ESTUDO SOBRE SISTEMA DE DETECÇÃO DE INTRUSÃO POR ANOMALIAS

UMA ABORDAGEM UTILIZANDO REDES NEURAIS

Eusam P. de Souza, José Augusto Suruagy Monteiro

Núcleo de Pesquisa em Redes e Computação – Universidade Salvador (UNIFACS)
R. Ponciano de Oliveira, 126 – Rio Vermelho – 41.950-275 – Salvador – BA
eusampereira@yahoo.com.br, suruagy@unifacs.br

Abstract. *The main goal of an intrusion detection system (IDS) is to be able to achieve a high hit and a low false alarm rates. Anomaly based IDS using techniques that seek to identify differences based on comparison of traffic patterns deemed normal, with anomalous patterns. Using the generalization ability of neural networks, it was possible to make the classification of attacks not yet known by the neural network, during its training stage. The results obtained in this work were compared with those from other methods of classification, to assess the effectiveness of the method. Then, it was submitted to the neural network only the connection record characteristics considered relevant. In this way, the neural network had its size reduced, reducing the attacks classification time without compromising its hit rate.*

Words key: IDS, Artificial Neural Nets, Traffic Anomalies.

Resumo. *O principal objetivo de um sistema de detecção de intrusão é ser capaz de alcançar uma alta taxa de acertos e uma baixa taxa de alarmes falsos. IDS por anomalias utiliza técnicas que procuram identificar diferenças baseadas na comparação de padrões de tráfego considerados normais, com padrões anômalos. Utilizando a capacidade de generalização das redes neurais, foi possível realizar a classificação de ataques ainda não conhecidos pela Rede Neural, durante a etapa de treinamento. Foram feitas comparações entre os resultados obtidos neste trabalho, com os obtidos por outros métodos de classificação, para avaliar a eficácia do método. Em seguida, foram apresentadas à Rede Neural apenas as características consideradas relevantes dos registros de conexões, com o objetivo de reduzir o tempo de classificação da Rede. Com isso, a Rede Neural teve seu tamanho reduzido, diminuindo o tempo de classificação dos ataques, mas sem comprometer sua taxa de acertos.*

Palavras chave: IDS, Redes Neurais Artificiais, Anomalias de tráfego.

1. INTRODUÇÃO

Os sistemas detectores de intrusão têm por finalidade oferecer um mecanismo que venha a reduzir a possibilidade de intrusão, através da antecipação e acompanhamento dos ataques, bem com auxiliar a compreensão das estratégias utilizadas pelos atacantes e códigos de computador maliciosos. Heinen e outros (2006).

As técnicas de detecção de intrusões pressupõem que o comportamento e as atividades de um usuário, ou programa, legítimo são diferentes do comportamento e das atividades de um invasor. Elas funcionam com base na análise e classificação dos comportamentos e atividades de um usuário como legítimas ou ilegítimas. Ou seja, essas técnicas também assumem que o comportamento e as atividades de um usuário são observáveis e possíveis de serem analisadas.

1.1. ATAQUES E INTRUSÕES

Ataques e intrusões podem ser classificados em quatro categorias distintas: negação de serviço (denial of service), remoto para usuário (R2L), usuário para superusuário (U2R) e reconhecimento (probing).

Negação de serviço é uma classe de ataques contra a disponibilidade de sistemas computacionais. Ataques da classe usuário para superusuário (User to Root) englobam todos os ataques em que o intruso possui acesso ao sistema como um usuário normal e consegue elevar seu nível de privilégio para o de um usuário especial (como o usuário root em sistemas Unix ou administrador em outras plataformas).

Ataques da classe remoto para local (Remote to Local) correspondem a situações em que o intruso possui conectividade com a máquina vítima, sem possuir uma conta de usuário e explorando alguma vulnerabilidade existente, consegue obter acesso local à máquina. Ataques da classe reconhecimento (Probing) são normalmente empregados em uma etapa que antecede o ataque ou intrusão. Do ponto de vista dos sistemas de detecção de intrusão, detectar atividade relacionada a esta categoria é importante uma vez que permite uma preparação e resposta adequada à próxima etapa que provavelmente será iniciada pelo intruso.

