O Uso da Transformada de Haar na Detecção de Anomalias no Tráfego Web

Cristian Cappo[1], Raul Ceretta Nunes[2], Bruno Augusti Mozzaquatro[2], Alice de Jesus Kozakevicius[2], Christian Schaerer[1]

[1]Facultad Politécnica, Universidade Nacional de Assunção (UNA)

[2]Centro Tecnológico – Universidade Federal de Santa Maria (UFSM)
Caixa Postal 15.064 – 91.501-970 – Santa Maria – RS – Brasil

{ccappo, cschaer}@pol.una.py, {brunomozza, ceretta, alicek}@inf.ufsm.br

Abstract. *Today, information in Computer Systems is a valuable asset which is subject to numerous threats. In web traffic, the set of characters contained in HTTP requests sent to a web application is the main data input for malicious sequences that are created for attackers. Intrusion Detection Systems based on the frequency distribution analysis of this character set are used to identify malicious actions. This paper describes an algorithm for detection of web attacks in HTTP traffic based on the Haar Wavelet Transform and the Hard Threshold. The comparison with other algorithms with different well established strategies showed the efficiency of our approach, that obtained high detection rate with low false positives.*

Resumo. *Em sistemas computacionais a informação é um ativo que está sujeito a inúmeras ameaças. No tráfego web, o conjunto de caracteres contido nas requisições HTTP enviadas a uma aplicação web é a principal entrada de sequências maliciosas dos atacantes. Sistemas de detecção de intrusão baseados na análise da distribuição da frequência deste conjunto de caracteres são utilizados para identificar ações maliciosas. Este artigo descreve um algoritmo de detecção de ataques web baseado em anomalias no tráfego HTTP que aplica a Transformada Wavelet Haar Bidimensional e Hard Threshold. A comparação com algoritmos que usam estratégias diferentes indica a eficiência da abordagem na detecção de ataques web, possibilitando elevar a taxa de detecção.*

1. Introdução

Com o advento da Internet, o número de serviços web disponibilizados aos usuários cresceu consideravelmente. Por outro lado, os programadores tendem a concentrar sua atenção nos aspectos funcionais da aplicação, relaxando com os aspectos de segurança e desencadeando vulnerabilidades. Deste modo, ataques são utilizados para explorar falhas e violar a segurança da informação de dados sigilosos. Segundo [Symantec 2013], foram descobertas 5.291 novas vulnerabilidades só no ano de 2012, o que gerou um aumento de 30% nos ataques baseados na web.

Os Sistemas de Detecção de Intrusão (*Intrusion Detection System* - IDS) são ferramentas que visam proteger as vulnerabilidades tanto em redes de computadores como em aplicações web [Robertson 2009], pois analisam a existência de assinaturas

de ataque e/ou anomalias a fim de detectar eventuais ameaças ao sistema computacional [Northcutt and Novak 2002].

A abordagem baseada em assinatura identifica ataques através de um conjunto de informações contendo padrões de ataques (assinaturas) previamente definidos [Kruegel et al. 2004]. A vantagem dessa abordagem é a baixa taxa de falsos positivos na detecção dos ataques. Por outro lado, a abordagem baseada em anomalias identifica ataques através de um conjunto de observações (sinal) e suas variações [Chandola et al. 2009]. Essa abordagem visa identificar o comportamento que não segue a definição considerada usual, portanto, proporciona a detecção de ataques desconhecidos (ataques *zero-day* e/ou mutações).

Abordagens baseadas em anomalias têm sido utilizadas em diversos trabalhos propostos na literatura para a detecção de ataques web [Kruegel and Vigna 2003] [Ingham and Inoue 2007] [Kiani et al. 2008]. Para detecção, esses trabalhos utilizam um perfil normal do comportamento (conhecido como *normal profile*), que é definido numa fase de treinamento através de um conjunto de dados sem a ocorrência de ataques (dados de treinamento). Porém, em sistemas de detecção de intrusão baseados em anomalias, a definição do perfil normal é um desafio, principalmente porque além de ser difícil obter um conjunto de dados sem ataques que represente todas as atividades reais, também possui um alto custo financeiro e temporal [Jamdagni et al. 2010].

Ataques via comunicação HTTP manipulam requisições, colocando em risco as funcionalidades dos serviços na Internet [Alvarez and Petrovic 2003] e, podem causar anomalias no tráfego. Nos ataques, a inserção de informações é realizada principalmente via técnicas de injeção de código ou conteúdo malicioso [OWASP 2013] e pode causar variações significativas na frequência dos caracteres em requisições web [Su and Wassermann 2006]. Por exemplo, o *Path Traversal* é caracterizado pela utilização de uma grande quantidade de caracteres "." e "/" em relação aos outros caracteres [Robertson et al. 2006]. A variabilidade dos caracteres contidos nesses tipos de ataques podem ser assim detectados através do uso de ferramentas que proporcionam uma análise entre os caracteres de uma mesma requisição e também entre um grupo de requisições web.

O desafio é que as ferramentas de detecção atuais necessitam processar milhões de conexões e possuem alto custo computacional para identificar ameaças no tráfego HTTP em tempo real [Marques and Baillargeon 2005]. Neste cenário, a transformada wavelet [Daubechies 1992] é uma ferramenta de processamento de sinais que potencializa a aplicação em diferentes contextos. Sua principal característica é a identificação de variações significativas e a correlação entre diferentes fontes de detalhes para a análise do sinal. Em especial, a aplicação da transformada wavelet de Haar [Stollnitz et al. 1995] possui baixo custo computacional, devido ao uso exclusivo de operações de adição e subtração, e a transformada wavelet bidimensional é apropriada para a análise de caracteres em grupo de requisições web [Mozzaquatro et al. 2011].

