Crowds Using Probability Metrics. In: *Proc. of the International Conference on Network and System Security* - NSS '09, p.09-17.

Lu, K.; Wu, D.; Fan, J.; Todorovic, S.; Nucci, A. (2007). Robust and efficient detection of DDoS attacks for large-scale internet. *In: Computer Networks*, n. 51, Elsevier, p. 5036-5056. doi:10.1016/j.comnet.2007.08.008

Lucena, S. C. and Moura, A. S. (2010). Estimativa de Holt-Winters para Detecção de Ataques em Redes WAN. *X Simpósio Brasileiro Em Segurança Da Informação E De Sistemas Computacionais*, v. d, p. 157–170.

Oikonomou, G. and Mirkovic, J. (2009). Modeling Human Behavior for Defense against Flash-Crowd Attacks. *ICC'09 Proceedings of the 2009 IEEE international conference on Communications*, n. 0430228, p. 625–630.

Pan, J., Hu, H. and Liu, Y. (2014). Human behavior during Flash Crowd in web surfing. *Physica A: Statistical Mechanics and its Applications*, v. 413, p. 212–219.

Prasad, K. M., Reddy, A. R. M. and Rao, K. V (2013). Discriminating DDoS Attack traffic from Flash Crowds on Internet Threat Monitors (ITM) Using Entropy variations. *African Journal of Computing & ICT*, v. 6, n. 2, p. 53–62.

Righi, M. A., Nunes, R. C. (2015). Detecção de DDoS Através da Análise da Recorrência Baseada na Extração de Características Dinâmicas. *XV Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais*, p. 327–330.

Saravanan, R., Shanmuganathan, S. and Palanichamy, Y. (2013). Behavior-based detection of application layer distributed denial of service attacks. *Turkish Journal of Electrical Engineering & Computer Sciences*, p. 1–14.

Singh, B., Kumar, K. and Bhandari, A. (2015). Simulation Study of Application Layer DDoS Attack. *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, p. 893–898.

Thapngam, T., Yu, S., Zhou, W. and Beliakov, G. (2011). Discriminating DDoS Attack Traffic from Flash Crowd through Packet Arrival Patterns. *Computer Communications Workshops* (*INFOCOM*), 2011 *IEEE Conference*, p. 952–957.

Valcy, I., Barreto, L. P., Bezzera, J. (2011). Tratamento Automatizado de Incidentes de Segurança da Informação em Redes de Campus. *XI Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais*, p. 29–42.

Waikato (2016). Waikato Internet Traffic Storage - WITS. http://wand.net.nz/wits/, [acessado em Junho de 2016].

Walfish, M.; Vutukuru, M.; Balakrishnan, H.; Karger, D.; Shenker, S. (2010) DDoS defense by offence. ACM Trans. on Computer System, v. 28, n. 1, Article 3 (August 2010), 54p.

Xie, Y. and Yu, S. Z. (2009). Monitoring the application-layer DDoS sttacks for popular websites. *IEEE/ACM Trans on Networking*, Feb., v. 17, n. 1, p. 15–25.

Xu, C., Du, C. and Kong, X. (2012). An Application Layer DDoS Real-Time Detection Method in Flash Crowd. *International Association of Computer Science & Information Technology (IACSIT)*, v. 30, p. 68–73.

Ye, C. and Zheng, K. (2011). Detection of application layer distributed denial of service. *2011 International Conference on Computer Science and Network Technology Detection*, IEEE, p. 310–314.

Yu, S., Zhou, W. and Member, S. (2012). Discriminating DDoS Attacks from Flash Crowds Using Flow Correlation Coefficient. *Parallel and Distributed Systems, IEEE Transactions*, v. 23, n. 6, p. 1073–1080.