Este trabalho está organizado da seguinte forma: na seção 2 é apresentada uma introdução teórica sobre IDS e tipos de IDS; a seção 3 mostra conceitos teóricos relacionados a Redes Neurais Artificiais; a seção 4 apresenta a utilização de Redes Neurais em IDS; a seção 5 aborda a proposta de solução para um Sistema de Detecção de Intrusão baseados em Redes Neurais, para redes TCP/IP, os resultados obtidos e análises sobre tais resultados, assim como a utilização apenas das características consideradas relevantes dos registros de conexões, com base nos resultados do trabalho de [14]; na seção 6 são apresentadas as conclusões e apontados os aspectos que poderão ser tratados em trabalhos futuros.

2. SISTEMAS DE DETECÇÃO DE INTRUSÃO

Os Sistemas de Detecção de Intrusão podem ser classificados quanto à técnica de detecção.

2.1. CLASSIFICAÇÃO DE IDS QUANTO ÀS TÉCNICAS DE DETECÇÃO

A detecção por abuso, por uso indevido ou por mau uso (misuse detection), tem o objetivo de comparar eventos do sistema em tempo real a cenários de ataques generalizados, ou seja, a modelos de detecção de abusos.

A principal desvantagem dos sistemas de detecção de intrusos por abuso é a sua incapacidade de detectar novos ataques, ou mesmo, ataques que ainda não façam parte da sua base de assinaturas e padrões.

A detecção por anomalia ou por comportamento (anomaly detection), diferente da detecção por abuso, compara os eventos correntes do sistema ou do tráfego de rede a perfis de atividades normais, ou seja, aos modelos de detecção por anomalia. Nesta abordagem, as anomalias são consideradas como possíveis intrusões.

Sistemas de detecção de intrusão por anomalia procuram determinar ou criar modelos que representem o comportamento normal ou esperado do sistema computacional ou rede em análise e alertam sempre que desvios neste comportamento esperado forem encontrados. A premissa básica destes sistemas é que atividades de intrusão ou ataque fazem parte do subconjunto composto por atividades anômalas.

A abordagem de detecção por anomalia apresenta como primeira grande vantagem, a capacidade de detectar novos tipos de ataques, visto que quaisquer novos ataques diferem de comportamento normal.

A detecção por anomalias também apresenta problemas. A principal limitação é a ocorrência de alarmes falsos. Isto ocorre porque nem toda atividade não usual é ilegítima ou representa um ataque.

2.2. TRABALHOS RECENTES EM IDS

O trabalho de Mafra e outros (2008) apresenta um modelo de sistema de detecção de intrusão que classifica o tráfego de rede por análise comportamental como normal ou anômalo. Para detecção de anomalias são utilizadas duas técnicas de inteligência artificiais chamadas Support Vector Machine (SVM) e Redes Neurais de Kohonen (KNN).

Mafra e outros (2008) desenvolveram um sistema multicamadas chamado POLVO-IIDS, utilizando as Redes Neurais de Kohonen para classificar dados de forma genérica (comportamentos normal e anômalo), e Redes Neurais do tipo Support Vector Machine (SVM), sendo que para cada classe de ataque (DoS, Worm, Scan ou Normal) existe uma SVM especializada para detectar o tipo correspondente de ataque. Portanto, terá duas opções como saída: tráfego normal ou atividade maliciosa. O uso de Redes Reurais com aplicações mais especializadas, segundo Mafra e outros (2008), teve como objetivo obter-se maior precisão, pois cada SVM foi treinada para separar os dados em apenas duas classes.

Assim, no trabalho supracitado, o classificador faz uma pré-seleção do tráfego de entrada, através da análise de características contidas nos pacotes em um determinado período de tempo. Para minimizar a taxa de falsos positivos, foi utilizada uma outra Rede Neural na saída do classificador, rede essa com função mais especializada, pois é responsável por analisar determinado tipo de ataque e identificar com mais precisão o que é anomalia e o que é tráfego normal. Dessa forma, pretendeu-se reduzir a taxa de falsos positivos dos IDSs convencionais e melhorar a taxa de acerto. Para a realização de testes com o modelo apresentado, foi usado o tráfego KDD Cup (1999). Data disponível na Internet.