A aplicação da transformada de wavelet como ferramenta para a detecção de variações significativas na distribuição da frequência dos caracteres em um conjunto de requisições web e a aplicação de critérios para a operação de corte sobre os coeficientes resultantes da transformada foram explorados em [Mozzaquatro et al. 2011] e

[Cappo et al. 2012], respectivamente. Porém, ataques que geram perturbações de menor escala na frequência dos caracteres permanecem de difícil detecção.

Este trabalho explora a eficiência de uso da transformada de Haar na detecção de anomalias quando incrementado com uma técnica de realce das variações de menor escala. Como resultado, o algoritmo de detecção de anomalias proposto permite detectar ataques que eram inicialmente encobertos por outras variabilidades nas requisições web. A técnica considera a sensibilidade dos coeficientes wavelets diante de variações do sinal, por isto estabelece um limiar de corte adequado conforme a aplicação web analisada e atua na potencialização das diferenças entre coeficientes (realce).

O artigo está organizado como segue. A seção 2 apresenta os trabalhos relacionados. Na seção 3 a transformada wavelet de Haar é descrita, bem como suas características que potencializam a detecção de intrusão em sistemas computacionais. A seção 4 apresenta o modelo de dados e o algoritmo de detecção com a fase de realce. Os resultados dos experimentos são descritos na seção 5. As conclusões são apresentadas na seção 6.

2. Trabalhos Relacionados

O desafio dos algoritmos de detecção de ataques web é identificar ataques analisando perturbações nos caracteres de uma requisição web [Robertson et al. 2006]. Tradicionalmente, as perturbações, anomalias no caso de ataques, são analisadas através de máquinas de aprendizagem que utilizam uma fase de treinamento, para definição do comportamento usual da aplicação web (*normal profile*), e uma fase para a detecção de ataques [Henke et al. 2011].

Na fase de treinamento as frequências usuais dos caracteres são contabilizadas e organizadas. Kruegel e Vigna [Kruegel and Vigna 2003] agrupam as frequências relativas dos caracteres e as adicionam em agrupamentos em ordem decrescente tal como, por exemplo: $\{$[0], [1-3], [4-6], [7-11], [12-15] e [16-255]$\}$. O trabalho de [Kiani et al. 2008] explora a frequência cumulativa, realizando agrupamentos seguindo a correção de Yates [Mamahlodi 2006], isto é, cada grupo contém no máximo 5 amostras. A quantidade de agrupamentos depende dos dados analisados. Este trabalho explora a frequência cumulativa dos caracteres, mas não utiliza fase de treinamento. Diferentemente, utiliza uma fase de determinação de pesos para o processo de realce das frequências de interesse, a qual é computacionalmente mais eficiente, pois não realiza agrupamentos.

Na fase de detecção os algoritmos [Kruegel and Vigna 2003] [Kiani et al. 2008] comparam os dados previamente analisados e agrupados (fase de treinamento) com os dados observados. O teste de Pearson χ^2 tem sido a técnica mais utilizada para comparação da similaridade entre as distribuições de frequências dos caracteres (esperada e observada). Diferentemente, a aplicação da transformada wavelet permite identificar anomalias analisando apenas as frequências observadas [Mozzaquatro et al. 2011]. Em Mozzaquatro et al. [Mozzaquatro et al. 2011], a transformada wavelet é aplicada diretamente sobre as frequências acumuladas dos caracteres observados, eliminando a necessidade de uma fase de treinamento. Diferentemente, neste trabalho, que também aplica a transformada wavelet e não usa fase de treinamento, a estratégia proposta inclui uma fase preliminar de realce das frequências de interesse. Esta fase permite melhorar a eficiência da fase de detecção, favorecendo a identificação de ataques de menor escala de perturbação.

3. Transformada Wavelet

A transformada wavelet é utilizada para decomposição de um sinal em diferentes níveis de resolução com o objetivo de realçar e salientar informações relevantes da composição do sinal. Ela permite a decomposição hierárquica de um sinal em uma representação grosseira e um grupo de detalhes [Stollnitz et al. 1995]. Os detalhes são informações complementares da representação grosseira e são necessários para reconstrução do sinal original. Em aplicações de análise de sinais, eles são manipulados para permitir a compressão, remoção de ruídos ou outros extrações de características dos dados analisados.

Conforme Daubechies [Daubechies 1992], a família ortonormal de funções wavelet é definida por seus filtros passa-alta e passa-baixa de tamanho H e L, respectivamente. Estes filtros são responsáveis pela rápida transformada unidimensional (TW1D), cuja versão discreta é definida pela seguinte relação conhecida como algoritmo cascata rápida [Mallat 2009]. Considerando como entrada inicial o vetor $c_{J,s}$ $s = 0, ..., M_J - 1$ no nível mais fino J com $M_J = 2^J$ pontos, teremos as seguintes relações para k níveis de uma transformada wavelet discreta, quando $j = J, J-1, ..., J-k$:

$$c_{j-1,i} = \sum_{k=0}^{2N-1} L_k c_{j,2i+k}, \quad i = 0, ..., M_{j-1} - 1, \tag{1}$$

$$d_{j-1,i} = \sum_{k=0}^{2N-1} H_k c_{j,2i+k}, \quad i = 0, ..., M_{j-1} - 1, \tag{2}$$

gerando o vetor $c_{j-1,i}$ com a informação de aproximação e o vetor $d_{j-1,i}$ com os coeficientes wavelets (detalhes), ambos com $M_{j-1} = M_j/2$ pontos. A informação complementar para recuperar o nível j baseada em $c_{j-1,i}$ é dada pelos detalhes.

Na transformada wavelet de Haar os filtros que definem a família wavelet são dados por $L_0 = \frac{1}{\sqrt{2}}, L_1 = \frac{1}{\sqrt{2}}$ e $H_0 = \frac{1}{\sqrt{2}}, H_1 = -\frac{1}{\sqrt{2}}$. Os diferenciais da transformada de Haar são: a simplicidade de implementação e a facilidade de preservação da localização das informações. Além disso, a identificação de variações abruptas também pode ser realizada através de suas transformadas associadas [Mallat 2009], o que é uma propriedade muito importante para preservar a localização precisa das variações.

