# Detecção de Ataques Web usando Técnicas de Detecção de Anomalias

Bruno Augusti Mozzaquatro, Renato de Azevedo,
Raul Ceretta Nunes, Alice Kozakevicius
Universidade Federal de Santa Maria, UFSM
{brunomozza, renato.azevedo, ceretta, alicek}@inf.ufsm.br

Cristian Cappo, Christian Schaerer
Faculdade Politécnica da Universidade
Nacional de Assunção, FPUNA
Email: {ccappo, cschaer}@pol.una.py

***Resumo*—O crescente uso da Internet vem acompanhado de severas ameaças para a segurança das aplicações web. No entanto, os Sistemas de Detecção de Intrusão (IDS) têm sido usados para suprir a diversidade e a complexidade dos ataques web. Neste contexto o trabalho propõe um algoritmo para detecção de ataques web baseado na transformada *Wavelet* bidimensional. Este algoritmo explora as anomalias contidas nas frequências dos caracteres das requisições web. Os resultados experimentais mostram que nosso algoritmo obteve alta taxa de detecção sem a ocorrência de falsos positivos.**

## I. Introdução

A Internet está em constante progresso, de modo que se tornou um sistema global de interconexão entre as redes de computadores e utiliza aplicações web para servir bilhões de usuários. Essas aplicações vêm ganhando múltiplas funcionalidades e, infelizmente, surgem novas vulnerabilidades [1]. Paralelamente, diversas ameaças exploram essas vulnerabilidades com o intuito de violar a segurança ou afetar a disponibilidade das aplicações [2].

Neste contexto, os Sistemas de Detecção de Intrusão (IDS) são necessários para garantir a segurança da informação. Esses sistemas possuem duas abordagens: baseadas em assinaturas e baseadas em anomalias. IDS baseados em assinaturas identificam ataques através de um conjunto de informações contendo padrões de ataques definidos previamente [3]. IDS baseados em anomalias identificam ataques através da observação de variações nas características relacionadas a padrões de comportamento dos ataques [4].

A abordagem baseada em assinatura possui a vantagem de detectar ataques com baixa taxa de falsos positivos. No entanto, essa abordagem somente detecta ataques conhecidos e, para isso, necessita de frequentes atualizações no conjunto de assinaturas para manter a confiabilidade da detecção. A abordagem baseada em anomalias identifica padrões que não seguem a definição do comportamento normal, além de identificar novos ataques e suas variações sem assinaturas [4].

Ataques são aplicados em diferentes contextos, entre eles, ataques web que manipulam as requisições web têm recebido ampla atenção [1]. Esses ataques exploram as vulnerabilidades de manipulação de parâmetros para alterar os valores de uma requisição web [5]. Para isso são inseridos caracteres especiais, resultando em sérios problemas como disponibilidade, integridade, autenticação e autorização de serviços na Internet.

Neste contexto, alguns trabalhos exploram a detecção de anomalias contidas no tráfego web [6] [7] [8]. Para isso, foram usadas técnicas baseadas em *n-grams* [7], informações estatísticas [6], cadeias de *Markov* [8], entre outras. Técnicas de processamento de sinais foram aplicadas na detecção de anomalias em redes de computadores [9].

Este trabalho propõe uma abordagem para detecção de ataques web também baseado em técnicas de processamento de sinais, tais como as que utilizam funções *Wavelet* [10]. Motivado pela análise realizada anteriormente em [11], neste trabalho considera-se a aplicação da transformada *Wavelet* discreta bidimensional, que possibilita a exploração de variações na distribuição da frequência dos caracteres nas requisições web. O uso da transformada *Wavelet* bidimensional permite analisar em mais de uma direção as variações das frequências dos caracteres contidas nos ataques web. A análise via transformada Wavelet depende dos dados em questão, descartando a fase de treinamento usada por técnicas baseadas em aprendizagem [6] [12].