No trabalho de Haijun e outros (2007), foi elaborado um estudo comparativo entre o uso de SVM (support vector machine), redes neurais e outras técnicas de mineração de dados para a detecção de intrusão. Nos testes realizados, a taxa de detecção de intrusão obtida com SVM e redes neurais não apresentaram diferenças significativas entre si, mas essas duas técnicas mostraram-se superiores às demais utilizadas.

Em Cândido Júnior e outros (2005) apresentaram as Redes Neurais Artificiais aplicadas ao reconhecimento de padrões obtidos em fluxos de dados para detecção de ataques e anomalias. A ferramenta Neuro-sig foi desenvolvida em ambiente UNIX com o intuito de gerar padrões para a rede neural, a partir de uma base de dados de registros de conexões. Assim, a ferramenta Neuro-sig foi desenvolvida com o propósito de auxiliar a geração de padrões para treinamento de redes neurais. A rede neural foi treinada a partir das informações fornecidas no arquivo de padrões, estando pronta para a classificação de ataques.

Os fluxos de dados foram coletados em uma instituição acadêmica. No treinamento de cada modelo foram utilizados fluxos obtidos em um período de duas semanas de tráfego. Adicionalmente, tráfego de uma semana foi utilizado para a validação da Rede Neural (Cândido Junior et al., 2005).

Em Cândido Junior e outros (2005), optou-se por simular ataques de negativa de serviço. Em ataques dessa natureza é observado um número excessivamente grande de fluxos de dados por segundo. O evento “tráfego desconhecido” foi criado para

classificação de um evento anômalo de difícil classificação. Esse evento é semelhante ao tráfego lícito, ou seja, não demanda nenhum tipo de ação quando detectado. Estes testes mostraram que Redes Neurais podem ser utilizadas com sucesso na detecção de ataques e anomalias em redes de computadores com um baixo índice de erros.

Dalmazo e outros (2008) classificaram os sistemas de detecção de intrusão em três componentes fundamentais: fonte de informação, análise e resposta. A fonte de informação foi representada por um coletor associado a um host, rede ou segmento de rede. A análise, como a parte do SDI que verifica os eventos derivados da fonte de informações, determinando quando estes eventos indicam que uma intrusão está ocorrendo ou já ocorreu. E a resposta foi representada como o conjunto de ações que o IDS realiza quando detecta uma intrusão, por exemplo, a intervenção automatizada ou a geração de alertas e relatórios para a interpretação e intervenção humana.

Dalmazo e outros (2008) propuseram a utilização da teoria de séries temporais na fase de análise de um IDS, objetivando de identificar com maior confiança uma intrusão e diminuir o número de falsos positivos. Uma série temporal é um modelo matemático para representar amostragens periódicas que apresentam dependência entre as amostras.

Como resultado preliminar, os testes realizados com o Detector de Intrusões Baseado em Séries Temporais (DIBSeT), no trabalho supra citado, demonstraram que a utilização de séries temporais para a detecção de ataques (negação de serviço com ataque SYN e SMURF) apresentaram resultados satisfatórios quanto a identificação de um ataque, além de consumir pouco tempo de processamento.

Do ponto de vista algorítmico, o uso de séries temporais exige uma fase de preenchimento da série apenas no início da computação, seguida de um ajuste do modelo de previsão, o qual pode ser revisto sempre que o erro de predição começar a aumentar. O ajuste do modelo significa determinar quais os parâmetros do modelo ARIMA (número de termos auto-regressivos e de médias móveis, e número de integrações necessárias para tornar a série estacionária). Realizado os ajustes, o preditor possibilita realizar as previsões, ou seja, detectar ataques presentes no tráfego de rede (Dalmazo et al., 2008).

O trabalho de Ferreira e outros (2008) apresentou uma proposta de uso das Transformadas de Wavelets para detecção de anomalias, e classificação dos ataques através de Redes Neurais Artificiais. A transformada de Wavelet é uma técnica matemática com capacidade de realizar a decomposição de funções.

O emprego de Wavelets para a detecção de anomalias requer o uso de uma função que represente o comportamento da rede de forma mais realística possível. Além disso, as métricas utilizadas, tais como banda utilizada, número de conexões ativas, número de pacotes transmitidos/recebidos etc., para identificação do comportamento da

rede, devem ser escolhidas mediante dois fatores importantes: facilidade de uso e independência dos sistemas operacionais em cada nó da rede (Ferreira et al., 2008).