3.1. Transformada Wavelet 2D

A transformada wavelet bidimensional (TW2D) é a generalização da TW1D, através da aplicação da transformada wavelet unidimensional nas linhas e nas colunas. A TW2D proporciona a realização de uma análise dos dados e identificação de variações entre as linhas e também entre as colunas. Além disso, na decomposição da TW2D outro conjunto de variações está relacionado com a combinação linear de ambos. Isto é, além de considerar as variações das direções horizontal e vertical, também considera-se a direção diagonal. A Figura 1 ilustra um nível de transformação para uma matriz X de entrada.

Quando é realizada a TW1D nas linhas (aplicando o filtro de passa-baixa (L) e o filtro de passa-alta(H)), cada linha é comprimida ("c") gerando um conjunto de coeficientes wavelet, etiquetados por "d". Logo, a TW1D é aplicada em todas as colunas sobre os dados após a transformada das linhas. Como consequência, quatro blocos são gerados: o bloco de aproximação ("cc") e os blocos de coeficientes wavelet, considerados os detalhes

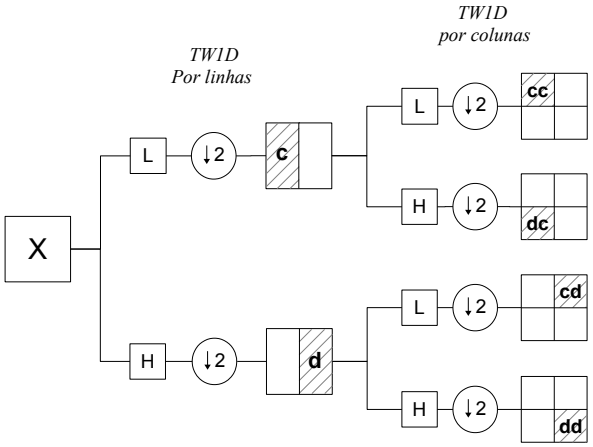

Figura 1. Esquema da TW2D para um nível de transformação.

(dc, cd, dd) de tamanho T, onde $T = M_j/4$ considerando o nível mais fino j. Quanto mais níveis de transformação são realizados, o bloco associado ao "cc" é decomposto, nomeadamente, em bloco de coeficientes de aproximação.

A principal propriedade da TW2D permite verificar a variabilidade do sinal em múltiplas direções. A TW2D proporciona uma análise do sinal na direção horizontal, vertical e diagonal. No entanto, mais fontes de informações com detalhes são disponibilizadas na análise da TW2D, que na TW1D não é possível, visto que é gerado somente um conjunto de detalhes [Mozzaquatro et al. 2011].

3.2. Operação de Corte

Após a aplicação da transformada wavelet bidimensional, o sinal é representado por quatro sub-bandas diferentes *cc*, *cd*, *dc* e *dd*, conforme explicado na subseção 3.1. As subbandas associadas aos coeficientes wavelets representam as variações dos dados capturados em três diferentes direções. De acordo com uma das principais propriedades da transformada wavelet, as informações grosseiras (médias dos coeficientes) representam com precisão os dados originais. Portanto, a determinação do tamanho (significado) dos coeficientes wavelets é uma ferramenta para eliminar dados irrelevantes da representação ou para detectar variações relevantes nos dados analisados.

Assim, os critérios para descartar coeficientes wavelets são baseados na comparação entre os coeficientes de bandas *cd*, *dc* e *dd* com algum valor de referência, denominado valor de limiar (*threshold*). Um dos valores mais utilizados em aplicações de filtragem é o *threshold* universal proposto por [Donoho and Johnstone 1995]. Esse valor de filtragem é determinado pelo $\lambda = \sigma\sqrt{2\log(N)}$, onde σ corresponde ao desvio padrão dos coeficientes, N corresponde ao número de amostras e $(\frac{N}{2})^2$ é o tamanho de cada subbanda. Quando o valor do *threshold* é escolhido de forma consistente, a operação de *threshold* não destrói as propriedades fundamentais do sinal [Donoho and Johnstone 1995]. A definição do *threshold* universal é um valor de corte para filtragem de ruídos.

Neste trabalho o valor de corte é usado para determinar os valores que são con-

siderados variações significativas e que correspondem a uma anomalia. Assim, para determinar o valor de corte é utilizado a estratégia de [Bilen and Huzurbazar 2002], que estima σ como a média do desvio absoluto da mediana dos dados, denominado em [Cappo et al. 2012] como AD. Desta forma, o algoritmo de detecção difere do apresentado em [Mozzaquatro et al. 2011], onde é apresentado um fator de ajuste ρ de acordo com a aplicação ($\lambda = \rho \cdot \sigma\sqrt{2\log(N)}$).

