Investigação sobre Técnicas de Detecção de Intrusões em Redes de Computadores com base nos Algoritmos Knn e K-Means

Mauricio Mendes Faria[1, 2], Ana Maria Monteiro[1]

[1]FACCAMP – Rua Guatemala, 167, 13231-230, Campo Limpo Paulista, Brasil
[2]Universidade Anhembi Morumbi - Rua Casa do Ator, 04546-001, 275, São Paulo

mauriciofaria@gmail.com, anammont@cc.faccamp.br

Abstract. *Today many applications of large organizations generate petabytes of data that are stored to be processed someday to produce information for decision-making. Data from logs of these applications are also generated in exponential order and within that mass of data is possible to detect if the applications are undergoing some instability due to malicious users. This work proposes a performance comparison between the algorithms families KNN (k Nearest Neighbors) based on instances and K-Means clustering based on collations, in intrusion detection using data from the logs of computer networks.*

Resumo. *Hoje em dia inúmeras aplicações de grandes organizações geram petabaytes de dados que são armazenados na intenção de que algum dia possam ser processados para produzirem informações para a tomada de decisões. Dados provenientes de logs dessas aplicações também são gerados em ordem exponencial e dentro dessa massa de dados é possível detectar se as aplicações estão passando por alguma instabilidade por conta de usuários mal-intencionados. Este trabalho propõe um comparativo de desempenho entre algoritmos das famílias KNN (K Nearest Neighbors) com base em instâncias e K-Means (K-Médias) com base em agrupamentos, na detecção de intrusões utilizando os dados dos logs de redes de computadores.*

1. Introdução

Atualmente enormes massas de dados são geradas através de diversos tipos de aplicações tais como as bancárias, médicas, tecnológicas, ambientais e de comércio eletrônico, envolvendo diversos tipos de arquiteturas como desktops, móveis ou a web. Isso acarreta às organizações grandes preocupações no tocante à segurança da informação.

O gerenciamento desses dados torna-se complexo quando se verifica a possibilidade de acessos acontecerem de forma local ou remota. Por isso grandes organizações são desafiadas todos os dias a manterem os dados de forma totalmente segura. Esses dados vão desde uma simples identificação como o CPF ou CNPJ bem como dados de localização geográfica, períodos entre compras, movimentações financeiras, investimentos, números de cartões de créditos, valores de investimentos, formas de pagamentos de faturas, etc.

Aplicações em diferentes arquiteturas computacionais produzem massas de dados em ordem exponencial e requerem processos eficientes de descoberta do conhecimento para que possam ser usadas em benefício da organização que os detém. Grande parte desses dados são provenientes dos logs de acessos das aplicações. Informações como

logins de usuários, hosts, IPs (*Internet Protocol),* portas de acesso, tipos de protocolos de acesso, data e hora de acesso devem ser armazenados em grandes quantidades nesses arquivos de logs. Também podem ser considerados como o "*calcanhar de Aquiles"* das aplicações, pois proporcionam a um indivíduo mal-intencionado oportunidades de obter informações valiosas com os dados que representam as rotinas diárias dos usuários dos sistemas. Através dos logs também é possível detectar se o sistema está passando por alguma instabilidade por conta de tentativas de invasão. Para investigar essas fragilidades, as organizações apontam a necessidade de investir em um *Intrusion Detection System* (IDS) para que ações de prevenção sejam feitas.

Um IDS monitora e analisa o tráfego da rede, utilizando múltiplos sensores para detectar intrusões de redes externas e internas. Um IDS analisa a informação coletada pelos sensores e retorna uma síntese da entrada desses sensores para o administrador do sistema ou para o sistema de prevenção de intrusão. Inúmeros algoritmos podem ser utilizados para a detecção dessas intrusões ou anomalias no uso do sistema, Entre os algoritmos utilizados podem ser mencionados os classificadores Bayesianos, árvores de decisão, baseados em regras, os da família KNN baseados em proximidade bem como os da família K-Means que usam técnicas baseadas em particionamento de grupos e foram estes dois últimos os escolhidos para este estudo.

2. A Problemática na Análise e Detecção Anomalias

Segundo Han & Kamber (2006), os bancos de dados são ricos em informações ocultas que podem ser utilizadas para a tomada de decisão. O uso dessas informações necessita de algumas formas de análise que permitam a extração de modelos que descrevem as classes de dados importantes ou que permitem a previsão de dados futuros

Conforme estabelecido por Han & Kamber (2006), a classificação consiste em construir um modelo que possa ser aplicado a dados não classificados visando categorizá-los em classes. A tarefa de predição é similar à tarefa de classificação, porém ela visa descobrir os valores futuros de um determinado atributo. Alguns métodos de classificação e regressão podem ser usados para predição, com as devidas considerações. A análise por regressão é um método estatístico que é usado mais frequentemente para previsão numérica, portanto, os dois termos são frequentemente usados como sinônimos.