O algoritmo simulado detectou mudanças abruptas no comportamento da rede. Em seguida, uma rede neural, previamente treinada, foi utilizada para classificar os ataques detectados na etapa anterior. Assim, Ferreira e outros (2008) tiveram como proposta demonstrar que é possível utilizar transformadas de wavelet para detectar anomalias na rede.

Dessa forma, a utilização conjunta das transformadas de wavelet com as Redes Neurais Artificiais pode realizar a classificação dos ataques com maior precisão. Ferreira e outros (2008) consideraram que os resultados iniciais foram promissores.

3. REDES NEURAIS

A tentativa inicial de reproduzir o alto desempenho do cérebro humano em tarefas cognitivas extremamente complexas motivou o desenvolvimento dos modelos de Redes Neurais Artificiais (RNA). Tais modelos representam um tipo especial de processamento da informação, que consiste em muitas células primitivas que trabalham em paralelo e estão conectadas através de ligações diretas, cuja principal função é distribuir padrões de ativação, de maneira similar ao mecanismo básico do cérebro humano.

A propriedade primordial das redes neurais é aprender conforme o ambiente onde estão inseridas e assim melhorar seu desempenho. Para que uma rede neural possa aprender, se faz necessário apresentar um conjunto de exemplos à mesma de forma seqüencial e iterativa. Este processo é chamado de treinamento de redes neurais.

Backpropagation é um método simples de treinamento no qual os pesos são atualizados de padrão em padrão, até formar uma época, isto é, uma apresentação completa do conjunto de treinamento inteiro que está sendo processado.

4. REDES NEURAIS APLICADAS À IDS

A capacidade de generalização das Redes Neurais permite, ao sistema de detecção, detectar padrões de ataques e intrusões que não foram apresentados anteriormente à rede, durante a etapa de aprendizagem.

4.1. SOLUÇÃO PROPOSTA

Segundo Haykin (2001), o projeto de uma rede neural utilizando o algoritmo de retropropagação é mais uma arte do que uma ciência, significando que muitos dos numerosos fatores envolvidos no projeto são resultados da experiência particular de

cada um. Entretanto, ainda segundo Haykin (2001), existem métodos que melhoram significativamente o desempenho do algoritmo de retropropagação.

Algumas heurísticas abordadas por Haykin (2001) foram consideradas no presente trabalho, tais como:

- O modo seqüencial da aprendizagem por retropopagação, envolvendo atualização de padrão em padrão, é computacionalmente mais rápida que o modo por lote, especialmente quando o conjunto de dados de treinamento for grande e altamente redundante, como é o caso da base de registros de conexões da competição KDD CUP 99 [9] utilizada neste trabalho;

- Tornar aleatória, ou seja, embaralhar a ordem em que os exemplos são apresentados ao perceptron de múltiplas camadas de uma época para a seguinte. Idealmente, a aleatoriedade garante que os exemplos sucessivos apresentados à rede em uma época raramente pertençam à mesma classe;

- Normalizar as variáveis de entrada da rede neural, sobre todo o conjunto de treinamento, de modo que os valores sejam próximos de zero ou sejam pequenos comparados ao desvio padrão. Isso evita que, por exemplo, as variáveis de entrada apresentem valores positivos de modo consistente, onde os pesos sinápticos de um neurônio da primeira camada oculta apenas cresçam ou decresçam juntos. E como o vetor peso desse neurônio deve mudar de direção, ele só pode fazer isso ziguezagueando seu caminho através da superfície de erro, o que é tipicamente lento e deve ser evitado.

- Treinar a rede neural com o subconjunto carregado até que pelo menos uma das seguintes condições seja alcançada: o erro calculado seja menor ou igual ao erro desejado; o erro se estabilize em determinada faixa de valor; o número máximo de épocas (apresentação de todos os padrões de treinamento para a rede) seja alcançado; ou o tempo máximo configurado em parâmetro seja alcançado.

Os dados utilizados neste trabalho para treinamento e testes das redes neurais, correspondem a subconjuntos da base de dados disponibilizada na competição internacional de mineração de dados KDD Cup [9] – Knowledge Discovery and Data Mining Competition - em 1999 e gerados em projeto de análise de sistemas de detecção de intrusão realizado. Esta base foi gerada através da captura, durante nove semanas, de todos os pacotes TCP/IP em uma rede real.