Uma vez definido o valor de limiar, estratégias de filtragem dos coeficientes são utilizadas. O trabalho de [Donoho and Johnstone 1995] propôs duas maneiras de filtrar os coeficientes wavelets: aplicando o *Hard threshold* (Equação 3) ou através da aplicação do *Soft threshold* (Equação 4). O *Hard threshold* corta os coeficientes wavelet sem modificar os coeficientes maiores do que o valor do *threshold* e, portanto, considera como significativos. Essa estratégia é mais restritiva do que o *Soft threshold* que suaviza todo o conjunto dos coeficientes wavelets. Neste trabalho consideramos o uso do *Hard threshold* definido pela Equação 3 e os elementos analisados são os coeficientes wavelet dos blocos *cd*, *dc* e *dd*, para cada nível e cada posição com relação ao nível.

$$d(k) = \begin{cases} 0 & , \mid d(k) \mid < \lambda \\ d(k) & , \mid d(k) \mid \geq \lambda \end{cases} \tag{3}$$

$$d(k) = \begin{cases} 0 & , |d(k)| < \lambda \\ d(k) - \lambda & , |d(k)| \geq \lambda \wedge d(k) \geq 0 \\ d(k) + \lambda & , |d(k)| \geq \lambda \wedge d(k) < 0 \end{cases} \tag{4}$$

4. Proposta

Nesta seção é proposto um novo detector de anomalias baseado na utilização da transformada wavelet de Haar. O detector explora a transformada Haar bidimensional, juntamente com um mecanismo para o realce nos dados de ataques que apresentam baixa variação na distribuição de frequência de caracteres, para elevar a eficiência na detecção de anomalias no tráfego web e maximizar a taxa de detecção. O detector avalia as frequências dos caracteres com base no modelo de dados apresentado na seção 4.1. Denominado Algoritmo de Detecção baseado na TW2D com Fase de Realce (descrito na seção 4.2), o algoritmo utiliza como critério de decisão de anomalia um limiar *Hard Threshold* denominado *AD*, cuja efetividade já havia sido apresentado em [Cappo et al. 2012].

4.1. Modelo de Dados

A análise com a transformada wavelet considera as frequências dos caracteres contidos em cada requisição web. O conjunto de requisições web é agrupada em uma janela de processamento. Desta forma, os dados são organizados como uma matriz X_{rc}, $0 \leq r \leq 255$ e $1 \leq c \leq m$, onde o valor m é o número de requisições para a análise com a TW2D e cada uma das 256 linhas representa o conjunto de caracteres ASCII. Este modelo é graficamente representado na Figura 2. Note que um ataque web que afeta a distribuição de frequência gera variações que são refletidas na matriz de dados.

Por exemplo, considere uma aplicação denominada "*page.php*" com um parâmetro na consulta (*query*) com formato `GET /page.php?p=horarios` . A Figura 3 ilustra o comportamento de requisições com ataques que perturbam a distribuição dos caracteres

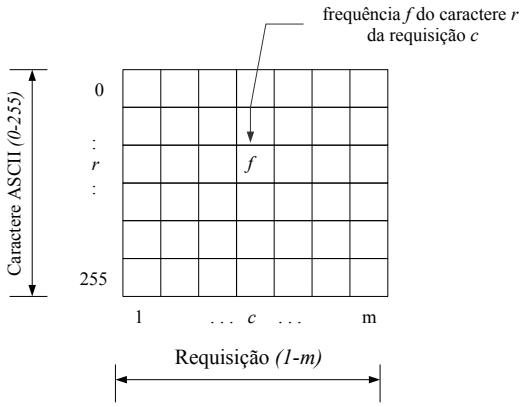

Figura 2. A matriz de entrada de dados, onde a interseção entre cada linha e coluna consiste na frequência do caractere ASCII (linha) de cada requisição (coluna).

numa janela de processamento. A parte a) da figura ilustra a regularidade dos caracteres, permitindo identificar a frequência dos caracteres através da intensidade de cor: uma frequência alta corresponde a uma cor intensa e uma frequência baixa a uma cor clara. A inclusão de ataques dos tipos *Directory Traversal* (requisição de número 50) e XSS (requisição de número 100) podem ser visualizados na parte b) da Figura 3 (picos na figura). Nota-se assim, que esta representação de dados habilita a análise da distribuição da frequência de caracteres em requisições web, sendo apropriada para a análise wavelet.

(a) (b)

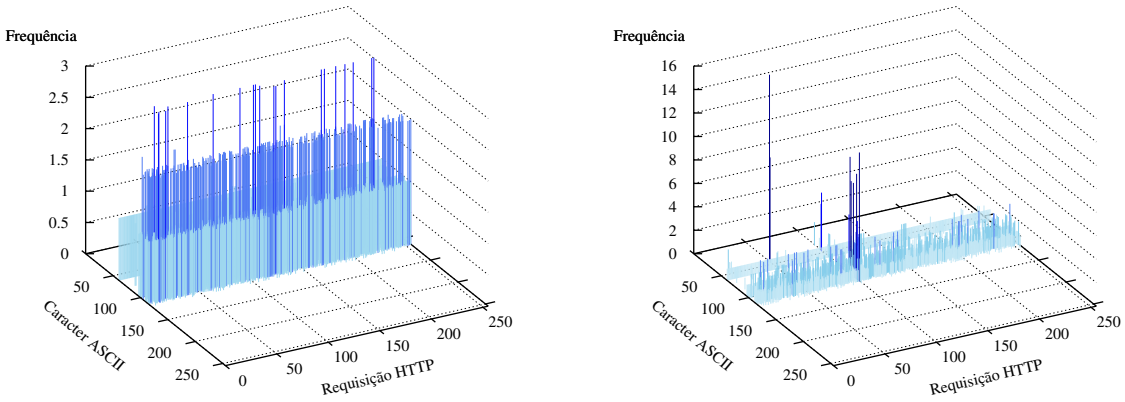

Figura 3. Comportamento dos ataques inseridos na distribuição de caracteres para uma janela de processamento. a) Sem ataques b) Com ataques

4.2. Algoritmo de Detecção de Anomalias com Fase de Realce

A proposta de detecção de anomalias é ilustrado na Figura 4. Este esquema é composto de duas fases: a primeira corresponde ao cálculo dos pesos de acordo com as frequências dos caracteres (pré-detecção). A segunda corresponde ao processo de detecção. Os pesos são utilizados para realçar as frequências dos caracteres menos frequentes. Uma vez realizada a análise das frequências, é aplicado o algoritmo de detecção de anomalias. Se for detectado algum ataque, um alarme é emitido.