2.1. Análise de Anomalias

Chandola (2009), ressalta que as anomalias podem ser consideradas ruídos nos dados. A remoção de ruído é impulsionada pela necessidade de remover os objetos indesejados antes de qualquer análise seja realizada sobre os dados. Porém esses ruídos também podem ser dados importantes que podem denunciar padrões fora do normal. A distinção entre os novos padrões e as anomalias é que os novos padrões são tipicamente incorporados no modelo normal depois de serem detectados. A Figura 1 ilustra as anomalias em um conjunto de dados em 2 dimensões.

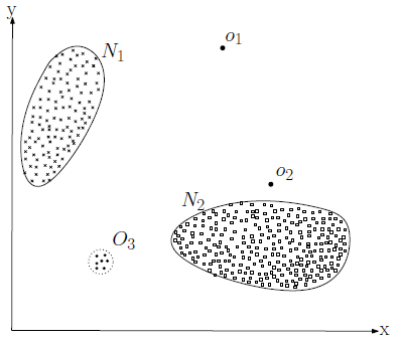

Figura 1 - Um simples exemplo de anomalias em um conjunto de dados de 2 dimensões. Fonte: Varun Chandola, (2009)

Ao analisar a Figura 1 notam-se duas regiões normais, *N1* e *N2*, uma vez que a maioria das observações se encontra nessas duas regiões. Alguns pontos estão distantes destas regiões, o_1 e o_2, e pontos na região O_3, são anomalias. Anomalias podem ser induzidas nos dados por uma variedade de razões, tais como atividades maliciosas, por exemplo, fraude de cartão de crédito, cyber-intrusão, atividade terrorista ou avaria de um sistema, mas todas essas razões tem uma característica comum, o interesse para o analista [Chandola, 2009].

2.2. Detecção de Intrusão

Segundo Siddiqui (2000), a detecção de intrusão é o processo de determinar uma invasão num sistema pela observação das informações disponíveis sobre o estado do sistema e monitorar as atividades dos usuários. Jones & Sielke (2000), afirmam que os intrusos podem ser entidades de fora ou usuários de dentro do sistema tentando acessar informações não autorizadas. Com base nessas observações os intrusos podem ser amplamente divididos em duas categorias, Intrusos externos (A) e Intrusos internos (B). (A) Intrusos externos são aqueles que não têm um acesso autorizado ao sistema com o que estão lidando. (B) Intrusos internos são aqueles que têm acesso autorizado a apenas uma parte do sistema e eles ultrapassam seus direitos de acesso originais legítimos. Os usuários internos podem ser divididos em: (I) mascarados que são aqueles que usam a identificação e autorização de outros usuários legítimos. (II) clandestinos que são aqueles que fogem com êxito das medidas de auditoria e monitoramento.

Quando é feita uma análise de uma conexão temos uma sequência de pacotes TCP *(Transfer Control Protocol)* iniciando e finalizando em algum tempo bem definido com dados fluindo do endereço IP *(Internet Protocol)* fonte para o IP destino sob algum protocolo bem definido. A cada conexão pode ser atribuído um rótulo de normal ou de ataque. Os ataques, por sua vez, podem ser rotulados nas seguintes categorias: (1) DoS *(Denied of Service),* como por exemplo, Syn Flood *(Inundação de pedidos de Sincronização).* (2) R2L *(Remote to Local Attack):* acesso não autorizado a partir de uma máquina remota, por exemplo, guessing password *(Adivinhação de Senha)*. (3) U2R *(User to Root)*: acesso não autorizado com privilégios de super usuário, por exemplo, buffer overflow *(Transbordamento de Buffer).* (4) Probing *(Sondagem)*: monitoramento e método de tentativa e erro, por exemplo o Port Scanning (escaneamento de portas). Lembrando que as categorias mencionadas acima não são necessariamente intrusões, mas podem evidenciar a fragilização do servidor em uma sequência para uma posterior

tentativa de intrusão. A detecção de intrusão é a segunda linha de defesa de uma rede e ela é feita através dos IDS.

O objetivo deste estudo é propor preliminarmente um teste comparativo do comportamento dos algoritmos KNN e K-Means como motores dos IDS na detecção de intrusões em redes de computadores. O teste abordará o aspecto de desempenho (acuracidade e velocidade), dos algoritmos, na detecção de intrusões.

3. Algoritmos na detecção de intrusão

A detecção de intrusão em redes de computadores é uma área de estudo bastante ativa e várias técnicas são usadas. A classificação das técnicas de detecção de anomalias de rede presentes na literatura é uma tarefa difícil devido à diversidade e ao desenvolvimento constante de novas técnicas. Perlin (2011), em seu trabalho classificou os métodos de detecção de anomalias de rede em métodos baseados em Conhecimento, Aprendizagem de Máquina e Análise Estatística. (A) Conhecimento: Máquina de Estados Finitos; Sistemas especialistas ou baseado em regras; Busca por Padrões *(Pattern Matching)*. (B) Aprendizagem de Máquina: Redes Bayesianas; Cadeias de Markov; Redes Neurais; Lógica Difusa *(Fuzzy Logic)*; Algoritmos Genéticos; Algoritmos de Agrupamento *(Clustering)*; Sistemas Imunológicos Artificiais. (C) Análise de Sinais: Análise Estatística; Filtros de Kalman; CUSUM *(CUmulative SUM)*; Séries Temporais; Wavelets. A escolha dos algoritmos K-Means e KNN para serem comparados surge do fato de que o primeiro tem sido utilizado em várias pesquisas mas o último não tem sido muito usado nos motores de IDS, deixando uma lacuna para pesquisa que versará sobre o comportamento dos dois algoritmos.