A organização dos tópicos é apresentada como segue: A seção II apresenta a transformada *Wavelet*. Na seção III é apresentada a abordagem proposta para detecção de ataques web usando *Wavelet* bidimensional. Os experimentos foram realizados na seção III-C, a fim de validar a proposta. Na seção IV os trabalhos relacionados são apresentados. E por fim, na seção V são apresentadas as considerações finais e as conclusões.

## II. Transformada *Wavelet*

A transformada *Wavelet* é utilizada para decompor um sinal em diferentes níveis de resolução com o objetivo de salientar e reconhecer informações mais relevantes que compõem o sinal. A transformada *Wavelet* permite a decomposição hierárquica de um sinal em uma representação grosseira e um conjunto de detalhes [10]. Os detalhes equivalem à informação complementar da representação grosseira e são necessários para a reconstrução dos dados originais.

Neste trabalho, a transformada *Wavelet* de *Haar* foi considerada devido à facilidade para a identificação de variações bruscas, assim como a simplicidade de implementação sem tratamento especial para as fronteiras, já que os sinais analisados são formados por conjuntos finitos de dados discretos. Além disso, a localização das variações significantes dos dados é melhor preservada pela aplicação da transformada *Wavelet* de *Haar* [13].

A aplicação da transformada *Wavelet* utiliza os filtros passa-alta (L) e passa-baixa (H) para obter os coeficientes Wavelet. A aplicação do algoritmo piramidal [13] entre os valores adjacentes possibilita obter dois conjuntos de coeficientes Wavelet: um conjunto das médias $c_{j-1,i}$ que contém os valores de baixa frequência; um conjunto dos detalhes $d_{j-1,i}$ que contém os valores de alta frequência. Considerando um vetor de dados de uma dimensão $c_{j,s}, s = 0, ..., M_j - 1$ no nível mais fino $j$, com $M_j = 2^J$ pontos:

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

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

A principal propriedade da *Wavelet* é a extrema sensibilidade de identificação de ruídos, saltos e picos nos dados do conjunto de detalhes [14]. Por isso, os detalhes vão desempenhar um papel importante para a identificação de qualquer variação abrupta. O algoritmo implementado neste trabalho é baseado nos algoritmos propostos por [10].

### A. Wavelet *bidimensional*

A transformada *Wavelet* bidimensional é usada para realizar a análise dos dados em duas dimensões, aplicando a transformada de uma dimensão nas linhas e depois nas colunas já com resultados da transformada nas linhas. Esse processo é repetido para cada nível de decomposição da transformada.

A primeira etapa da decomposição de um nível é a aplicação da transformada *Wavelet* 1D nas linhas, onde os dados originais são comprimidos nas médias e nos detalhes. Na segunda etapa o mesmo processo é realizado sobre as médias (c) e os detalhes (d), resultando em quatro tipos de dados (sub-bandas): (cc) representação grosseira dos dados; (cd) representação grosseira dos dados das linhas; (dc) detalhes da representação grosseira das linhas; (dd) detalhes dos dados.

A partir do primeiro nível de decomposição, somente as informações grosseiras (sub-banda cc) são usadas pela transformada *Wavelet* bidimensional, desconsiderando as sub-bandas de detalhes (sub-bandas cd, dc e dd). Ou seja, nos próximos níveis de decomposição a transformada é aplicada somente na sub-banda com a representação grosseira dos dados.

### B. Operação de Corte

A operação de corte, também conhecida como operação de truncamento (*Threshold*), define um critério para truncar os coeficientes *Wavelet* associados a informações irrelevantes. Esta operação é realizada através da comparação dos coeficientes das sub-bandas dc, cd e dd com um limiar de corte (valor de referência). Um dos valores mais utilizados em aplicações de filtragem é o valor de truncamento Universal proposto por *Donoho* [15], que é determinado por $\lambda = \sigma\sqrt{2\log(N)}$, sendo que $\sigma$ é o desvio padrão dos coeficientes e $N$ é o número de amostras e o tamanho de cada sub-banda é $\frac{N}{2}$ x $\frac{N}{2}$. Quando o limiar de corte é escolhido de forma coerente, a operação de truncamento não destrói as propriedades fundamentais do sinal [15]. Sendo assim, a questão toda de análise é a determinação de valores de corte que sejam capazes de separar o que é ou não associado à informação irrelevante, dependendo da aplicação em questão.