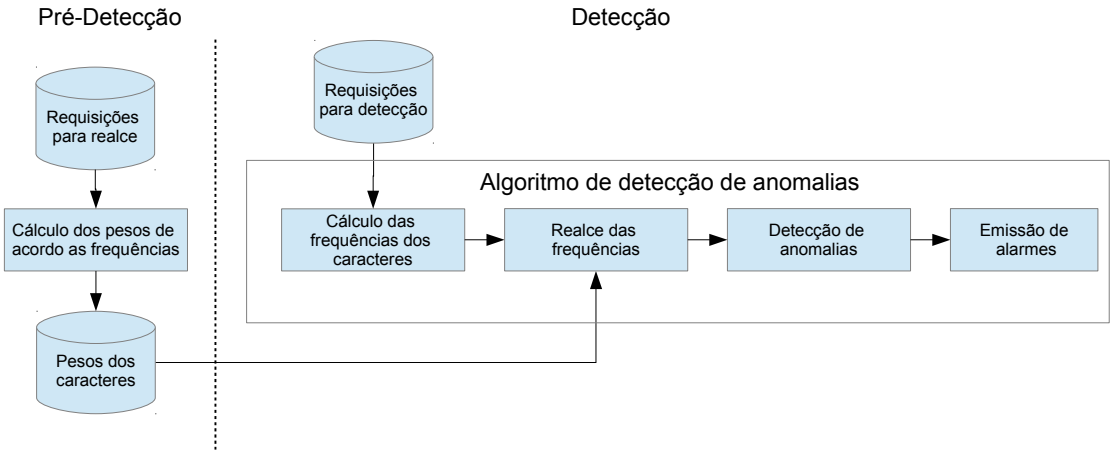

Figura 4. Esquema de detecção de anomalias.

A seguir é descrita a fase de realce (seção 4.2.1) e o algoritmo de detecção baseado na TW2D de Haar (seção 4.2.2).

4.2.1. Fase de Realce das Frequências

A fase de realce das frequências é realizada em duas etapas.

(Etapa 1) *Cálculo de pesos por caractere*. Considere k matrizes de entrada X e a frequência $f_j(c)$ do caractere c na janela $j = 1..k$. A Equação 5 determina o peso $p(c_i)$ de cada caractere. Este processo é realizado antes da etapa de detecção sendo os pesos parte da entrada do algoritmo de detecção. O peso $p(c_i)$ calculado para o *i*-ésimo caractere aumenta na medida que a frequência do caractere é menor e diminui quanto maior a frequência do caractere.

$$p(c_i) = \begin{cases} \frac{1}{\sum_{j=1}^{k} f_j(c_i)} & , \quad \sum_{j=1}^{k} f_j(c_i) > 0 \\ 1 & , \quad \sum_{j=1}^{k} f_j(c_i) = 0 \end{cases} \quad i = 0..255 \tag{5}$$

(Etapa 2) *Realce das frequências*. Quando aplicada a fase de realce, a frequência observada na janela de processamento é ajustada de acordo com a Equação 6, onde CTE é uma constante definida pelo usuário e que permite adequar a sensibilidade do ajuste de acordo com a aplicação. Quanto maior a variabilidade dos dados da aplicação maior deve ser o CTE. Como resultado, o valor $f^*(c_i)$ corresponde a frequência ajustada ao caractere c_i e o valor $f(c_i)$ ao valor da frequência observada na janela de processamento.

$$f^*(c_i) = \begin{cases} f(c_i) + p(c_i) * CTE & , \quad f(c_i) > 0 \\ 0 & , \quad f(c_i) = 0 \end{cases} \quad i = 0..255 \tag{6}$$

4.2.2. Algoritmo de Detecção de Anomalias

A detecção de anomalias é realizada aplicando o algoritmo da Figura 1.

Algoritmo 1: Algoritmo de Detecção baseado na TW2D com Fase de Realce.

Entrada: Conjunto P de m requisições web
Conjunto p de pesos por caractere p_c $c = 0..255$
Constante CTE de ajuste para as frequências
Saída : Conjunto A das posições com ataques

```
1  X = calcular frequências do conjunto P de acordo com o modelo da Figura 2
   // Fase de realce de acordo com a Equação 6
2  for r = 1 to m do
3  |  for c = 0 to 255 do
4  |  |  if X_rc > 0 then
5  |  |  |  X_rc = X_rc + p_c * CTE
6  |  |  else
7  |  |  |  X_rc = 0
   //Fase de Detecção
8  A ← ∅
9  (cc, dc, cd, dd) ← TW2D[X]   //Um nível de transformação
10 Para cada bloco (cc, dc, cd, dd) calcular o threshold λ do bloco
11 Para cada bloco (cc, dc, cd, dd) marcar a posição x, y se |d_xy| > λ do bloco
12 if a posição x, y foi marcada em pelo menos dois blocos then
13 |  A ⟵ A + (x, y)   // É um ataque na requisição x causada pelo caractere y
14 return A
```

Baseado na transformada wavelet bidimensional de Haar e na aplicação de operação de corte sobre os coeficientes wavelets ($threshold$), os parâmetros de entrada do algoritmo correspondem a m requisições web, ao conjunto de pesos p calculados segundo a Equação 5 e à constante CTE, utilizada para a fase de realce das frequências.

Na linha 1, é gerada a matriz X de acordo com o modelo detalhado na seção 4.1. Entre a linha 2 e a linha 7 a matriz é processada para realçar as frequências de acordo com o conjunto de pesos p e o valor CTE. A partir da linha 8 é realizado o processo de detecção. A transformada wavelet é aplicada sobre a matriz de dados na linha 9. Aplica-se apenas um nível de transformação, o que é suficiente para a detecção de picos [Bilen and Huzurbazar 2002]. Para cada bloco da transformada (quadrante da transformada 2D) é calculado um valor de limiar utilizando o esquema AD apresentado em [Cappo et al. 2012] de acordo com [Huber 1981]. O esquema AD é uma estimativa do valor σ do limiar universal (*Threshold Universal*) apresentado na seção 3.2. O mesmo corresponde à média do desvio absoluto dos coeficientes wavelets e $ad = \frac{1}{N}\sum_{i=1}^{N}|d_i - \gamma(G)|, i = 1 \ldots T$, onde γ denota a mediana dos coeficientes wavelets $d_i > 0$ que correspondem ao bloco analisado G.