Diversos tipos de ataques a redes TCP/IP (22 na base de treinamento e 37 na base de teste/validação, sendo que, para esta última, 15 tipos não estavam presentes na base de treinamento sendo, portanto, novos ataques a serem apresentados à RNA),

foram inseridos e identificados na base de dados, juntamente com milhares de registros normais.

A linguagem SQL foi utilizada neste trabalho, com o objetivo de adequar os campos de cada registro de conexão da base de dados do KDDCUP99, para apresentá-los à Rede Neural. Transformação de registros não numéricos em valores numéricos, normalização de valores e criação de tabelas de saída com classes de ataques obtidas aleatoriamente na base de registro de conexões, são exemplos de tarefas realizadas através da linguagem SQL.

O simulador JNNS foi utilizado para simular redes neurais, por ser um software de distribuição livre e possuir os principais recursos utilizados em simuladores comerciais. A Tabela 1 mostra os parâmetros utilizados no simulador de redes neurais.

Tabela 1 – Parâmetros utilizados na RNA

PARÂMETRO	VALOR
Algoritmo de treinamento	Backpropagation
N° nós de fonte na camada de entrada	41
N° neurônios na camada de saída	5
N° de camadas intermediárias	2
N° de neurônios em cada camada intermediária	15
Função de ativação das camadas intermediárias	Logística
Função de ativação da camada de saída	Logística
Taxa de Aprendizagem	0,2
Modelo da Rede Neural	Multilayer *Perceptron* Feed-forward
N° de épocas de treinamento	100
N° de amostras do conjunto de treinamento	30170
N° de amostras do conjunto de teste/validação	35366
Base de Treinamento	Whole KDD
Base de Teste/Validação	Corrected KDD
Erro mínimo	1
Intervalo de valores de saída dos neurônios	de -1,0 a 1,0
Intervalo de pesos aleatórios iniciais	de -1,0 a 1,0

5. DESENVOLVIMENTO E RESULTADOS

Cada saída da rede neural, representando uma classe de ataque/intrusão, foi comparada com a classe real da base de registro de conexões, utilizando o MSExcel e o resultado da comparação de cada registro (classe real da intrusão versus classificação

feita pela RNA) é avaliado e apresentado na Tabela 2, indicando assim a quantidade de previsões corretas e incorretas feitas pela rede neural.

Tabela 2 - Previsões Corretas e Incorretas por Classe.

VALOR DE SAÍDA DA RNA	VALOR REAL DA BASE DE REGISTRO DE CONEXÕES				
	Normal	DoS	Probe	R2L	U2R
Normal	7952	263	129	1708	7
DoS	66	9078	0	0	0
Probe	37	3	339	2	0
R2L	0	0	0	6	0
U2R	8	1	0	143	0

As previsões corretas, na Tabela 2, correspondem à interseção de uma coluna com uma linha de mesma classe, ou seja, quando a classe real de um registro de conexão (coluna) corresponde à classe que a Rede Neural determinou quando esse mesmo registro foi apresentado à entrada da RNA (linha). Por outro lado, as previsões incorretas são todas aquelas onde a classe real de um registro de conexão (coluna) corresponde a qualquer outra classe de saída da Rede Neural (linha) diferente daquela a que o registro de conexão realmente pertence.

A Tabela 3 mostra o percentual de detecção por classe, indicando que a detecção de tráfego normal e a dos ataques dos tipos DoS e probe, tiveram excelente percentual de detecção pela rede neural, mas em contrapartida, ataques das classes R2L e U2R não tiveram percentual de detecção significativo pela RNA. Ainda, pode-se verificar na referida tabela, que o sistema apresentou baixo percentual de falsos positivos, ou seja, apenas 1,38% do tráfego normal, foi classificado erradamente como pertencente a uma das classes de ataques (falsos positivos) e o restante, 98,62% do tráfego normal, foram corretamente classificados como normal pela rede neural.