Neste trabalho, os coeficientes *Wavelet* são truncados de acordo com o critério de restrição *Hard thresholding*, apresentado na Equação 3.

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

Para uma explanação mais detalhada sobre outras maneiras de definição do valor de corte, assim como diferentes critérios para a escolha dos limiares de corte, ver *Donoho* [15].

## III. Algoritmo Proposto para Detecção de Ataques Web

Anomalias podem ocorrer na comunicação do tráfego web, provenientes de ataques web e causando sérios problemas para clientes e servidores web. O foco deste trabalho é a detecção de ataques web, que geram anomalias nas requisições enviadas pelos clientes aos servidores. Neste trabalho serão considerados ataques que geram perturbações nas requisições web.

Na seção III-B é apresentado o algoritmo para detecção de ataques web baseado na análise da frequência dos caracteres contidos nas requisições enviadas aos servidores web. Na seção III-A é apresentado o comportamento dos ataques que serão analisados, enquanto que na seção III-C, os experimentos e resultados são apresentados para validar o algoritmo proposto.

### A. Ataques Web

Segundo OWASP [1], vulnerabilidades de injeção de conteúdo estão no topo da lista de vulnerabilidades técnicas. Com isso, o número de ataques web que visam essas vulnerabilidades tende a aumentar causando ainda mais prejuízos para as organizações. Neste trabalho, somente os ataques web que inserem conteúdo malicioso, alterando a frequência dos caracteres das requisições web, serão considerados como anomalias (Figura 1).

```
121.112.21.23 " GET /index.php?v1=20112320&v2=admin HTTP/1.0" 200
154.127.15.33 " GET /index.php?v1=../../../../../../../etc/passwd%00 HTTP/1.0" 200
163.12.165.12 " GET /index.php?v1=20112523&v2=user HTTP/1.0" 200
```

Figura 1. Exemplo de ataque web usando injeção de conteúdo malicioso.

### B. Um algoritmo para Detecção de Ataques Web usando Wavelet *bidimensional*

As requisições web consideradas normais possuem uma estrutura regular com relação a distribuição da frequência dos caracteres, bem como a utilização de caracteres legíveis [6] e variações na estrutura regular podem indicar

eventos indesejáveis. Este comportamento motiva e justifica a utilização da transformada *Wavelet*, justamente por suas vantagens de detecção de variações. Este trabalho propõe um algoritmo para a análise da frequência dos caracteres utilizando *Wavelet* bidimensional.

Os dados de entrada do algoritmo para detecção de ataques web dependem da construção de uma matriz com as frequências dos caracteres contidos nas requisições. A matriz é definida por: $x[n][m] = \{v_0, v_1, v_2, ..., v_{m-1}\}$, onde $n$ é o valor dos caracteres da tabela ASCII. O $m$ é o tamanho do conjunto de requisições que serão analisadas.

**Algoritmo 1:** Abordagem proposta

```
   Entrada: M : Matriz de dados
   Saída  : A : Conjunto de posições dos ataques
1  A ← ∅
2  N ← |M|
3  (cc, dc, cd, dd) ← TW[M] [um nível]
4  {c̃d, d̃c, d̃d} ← Hard − Tresholding(cd, dc, dd, λ)
5  for i ← 1 to N/2 do
6      for j ← 1 to N/2 do
7          if na posição (i,j) pelo menos duas sub-bandas de detalhes
           contém informações relevantes. then
               Ataque detectado, então salva a posição.
8              A ⟵ A + (i, j)
9          end
10     end
11 end
12 return A
```