Finalmente, na linha 12, uma posição da matriz de entrada é considerada uma anomalia se, pelo menos em dois dos três blocos (cd, dc, dd), a frequência é maior que o valor do limiar calculado. Esta heurística assume que uma mudança abrupta na variação deve ocorrer em pelo menos duas direções de análise na mesma posição x, y. Cada bloco gerado pela TW2D fornece informações sobre a variação em direções vertical, horizontal

e diagonal ao mesmo tempo. Como resultado, o conjunto A de posições anômalas é informado ao módulo de emissão de alarmes.

5. Experimentos e Comparações

Esta seção apresenta o efeito da aplicação da fase de realce na detecção de anomalias. Portanto, são apresentados os resultados obtidos dos experimentos numéricos com o método de realce das frequências e o algoritmo de detecção baseado na TW2D (denominado algoritmo TW2D). Ao final foi adicionada a comparação com outros três métodos de detecção de anomalias discutidos na seção 2.

O conjunto de dados utilizado nos experimentos para a análise da identificação de anomalias contém as consultas realizadas pelos clientes a servidores web. Para os experimentos foram consideradas somente requisições do tipo GET (solicitações de recursos). Estas requisições foram coletadas no formato de log (Apache log) no servidor web da Faculdade Politécnica da Universidade Nacional de Assunção (UNA) durante um período de três meses. Estes dados correspondem ao servidor web da Faculdade Politécnica da Universidade Nacional de Assunção (UNA), coletados durante um período de três meses. No total foram analisadas 59248 requisições distribuídas em 252 janelas de processamento contendo 256 requisições para cada uma ($m = 256$).

Neste conjunto de dados foram injetados diferentes tipos de ataques para observar a eficiência da proposta de detecção de anomalias. Na Tabela 1 são especificados os tipos de ataques inseridos e a quantidade presente no conjunto de dados para análise.

Tipo de ataque	Exemplo	Quant.
FileInclusion	`/page.php?p=http://www.manchenumerique.fr/voeux2008/rss.txt??`	1
CodeRed	`/page.php?p=xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx`	2
Directory Traversal	`/page.php?p=../../../../../../etc/passwd%00`	8
XSS	`/page.php?p=<scr<script>ipt>alert(document.cookie)</script>`	5
SQLInjection	`/page.php?p=gd_index and 1 = 1`	5
OSInjection	`/page.php?p=/bin/ping`	1
Total		22

Tabela 1. Tipos de ataques web e a quantidade inseridos no conjunto de dados analisado.

5.1. Efeito do Realce das Frequências

Para demonstrar o efeito do realce das frequência dos caracteres, a Figura 5(a) ilustra o momento antes da aplicação do módulo de realce e na Figura 5(b) após a aplicação do módulo de realce. Neste exemplo foi considerado uma janela de processamento com 6 requisições anômalas (4 do tipo *Directory Traversal*, 1 *CodeRed* e 1 *FileInclusion*).

Como pode ser visualizado, quando não se aplica o módulo de realce, o ataque do tipo *FileInclusion* (requisição 100) passa despercebido entre as demais requisições. Entretanto, quando é aplicado o módulo de realce é possível observar na Figura 5(b) que o ataque pode ser perfeitamente distinguido entre as demais requisições. Além disso, pode ser observado o caractere %00 (byte 0) utilizado em muitos ataques com uma única aparição e com o realce torna-se distinguível (posição 0 na legenda "Caracteres ASCII").

5.2. Resultados

Com intuito de aplicar o algoritmo de detecção de anomalias foram utilizadas 1024 requisições para o processo de pré-detecção utilizado no módulo de realce de frequências. Os dados utilizados estavam livres de ataques e selecionados conforme a definição de

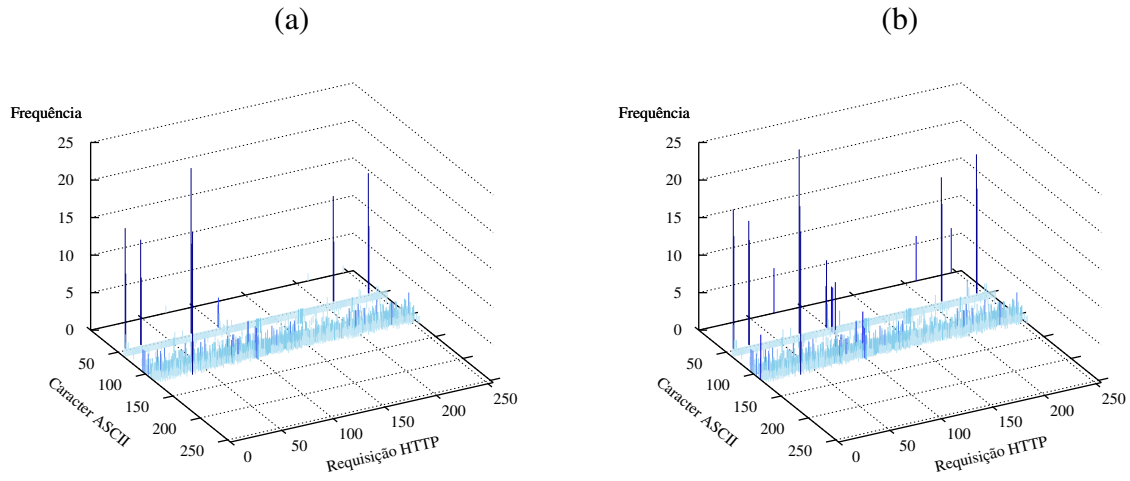

Figura 5. Comportamento do módulo de realce das frequências (com 6 anomalias inseridas). a) Sem realce b) Com realce.

requisições usuais para uma determinada aplicação. A quantidade de requisições selecionada corresponde a 4 janelas de processamento.