3.1. K-Nearest Neighbors (KNN)

O KNN é um classificador onde o aprendizado é baseado na analogia. O conjunto de treinamento é formado por vetores n-dimensionais e cada elemento deste conjunto representa um ponto no espaço n-dimensional. Para determinar a classe de um elemento que não pertença ao conjunto de treinamento, o classificador KNN procura *k* elementos do conjunto de treinamento que estejam mais próximos deste elemento desconhecido, ou seja, que tenham a menor distância. Estes *k* elementos são chamados de k-vizinhos mais próximos. Verifica-se quais são as classes desses *k* vizinhos e a classe mais frequente será atribuída à classe do elemento desconhecido.

KNN é um classificador que possui apenas um parâmetro livre (o número de k-vizinhos) que é controlado pelo usuário com o objetivo de obter uma melhor classificação [Silva, 2015].

3.2. K-Means (k-médias)

O algoritmo K-Means é uma heurística de agrupamento não hierárquico que busca minimizar a distância dos elementos a um conjunto de *k* centros $\boldsymbol{\chi} = \{x_1, x_2, \ldots, x_k\}$ de forma iterativa. A distância entre um ponto p_i e um conjunto de clusters é definida como sendo a distância do ponto ao centro mais próximo dele. A função a ser minimizada então, é dada pela equação: $\boldsymbol{d}(\boldsymbol{P}, \boldsymbol{\chi}) = \frac{\mathbf{1}}{\boldsymbol{n}} \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{n}} \boldsymbol{d}\,(\boldsymbol{p_i}, \boldsymbol{\chi})^{\mathbf{2}}$

O algoritmo depende de um parâmetro (*k*=número de clusters) definido de forma *ad hoc* pelo usuário. Isto costuma ser um problema, tendo em vista que normalmente não

se sabe quantos clusters existem a priori. O algoritmo K-Means pode ser descrito conforme os seguintes passos: (1) Escolher *k* pontos diferentes para centros dos grupos (possivelmente, de forma aleatória). (2) Associar cada ponto ao centro mais próximo. (3) Recalcular o centro de cada grupo. (4) Repetir os passos 2-3 até nenhum elemento mudar de grupo. Este algoritmo é extremamente veloz, geralmente converge em poucas iterações para uma configuração estável, na qual nenhum elemento está designado para um cluster cujo centro não lhe seja o mais próximo [Linden, 2009].

5. Metodologia

O comparativo de desempenho dos algoritmos será elaborado através do processamento de uma massa de dados denominada *KDDCUP99*[*], no formato ARFF *(Attribute-Relation File Format)* para o processamento na ferramenta WEKA (*Waikato Environment for Knowledge Analysis*). Através dos testes serão estabelecidas tabelas e gráficos comparativos que permitirão apontar as vantagens e desvantagens dos dois algoritmos para a tarefa estabelecida.

6. Considerações Finais

A detecção de anomalias é um problema que é pesquisado em diferentes áreas da Computação, a Estatística e a Matemática com aplicação em domínios que vão da Medicina até a questões de Segurança Militar. Muitas técnicas foram propostas para resolver o problema, algumas das quais são específicas para domínios particulares.

O que foi apresentado neste artigo é parte de um trabalho preliminar que está sendo realizado para analisar as técnicas de detecção de anomalias no contexto da detecção de intrusões em redes de computadores.

Referências

Han, L. & Kamber, M., 2006. "Data mining concepts and techniques". 2ª ed. São Francisco: Morgan Kaufmann & Elsevier.

Linden, R., 2009. "Técnicas de agrupamento". Revista de sistemas de informação da FSMA", p. 18-36.

Perlin, T., Nunes, R. C. & Kozakevicius, A. d. J., 2011. "Detecção de Anomalias em Redes de Computadores através de Transformadas Wavelet". Revista Brasileira de Computação Aplicada, 3(1), pp. 2-15.

Silva, L. M. O. D., 2015. "Uma aplicação de árvores de decisão, redes neurais e KNN para a identificação de modelos arma não-sazonais e sazonais", (Online) disponível em: http://www.maxwell.vrac.puc-rio.br/7587/7587_6.PDF, Acesso em 13: 07: 2015.

Varun Chandola, A. B. V. K., 2009. "Anomaly detection: A Survey", ACM Computing Surveys, 07, p. 15-49.

[*] Data set usado na The Third International Knowledge Discovery and Data Mining Tools Competition (KDD-99). A competição consistia em implementar um detector de instrusões