O Algoritmo 1 aplica a transformada *Wavelet* bidimensional na matriz gerada (Linha 2), assim como a operação de *Threshold* nas sub-bandas de detalhes, na linha 3, onde são selecionados os picos. Um pico na frequência de um caractere é considerado anômalo quando esse caractere possuir a frequência elevada com relação aos demais caracteres. A *Wavelet* bidimensional possibilita a verificação de picos na frequência entre os caracteres na mesma requisição (aplicação da *Wavelet* nas linhas) e entre as requisições analisadas (aplicação da *Wavelet* nas colunas), o que se considera uma vantagem na diminuição dos falsos positivos, como será apresentado na seção dos experimentos. Além disso, anomalia na frequência de um caractere deverá aparecer em pelo menos duas sub-bandas de detalhes para ser considerado um ataque (Linha 6). A operação de corte é responsável pela seleção dos picos, considerados anomalias em nossa análise.

A simplicidade e rapidez da transformada *Wavelet* caracteriza-se pelo uso de algoritmos computacionais simples [13]. Ainda, a complexidade algorítmica da aplicação da técnica é $O(nm)$, onde o número de operações do cálculo da transformada é linearmente proporcional à quantidade de dados contidos na matriz de entrada.

### C. Experimentos e Resultados

Na validação da abordagem proposta foi utilizado um conjunto de dados da Faculdade Politécnica da Universidade Nacional de Assunção (FPUNA), Paraguai. Esses dados contêm informações de uma aplicação hospedada no servidor web de produção, contendo 71 dias de dados do tráfego web coletado. O conjunto de dados FPUNA possui um total de 59.248 requisições web.

Para a validação da proposta os ataques contidos no conjunto de dados FPUNA foram retirados manualmente e inseridos outros ataques (com comportamento descritos na seção III-A) em pontos conhecidos. A Tabela I apresenta os resultados da detecção com a aplicação do algoritmo proposto.

Tabela I
RESULTADOS ALCANÇADOS COM A ABORDAGEM PROPOSTA.

| | # Ataques detectados com diferentes $\rho\lambda$ | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | 1λ | | 2λ | | 3λ | | 4λ | | 5λ | |
| Ataques | FP | VP | FP | VP | FP | VP | FP | VP | FP | VP |
| 0 | 15971 | 0 | 1939 | 0 | 7 | 0 | 1 | 0 | 0 | 0 |
| 1 | 15970 | 1 | 1934 | 1 | 8 | 1 | 2 | 1 | 0 | 1 |
| 2 | 15969 | 2 | 1934 | 2 | 7 | 2 | 1 | 2 | 0 | 2 |
| 3 | 15911 | 3 | 1934 | 3 | 7 | 2 | 1 | 3 | 0 | 3 |
| 4 | 15909 | 4 | 1934 | 4 | 7 | 4 | 1 | 4 | 0 | 4 |
| 5 | 15887 | 5 | 1874 | 5 | 3 | 5 | 1 | 5 | 0 | 5 |
| 10 | 15774 | 10 | 1940 | 10 | 7 | 10 | 1 | 10 | 0 | 10 |
| 20 | 15420 | 20 | 1876 | 20 | 3 | 20 | 1 | 20 | 0 | 20 |

FP: Falso Positivo VP: Verdadeiro Positivo

É importante salientar que a determinação do valor de corte é o ponto nevrálgico em análises via transformadas wavelet. Neste trabalho o valor de truncamento Universal de *Donoho* foi tomado como um dos possíveis valores de referência para o limiar de corte, pois se trata de uma estimativa inicialmente definida assumindo-se a presença de ruídos brancos gaussiano nos dados. Assim, para outras aplicações web a análise e determinação do valor de corte se fazem necessária. Neste trabalho, o valor de corte utilizado foi cinco vezes o valor de truncamento Universal, com o qual todos os ataques inseridos foram detectados sem a ocorrência de falsos positivos.