A Tabela 2 apresenta os resultados entre a diferença na aplicação do módulo de realce de frequências incorporado ao algoritmo TW2D com a quantidade de verdadeiros positivos (TP) e falsos positivos (FP), obtidos durante o experimento. Para o método de realce de frequências foi utilizado $CTE = 7$ na Equação 6, com o qual se obteve os melhores resultados. Também é apresentado os resultados de precisão (P), recuperação (R) e o cálculo do F-Measure (F), que avalia o *trade-off* entre a precisão e a recuperação de acordo a seguente fórmula: $F = \frac{2*R*P}{R+P}$.

Tipo de ataque	Total	Sem Realce	Com realce
FileInclusion	1	0	1
CodeRed	2	2	2
Directory Traversal	8	8	8
XSS	5	2	5
SQLInjection	5	0	5
OSInjection	1	0	1
TP	22	12	22
FP	0	0	4
Precisão		100%	85%
Recuperação		55%	100%
F-Measure		71%	92%

Tabela 2. Resultados da detecção obtidos com realce e sem realce do alg. TW2D.

Os resultados demonstraram que a aplicação de realce nas frequências possibilitou a identificação dos ataques web que foram inseridos no conjunto de dados, inclusive aqueles com baixa frequência de caracteres (o caso do ataque *FileInclusion*, *SQLInjection* e *OSInjection*). Entretanto, 4 falsos positivos foram detectados, isto é, foram gerados alarmes para requisições sem nenhum tipo de ataque inserido. A justificativa para esses falsos positivos é que alguns caracteres não usuais para a aplicação web foram utilizados nas requisições. Como se pode observar o valor de F-Measure é próximo do ótimo para o método com realce.

5.3. Comparação com Métodos de Detecção de Anomalias

Os métodos utilizados para realizar as comparações estão baseadas no método do Teste de Person χ^2 [Kruegel and Vigna 2003] [Kruegel et al. 2005] com 6 grupos (*6BIN*), na distância de Mahalanobis [Wang and Stolfo 2004] (*MD*) e usando *ngram* [Ingham and Inoue 2007] (*NGRAM*). Nas comparações com NGRAM foi considerado o seguinte: c é o número de n-grams presentes no conjunto de requisições normais e t é o número total de n-grams na requisição analisada na fase de detecção. Dado $T \in [0, 1]$ é o limiar (threshold), a requisição analisada é uma anomalia, se $\frac{c}{t} < T$. Para essas comparações foi considerado $T = 0.95$ (pelo menos 95% dos n-grams devem pertencer ao conjunto de n-grams normais). O tamanho do n-gram utilizado nos testes foram de 2 a 10. Os resultados das comparações estão apresentados na Tabela 3 com uma estrutura similar à Tabela 2.

Tipo de ataque	Total	*TW2D com realce*	*6BIN*	*MD*	*NGRAM*
FileInclusion	1	1	0	1	1
CodeRed	2	2	2	0	2
Directory Traversal	8	8	8	6	8
XSS	5	5	0	5	5
SQLInjection	5	5	0	5	5
OSInjection	1	1	0	1	1
TP	22	22	10	20	22
FP	0	4	26	21	231
Precisão		85%	28%	48%	9%
Recuperação		100%	46%	91%	100%
F-Measure		92%	34%	63%	16%

Tabela 3. Resultados da detecção obtidos com o método TW2D com realce e com outros métodos de detecção de anomalias.

A Tabela 3 mostra que o algoritmo TW2D com realce de frequência detectou todos os ataques com uma quantidade mínima de falsos positivos. Assim, o método proposto obteve o melhor valor de F-Measure (92%). Nos demais métodos de detecção de anomalias, um conjunto de dados considerados normais é utilizado na etapa de treinamento utilizado pelos métodos. Para essas comparações os métodos 6BIN, MD e NGRAM foram pré-processados pela fase de treinamento utilizando a mesma quantidade de dados utilizado na etapa 1 da fase de realce de frequência, no qual foram usadas 1024 requisições.

O método NGRAM detectou todos os ataques, no entanto, com uma grande quantidade de falsos positivos (231), o que gerou o menor F-Measure. Foram realizadas variações com diferentes tamanhos de n-gram, sendo o tamanho de *n-gram* igual a 2 para os melhores resultados (detecção dos ataques com menor quantidade de falsos positivos). Percebe-se que este método requer um conjunto maior de dados normais para minimizar os falsos positivos em contraste com o método apresentado neste artigo, com uma quantidade de falsos positivos com o mínimo de dados para a etapa de realce de frequências. O método MD também detectou todos os ataques, com exceção para os que possuem repetição de um caractere normal (*CodeRed*). Também este método necessita de um conjunto de dados maior para a fase de treinamento. Por fim, o método 6BIN somente identificou os ataques que provocam anomalias, correspondendo a ataques do tipo *DirectoryTraversal* e *CodeRed*. Os ataques mais sutis e de menor frequência não puderam ser detectados por este método que também gerou uma quantidade considerável de falsos positivos.

Todos os experimentos e as técnicas foram implementados utilizando a linguagem de programação C sobre o sistema operacional Unix (distribuição Ubuntu Linux).