As duas classes U2R e R2L foram classificadas com percentagens de êxito de detecção relativamente baixas, devido ao reduzido número de amostras dessas duas classes no conjunto de treinamento. Em outras palavras, não foi possível, com a quantidade de ataques das classes U2R e R2L, presentes no conjunto de dados de treinamento, estabelecer a identidade dessas classes no interior da Rede Neural. Logo, quaisquer ataques, sejam ataques já apresentados à rede neural na etapa de treinamento (conhecidos) ou novos ataques (desconhecidos), pertencentes a essas duas classes, poderão não ser corretamente classificados pela RNA. Assim, a RNA classificou a maioria dos ataques de classes U2R e R2L como sendo de classe NORMAL, pois o conjunto de características do registro de conexão, que representava um ataque como pertencente às classes U2R e R2L, não foi suficientemente caracterizado pela RNA durante a etapa de aprendizagem.

Tabela 3 – Percentual Detecção por Classe

CLASSE	% DETECÇÃO
NORMAL	98,62%
DoS	97,14%
PROBE	72,44%
R2L	0,32%
U2R	0,00%

É importante lembrar que o desempenho de reconhecimento da rede treinada, ou seja, sua capacidade de generalização (um termo emprestado da psicologia) está relacionada com a capacidade da rede de estabelecer uma associação correta entre a classe real a que pertence um registro de conexão, com a classe indicada pela RNA, como resultado do reconhecimento feito pela Rede Neural deste registro de conexão. Assim, se uma determinada classe não foi suficientemente caracterizada durante o treinamento da rede, sua capacidade de generalização para todos os ataques já apresentados ou não à RNA na etapa de treinamento, pertencentes a esta classe, fica comprometida.

Na Tabela 4, pode-se fazer uma comparação entre os resultados obtidos neste trabalho (segunda coluna da Tabela) e o resultado obtido pelo vencedor da competição do KDDCUP - Elkan (1999) (terceira coluna da Tabela), o Dr. Bernhard Pfahringer, do Instituto Austríaco de Investigação para a Inteligência Artificial, que utilizou árvores de decisão em seu trabalho. Nessa comparação, percebe-se uma similaridade nos resultados, para os tipos de ataques, cujas classes foram suficientemente caracterizadas durante o treinamento da Rede Neural (Normal, DoS e Probe).

Tabela 4 - Comparação dos Resultados

Tipos de Ataques	Resultados deste Trabalho	Resultados KDDCUP
NORMAL	98.62%	99.50%
DoS	97.14%	97.10%
PROBE	72.44%	83.30%
R2L	0.32%	8.40%
U2R	0.00%	13.20%

Os resultados mostrados na tabela 4 foram obtidos com o uso de uma rede neural com 41 entradas e 30 nós ocultos distribuídos em duas camadas ocultas. Reduzir o tamanho da rede neural significa diminuir a quantidade de cálculos necessários para obter a resposta desejada. De modo a obter um melhor desempenho da Rede Neural, no processo de detecção de intrusão, foram selecionadas apenas as características de maior

relevância para a determinação das classes de ataques presentes nos registros de conexões da base KDD 99.

As características relevantes dos registros de conexões da base KDD 99 [9], foram determinadas por Zincir-Heywood e outros (2005), dentre as 41 características existentes em cada registro, através de uma abordagem baseada no ganho de informação de cada característica para cada classe. Com base na entropia de um recurso, o ganho de informação mede a relevância de um recurso para uma dada classe, ou seja, o seu papel na determinação da classe de um ataque.

Uma dada característica de um registro de conexão é considerada relevante se, em outras palavras, for extremamente útil para uma determinação rigorosa das classes de ataque a que aquele registro de conexão pertence - Bolzida e outros (2005). Em Zincir-Heywood e outros (2005), uma característica relevante é expressa em termos de ganho de informação, que é maior para características mais discriminatórias.

A Figura 1 mostra o máximo de ganho de informações para cada característica do registro de conexão presente na base KDD 99 [9]. Há 10 características com ganho de informação máximo menor do que 0,001 e dessa forma, contribuem muito pouco para a detecção de intrusões. Além disso, as características 20 e 21 (outbound command count for FTP sessions e hot login, respectivamente), não mostram quaisquer ganhos de informação, por isso, eles não têm qualquer relevância para a detecção de intrusões. As características determinadas como relevantes no processo de classificação dos registros de conexões em classes de ataques, ou seja, aquelas que podem discriminar com precisão essas classes, foram utilizadas na rede neural, a qual passou a apresentar 29 entradas, para as 29 características consideradas relevantes por Zincir-Heywood e outros (2005).