A Tabela II apresenta a comparação do algoritmo proposto com o teste *Person* $\chi^2$, que é frequentemente usado pelas técnicas de detecção [6] [12] com o mesmo propósito deste trabalho.

Tabela II
COMPARAÇÃO DO ALGORITMO COM O TESTE *Person* $\chi^2$

| | # Ataques inseridos na base de dados FPUNA | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Técnica | 0 | | 1 | | 2 | | 3 | | 4 | | 5 | | 10 | | 20 | |
| | FP | VP | FP | VP | FP | VP | FP | VP | FP | VP | FP | VP | FP | VP | FP | VP |
| $\chi^2$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 2 | 0 | 2 |
| (*) | 0 | 0 | 0 | 1 | 0 | 2 | 0 | 3 | 0 | 4 | 0 | 5 | 0 | 10 | 0 | 20 |

(*) Algoritmo proposto FP: Falso Positivo VP: Verdadeiro Positivo

O teste *Person* $\chi^2$ utilizou uma fase de treinamento e o valor de corte sendo 10% somado ao valor máximo obtido na fase de treinamento, enquanto que o algoritmo proposto utilizou o valor de corte $5\lambda$. Os resultados obtidos na comparação do algoritmo proposto com o teste *Person* $\chi^2$ foram satisfatórios.

## IV. TRABALHOS RELACIONADOS

O trabalho de Kruegel [6] propôs um modelo de distribuição de frequências dos caracteres, conhecido como *Frequency Character Distribution* (FCD), para análise das requisições web para a identificação de ataques web. O modelo utiliza a abordagem baseada em aprendizagem, que possui duas fases: treinamento e detecção. Na fase de treinamento, requisições sem ataques são usadas para obter a distribuição de caracteres ideal, conhecida como *Idealized Character Distribution* (ICD). Desta forma, as

frequências relativas dos caracteres são calculadas e agrupadas com ordenação decrescente em seis grupos. Na fase de detecção, o teste *Person* $\chi^2$ é usado para calcular a probabilidade das frequências das requisições observadas com a ICD. Por fim, o valor de truncamento definido para a identificação de anomalias é obtido com a maior pontuação da fase de treinamento somado a 10% do valor.

No trabalho de Kiani [12] é proposto outro modelo de distribuição de frequências dos caracteres para detectar ataques web, conhecido como *Same Character Comparison* (SCC). O modelo SCC intercepta as requisições web para realizar a análise e utiliza uma fase de treinamento para construir a ICD que será usada na fase de detecção, assim como é realizado pelo modelo FCD. A contabilização das frequências é cumulativa e os agrupamentos utilizados devem conter o valor mínimo igual a cinco para as frequências de cada grupo. Esta restrição é imposta pela correção de Yate's, usada na fase de detecção quando é realizada a comparação dos valores observados com os valores esperados pelo teste *Person* $\chi^2$. A definição do valor de truncamento no modelo SCC ocorre na fase de treinamento e somente é considerado se não possui a ocorrência de falsos positivos, permitindo a definição de um valor de truncamento confiável.

O trabalho de Rajagopal [16] propôs uma melhoria para o modelo FCD [6]. Ao contrário do modelo FCD, na fase de treinamento eles agruparam as frequências relativas dos caracteres alfabéticos, numéricos e especiais em grupos separados. Ao invés de usar seis grupos usados no modelo FCD, eles usaram três grupos para o agrupamento. A fase de detecção é realizada da mesma forma que o trabalho de Kruegel. A mudança na quantidade de grupos permitiu melhores resultados na detecção de ataques.

O algoritmo proposto neste trabalho difere dos trabalhos relacionados por não utilizar uma fase de treinamento, a qual é considerada um desafio na área de detecção de intrusão [3]. Além disso, o algoritmo não utiliza nenhum tipo de agrupamento das frequências dos caracteres.