6. Conclusões

O presente artigo apresenta uma nova estratégia de detecção de anomalias no tráfego web. As anomalias são detectadas através de um algoritmo baseado na utilização da transformada wavelet de Haar bidimensional aplicado na análise das frequências realçadas dos caracteres utilizados nas requisições web. A decisão de identificar uma anomalia é baseada na operação de corte para a definição de um limiar (*threshold*) sobre os coeficientes wavelet. O valor de *threshold* utilizado se adapta aos dados analisados e não requer um fator de correção conforme a aplicação web analisada.

A introdução de uma fase prévia para realce de frequências permitiu que o algoritmo baseado em wavelet identificasse variações menos significativas em caracteres de uma requisição web, melhorando consideravelmente a eficácia do algoritmo TW2D. Como resultado, foi possível detectar ataques mais sutis, assim como aqueles que se encontram camuflados entre ataques com alta frequência ou grandes variações inerentes ao próprio tipo de requisição. Comparativamente, o método proposto obteve os melhores resultados em relação aos demais métodos utilizados para detecção de anomalias que observam também as características dos caracteres.

Como trabalhos futuros, pretende-se ampliar os testes aplicando o algoritmo a outras bases de dados, bem como aplicar o método proposto para identificações de outros tipos de ataques web.

Referências

Alvarez, G. and Petrovic, S. (2003). A new taxonomy of web attacks suitable for efficient encoding. *Computers & Security*, 22(5):435–449.

Bilen, C. and Huzurbazar, S. (2002). Wavelet-based detection of outliers in times series. *Journal of Computational and Graphical Statistics*, 11:311–327.

Cappo, C., Schaerer, C., Kozakevicius, A. d. J., Nunes, R. C., and Mozzaquatro, B. A. (2012). Comparison of different threshold values for a wavelet designed attack sensor. In *XXXIV Congresso Nac. de Matemática Aplicada e Computacional*, pages 360–366.

Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM Comput. Surv.*, 41:15:1–15:58.

Daubechies, I. (1992). *Ten lectures on wavelets*. SIAM, Philadelphia, PA, USA, 1 edition.

Donoho, D. L. and Johnstone, I. M. (1995). Adapting to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association*, 90(432):1200–1224.

Henke, M., Costa, C., dos Santos, E. M., and Souto, E. (2011). Detecção de intrusos usando conjunto de k-nn gerado por subespaços aleatórios. In *XI Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais*.

Huber, P. (1981). *Robust Statistics*. Wiley, New York.

Ingham, K. L. and Inoue, H. (2007). Comparing anomaly detection techniques for http. In *Proceedings of the 10th international conference on Recent advances in intrusion detection*, RAID'07, pages 42–62, Berlin, Heidelberg. Springer-Verlag.

Jamdagni, A., Tan, Z., Nanda, P., He, X., and Liu, R. P. (2010). Intrusion detection using gsad model for http traffic on web services. In *Proceedings of the 6th Internatio-*

nal Wireless Communications and Mobile Computing Conference, IWCMC '10, pages 1193–1197, New York, NY, USA. ACM.

Kiani, M., Clark, A., and Mohay, G. (2008). Evaluation of anomaly based character distribution models in the detection of sql injection attacks. In *Availability, Reliability and Security, 2008. ARES 08. Third International Conference on*, pages 47–55.

Kruegel, C., Valeur, F., and Vigna, G. (2004). *Intrusion Detection and Correlation Challenges and Solutions*. Springer-Verlag TELOS, Santa Clara, CA, USA, 1 edition.

Kruegel, C. and Vigna, G. (2003). Anomaly detection of web-based attacks. In *Proceedings of the 10th ACM Conference on Computer and communications security*, CCS '03, pages 251–261, New York, NY, USA. ACM.

Kruegel, C., Vigna, G., and Robertson, W. (2005). A multi-model approach to the detection of web-based attacks. *Computer Networks*, 48:717–738.

Mallat, S. (2009). *A wavelet tour of signal processing*. Elsevier/Academic Press, Amsterdam, third edition. The sparse way, With contributions from Gabriel Peyré.

Mamahlodi, M. (2006). What is the chi-square statistic? Connexions Web site. http://cnx.org/content/m13487/1.2/.

Marques, O. and Baillargeon, P. (2005). A multimedia traffic classification scheme for intrusion detection systems. In *Information Technology and Applications, 2005. ICITA 2005. Third International Conference on*, volume 2, pages 496–501.

Mozzaquatro, B. A., De Azevedo, R. P., Nunes, R. C., Kozakevicius, A. d. J., Schaerer, C., and Cappo, C. (2011). Anomaly-based techniques for web attacks detection. *Journal of Applied Computing Research (JACR)*, 2(2):111–120.

Northcutt, S. and Novak, J. (2002). *Network Intrusion Detection*. N.R. Pub., 3 edition.

OWASP (2013). The open web application security project - top 10 web application security risks. Disponível em `https://www.owasp.org/index.php/Category:OWASP\_Top\_Ten\_Project`. Acesso em: 02/06/2013.

Robertson, W., Vigna, G., Kruegel, C., and Kemmerer, R. (2006). Using generalization and characterization techniques in the anomaly-based detection of web attacks. In *ISOC Symposium on Networks and Distributed Systems Security*, San Diego, CA.

Robertson, W. K. (2009). *Detecting and Preventing Attacks Against Web Applications*. PhD thesis, University of California, Santa Barbara.

Stollnitz, E., DeRose, A., and Salesin, D. (1995). Wavelets for computer graphics a primer 1. *Computer Graphics and Applications, IEEE*, 15(3):76–84.

Su, Z. and Wassermann, G. (2006). The essence of command injection attacks in web applications. *SIGPLAN Not.*, 41:372–382.

Symantec (2013). Internet security threat report. Technical report. Acesso em: 02/06/2013.

Wang, K. and Stolfo, S. (2004). Anomalous payload-based network intrusion detection. In Jonsson, E., Valdes, A., and Almgren, M., editors, *Recent Advances in Intrusion Detection*, volume 3224 of *LNCS*, pages 203–222. Springer.