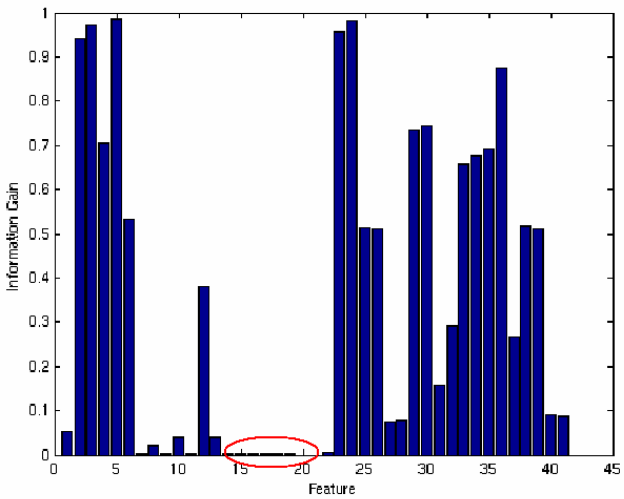

Figura 1 - Máximo Ganho de Informações Zincir-Heywood e outros (2005)

A Tabela 5 mostra os resultados obtidos, utilizando-se como entrada da rede neural apenas os parâmetros de maior relevância para a classificação dos ataques, segundo Zincir-Heywood e outros (2005).

Tabela 5 – Resultados.

Tipos de Ataques	Resultado com todas as Características	Resultado com Características Relevantes	Ganho no Percentual de Detecção com uso das Características Relevantes
Normal	98,62%	99,60%	0,98%
DoS	97,14%	97,07%	-0,07%
Probe	72,44%	68,64%	-3,80%
R2L	0,32%	2,96%	2,64%
U2R	0,00%	0,00%	0,00%

Dessa forma, os resultados mostraram-se satisfatórios, por estarem bem próximos daqueles encontrados anteriormente, quando foram usadas todas as 41 características dos registros de conexões. Além disso, houve uma redução de 24% no tempo de classificação da Rede Neural, devido à diminuição no tamanho da rede.

A rede neural, por apresentar menor número de neurônios de entrada e nas camadas ocultas, foi capaz de realizar a classificação dos ataques de forma mais rápida e eficaz, o que contribui para o uso desta técnica, em trabalhos futuros, na classificação em tempo real dos padrões de ataques em redes de computadores.

6. CONCLUSÕES

Como apresentado neste trabalho, o sistema de detecção de intrusão por anomalias, utilizando Redes Neurais Artificiais, procurou alcançar uma baixa taxa de erros no processo de detecção de padrões de ataques, além de ser capaz de detectar intrusões ainda não conhecidas pela rede, mas pertencentes a uma das classes de ataques caracterizadas durante o treinamento da rede neural.

Logo, a RNA classificou os tipos de ataques presentes na base de registros de conexões KDDCUP99 [9], assim como classificou novos ataques com base nas características apresentadas, para que o analista (elemento humano) pudesse tomar alguma ação contra o novo ataque, por similaridade a algum já conhecido.

Foi utilizada neste trabalho uma rede neural com o número de entradas correspondente a todas as características presentes nas conexões, que são as características básicas, as características por conhecimento especialista e as temporais. A rede foi projetada para classificar os registros de conexões, utilizando uma saída para cada classe, de modo a determinar se o registro da entrada corresponde a uma conexão

normal (livre de ataques), ou aos ataques das classes Denial of Service (DoS), probe, User to Root (U2R) ou Remote to Local (R2L).

Os resultados usando todas as características da base de registros de conexões, foram considerados satisfatórios, quando comparados com os dados obtidos pelo vencedor da competição KDDCUP99 [9]. A baixa percentagem de êxito de detecção para as classes U2R e R2L ocorreram devido ao reduzido número de amostras dessas duas classes no conjunto de treinamento, não tendo sido portanto, suficiente para um bom treinamento da rede neural para essas duas classes de ataques.