## V. Conclusões

O presente trabalho apresentou uma abordagem de detecção de ataques web utilizando técnicas de processamento de sinais. Os experimentos realizados demonstraram que a utilização da transformada *Wavelet* bidimensional permitiu a detecção dos ataques web que causam perturbação na distribuição da frequência dos caracteres. Além disso, a taxa de falsos positivos foi nula, quando associado com cinco vezes o valor de truncamento.

Através das propriedades da transformada *Wavelet* foi possível analisar as variações nas frequências dos caracteres em diferentes direções. Como resultado, tem-se um método rápido, pois não considera a fase de treinamento e obteve uma alta taxa de detecção dos ataques web.

## Referências

[1] OWASP, "Top 10 web application security risks," 2010, the Open Web Application Security Project.

[2] J. Fonseca, M. Vieira, and H. Madeira, "The web attacker perspective - a field study," in *Software Reliability Engineering (ISSRE), 2010 IEEE 21st International Symposium on*, nov. 2010, pp. 299–308.

[3] C. Kruegel, F. Valeur, and G. Vigna, *Intrusion Detection and Correlation: Challenges and Solutions*. Santa Clara, CA, USA: Springer-Verlag TELOS, 2004.

[4] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, 2009.

[5] G. Álvarez and S. Petrovic, "A new taxonomy of web attacks suitable for efficient encoding," *Computers & Security*, vol. 22, no. 5, pp. 435–449, 2003.

[6] C. Kruegel and G. Vigna, "Anomaly detection of web-based attacks," in *Proceedings of the 10th ACM Conference on Computer and communications security*, ser. CCS '03. New York, NY, USA: ACM, 2003, pp. 251–261.

[7] K. Wang and S. Stolfo, "Anomalous payload-based network intrusion detection," in *Recent Advances in Intrusion Detection*, ser. Lecture Notes in Computer Science, E. Jonsson, A. Valdes, and M. Almgren, Eds. Springer Berlin / Heidelberg, 2004, vol. 3224, pp. 203–222.

[8] J. M. Estevez-Tapiador and J. E. Diaz-Verdejo, "Detection of web-based attacks through markovian protocol parsing," in *ISCC '05: Proceedings of the 10th IEEE Symposium on Computers and Communications*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 457–462.

[9] W. Lu and A. A. Ghorbani, "Network anomaly detection based on wavelet analysis," *EURASIP J. Adv. Signal Process*, vol. 2009, pp. 4:1–4:16, January 2009.

[10] E. Stollnitz, A. DeRose, and D. Salesin, "Wavelets for computer graphics: a primer 1," *Computer Graphics and Applications, IEEE*, vol. 15, no. 3, pp. 76–84, May 1995.

[11] C. Cappo, R. C. Nunes, and C. Schaerer, "On using wavelets for detecting attacks to web-based applications," in *XXXII Congresso Nacional de Matemática Aplicada e Computacional - CNMAC 2009*, set. 2009, pp. 1040–1041.

[12] M. Kiani, A. Clark, and G. Mohay, "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*, Mar 2008, pp. 47–55.

[13] S. Mallat, *A wavelet tour of signal processing*, 3rd ed. Elsevier/Academic Press, Amsterdam, 2009, the sparse way, With contributions from Gabriel Peyré.

[14] A. Grane and H. Veiga, "Wavelet-based detection of outliers in volatility models," Universidad Carlos III, Departamento de Estadística y Econometría, Statistics and Econometrics Working Papers, 2009.

[15] D. L. Donoho and I. M. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," *Journal of the American Statistical Association*, pp. 1200–1224, 1995.

[16] R. Sriraghavan and L. Lucchese, "Data processing and anomaly detection in web-based applications," in *Machine Learning for Signal Processing, 2008. MLSP 2008. IEEE Workshop on*, oct. 2008, pp. 187–192.