Os resultados obtidos com a utilização somente das características consideradas essenciais dos registros de conexões, para a classificação dos ataques, foram satisfatórios, pois se aproximaram bastantes dos resultados obtidos anteriormente, com o uso de todos os campos do registro de conexão. Dessa forma, com o uso de uma rede neural com menos neurônios nas camadas de entrada e ocultas, associados à heurísticas que dão maior eficácia às Redes Neurais na classificação de padrões, é possível utilizar-se dessa técnica no desenvolvimento de ferramentas mais eficazes, baseadas em Redes Neurais, para detecção em tempo real, de ataques em redes de computadores reais.

6.1. PROPOSTAS PARA TRABALHOS FUTUROS

Para o treinamento da rede neural com tráfego e ataques reais, é necessário, entre outros requisitos, o desenvolvimento de uma ferramenta que gere padrões para a rede neural, com base em arquivos de fluxos coletados por alguma ferramenta de captura de pacotes e análise de fluxos. Dessa forma, primeiramente os fluxos serão armazenados em disco, utilizados para gerar um arquivo de dados. A partir dos dados armazenados será possível selecionar intervalos de tempo para a geração do arquivo de padrões na segunda fase. A rede neural poderá então ser treinada a partir das informações fornecidas no arquivo de padrões. Depois de treinada, a rede neural poderá ser agregada a uma ferramenta para monitoramento de fluxos em tempo real.

Além disso, outros desafios surgem no desenvolvimento de um IDS para detecção em tempo real de anomalias de tráfego em redes, tais como: tempo resposta do IDS e dificuldade de treinamento da rede neural com tráfego real.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] BOLZIDA, Yacine et al. Efficient Intrusion Detection Using Principal Component Analysis. Departement RSM GET. France, 2005.

[2] DALMAZO; B. Lopes et al. Detecção de Intrusões baseada em Séries Temporais. Santa Maria – RS. 2008. http://sbseg2008.inf.ufrgs.br/proceedings/data/pdf/st02_01_resumo.pdf.

[3] ELKAN, Charles, Results of the KDD99 Classifier Learning Contest, http://www-cse.ucsd.edu/~elkan/clresults.html. Setembro de 1999.

[4] FERREIRA; W. Tavares et al. Uma Proposta de Utilização da Transformada de Wavelet e Redes Neurais para Detecção de Ataques em Redes Ad Hoc Sem Fio. Mato Grosso. 2008. http://sbseg2008.inf.ufrgs.br/proceedings/data/pdf/st01_02_resumo.pdf.

[5] HAYKIN; Simon. Redes Neurais: Princípios e Prática. Segunda edição. Porto Alegre. Ed. Bookman. 2001.

[6] HEINEN; Milton Roberto, OSÓRIO; Fernando Santos. Autenticação de Assinaturas Utilizando Algoritmos de Aprendizado de Máquina. Universidade do Vale do Rio dos Sinos (UNISINOS). São Leopoldo – RS. 2006.

[7] Homepage of Bernhard Pfahringer, http://www.cs.waikato.ac.nz/~bernhard/.

[8] JUNIOR; A. Candido et al. Uso de Redes Neurais para Detecção de Anomalias em Fluxos de Dados. São José do Rio Preto – SP. 2005. http://www.acmesecurity.org/publicacoes/artigos/acme-artigo-sige-2005-arnalmadr.pdf/view.

[9] KDD Cup 1999 Dataset, UCI KDD repository, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

[10] MAFRA; Paulo M. et al. POLVO-IIDS: Um Sistema de Detecção de Intrusão Inteligente Baseado em Anomalias. Florianópolis – SC. 2008. http://sbseg2008.inf.ufrgs.br/proceedings/data/pdf/st02_02_artigo.pdf.

[11] MAI, J. et al. Is Sampled Data Sufficient for Anomaly Detection? University of California. California, 2006.

[12] PFAHRINGER, B. Winning the KDD99 Classification Cup: Bagged Boosting. Austrian Research Institute. Viena, 2000.

[13] SILVA, L. S. et al. Uma Solução Híbrida para detecção de Anomalias em Redes. Instituto Nacional de Pesquisas Espaciais. São José dos Campos, outubro de 2004.

[14] ZINCIR-HEYWOOD, A.; kayacik, H.; Heywood, M. Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets, Proceedings of the Third Annual Conference on Privacy, Security and Trust, October 2005, St. Andrews, Canada. http://projects.cs.dal.ca/projectx.